Software Requirement Specification

Sri Lanka Institute of Information Technology.

Comprehensive Design and Analysis (CDAP)

15th May 2018

Project Id 18_007

Regular Intake

BSc (Hons) in Information Technology Specialized in Software Engineering

Kodithuwakku K.C

IT14136252

Automated Processing for Data in a Mass Emergency.

Sri Lanka Institute of Information Technology.

Comprehensive Design and Analysis (CDAP)

15th May 2018

Project Id 18_007

Regular Intake

BSc (Hons) in Information Technology Specialized in Software Engineering

Author

Kodithuwakku K.C

IT14136252

Supervisor

Mr. Nuwan Kuruwitaarachchi

External Supervisor

Dr Raj Prasanna

Declaration

I declare that this is my own work and this SRS does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

K.C Kodithuwakku

Table of Contents

Declaration 2

1 Introduction 4

1.1 Purpose 5

1.2 Scope 5

1.2.1 Objectives 6

1.2.2 Goals 6

1.2.3 Benefits 6

1.3 Definitions, Acronyms, and Abbreviations 6

2 Overall Description 8

2.1.1 System interfaces 9

2.1.3 Hardware interfaces 10

2.1.4 Software interfaces 10

2.1.5 Communication interfaces 10

2.1.6 Memory constraints 11

2.1.7 Operations 11

No specific site adaptation is required. 11

2.2 Product functions 11

2.3 User characteristics 14

2.4 Constraints 14

2.4.1 Data Usage policies. 15

2.5 Assumptions and dependencies 15

2.6 Apportioning of requirements 15

3 Specific requirements 16

3.1 External interface requirements 16

3.1.2 Hardware interfaces 19

3.1.3 Software interfaces 19

3.1.4 Communication interfaces 20

3.2 Performance requirements 20

3.3 Design constraints 20

3.4 Software system attributes 21

3.4.1 Availability and Reliability 21

3.4.2 Accuracy 21

3.4.3 Security 21

3.4.4 Maintainability 22

4 Supporting information **24**
4.1 Appendices 24
APPENDIX A – Twitter JSON object 24
APPENDIX B – Literature 26
APPENDIX C – Background 29
APPENDIX D - Visual Tools 31
REFERENCE **32**

1 Introduction

The introduction of the Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references and overview of the SRS. The aim of this document is to gather and analyze and give an in-depth insight of the complete Data Extraction, Categorization and Prioritizing module in detail. Nevertheless, it also concentrates on the capabilities required by stakeholders and their needs while defining high-level features.

1.1 Purpose

The intention of this document is to provide requirements for the Automated Processing for Social Media Data in a Mass Emergency project. All parts are intended primarily for stakeholders and researchers. But it will also be of interest to Disaster Management related personals and government bodies. This will also serve the purpose of guiding future researchers through the process / approach that was followed so that this becomes a reference for their own work.

When maintaining and scaling or adding new features this document will be used as a reference by the responsible parties. Furthermore, it will act as the legal document between SLIIT and the student during research to confirm the deliverables are according to the specified requirements.

1.2 Scope

SRS covers all the requirements i.e. functional, non-functional, system requirements etc. of the project for the development and maintenance of the first stage of the research. The main deliverable of the project is an algorithm that processes social media data automatically as the name implies. Deliverable of the sub component is to develop an algorithm to extract, identify the relevant data and classify.

Algorithm will be developed using extracted data from different types of sources. But it will mainly depend upon popular microblogging platform called Twitter for the output.

This part of the project will provide filtered / pre-processed data for the next component in the main process.

The output would be accessed through the API that would be exposed.

1.2.1 Objectives

- Extract data from social media to identify the relevant data.
- Identify the relevant data from the extracted data.
- Categorize (Classify) the relevant data into meaningful categories.
- Prioritize classified entries.

1.2.2 Goals

- Provide a better tool for disaster management.
- Reduce the disaster response time.
- Filter the relevant entries and provide them to the main flow of the process.

