TNI Journal of Engineering and Technology
Vol.5 No.1 January - June 2017

Design of Bus Tracking Framework Based on Android Mobile Application by Using GPS

Sarunyoo Srivichitranond[1], Bunphong Mannnaitham[2], Techit Taesrikul[3], Thagoon Jamnongtanachot[4], Teerapat Nirapienranant[5], Prarinthorn Charoenying[6], Thanawat Chutigriwan[7], Thongchai Kaewkiriya[8]

Faculty of Information Technology, Thai-Nichi Institute of Technology
1771/1 Pattanakarn Rd., Suanluang, Bangkok, Thailand
[1]sr.sarunyoo_st@tni.ac.th
[2]ma.bunphong_st@tni.ac.th
[3]ta.techit_st@tni.ac.th
[4]ja.thagoon_st@tni.ac.th
[5]ni.teerapat_st@tni.ac.th
[6]ch.prarinthorn_st@tni.ac.th
[7]ch.thanawat_st@tni.ac.th
[8]thongchai@tni.ac.th

***Abstract*— In Thailand, bus is one of the public transport choices which have lots of customer every day. But most of the time bus does not arrive on time. The objective of this paper is to design and develop an application for tracking a position of a bus, which can help passengers to make decisions. There are three parts of this system. First, the position of the bus will be sent to a server by GPS on smartphone via cellular network every 20 seconds. Second, the server will gather data from the GPS and calculate the average speed of the bus within 20 minutes. The last part, the application which operates on Android can be used to check the position of the bus by making a mark on Google Maps, and calculating the estimated time of the bus arrival to the passenger's' position. The last step is to present the synthetic framework to users for appropriate evaluation. The result from users are very appropriate with the average score equal to 7.9/10 (S.D. = 1.22)**

***Keywords* - Bus Tracking, GPS, Cellular Network, Android, Smartphone, Haversine Formula, Google Maps API**

I. INTRODUCTION

Time is a valuable asset and usually is the most important factor to decide which transport is suitable. Bus is a public transportation option which is cheap in price and eliminate sparking problems. But the arrival time is unpredictable and information about its location is not distributed among passengers. So it's easy to make a wrong decision due to lack of information.

GPS technology is a technology which can be found in everyday life. Usually it is used for tracking position and direction of cars, boats, planes and many other things. When the location is known, navigation system is an adaptation of GPS technology. Nowadays, smartphones are affordable and still embedded with a GPS receiver which also supports connection to the internet via cellular network.

In the past, Thongchai Kaewkiriya [1] proposed a framework of Controlling and Tracking Vehicles Using Mobile GPS Based on GPRS/3G Networking. Which limited location data distribution only for vehicle's owner.

Wichet Darakai [2] has made a software which processes geographic coordinates into position on a map and made comparisons with other software. The self-made software only has a little dislocation. The only limitation of this software is its input method which used SMS via GSM network. This method is not suited with vehicles tracking which needs to send coordinates constantly and active for a long time on daily basis.

Surachet & Settha [3] proposed Tracking and Time-Estimating System for Electric Vehicle via Smart Phone over 3G Network. This system also applied Fuzzy Logic to reduce data transmission rate. The restriction of this system is its only bound to campus area.

David Breed [4] proposed human monitoring systems using a mobile device such as cellular phone or PDA. The device requires at least one kind of sensor to gather surrounding data, such as signal strength, and send it to a control centre. The tracked human doesn't need to have any interaction with the device. This system can track the device by using embedded GPS or DGPS receiver. This system can also use other sensors that are provided with the device such as a microphone to enable voice recognition features. However, this system did not use voice recognition to recognize human voices but used to compare with specific patterns, such as an accident's sound patterns. If voice recognition can match incoming sound with specific patterns, the device will send predefined signals to control centre to report about accidents near the device holder. The only limitations of this paper are patterns of sound which use a lots of resource, depending on how much patterns are used to compare with incoming sound.

Alan J. Dabbiere [5] proposed methodology and tools for utilizing a GPS tracking system when gathering additional data such as activities and working patterns of employees of a warehouse. This is done by collecting data of geographic coordinates, height and time. The collected data is used for making a statistic of a specific activity to reduce wasting time and help to evaluate an employee's salary depending on difficulty of an activity. Restriction of this method is unable to tracking warehouse's tools usage so activity tracking might not be reliable.

