**CERTIFICATION**

This is to certify that this project was written by H[alima Grema](mailto:halimagrema05@gmail.com) and that the project has been read and approved having met the requirements for the award of a bachelor degree, Bachelor of Science in Software Engineering ,(School). The copyright of this project belongs to the institution where the project was carried out, under the supervision of :

…………………………... …………………………....

Name Required

(Project Supervisor)

…………………………... …..………………………..

Name Required Date

(Head of Department)

**DEDICATION**

This project is dedicated to God who has kept me to see this day. It is also dedicated to Mrs Grema ,my siblings ,the developers Orb, EMI Systems Inc and my friends.

**ACKNOWLEDGMENT**

I am specially grateful to those who helped make this project a reality. I thank Mrs Grema for her support and encouragement. My appreciation also goes to Mr Ore Deji and Miss Annabel Peters who found time out of their busy schedule to look into this work, I express my gratitude for their patience and assistance.

I also would like to thank the people from the Sector for Analytic, Telecommunication and Information Technologies, Ministry of science and technology , Federal Republic of Nigeria, , as well as people from Government Office of Information Technology and e-Government, Federal Republic of Nigeria, for their support and constructive comments during the implementation and experimental testing.

**ABSTRACT**

This project will provide an introduction to the design and implementation of an android application named

Decoy Mobile Application for Kidnapped Victims as a case study. It also introduces some software concept for building a quality mobile application. The mobile application will enable the kidnapped victim to send a SOS message to the police center and also can be used to send an SOS to a number of the users choice registered on the application. The application can be used by a combination of the volume buttons to easily send the users location in a matter of Minutes.

**Table of contents PAGE**

1. Title Page ------------------------------------------------------------------------- 1
2. Certification ---------------------------------------------------------------------- 2
3. Dedication ------------------------------------------------------------------------ 3
4. Acknowledgment --------------------------------------------------------------- 4
5. Abstract ---------------------------------------------------------------------------- 5

**PART ONE :**

**INTERNSHIP REPORT**

1.1 Preparation ------------------------------------------------------------------------ 9

1.2 History ----------------------------------------------------------------------------- 9

1.3 Mission,Vision and Values ---------------------------------------------------- 9

1.4 Structure of EMI Systems --------------------------------------------------- 10

1.5 Personal Roles in the Company --------------------------------------------10

1.6 Comments and Critics ------------------------------------------------------- 11

**PART TWO :**

**CHAPTER 1 : Introduction to the study**

1.1 Introduction --------------------------------------------------------------------- 13

1.2 Project Scope --------------------------------------------------------------------14

1.3 Problem Statement ------------------------------------------------------------14

2 Related work ---------------------------------------------------------------------14

3 The Architecture of the Proposed Platform -----------------------------19

4 Implementation ---------------------------------------------------------------- 21

4.1 Release Stage -------------------------------------------------------------------21

4.2 User Interface --------------------------------------------------------------------22

4.3 Widget -----------------------------------------------------------------------------24

4.4 Service------------------------------------------------------------------------------28

4.5 Volume Service------------------------------------------------------------------28

##### 4.6 Widget Service-------------------------------------------------------------------30

##### 4.7 Presentation of Solution-------------------------------------------------------31

#### 5 Experimental Usage and Evaluation--------------------------------------32

6 Conclusion-------------------------------------------------------------------------33

Data Availability-----------------------------------------------------------------33

Conflict of interest--------------------------------------------------------------34

References ------------------------------------------------------------------------35

**PART ONE**

**INTERNSHIP REPORT**

**1.1 Presentation**

I did my internship at EMI Systems. EMI Systems is a private security system company that maintain and manage security systems for Micro-finance Banks, Finance Companies, Savings and also alarm systems for companies such as NNPC, Julius Berger, and Madeira Estate. It is located at Lobito Crescent Wuse 2 Abuja.

**1.2 History**

EMI Systems was conceived in 2010 by a group of Israelis from Israel looking to extend their companies to the heart of Africa, Mr. WolfGarrd .They Started by introducing state of the art security systems to all types of countries nation. Consequently, they grew into one of the best Financial Solutions companies in Nigeria.

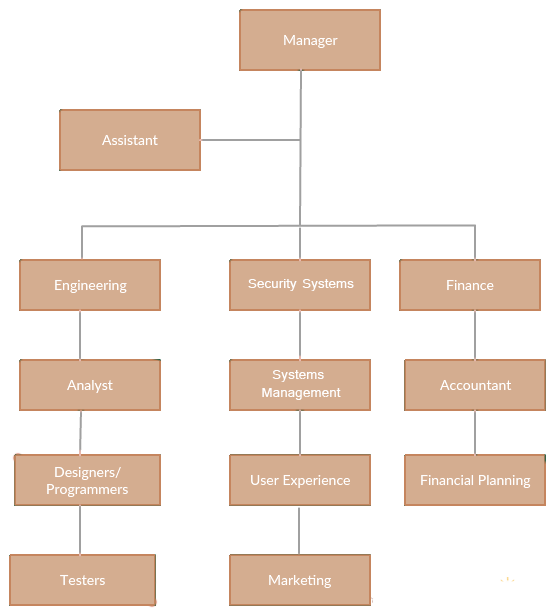
**1.3 Mission, Values and Vision**

Their Mission is to analyze security problems and build the best and lasting solutions, and by that meeting client’s requirements and to achieve complete customer satisfaction with best quality and timeliness in project delivery.

Their core values are Quality and Cost-effective services by team of young and experienced workforce; Constant Learning, Deploying New Skills and Offering New Solutions.

Their vision is to be a leading global player in Financial Software Solutions.

**1.4 The Structure of Emi Systems**



**1.5 Personal Roles in the Company**

As an intern, I was new to the system of the company and the daily routines so I was assigned a supervisor who puts me through every job assigned to me. The first week, I worked as a technical support team and my daily task was to report to sites with the necessary equipment’s needed to perform the job required . Then few weeks later, I worked at the internet Unit, where I was exposed to setting up and providing internet access to the employees and thereafter, I was assigned jobs to build an Android Application for clients to submit their Issues regarding the security system using Android Studio and some other platforms. On some occasions, I was chosen to go out with some top employees to companies for consultancy to fix some bugs in their security system.

**1.6 Comments and Critics**

EMI systems limited comprises of young programming geeks who have cultivated a culture of working tirelessly to help the company grow and succeed. Their major strength is their dedicated team of experienced software engineers and technicians who work across a wide range of programmes. EMI systems limited has helped me understand how to move a software through the software development lifecycle and taught me the intricacies involved in those phases.

EMI systems limited also taught me the pre-requisites of a software code which includes;

* Give clear comment of every function/method used in your program by declaring the name, the parameter and return type at the beginning of such function/method.
* Write as simple as you can in order for other programmers to be able to take up the job.

EMI systems limited lacks the proper way of managing human resources like employee benefits and recruitment which exerts more pressure on the employees.

