

TOWARDS A COMMUNITY-BASED MOBILE NEARBY ASSISTANCE SOLUTION: FRAMEWORK AND FIRST PROTOTYPE

Amine Boulemtafes¹, Dalila Hamidouche², Chayma Zatout², Chafika Benzaid² and Nadjib Badache^{1,2}

¹CERIST Research center, Algiers, Algeria

²USTHB University Algiers, Algeria

ABSTRACT

In some cases, being able to appeal to a nearby assistance could be a determining factor possibly saving lives. In fact, a person can at any time encounter a problem or a danger, such as getting lost, falling, having a cardiovascular accident, facing a danger of aggression or theft or having a road accident, which becomes more critical if the person is being far from the eyes. From this perspective, this work aims to study the design of an effective mobile solution focusing on community involvement, cost and reactivity factors -with the help of a set of today's technologies in particular Smartphone, Cloud and sensors- allowing notifying users as well as competent services neighboring a person in need of help.

KEYWORDS

Nearby assistance, mobile, cloud, sensors, smartphone

1. INTRODUCTION

Some time ago, media talked about a motorcycle that recorded a video of a man losing the control of his car after having a stroke; fortunately, the motorcycle could follow the driver while assisting him in order to stop the car. Such situations could happen at any time and anywhere but unfortunately there's not always a motorcycle behind us. In fact, at any time, while being far from the eyes, a person could get lost, fall, have a cardiovascular accident, face a danger of aggression or theft or have a road accident.

Within this perspective, being able to appeal to a nearby assistance in case of a problem or danger even being far from the eyes could be a determining factor possibly saving lives. For this, community cooperation should be promoted especially with the help of today's technologies particularly Smartphone, Cloud and sensors which are becoming more and more part of our daily life, and thus are likely to facilitate the design and implementation of an effective solution for nearby assistance supporting the involvement of everyone and optimizing the reactivity towards persons in need of help.

Taking a look at the existing solutions, various systems for mobile emergency assistance call seem to be available on the market. Generally, such systems either make use of a dedicated device with an emergency button or just take advantage of Smartphone capabilities to do the job. From this, two main categories of such systems can be distinguished:

1.1. Systems based on an external device

Through an external device, a set of systems provide an emergency button used to trigger an alert, either independently or through the user's Smartphone. However, systems relying only on external device are much more expensive than those that make use of Smartphone, probably because of the needed built-in technologies. Examples of such systems include MobileHelp [1] a medical alert system working with local emergency service, relying on cellular/GPS without the need of a Smartphone and offering in extra an automatic fall detection device; Fall detector GPS [2] relying on built-in 3G and GPS to send help text/voice messages with position to chosen contacts; CareTracker SOS Voice [3] a personal alarm system relying on GSM and GPS to email/text message a defined caretaker and allow voice communication and tracking; or SafetyAnchor [4] which combines a Safety button and the user's Smartphone and its capabilities (communication, GPS, ...) to notify through Internet, family/friends contacts, local emergency service, as well as nearby volunteers for community help.

1.2. Systems based on a Smartphone

A more cost-effective solution for emergency assistance call systems is mobile emergency applications, which directly take advantage of Smartphone capabilities. Many examples of such applications can be cited, each having its features and advantages; Allô-Chorta [5] for example is a recent initiative from the Algerian police allowing to trigger an alert through Internet and text message in case of a danger both to the their services and predefined contacts, as well as offering the possibility to report a salient or dangerous event such as a theft or road accident; bSafe [6] is a more advanced application for safety and emergency assistance, keeping the user connected with a safety network through a set of features including tracking to let permitted contacts following you, and alert button triggering an alarm, starting video record and notifying all your friends with live location sharing; Bugle [7] solution, with a different logic, lets the user set an activity plan such as walking or jogging and automatically trigger through text messages and email an alert to predefined emergency contacts if the user doesn't check-in at time from his last activity. SafeTrek [8] came with another idea through a "hold until safe" button, where the alert button should be hold if the user feels scared, once the button is released, the user is asked to enter a code, if he does not, the local police is automatically notified of his location and emergency. Many other applications with other features can also be found such as Red panic button [9] which introduces social network integration, ICEcontact [10] which introduces delayed message allowing to automatically send an alert message after a predefined time if something goes wrong even if your phone dies or loses service, iGoSafely [11] which allows to trigger an alert by plugging in your headphone, iSurvive [12] allowing to use a Smartphone as a flashlight for SOS Morse code for example, and integrate a false deactivation option in case an aggressor tried to cancel the alert, or PanicGuard [13] and Eyewatch [14] which both feature automatic raise of alert if a fall is detected.