1.2.3 Benefits

- Ability to extract information quickly.
- Identify trends.
- Ability to visualize data.
- Improved response time.
- Open source API for others to develop their own GUIs.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
SRS	Software Requirement Specification.
API	Application Programming Interface.
JSON	JavaScript Object Notation.
AWS	Amazon Web Services.
DMC	Disaster Management Center.
REST	Representational State Transfer.
RDS	Relational Database Service.
S3	Simple Storage Service.
Crowdsourcing	The practice of obtaining information or input into a task or project by enlisting the services of a large number of people, either paid or unpaid, typically via the Internet.
VCS	Version Control System.
Stakeholder	Any person with an interest in the project.
Open-source software	Software for which the code is freely available for use and research
Tweet	Tweet is a buzz word used to refer to a text message which contains maximum 140 characters.

1.4 Overview

The remainder of this document includes three sections and appendices. The second one provides an overview of the system functionality and system interaction with other systems. This chapter also introduces different types of stakeholders and their interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the product. The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used to specify the requirements more precisely for different audiences. The fourth chapter deals with the prioritization of the requirements. It includes a motivation for the chosen prioritization methods and discusses why other alternatives were not chosen. The Appendices in the end of the document include all the background information and additional information regarding the project.

2 Overall Description

The general idea behind the project is to help organizations like DMC, humanitarian organizations, volunteers and services to get situational information more quickly and more accurately. In recent literature (refer APPENDIX B) it was clear that one way of getting information quickly is to analyze the huge throughput of social media activity during an emergency. But there are major problems with this approach.

- Scarcity of entries (Messages, Posts, Comments) with relevant information (Noise factor). A lot of entries show emotions or prayers.

- The ability to analyze these massive amounts of information to be useful, it must be faster. (Low Latency)

The proposed system will address the above problems with computational software solution.

2.1 Product perspective

During the Background research similar systems were found which were using twitter as their main datasource. (Figure 2.1). [1] . The most popular one among those existing system is known as the AIDR (Artificial Intelligence for Disaster Response). It uses few categories to classify the incoming stream of data namely casualties, infrastructure damages and donations. Furthermore it uses Crowdsourcing to train a model during an emergency which leads to response latency.

Proposed tool /system has

- Extended number of categories
- Prioritization of entries in each category.

And it doesnt use crowdsourcing to train a model.to minimize the response latency.

2.1.1 System interfaces

Since the product is not depending on or created on top of any existing systems there will be no system interface requirements.

2.1.2 User interfaces

User interfaces are an optional feature of the subcomponent since the research/project is conducted on underlining functionality. Although for commercial purposes and to improve user interaction. Interfaces would be designed to serve the purpose of data visualization.

The purpose of providing an open API is to provide interested parties to develop their own tools to visualize the data published through the API. Basic user interfaces will be provided to match the needs of user's data visualization requirements. All user interfaces are described in detail in section 3.

2.1.3 Hardware interfaces

Not applicable.

2.1.4 Software interfaces

Twitter Streaming API

Twitter is a free microblogging social networking platform which allows its registered users to publish 140 characters of maximum length messages. To weave tweets into a conversation thread or connect them to a general topic, members can add hashtags to a keyword in their post. The hashtag which acts like a meta tag, is expressed as #keyword. It uses special tags called Hashtags to annotate/ filter text messages (which are called Tweets).

Facebook Graph API

Facebook Graph API will be used to extract data only from official Facebook groups and chat messages which are owned by DMC's and other organizations due to the restrictions on Facebook Policy.

Amazon Web Services

To provide backend infrastructure AWS will be used. It is collection of cloud services which provides the facility of pay as you go. Using AWS, the cost to maintain infrastructure physically can be reduced drastically. These will be required when the final product is ready to be used in real life situations.

2.1.5 Communication interfaces

To connect to internet router/ modem is required. Https will be used as the communication protocol to make it secure.

2.1.6 Memory constraints

Ram of 4GB or higher.

2.1.7 Operations

Operations of the system and subsystems can be carried out at three different levels. There will be parameters associated with each level of operation which define the status of the system and / or control the system.

Observing Level

This is the normal operational mode. It allows a user to access the API through a Web at a fairly high level. Monitoring is also done at this level. It is anticipated that all user categories have access to this level.

Maintenance Level

This allows the system to be maintained in a more efficient way. New releases would be planned depending on this level. Limited access would be given to the system administrators.

Test Level

Sub-component level testing would be carried out in this level. This would also be given restricted access

2.1.8 Site adaptation requirements

No specific site adaptation is required.