**PART TWO**

**Chapter 1 : Introduction to the study**

**1.1 Introduction**

Nowadays, IT technologies grow rapidly and constantly. Our daily life cannot be imagined without using these technologies. An accelerated development of the mobile OS, such as Android and iOS, has changed the main point of using mobile phones. A mobile phone is not used only for telephoning and sending messages but also is used for many new and smart features. Some of these features allow location sharing and tracking, which denotes a powerful and efficient tool that should be used carefully because of the private information, such as location. In general, people could experience a lot of unexpected situations like accidents, hijacking, and criminal rate on a daily basis. Fortunately, people have their mobile phones next to them at any moment, so they can feel more protected. Thus, they can act in emergency situations quickly and save their lives. Due to the fact that Android is the most commonly used OS for mobile, there are many applications developed and specialized for its easy use. In other words, nowadays, the problem of emergency, dangerous situations can be potentially solved to the certain extent. According to statistics in Nigeria over the past years, there were 139 kidnapping, 791 rapes, 257 rape attempts, 43,482 cases of home violence, 1446 thievery, 32,584 robberies, 426 trafficking, 4 terrorism actions, 791 cases of school violence, 439,368 car accidents, and 216,041 fire accidents. This paper suggests an approach to the kidnapping problem in Nigeria, through the implementation of modern mobile architecture, to address the mentioned issues of emergency and dangerous situations for people who are in trouble. The main aim is to enable people to send in an easy and unnoticeable way a SMS message containing their location to the Police Operating Center or individual of their choice. To the best knowledge of the authors, this appropriate combination of advanced Android technologies proposed in this paper for the first time constructs unique, complete, and operative usable software platform for sending emergency messages. The proposed platform clearly overcomes the problems introduced in previously proposed solutions described in literature.

**1.2 Project scope**

The rest of the paper is organized as follows. In Related Work, the related works and applications are presented. In The Architecture of the Proposed Platform, the architecture and operation of the proposed solution are presented. A description of implementation of the proposed platform is presented in Implementation. The next section, Experimental Usage and Evaluation, is provided and discussed.

The focus of the project will be the:

1. Existing problem faced by mobile user to request help when in need. As stated in the Introduction of the study, victims of a criminal situation have problems of requesting help during these situations. Without the ability to think fast, the victim would be unable to think of any relevant way of escaping the potentially life threatening situation. Calling for help using the mobile devices would be almost impossible if the victims were involved in a kidnapping situation. Therefore, this study will be focusing on the solution for requesting help during these critical situations.
2. Users’ perspective regarding the implementation of mobile application towards personal safety. With the significant rise in technological development of mobile application, the author will take a closer look regarding the implementation of mobile devices in the issue of personal safety. This initiative will make use of the data collected via the literature review and other research methodology planned later for the research process.
3. Review and analysis of the existing mobile application for personal safety purpose. The author will also focus on identifying the essential attributes for a personal safety mobile application to be effective towards the user and compare the proposed idea of the study with other related studies of the implementation of mobile application toward personal safety.

**Why mobile application ?**

* **Ubiquity of smartphones/tablets** : Smartphones/Tablets have increasingly replace Large computers because of their portability. A mobile app, remains on the user's smartphone, and is therefore ideal for frequent and repeated use.
* **Proximity to people**: Mobile devices, especially Smartphones, are much more personal and intimate devices than a laptop or desktop device. For most people their phones are never further than 1 meter away 24/7.
* **Unmatched user-experience**: Apps offer a user-experience which mobile Websites or widgets are unable to provide.
* **Ubiquity of app stores**: Smartphones will always depend on the app store. so you will have easy and convenient access to the world of apps .

**1.3 Problem Statement**

When confronting a potential life- threatening situation such as kidnapping, mugging or even robbery, the victim has very little time gap to think of the best way to escape from the situation. However, requesting for help is the most instant procedure for the victim to think of during the critical situation. In light of rapid concern of safety issue, mobile apps implementation may be extended into personal safety procedures.

**2 Related Work**

Mobile applications and mobile services are becoming one of the technology mainstreams in recent years. Android-based applications are becoming a proper tool in order to solve different everyday life problems [1, 6, 7]. Recently, many researchers have tried to find a proper solution to address the security issue in the case of emergency [8–13], but there has been no proper solution. In [8], an Android application that offers SOS message sending using the GPS location via a Whats App messenger to redefine recipient was proposed.

To activate an SOS message sending, user has to shake his Android phone while the application is running. The main idea the author had was to enable sending of user’s location via some of the modern services such as Whats App. However, if the application is not running, and the user is in a situation where he cannot pull out his or her phone from a pocket or a bag, this application is of no use. Similar application intended for the cases where there is no operation of mobile communication systems was presented in [9].

The proposed application is used in a way that a group of phones, which have this application installed, create an ad hoc, a peer-to-peer-like, wireless network. On the one hand, this application is very good because in many cases, such as earthquake and other natural disaster cases, due to the damages, there is no proper operation of standard communication systems. On the other hand, this is not a good solution because it relies on a fact that each user has an installed proposed application, which could not be the case and which surely decreases the effectiveness of the proposed application.

This application is also convenient for the places when there is no mobile communication signal, such as rural places. However, in this modern society, there is almost no place where there is no mobile communication signal. Due to the all above mentioned options, it can be concluded that the proposed application is good to be used in some situations but it cannot address the problem that we try to solve here.

If other Android phones near us do not use this application, then this is useless, and the messages go as far as the network of phones

goes, and apart from all that, in most cases when there is an emergency, the mobile network is available, so there is no need for this application. The application that addresses the problem of the location on the roads is presented in [10].

Namely, it was concluded that road accidents are the factors that increase the mortality level. So the basic idea of the application is to warn a driver that he is approaching dangerous corner and help him slow down and prepare for the corner. The application is notifying a driver about the dangerous corner within 700 meters before the sharp corner. The warning is realized by playing “buzz” sound as the alert, to tell the driver that there are dangerous corners ahead. Besides, this system will give suggestions to the closest emergency places by only refreshing the list of the places and pin point the emergency places on map.

The localization technique used in this application is very similar to those we use in our application. Both the application presented in [10] and our application use the GPS service to obtain accurate user position.

However, the application [10] is customized for road transport and cannot solve several different kinds of dangerous situations. Furthermore, there is a similar application is bSafe Android application [11]. This application provides more than one option to the user; namely, four services are provided: bSafe Alarm, Follow me, Timer Alarm, and Fake call. The first one is intended for sending an SMS with the user location, but besides the location, both audio and video data can be sent together with the information about the location.

When this option is activated, an automatic recording is triggered and the recorded data is sent to the desired number or numbers. In this application, user can define a friend cycle containing as mush numbers as the user wants, and these numbers can be edited unlimited number of times. The second option allows tracking of the user’s location on the map in real time. The third option allows user to set the time he thinks he will reach some location, and if he does not reach the location in defined period, SMS message with his current location will be sent. The last, but not the least, option provides a face call. When user activates this option, the face call is performed, which

can be beneficial in the case when user wants to divert attacker’s attention by letting him know that there is someone who could hear if attack happens, which could be used as a kind of evidence.

However, this app is not integrated with any centralized system and is not useful in several dangerous situations where there is no time for active usage of mobile phone. The application GoSuraksheit [12] is also one of the similar applications. This application is also intended for sending of the SMS with the user’s location. The operation principle is similar to the presented applications [10, 11].

The user’s location is obtained by using the GPS and then forwarded to the desired numbers (up to five numbers). The main advantage of this application is that it provides a possibility to share the location on a popular social network, Facebook.

