https://doi.org/10.2352/ISSN.2470-1173.2020.3.MOBMU-335

Performance analysis of Mobile Cloud Computing Architectures for mHealth app

D. Inupakutika, D. Akopian; The University of Texas at San Antonio; San Antonio, P. Chalela, A. G. Ramirez; University of Texas Health Science Center at San Antonio; San Antonio, TX, USA

Abstract

Mobile Health (mHealth) applications (apps) are being widely used to monitor health of patients with chronic medical conditions with the proliferation and the increasing use of smartphones. Mobile devices have limited computation power and energy supply which may lead to either delayed alarms, shorter battery life or excessive memory usage limiting their ability to execute resource-intensive functionality and inhibit proper medical monitoring. These limitations can be overcome by the integration of mobile and cloud computing (Mobile Cloud Computing (MCC)) that expands mobile devices' capabilities. With the advent of different MCC architectures such as implementation of mobile user-side tools or network-side architectures it is hence important to decide a suitable architecture for mHealth apps. We survey MCC architectures and present a comparative analysis of performance against a resource demanding representative testing scenario in a prototype mHealth app. This work will compare numerically the mobile cloud architectures for a case study mHealth app for Endocrine Hormonal Therapy (EHT) adherence. Experimental results are reported and conclusions are drawn concerning the design of the prototype mHealth app system using the MCC architectures.

Index Terms – *mobile cloud computing, healthcare, apps, cloud, performance*

Introduction

With the proliferation of smartphones, and the introduction of 4G networks, mobile technology has been experiencing progressive changes. Mobile networks are achieving higher throughputs and newer technologies such as 5G are providing higher rates. Hence, protocol traffic overhead does not appear to be a crucial issue anymore. Although smartphones are capable of processing many mobile apps, but they have limited battery lifetime (power resources). MCC paradigm is therefore attracting attention of researchers towards the technology that helps in lowering power consumption and delays without effecting the mobility and availability offered by the mobile networks. It integrates mobile devices with cloud computing and utilizes services provided by the cloud through the client (mobile device). It is a technology that computes complex operations or complete (or some portions of) resource-intensive apps on the distant or nearby cloud or servers, which cannot be handled on the mobile devices properly. The aim is to decrease device's power consumption and associated expenses.

Many sectors including cloud-healthcare systems can benefit from MCC. MCC healthcare systems [1] [2] are being built to capture and analyze real-time biomedical signals such as ECG, blood pressure or even simply self-reported symptom data for any chronic condition supportive care. Typically, these data can be synced through a personalized healthcare apps that are installed on users' mobile devices and health data can be synchronized to the healthcare cloud service for analysis and/ or storage. Different MCC architectures and a significant number of mobile cloud apps have been proposed by research initiatives. However, there is no standardized platform that aids wider deployment of apps. In [3], authors compare the performance of existing MC architectures in supporting resource-intensive apps. They numerically compare the performance of MC architectures across various MC apps. While there are huge benefits of using the MCC, there are some limitations such as the delays or latencies observed when the remote cloud services are accessed by the mobile devices from farther distances. This is where using the cloudlet between public or enterprise remote cloud and the mobile device reduces connection latencies and power consumption [4].

The motivation of why we need MCC and differences between traditional cloud computing and MCC are clearly explained in [5]. Different concepts related to mobile cloud, called remote, local, and the cloudlet are emphasized by them. Most common one is the remote cloud where a mobile cloud app leverages remote cloud for accessing different services that are complex. The local cloud is where the mobile devices communicate to each other to share the relevant data with other partner devices. It is used when a group of devices are working towards a common task. Cloudlet is another important and well-known, widely used concept that concerns with a nearby cloud server, which interacts with client devices over wireless local area network (WLAN). It has been initially proposed in [6]. It is discussed that the remote server allows the mobile devices to perform complicated processes that require data intensive transfers such as videos retrievals and streaming, games, face recognition and image processing etc. Data exchanges or transfers like these increase latency severely. They proposed cloudlet that is a server similar to a Wi-Fi access point to overcome such limitations. Users can save their virtual machine (VM) on their mobile devices in this cloudlet infrastructure and then run intricate tasks by launching their VM image to the cloudlet. While the cloudlet is not as powerful as a cloud server, but it is just one-hop over Wi-Fi connection between the cloudlet and the mobile device. Mobile devices can connect to the remote cloud if any cloudlet is not available in the infrastructure setting. Additionally, in the absence of any internet connection (Wi-Fi or Cellular LTE data), the mobile devices can use their on device computation capabilities.

