IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 3, 2020, accepted July 28, 2020, date of publication August 10, 2020, date of current version August 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3015183

SIGPRO: A Real-Time Progressive Notification System Using MQTT Bridges and Topic Hierarchy for Rapid Location of Missing Persons

ANA MARÍA ZAMBRANO V[1], MARCELO ZAMBRANO V[2], EDUARDO LUIS ORTIZ MEJÍA[1], AND XAVIER CALDERÓN H[1]

[1]Departamento de Telecomunicaciones y Redes de Información, Escuela Politécnica Nacional, Quito 170517, Ecuador
[2]Facultad de Ingeniería en Ciencias Aplicadas, Universidad Técnica del Norte, Ibarra 100105, Ecuador

Corresponding author: Ana María Zambrano V (ana.zambrano@epn.edu.ec)

This work was supported by the Escuela Politécnica Nacional through the Project Smart Safe Cities: Un Sistema Genérico en Tiempo Real de Ágil Notificación de Rescate de Personas Desaparecidas Utilizando Comunicación IoT under Grant PII-DETRI-2019-02.

ABSTRACT Currently, one of the main challenges that the world's big cities are facing is the security problem of their citizens and specifically the disappearance of people (even more in Latin America countries). Thus, to address this problem, a software system called SiGPro has been developed, which allows notifications for the early location of missing persons, using some technological tools, being one of the most prominent application of the MQTT protocol frequently used in the Internet of Things (IoT). SiGPro applies a MQTT-Bridge for hierarchical server communication and developing the software system for real-time performance. SiGPro innovates a Progressive Notification Mechanism, not considered in other similar projects and being one of the main contributions of this research; it delivers notifications depending on the disappearance distance in order to not saturate the end user. In addition, SiGPro was deployed in three cities in Ecuador, (being Quito city the MQTT-broker), where more than 140 users participated in each city and in different test scenarios. Time and distances of disappearance solution were obtained, taking into account the total anonymity of the victim. Finally, SigPro has shown that increasing crowdsensing (number of users), reduces the distance at which the missing person has been located.

INDEX TERMS Crowdsensing, notification system, progressive notification mechanism, SiGPro, MQTT.

I. INTRODUCTION

The missing-persons phenomenon is a social problem of vital importance for all governments around the world because it threatens the integrity and life of their citizens, especially vulnerable people such as children, adolescents, and senior citizens. For example, Mexico maintains a registry of 32,277 missing persons, of which 73.7% are men. The situation is alarming since 41% of these people are between 15 and 29 years of age [3]. In Guatemala, the problem has worsened. In 2005, there were 1,456 registered cases of missing persons, while in 2016 there were 3,526 cases, an increase of 242% [12]. The data makes clear that Latin America, as a whole needs an effective solution to reduce the number of missing persons.

The associate editor coordinating the review of this manuscript and approving it for publication was Florin Pop.

This problem also persists in Ecuador, given that the Office of the Attorney General of Ecuador registers approximately 10,000 complaints per year [7] or, in other words, 28 reports per day regarding kidnappings and disappearances. According to Ecuador's Department of State website, 56% of the missing-persons cases involve adults, 16% involve minors, and 56% involve males [8]. These figures demonstrate that the problem also exists in Ecuadorian society, which has a negative impact on it.

This article presents a customizable, real-time, progressive notification system (called SiGPro) focused on the first few minutes when someone goes missing. It is built to integrate into and improve search and rescue operations. It incorporates both the participation of security agents (administrative users) and general community members (community users) in the system. The concept of crowdsensing [6], [17] plays a fundamental role in the solution, and its real-time nature is crucial for sending timely alerts to end-users.

This article contains the following sections: The State-of-the-Art section summarizes prior and ongoing research related to notification systems for missing persons, and compares this solution to its predecessors. Architecture describes the internal structure of the system, including its key components: the forward notification function and the MQTT bridge [19]. Technical Validation describes the system deployment and the set of tests that were carried out to validate the system. Results Analysis tabulates the test data and interprets them. Finally, we present our Conclusions and plans for Future Research.

II. STATE OF THE ART

During the review of state-of-the-art projects, there were not many articles, which dealt with a community-based solution to the missing-persons problem. However, we did find other notification projects using both real-time and delayed communications. Therefore, we used these projects as reference points for our proposal.

We should first mention that this research is a follow-up to our preliminary project, Quito Smart Safe City [16]. The main goal was to raise situational awareness around this vulnerable population, inviting the community to collaborate in searching for and locating missing persons in a city. [16] It used a layered software architecture with three separate modules: Data Module (MySql), the Business Module (Ubuntu Server 16.04 LTS in an AWS cloud - PHP) and a Presentation Module (an Android Mobile Application). We used this scripting language because PHP is compatible with Android, and therefore the application can have widespread distribution. On top of the server rests the MQTT protocol to support all the notifications, and the Apache Web service to publish web-based services. Being designed for a single city, it was not necessary to involve extra elements, nor extra configurations of the MQTT protocol, which is necessary for this new proposal.

However, users experienced various problems with system deployment. First, notifications were broadcast to all users in the city/region (Quito) regardless of their location relative to the disappearance, meaning that users received both failed alerts and repeating notifications about the same case. Second, [16] we didn't design for the need to send notifications between two or more notification systems (inter-city communications). Up until now, each city has a notification system that operates independently from the others. This proposal presented here solves these two issues. The new system implements a progressive notification option (community segmentation) using the MQTT bridge feature of the MQTT broker [2] to manage hierarchical communications between servers.

Another system we examined was the one developed by the University of Alicante in Spain. This system uses a mobile application to locate missing persons in remote places, an ideal solution for mountain climbers who experience accidents in locations without cell phone coverage and who need urgent attention [15]. This solution employs WiFi communications to send an SOS signal that carries data about the geographic location of the missing person. The cost of deploying the system is 600 euros per receiver. This innovative solution implements a notification system that does not require cell phone service. However, the application requires a sophisticated RF receiver with a correspondingly high price tag (beyond the range of the Latin America situation), nor can the system incorporate multimedia data to help locate the victim, something that our proposal supports.