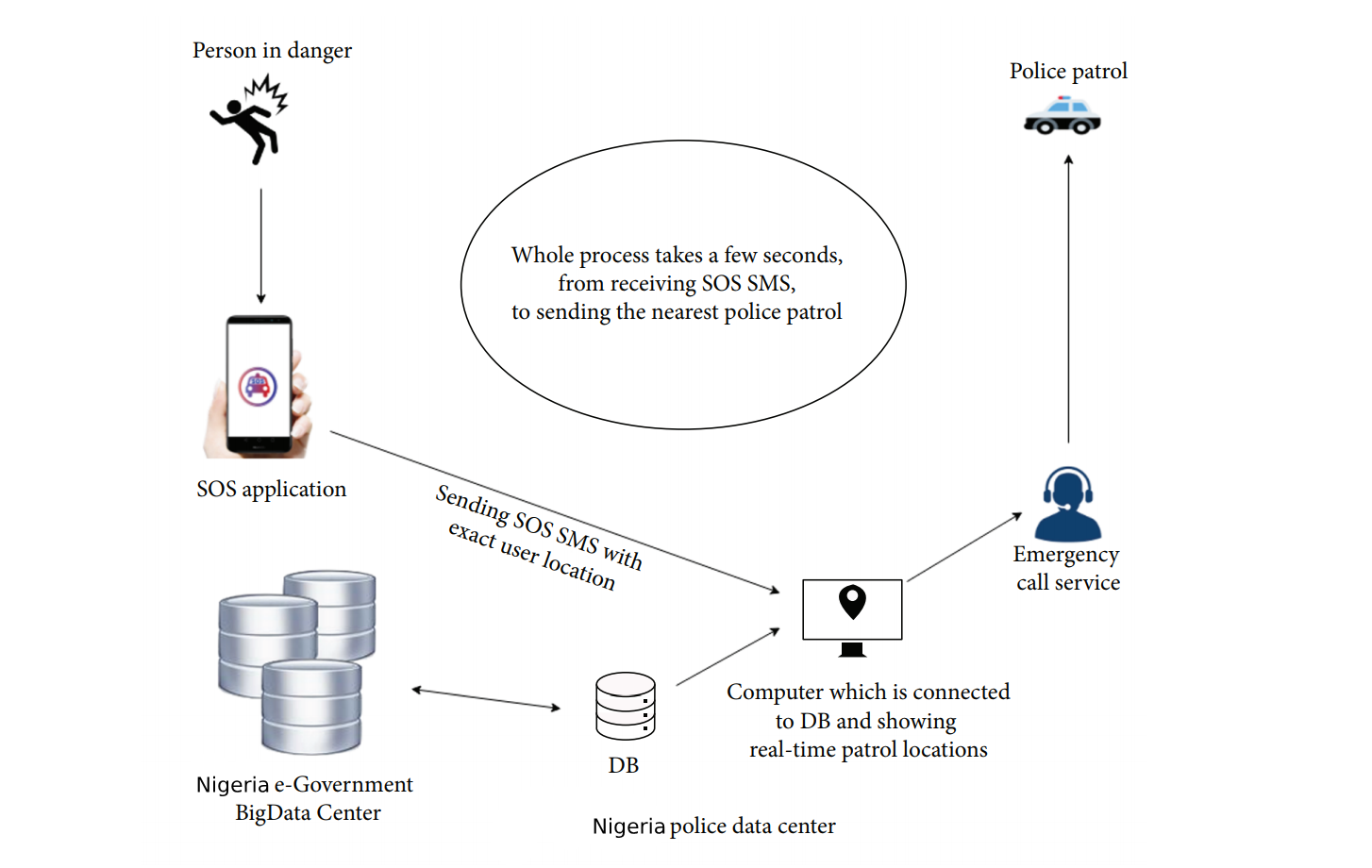


Figure 1: The architecture of the proposed platform.

The main disadvantages of the service presented in paper [12] are as follows:

(i) The service does not include any centralized logic and integration into government system.

(ii) The client mobile app is not applicable in many dangerous situations.

The application presented in [13] allows mobile phone user in an emergency to send the SMS to the police or rescue center. With each request, user’s location coordinates are sent to the police center number. This application can operate in two modes, network mode and GPS mode. All the details of a user, such as description, priority, and current location report, are sent to the server. In other words, this app solves the problem of integration with the centralized system in order to help the citizens, but it is also inapplicable in several dangerous situations when the victim cannot actively use mobile phone. To summarize, in order to trigger previously listed applications, it is necessary to open the application on a mobile phone which is sometimes impossible to do. The shortcomings of these applications is that there is no emergency button or trigger for fast and silent alert, even if the application has some sort of silent switch, such as phone shaking or detection of user running, which is not good because the application can be triggered accidentally. In addition, some of the presented applications do not have the widget. Moreover, there is no any information about experimental usage of the proposed app [8–13] and its reaction time measurements in real-life situation. These services also cannot give any useful analytic after long-time operative usage of client app. On the other hand, the application we propose

here effectively works in the situations when the victim cannot easily and actively use mobile phone. It can even work if the screen is broken and the device is locked, and it sends messages directly to the Police Data Center or the registered number as the case may be. This center is integrated with the Nigerian e-Government BigData Center and it will be possible to apply advanced analytics solutions after the proposed service generates a sufficiently large amount of data.

**3. The Architecture of the Proposed Platform**

The architecture of the proposed application and its operation is presented in Figure 1. As it can be seen, the first step is sending of the SOS message when the user is in danger. The second one is processing of sent data at the police computer center that has a DB connected to the Nigerian eGovernment BigData Center. This DB has a list of users and exact real-time locations of police patrols, so it can be easily calculated the nearest patrol distance. After this step, patrol is informed by the dispatch center about location where they should go. The main purpose of this process is to minimize the time from sending the SMS to getting a help from the police. By using the proposed application, in only few steps, of which the first one is triggering SMS with buttons, the second one is related to the DB with users where we already know who is sending the SMS, and the third is a determination the closest police patrol according to the obtained information on user’s location (each patrol is equipped with a GPS transmitter). The patrol should be informed where to go by the operator from Police Data Center, and that is the only part that is not automated.

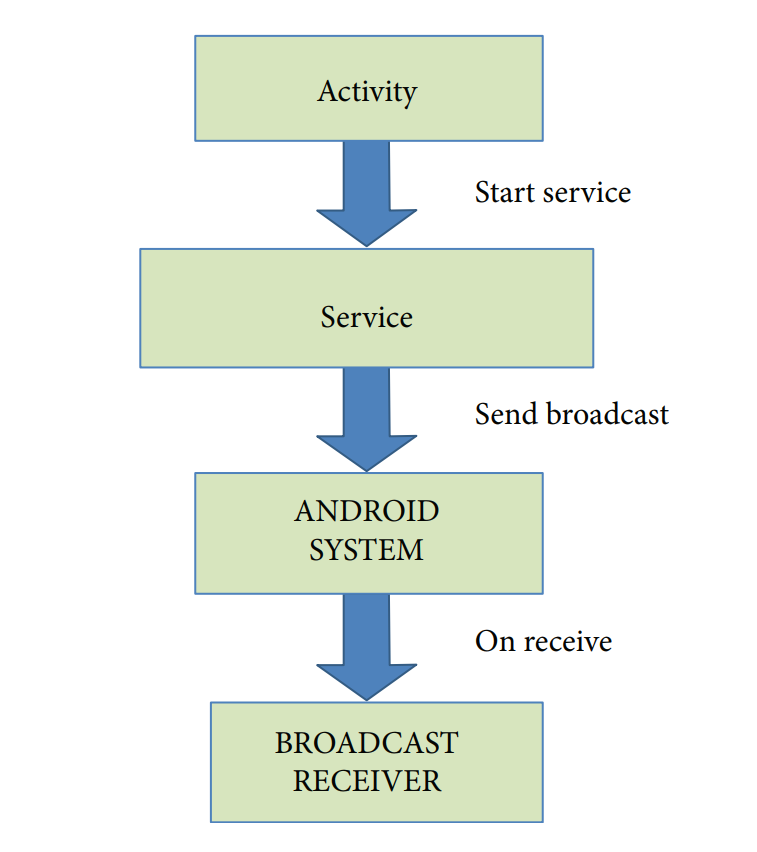


Figure 2: The architecture of the client Android application.