Healthcare apps usually require greater amounts of computational and communication resources. These are also associated with accessing large quantities of data dynamically. Through cloudlet architectures, MCC could hence provide the required resources (computation or storage) at the right time and place. In this paper, we initially survey existing MCC architectures and study the performance of a prototype mHealth app using cloud-based and cloudlet-based architecture against a resource-intensive representative testing scenario.

The organization of this paper is as follows: In Related Work Section, the MCC background and related works are discussed.

MCC Architectures Survey Section surveys existing MCC architectures. In Case Study mHealth App Section, the prototype cloud-based and cloudlet-based mHealth app are discussed. In Experiments and Results Section, the testing and the simulation environment is explained followed by the implementation results and measurements corresponding to the metrics. Conclusion Section concludes our study and provides possible future research directions.

Related Works

MCC has grown to be an important technology lately with the addressing of mobile devices' challenges such as limited battery life (power resources), latency, bandwidth costs etc. There are significant number of related work in many domains including finance and health that found the usefulness of MCC. A prototype implementation of cloudlet architecture is presented in [7]. Advantages of such architecture in real-time apps is shown. As mentioned, cloudlet is fixed near a wireless access point in a simpler approach. However, in [7], it has options to choose the available cloudlet resources dynamically.

To reduce network delay and power dissipation for apps such as multimedia-based intensive apps, a large scale MCC model was set up in [8]. This model has an added advantage that considers the mobility of mobile users, which allows mobile users moving in this area to be connected to remote cloud services with less bandwidth needs and without compromising on the quality of service (QoS). The work in [9] analyses the impact of using cloudlets with MCC on interactive apps that include video streaming. The two models were compared based on the system throughput and data transfer delay metrics, where in majority of scenarios, the cloudlet-based model proved to be better than the cloud-based one.

The spread of cloud computing infrastructure to meet the ever increasing demand for computation and communication, operating data centers consuming huge amounts of power increase, which raises concern for energy conservation. To address this, resource management and optimization policies in the cloud that include VM like migration, server consolidation and virtualization are proposed in [10].

Generally, mobile devices can perform a task that cannot be handled locally with computation offloading that is one of the important features of MCC. The main idea is computation performance improvement and to save power or energy while the choice of performing a task locally or remotely is not straight forward. Offloading tasks to the cloud does not necessarily deliver the best performance always due to huge volumes of data transfer involved as contended in the work in [11].

The objective behind our study is the cloudlet concept, primarily for data storage and retrieval or transmission (in the discussed case study). In [12], the cloudlets are used in environments with limited internet connection to decrease the end-to-end latency for the latency-sensitive and the resource-intensive apps. While, authors in [7] used them for the real-time immersive apps, where they propose a framework to optimize the app performance at run-time as well as for determining whether to offload or not and which parts of computation.

As for the cloudlet architectures, the two-tier architecture was predominantly used since its inception. However, in recent years, the multi-tier architectures have also been considered as in [13] [14]. The work in [15] proposes a multi-tier architecture for the big data apps, where the cloudlets are placed in the middle tier. The authors in [16] highlight that the cloud communication solely does not provide satisfactory performance for the resource-intensive and delay-sensitive apps (like mHealth). Decision about handling the client request on cloud or cloudlet was formulated as an optimization problem and solved by a linear programming method. In our work, we are evaluating the performance of a cloud-based mHealth app for improving adherence in breast cancer patients who received Endocrine Hormone Therapy (EHT) with its cloudlet-based implementation.

