DESIGN AND IMPLEMENTATION OF DISASTER MANAGEMENT APPLICATION USING REACT-NATIVE

Shivansh Srivastava*1, Rohan Kumar*2

*1,2Computer Science & Engineering, Galgotias University, India.

ABSTRACT

We are building a disaster management app to be used by NGO's[1] and other helping organizations in a time of it can help to report missing people and give information about the stock of supplies available and safe place to stay in time of disaster this app is made on react native nodejs and firebase and TensorFlow for gathering information about the disaster how long it can last and what news government is providing our app could also work offline[3] because in these type of area there is monthly no network once there is a network for a little while this app will fetch all the required information and we can use that offline too and other many other technologies there is no other alternative available on any app store our app will be free of cost and managed by admins and some volunteer which can be changed according to the place hence this app can be a lifesaver in crucial times.

Keywords:- React-native, nodejs, firebase, MongoDB, Heroku, Docker, Tensorflow, AWS.

I.   INTRODUCTION

React Native is a framework used to develop android as well as ios applications with the same code base it is developed and maintained by Facebook. our app is divided into some parts starting with an analysis of the market for any already existing application and what features is missing or what could be added to make it more useful the second step was the content needed for the app because it is a crucial part which attracts people and motivates them to use this application then comes to the main part coding where it is also divided into several subparts machine learning models for gathering data and sending it to backend using natural language processing for chatbots and aging integration to backend the backend was the most difficult part because it utilizes all the ML models and NLP[4] and connection to multiple databases and offline support databases like MongoDB firebase and hosting on services like AWS and Heroku.

App development process

This research paper talks about an application that can be used in times of disaster so it can help people who need help in this part we will give you a glimpse of how the app has been made by talking about it step by step

Analysis:- The research stats form analysis and data collection about the app we are trying to build market analysis shows that there are no other alternatives to this app and the one which is present are either inactive or lack the required feature some of them are paid which we don't think is a good way to start because no one is willing to sign in or pay for just information and help with that in mind[6] we have created our app which includes a lot more useful features.

Content Writing:- This is a crucial part for any application because content attracts people to the tag lines and the easier the information is conveyed to other people the easy it is to use the app the main content is about the missing person,[8] supplies templated and home screen which will keep the app user updated about all the information.

Requirements:- This app required a lot of analysis work and a lot of work on machine learning part as it was responsible for gathering information using IBM Watson which used NLP in the background then we required backend developer and DevOps engineer to deploy everything and maintain it and frontend developer to design in react native and backend developer to connect everything

Coding:-The front-end developer main work is to design given by UI/UX designer and replicate it with animations and connect backend it utilizes everything available let it be machine learning model or NLP for a chatbot or another API [6]

Backend developer task is to combine machine learning and natural language processing in API form and database like firebase and MongoDB and MongoDB realm together and making that available to the frontend developer.