From the above quick overview, it is noted that the first category of solutions are interesting in the way that external devices are generally easy to use especially for elderly, while a number of these are also waterproof and thus can be used even under the shower or out in the garden; however, these kind of solutions are quite expensive seemingly due to the need of an external device especially if Smartphone is not involved where external device would be required to do all the job by itself and thus would requires more features. From another hand, the second category is clearly more cost effective and even generally free apart from services like cellular or Internet.

It is also noted that, for both categories, Internet, call and text messages are generally the means used to raise an alert and that only few solutions integrate automatic triggering, but usually only for sudden fall, jerk or stop. In another hand, a number of solutions also from both categories rely

on local emergency number (such as 911), as well as solution provider own responders who unfortunately don't act world widely and might require periodical (like monthly or annually) fees; Also, most of these solutions -except some, like Safety Anchor- don't involve the community and are only limited to contacts and/or competent services which might be too far from the person in need of help.

From this point, this work aims to study the design of a cost-effective nearby assistance solution by its architecture, features and possible enhancements, focusing on reactivity optimization in terms of speed and efficiency through community involvement, triggering and notification means and automation of emergency appeal with sensing and data analysis. To this end, we first define the solution's general architecture, its components and their roles. The different communication aspects are also described. Lastly, the proposed solution brings together a set of novel and inspired features going in favor of the target goals.

The remainder of this paper is organized as follows: The next section presents the proposed solution through the description of a framework for the design of an effective mobile solution for nearby appeal to an aid. The third section presents a first prototype intended to fall detection and notification as well as a couple of notes observed during the implementation of some features. The last section discusses the functionalities and future works related to the described framework and presented prototype, and concludes the paper.

2. NEARBY ASSISTANCE SOLUTION FRAMEWORK

From what has been previously mentioned, it turns out that in order to come up with an affordable solution for an effective reactive assistance, two important factors among others should be addressed namely infrastructure and services costs, and reactivity optimization means.

Opting for the second category of solutions i.e. Smartphone-based seems to be a suitable choice knowing that these smart devices are nowadays widely popularly used even by elderly, are more and more convenient and powerful, and almost not forgettable, which make them an ideal candidate to turn from a smart device to a safety tool; the use of voice or countdown alert triggering as it will be presented later can be imagined as alternatives to waterproof previously described feature found in the first category. Therefore, from an end-user perspective, the solution should be operational with only a Smartphone and a free mobile application; however, some advanced features could need extra equipments like sensors as it will be described later. For services, the solution should be able to select between communication means if different possibilities are offered in order to raise an alert while maintaining at best a tradeoff between cost and reliability; at last, community involvement should be promoted in order to enhance reactivity and as an alternative to paid emergency responders.

2.1. Architecture

Figure 1 presents the general architecture of the proposed solution composed from four main parts namely, the user in need of help, the safety tool (his Smartphone), the central server for coordination, and the potential helpers.

2.1.1. The user in need of help can optionally wear sensors which might be interesting for two reasons: (1) Some types of alerts could be automated according to related sensors data after analysis, such as accelerometer for fall and pulse for heart rate problems; (2) Sensors data might contain important and useful information for competent services reflecting for example the health

state of the person in need of help, which could be an important decision support regarding relevant services to send and equipments or medicines needed on site.

2.1.2. The safety tool or Smartphone, whose primary role is to raise and broadcast alerts, is used through a dedicated application and with the help of its capabilities, to provide different features enhancing the reactivity towards its owner as it will be described later.