Another interesting solution has been developed in India. It is a facial recognition application that searches a missing-persons database. The software allows users to upload a photo of the target person. An algorithm then compares the photo to others stored in the database in order to find a match. This innovative solution is known as TrackChild [9], and after its first few days of deployment, it helped locate more than 3,000 missing children. Unlike our proposal, TrackChild has not implemented a real-time notification system, nor does it allow community members to participate in search operations during the first few crucial hours after a disappearance.

The last project we reviewed, and perhaps the most attractive one for our purposes—its success marked by its years of operation and high budget—is Amber Alert [5], [14]. Amber Alert began as a project for the search and rescue of kidnapped children in the United States. This project distributes alerts via mass media, including television and radio. The goal is to notify citizens about kidnapping events and thus speed up the recovery of missing children. This project has been adopted by both European and Latin American countries, including Guatemala, Argentina, and Ecuador. Nevertheless, the project hasn't maintained its efficiency, since Amber Alert still relies on an extremely complex logistical protocol prior to emitting any alert. For example, in a recent case involving the ''Emilia'' system (the Amber Alert system in Ecuador) ASFADEC (the Association of Families and Friends of Missing Persons in Ecuador) and other people involved have commented: ''A lot of time is lost. The first few hours, after a kidnapping or disappearance, are crucial.'' Our research focuses on creating a real-time, mass dissemination medium in which all members of a community can respond together so that the missing person can be located and recovered within the first few hours of their disappearance—once user administrators validate their access.

Our new initiative takes into consideration three features not found in the other projects we have reviewed and discussed here. First, we want a real-time community-based collaborative system centered on optimizing event notifications using our Progressive Notification Feature. Second, we need a customizable architecture that can be configured for different cities. Third, our goal is to extend the benefits of the system nationwide by managing it through an MQTT-based communications hierarchy.

III. ARCHITECTURE

The architecture is based on the features of our initial project described previously [16]. This article describes the

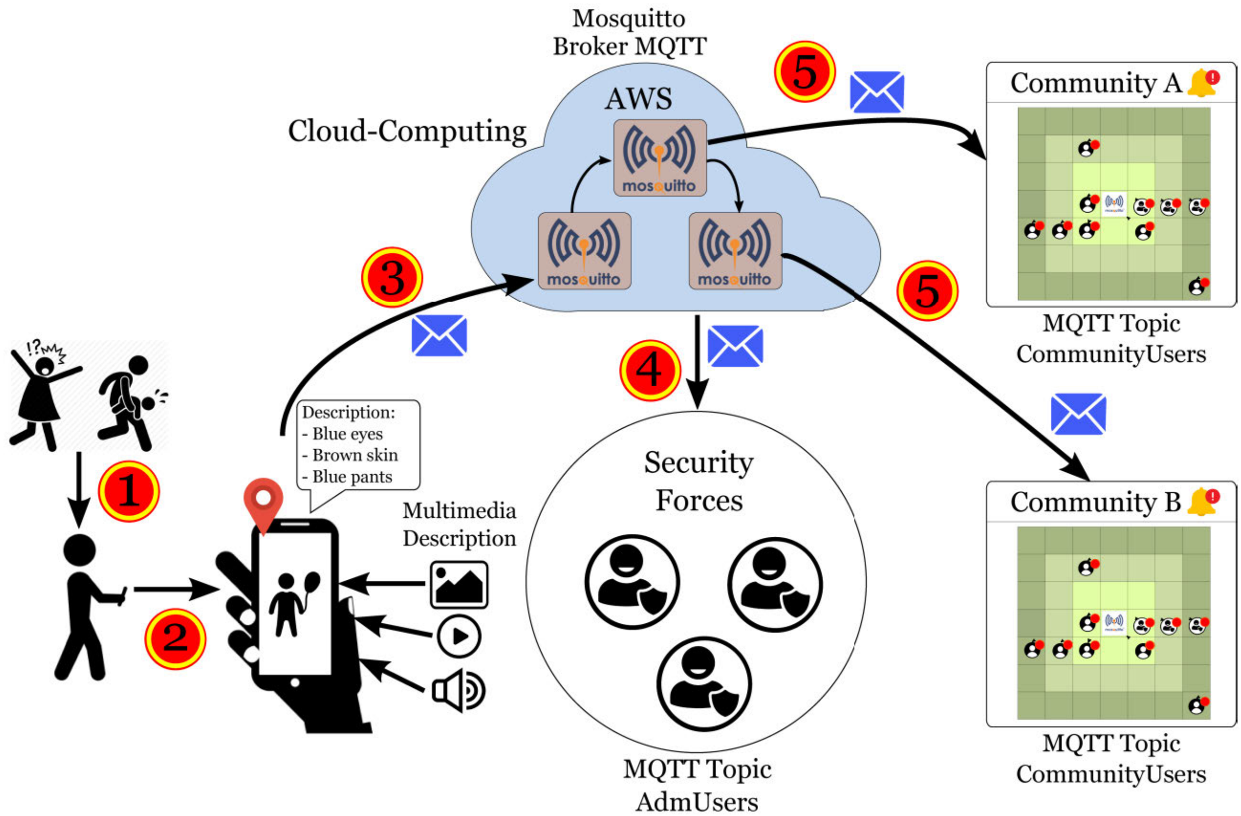

FIGURE 1. Proposed Architecture.

improvements we have added to correct problems encountered in the first project. We defined two classes of users: administrative users and community users. Administrative users include security agents such as Ecuadorian police and criminal investigators. These users validate the cases of missing persons and then inform the community about them. Community users represent citizens. They can register a missing-persons case in the system—after being authorized by an administrative user—and collaborate on solving a case. Participatory actions include appending multimedia evidence (such as images, videos, audio tracks), and/or commenting with text.