The patrol should be informed where to go by the operator from Police Data Center, and that is the only part that is not automated. The proposed architecture of the client Android application is shown in Figure 2. Three components properly combined are necessary to provide a good operation flow of suggested solution: activity, service, and broadcast receiver [14]. An activity interacts with user so it creates a window to place UI (user interface) elements. An Android application can contain several activities, which means that many different screens can interact with each other [15]. In proposed approach, user via view in activity sends a request for starting service that is responsible for getting current location. To explain, service is a component which runs in the background without direct interaction with the user and it is used for repetitive and potentially long running operations. As the service has no user interface, it is not bound to the life-cycle of an activity [14]. In the proposed client app, broadcast receiver is responsible for registering system events and allows reading pressed buttons for default combination as an Android component which allows you to register for system or application events. All registered receivers for an event are notified by the Android run-time once this event happens. To sum up, implemented client solution is consisted of these three components. For release stage, SOS application also uses Google API Client library for locations (“com.google.android.gms:play-services-location:10.2.6”) for getting user’s location. Google Location Services API is the most popular service for adding location awareness with automated location tracking, geofencing, and activity recognition. The proposed architecture of the client Android application also allows implementation of multiple options, such as ;

i) emergency trigger—for emergency SMS sending to the emergency services

(ii) widget trigger—same as previous, but only for sending when phone is unlocked

(iii) pattern/pin—for opening and activating application

(iv) time locker—for setting application’s active hours

(v) SMS remote control—if application receives a message from predefined number with certain code (#123backup), for example, it will trigger data backup (SMS, images, videos, and documents), user will be able to select what he wants to save or upload to cloud (

vi) restore/factory reset—in case the device is stolen or lost (vii) app icon hidden in whole user system—in case the phone is stolen

(viii) camera remote control—again trigger feature remotely, this time turn camera on, take a picture with front and rear camera, and send it to predefined number.

For the first release of the service, we have selected first two options to implement, emergency trigger and widget trigger. In other words, our SOS application is developed for working in two ways. First, service is implemented to have listener for buttons that user needs to press (default combination for sending SOS message). Second, service is implemented when widget has occurred. When user presses widget, it automatically sends message with its own location and any other information if the service is customized for different users.

**4. Implementation**

**4.1. Release Stage**

SOS application for sending emergency messages uses Android platform. This is an offline app, which means that it can work without access to the Internet. What this essentially means is that it gives a better experience to the users, and it can even be a key factor for

them in order to retain or uninstall application. Further, the main reasons for using Android as a client platform are as follows:

(i) Open source (can be leveraged without having to worry about the licensing costs)

(ii) Dissociation of the user’s interface from the business logic

(iii) Asynchronous calls (easy to code client-side multi-threading)

(iv) Customization of user interface (create custom interfaces for different business)

(v) Reusable and responsive components (support for Android material)

(vi) Portability (can easily be ported to other mobile operating systems)

The list of features and advantages embedded in Android OS is quite long. The core release-stage functionality is to send message, prompted by the users, based on existing services implemented for listening and getting location. This client-side module provides integration of Google Service API Location and JSON for parsing model.

On the one hand, with the listener services used on the UI side, SOS client communicates with this service that has been started in background and performs location task and starts broadcast receiver for media. On the other hand, there is a widget, which user presses and triggers service during this process.

##### **4.2. User Interface**

In one word, user interface is everything that the user can see and interact with. User interface of the proposed client Android application is shown in Figure ([3](https://www.hindawi.com/journals/complexity/2018/8283919/fig3/)). Android provides a variety of rebuilt UI components, such as structured layout objects and UI controls that allow to build the graphical user interface (see Figure 4). UI in SOS application is implemented in XML, using primarily Linear-layout [[15](https://www.hindawi.com/journals/complexity/2018/8283919/" \l "B15)]. Linear-layout is a view group that aligns all children in a single direction, vertically or horizontally and also supports assigning a weight to individual child with the android: layout\_weight attribute. This attribute assigns an “importance” value to a view in terms of how much space it should occupy on the screen.

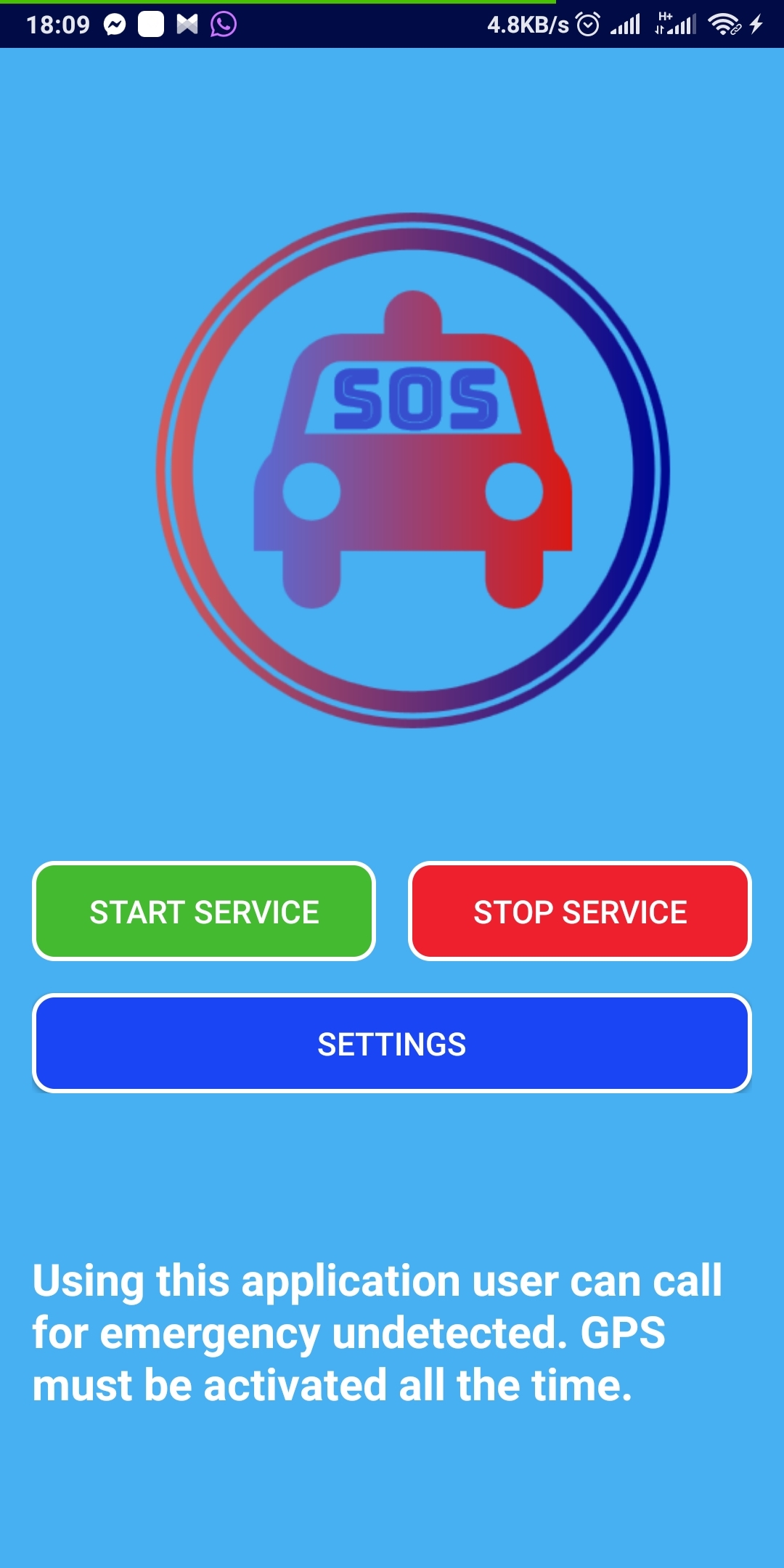


Figure 3: Main screen implementation.