Chaiyaporn Khemapatapan [6] proposed to have a car's position tracking and route tracing system by satellite signal. This paper objective is to create a prototype for a real-time car tracking device that has temporary memory in case of transmission is not available for a moment. The device will send predefine data along with geographic coordinates when data transmission networks such as GPRS is available. In additional of temporary memory, this system also supports connection with a database for each car. The database purpose is to analyse and evaluate routes of logistic. The prototype device also come in cheap price. Each device cost about 110 $US. The limitation of this device is the transmission delay when used in urban areas, and also have a dislocation for five meters.

With the papers mentioned above as references. This paper objective is to propose system which used Android application for tracking public transport such as buses and distribute location of buses to passenger via 3G network. This system also provides average speed of buses from last 20 minutes. With buses' location and speed information the passengers can manage their traveling more efficiently.

II. BACKGROUND

A. GPS [7]

GPS or Global Positioning System is a system used for identifying geological coordinates of GPS receivers. GPS receiver needs to receive satellite signals from at least 4 satellite. After receiving satellite signal, the GPS receiver can calculate positions and return values such as latitude and longitude. These returned values can apply for navigation systems. Direction and speed data can be uses to estimate arrival time.

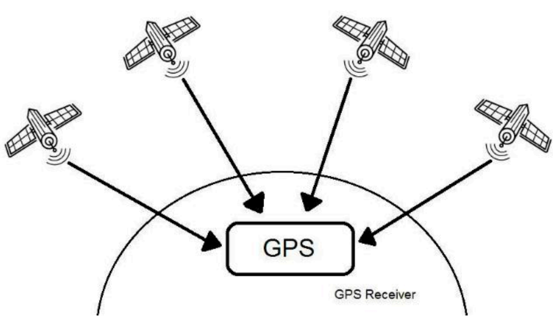

Fig. 1 GPS Receiver receive 4 satellite signal

B. Smartphone [8]

The smartphone is a type of mobile phone which comes with operating systems such as Android, iOS, Tizen, BlackBerry and Windows Phone. All of the mentioned operating systems are a combination between personal computers and mobile phones. Smartphone usually include GPS navigation units and support internet connections. The smartphones used in this paper have three requirements.

1.GPS or DGPS must be available.

2.Internet connection via 3G network.

3.It is an Android Smartphone.

C. Haversine Formula [9]

This is a formula used to calculate distance between two point on earth by using latitude and longitude.

$$d = 2r \arcsin\left(\sqrt{sin^2\left(\frac{Ø2-Ø1}{2}\right) + cos(Ø1)cos(Ø2)sin^2\left(\frac{\lambda2-\lambda1}{2}\right)}\right)$$

Fig. 2 Haversine Formula

Variable d is distant in km.

Variable r is earth's radius in km (Average is 6,371 km).

Variable $\Phi1$, $\Phi2$ are first and second latitude. Variable $\lambda1$, $\lambda2$ are first and second longitude.

D. API [10]

API or Application Program Interface is used to create software applications. API controls software components to work in a specific way and are also used to make a Graphic User Interface(GUI). API controls interaction between the application and operating system. Operating system's developer usually distributes their OS API for application developers to create application on their OS. An OS can have multiple API sets depend on hardware.

E. Google Maps [11]

Google Maps is one of google services for developer. This service provides map data and is supported by many platforms such as webpages and smartphones.

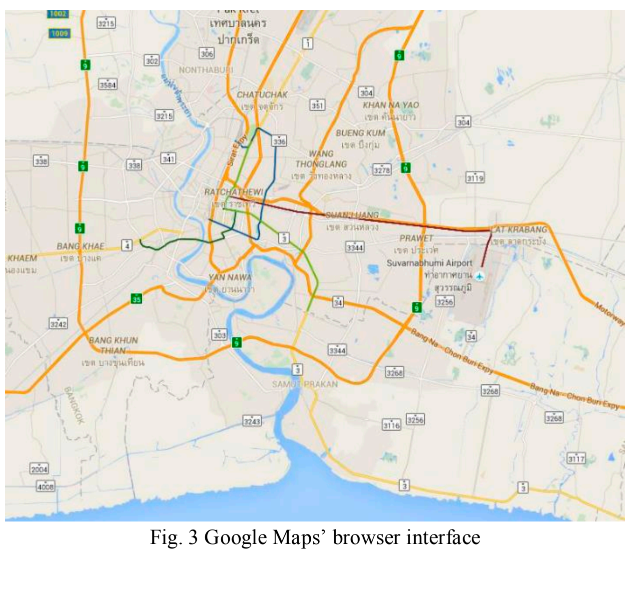

Fig. 3 Google Maps' browser interface

F. 3G/UMTS [12]

UMTS or Universal Mobile Telephone System is one of the 3G (Third Generation) standards. UMTS is developed under IMT-2000 framework by ITU. This standard required a minimum speed of 384 Kbit/s and UMTS use 850, 900, 1800, 1900 and 2100 MHz frequency as transmission medium, so UMTS is better than 2G in transmission quantity which means better image and voice data.

