2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)

Real-Time Communication Network using Firebase Cloud IoT Platform for ECMO Simulation

*Abdullah Alsalemi, Yahya Alhomsi, Mohammed Al Disi, Ibrahim Ahmed, Faycal Bensaali, and Abbes Amira
College of Engineering
Qatar University
Doha, Qatar
Email: *a@fkrtech.com

Guillaume Alinier
Director of Research
Hamad Medical Corporation Ambulance Service
Doha, Qatar

***Abstract*—Extracorporeal membrane oxygenation (ECMO) is a high-complexity life-saving procedure where quick and coordinated interventions are required to avoid mortality. ECMO perfusionists and nurses must be well trained to respond to time-critical emergencies, using simulation-based education, which currently rely on rudimental ECMO simulation processes. Implementing a simulation system for such complex medical procedure necessitates a robust, well-coordinated underlying architecture. In this paper, the use of Internet of Things (IoT) service Firebase is presented in the case study of an ECMO simulator. Features of the proposed solution are discussed and justified in the given use case including latency, hardware compatibility, and data management. For the used case of ECMO simulation system, Firebase proved to be a proper communication solution that satisfied the urgent nature of medical training.**

Keywords—ECMO simulation; real-time communications; internet-of-things; and clinical training.

I. INTRODUCTION

Extracorporeal Membrane Oxygenation (ECMO) is a lifesaving procedure developed for the care of patients with short-term respiratory, and/or circulatory issues that is proven to improve survival rates up to 75% [1]. Patients connected to ECMO must be monitored twenty-four hours a day by an ECMO-trained multidisciplinary team due to numerous potential patient-side health complications caused by this method. The trained healthcare professional needs to watch over fifty variables and rapidly intervene to assess and resolve any given emergency [1]. Hands-on training is a necessity to help trainees understand the interactions that occur between the patient and the system. Emergency simulation sessions are employed to develop quick interventional skills, coordination, and teamwork that are required to identify and treat life-threatening complications. At Hamad Medical Corporation (HMC) in Qatar, the instructors are looking for developing a simulation system that can simulate ECMO blood-circuit parameters, patient-related parameters, and related emergencies. The aim of this initiative is to design and implement an ECMO simulation system for the use of patient management training at HMC while considering different design constraints laid down by the local clinical experts.

The design of the simulation system consists of three units placed physically distant from each other: ECMO unit, patient unit, and heater unit. To achieve realistic operation, each unit must execute operations in synchrony with other units. Basically, the three units should be continuously synchronized through a real-time and stable communication system.

In consideration of physical distance between the three units and the unpracticality of establishing a wired network, HMC has set a constraint to employ a proper wireless technology that ensures uninterrupted connection and low-latency (100-500 msec). Furthermore, the instructor must be able to monitor and control all system parameters in real-time via a smartphone application to simulate different training scenarios. Upon these constrains and requirements, a low-latency wireless network must be implemented in the simulation system to achieve realistic conditions during ECMO simulation sessions. In this paper, the use of Firebase cloud-based Internet of Things (IoT) platform as a communication system for connecting all simulation system units and the instructor's smartphone is presented.

The remaining sections of this paper are organized as follows. Section II describes the ECMO simulation system units and the different types of operating systems and hardware of each unit. The implementation of Firebase is discussed and justified in section III. Section IV presents Firebase's core aspects. Section VI concludes the paper.

II. ECMO SIMULATION SYSTEM OVERVIEW

To simulate ECMO processes without recreating the complicated mechanical and electrical operations of an actual ECMO machine, the simulator will consist of three units that operate simultaneously with each other to generate visual and audio effects that mimic the real operation of the ECMO circuit and its interaction with a patient. Each unit is responsible for creating multiple physical cues and visual effects based on various values of system parameters. These parameters must be accessible by all units to be tracked or modified. The instructor smartphone application is the main controller of these parameters and it defines what emergency scenario the simulation system must simulate. The units of the simulation system are the ECMO unit (with its mock oxygenator), the patient unit, and the heater unit. Fig. 1 shows a block diagram of the simulation system [2].