Figure 4: Main screen implemented in XML.

##### **4.3. Widget**

App Widgets are miniature application views that can be embedded in other applications (such as the home screen) and receive periodic updates [[15](https://www.hindawi.com/journals/complexity/2018/8283919/" \l "B15)]. These views are referred to as widgets in the user interface, and it can be published one with an App Widget provider. An application component that is able to hold other App Widgets is called an App Widget host. Figure ([5](https://www.hindawi.com/journals/complexity/2018/8283919/fig5/)) shows implemented SOS App Widget.



Figure 5: SOS widget.

As mentioned, SOS application uses App Widget declared in manifest file (see Figure 6) implemented in class (CustomAppWidgetProvider) which extends class (AppWidgetProvider). The (AppWidgetProvider) class extends (BroadcastReceiver) as a convenience class to handle the App Widget broadcasts. The AppWidgetProvider receives only the event broadcasts that are relevant to the App Widget, such as when the App Widget is updated, deleted, enabled, and disabled. When these broadcast events occur, the AppWidgetProvider receives method calls such as OnUpdate() and OnReceive() method.

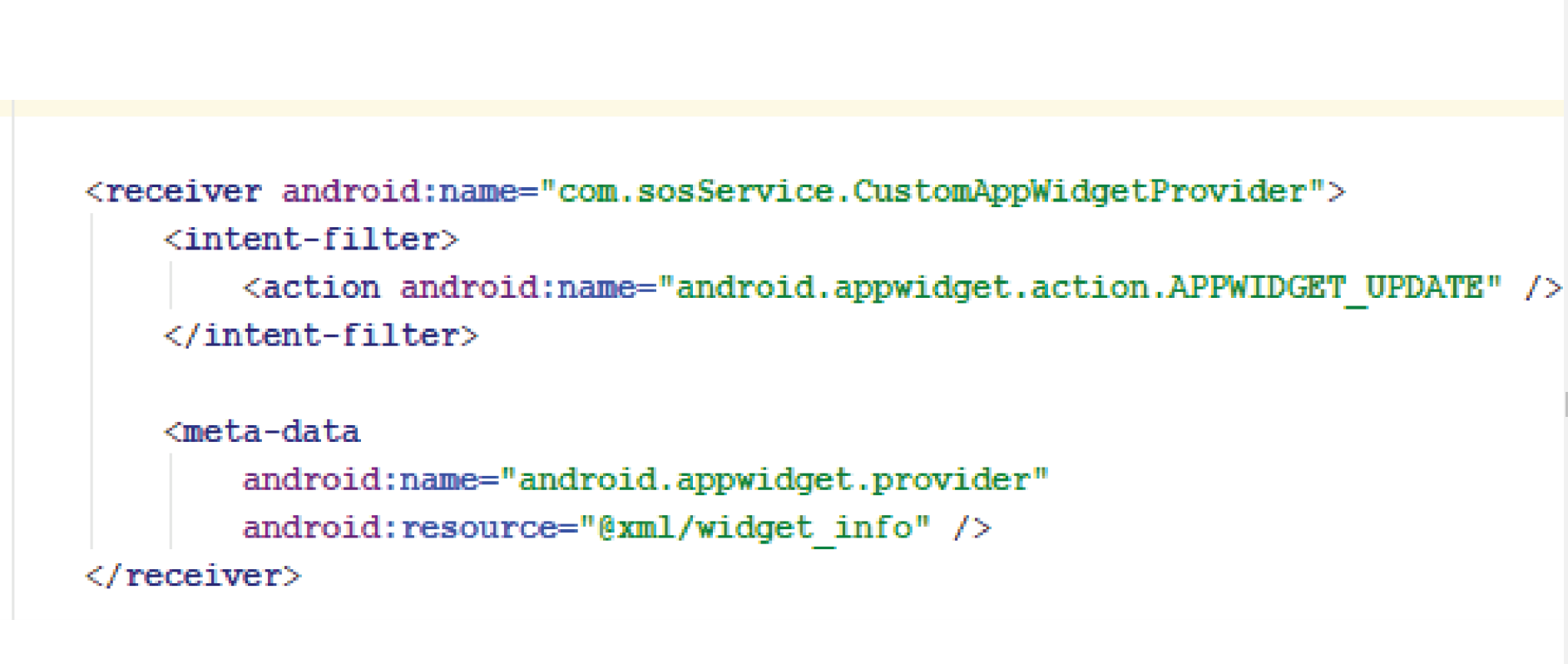


Figure 6: CustomAppWidgetProvider declared in manifest file.

The proposed client application calls OnUpdate() method when the user adds the App Widget, so it should perform the essential setup, such as to define event handlers for views and to start a temporary service, if necessary. However, if you have declared a configuration activity, this method is not called when the user adds the App Widget, but is called for the subsequent updates. It is the responsibility of the configuration activity to perform the first update when configuration is done (see Figure 7).