MCC Architectures Survey

While MCC technology has a lot of potential application, there is no one single standardized architecture. Of the various architectures proposed by researcher, each one of them tries highlights a certain prerequisite i.e. power consumption, mobility, delay or response times etc. The methodology, system architecture and the services used by mobile devices vary. This section surveys existing MCC physical and application layer architectures. The choice of the MCC architecture depends on the technical design choices based on the application as well as the availability of infrastructure components at the facility.

MCC physical architectures include the following:

1. Cloud computing with mobile terminals

Cloud services like storage and computation are deployed in a remote cloud server as in conventional cloud computing architectures. Mobile devices such as smartphones are the terminals that connect to the internet over LTE or UMTS or GPRS mostly. Some of the advantages of this architecture include availability, handover, location privacy and native encryption. Leaving the network area in such architecture causes drop in the connection and the cloud service.

2. Virtual cloud computing provider

To share processing load, a virtual cloud is created from peer-to-peer (P2P) connected mobile devices over an ad-hoc Wi-Fi connection. P2P nodes participate when in the Wi-Fi range. After the selection of participant mobile devices in this architecture, the job is divided into tasks and is offloaded to a participant mobile device. Requestor consolidates the final result when the mobile device sends its processed data.

3. Cloudlet

Cloudlet servers are typically collocated with Wi-Fi hotspots and installed in densely populated areas. Users with mobile devices can connect to these servers and offload the task processing. Depending on the availability of infrastructure and organization capacity, cloudlet architectures can be of below three types [13]:

a. Single-tier

This architecture is suitable for organizations with their own cloudlet infrastructure where the on premise cloudlet hardware responds to users' mobile devices' service requests. Mobile devices and the cloudlet are in the same WLAN in this architecture.

b. Two-tier

This architecture integrates a cloud tier to the single-tier architecture setup for users to be able to offload their tasks completely through a Wide Area Network (WAN) connection. Cloud services could be on Amazon web services, Microsoft Azure or Google Cloud. The major difference between two-tier and single-tier is the availability of cloud service or not respectively.

c. Two-tier with load balancer

This architecture is feasible when the MCC facility owners are also part of the same organization. Hence, the common infrastructure can contribute to the overall system design but based on organizational constraints. Load balancer in such architecture handles the task requests from users' mobile devices and reshuffles them to the available cloudlets. The communication overhead and the transmission delays are assumed to be negligible. Mobile device can be redirected to any cloudlet in this architecture unlike in two-tier architecture without load balancer where cloudlets in the same WLAN can only serve the mobile devices.

4. Operator centric mobile cloud architecture (OCMCA)

A cloud server is installed within the network of mobile operator in this architecture that brings the cloud closer to the user, hence decreasing the delay. The services offered by mobile communication, availability, location privacy and the handover are not affected.

MCC application layer architectures include the following:
Unlike physical layer architectures as discussed above, application layer architectures are the offloading mechanisms that offload resource demanding apps or portions of the apps. These architectures address the issue based on which portion of an app is consuming more resources (power or battery life) on mobile device. We do not leverage application layer offloading mechanisms for the performance evaluation in this work. The application layer methods can be used in association with the physical layer architectures.

1. CloneCloud

CloneCloud [17] partitions application and assists mobile apps in offloading portions into the remote cloud. It employs dynamic profiling and static analysis to partition the apps in order to optimize overall execution cost. Execution cost typically comprises of computation, migration and the energy consumption. It uses application level VM to partition the app and runs one part on the mobile device itself and the other on the clone.

2. OpenMobster

OpenMobster [18] is a mobile cloud platform open-source project that provides mobile apps to move to MCC by abstracting that arise from the access of remote resources such as inter-application communication, local database, and network connection management, synchronization and queuing of messages to and from the cloud server, and the availability detection of the cloud server.