A. Overview of ECMO Unit

The ECMO unit consists of a mock oxygenator and replicated ECMO machine casing (Representing the Maquet CARDIOHELP) containing a single-board computer (Raspberry Pi) driving a touchscreen that display a mock ECMO console. The mock ECMO console displays multiple patient related parameters from which the healthcare professional (operator) can diagnose what kind of emergency is occurring and what actions must be taken. Knobs and buttons are connected to the same single chip computer for the operator to "control" the blood loop (ECMO circuit). Each change, the operator applies to the system parameters using the mock ECMO screen, will result in changing other parameters and generating different types of physical cues. Since the system parameters are accessible by the instructor smartphone application, the instructor can monitor the changes the operator is applying and track whether the right actions is taken.

B. Overview of Patient Unit

The patient unit includes a circulation pump and *modules* that generate the physical cues induced by ECMO emergencies. The ink circulation pump is controlled via a microcontroller (Atmega328). The microcontroller accesses the system parameters and applies the changes on the pump and modules to generate visual effects that the trainee can discern and act upon accordingly.

C. Overview of Heater Unit

The heater unit includes multiple sensors to monitor the flow of simulated blood trough the tube. A microcontroller (Atmega328) reads the sensors' values and informs the instructor about the status of the simulated case. In addition, the heater unit contains two electrical valves to simulate multiple physical cues based on parameters the instructor modifies using the smartphone app.

Overall, the simulation system consists of multiple types of boards and devices that each run one different operating system and firmware. The current development version of the communication system of the simulator is compatible with iOS/Android device, Ubuntu (Raspberry Pi), and ATmega16U2 firmware (Atmega328 microcontroller) to create a real-time network and thus operate the simulation in realistic operational conditions. Note that an iOS device is used for initial prototyping. More platforms can be supported in the future.

III. ECMO SIMULATOR COMMUNCATION SYSTEM OVERVIEW

One of the cornerstone features of the simulator is remote scenario management; whereby instructors can manipulate scenario parameters, trigger physical cues, and switch on/off modules from behind the scene without being visible to the scenario participants. Likewise, intercommunication between system components requires a proper and reliable communication system. This necessitates a wireless communication scheme that satisfies the following criteria:

1) Low-latency: real-time parameter control is a requirement in ECMO training due to the time critical nature of any issue associated with this life-saving procedure. Latency is constrained between 100-500 msec overall.

2) Adequately reliable, moderately uninterrupted connection: data transfer between system modules and instructor's interface should be relatively consistent and robust.

1) Compatible with smart mobile devices: to design a convenient remote scenario management interface, a

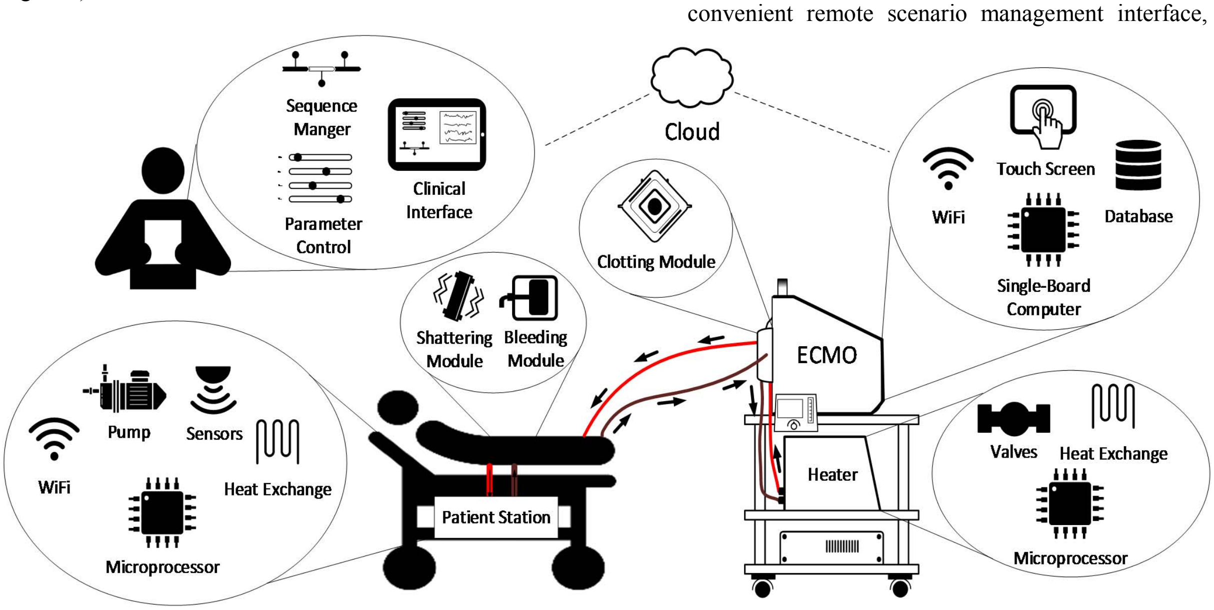

Fig. 1. Block diagram of the ECMO simulator [2].