2.2 Product functions

- **Extract data from social media to identify the relevant data.**

The system would use APIs mentioned in section 2.1.4 as its main data source. This is first step which other functions would depend on.

- **Identify the relevant data from the extracted data.**

This would extract/ filter only the entries that are relevant to an emergency.

- **Categorize (Classify) the relevant data into meaningful categories.**

This is where the entries that were filtered in the previous step would be categorized into groups with similar content.

- **Prioritize the categorized (classified) entries.**

Response time is a critical factor when it comes to an emergency. Responsible parties must know where to start and where to end in other words how to prioritize events/ actions during an emergency.

- **Allow users to link their Facebook pages to the system to retrieve data.**

For example, if a certain DMC has an official Facebook page then they should be able to connect their page to the system easily to allow the system to retrieve data.

2.3 User characteristics

The application is intended to be used by any personal who is interested in an emergency. Although DMC, Humanitarian Organizations, Victims, General Public and Journalists are the main benefactors of the system.

User does not need to have a special training in other words user doesn't have to be a expert in technology. Anyone with basic knowledge and understanding of domain should be able to operate the system.

2.4 Constraints

- Python shall be the implementation language for the algorithm.
- Since occurring an emergency is natural and unpredictable it is not possible to train an algorithm with live data. Due to this reason dummy data must be use.
- Use NoSQL databases.
- Code base must be open source.
- Data usage should strictly follow the policies mention by the data source providers.
- Git must be used as the VCS for the project.
- Use only open source freely available tools during development process.

- Due to the restrictions in Facebook Graph API. The analysis is limited only to posts / comments that are posted to groups which are owned by the users. In the case of twitter, it doesn't limit to the user's followers any public tweet can be obtained.

2.4.1 Data Usage policies.

It is important to realize that Data Usage Policies are vital when it comes to applications build upon user data. The proposed system has to comply to the terms and agreements provided in the API usage policies by Twitter and Facebook.

Twitter API usage policy.
https://developer.twitter.com/en/developer-terms/agreement-and-policy
Facebook Graph API usage policy.
https://developers.facebook.com/policy

In addition internal data usage policy contains

- Data should not be shared across different users.
- User sensitive data should be encrypted.

2.5 Assumptions and dependencies

Assume that Twitter and Facebook API doen't change frequently..System is highly dependent on the data sources APIs. Since a noticeable change to those APIs could break the system.

2.6 Apportioning of requirements

The implementation of the requirements must follow the same flow of the component process.Essential Requirements should be made with the highest priority according to the given order. Additional requirements are "Nice to have features" which are not necessary to be implemented in the first revision / version 1.

Essential Requirements

- Extract data from social media to identify the relevant data.
- Identify the relevant data from the extracted data.
- Categorize (Classify) the relevant data into meaningful categories.
- Prioritize the previously classified entries.
- User must be able to link their own Facebook Pages to provide data to the system.
- Identify posts with images (visual evidence) and collect images.

Additional Requirements

- Heat Map using the location information embedded in the metadata of the messages (entries)
- Visual analysis tools.- Maps, Graphs etc.

3 Specific requirements

3.1 External interface requirements

3.1.1 User interfaces

Provided user interfaces are only for the purpose of giving the stakeholders a basic idea how the features would be organized. All the user interfaces are bound to change as the requirements change. Going through several iterations of demonstrations / prototypes final interfaces would be designed.

Figure 3.1.1 Categorized data view.

Figure 3.1.2 Charts view

3.1.2 Hardware interfaces

Not Applicable.

3.1.3 Software interfaces

Twitter Streaming API

Twitter has exposed an API for developers/businesses to create applications on top of the public information available in Twitter. It provides and extensive documentation which can be found at https://developer.twitter.com/en/docs. There is some limitation to the API when it come to free usage. Part of the API called Streaming API which will be used in the project as a source of data. For more information about the Twitter API refer to APPENDIX A.

Facebook Graph API

This API is intended to allow developers to easily get data from Facebook application.

AWS

Specifically.

- EC2 instances: to be used as a cloud server.
- Amazon S3: As a storage to store images/ videos and to host web pages.
- Lambda functions: computations to be done when necessary.
- RDS.