3. Some other similar application models are categorized as Performance-based, Energy-based, Constraint-based, and Multi-objective models whose details are explained in [3].

Case Study mHealth App

This section presents our prototype Android HT Patient Helper mHealth app for the evaluation of performance of the app's cloud-based and cloudlet-based version. We performed two assessment campaigns each for both the versions and observe the Firebase cloud (for cloud-based) and cloudlet database (for cloudlet-based) performance for a data intensive videos retrieval test scenario. We load the databases in both the versions and study the effect of Wi-Fi and 4G LTE (only with cloud-based) network connectivity under the load on the quality of user experience in terms of response time. We provide overview of the app with cloud and cloudlet assistance initially and then describe test scenario. As for the test campaigns that are described in [20], in the current work, we load Firebase in the cloud-based model while assessing the impact of wireless network connectivity (Wi-Fi and 4G LTE) and NoSQL database in the Wi-Fi cloudlet-based model with just Wi-Fi.

1. Overview of App and cloud-based model

Fig. 1 shows the cloud-based traditional MCC architecture model for HT Patient Helper mHealth that was originally developed as a cross-platform cloud-based app for self-monitoring, self-efficacy, and self-education of breast cancer patients who have undergone Endocrine Hormone Therapy (EHT). It is for patients to improve their adherence to medication [19]. Google Firebase was chosen as a cloud database server in the prototype implementation and its services can be accessed or delivered via wireless network technologies (Wi-Fi or 4G LTE). The region was *us-central.* The app has the following features overall:

a) Educational content that provides access to breast cancer specific symptoms, medication information, and reference or motivational videos from peers and doctors.
b) Tracking symptoms that enables patients undergoing therapy to keep track of their symptoms on a regular basis, go to place for relevant educational content as well as for progress updates in the form of summary graphs.
c) Push notifications to be sent to patients using the mobile app for sending reminders for medication, symptoms submission, doctor appointments, and motivational messages to keep them motivated.
d) Calendar for reminders as in (c), contact details of healthcare professionals, healthcare centers, support groups and technical support team.

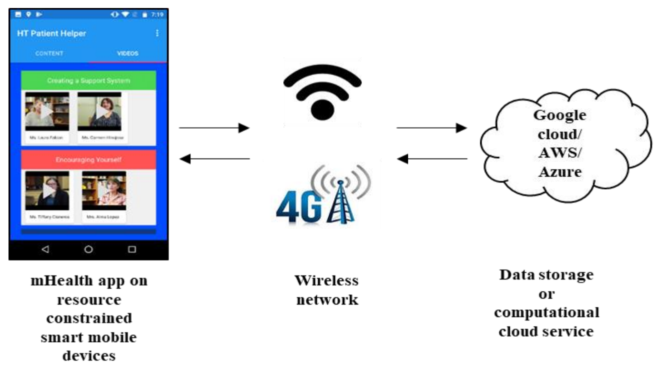

Figure. 1. Cloud-based MCC model for mHealth app

2. Wi-Fi Cloudlet-based model

Healthcare apps are time and delay-sensitive. Hence, it is required to offer crisp interaction to patients while allowing the mobile devices to have acceptable battery life yet responding patients with the requested services in a timely manner. Figure 2 shows the Wi-Fi cloudlet-based MCC model, where the mobile devices with app installed retrieves data from the cloudlet, reducing the transmission delay, and also reduces the mobile device power consumption. Cloudlet is a decentralized cloud computing element and is located just one wireless hop away. In the current work, relevant to the test scenario, cloudlet acts a data storage repository. The local cloudlet over Wi-Fi that is used has following specifications: Intel core i7, 3.2 GHz processor, with 16 GB RAM, and 500 GB SSD. A NoSQL database is setup with the app-based data models.