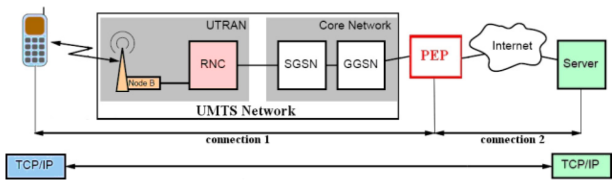

Fig. 4 Process Diagram of UMTS [13] consist of RNC (Radio Network Controller), SGSN (The Serving GPRS Support Node), GGSN (The Gateway GPRS Support Node), PEP (Performance-Enhancing Proxy)

G. Android Studio [14]

Android Studio is an application development tool by using IDE (Integrated Development Environment) which is officially provided by Google. Android Studio is based on IntelliJ IDEA which is another Java IDE development tool, but has different features such as a flexible system development and drag and drop interface.

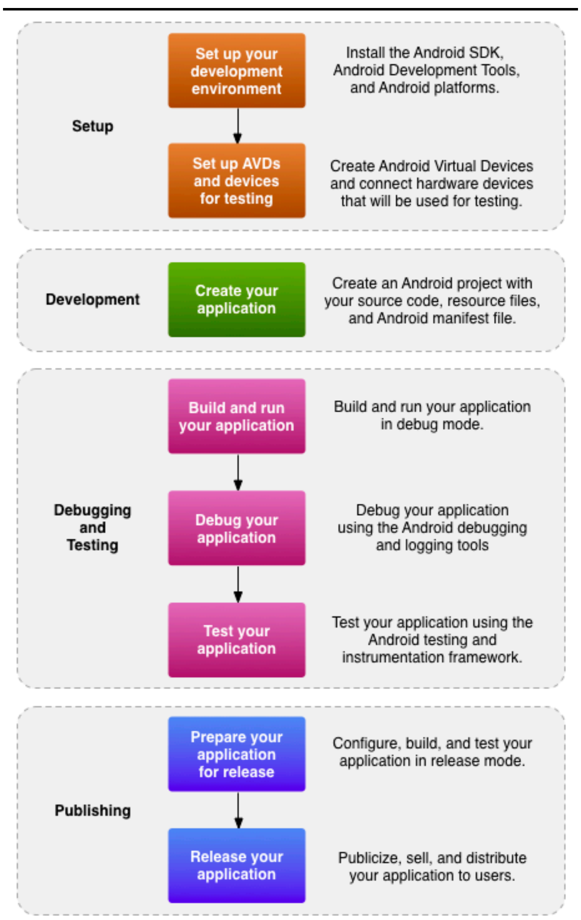

Fig. 5 Basic processes of application developing

III. METHODOLOGY

A. Design of Framework

The framework of bus tracking system by smartphone via 3G network consist of 4 fundamental parts. The first is bus attached with android smartphone. The second is a 3G network provider in Thailand such as AIS, DTAC, True, etc. The third is a server used for store buses position and timestamp on database. Fourth is an android application which uses data from server to create marks on Google Maps which represent bus positions.

As mentioned in figure 6 on the next page this framework divided into four parts.

The first part is a tracker side. A tracker side needs to install an android smartphone on the buses. The smartphone must have a GPS receiver and can return geographic coordinator of itself correctly and support data transfer via 3G networks. The set of data sent from smartphones consist of latitude, longitude, timestamp and bus id. This data has an update interval of 20 seconds.

The second part is method to connect tracker smartphone to the internet for data transfer from first to third part. The solution is access to internet via 3G network provided by service provider. To send data through 3G networks, a SIM Card is required. This prototype used a prepaid voice SIM Card.

The third part is the server which stores every data from the first part in form of a database and sends desired data to the passenger application.

The last part is the passenger side application. The user needs to enter the desired bus number to filter data from servers, then use the data received from server to mark the bus position on Google Maps. This application has a five seconds refresh rate while the application is active.