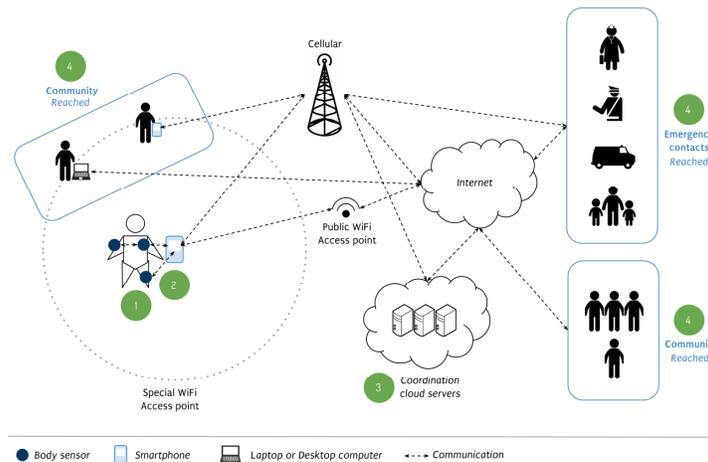


Figure 1. Nearby assistance solution framework

2.1.3. Central server, is the coordinating Cloud of the solution playing a number of roles over Internet network such as managing users, storing sensitive information, dispatching alerts, providing consultation on the web, streaming video and audio, textual communication, etc.

2.1.4. Potential helpers, include competent services such police or firefighters, user's predefined emergency contacts and registered community.

2.2. Communication

Different communication means between user's Smartphone and potential helpers are possible; the framework's scenario of communication proposal is as follows:

For direct connections between user and emergency contacts (family, emergency service, etc.) cellular network is usually used such as assistance through voice call.

For broadcast of alerts and small-medium size data like current position, Internet network if available with enough reliability should be prioritized given its cost-effectiveness; an alert can be sent to important contacts through email, social networks, ...etc. as well as to the central server which will be in charge of dispatching the alert to relevant users according to location for example as it will be described later. In case Internet is not available or not enough reliable, help request text message over cellular network can be used to reach helpers directly such as for important contacts and/or through central server for dispatching.

In order to widen scope and speed up reactivity, a costless technique for broadcasting alerts through Wi-Fi Access Point SSID can be introduced as it will be presented later in this document. Regarding live audio/video sharing, streaming could be performed through Internet if available or cellular 3G/4G video calls if available and accessible for the user but with limited participants.

From another hand, since Internet through cellular network might sometimes be intermittent, or provides low bandwidth and sometimes it is simply expensive, it seems important to optimize Internet-based text-messaging communications, such as for alerts dispatching. To this end, a suitable communication protocol taking into consideration these constraints should be used. MQTT a lightweight messaging protocol with mobile sector focus seems to be a reasonable choice due to its advantages [15]. In fact, it is reported that MQTT, with its characteristics, is ideal in case of constrained environments limitations such as network expensiveness, low bandwidth or unreliability [16]; moreover, MQTT implements a security mechanism ensuring that all messages are transmitted even in the presence of brief disconnections [15].

2.3. Features

Before describing the main features of the proposed solution, we first start by giving a brief description of how the overall solution should work.

When a person is potentially in need of help, an alert with current location is raised and a set of features are triggered or become available depending on his situation such as video streaming; the alert, via available appropriate communication means, is broadcasted to nearby potential helpers including competent services such as police, defined emergency contacts and registered community. Potential helpers if permitted should be able to track the person in need of help in order to reach him, while competent services as well as emergency contacts particularly should meanwhile also be able to interact with him for remote assistance, depending of course on the emergency situation.

Starting from the above overall steps, details of “how it works” are presented below through a set of features description. A part of these going in favor of reactivity optimization are summarized in Table I, while Table II summarizes features related to security.

2.3.1. Different alert types such as for road accident, fall, theft ...etc. can be raised; this can enhance reactivity in terms of who should come to help (police, firefighters ...) and what is expected to do and to find on site.

- Alert could be raised after a quick countdown timer to avoid accidental alerts.
- To deactivate an alert, the application could ask for a secret code before deactivating; this could avoid bad attempts of deactivation such as from an aggressor. If the introduced code is wrong, a fake deactivation can also be used to deceive the aggressor.

2.3.2. Different ways of triggering (such as shaking or long button press), one for each alert type which could enhance speed of triggering and thus reactivity.

2.3.3. Alert type dependant features; i.e., trigger or make available a set of features on the basis of the type of alert such as light and sound signals in case of lost; this could contribute in reactivity speed and simplify the use of the solution.