Our new proposal incorporates a Forward Notification Feature, which organizes participants into "communities" according to their location in reference to the missing-persons incident (See Fig. 1: Community Users). This feature allows users to notify the closest "community," that is, those who have a real chance of obtaining information regarding the disappearance. If the first community fails to locate the victim after a specified (configurable) time, the alert is escalated to communities further away. This logic avoids bombarding the same user with repeat notifications about the same event. Also, it avoids alarming distant communities with premature warnings. The process repeats itself according to parameters set for the system by the administrative users via a web interface. Once the notification process is finalized, if the incident is not resolved, it is possible to broadcast a notification at the national level that notifies all communities on all registered system servers about the disappearance.

Our proposed system uses a client-server model for communications, along with a REST-style architecture that can identify a wide variety of server resources. Our previous project was hosted [16] on a physical server at the Escuela Politécnica Nacional (since it had enough resources) configured for database services, web hosting, and notifications. This previous design used the cloud service, Amazon Web Services (AWS) [18] since service availability (targeted at 99.9%) was so critical to the goal of finding missing persons [1]. AWS offered flexibility in terms of resource scalability. However, our new project incorporates a hierarchical organization that will eventually cover all of Ecuador, and since we estimate exponential growth in the number of users, the older system would likely saturate the service platform.

We migrated our data from MySQL to MongoDB, a non-relational database. Besides allowing designers to easily index data in any format in a well-organized manner, MongoDB offers geospatial functionality such as GeoJSON [10]. GeoJSON allows users to search registers and documents according to geographic location. It includes spatial filters organized as polygons whose vertices are latitude-longitude pairs. GeoJSON also allows users to add time filters such as intervals between dates. These filters facilitate searches of registers and documents located in specific geographic areas over specified intervals of time. Incorporating these features has allowed us to optimize the analysis and divide the city into "communities" in a flexible way.

The server includes a web interface that uses the PHP, TML, and JavaScript programming languages. The interfaces allow us to configure parameters for the Progressive Notification Feature for different regions (cities). In addition, the interface allows us to visualize and organize both the users and registered cases of missing persons. The optimal configuration of these parameters will define whether we can successfully broadcast successive alerts to corresponding communities.

Our client is a mobile application based on the Android Studio platform. Along with this IDE and the JAVA language, it is possible to access the sensors of the smartphone, such as GPS, in order to capture geographic location in real-time. In addition, we have implemented an MQTT server on the client that, together with MQTT communications (MQTT topics), manages the publication of and subscription to alert messages. To support these features, we are using the Eclipse Paho Android Service library [4] to establish network connections with remote MQTT servers.

Together, this set of components and technologies comprises the SiGPro notification system, whose communications and Progressive Notification Feature are detailed in the next section.

IV. PROGRESSIVE NOTIFICATION FEATURE

The main goal of the Progressive notification system is to alert users who are close to events with an initial notification so that they can transmit any relevant information to authorities. Second, the system needs to avoid saturating any single user with multiple notifications about the same case.

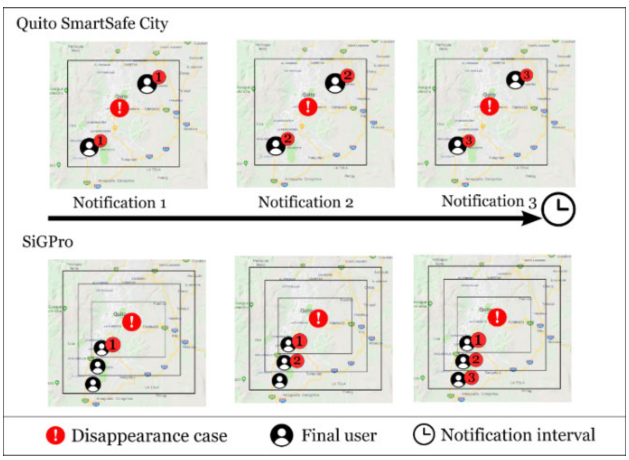

FIGURE 2. Progressive Notification System.

We should mention that real-time performance is essential to fulfilling the application's goal. Although [16] it has this characteristic, this solution experienced problems maintaining an adequate number of users in the system —a sufficient level of crowdsensing—that affected the response time to missing-persons events. Fig. 2 compares the prior solution to the updated one. The data clearly shows the differences between the two notification operations. The SiGPro section describes how users are organized within the city by area (communities) depending on their distance from an event—which represents their capacity to help solve a case. The system slots three notifications at different (configurable) times. In the project [16], the same user receives the same notification multiple times. However, SiGPro notifications avoid overwhelming any single user in the system. In the new version, an administrative user can modify the system's notification configuration through the web application, thus creating a customizable system adaptable to each region.

Our idea is to subdivide the geographic area of Ecuador into zones of configurable size, which correspond to the "communities" referred to in this article. These community squares, unlike the radial notifications of our previous system, will improve the communications server processing. To identify the users in the zone closest to an event, [16], the system reviews the list of registered users and identifies their correct geographic location, for each notification and each update to a missing-persons case. In the current version, we have designated zones that represent geographic areas that also represent MQTT protocol topics to which end users (app clients) must subscribe. From the server, the MongoDB GeoJSON speeds up and supplements subscription, reducing server processing, since each user is completely identified and on standby to receive new notifications within a defined zone.

