iium ej 1596

by 1596 lium Ej

Submission date: 25-Aug-2020 09:12PM (UTC+0800)
Submission ID: 1373867725
File name: 1596-Article_Text-9611-1-2-20200825_no_names_no_ref.docx (96.12K)
Word count: 3933
Character count: 18372

MONITORING AND CONTROL SYSTEM WITH A CLIENT-SERVER MODEL BASED ON INTERNET OF THINGS (IOT)

ABSTRACT: Safety and comfort are the needs that all humans want to get. Meanwhile, the crime rate is increasing. Therefore, we need a remote monitoring and control system. This research offers a home monitoring and control system with a client-server model using NodeMCU ESP-12E. The equipment designed can be used to monitor the condition of the house through sensors installed in each room. Home monitoring includes motion detection using a motion sensor, detection of the condition of the house door using a magnetic sensor, and remote door locking using a solenoid. The system can be used offline or online using an Android smart phone. The communication model used on client-server using the transport-layer protocol is User Datagram Protocol (UDP), so the server can communicate simultaneously on two clients with the fastest average response time is 0.653 seconds. Communication model between a server and a cloud uses the Transmission Control Protocol (TCP) so that the data sent or received by the server through the internet more reliability. The cloud used is Firebase which has real-time database facilities and historical data. On the State of the online, sensor response time average is the fastest on an android is 3.898 seconds, response time control the fastest average on a client is 7.157 seconds and the control response time average is the fastest on an android is 9.495 seconds.

KEY WORDS: *IoT, client-server, control, monitoring, android, and NodeMCU ESP-12E*

1. INTRODUCTION

Home is where we live to meet daily needs. Along with the development of the current times, many are using the electronic system inside the House or building. Home security technologies in use today's web-based, so a home security monitoring is only when the owner opened the web address. Homeowners do not know the condition of the House directly, so that when there is a danger, houses can't be prevented timely or unwanted things happen.

One of the solutions to find out the conditions of security and home control each time then needed a tool that can monitor the application via Smartphone with an internet connection using the concept of the Internet of Things (IoT), so that users can know the State of the home each time with great distances via the internet network. This concept is where objects with other objects can communicate or any embedded with sensors connected through a network of the internet [1]. One of the services used in the concept of the IoT is Firebase. Firebase is a Cloud Service Provider (CSP) and a Backend as a Service (BaaS) owned by Google which allows users to store data and read data in real time [2, 3]. So we don't need to build features from the beginning so we can focus to develop IoT-based applications easily without the need to create a cloud of his own.

Previous research has been carried out by Nama who designed a surveillance system over the internet using a BCM 2835 microcontroller and sent notifications via a short message service [4], and Despa who designed smart monitoring of electrical quantities based on single board computer BCM2835 [5]. Other research was conducted by Nurfaif who designed a smart home using Raspberry Pi through the Global System for Mobile Communication (GSM) network [6]. Other research has also been carried out by Kodali whose home security system

uses the CC3200 IT microcontroller. The system works on a single microcontroller with sensors via cables and sends notifications via telephone calls [7]. Subsequent research was carried out by Malche who designed a monitoring system via the MQTT-based internet web [8].

On this research designed to make home security maintained by monitoring the condition of the House, turn on the blower when there were LPG gas, electronic door lock control and home owners can control the electric light in each room through the application Android with the network the internet (online) or local network (offline). The following is a related journal references in this study are as follows:

2. THE PROPOSED SYSTEM

This research uses firebase to facilitate the communication of mobile and Web applications. Firebase is a Cloud Service Provider offered by Google to facilitate the development of mobile and web applications. Firebase has real-time database facilities and historical data. The block diagram of the monitoring and control equipment designed through this research is shown in Fig. 1.

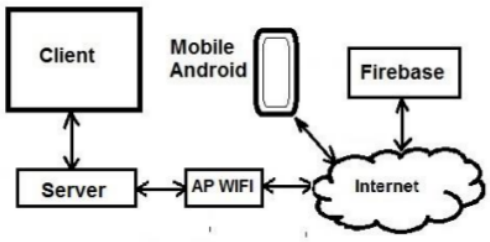

Fig.1 Block diagram of system design

The client consists of some sensors, each of which is connected to NodeMCU ESP-12E. Client 1 is the MC-38 magnetic sensor and HC-SR505 motion sensor, the client 2 is a HC-SR505 motion sensor, and the client 3 is a HC-SR501 motion sensor and MC-06 gas sensor. Each client is connected to NodeMCU ESP-12E which then connects to a server in the form of NodeMCU SP-12E as well.