3.1.4 Communication interfaces

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying infrastructure.

3.2 Performance requirements

- Time to process a single 140 maximum length tweet should not exceed 20 seconds.
- Time to process a tweet with a single image must not exceed 40 seconds.
- Time to process a Facebook post must not exceed 40 seconds.
- The system is required to support multiple terminals simultaneously. The system should handle reasonable number of users without break or inconsistency.

3.3 Design constraints

- System must be easily scalable.
- Low maintenance cost and high availability is expected.
- RESTful adaptation for backend API.
- JSON should be used for message passing. In between modules.

3.4 Software system attributes

Non-Functional Requirements.

3.4.1 Availability and Reliability

The system is required stay to up and running 24/365 as the emergencies are unpredictable. Although it is not possible to achieve an availability of 100%. Strict measures are necessary to be followed to make sure maximum level of availability. For instance,

- Periodical (Every two days) performance checks
- Server heartbeats
- On demand and Automated restart.
- AWS Logging mechanisms to identify interruptions.

3.4.2 Accuracy

Algorithm must show an accuracy level above 90%.

3.4.3 Security

- External developers who would be using the API as their data source is required to obtain an API key (Token) to authenticate.
- In addition to encrypting usernames and passwords all other sensitive data must be encrypted. Specially if there are personally identifiable information like names, emails, locations.
- Link Facebook pages to the system securely.

3.4.4 Maintainability

- Due to the progressive development of the product it is prone to change time to time with new updates.
- For the ease of tracking and maintaining a good code base version controlling software must be use. Specifically, public GIT repository to make it open source. Other developers would be able to contribute to the project after the first stage of the research is completed.
- Proper coding standard and best practices, easily readable code and proper testing documentation required.

3.5 Functional Requirements

Extract data from social media to identify the relevant data.

- In order to train a model use data from www.disaster.com.
- In a real emergency data sources would be twitter and facebook.
- Relevant data are anything that contains situational information about an ongoing emergency.
- If a stakeholder wishes to connect their facebook pages it is required obtain a token with the consent of the user.

Identify the relevant data from the extracted data.

- Two sets of data required, one for training and one for testing.
- Datasets (corpus/ dictionaries) needs to be labeled (marked relevant and irrelevant).
- Computational Linguistics, probabilistic analysis and natural language processing required for the implementation.

Categorize (Classify) the relevant data into meaningful categories.

- Extended and basic categories needs to be implemented.
- Classification should show a accuracy level of 90%.
- Posts with images should be categorized separately. (not image processing).

Prioritize the previously classified entries.

- For each category entries should be prioritized.

User must be able to link their own Facebook Pages to provide data to the system.

- If a stakeholder wishes to connect their facebook pages it is required obtain a token with the consent of the user.

User must be able to create reports and download results.

- Should support CSV , PDF file formats .
- Should support upto 2MB of file size.
- Charts should be included in the report if any exists.

User should be able to see the trends.

- Generate word clouds.

User should be able to sort the results.

4 Supporting information

4.1 Appendices

APPENDIX A – Twitter JSON object

Figure 4.1 show a sample twitter post, more commonly known as a tweet. Figure 4.2 shows the relevant JSON object for it.

Figure 4.1. Sample Tweet [2]

Figure 4.2 Json Object returned by the Twitter Streaming API
for the sample Tweet Figure 4.1 [2]

Key	**Description**
tweet	The root of the JSON object.
created_at	Timestamp of the post.
id_str	Tweet id.
text	Body of the tweet.
user	Contains details about the user who created the tweet.

Table 4.1 Description for Twitter API streaming JSON object keys

APPENDIX B – Literature

By Ariel Evnine, Andreas Gros and Aude Hofleitner - Facebook Data Science team [6]

On Sunday August 24th, 3:20 a.m Pacific time, an earthquake of magnitude 6.0 occurred in the Bay Area, 3.7 miles (6.0 km) northwest of American Canyon near the West Napa Fault. It was the largest earthquake in the Bay Area since the 1989 Loma Prieta earthquake.

During a crisis, people turn to Facebook to stay connected to their friends and family. They use it to receive social support and keep the people they care about informed on how they are doing.