2.3.4. Storing alerts on central server allowing keeping track of reported assistance appeals and consultation on map.

- Alerts can also be forwarded to central server for storing through potential helpers if this was not done by the person in need of help (after verification to ensure uniqueness of stored alerts) such as in case of network service unavailability.

2.3.5. Involve social networks, through dedicated accounts where alerts can be posted in order to reach more potential helpers even non-registered community and people with Internet access restricted to social networks.

2.3.6. Delayed alerts, where an alert can be stored on the central server to be dispatched at a certain time; this allows to take precaution and ensure that an alert will be triggered in case something goes wrong even in case of out of network service or Smartphone shutdown; example of such situations include going for a walk where an alert will be automatically raised if you don't cancel it, or when feeling discomfort and you don't want to bother anyone since it's a slight malaise but at the same time you are afraid you will not be able to raise an alert if things goes worse.

2.3.7. Broadcast alert through Wi-Fi access point, because sometimes the person that could quickly help is just behind a wall, in the next street or in a near store or house, it's seems therefore interesting to be able to reach very near persons even if they are not connected or the service network is unavailable; this could be done through creating a Wi-Fi access point and integrate a help request into its SSID which could widen the appeal scope to nearby houses, stores and streets; alert received can even be rebroadcasted to neighbors of neighbors through recreating the access point, and also dispatched by forwarding the message via the central server if a receiver has an Internet access.

- In order to avoid false alerts, information provided by SSID should have a special format or be encoded; however, since SSID only authorizes a maximum of 32 characters, information included should be minimalist and techniques to reduce message length could be used such as defining acronyms for possible situations for example, THF for theft and FAL for fall.

-

2.3.8. Integrate data from biosensors in order to ensure alerts' raising and speed-up reactivity by automating assistance appeals with the help of data analysis algorithms such as fall or no movement detection using accelerometer.

2.3.9. Sound & light signals in order to ease localization as well as reach more nearby people such as in case of lost.

2.3.10. Use targeted broadcasting using location, such as nearest competent services; registered community can also be targeted through location request from central server; such feature could be optional for potential helpers giving them the possibility to adjust assistance appeals scope avoiding receiving too far requests for example, if this reduces their willingness to help by the time.

2.3.11. Keep at best a tradeoff between communication speed and cost for sending alerts, such as prioritizing Internet messaging over service network messages.

2.3.12. Offline map for tracking allows faster reactivity of potential helpers for localization and tracking of persons in need of help even with low bandwidth or unavailability of Internet.

2.3.13. Navigation to nearest competent services using offline map itinerary, which might speed up reactivity towards the person.

2.3.14. Remote assistance and information through video, audio and textual communication, instant photos sharing ..., which might enhance reactivity.

Table 1. Summary of reactivity optimization features.

Feature	Description
Different alert types	Provides information about appropriate services to send on site and what is expected to find there
Different alert triggering ways	Eases the use and allows quick triggering
Social networks involvement	Reach more people even with restricted to social networks Internet as well as people not using the application
Wi-Fi access point broadcasting	Reach more nearby potential helpers even those not using the application or without Internet
Biosensors data	Automate alerts and ensure their sending even when it is not possible manually
Sound & light signals	Speed up localization and reach more nearby potential helpers
Offline map	Localization even with low bandwidth or unavailability of Internet
Remote assistance & information	Start assistance before arriving on site or reaching the user

Table 2. Summary of security features.

Feature	Description
Countdown before alert	Prevents accidental alert raising
Code for deactivation	Prevents accidental alert canceling as well as bad attempts such as aggressor wanting to cancel an alert
Fake alert deactivation	Prevents (through deceit) from bad attempts
Access point SSID information format or encoding (user ID, situation, location, date/hour)	Protection against false alerts
Broadcast and tracking permission	Public or private (competent services and emergency contacts)
Store on server information about identity of approaching potential helpers	Protection(through discouragement) against misusing such as a bad person taking the opportunity for attacking a lost person
Store on server information about identity of help appeal receivers	For investigation like interviewing people close of an accident

It should be pointed out that such system should be under the supervision of competent services and registered users should be aware about possible information stored on central server such as their Id in case of approaching or receiving a help appeal, as well as possible communication costs such as for retransmission of help appeals to central server.

3. FIRST PROTOTYPE & OTHER PRELIMINARY TESTS

A fall detection solution's prototype still in its infancy is described showing some of the framework features described in the previous section.