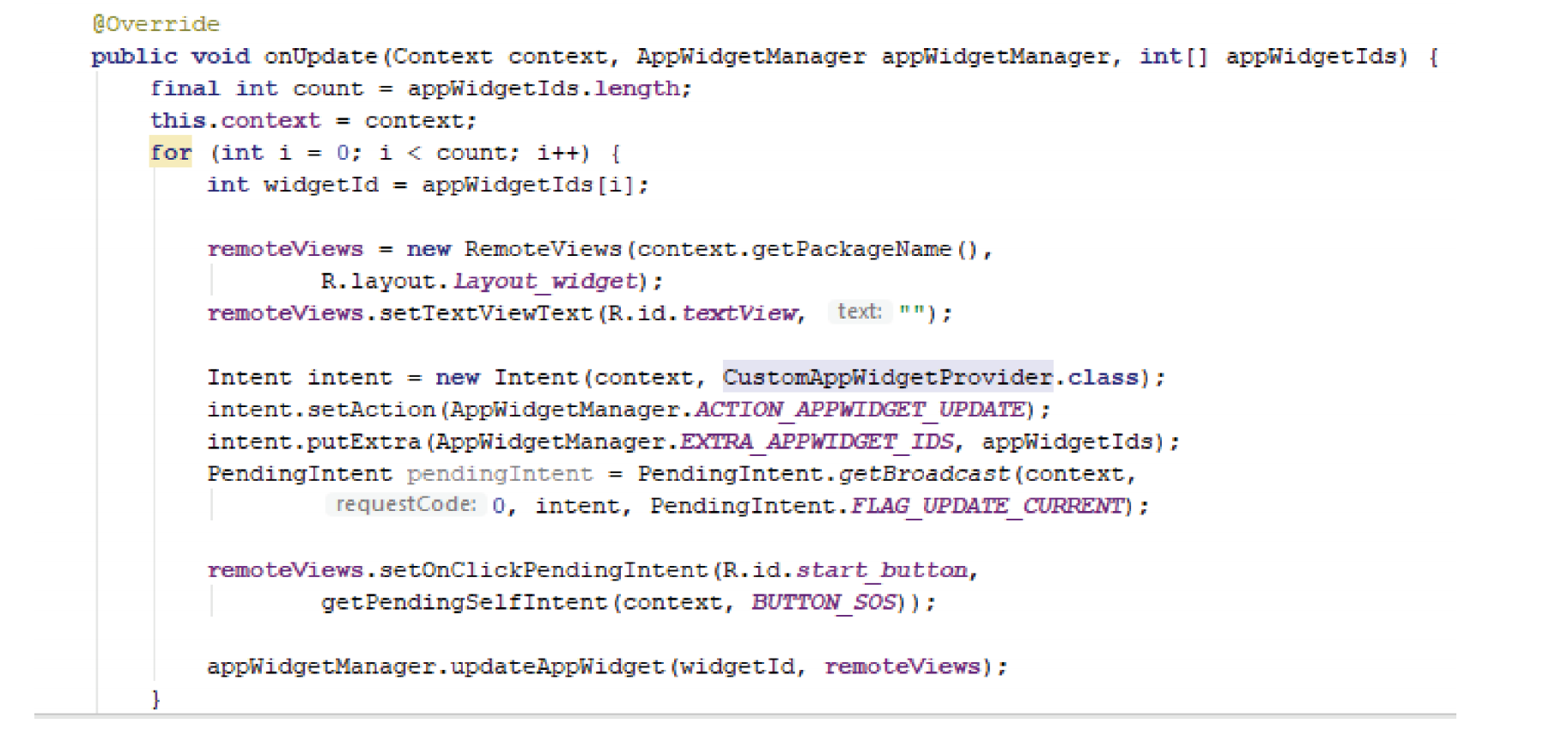


Figure 7: OnUpdate method in CustomAppWidgetProvider class.

The proposed client application calls OnReceive() method for every broadcast and before each of the rest callback methods. This method checks if location is turned on. If yes, service is starting. If not, it shows toast message for warning to turn the location on in settings (see Figure [8](https://www.hindawi.com/journals/complexity/2018/8283919/fig8/)).



Figure 8: OnReceive() method in CustomAppWidgetProvider class.

##### **4.4. Service**

A service is an application component representing either an application’s desire to perform a longer-running operation while not interacting with the user or to supply functionality for other applications to use [[14](https://www.hindawi.com/journals/complexity/2018/8283919/" \l "B14)]. Each service class must have a corresponding <service> declaration in its package’s AndroidManifest.xml. Services can be started with Context.startService() and Context.bindService(). Services run in the main thread of their hosting process. In this work, two services are implemented: volume service and widget service.

##### **4.5. Volume Service**

A volume service is one of two applications services, as we can see in Figure ([9](https://www.hindawi.com/journals/complexity/2018/8283919/fig9/)). Service can be started and stopped by the user, by pushing one of those two buttons (Start service) and (Stop service). It is called volume service mainly because this service is working in background and waiting to capture right the combination of volume buttons sequentially pressed (at this particular implementation case that combination is set to up-down-up-down). This combination of volume buttons pressed triggers the SMS sending (see Figure [10](https://www.hindawi.com/journals/complexity/2018/8283919/fig10/)). After service is started, location services must be enabled so that volume service can send user’s location in message. Basic principle is that after user triggers the right combination, service starts with gathering information about user’s location which usually takes between 1 and 5 seconds (mainly depends on connection quality at the moment of sending SOS SMS message) and immediately after that it sends message with map link. After sending the message, volume service remains active if there is need for a new send.