DevOps engineer task is to deploy the machine learning and NLP model to AWS and use IBM Watson[18] to extract data and send it to the backend and deploy the app to the app store and ios store and firebase authentication on firebase and node API to Heroku and for error handling use sentry and Redis for caching.
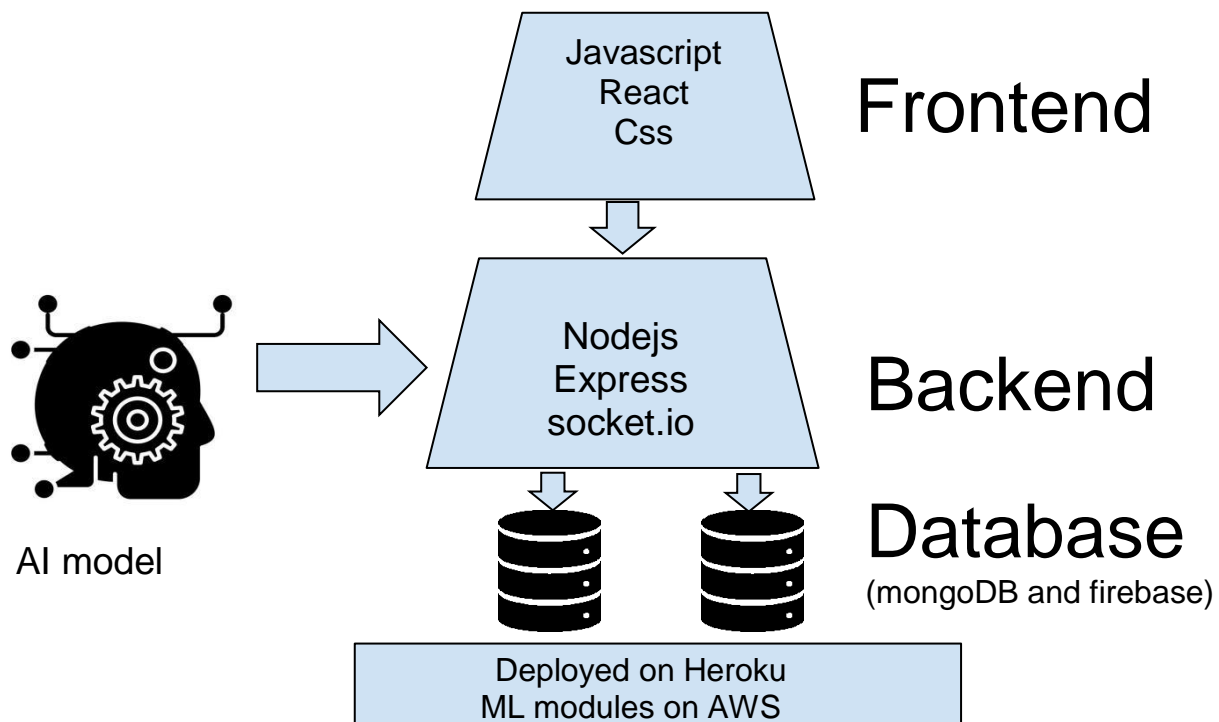

Fig-1: Architecture

II.   LITERATURE REVIEW

The problem arises when there is a natural disaster like flood tsunami earthquake or man-made disaster forest fire etc the issue is there is no service in those areas and getting any information[1] regarding the updated of the situation or which place is safe or where the rescue operation is being held and much other useful information in time of need [9]

In march 2019 there was a cyclone in zimbabwe which killed nearly 1000 residence in the affected area which affected cultivation area houses and everything[11] and just when the people were recovering there was a second cyclone kenneth which was not expected in that area.

There was a trending twitter topic about the australian wildfire[16] which started in 2020 and continued for a very long time and nearly burned 10 million hector of land and kkilled a lot of people but unfortuntly this news was faded as the time went by African drought climate is changing and the temperature is getting high day by day results in drought which hit africa[9] and made a lot of people homeless and with no food or crop the condition was so bad that they have to leave their home and flee to take shelter.

Asia floods has now became a normal problem in asian countries which comes every year and destroy houses crop field and and leave people home less and jobless[6]it can be monitored using data provided by scientists based on sea levels.

Our app gives users all the functionality they need starting with the current update about the situation and missing person report functionality second feature is supplies information and places where rescue operation is being held[3] these being the basic priority of the application there is a lot of other functionality like chatbots which can also be controlled by our volunteer there will be separate volunteer[3] form so anyone could

contribute and register as a volunteer and admin pannel[20] would be there so no one can misbehave admin has all the access of adding or removing users and at last a contribution page so new ideas can be incorporated into the app.

Static and dynamic data can be used to extract feature which is used in feature selection Relief [17], another type of filter, a feature selection algorithm based on the relationship of features and a well-known division solves the problem of two categories by providing weight points. However, relief it has a defect in its deficiency in eliminating unwanted traits, with high-quality traits that are related. will reduce the effect.

PCA is a size-reduction algorithm that helps to turn real-life sequences into low-dimensional subspace to reduce the size of the raw data set. PCA is often used to eliminate unwanted features. So, in order to correct the error of liberty, it is proposed to select a new feature PCA-RELIEF to get more. discriminate the lower set of Android feature.