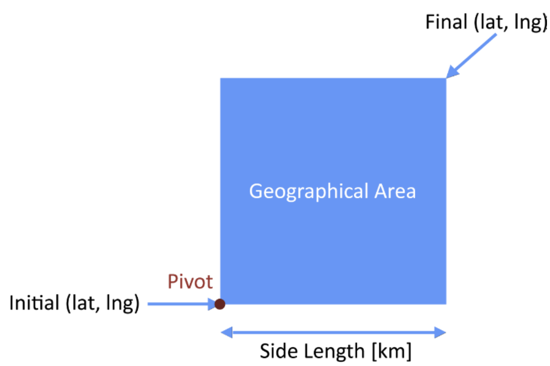

FIGURE 3. Configuration parameters for zones.

A community, or zone, is identified by two geographic points based on its latitude and longitude. In addition, the system records the width of the zone (longitude of side) expressed in kilometers, as recorded in Fig. 3. We denominated the whole set of zones as the Grid. We should stress that the groupings of various zones constitute notification areas, and the whole group of areas constitute the Grid (See Fig. 4). The size of each zone, as well as the quantity of zones within the Grid, is configurable through the web interface.

In reference to Fig. 4, "Notification Area 1," corresponds to the nine zones surrounding the location of a missing-persons case. Users subscribed to these zones will receive the initial notification on the case. After a specified (configurable) time, if the missing person is not found, the system

FIGURE 4. Determining the Notification Areas.

automatically generates "Notification Area 2," composed of all zones that surround "Notification Area 1," and sends a warning notification to all users subscribed in this new area.

This process repeats until it covers all areas configured in the system. We call this cycle the Progressive Notification System. The Progressive Notification System applies to community users, whereas administrative users are always notified regardless of their geographical location.

To generate the Grid, the system needs one geographic point on the map (Pivot 1), the value of the zone side, and the number of rows and columns in the zone. To make the calculation easy, we assign the value of 1 kilometer to the side of a zone. According to [11], 1 equals 30.92 meters in the decimal-based, geographic coordinate system. Thus, 1 kilometer equals 33 seconds, also equal to 0.008983757 decimal degrees. These data are important for generating the zones. Following this logic, Fig. 5 identifies the initial coordinates (Pivot 1) of latitude (x) and longitude (y). The pivots are used as references for generating a row of zones, where the quantity of zones is defined by the number of columns (pivots) specified by the administrative user. Each pivot is used to generate the next pivot (See Fig. 5). This process repeats until all rows configured for the system are accounted for.

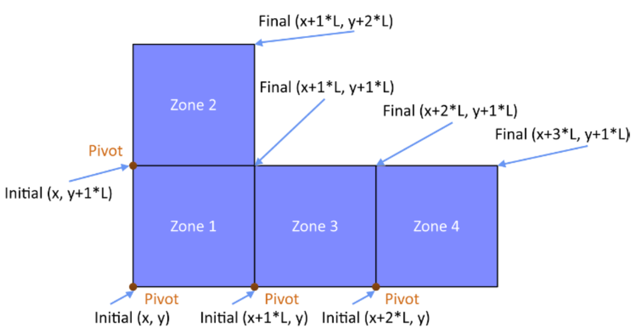

FIGURE 5. Pivot and zones generation.

As observed in Fig. 5, the vertical process for generating pivots only requires the initial coordinate since the final pivot coordinate is automatically derived from the configuration information. Equation 1 and Table 1 reference the main coordinate, which allows us to identify the initial coordinate of any pivot point in the Grid.

TABLE 1. Symbols used in equations.

Symbol	Description
X_0	Latitude of the reference coordinate.
Y_o	Longitude of the reference coordinate.
n	Number of horizontal or vertical rows/columns, as appropriate.
L	Longitude (distance) of the zone.

$$(X_0, Y_0 + (n-1) * L) \tag{1}$$

To determine the initial coordinate in terms of the horizontal process, we came up with Equation 2. Likewise, we can derive the final coordinate by using Equation 3.

$$(X_0 + (n-1) * L, Y_0) \tag{2}$$

$$(X_0 + n * L, Y_0 + 1 * L) \tag{3}$$

Notification Area 3	Notification Area 2	Notification Area 1
(x - 3*d, y + 3*d)	(x - 2*d, y + 2*d)	(x - 1*d, y + 1*d)
(x - 3*d, y + 2*d)	(x - 2*d, y + 1*d)	(x - 1*d, y + 0*d)
(x - 3*d, y + 1*d)	(x - 2*d, y + 0*d)	(x - 1*d, y - 1*d)
(x - 3*d, y + 0*d)	(x - 2*d, y - 1*d)	(x - 0*d, y + 1*d)
(x - 3*d, y - 1*d)	(x - 2*d, y - 2*d)	(x - 0*d, y + 0*d)
(x - 3*d, y - 2*d)	(x - 1*d, y + 2*d)	(x - 0*d, y - 1*d)
(x - 3*d, y - 3*d)	(x - 1*d, y - 2*d)	(x + 1*d, y + 1*d)
(x - 2*d, y + 3*d)	(x + 0*d, y + 2*d)	(x + 1*d, y + 0*d)
(x - 2*d, y - 3*d)	(x + 0*d, y - 2*d)	(x + 1*d, y - 1*d)
(x - 1*d, y + 3*d)	(x + 1*d, y + 2*d)	
(x - 1*d, y - 3*d)	(x + 1*d, y - 2*d)	
(x + 0*d, y + 3*d)	(x + 2*d, y + 2*d)	
(x + 0*d, y - 3*d)	(x + 2*d, y + 1*d)	
(x + 1*d, y + 3*d)	(x + 2*d, y + 0*d)	
(x + 1*d, y - 3*d)	(x + 2*d, y - 1*d)	
(x + 2*d, y + 3*d)	(x + 2*d, y - 2*d)	
(x + 2*d, y - 3*d)		
(x + 3*d, y + 3*d)		
(x + 3*d, y + 2*d)		
(x + 3*d, y + 1*d)		
(x + 3*d, y + 0*d)		
(x + 3*d, y - 1*d)		
(x + 3*d, y - 2*d)		
(x + 3*d, y - 3*d)		

FIGURE 6. Ordered pairs algorithm for defining zones.