smartphone application should be developed, and thus the communication scheme must be compatible with the selected mobile device to be used.

Note that some design criteria such as security, high throughput, large user-capacity are not in the priorities of this design; which is due to the very nature of the medical training.

WiFi is chosen as the preferred communication technology. WiFi is an IEEE 802.11 standard for wireless local area networks (WLAN) [3]. The key advantage is its compatibility with smartphones and the presence of state-of-the-art software libraries that accelerate the development of mobile applications. This is mainly because WiFi enables internet connection to cloud-computing services that handle database management; which is much needed in remote scenario management. Thus, a proper cloud-computing service shall be selected. The benefit of using a centralized, cloud service is increased connection reliability due to the sophisticated measures employed by some cloud-computing services to ensure uninterrupted connection [4]. Additionally, some cloud services provide Application Programming Interfaces (API) that facilitate the use of the service. However, latency must be kept below the specified range to allow real-time control of ECMO parameters during training sessions.

A cloud service that satisfies the above considerations is Firebase, a Google-owned multi-service cloud-computing solution for mobile and web developers [5]. The service includes a real-time NoSQL database management system that enables fast queries execution. Each database is consolidated in a single JavaScript Object Notation (JSON) file, where tabular relations are used rather than traditional relational databases. This allowed highly-responsive data synchronization across system modules with latency in the range of milliseconds. Note that the use of Firebase does not ensure connection reliability, rather, real-time data transmission/retrieval is procured when a stable WiFi connection is available. A basic test was conducted on Firebase database; where a sample parameter is adjusted from one module (an iPhone application) and the change is reflected on another module (a Raspberry Pi running a Python script). Experimental results showed consistent and successful data retrieval with latency around 100-500 msec at normal operation and 500-1000 msec when re-establishing connection after 15-20 min of idle time. Thus, WiFi alongside Firebase real-time database were used as the core of the communication system.

Fig. 2 depicts a block diagram of the proposed communication system. The system can be divided into three parts: the instructor side, the patient side, and the cloud. In the instructor side, a remote scenario management smartphone application communicates parameter changes, modules and scenario instructions to Firebase. Instructions are stored in Firebase real-time database and directly synchronized with the patient side. The patient side includes the ECMO unit; which is a single-board computer running a Python application that synchronizes database changes into the system locally and translates them into meaningful instructions. It also runs an application that mimics the console's GUI of CARDIOHELP. The ECMO unit communicates with physically-distant simulation modules via Firebase; sending instructions and receiving status updates from the module. For example, an instructor can initiate a line-shattering cue from the smartphone app from behind the scene of the training session; increasing simulation fidelity. The main unit chosen is the Raspberry Pi 3 Model B which includes WiFi support. An ESP8266 WiFi module will be installed on each module microcontroller. Fig. 3 shows the instructor app alongside the emulated GUI.

IV. Communication System Core Aspects

A. Latency

One of the challenges of employing a cloud-based solution is the risk of high latency–compared to using short-range

