Safety Monitoring System for Dementia Patients Using Smart Watch

Seokjoo Cho
Mechanical Engineering
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
lnxlht4j@gmail.com

Nathan Gronowski
Computer & Information Technology
Purdue University (undergraduate)
Indiana, USA
ngronows@purdue.edu

Bohyun Kim
Computer Science & Engineering
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
kbh5640@gmail.com

Namhyo Kim
Computer Science & Engineering
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
namhyo01@naver.com

Dongjin Shin
Computer Science & Engineering
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
ilovethisid@naver.com

Nayun Kim
French language & Literature
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
0912tiffy@gmail.com

Chaeyeon Song
Biomedical Engineering
Chung-Ang University (undergraduate)
Seoul, Republic of Korea
codus503@gmail.com

***Abstract*—The rate of dementia is increasing due to the increased average life expectancy and a lack of knowledge and cure in dementia. Over 47 million people have dementia globally, and this number is expected to triple by 2050. In these circumstances, we are seeking to make devices to help dementia patients and their caregivers. There are many problems that dementia patients might face, but this project decided to work on preventing dementia patients from getting lost. To do that, a tracking system with a warning if the patient gets far away can be effective. Among the devices, smartwatch can provide convenience to patients for its ubiquitous and familiar features. So, this project devised a smart watch to track dementia patient and warn them if they get far away. Our system consists of three components, which are the smartwatch application, the server, and the smartphone application. A smartwatch tracks the dementia patient with its GPS feature and sends data to server. The server connects the smartwatch and the smartphone application, and the smartphone warns caregivers if the patient gets far from home. A warning message is sent to the smartwatch as well, to warn the patient. Then, the smartwatch runs navigation to help patient get back to home. In our test, we successfully connected the smartwatch and smartphone. The patient and caregiver got warning when patient went out of boundary, and navigation was executed in patient's smartphone. By using a smartwatch and applying monitoring services, this project is expected to prevent dementia patients from getting lost and help their caregiver to find them.**

***Index Terms*—Alzheimer's disease (AD), Android application, API, dementia, geofence, GPS, smartwatch, wearable device, Wi-Fi**

I. Introduction

Due to the global phenomenon of average life expectancy going up and absence of complete cure to dementia, dementia cases are increasing globally. According to [1], over 47 million people have dementia globally, and this number is expected to nearly triple to 131 million by 2050. Aging is the most affecting risk factor in every kind of dementia and 96 percent of people with dementia caused by Alzheimer's disease (AD) are 65 years old or more in the US [2], [3]. AD is the most common cause of dementia and accounts for 60 percent to 80 percent of total dementia [1]. In the end of the AD, the dementia patients are led to death. However, there is no drug to completely stop the progressions of AD [1]. Even in the medically advanced country like US, the number of patients with Alzheimer-type dementia is estimated to be 4.7 million in 2010, and 13.8 million by 2050 [4].

There are many manifestations of dementia patients in earlier stage. For instance, they have cognitive problems such as forgetting how to use household technology. There are also many problems in later stage such as false belief systems or walking out of home in the middle of the night and getting lost [2]. Especially, 63 percent of dementia patients are suffered by wandering and people with dementia due to AD are more likely to suffer from it than vascular dementia [5], [6]. According to [6], wandering is associated with impaired spatial perception and cognitive impairment. The causes of wandering have not been clearly understood yet. In addition, even for people with dementia who have difficulty in moving, wandering happens to them. These problems make it harder for caregivers of dementia patients to care dementia patients than caregivers of other diseases [2], [3], [6]. For these reasons, the devices which help caregivers of dementia patients have been developed.

In many devices, the smartwatch was chosen to help caregivers as it has ubiquitous and familiar features for dementia patients [7]— [9]. By using smartwatch and applying tracking services [7]– [12], this project will prevent patients from getting lost, and will benefit caregivers by locating and warning patients.

II. Literature Review

Generally, there are three ways for developing android applications to monitor dementia patients. The ways are divided into using only a smartwatch [9], [10] smartwatch and smartphone together, [11] and only a smartphone [12]. Specifying location of patients is important in monitoring dementia patients. GPS can be used by receiving signals from different satellites, and android application can connect to Google Maps API [13] to get the location. Regarding the purpose of this paper, an Android smartwatch that can connect to Wi-Fi and has GPS sensor was used in this paper.