The map above shows the relative difference in activity on Facebook between the 24th of August, 3:21 a.m. and 3:26 a.m., and the same time period one week earlier.
For visualization, we cluster together nearby cities which showed similar changes in activity. The color represents the percent variation in activity (red: largest activity, yellow: lowest activity). The size represents the area covered by the cluster. The blue cross indicates the location of the epicenter.

We looked at all public posts from people within 300 km from the epicenter during the hour following the earthquake on August 24th. The following word cloud shows the frequency of words used. “Earthquake” comes as the most commonly used word, also very common are “American Canyon”, which is the location where the earthquake occurred. Happening in the middle of the night, the earthquake had a strong effect on people's sleep ("wake", "sleep"). People also express their fear and general feelings and enquire about friends and family.

We see very significant spikes in Facebook activity for people located in a 300 km radius of the earthquake. In the beginning of the night, the activity is very similar on both August 17th and 24th. The difference spikes at 3:21a.m., just following the shake. We notice people staying more active than usual throughout the night. The difference decreases in the early morning (more than two hours after the earthquake) but never to the usual level of activity. In the morning, the number of posts increases above normal number of posts and the additional activity remains for the entire day.

Similarly, we compare the variation in the number of posts in a city to the city's distance from the epicenter. The variation in the number of posts is computed as the ratio between the number of posts in a city within one hour following the earthquake to the number of posts on August 17th at the same time. We ran a linear regression between the distance to the earthquake and the posting variation (in log scale).

APPENDIX C – Background

Social media has played an important role in society today. It is a wired communication medium that shares not only text but also picture and voice information. Millions of people have surfed and socialized through social media every day to find and share information. Through the social media platform, the researchers found that the information that was squeezed after a mass emergency was useful for gaining insight into situational awareness.

In an emergency such as earthquake there can be entries which contain facts related to the extent of the affect, infrastructure damages, casualties, donations available or requested, the kinds of necessities required, for instance food or water. Trustworthiness and reliability of incoming or extracted information is a matter that is yet to be solved. Formal first responders, disaster managers' humanitarian organizations, NGOs, general public, local police, area firefighters are few of the stakeholders who benefit from such information. Moreover, different types of information sought by different stakeholders. For example, humanitarian organizations might be interested in potential donors and the requirements of the victims while first responders and disaster managers are interested in the infrastructure damages and casualties.

To take advantage of this information, government agencies hired teams to analyze and generate reports that are useful for the first responders to make timely, life and death decisions. Reading and analyzing a continuous stream of unstructured text information generated at an increased rate is a tedious and stressful task. An employee may be required to respond to each incoming message in a timely manner. Therefore, processing mass media data in social media requires a new way of processing data that reduces the amount of information that human research.

The solution to the research issues mentioned above is a real-time system that automates data processing in social media and other dedicated media in an emergency. It will provide a tool to visualize trends and important factors related to situations that minimize human effort and save a lot of time. It also can filter, aggregate, summarize, and rank the information extracted from the collected data and focuses on analyzing the emotion of the post, focusing on whether each post can be categorized and prioritized we summarize each post to have a better insight into the situation and verify the accuracy of each post through follow-up comments. This solution will focus entirely on the Natural Disaster Support team, which will analyze the large amount of real-time information generated from social media and take immediate action.

Natural language processing (NLP) technology is used to learn and test models to support the internal operation of the system. The solution will help you to understand the response of the local audience in the event of a disaster. It will also help you assess the extent of the devastation and find people in need during an emergency. Contrary to manual analysis, the proposed system will be more productive in providing insight into disasters. Providing improved situational awareness through rapidly increasing throughput will support important lifesaving decision making and coordination emergency response activities

APPENDIX D - Visual Tools

Word Clouds

Map showing the locations of live generated data.

REFERENCE

[1] Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2015. Processing social media messages in mass emergency: A survey. ACM Comput. Surv. 47, 4, Article 67 (June 2015), 38 pages.DOI: http://dx.doi.org/10.1145/2771588

[2] https://developer.twitter.com/en/docs

[3] https://aws.amazon.com/documentation

[4] https://www.facebook.com/about/privacy/

[5] https://napoleoncat.com/

[6]https://www.facebook.com/notes/facebook-data-science/on-facebook-when-the-earth-shakes/10152488877538859