B. Scenario

Attach tracker side smartphone to a bus and set the bus ID manually. Format for bus ID is first digit for bus type, such as standard bus or air conditioned bus. Next 3 digits is bus's line number. And last 3 digits is number that can specific the bus from same line number. After finished tracker ID setting, next is check the internet connection. If smartphone can connect to internet without any problem, activate the application. Tracker application will send current latitude, longitude and bus ID to server immediately and send again every 20 seconds. Then select desired bus number on client application. The client application will fetch geographic coordinate which match with user input from server. The client application will query all positions up to sixty-one positions for each bus to calculate the average speed from the last 20 minutes and display along with the bus's position mark on Google Maps.

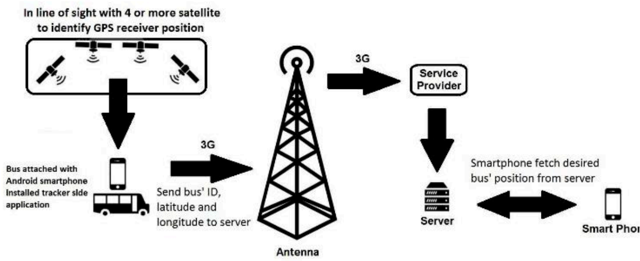

Fig. 6 Bus tracking by using smartphone via 3G network framework

C. Tracker Application Flowchart

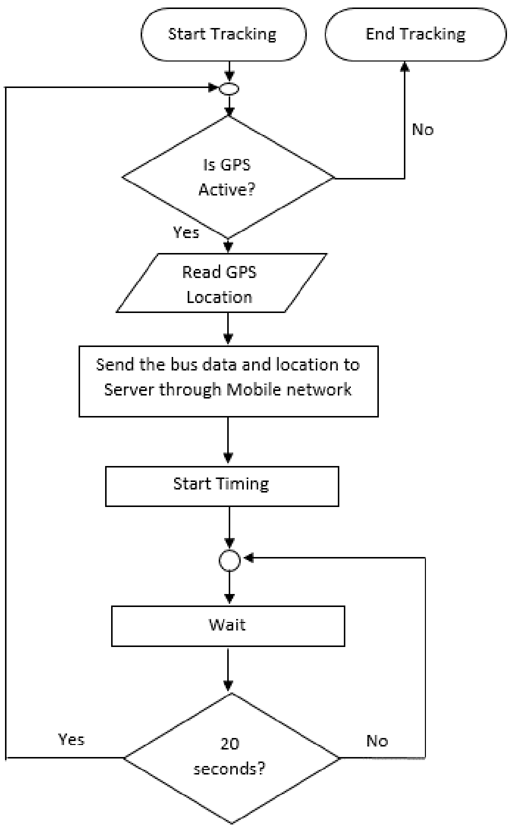

Fig. 7 Tracker Application Flowchart

Figure 7 is a flowchart after the tracker application has entered bus ID. After setting the application status to active, the application reads the geographic coordinator and sends the latitude, longitude and bus ID immediately. Then it starts the timer to update the bus's position every 20 seconds. The application stops tracking when status is set to inactive.

D. Client Application Flowchart

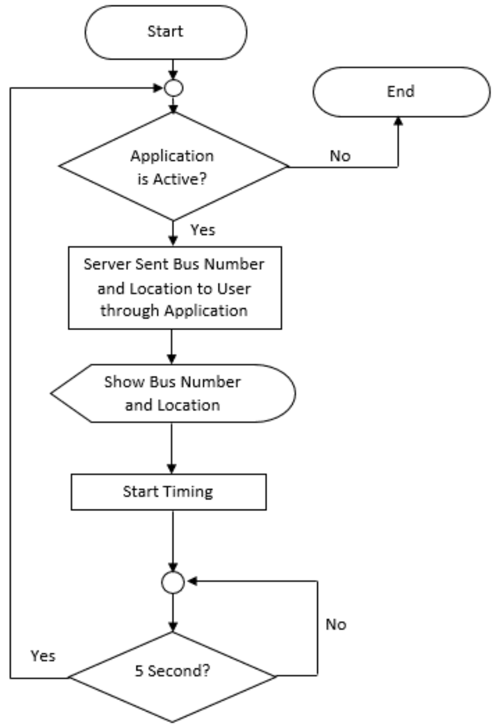

Fig. 8 Client Application Flowchart

Figure 8 is a flowchart of the client application. Similar to the tracker application, the client application starts working when the status is active. Then user select a desired bus number to filter the bus data and fetch the specific bus number from the server. The client application fetches two records of

data from the server for each bus. Then start timer for fetching bus data every five seconds. Application stops updating when closed.