A. Android Applications for Monitoring Dementia Patients

The proper devices for dementia patients are those that is familiar and nonintrusive to dementia patients and has uncomplicated and reliable characteristics [8]. As mentioned by [8], Boletsis *et al.* [9] argued that a smartwatch can be used as a solution that breaks through existing challenges such as considering only time for dementia home-visit service rather than considering cognitive or physical changes of the patient. D. Shin *et al.* [10] developed the health management device which works by using the server and the smartwatch including a GPS sensor, a 3-axis acceleration sensor, and an ambient light sensor. By using a GPS sensor, the smartwatch could measure walking steps and find the location of patients. Alraddadi *et al.* [11] also developed the Android application for dementia patients and caregivers by using the smartwatch and the smartphone application. In their research, to connect Wi-Fi on the smart watch, dementia patients must carry their smart phone together. The devices can track the location of dementia patients and show the place that patients want to go. By using two smartphones with Android OS, Shinde and Chawan [12] proposed the applications for dementia with GPS tracking and check fall detection. The problems were that the smartphone of dementia patients needed to have a good battery power and that the smartphone should be carried by dementia patients.

B. Geofence in Tracking Services

Tracking services for dementia patients are the Location-Based Services (LBSs) in public safety category [13]. The virtual perimeter called geofence is usually used as checking whether the clients are in boundary or not [14]. Geofence can be applied to circular shape to calculate distance easily by using Haversine formula [15]. Geofence can be applied to monitoring whether dementia patients get out of the boundaries designated by their caregivers or not.

III. Implementation

A. Server

For the server, AWS-EC2 service as shown in Fig. 1 is being used to host a version of Ubuntu to hold databases and create a connection between the applications in this paper. Python programming language is selected as constructing server. For managing APIs requested from a smartphone and a smartwatch, there was a choice between Django and Flask, which are the common web frameworks of Python. Ghimire notes:

> It is evident that Django can be best fit for large-scale projects with the cost of the learning curve. Flask is best fit for the prototyping and small-scale projects but not limited to it [16, p. 32].

According to [16], Flask was chosen to create a server in this project. As setting standard API about HTTP request methods in the server, *GET* was used when reading data, and *POST* was used when writing data [17]. Therefore, a client should request through *POST* method when passing data that are personal like the location or the ID to the server. And the server transfers data to the client through *GET* method in the cases that the data are public.

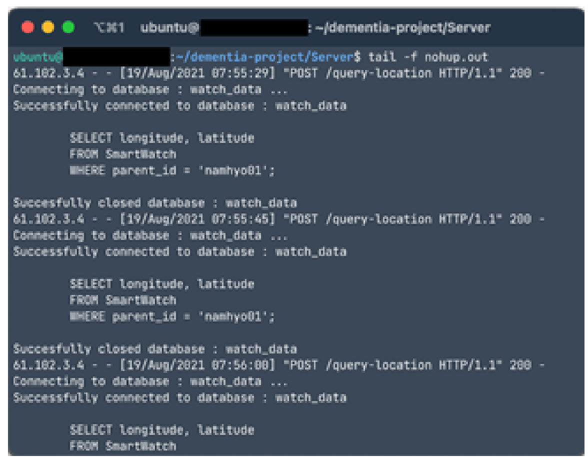


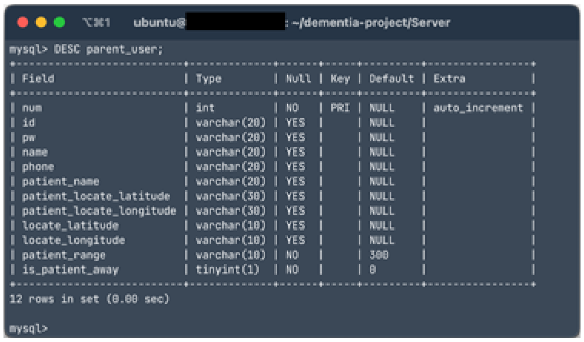
Fig. 1. AWS-EC2 server running Flask app.

B. Databases

As Flask does not have database abstraction layer, the databases should be selected between MySQL and SQLite [18]. MySQL was selected to handle multiple users [19]. MySQL was installed on the AWS-EC2 Ubuntu server, and two databases were created for the purpose of this project.

1) Credential Database: A credential database as shown in Fig. 2 was created to store usernames and passwords of users so that users can successfully login into the smartphone application.