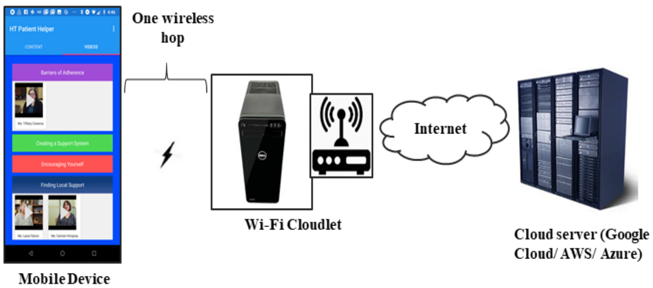

Figure. 2. Wi-Fi Cloudlet-based MCC model for mHealth app

3. Test Scenario

In the current work, we model the test scenario for one of the core features of medical reference apps as the case study app that involves retrieving videos of medication related information, tips to manage symptoms and procedures from peers and doctors. The test scenario consists of the navigation sequence: Splash page → Login page → Home Page → HT Learn More (educational content feature) → Videos tab → Video categories → Videos.

Experiments and Results

This section discusses the measurements results from the experiments that we conducted on real Android device to evaluate the two MCC architectures. We evaluate the performance of our prototype mHealth app in terms of response times, for different network configurations of the cloud-based and cloudlet-based models. We compare offloading the app database back-end to different sites, including the Firebase cloud database, and a local cloudlet over Wi-Fi. We collected the raw trace data by timestamping critical events of the app based on the described test scenario, and calculated response times for the videos retrieval based on the timestamp of the response from database (remote or on local cloudlet) is received and the timestamp of request sent from the app. Additionally, we calculated the power consumed by the mobile device during video data retrieval and waiting as a quantifiable metric along with the response time. An architecture that achieves lower response time and lower power consumption is better suited for the application. We leveraged PowerTutor [21] to measure the consumed power during the experiments.

1. Experimental Setup

The specifications of the hardware used in our experiments are shown in Table1. We have set up a desktop machine to represent a cloudlet. The list of testing tools and application programming interfaces (APIs) used are also described in [20]. Each experiment typically runs for 5 minutes. Android mobile device stores the raw trace data collected from the experiments for further analysis of getting the 95th percentile of the response times based on the configuration.

Table 1: Hardware Used in Experiments

	CPU	RAM
Test Computer	2015 Macbook Pro 15" running macOS Mojave, Processor 2.5 GHz i7	16 GB
Android Phone	Nexus 6P Octa-core (4x1.55 GHz Cortex-A53 & 4x2.0 GHz Cortex-A57) running android 8.0 Oreo	3 GB
Cloudlet	Intel core i7, 3.2 GHz processor	16 GB

2. Simulations and Measurements

For both cloud-based (default version) and Wi-Fi cloudlet-based versions of the app, the video sections are retrieved from the corresponding backend databases consisting of data models. With the increase in the database load, the response time should increase. We conducted experiments on with mobile device in a full battery regular mode. For every simulation based on Wi-Fi connectivity with both MCC model architectures and 4G LTE with cloud-based architecture, we load corresponding database by querying the entire *videos* node at different time intervals. Hence, the backend services run on a Wi-Fi cloudlet machine, and the Android phone is used as a client device.

The simulation scripts are run on the test computer mentioned in Table 1, which provides us the ability to control different threads and simultaneous instances of the app to simulate multiple users, user interface (UI) interactions based on the test scenario and submit requests to concerned database. The mentioned parameters are adjusted to achieve the required database load in the increments of 25% from no load to 100% load. Both the response time (in terms of raw trace data collection) and power consumption are measured on the mobile device. Response time data is further processed on the test computer for the response time plots, mean and standard deviation values.

Table 2 shows the measurement results across the two architectures for the app indicating the mean and standard deviation values of the response times along the power consumed in each setting. The values clearly indicate that the Wi-Fi based cloudlet model performs better in executing the videos retrieval in the app and has lower response time and consumes less power, showing the effect of retrieving data from a server at one wireless hop distance as opposed to the cloud database in the *us-central* region.