A first mobile application (BFall, see Fig. 2) running in background of the user's Smartphone can detect falls and send accordingly an alert to the user's family and/or medical staff allowing them to locate him and take necessary measures.

Fall detection relays on data collected from two Smartphone built-in sensors namely accelerometer and gyroscope; with the help of a data analysis algorithm, the application continuously check if a fall has been occurred; once detected, the localization mechanism is triggered and a verification process (see Fig. 2) is launched giving the user 5 seconds in order to

cancel the alert in case he is doing well or it is just an accidental alert; if the user reacts positively, the localization mechanism is stopped, otherwise, a communication phase between the user and his emergency contacts is started.

If Internet is available, an alert message containing the user's location is sent through MQTT messaging protocol to emergency contacts; otherwise, the alert is sent as a simple text message (SMS) including GPS coordinates (if available) through cellular network service.

MQTT through publish/subscribe scheme allowed designing a one-to-many alert transmission architecture in order to be able to notify all emergency contacts at the same time including caregiver and family members. From another hand, among the three quality of service modes offered by MQTT, the third one i.e. QoS=2, considered as the safest transfer mode but also the slowest, was used since alert reception is mandatory.

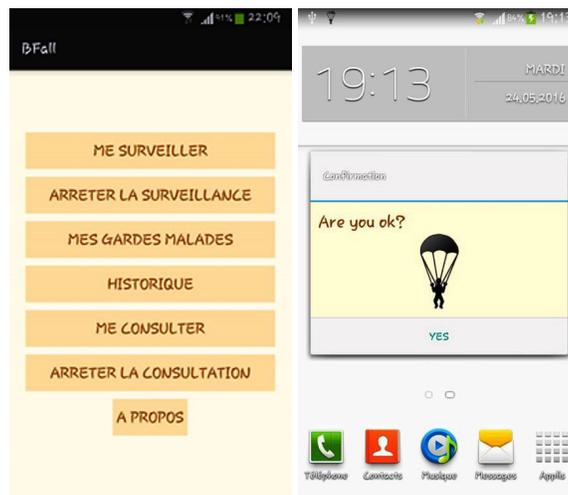


Figure 2. BFall: Main screen (left) and verification process screen (right)

A second application (CFall, see Fig. 3) running in background on the emergency contact Smartphone is used in order to handle alerts.

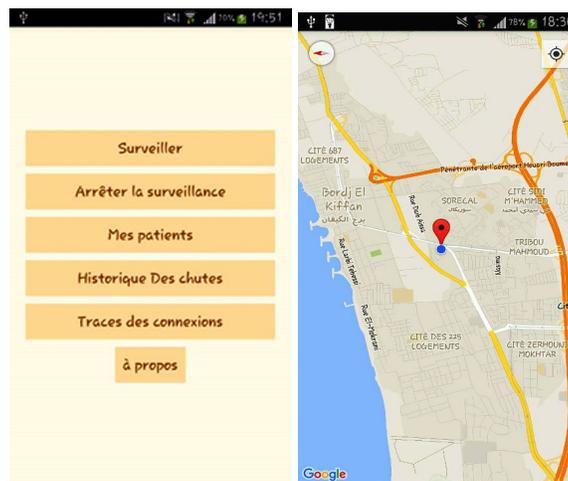


Figure 3. CFall: Main screen (left) and Localization on the map (right)

Once received, alerts are displayed with the identifier of the sender on the screen and through a simple tap a map is opened showing the location of the person in need of help (see Fig. 3).

In case an alert was received through an SMS, the application automatically and seamlessly retrieves location coordinates and displays it on the map (see Fig. 4). History of received falls is stored on a database in order to help doctors in their diagnostics (see Fig. 5).