2) Location Database: A database for the location data was created to store the data which are longitude and latitude as shown in Fig. 3. When smart watches store their location into the database, smartphones can read the location and then mark



⌥⌘1 ubuntu@[redacted]:~/dementia-project/Server

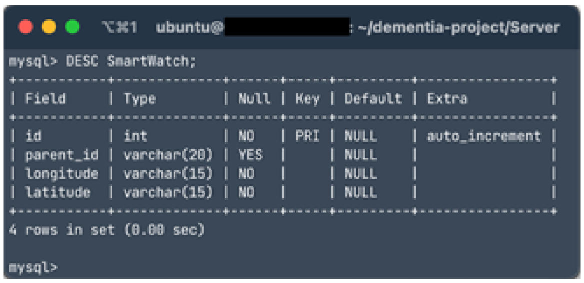
mysql> DESC parent_user;

Field	Type	Null	Key	Default	Extra
num	int	NO	PRI	NULL	auto_increment
id	varchar(20)	YES		NULL	
pw	varchar(20)	YES		NULL	
name	varchar(20)	YES		NULL	
phone	varchar(20)	YES		NULL	
patient_name	varchar(20)	YES		NULL	
patient_locate_latitude	varchar(30)	YES		NULL	
patient_locate_longitude	varchar(30)	YES		NULL	
locate_latitude	varchar(10)	YES		NULL	
locate_longitude	varchar(10)	YES		NULL	
patient_range	varchar(10)	NO		300	
is_patient_away	tinyint(1)	NO		0	

12 rows in set (0.00 sec)

mysql>

Fig. 2. MySQL Credential database.



⌥⌘1 ubuntu@[redacted]:~/dementia-project/Server

mysql> DESC SmartWatch;

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
parent_id	varchar(20)	YES		NULL	
longitude	varchar(15)	NO		NULL	
latitude	varchar(15)	NO		NULL	

4 rows in set (0.00 sec)

mysql>

Fig. 3. MySQL Location database.

the current location on the map. If the relationship between the smartphone and the smartwatch is 1:N or N:M, it is necessary for smartwatches to use the credential database to identify users.

C. Smartwatch Application

The purpose of the smart watch is to track dementia patients and show them the direction to home. In this case GPS feature and cellular feature were needed if possible. But the smart watch without cellular feature was used due to limited budget. The model called TicWatch S2 was chosen, which has GPS feature but not cellular one. TicWatch S2 can connect to a smartphone by using Wear OS by Google, which makes it easier to develop the smart watch application. These specifications of TicWatch S2 are shown in TABLE I.

TABLE I
THE SPECIFICATIONS OF TICWATCH S2

Specifications	**TicWatch S2**
Dimensions	46.6 mm x 51.8 mm x 12.9 mm
Operating System	Wear OS by Google
GPS	GPS, GLONASS, and Beidou
Sensors	Accelerometer, gyroscope, heart-rate sensor, low latency off-body sensor
Battery Capacity	415 mAh
Connectivity	Bluetooth v4.1, Wi-Fi 802.11 b/g/n

To track dementia patients and send data to the smartphone, we needed latitude and longitude data. Android developers guides [20] were mainly used in learning and performing the smartwatch application.

Since the smart watch did not have cellular feature, an alternative method was needed. If the smartwatch is connected to smartphone with Bluetooth, the smartwatch can get the location data from the smartphone, but in the case when the patient leaves the smartphone in his home, the smartwatch could not function properly. Thus, the smartwatch was implemented to get the location data through Wi-Fi connection. Of course, in this case if there is no Wi-Fi the smartwatch cannot locate the patient, but we assumed that in real case when we make this smartwatch as a product, we will make it cellular. Getting the location through Wi-Fi connection is just for the testing purpose, and in the case of connecting through cellular feature, the application should work without a problem.

Android Studio provided some samples for getting location, so we used an appropriate sample from android studio. Sample was ‘Location Updates (Java).’ It showed current address of the watch using ‘*FetchAddressIntentService*.’ We used ‘*getAddress*’ method to get the latitude and longitude for current location. We also added handler and runnable to update location on a regular basis. There weren’t any methods that connect to a server in a sample, so we created one. We made ‘*connect_server*’ method to connect to server and get and post data. The method uses HTTP connection in connecting with server, sending a request and waiting for a response. Request and response are sent as a JSON file format. Data of latitude and longitude and patient’s ID are sent with the request, and in response server gives location of patient’s home and whether the patient is out of boundary or not.