2.1 Internet of Thing

Internet of Things is a concept where certain objects have the ability to transfer data through the Internet network. This process does not require interaction from humans to humans or humans to a computer or run automatically with the program. The constituent elements of IoT are artificial intelligence, sensors, connectivity, and various devices that are small in size.

Control and monitoring from very long distances can be done using IoT through the internet network. Sensors on the client side feel the phenomena that occur around it and send data to the microcontroller to be processed into information. Information from the microcontroller is then sent through the internet to the user's smart phone. Smartphone users can provide actions in the form of controlling from remote.

2.2. Firebase

Firebase is a Cloud Service Provider (CSP) and a Backend as a Service (BaaS) which is owned by Google. Firebase is the solution offered by Google to simplify mobile application development as well as in the web [2]. We don't need to build features from the beginning so

we can focus to develop IOT-based applications easily without the need to create a cloud of his own. Firebase has many SDKS that allow to integrate it this service with Android, iOS, Javascript, C++ to Unity.

2.3. Client-Server

Client-server is a work arrangement in accessing a server on a particular network between client and server. Data communication on the client-server arrange of interfaces that function as a place to run database applications. A client-server network is a computer network architecture where the client device will process the request for data, and the server has the task of responding in the form of data to the request [7]. The communication model of client and server network systems is shown in Fig. 2.

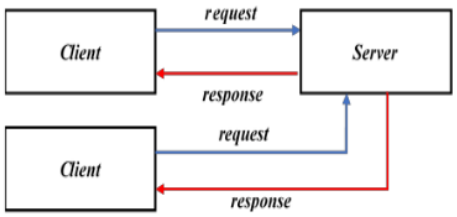

Fig. 2 Client-server network block diagram

The client 1 used for controlling solenoid door locks so we can unlock and lock the doors, monitor motion and monitor the condition of the door that there is on the front porch of the House. Client 1 can be installed at the front door of the house. The goal is to be able to open and close the door remotely and notify the homeowner if there is someone in front of the house. The client 2 is used to control the lights in the living room and monitor the movements in the house. Client 2 can be installed in the living room and used to monitor if there are people in the house.

3. EXPERIMENTAL RESULT

In this research, the time response of the equipment in responding to sending commands is compared between when using a local network and the internet network. A local network is a communication network using a Wi-Fi connection so that gives commands to equipment the maximum distance is only 10 meters. Communication or giving commands to equipment using the internet network can be done from anywhere with unlimited distance. **Fig. 3** shows a block diagram of a local network where communication is done between an Android mobile and equipment via a Wi-Fi AP.

3.1 Client 1

Data response time on client 1 on the local network is shown in **Table 1**. The response time is the delay of the action that occurs on the sensor when given command through the local network. Data is taken from five attempts with the initial condition of the door being closed and the initial condition of the door is open.

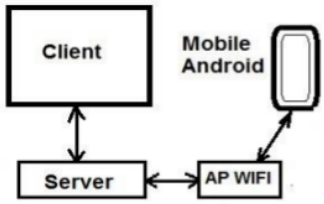

Fig. 3 Local Network Block Diagram

Table 1. Response time client 1 and android on the door with a magnetic sensor (mc-38) when offline

No	Response Time of Android (seconds)	The Initial State of The Door	The Final State of The Door
1	2.24	Closed	Open
2	3.55	Closed	Open
3	2.88	Closed	Open
4	2.65	Closed	Open
5	1.65	Closed	Open
6	1.78	Open	Closed
7	2.91	Open	Closed
8	3.28	Open	Closed
9	2.69	Open	Closed
10	1.58	Open	Closed

Table 1 shows when magnetic sensors (MC-38) given the State of the door with certain conditions on the local network or offline. Response time where the client sends data or sensor feedback results to the server and the server data or sensor results broadcast feedback to android Wi-Fi AP via a network. The value of the response time average Android is 2.521 seconds.

On the server the data processed in the form of JSON data type and then the sensor or result data broadcast feedback to AP WiFi. Then android listening data from the server via WiFi AP network and receives data the sensor results.

On android applications listening data sensor results each client from the server via WiFi AP. the client had arranged time span to send data or sensor feedback results to the server with a range of 1000ms and span of time servers for broadcast data or sensor feedback results to AP WiFi i.e. 1000ms. So the total time span minimal required from the client to the application android through the local network or offline i.e. 1000ms.

The existence of a delay in response time, because when the client sends to the server, the server is not processing data at the time of the time such as still processing data from the cloud and a span of 300ms for listening client. When a server sends a broadcast data to android, android receives data every 1000ms and process data from the cloud conditions firebase.