It is important to recall that the longitude of the side of zone (L) is, by default, 1 km. But if another value is specified, for example, 10 km, the corresponding value to use in calculations will be 10∗0.008983757. Given this adjustment, Fig. 6 shows the calculated notification areas. For example, Notification Area 1 corresponds to the zone in which the report and the zones that correspond to the sum of 1∗L for the latitude and longitude of the referential geographic location

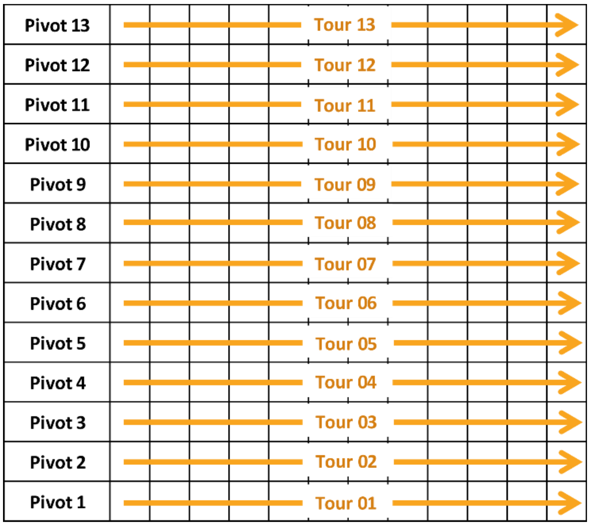

FIGURE 7. Grid generation.

were generated. To generate Notification Area 2, we calculate the sum 2∗L at the aforementioned coordinate. This process repeats until all the notification areas defined for the system are covered.

The algorithm creates a grid (See Fig. 7), that is, it obtains the initial and final coordinates for each zone. This data is sent to the communications server to be stored in the MongoDB database and then generates topics for use by the MQTT server.

Each of the zones generated is assigned a unique identification number that is used as an MQTT topic. The end-users, i.e., clients, subscribe to these topics in order to receive future notifications automatically. We should mention that the generation of notification areas is dynamic. Fig. 8 shows the list of ordered pair equations that identify the final geographic location of each notification area. For example, "Notification Area 1" has nine geographic locations.

(x-2*d, y+2*d)	(x-1*d, y+2*d)	(x+0*d, y+2*d)	(x+1*d, y+2*d)	(x+2*d, y+2*d)
(x-2*d, y+1*d)	(x-1*d, y+1*d)	(x+0*d, y+1*d)	(x+1*d, y+1*d)	(x+2*d, y+1*d)
(x-2*d, y+0*d)	(x-1*d, y+0*d)	(x+0*d, y+0*d)	(x+1*d, y+0*d)	(x+2*d, y+0*d)
(x-2*d, y-1*d)	(x-1*d, y-1*d)	(x+0*d, y-1*d)	(x+1*d, y-1*d)	(x+2*d, y-1*d)
(x-2*d, y-2*d)	(x-1*d, y-2*d)	(x+0*d, y-2*d)	(x+1*d, y-2*d)	(x+2*d, y-2*d)

FIGURE 8. Identification of notification areas per zone.

The final result gives us the unique identifications for each zone in each notification area. With this functionality, we can make the Progressive Notification System adaptable to any region.

V. MQTT BRIDGES AND QUEUE MANAGEMENT

Message Queue Telemetry Transport (MQTT) is an Internet of Things (IoT) and Machine to Machine (M2M) communications protocol for publish-subscribe messaging with very light loads. MQTT carefully manages battery and bandwidth resources on equipment, which makes it ideal for mobile applications such as the one implemented in this project.

Standard MQTT uses TCP/IP port 1883 by default and port 8883 for SSL encryption. MQTT was developed in 1999, when it was applied to many large, industrial projects. MQTT version v3.1.1 was used for our project [16] in 2017, while our current project, SiGPro, uses the latest version of the standard (v5.0). Both projects are configured with the highest level of QoS (QoS 2.0 – received only once). The hierarchy of topics uses hidden topic prefixes, which gives it an extra level of security. We used IDE Eclipse Paho, since it already has an integrated MQTT client. We took an Object-Oriented Programming approach using Java for the popular Eclipse-Mosquitto broker. We verified notification reception under 1 second for both WiFi and 4G in our proposed system, which meets our requirement for real-time 1-to-1 and 1-to-many connections.

Given our goal to expand our project to other cities (servers), we incorporated MQTT bridges for message sharing between real-time systems. Fig. 9 represents a hierarchy for two MQTT servers with a corresponding MQTT Bridge hosted on an AWS server in Quito. This system will share messages depending on input from the mosquitto.conf file of the broker. Each independent broker becomes an MQTT client capable of subscribing and publishing on generated topics. Important configuration parameters include: the broker's address and port (Address keyword), client names (Connection keyword), subscription prefixes and topics, quality of service (Q0 – Q1 or Q2) and communications directionality. Each topic must be configured with a direction (out, in, or both). The MQTT broker's mission is to correctly assign the messages using the topic hierarchy generated by the Progressive Notification Feature and its MQTT clients.

Depending on the requirements, the system can be configured for multiple remote brokers, leaving the system with virtually uninterrupted communications in the case in which one component fails. Thus, our system achieves the scalability over time and distance that enables it to cover any region in the country.

VI. RESULTS ANALYSIS

The design and development of the SiGPro Notification System have received support from the National Agency to Investigate Crimes Against Life, Violent Deaths, Missing Persons, Extortion, and Kidnappings (DINASED). This agency is charged with searching for and locating missing persons in Ecuador, and worked with us during the requirements definition phase of our current project.