If the patient is out of boundary, watch application runs navigation using google map application. It uses *setNavigation* method to create intent for google navigation and run it. Unfortunately, watch didn’t have feature to turn on navigation on itself, rather, it just showed a message to use the smartphone to execute navigation. So, if the patient has his phone, the navigation is executed, and it would guide the patient to his home.

Fig. 4. App UI.

Fig. 5. Message for using navigation on a smartphone.

D. Smartphone Application

The main purpose of the smart phone application is to let the guardians track the dementia patient's current location. Second purpose is to send alarm when the patients get out of the accepted the range like home, hospital and so on. Location and boundary are set by application owner, mostly guardians. First, it is important to connect to the server and database to achieve main goal. Current location marked with latitude and longitude of patient is stored in location database when passed by the smart watch. *HttpURLConnection* is used to connect to the sever and start query to get data from database. Data can be received by JSON format and then parsed. This application is serviced only by android devices due to the difficulties of developing in an iOS environment. Our application was programmed by Android Studio.

1) Execute: Android Studio version is 4.3.1 and software version of android smart phone for test is Android 11.0 (API 30). This application service targets over Android 8.0 (API 26).

2) Login: Users can input username and password in the Login Page. There are two buttons which are Login and Signin. If the Login Button is clicked, then calling method is executed to find a user that has the same ID and password in the credential database. If user ID and password exist in the credential DB, 'Success' is returned. If not, 'Wrong' is returned. In Success case, 'SUCCESS' is printed, and users move on to Patient Location Page. In Wrong case, 'FAIL' is printed, and users must reinput ID and password. In Success case, to maintain the login, ID, name, patient's basic latitude, longitude, and the range are saved in the app by referring to *SharedPreferences* class and making a *SharedPreferences* class.

3) SignUp: Signup Page gets ID, Password, Name, Phone number, Patient's ID (for guardian). When user presses 'complete' button, calling method to the server is executed, and input data are saved in DB. Then the activity moves to the Location Register page.

4) Location Register & Location Setting: In Location Register Page and Location Setting page, user can set up Patient safety location. The primary reason of these pages is to send push alarms when patient gets out of the boundary. The only difference between Location Register page and Location Setting page is whether it can be canceled or not. Location Register page is proceeded obligatorily right after signing up. However, in the case of Location Setting page, user can modify the location range if they want. There is a cancel button only in the Location Setting page. Using Google Maps API, the map is shown at the WebView in the bottom of the page.

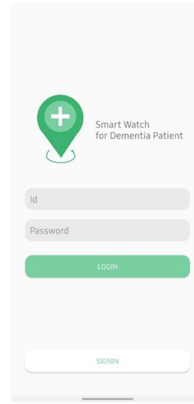


Fig. 6. Login activity.

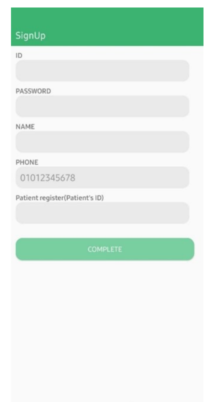


Fig. 7. SignUp activity.

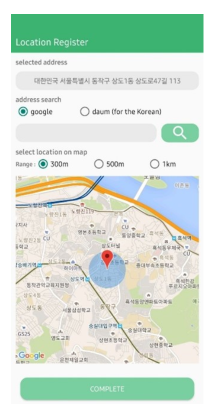


Fig. 8. Location register activity.

Fig. 9. Geocoder address (Google Maps API).

Fig. 10. Daum address API.

First, user can basically search address using Geocoder of Google Maps API or Daum address API (only for the South Korean). When using Google Maps, list is shown by ListView as Fig. 9, and user can see Korea or other countries. If listview is clicked, corresponding latitude and longitude is provided. Then the page shows exact address to selected address by using latitude and longitude from Google Geocoding. Center marker of Google map is changed according to corresponding latitude and longitude. If user uses Daum API, screen is shown by webview, and user can input address. However only Korean address works normally as Fig. 10.

Second, user can select the range of location by radio buttons as shown as Fig. 11. There are 3 options; 300 m, 500 m and 1 km. Distance is decided temporarily, so this can be modified while providing service.