Table 2: Delay and Power measurement results across different architectures

MCC Architecture	**Cloud-based**	**Wi-Fi Cloudlet-based**
Wi-Fi (Mean/ std. dev (in ms))	328.28/ 992.84	286.84/ 863.77
Wi-Fi (Power (0%/ 100% load) (in mW))	483/ 522	400/ 450
4G LTE (Mean/ std. dev (in ms))	403.28/ 1095.89	-
4G LTE (Power (0%/ 100% load) (in mW))	503/ 546	-

We further plot the 95th percentile response time plots to compare the two architectures to see the effect of increasing the load on the data retrieval from the database on the user experience. Figures 3 and 4 show the plots for the same with LTE and Wi-Fi modes in the cloud-based architecture, and only Wi-Fi mode in the cloudlet-based architecture.

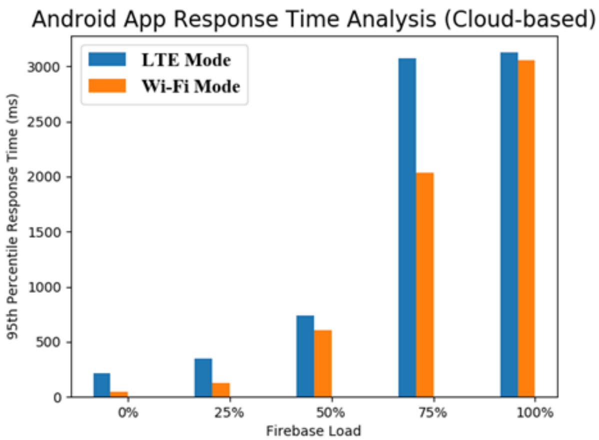

Figure. 3. Android LTE vs Wi-Fi 95th percentile Response time plot for Cloud-based architecture

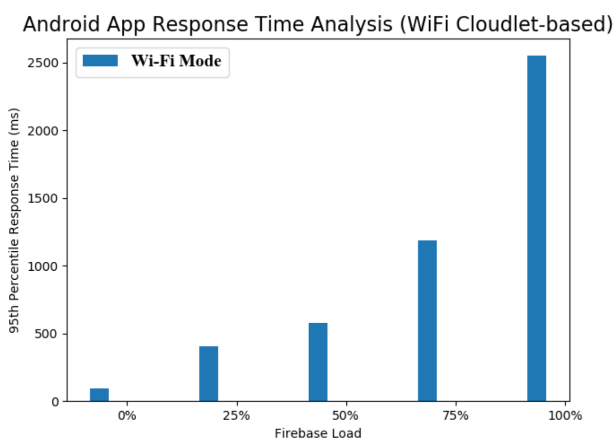

Figure. 4. Android Wi-Fi 95th percentile Response time plot for Wi-Fi Cloudlet-based architecture

While irrespective of the architecture, the response time increases with the increase in the database load. However, Wi-Fi cloudlet-based architecture has lower response times at nearly every load intensity as compared to cloud-based version. In general, offloading to nearby local cloudlet proved to be better for the app, which will likely to make a difference in user experience when the loads are higher. As for the original cloud-based version with Firebase database, LTE measurements are higher than the corresponding measurements over Wi-Fi. This could be because in general, Wi-Fi is more efficient if the signal strength is good while retrieving huge data chunks.

Conclusion and Future work

The current work surveyed existing MCC architectures from both the physical and application layers perspective. Based on our access to the infrastructure, we realized a prototype mHealth app with two MCC architectures. We then compared the performance of this app with cloud-based and Wi-Fi cloudlet-based MCC architectures. Cloudlet-based architecture over Wi-Fi performs better and provides better user experience when performing resource-intensive data retrievals which is crucial for delay-sensitive medical reference-based healthcare apps. To the best of our knowledge, in order to assess the cloudlet-based performance over LTE network, we need access to LTE-connected cloudlets that are not yet commercially deployed and needs a prototype setup with assistance from telecommunication companies to get an experimental FCC license. A Cloudlet server then needs to be connected to this new network's eNodeB base station. As a future work, we would like to explore this methodology and setup an experimental low power lab prototype LTE network and perform further assessment of the prototype app's cloudlet-based model over LTE. We would also like to consider the mobility model and the ability for the app to perform nearby cloudlet discovery and selection as opposed to a dedicated server as a future extension.