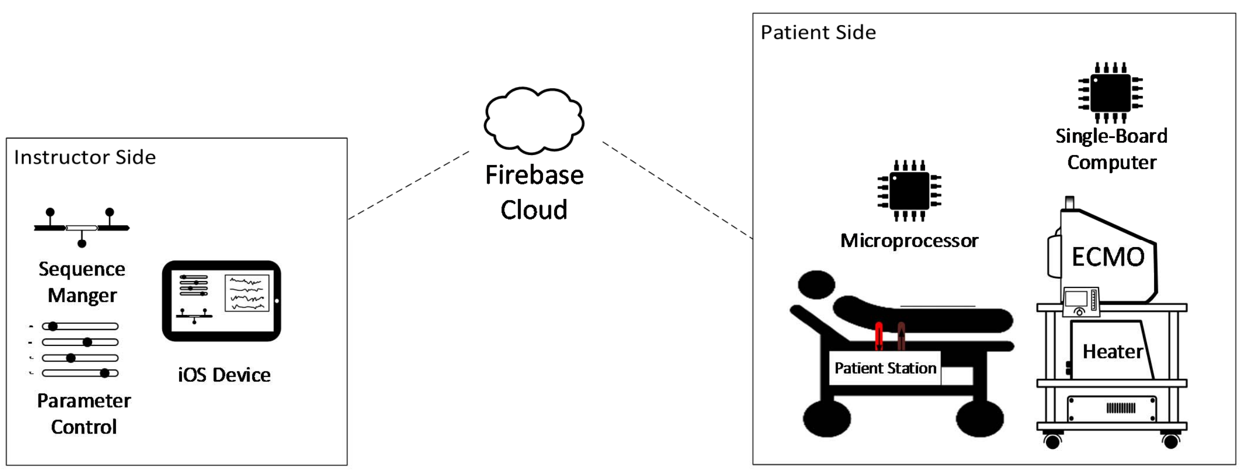

Fig. 2. Overview of the simulator's communication system.

technologies such as Bluetooth. The selection process of a proper cloud service and its underlying storage mechanism is a crucial factor for diminishing communication latency. Another limiting factor is the nature of the transferred data. Large files such as photos and videos require a high data rate to ensure low-latency. In the case of this ECMO simulator,

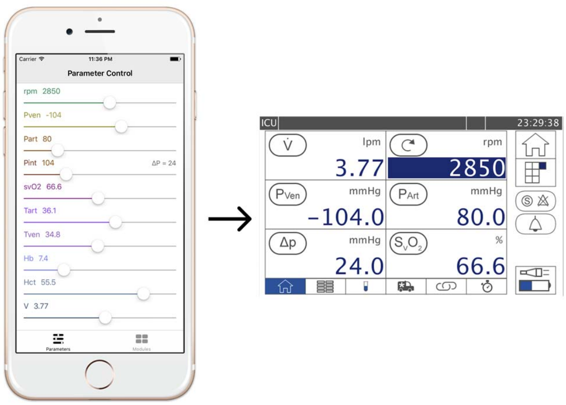

Fig. 3. Instructor app controlling the emulated ECMO console.

transferred data are numerical parameters that control different aspects of the simulated ECMO machine (such as RPM, pressure, and patient bleeding). Therefore, the data rate required is significantly less compared to large file.

Firebase offers a NoSQL real-time database that stores data in JavaScript Object Notation (JSON) files in a key-value mechanism [6]. Compared relational database systems such as SQL, NoSQL does not employ tabular relationships; which enables frequent value change [7]. NoSQL databases behold dynamic schemes; meaning that database structure can be changed without predefined scheme (compared to relational database; where a concrete structure must be put forth before inserting data). This property, among others, allows agile development of the database system that adapts to the system requirements as it evolves during its development cycle. It also enables rapid data insertion/modification and information retrieval. Many NoSQL databases systems employ integrated caching capabilities. This allows frequently-queried data to be stored locally on the client side to speed up data retrieval [8]. Nonetheless, for this ECMO simulator application, integrated caching is not required since the nature of data is light.

From the aforementioned reasons, the Firebase's NoSQL database system is a strong candidate to satisfy the latency condition. After conducting a test where 12 ECMO parameters (e.g. pressure values, flow rates, pump speed, hemoglobin levels, oxygen saturation, and temperature) are send/received from two entities (an iOS application and a Python script running a Raspberry Pi 3), experimental results showed consistent and successful data transmission/retrieval with latency of around 100-500 msec at normal operation and 500-1000 msec when re-establishing connection after 15-20 min of idle time.

B. Compatibility

Firebase Database, which is owned by Google, supports a number of platforms and devices, such as iOS, Android, Linux (via Python), and Arduino firmware (via REST API) [6]. This compatibility is especially useful in time-limited development cycles. First, it encapsulates development efforts into one centralized service instead of using different scattered services tied together. This reduces the development time and lets the engineer focus on other parts of the system being developed. Second, it allows the communication between different hardware systems with less effort and a shorter learning curve due to the availability of Software Development Kits (SDK) on different platforms. The core of this argument stands out when compared to other communication technologies such as ZigBee; which are not widely and readily supported across different platforms (such as iOS and Android) [9]. Thus, Firebase with WiFi is a relatively more compatible wireless communication approach.