5) Patient's Current Location: This Page shows the patient's current Location as shown as Fig. 12. Patient's location is provided in Location DB. It marks the location on Google map by longitude and latitude and shows the exact address of corresponding longitude and latitude by Geocoding. Patient's current location is received (latitude and longitude) every 13 seconds from the server and is marked on Google map. Geocoding indicates the current address, and displays latitude and longitude for more accurate information. Push alarm is provided if the patient is out of the range specified by the membership. 'Out of range' was measured through the Location class and *distanceTo* method. Likewise, this function keeps running in the background, so push alarms work when the patient is out of the specified range even if the app is turned off. Service Class is made for the function. Among this project's codes, the RealService class is the one that runs in the background, and the AlarmRecever class allows user to return to the background (run the RestartService class or RealService class according to the version) when user exit the app.

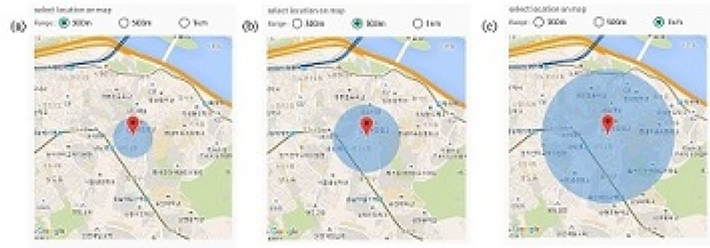


Fig. 11. The user can select the range of location. (a) 300 m range of the point. (b) 500 m range of the point. (c) 1 km range of the point.

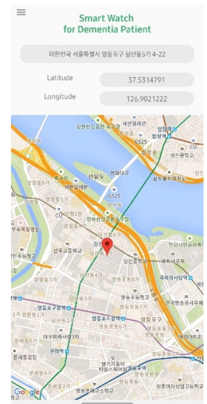


Fig. 12. Patient current location page.

IV. Conclusion

The goal of this project was to prevent patients from getting lost and show them way to their home. After the development, project's system using a wearable smart watch and a smartphone was tested and have successfully connected smart watch and smartphone. In the process server was used to exchange data from smart watch and smartphone. For the test, location of home was set to patient's home, and distance of 100 meter was used as location measurement unit. A person representing guardian tracked the patient, and person representing patient moved to nearby café to connect to Wi-Fi. As the smart watch connected to Wi-Fi, patient's location changed, and guardian got push alarm warning that the patient is out of the range. For the patient, patient also got a warning message, and navigation got started. By the results of the test, it is shown that when patient moves out of the range, both patient and guardian get noticed, and patient can go back to home using navigation.

This system also has few limitations to run successfully. First, it needs Wi-Fi to connect to server, so it doesn't work when Wi-Fi is not provided. Except for public stations or café or restaurants, in the street for example, when the Wi-Fi isn't provided, system cannot track patients. This problem can be addressed if the watch has cellular feature, but further programming is required. As being mentioned before, using Wi-Fi is just for testing purposes, and in further research, cellular models can be used to provide same service.

Second, we tried to make the service for multiple users to use, but due to the difficulties we had, we didn't manage to create application suitable for multiple users. Smartphone application has user credentials and login system, but user connection between smartphone and smart watch is not established. Instead, smart watch sends user ID from smartphone to confirm user. So, to connect smartphone application and smart watch, user needs to know ID from smartphone. There is also a problem in server. There wasn't enough time to test environment of multiple users, so the server might not work if there are too much traffics. In fact, our server doesn't have distributing system or an algorithm to handle overwhelming traffics, so server has high chance of not working if many people connect to server at the same time.

In conclusion, the initial goal of tracking the patient works well, and the system notifies guardians and warns patient, making patient return home. This service is expected to comfort guardians by showing the position of patient and prevent patient from getting lost by warning them and guiding them home. Still, the service cannot hold multiple users and needs Wi-Fi for running, so further research about handling multiple users or using cellular watch would extend usability for this project.

References

[1] Alzheimer's Assoc., "2018 Alzheimer's disease facts and figures." vol. 14, no. 3, pp. 367–429, Mar. 2018, [Online]. Available: https://alz-journals.onlinelibrary.wiley.com/doi/abs/10.1016/j.jalz.2018.02.001.

[2] Z. Arvanitakis, R. C. Shah, and D. A. Bennett, "Diagnosis and Management of Dementia: Review," *JAMA*, vol. 322, no. 16, pp. 1589-–1599, Oct. 2019.