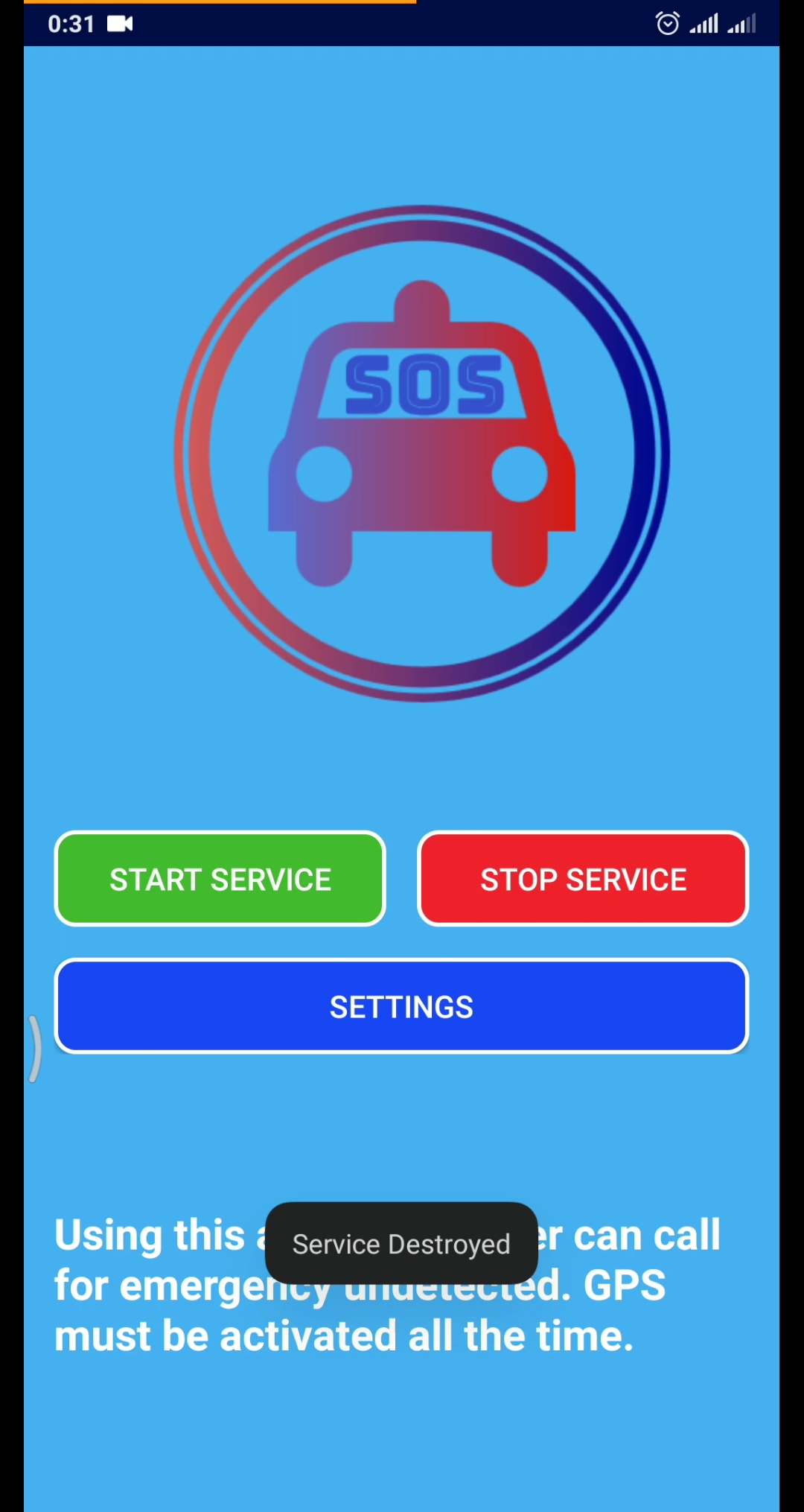
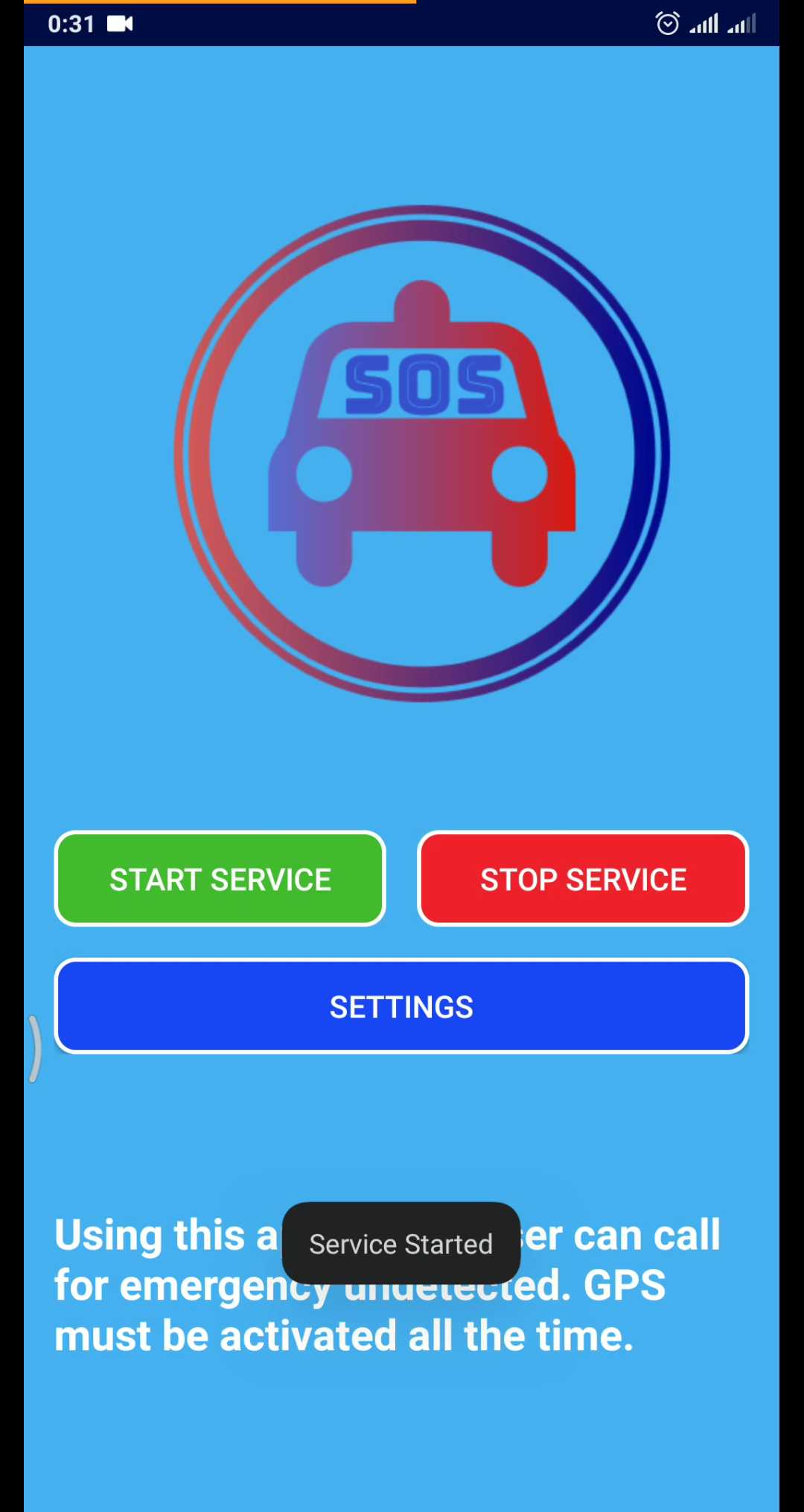


Figure 9: Main screen with start and stop buttons for service and settings button for the app.

Figure 10: Implemented volume service logic.

##### **4.6. Widget Service**

A widget service implementation shown in Figure (1[1](https://www.hindawi.com/journals/complexity/2018/8283919/fig11/)) has almost the same logic as volume service; the difference between them is that the widget service is triggered by widget button, after which is not listening for volume button-pressed combination, but instead it starts location services and sends message after finding the right location. Also, widget service is destroyed after sending message, because its only purpose is to send message in the shortest way. So, the implemented principle is that the service sends a message on location change, as we can see in Figure ([12](https://www.hindawi.com/journals/complexity/2018/8283919/fig12/)). When service finds a location, it sends a message, stops the location updates, and destroys the widget service.



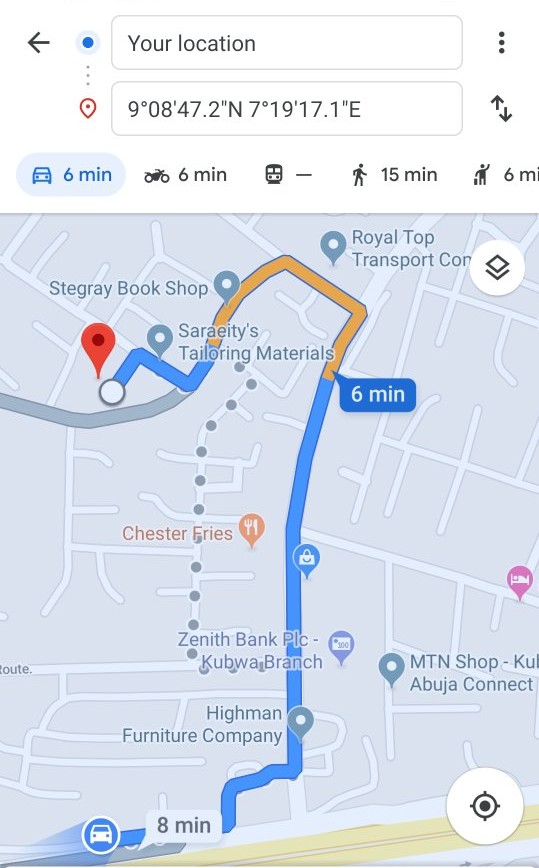
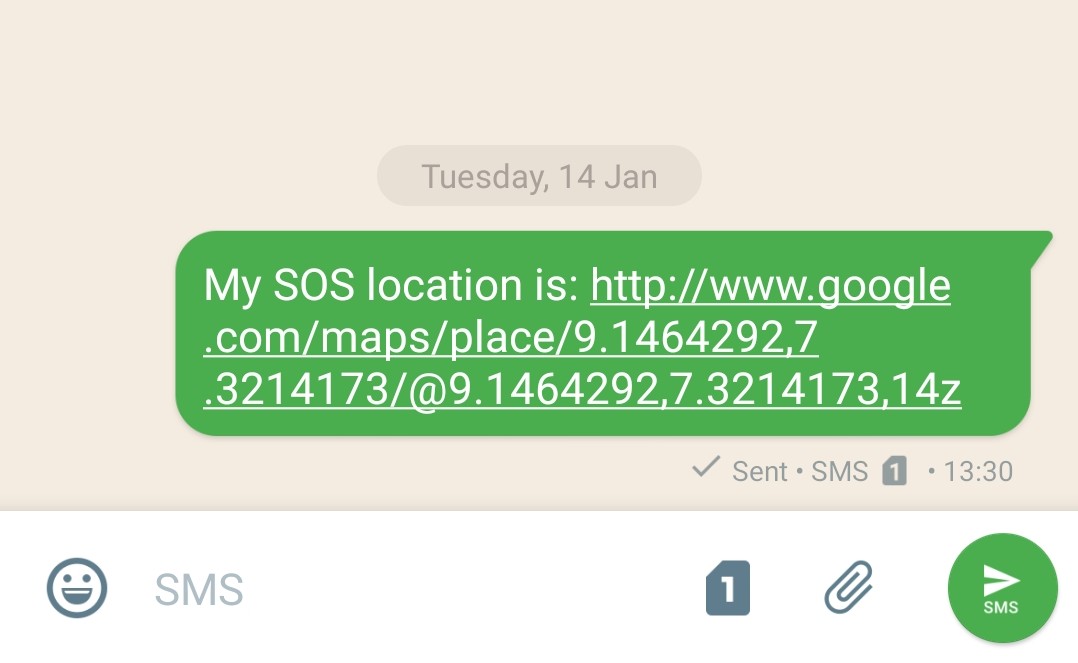
Figure 11:  Implemented widget service logic.