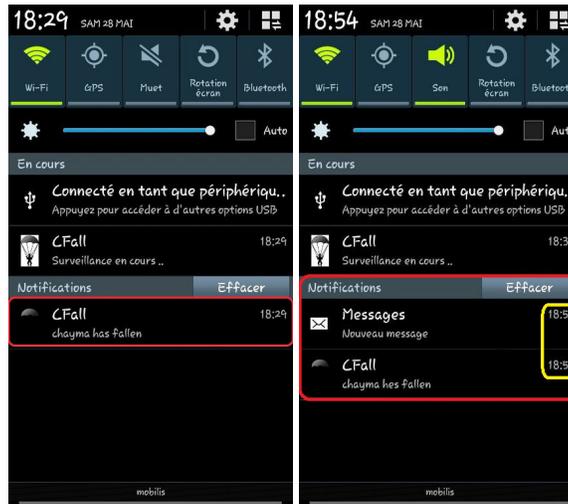


Figure 4. CFall: Alert notification using Internet (left) and SMS (right)

It should be noted that if an MQTT server doesn't receive any message after a certain time interval from a client, this latter is considered as disconnected and will no longer receive notifications; for this, MQTT Keep Alive option was used to ensure the connection between client and server through a regular sending of pings.

At last, in order to allow information exchange between the monitored person and his emergency contacts, a chat feature (see Fig. 5) was also implemented through MQTT messaging protocol.

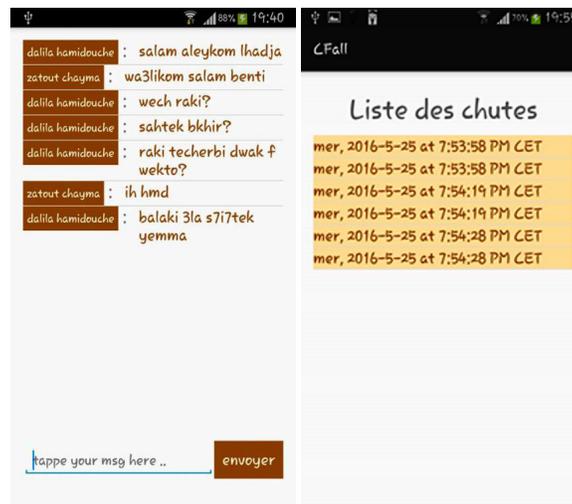


Figure 5. Chat feature (left) and Received falls history (right)

Some other features preliminary tests were also done i.e. alert triggering and notification means (see Fig. 6). From tests done on an old Smartphone running Android 2.3.4, it was noted that some actions -in order to trigger an alert- could be used even when the application was in background such as shaking the Smartphone, while some other actions could be used only when the application was in foreground such as catching volume (up/down) or other Smartphone keys which might limit the number of possibilities regarding triggering means.

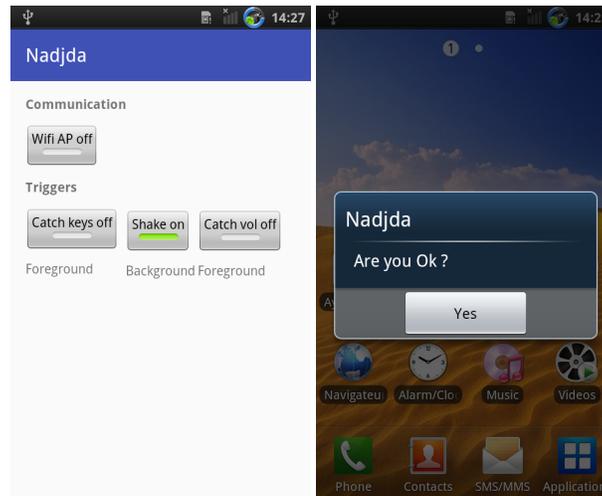


Figure 6. Preliminary tests (left) and confirmation dialog after shaking action while application is in background (right)

4. DISCUSSION AND CONCLUSION

The presented work described an assistance framework through its architecture and features. The framework aims, ultimately, to lead to a mobile nearby assistance solution taking into consideration three important criteria, namely: reactivity for reaching its main goal, affordability for wide public access and security for preventing misuse. To this end, the proposed framework conceptualizes a simple solution relying essentially on user's Smartphone, and combining security mechanisms and a set of interesting features mainly for promoting community cooperation, enhancing alert triggering and notification while ensuring its reception, and supporting reaching the user; the overall with a view to optimize reactivity towards persons in need of help.

Relying only on user's Smartphone for alerts' triggering, unlike the first category of solutions previously described, not only contributes to cost-effectiveness but also generally allows to offer more ways for triggering while enabling customization.