References

[1] E. –M. Fong, and W.-Y. Chung, "Mobile cloud-computing-based health-care service by noncontact ECG monitoring," *Sensors,* vol. 13, no. 12, pp. 16451-16473, 2013.

[2] P. Chalela, E. Munoz, D. Inupakutika, S. Kaghyan, D. Akopian, V. Kaklamani, K. Lathrop, and A. Ramirez, "Improving Adherence to Endocrine Hormonal Therapy among Breast Cancer Patients: Study Protocol for a Randomized Controlled Trial," *Contemporary Clinical Trials Communications,* vol. 12, pp. 109-115, 2018.

[3] J. B. Abdo, and J. Demerjian, "Evaluation of mobile cloud architectures," *Pervasive and Mobile Computing,* vol. 39, pp. 284-303, 2017. doi: http://dx.doi.org/10.1016/j.pmcj.2016.12.003.

[4] A. M. Whaiduzzaman, "Performance enhancement framework for cloudlet in mobile cloud computing/ Md Whaiduzzaman," *Ph.D. diss.,* Univ. Malaya, Kuala Lumpur, Malaysia, Tech. Rep., 2016.

[5] N. Fernando, S. W. Loke, and W. Rahayu. "Mobile cloud computing: A survey," *Future Generation Computer Systems,* vol. 29, no. 1, pp. 84-106, 2013.

[6] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. "The case for vm-based cloudlets in mobile computing," *Pervasive Computing, IEEE,* vol. 8, no. 4, pp. 14-23, Oct 2009.

[7] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *Proc. 3rd ACM Workshop Mobile Cloud Comput. Services,* pp. 29-36, 2012.

[8] L. Tawalbeh, Y. Jaraweh, and F. Dosari, "Large scale cloudlets deployment for efficient mobile cloud computing," *J. Netw.,* vol. 10, no. 1, pp. 70-76, 2015.

[9] D. Fesehaye, Y. Gao, K. Nahrstedt, and G. Wang, "Impact of cloudlets on interactive mobile cloud applications," in *Proc. IEEE 16th Int. Enterprise Distrib. Object. Comput. Conf. (EDOC),* pp. 123-132, Sep. 2012.

[10] Y. Jawareh, H. Ababneh, M. Alhammouri, and L. Tawalbeh, "Energy efficient multi-level network resources management in cloud computing data centers," *J. Netw.,* vol. 10, no. 5, pp. 273-280, 2015.

[11] X. Ma, Y. Zhao, L. Zhang, H. Wang, and L. Peng, "When mobile terminals meet the cloud: computation offloading as the bridge," *Network, IEEE*, vol. 27, no. 5, pp. 28–33, September 2013.

[12] M. Satyanarayanan, G. Lewis, E. Morris, S. Simanta, J. Boleng, and K. Ha, "The role of cloudlets in hostile environments," *Pervasive Computing*, IEEE, vol. 12, no. 4, pp. 40–49, Oct 2013.

[13] C. Sonmez, A. Ozgovde, and C. Ersoy, "Performance Evaluation of Single-Tier and Two-Tier Cloudlet Assisted Applications," in *Proc. Workshop on Flexible Networks at IEEE Int. Conf. on Comm.,* July 2017. doi: 10.1109/ICCW.2017.7962674.

[14] H. Routaib, E. Badidi, M. Elmachkour, E. Sabir, and M. ElKoutbi, "Modeling and evaluating a cloudlet-based architecture for mobile cloud computing," in *Proc. 9th International Conference on Intelligent Systems: Theories and Applications (SITA-14)*, pp. 1–7, May 2014.