Figure 12:  Implemented location update logic.

##### **4.7. Presentation of Solution**

Final result generated by proposed client is shown in Figure ([13](https://www.hindawi.com/journals/complexity/2018/8283919/fig13/)). We can see a message which contains link to the Google maps with user’s coordinates shown as a pin. Message contains text in this form—“SOS my location is Google maps link.” With this location, user who sends an SOS message can be rescued in the short period of time. The form of SOS message can be easily customized to different kinds of app users.



Figures 13: Example of SOS emergency message.

#### **5. Experimental Usage and Evaluation**

For evaluation purposes, the platform has been tested multiple times in different real-life situations. Application is working as intended and sends a message every time after a given command.

During the period of experimental usage, the shortest reaction time was less than a minute that includes period from sending SOS SMS until the arrival of police patrol, and the longest period was 6 minutes, and that was the situation where police patrol was on the other side of their patrolling area; in other words, they were at furthest possible location from the accident (approximately 4 km). These specific numbers are roughly measured using few simulation attempts as well as some real-life situations and statistical data from the Ministry of science and technology , Federal Republic of Nigeria and they highly depend on internal police organization.

Drawbacks of the app are the locations, which are not correct in closed spaces or not even shown at all, and it is because of device’s inability to find GPS satellites. Also, other issue is the high buildings with lots of

floors, where the coordinates are the same for the whole building and there are lots of apartments in it, but that can be almost fixed by sending the elevation of device. However, we will still have a problem with multiple apartments, for example. So those are the few things that we have to look back at, during further development of the proposed platform.

**6. Conclusion**

In this paper, an Android-based SOS software platform has been presented. The client SOS application is designed in a way to enable user to send his location in a simple and unnoticeable way. The performance of the proposed application has been verified experimentally. The application is shown to be very useful in a large number of life-threatening situations. Moreover, this kind of applications is very suitable for children monitoring during many of their activities, such as going to the school, for a run, or during traveling, and in the situation where children find themselves in an unknown place or dangerous situation.

One of the main advantages of the proposed application is that it will directly connected to the police system and its database, which means that a needed and necessary help can be obtained as soon as possible. Besides, the police have the information on all mobile phone users and real-time locations of all police patrols and there are a large number of highly experienced people who can help a victim. The other advantage is that almost the entire process is automated; the only thing that should be done manually by an operator at the Police Operation Data Center is to determine which of the police patrols is the closest to the victim and to direct them to go the sent location to help a victim. This way, the time needed to react is decreased significantly, which can save many lives.

**Data Availability**

The source code of the proposed application used to support the findings of this study is included within the article. Executable installation file with installation instruction can be found ([https://drive.google.com/drive/folders/1WI4KDNrd9cX49Cf3BS6qJC2f9xMr30Y](https://drive.google.com/drive/folders/1WI-4KDNrd9cX49Cf3BS6qJC2f9xMr30Y)). This file can be easily generated from the source code included in the project. This app can be easily integrated and tested in any police or similar system.

#### **Conflicts of Interest**

#### The author declare that they have no conflicts of interest.

**References**

[1] G. Athanasopoulou and P. Koutsakis, “eMatch: an android application for finding friends in your location,” Mobile Information Systems, vol. 2015, Article ID 463791, 11 pages, 2015.

[2] Y. Yan, S. Cosgrove, V. Anand et al., “RTDroid: a design for real-time android,” IEEE Transactions on Mobile Computing, vol. 15, no. 10, pp. 2564–2584, 2016.

[3] Y. Yan, K. Dantu, S. Y. Ko, J. Vitek, and L. Ziarek, “Making android run on time,” in 2017 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), pp. 25–36, Pittsburgh, PA, USA, 2017.

[4] M. N. Soorki, M. H. Manshaei, W. Saad et al., “Collaborative real-time content download application for wireless deviceto-device communications,” in GLOBECOM 2017 - 2017 IEEE Global Communications Conference, pp. 1–6, Singapore, 2017.

[5] K. Dantu, S. Y. Ko, and L. Ziarek, “RAINA: reliability and adaptability in android for fog computing,” IEEE Communications Magazine, vol. 55, no. 4, pp. 41–45, 2017.

[6] D. Bucerzan, C. Ratiu, and M. J. Manolescu, “SmartSteg: a new android based steganography application,” International Journal of Computers, Communications & Control, vol. 8, no. 5, pp. 681–688, 2013.

[7] K. Bahadoor and P. Hosein, “Application for the detection of dangerous driving and an associated gamification framework,” in 2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), pp. 276–281, Vienna, Austria, 2016.

[8] H. Akshay Kumar, N. Divyashree, A. Nithu, R. Revathi, and Y. Suresh, “Anuti — an application to aid during emergency,” in 2016 International Conference on Circuits, Controls, Communications and Computing (I4C), pp. 1–6, Bangalore, India, 2016.

[9] V. Tundjungsari and A. Sabiq, “Android-based application using mobile adhoc network for search and rescue operation during disaster,” in 2017 International Conference on Electrical Engineering and Computer Science (ICECOS), pp. 16–21, Palembang, Indonesia, 2017.

[10] A. Z. Zulkafi, S. Basri, L. T. Jung, and R. Ahmad, “Android based car alert system,” in 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), pp. 501– 506, Kuala Lumpur, Malaysia, 2016.

[11] “bSafe Android app,” October 2018, https://play.google.com/ store/apps/details?id=com.bipper.app.bsafe&hl=en.

[12] “GoSuraksheit Android app,” October 2018, <https://gosuraksheit.soft112.com/>.

[13] R. Jadhav, J. Patel, D. Jain, and S. Phadhtare,“Emergency management system using android application,” International Journal of Computer Science and Information Technologies, vol. 5, no. 3, pp. 2803–2805, 2014.

[14] D. MacLean, S. Komatineni, and G. Allen, Pro Android 5, Apress, 2015.

[15] W. Jackson, Pro Android UI, Apress, 2014