The system was deployed in three Ecuadorian cities. Each one implemented the Progressive Notifications System with a configuration to meet its specific physical and logical characteristics, and each system functioned independently. For example, Quito, the main city that housed the MQTT-Bridge, was organized into six notification areas with a time interval of 30 minutes and with zones whose longitude of side

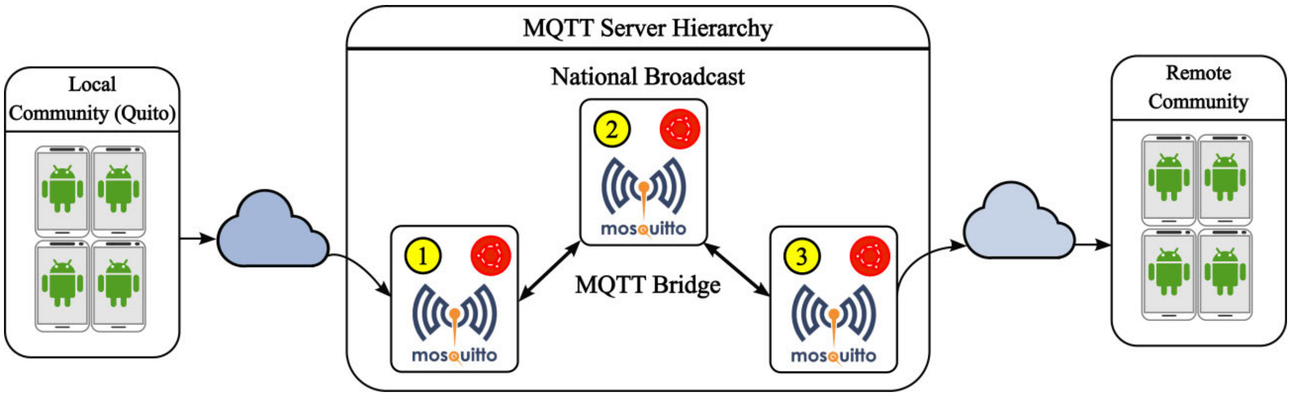

FIGURE 9. MQTT Server Hierarchy – National Notification.

equaled 10 km. This configuration was based on an analysis of our previous project [16], in which we found that a person (kidnapper) can leave Quito's city limits within 3 hours.

We ran a total of 19 tests (simulations of missing-persons cases), maintaining strict confidentiality regarding hints and characteristics that would reveal the missing person in the main city. More than 140 people participated in each test. Various students played the roles of community users and administrative users and, with their input, became researchers in this project.

First, we measured the percent improvement between the previous iteration of our project [16] and this version. Our prior project [16] lacked precision; it sent a single user multiple notifications about the same event. Obviously, the current system should eliminate these multiple notifications, otherwise, users will opt-out of the system and reduce the crowdsensing effect; fewer sensors implies a higher probability of error. To evaluate the percent of repeat notifications that a user receives, we tested two notification systems in the same test scenario. As shown in Fig. 10, SiGPro, on average, improves the previous notification process by 85%.

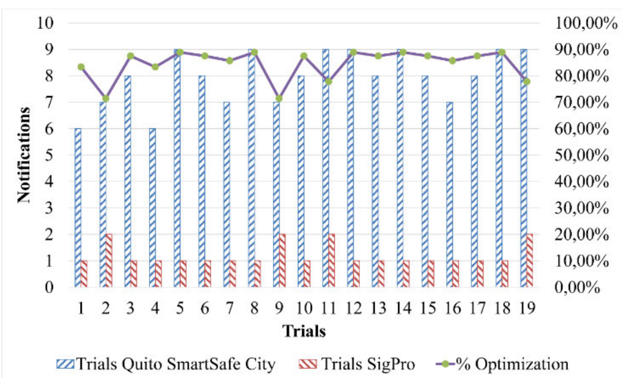

FIGURE 10. Percent improvement in the Notification System.

These tests show that any single user will receive a maximum of two notifications per missing-persons case. The "two" instances are due to the possibility that subscribed users may be in motion and move from one notification area to another.

TABLE 2. Distance from a solved missing persons case.

Num. Users	Distance sol. (in km)	Nationwide alert	Sol. outside province
20	109	Yes	Yes
40	105	Yes	Yes
60	75	No	No
80	82	Yes	No
100	62	No	No
120	56	No	No
140	51	No	No

Crowdsensing is a basic characteristic of this project. It represents the number of active users in the system. Table 2 shows the when the number of users increases, the distance from the solution decreases considerably. The goal is to reduce the time needed for search and rescue to a minimum. In addition, the data shows that the number of users becomes insignificant if the distance from the solution surpasses 100 km in longitude, to the point where we no longer find the missing person in the target region. Therefore, during these tests, we decided at this point to issue a nationwide alert, and cases were solved in adjoining cities using an MQTT bridge.

By analogy, Fig. 11 shows the percent improvement gained when one introduces more users in the system in a progressive manner. We compared the results from a test with 20 users to a test that started with 40 users and incremented the number up to 140 users. On average, the more users were subscribed, the possibility of solving a case increased by 3.8%.

Fig. 12 indicates the time to solution of a case. First, the data shows the initial time to solution (the time to locate a missing person) was 45 minutes, measured from the time the first alert to citizens was issued. Since the crowdsensing saturation remained low, the missing person was located in another city. In contrast, when there were 140 subscribed

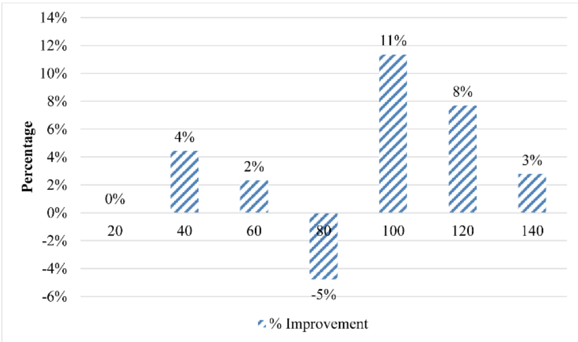

FIGURE 11. Percent improvement according to number of users.