[3] M. Prince *et al.*, "World Alzheimer Report 2015: The global impact of dementia. An analysis of prevalence, incidence, cost and trends", Alzheimer's Disease Int., Illinois, IL, USA, 2015. [Online]. Available: https://www.alzint.org/u/WorldAlzheimerReport2015.pdf

[4] L. E. Hebert, J. Weuve, P. A. Scherr, and D. A. Evans, (2013). "Alzheimer disease in the United States (2010-2050) estimated using the 2010 census," *Neurology*, vol. 80, no. 19, pp. 1778—1783, May. 2013.

[5] T. Hope *et al.*, "The structure of wandering in dementia," *Int. J. Geriatric Psychiatry*, vol. 9, no. 2, pp. 149–155, Feb. 1994.

[6] G. Cipriani, C. Lucetti, A. Nuti, and S. Danti, "Wandering and dementia," *Psychogeriatrics*, vol. 14, no.2, pp.135–142. Jun. 2014.

[7] M. T. Raghunath and C. Narayanaswami, "User Interfaces for Applications on a Wrist Watch," *Pers. and Ubiquitous Comput.*, vol. 6, no. 1, pp. 17–30, Feb. 2002.

[8] S. Cahill, J. Macijauskiene, A.M. Nygard, J. P. Faulkner, and I. Hagen, "Technology in dementia care," *Technol. and Disability*, vol.19, no. 2-3, pp. 55–60, Apr. 2007.

[9] C. Boletsis, S. McCallum, and B. F. Landmark, "The Use of Smartwatches for Health Monitoring in Home-Based Dementia Care," in *Human Aspects of IT for the Aged Population. Design for Everyday Life*, J. Zhou, G. Salvendy, Eds. Jul. 2015, pp. 15–26.

[10] D. Shin, D. Shin, and D. Shin, "Ubiquitous Health Management System with Watch-Type Monitoring Device for Dementia Patients," *J. of Appl. Math.*, vol. 2014, May. 2014, Art. no. 878741.

[11] L. Alraddadi, J. Fiaidhi, and S. Mohammed, "Developing an Android App for Dementia Patient Location: Prevention of Wandering Case study," *Int. J. of Bio-Sci. and Bio-Technol.*, vol. 9, no. 3, pp. 1–12, Jun. 2017.

[12] B. A. Shinde and P. M. Chawan, "Dementia Patient Movement Detection and Fall Detection using smart Phone Technology," *Int. J. of Adv. Technol. In Eng. and Sci.*, vol. 2, no. 07, pp. 155–160, Jul. 2014.

[13] M. Sinhal and A. Shukla, "Implementation of Location based Services in Android using GPS and Web Services," *Int. J. of Comput. Sci. Issues*, vol. 9, no. 2, pp. 237–242, Jan. 2012.

[14] D. Namiot, "GeoFence services," *Int. J. of Open Inf. Technol.*, vol. 1, no. 9, pp. 30–33, Sep. 2013.

[15] I. N. E. Indrayana, P. Sutawinaya, N. M. W. D. Pratiwi, P. M. Prihatini, and S. A. Asri, "Android-Based Child Monitoring Application Using A Smartwatch and Geofence Service," *J. Phys.: Conf. Ser.*, vol. 1803, Feb. 2020, Art. no. 012024.

[16] D. Ghimire, "Comparative study on Python web frameworks: Flask and Django," B. S. Thesis, Media Eng., Metropolia Univ. of Appl. Sci., Helsinki, Finland, 2020.

[17] L. Ibanez, *Development: Difference between the http requests POST and GET*, Accessed on: Month. Day, 2021. [Online]. Available: https://lazaroibanez.com/difference-between-the-http-requests-post-and-get-3b4ed40164c1

[18] F. A. Aslam, H. N. Mohammend, J. M. Mohd, M. A. Gulamgaus, and P. S. Lokhande, "Efficient Way Of Web Development Using Python And Flask," *Int. J. of Adv. Res. in Comput. Sci.*, vol. 6, no. 2, pp. 54–57. 2015.

[19] S. T. Bhosale, T. Patil, and P. Patil, "SQLite: Light Database System," *Int. J. of Comput. Sci. and Mobile Comput.*, vol. 4, no. 4, pp. 882–885, Apr. 2015.

[20] Google, "*Get started with Wear OS*." Android for Developers. https://developer.android.com/training/wearables (accessed July, 2021)