E. System Context Diagram

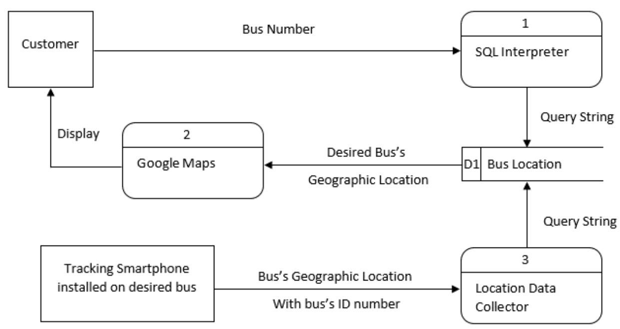

Fig. 9 Bus Tracking System Context Diagram

Figure 9 is a context diagram of a prototype system. It shows client or customer activity is not directly relating to tracker application. Client application used entered number to filter for desired bus. The client application also acts as a SQL interpreter to send a query string to fetch desired data from the server. The client application uses fetched data to make marks on Google Maps to represent the bus's location and calculates the average speed for each bus. The tracker application interprets geographic coordination and bus's ID to SQL and send data to server every 20 seconds for each tracker.

IV. INTERFACE

A. The tracker side application

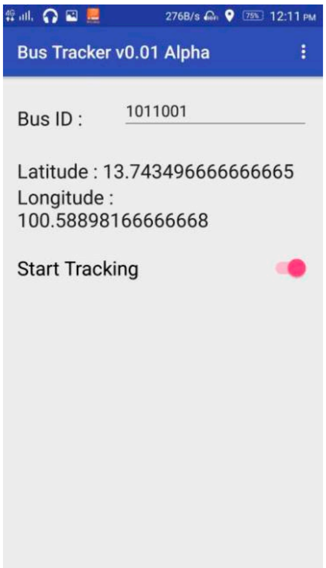

Fig. 10 The tracker side application

The tracker application interface shows bus id, current latitude, current longitude and application status which is active in figure 10. In the prototype version the user need to enter bus id and need to change the tracker application manually.

B. The server

Bus Location Log   Clear Database

Log Number	Bus ID	Latitude	Longitude	Timestamp
454	1133001	13.740026666667	100.60306833333	2016-03-12 07:19:37
453	1133001	13.740903333333	100.60127666667	2016-03-12 07:19:17
452	1133001	13.741296666667	100.60067833333	2016-03-12 07:18:56
451	1133001	13.741253333333	100.60055333333	2016-03-12 07:18:36
450	1133001	13.741298333333	100.60032333333	2016-03-12 07:18:15
449	1133001	13.741428333333	100.59949	2016-03-12 07:17:55
448	1133001	13.741906666667	100.59738833333	2016-03-12 07:17:34
447	1133001	13.742368333333	100.595305	2016-03-12 07:17:14
446	1133001	13.742486666667	100.59452333333	2016-03-12 07:16:53
445	1133001	13.74279	100.59323833333	2016-03-12 07:16:33
444	1133001	13.743281666667	100.591005	2016-03-12 07:16:12
443	1133001	13.743686666667	100.58935833333	2016-03-12 07:15:52
442	1133001	13.743848333333	100.58870666667	2016-03-12 07:15:32
441	1133001	13.744066666667	100.58740333333	2016-03-12 07:15:11
440	1133001	13.744405	100.58604	2016-03-12 07:14:51
439	1133001	13.744581666667	100.58541833333	2016-03-12 07:14:30
438	1133001	13.744816666667	100.58449333333	2016-03-12 07:14:10
437	1133001	13.744948333333	100.58414666667	2016-03-12 07:13:49
436	1133001	13.744948333333	100.58414666667	2016-03-12 07:13:29
435	1133001	13.74499	100.58381666667	2016-03-12 07:13:09
434	1133001	13.745011666667	100.583515	2016-03-12 07:12:48
433	1133001	13.745195	100.58272333333	2016-03-12 07:12:28

Fig. 11 Data samples on the servers

Figure 11 is an interface for developer which fetch all data from the database to check the data flow from trackers.