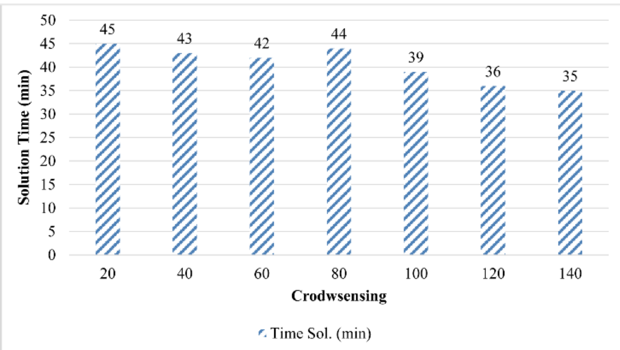

FIGURE 12. Time to solution for cases of missing persons.

users (last test), the time to locate the missing person was reduced by 10 minutes, and the missing person was located in the same city.

The data shows that the time to solution decreases when crowdsensing increases. (In parallel, it also improves the time to solution for Amber Alert leaders, according to a statistical report on this program given that, overall, notification is issued one hour after reporting an event, and without taking into account the additional 3 hours on average for recovering the missing person).

VII. CONCLUSION AND FOLLOW-UP RESEARCH

In Ecuador, the rate of missing persons reported per year has reached 10,000, according to the Office of the Attorney General, indicating that the risk is a constant for all Ecuadorians. Currently, the agency in charge of the problem is DINASED. However, their standard procedures do not incorporate the general population. Our research includes the citizenry as a principal-agent of support in search and rescue operations for missing persons. This project presents SiGPro, a Progressive Notification System that integrates IoT communications for real-time alert broadcasts to report missing-persons cases to citizens.

The MQTT communications protocol has fulfilled all our expectations for a communications platform. It has become our main resource for sending and receiving notification messages to end-users. The MQTT topics can be configured by defining geographic areas using the Progressive Notification Feature, which uses MongoDB's GoJSON to search registers and documents by applying geospatial filters as referential coordinates. We designed these components to guarantee crowdsensing and maintain a high number of users in the system, avoiding alert saturation around the same event and for the same user. As time passes, notifications enlarge their geographic coverage in order to reach more people (but without overwhelming any single user). Expansion occurs as MQTT bridges connect all the MQTT servers. This support allows us to promote our notification system to the national level by exploiting the server hierarchy.

The system was tested in three cities, the principal city being Quito (MQTT Bridge). We simulated 19 rescue scenarios, maintaining strict confidentiality regarding clues or characteristics of the missing person. Students and other people played their roles of community users and administrative users and benefited from being co-researchers in the project.

Crowdsensing is key to the success of the project. The project depends upon community participation to reduce search and rescue time. As the number of sensors (users) increases, the probability that the system will successfully locate the missing person also increases, i.e., the system returns a lower percent error. We verified that 20 users are inadequate to solve a missing-persons case in a single city, provoking a national alert. However, with 140 users, the case can be solved locally, since the time to solution and distance decrease. Comparing these two cases, we verified that time to solution improves by 10 minutes, and the search area decreases by 58 kilometers. These data are better than those returned by the Amber Alert system, the project that offers comparable state-of-the-art features. According to the latest report, Amber Alert requires an average rescue time of 1 hour (corresponding to the time to send an alert) + 3 hours (time for rescue).

We compared our earlier project with our current one [16]. The comparison shows that, on average, the new version achieves an 85% optimization, since users no longer receive redundant notifications, except when users cross boundaries between inner and outer notifications areas. We should emphasize that the prior version [16] did not support inter-system communications, i.e., each notification area remained isolated. SiGPro has implemented a web application for visualizing missing-persons cases and a configurable Progressive Notification System, thus fulfilling the goal of creating a system adaptable to the physical and rational characteristics of any city. We were able to optimize the previous research project and create a system that offers ease of use to both the national police and citizens who participate in search and rescue operations for missing persons.

In our follow-up research in the future, we want to incorporate Big Data technology to analyze data received by the system and perform statistical analyses that reveal+ where missing-persons events are most likely to occur. With these data, we can reinforce risky areas with more police in order to prevent kidnappings. We also want to develop a complete management system using SPA (Simple Page Application) so that data display while they are being generated.

REFERENCES

[1] AWS Official Page. *Acuerdo de disponibilidad, Amazon Web Services*. Accessed: May 23, 2019. [Online]. Available: https://aws.amazon.com/es/about-aws/whats-new/2019/01/amazon-ecr-announces-99-9–service-level-agreement/

[2] A. Banks and R. Gupta. (Oct. 29th, 2014). *MQTT Version 3.1.1. OASIS Standard*. Accessed: Dec. 20, 2019. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

[3] CNN Español. (Sep. 13th, 2017). *México, el País Donde Hay Más de 32.000 Desaparecidos*. CNN. Accessed: Apr. 1, 2019. [Online]. Available: https://cnnespanol.cnn.com/2017/09/13/mexico-el-pais-donde-hay-mas-de-32-000-desaparecidos/

[4] Eclipse Paho Official Page. *MQTT and MQTT-SN software*. Accessed: Apr. 3, 2019. [Online]. Available: https://www.eclipse.org/paho/clients/android/

[5] T. Griffin, "An empirical examination of AMBER Alert 'successes,'" *Journals Criminal Justice*, 38, no. 5, pp. 1053–1062, 2010, doi: 10.1016/j.jcrimjus.2010.07.008.

[6] B. Guo, Z. Wang, Z. Yu, Y. Wang, N. Yen, R. Huang, and X. Zhou, "Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm," *ACM Comput. Surv.*, vol. 48, no. 1, p. 7, 2015, doi: 10.1145/2794400.