1)Start: the algorithm will perform the PCA dimension reduction processing on the data set D to select s as the principal component, after that the dimension reduction of data sets is noted as D1.

2)Repeat for n times and output the weight of each feature $Wi=0$, i=1,,2,...s.

$$diff\left(i, M, P\right) = \sum_{j=1}^{k} \frac{\left|M(i) - P\{j\}(i)\right|}{\max(i) - \min(i)}$$

$$diff\left(i, M, Q\right) = \sum_{j=1}^{k} \frac{\left|M(i)\{j\}(i)\right|}{\max(i) - \min(i)} \qquad W_i = W_i - \frac{diff\left(i, M, P\right)}{n \times k} + \frac{diff\left(i, M, Q\right)}{n \times k}$$

3)if Wi>S, add weight to the output set, otherwise delete the feature. Eventually, the algorithm will extract the file Lastly set and arrange from top to bottom; Usually, firstly, PCA[6] will be used to reduce the size of the features and to eliminate reuse. initial features. Thereafter, the break will be used to provide the corresponding weights of the characteristics of each sample.Finally[19], agada attributes with high feature weight are selected to form the final feature section.

III.   RESEARCH METHODOLOGY

The project started with the analysis of the situation that there are very frequent natural disasters and people lose their life or could not get help early so we decided to make an app to try to help[4] those who are in need then there was an analysis of if there is any other alternative available and there was none so we decided to move on with the idea this app features a lot of features but the main feature is offline support[14] and data transfer within seconds so if there is a network just for a little while or a low network speed it can work by using MongoDB Relm for sync with MongoDB offline and online

IV.   GENERAL APPROACH

The general approach is creating design UI/UX design of the app of different screens like admin pannel home screen missing person volunteer registration and many other screens like login register missing person etc.

The frontend developer will then convert that UI idea into the app and connect the backend to make the app dynamic [3]

The backend developer combines everything and utilizes ml model and NLP using IBM Watson and DevOps deploy it to different platforms like Heroku AWS and firebase.

V.   DESIGN IDEA

This application is designed to be optimized so that users can get all the information in a short period of time that means the offline support should be good and the features which we have incorporated into the app should be useful to the people [16]

The app works on react-native so it will be fast and the user experience will be good and the codebase is easy to get used to so another developer can contribute to the project.

VI.   MODULES DESCRIPTION

1. Authentication Module

The user authentication for the application is handled by the authentication module. the authentication module is basically the JWT auth Jwt means "JSON web token"[11] authentication, where the user has a token after login and when it hits any protected screen then token from the AsyncStroage is verified, if the user has a token then, have access to the protected screen otherwise redirect to the home screen with the message of "Access denied". Authentication in our application is built upon nodejs and express js, these are the frameworks to build the backend or REST full APIs.[9] Express is basically used for API development. The Authentication module working is shown in the image below:
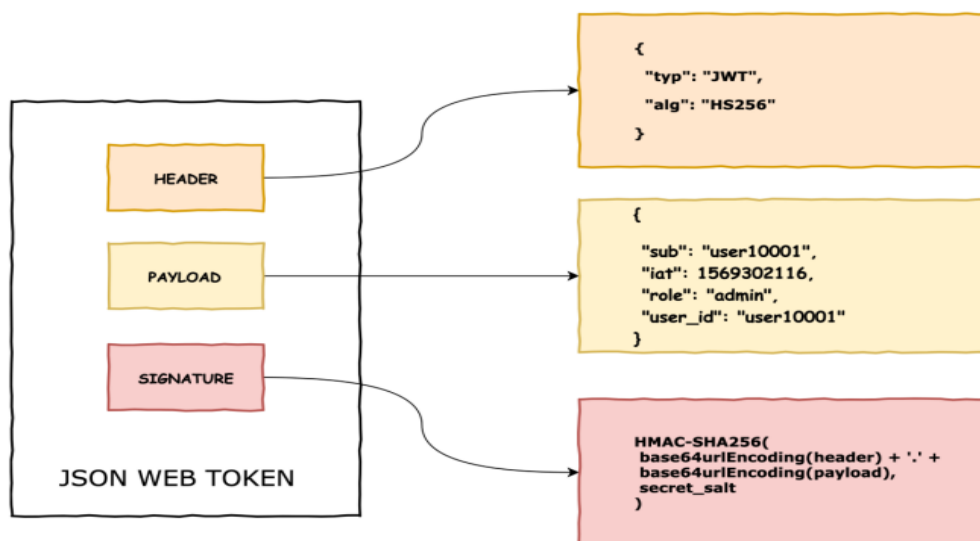
Doing authentication is part of the backend development. we use mongo Db as a database, MongoDB is a non-relational database and no-SQL database,[13] where you can store your data from the client, in the form of collections.