Varying alert types as well as notification and triggering ways including social networks, WiFi broadcast and automatic triggering through delayed alerts or biosensor data analysis allows -unlike a number of existing solutions- to address a large variety of danger and emergency cases and situations while ensuring alert delivery and enhancing reactivity towards people in danger from different ways as previously described like increasing triggering speed or reaching more neighboring community.

Unlike a number of exiting solutions which only relies on competent services such as police or solution provider own responders, the described framework promotes the involvement of community which supports cost-effectiveness as well as reactivity, while a number of security

features were also proposed to mitigate possible danger from potential community's malicious members.

Other presented features, some of which have also already been implemented by existing solutions, such as sound and light alerts or offline map with itinerary were brought together in the proposed framework for an increased optimization of the intended solution.

On the other hand, despite the presented prototype is still in its infancy and requires a lot of work to meet the goals set for the described framework, some of the core features were already implemented and a couple of observations were noted during the study and design.

Indeed, in comparison with the framework, the prototype already implements one-to-many alert sending mechanism to reach multiple receivers, automated alert triggering through Smartphone' sensors data analysis, localization of user on map, auto-switch to text message alerts via cellular network service if Internet is not available and remote assistance through chat feature. However, localization on map feature needs to be enhanced to work offline and allow navigation to nearest competent services while remote assistance should be widen to voice and video.

Regarding messaging communications, and more precisely MQTT compared to HTTPS, it was reported that according to test results of sending and receiving messages through both 3G and Wi-Fi connectivity, MQTT used less battery, was more reliable as well as faster [17], which consolidates and supports the MQTT choice; however, since Keep Alive option should be used in order to maintain connectivity between server and users, cost of such option as well as energy consumption should be more studied. Concerning MQTT server, it was noted that various brokers exist such as HiveMQ, Mosca, RabbitMQ, ActiveMQ or Mosquitto on which the choice fell for the implementation of the prototype; this latter is free and offers suitable quality of service option i.e. QoS=2, as well as SSL for secure access. However, benchmarks within the application context between messaging protocols such as MQTT, AMQP or KAFKA, ... should be considered in future works in order to come out with a more practical and validated choice.

Next prototype versions should, in addition to implement remaining features, incorporate the two current applications into one single application since a user can either be a helper or a person in need of help, especially with the involvement of community.

REFERENCES

- [1] <https://www.mobilehelp.com/> (Last access: 26/01/2017)
- [2] <https://www.medialarm.com.au/shop/fall-detector-gps/> (Last access: 26/01/2017)
- [3] http://ilcaustralia.org.au/products/21019?search_tree=631 (Last access: 26/01/2017)
- [4] <https://safetylabs.org/> (Last access: 26/01/2017)
- [5] <http://www.dgsn.dz/?-Allo-Chorta-> (Last access: 26/01/2017)
- [6] <http://help.getbsafe.com/> (Last access: 26/01/2017)
- [7] <http://www.gobugle.com/> (Last access: 26/01/2017)
- [8] <https://www.safetrekapp.com/> (Last access: 26/01/2017)
- [9] <http://redpanicbutton.com/> (Last access: 26/01/2017)

- [10] <http://icecontact.com/about-us/> (Last access: 26/01/2017)
- [11] <http://www.igosafely.com/> (Last access: 26/01/2017)
- [12] <http://www.isurviverescueapp.com/about-isurvive> (Last access: 26/01/2017)
- [13] https://panicguard.com/?page_id=652 (Last access: 26/01/2017)
- [14] <https://www.eye-watch.in/> (Last access: 26/01/2017)
- [15] J. E. Luzuriaga, M. Perez, P. Boronat, J. C. Cano, C. Calafate and P. Manzoni, "A comparative evaluation of AMQP and MQTT protocols over unstable and mobile networks," 2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, 2015, pp. 931-936.
- [16] J. E. Luzuriaga, J. C. Cano, C. Calafate, P. Manzoni, M. Perez and P. Boronat, "Handling mobility in IoT applications using the MQTT protocol," 2015 Internet Technologies and Applications (ITA), Wrexham, 2015, pp. 245-250.
- [17] Why HTTP is not enough for the Internet of Things – https://www.ibm.com/developerworks/community/blogs/mobileblog/entry/why_http_is_not_enough_for_the_internet_of_things?lang=en (Last access: 26/01/2017)