In the showcased ECMO simulator, three different hardware systems were intertwined using Firebase. An iOS application transmits ECMO parameters and simulation commands to a central Raspberry Pi unit running a Python script to retrieve updated parameters in addition to two ESP8266 microcontrollers receiving commands and executing different simulation of physical cues accordingly.

C. Data Management

In systems that contain different control units, it is crucial to organize and manage the flow of data between each unit to synchronize their operations. By utilizing Firebase, each unit can directly access the database and store/read any parameter once needed. Basically, once a unit manipulates any parameter within the database, there is no need for the unit to inform all the other units with the changes that have been made. Each unit will read the changes that happened in the database and start taking actions based on these changes. Such technique can increase the efficiency of the communication system and optimize it for faster data exchange since all units can access the database at the same time. Furthermore, the established star network between the units and Firebase (where firebase is the hub) will allow the communication system to maintain it's operation even if any unit in the network is not functioning or having a poor connection. Since the communication system uses star topology, more modules and units can be easily added and integrated with the current system.

Moreover, Firebase can store all parameters and data for further simulation scenarios without the need for each unit to store all the parameters at the end of training session. This can save a lot of time when setting up simulation sessions and provides a convenient solution for retrieving previous simulation scenarios and data that are important for establishing real-life conditions.

V. Conclusions

This work described the use of Firebase in a specialized medical and healthcare education context: ECMO simulation. ECMO training is a multi-disciplinary process that demands speed, flexibility, and cooperation. Based on these criteria, a simulator is designed accordingly with a proper communication system. Low-latency, robustness, remote control, and hardware compatibility were the selection constraints of the wireless technology. Firebase (along with WiFi) was chosen as the method of communication. For this application, it proved to be reliable, compatible with the hardware used, and fast as per the aforementioned constraints and tests.

ACKNOWLEDGMENTS

This publication was made possible by an internal Qatar University grant and UREP grant #19-062-2-026 from the Qatar National Research Fund (a member of Qatar foundation). The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] L. D. Pihera, T. R. Ender, and M. L. Paden, “Extracorporeal Membrane Oxygenation (ECMO) - A systems of systems engineering characterization,” in *2013 8th International Conference on System of Systems Engineering*, 2013, pp. 243–248.

[2] M. Aldisi *et al.*, “Design and implementation of a modular ECMO simulator,” *Qatar Med. J.*, vol. 2017, p. 62, Feb. 2017.

[3] C. Cordeiro, D. Akhmetov, and M. Park, “IEEE 802.11Ad: Introduction and Performance Evaluation of the First Multi-gbps Wifi Technology,” in *Proceedings of the 2010 ACM International Workshop on mmWave Communications: From Circuits to Networks*, New York, NY, USA, 2010, pp. 3–8.

[4] “Cloud storage shoot-out 2013 – performance and reliability,” *Pingdom Royal*, 09-Jul-2013. [Online]. Available: http://royal.pingdom.com/2013/07/09/cloud-storage-2013/. [Accessed: 22-Dec-2016].

[5] “Firebase | App success made simple,” *Firebase*. [Online]. Available: https://firebase.google.com/. [Accessed: 22-Dec-2016].

[6] “Firebase Realtime Database,” *Firebase*. [Online]. Available: https://firebase.google.com/docs/database/. [Accessed: 21-Feb-2017].

[7] N. Leavitt, “Will NoSQL Databases Live Up to Their Promise?,” *Computer*, vol. 43, no. 2, pp. 12–14, Feb. 2010.

[8] “NoSQL Databases Explained,” *MongoDB*. [Online]. Available: https://www.mongodb.com/nosql-explained. [Accessed: 21-Feb-2017].

[9] C. Withanage, R. Ashok, C. Yuen, and K. Otto, “A comparison of the popular home automation technologies,” in *2014 IEEE Innovative Smart Grid Technologies - Asia (ISGT ASIA)*, 2014, pp. 600–605.