2. Offline Support Module.

Offline support is something when the user has no internet or too low internet connectivity in some areas, so the user can have access to some screens which do not require any data packet to fetch the data from the storage, it will store the data in AsyncStorage[4] in the mobile application bundle, and sync the application according to that data from the storage matter, applications are getting slow because of internet connectivity.it means that without internet connection data cant be fetched from the cloud storage, but can use the existing data from the local storage [9]which is present under the application bundle called Async storage, this is just like the local storage that we have in our typical browsers like chrome and Mozilla firefox. Etc.

Offline Support to Access data-

The app needs to be able to display information even when there is no internet connection, but, in less connected situations,[11] the data needs to be updated as well. This is accomplished by somehow insisting on the data on the cell phone, usually for a longer period of time.

Data Sync:

The app needs to be able to display information even when there is no internet connection, but, in less connected situations[8], the data needs to be updated as well. This is accomplished by somehow insisting on the data on the cell phone, usually for a longer period of time.
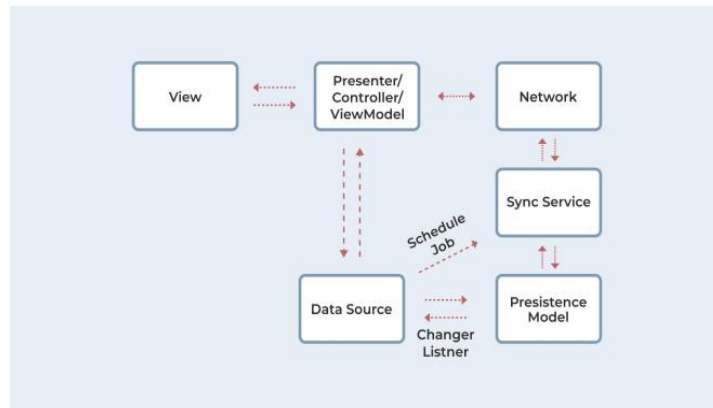
3. Data Collection Module

Data collection modules work is to collect data from the different websites and different sources if someone needs help and he posts it on FB or Instagram or any other NGO[5] website or any other social media platform. it will be redirected to us on the application with the help of IBM Watson Discovery and selenium which is a module in python is used for web scraping[14], it scrapped data from the different small sources.
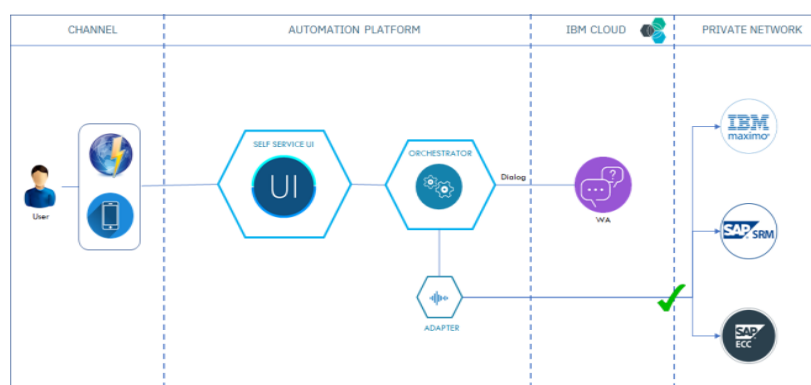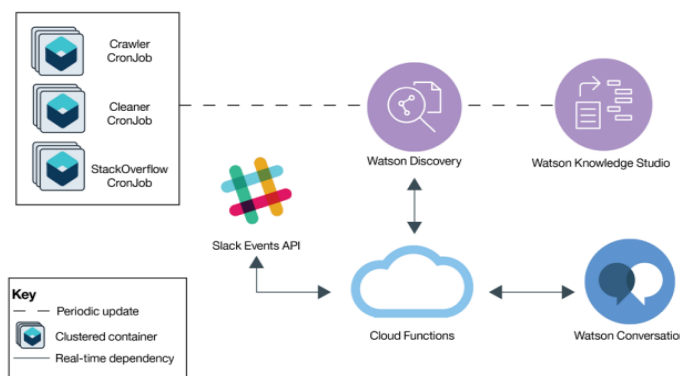