[15] L. A. Tawalbeh, W. Bakheder, and H. Song, "A mobile cloud computing model using the cloudlet scheme for big data applications," in *IEEE First International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE),* pp. 73–77, June 2016.

[16] Y. Liu, M. J. Lee, and Y. Zheng, "Adaptive multi-resource allocation for cloudlet-based mobile cloud computing system," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2398–2410, Oct 2016.

[17] B-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in *Proc. European conference on Computer Systems, EuroSys 2011,* pp. 301-314, April 2011. doi: 10.1145/1966445.1966473.

[18] OpenMobster. "OpenMobster," (2019). [Online]. Available: https://code.google.com/archive/p/openmobster/ [Accessed Nov. 25, 2019].

[19] P. Chalela, E. Munoz, D. Inupakutika, S. Kaghyan, D. Akopian, V. Kaklamani, K. Lathrop, and A. Ramirez, "Improving Adherence to Endocrine Hormonal Therapy among Breast Cancer Patients: Study Protocol for a Randomized Controlled Trial," *Contemporary Clinical Trials Communications,* vol. 12, pp. 109-115, 2018.

[20] G. Rodriguez, D. Inupakutika, S. Kaghyan, D. Akopian, P. Lama, P. Chalela, and A. G. Ramirez, "Performance Assessment of mHealth Apps," in *Accepted to Proc. IEEE PerCom (PerHealth'20),* March 2020.

[21] PowerTutor, "PowerTutor," (2019). [Online]. Available: http://ziyang.eecs.umich.edu/projects/powertutor/ [Accessed Dec. 22, 2019].

Author Biography

Devasena Inupakutika is a Ph.D. student in the Department of Electrical Engineering at the University of Texas at San Antonio. She is pursuing her studies in the development and analysis of systems and methods for enhancing mobility. Her research interests include web and mobile application development, cloud-IoT integration and deep learning based wireless LAN indoor positioning systems. For correspondence. Email: Devasena.inupakutika@my.utsa.edu.

David Akopian is a Professor and Associate Dean of Research at the University of Texas at San Antonio (UTSA). Prior to joining UTSA, he was a specialist with Nokia from 1999 to 2003. From 1993 to 1999, he was a staff member at the Tampere University of Technology, Finland, where he received his Ph.D. degree in 1997. His current research interests include signal processing algorithms for communication and navigation receivers, and implementation platforms for software defined radio, and mHealth.

Patricia Chalela is an associate professor at the Institute for Health Promotion Research at UT Health San Antonio and the associate director for Education and Training programs. She received her BS in community health in 1981 from Universidad Industrial de Santander, MPH in Health Promotion/ Health Communications in 1995 from UT Health San Antonio Houston, TX and DPH in Health Promotion in 2006 from UT Health Science Center at Houston, TX. Her research interests are in chronic disease prevention and control, particularly the role of epidemiological, environmental and individual psychosocial factors on health and disease, and on racial/ethnic disparities with emphasis on Latino populations.

Amelie G. Ramirez is a chair of the Department of Poplulation Health Sciences and a director and professor of the Salud America at the Institute for Health Promotion Research at UT Health San Antonio. She received her BS in Psychology in 1973 from the University of Houston, TX, MPH in Health Services Administration in 1977 and DPH in Health Promotion in 1992 from the UT Health Science Center Houston, TX. She is an internationally recognized cancer health disparities researcher, has spent 30 years directing research on human and organizational communication to reduce chronic disease and cancer health disparities affecting Latinos, including cancer risk factors, clinical trial recruitment, tobacco prevention, obesity prevention, healthy lifestyles, and more. She also trains/mentors Latinos in behavioral sciences and is on the board of directors for LIVESTRONG, Susan G. Komen for the Cure, and others. She is a member of the Institute of Medicine (IOM) of the National Academies.