C. The client side application

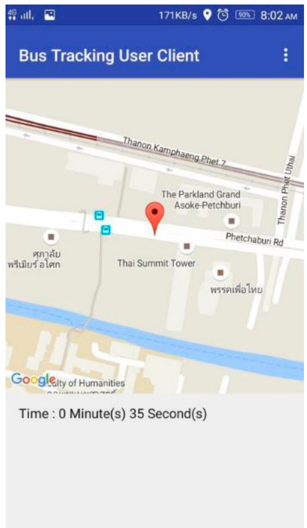

Fig. 12 The client side application

Figure 12 is the client application interface after user input desired bus number. The application fetch records from the database according to user's input. Figure 12 interface shows estimation of fastest bus arrival time. The application also shows bus position on the map.

V. EVALUATION

TABLE I
APPLICATION'S EVALUATION RESULT

No.	Details	x	S.D.
1	Client application is easy to use	8	1.18
2	Estimate arrival time accuracy is satisfying	8.3	1.27
3	Client application interface is easy to read	7.7	1.19
4	Accuracy of tracker application	7.6	1.02
5	System reliability	8	1.41
6	Easy to implement	8	1.26
	Total	7.9	1.22

The evaluation is done by demonstration the system processes to user and gives a questionnaire according to the table 1. The questionnaire is based on scales of 10. The table 1 also shows that the estimate arrival time is very satisfying ($\bar{x}$=8.3), while the accuracy of tracker application gains the lowest average score ($\bar{x}$=7.6), which means this prototype needs more of fine tuning on the tracker application.

VI. CONCLUSIONS

The objective of this paper was to design and develop an android application for tracking the position of a bus, which can help passengers to make decisions. The system consists of 3 parts. The first part is tracker part which is tracker applications that send location for each bus to the server. The second part is the server which stores data from tracker applications. The last part is client application which fetch data from the server according to bus id that input by the user. The client application also calculates the average speed and estimation of arrival time of the fastest bus according to user's position. According to user evaluation, the result of evaluation is very appropriate with the total average score of 7.9/10 (S.D.=1.22)

The limitations of this paper are the calculations, such as client application calculate the distant between bus and user in straight line along the earth's surface which can contradict with the actual road. Another major flaw is the client application don't actually calculate on the bus's route which means user need prior knowledge about that bus to make more precise decisions.

Benefit of this paper is application developers can adapt this framework for public transportation with more fine tuning to appropriate with specific transport.

REFERENCES

[1] T. Kaewkiriya, "Controlling and Tracking Vehicles Using Mobile GPS Based on GPRS/3G Networking", *Journal of Romphuk*, Vol. 32, No.2, pp.85-102, 2014.

[2] W. Darakai. (2015). *GPS Tracker page on Blogspot*. [Online]. Available: http://darakai.blogspot.com/2011/09/gps-tracker.html

[3] S. Kanprachar and S. Thangkawanit, "Tracking and Time-Estimating System for Electric Vehicle via Smart Phone over 3G Network", Naresuan University, Phitsanulok, Thailand, Sep. 2014.

[4] D. S. Breed, "Monitoring Using Cellular Phones", U.S. Patent 8035508 B2, Oct. 11, 2011.

[5] A. J. Dabbiere, "Methods and devices utilizing a GPS tracking system", U.S. Patent 6226622 B1, May. 1, 2001.

[6] C. Khemapatapan, "Car Tracking and Position Verifying System by Satellite Signal", Dhurakij Pundit University, Bangkok, Thailand, 2012.

[7] "What is GPS?," *The Library of Congress*, 2016 [Online]. Available: http://www.loc.gov/rr/scitech/mysteries/global.html.

[8] "Smartphone page on Wikipedia," *wikipedia*, 2015. [Online]. Available: https://en.wikipedia.org/wiki/Smartphone.

[9] "Deriving the Haversine Formula," *Ask Dr. Math*, 2016. [Online]. Available: http://mathforum.org/library/drmath/view/51879.html.

[10] B. Vangie, "API - application program interface," *Webopedia*. [Online]. Available: http://www.webopedia.com/TERM/A/API .html.

[11] "Google Maps APIs," *Google Maps APIs*, 2015. [Online]. Available: https://developers.google.com/maps/documentation/.

[12] "The UMTS World website.", 2015. [Online]. Available: http:// www.umtsworld.com.

[13] "TCP characteristics.", 2016. [Online]. Available: http://zst.utp .edu.pl/Pliki/Technologie/TCP/.

[14] "Meet Android Studio.", 2016. [Online]. Available: https://developer.android.com/studio/intro/index.html.