4)Admin panel module

This module is separately dedicated to the maintenance of the app so that the admins can maintain the dataflow of the app[6] and see if there is any issue regarding the misbehave and timely response from the volunteer and look for bugs in the deployment

VII.   RESULTS

This Application will be useful for many Rescue and disaster management NGOs. In the modern era being everything is going digital. this app will be helpful for disaster management. anyone see any disaster on the road or any anywhere, can use this, in this app we provide the helpline number to the rescue teams or NGO's for disaster management, they can contact and help, or user can post about the disaster in the news feeds

section from their profile, so that helping NGOs can go there for help .this process is in the very systematical way two machine learning model is working behind the application by this spamming and fake news can't be shown on the application. also, there is offline support for the application, because if internet connectivity is low in some areas[12]. So, users can access the application with low internet connectivity. This application will be useful for anyone who needs help.
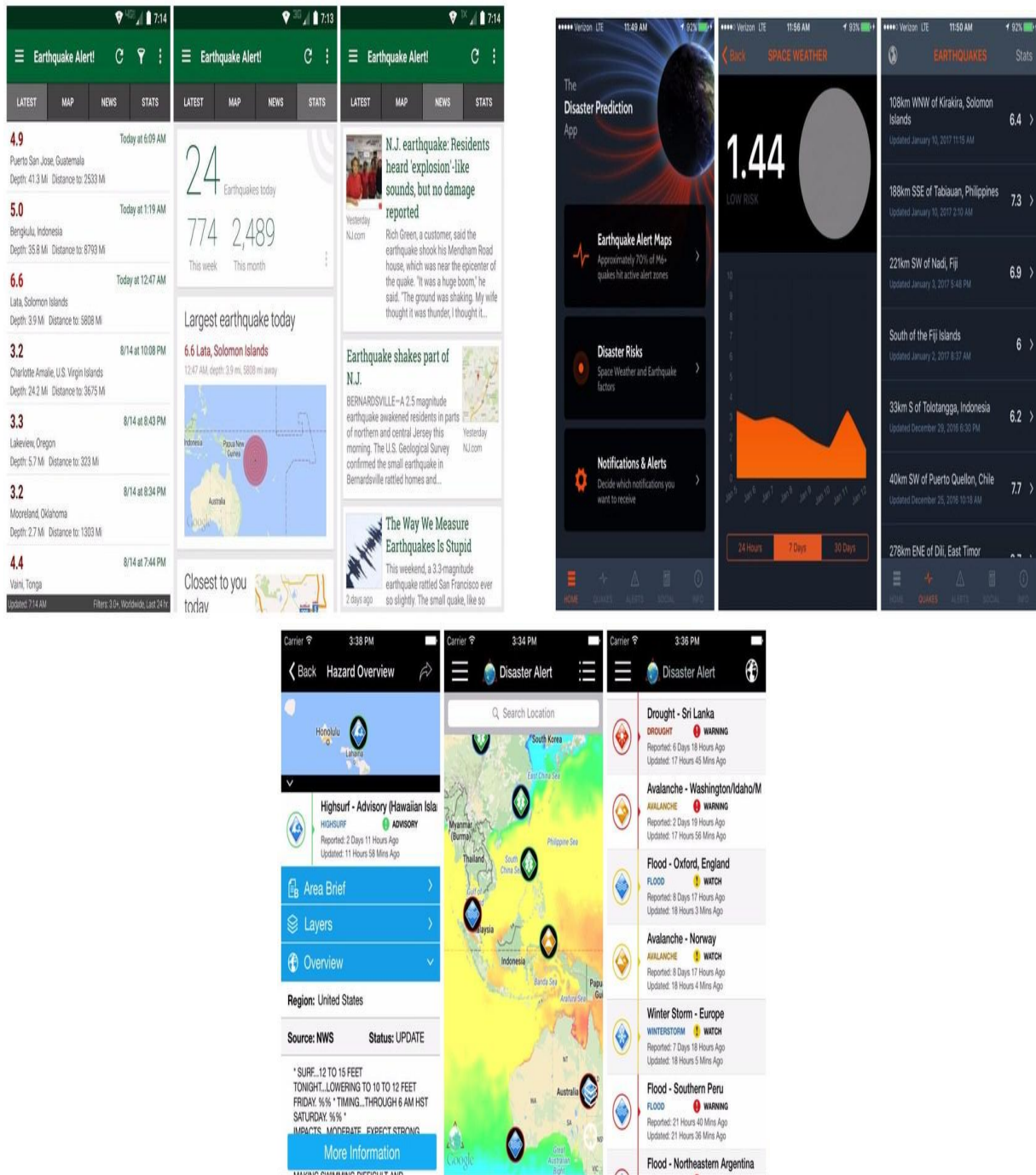
VIII.   FUTURE WORK