[7] *La Línea de Fuego, 10 Mil Personas Desaparecen Anualmente en Ecuador Desde el 2014*. Accessed: Mar. 29, 2019. [On-line]. Retrieved from. [Online]. Available: https://lalineadefuego.info/2018/09/11/10-mil-personas-desaparecen-anualmente-en-ecuador-desde-el-2014/

[8] Ministerio del Interior. (2018). *Estadísticas Generales de Personas Desaparecidas*. Accessed: Mar. 29, 2019. [Online]. Available: http://www.desaparecidosecuador.gob.ec/estadisticas-generales. [Retrieved: 29-mar-2019].

[9] Ministry of Women and Child Development. (2018). *National Tracking System for Missing & Vulnerale Children*. Accessed: Apr. 1, 2019. [On-line]. Retrieved from. [Online]. Available: https://trackthemissingchild.gov.in/trackchild/index.ph

[10] MongoDB Official Page. *GeoJSON Objects*. Accessed: Apr. 2, 2019. [Online]. Available: https://github.com/mongodb/docs/blob/v4.0/source/reference/geojson.txt [Online]. Available: and https://docs.mongodb.com/manual/reference/geojson

[11] A. Padilla. *Meridianos y Paralelos*. Accessed: Jun. 5, 2019. Online]. Available: https://sites.google.com/site/antoniopadillac/Home/pesosymedidas/meridianosyparalelos

[12] A. Rojas. and G. Sánchez. *Cada día desaparecen 26 personas en Guatemala*. Prensa Libre. Accessed: Apr. 1, 2019. [On-line]. Retrieved from. [Online]. Available: https://www.prensalibre.com/guatemala/pl-datos/cada-dia-desaparecen-26-personas-en-guatemala/

[13] A. Rosero. (Aug. 14th, 2019). *Tres unidades policiales buscan a Anahí de dos años; los padres piden ayuda*. El Comercio. [Online]. Available: https://www.elcomercio.com/actualidad/policia-busqueda-anahi-desaparicion-fiscalia.html

[14] U.S. Department of Justice. *AMBER Alert–America's Missing: Broadcast Emergency Response*. Accessed: Apr. 3, 2019. [Online]. Available: https://www.amberalert.gov/. [Retrieved: 03-apr-2019].

[15] Z. Viso. (Dec. 18th, 2017). *Nueva App Para Localizar a Personas Desaparecidas en Sitios sin Cobertura*. Nobbot Accessed: Apr. 1, 2019. [Online]. Available: https://www.nobbot.com/pantallas/app-localizar-personas-desaparecidas/

[16] Zambrano, A., Ortiz E., X. Calderón, and M. Zambrano, "Quito SmartSafe City: Un Sistema en Tiempo Real de Rescate Usando Comunicación IoT," in *Proc. CibSE*, vol. 8, Nov. 2017.

[17] D. Luan, Y. Yang, E. Wang, Q. Zeng, Z. Li, and D. L. Zhou, "An efficient target tracking approach through mobile crowdsensing," *IEEE Access*, vol. 7, pp. 110749–110760, 2019.

[18] P. Pierleoni, R. Concetti, A. Belli, and L. Palma, "Amazon, Google and microsoft solutions for IoT: Architectures and a performance comparison," *IEEE Access*, vol. 8, pp. 2169–3536, Dec. 2019.

[19] E. Al-Masri, K. R. Kalyanam, J. Batts, J. Kim, S. Singh, T. Vo, and C. Yan, "Investigating messaging protocols for the Internet of Things (IoT)," *IEEE Access*, vol. 8, pp. 5455–5470, 2019.

ANA MARÍA ZAMBRANO V received the degree in networking and electronics from the Escuela Politécnica Nacional, Quito, Ecuador, in 2010, and the Ph.D. degree in telecommunications from the Universidad Politécnica de Valencia, Spain, in 2015. She is currently with the Escuela Politécnica Nacional. Her research interests include real-time applications, distributed systems, and the Internet of Things alongside smartcities. She received the Scholarship for the Ph.D. Studies from the Universidad Politécnica de Valencia, in 2011.

MARCELO ZAMBRANO V received the degree in electronics and telecommunications engineering and the Master of Business Administration (MBA) degree from the Escuela Politécnica Nacional, Quito, Ecuador, in 2001 and 2009, respectively, and the Ph.D. degree in telecommunications from the Distributed Real-Time Systems and Applications Research Group, Universitat Politècnica de València, Spain. He has more than 15 years of experience in university teaching and more than 20 years leading projects and technology areas in different companies, Ecuador. Since 2018, he has been a Teacher-Researcher with the Faculty of Engineering in Applied Sciences/Telecommunications Engineering Career, Universidad Técnica del Norte, Ibarra, Ecuador. His research interests include command and control systems applied to emergency management.

EDUARDO LUIS ORTIZ MEJÍA received the degree in electronic engineering and information networks from the Escuela Politécnica Nacional, Quito, Ecuador, in 2018. He is currently with Company that provides software for financial systems. He has collaborated with the development of several research projects with the Escuela Politécnica Nacional, which includes the development of real-time systems, such as mobile applications and web systems. He has implemented a network of heterogeneous sensors for monitoring sensitive information (Quito case study) in the Internet of Things.

XAVIER CALDERÓN H is currently with the Escuela Politécnica Nacional, where he is also a Full-Time Principal Professor, a member with the Department Council (Department of Electronics, Telecommunications and Information Networks), the Head of the Computer Laboratory, Faculty of Electrical and Electronic Engineering, the Director of the Research Projects, such as Seed and Junior with the National Polytechnic School, a Co-Director with the CEPRA-CEDIA Research Project, and the Head of the Department of Electronics, Telecommunications and Information Networks (DETRI). He has Directed the research project related to intelligent transportation systems and collaborated in a research project related to internet accessibility, Ecuador.