What can be more added to this app starting with the scalability it is spread on too many services so there are chances of bugs so to remove that we should use Kubernetes and docker to maintain it and GitHub action for testing automation so there are fewer chances of bugs.

IX.   CONCLUSION

An application is to help in the entire gadget finding issue. It can extract useful information related to the disaster from the websites of NGOs and social media applications. and show it on the application in the form of the notification for the NGOs so that they have knowledge about the disaster at that time, and help is provided as soon as possible, we added a very fast module and querying algorithms so that the speed is proportional to the internet connectivity, also implemented the offline support within the application and spam detection and image recognition for getting rid of fake or false news about the disaster.

This application will be useful for the NGOs who want to help when disaster who need help, there is also some important helpline number so that anyone can just make a call simply and tell the problem.

X.   REFERENCE

[1]   Kung H-Y, Hua J-S, Chen C-T (2016) Drought forecast model and framework using wireless sensor networks. Journal of information science and engineering 22(4):751–769

[2]   Kotta HZ, Rantelobo K, Tena S, Klau G (2020) Wireless sensor network for landslide monitoring in nusa tenggara timur. TELKOMNIKA Telecommun Comput Electron Control 9(1):9–18

[3]   Terzis A, Anandarajah A, Moore K, Wang I (2018) Slip surface localization in wireless sensor networks for landslide prediction. In: Proceedings of the 5th international conference on information processing in sensor networks, ACM, pp 109–116

[4]   Rawat P, Haddad M, Altman E (2018) Towards efficient disaster management: 5 g and device to device communication. In: Information and communication technologies for disaster management (ICT-DM), 2018 2nd international conference on, IEEE, pp 79–87

[5]   Gorcin A, Arslan H (2008) Public safety and emergency case communications: opportunities from the aspect of cognitive radio. In: New frontiers in dynamic spectrum access Networks, 2018. DySPAN 2018. 3rd IEEE symposium on, pp 1–10

[6]   Perera, M.A.L.R. (2019). "Ex Ante Preparedness for Disaster Management: Sahana in Sri Lanka", in Amin, Samia, and Goldstein, Markus, eds., Data Against Natural Disaster (World Bank, http://siteresources.worldbank.org/INTPOVERTY/ Resources/335642-1130251872237/9780821374528.pdf), pp. 273-297.

[7]   de Silva, Chamindra (2016). Sahana Free Open Sources Disaster Management System: Project Overview, Draft 0.9 (Lanka Software Foundation, http://chamindra.googlepages.com/Sahana-Project-Overview-0.9.pdf).

[8]   United Nations (2019), 2019 Global Assessment Report on Disaster Risk Reduction: Risk and Poverty in a Changing Climate; Invest Today for a Safer Tomorrow (ISDR).

[9]   Miguel, A. C., Askew, A. R., Chang, A., Hauck, S., Ladner, R. E., & Riskin, E. A. (2004). Reduced complexity wavelet-based predictive coding of hyperspectral images for FPGA implementation. Data Compression Conference Proceedings, 1, 469–478.

[10]   Appleby-Arnold, S., N. Brockdorff, L. Fallou, and R. Bossu. 2019. Truth, trust, and civic duty: Cultural factors in citizens' perceptions of mobile phone apps and social media in disasters. Journal of Contingencies and Crisis Management 29(4): 293–305.

[11]   Bachmann, D.J., N.K. Jamison, A. Martin, J. Delgado, and N.E. Kman. 2015. Emergency preparedness and disaster response: There's an app for that. Prehospital and Disaster Medicine 30(5): 486–490.

[12]   Bopp, E., J. Douvinet, and D. Serre. 2019. Sorting the good from the bad smartphone applications to alert residents facing disasters—Experiments in France. In Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management (ISCRAM), ed. Z. Franco, J.J. Gonzales, and J.H. Canós, 435–449. València: ISCRAM

[13]   Estuar, M.R.J., M. De Leon, M.D. Santos, J.O. Ilagan, and B.A. May. 2014. Validating UI through UX in the context of a mobile-web crowdsourcing disaster management application. In Proceedings of 2014 International Conference on IT Convergence and Security (ICITCS), 28–30 October 2014, Beijing, China, ed. D. Lee, F. He, and H. Kim, 1–4. Red Hook, NY: IEEE.

[14] Fallou, L., L. Petersen, and F. Roussel. 2019. Efficiently allocating safety tips after an earthquake—Lessons learned from the smartphone application LastQuake. In Proceedings of the 16th International Conference on Information Systems for Crisis Response and Management (ISCRAM), ed. Z. Franco, J.J. Gonzales, and J.H. Canós, 263–275. València: ISCRAM.

[15] Felt, A.P., S. Egelman, and D. Wagner. 2012. I've got 99 problems, but vibration ain't one: A survey of smartphone users' concerns. In Proceedings of the Second ACM Workshop on Security and Privacy in Smartphones and Mobile Device, ed. T. Yu, W. Enck, and X. Jiang, 33–44. Raleigh: ACM.

[16] Gómez, D., A.M. Bernardos, J.I. Portillo, P. Tarrío, and J.R. Casar. 2013. A review on mobile applications for citizen emergency management. In Proceedings of International Conference on Practical Applications of Agents and Multi-Agent Systems–PAAMS, ed. J.M. Corchado, J. Bajo, J. Kozlak, P. Pawlewski, J.M. Molina, V. Julian, R.A. Silveira, R. Unland, and S. Giroux, 190–201. Berlin: Springer.

[17] Hedegaard, S., and J.G. Simonsen. 2013. Extracting usability and user experience information from online user reviews. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems—CHI'13, ed. W.E. Mackay, S. Brewster, and S. Bødker, 2089–2098. New York: Association for Computing Machinery.

[18] Houston, J.B., J. Hawthorne, M.F. Perreault, E.H. Park, M.G. Hode, M.R. Halliwell, S.E.T. McGowen, R. Davis, et al. 2015. Social media and disasters: A functional framework for social media use in disaster planning, response, and research. Disasters 39(1): 1–22.

[19] Korhonen, H., J. Holm, and M. Heikkinen. 2007. Utilizing sound effects in mobile user interface design. In Human-Computer Interaction–INTERACT 2007: Proceedings of the 11th IFIP TC 13 International Conference, Rio de Janeiro, Brazil, 10–14 September 2007, ed. C. Baranauskas, P. Palanque, J. Abascal, S.D.J. Borbosa, 283–296. Berlin: Springer.

[20] Mashhadi, A., A. Mathur, and F. Kawsar. 2014. The myth of subtle notifications. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct publication, ed. A.J. Brush, and A. Friday, 111–114. New York: Association for Computing Machinery.