Table 2 shows that data response time android on client 1, when magnetic sensors (MC-38) given the State of the door with a particular condition on the internet network or online. The response time of the android is where the client sends data or sensor feedback results to the server and the server is writing data to the sensor or the results of the feedback to the cloud through a network of Wi-Fi AP Firebase. The value of the response time average android is 4.373 seconds.

Table 2. Response time client 1 and android on the door with a magnetic sensor (mc-38) when online

No	Response Time of Android (seconds)	The Initial State of The Door	The Final State of The Door
1	4.04	Closed	Open
2	5.77	Closed	Open
3	2.84	Closed	Open
4	4.54	Closed	Open
5	3.61	Closed	Open
6	5.25	Open	Closed
7	5.99	Open	Closed
8	4.76	Open	Closed
9	2.47	Open	Closed
10	4.46	Open	Closed

On the server the data results of the sensors or feedback received from the client is processed in the form of JSON data type and, if the data changes, then the server write data the results of sensor or feedback to the Firebase network through cloud Wi-Fi AP.

Android app reading sensor data each client from the cloud Firebase. In the client has arranged time span to send data or sensor feedback results to the server with a 1000ms range and the span of time the server to write data or sensor feedback results to the cloud through a network of WiFi AP Firebase with 4000ms range. So the total time span minimal required from the client to the internet network through the android application or online i.e. 4000ms.

The existence of a delay in response time, because when the client sends data to the server at the time of that time, the server is still processing the data from the cloud and 300ms for listening span client and when the server sends a broadcast data to android, android receives data every 1000ms and process data from the cloud conditions firebase.

Table 3. Response Time Client 1 and android on the sensor motion (HC-SR505) when offline

No	Response Time of Android (seconds)
1	3.8
2	1.85
3	3.02
4	3.09
5	2.64
6	1.85
7	1.29
8	1.78
9	2.03
10	1.65

The data response time android on client 1 for motion sensors (HC-SR505) was given to the condition of the moving objects on the local network or offline shown by Table 3 and on the internet network shown by Table 4. The response time of the android is where the client sends data or sensor feedback results to the server and the server data or sensor results broadcast feedback to android WiFi AP via a network. In this research, data retrieval is carried out 10 times when the motion sensor senses the object moving. The value of the response time average android that is 2.3 seconds for a local network and 4.753 second.

Table 4. Response Time Client 1 and android on the sensor motion (HC-SR505) when online

No	Response Time of Android (seconds)
1	4.45
2	5.75
3	3.54
4	5.08
5	5
6	4.32
7	3.54
8	5.62
9	4.52
10	5.71

Android app reading sensor data each client from the cloud Firebase. The client has arranged a time span to send data or sensor feedback results to the server with a 1000ms range and the span of time the server to write data or sensor feedback results to the cloud through a network of Wi-Fi AP Firebase with 4,000ms range. So the total time spans minimal required from the client to the internet network through the android application or online is 4,000ms.

The existence of a delay in response time because when the client sends data the results of a sensor to the server, the server is still processing the data from the cloud and the server has a server listening span i.e. 300ms. When the server sends data from the cloud conditions firebase and android are reading data from a cloud firebase every 1,000ms.

Table 5 shows data for the response time of a client and android on client 1, when the solenoid door lock in execution with Open and Closed conditions on the local network or offline. Response time on client response time is faster than android because the response time on the client is the response which the time span between the android data gives instructions to the server through a Network Access Point (AP) Wi-Fi broadcast and server data instructions to the client via the Wi-Fi network server. While the response time on android is the response time of a client after the process and the process of listening time android. Android listening time process is a time span of listening data from the server via the Wi-Fi AP network. The value of the response time average client is 0.653 seconds and the value of the response time averages android is 4.108 seconds.

On the server, data, instructions from android treated in the form of JSON data type and then the instruction data to the broadcast client, so the client receives the instruction data and executes it. The server had orchestrated a span to send data to the client with an instructions range of 500ms. So at least the range of client response time via local network or offline is 500ms.

On Android, the client sends data measurement results or feedback data to a server with a span that is 1000ms, the server listening to the client is set up with a span that is 300ms, and server processing the data of each client in the form of JSON data then Server broadcast to android apps via the AP with Wi-Fi is 1000ms. So at least the range of response time from android to the server to the client after that client to the server through a local network to the android or offline is 1,500ms.

Table 5. Response Time Client 1 and android on the solenoid door lock when offline

No.	Response Time of Client (seconds)	Response Time of Android (seconds)	The Initial State of The Lamp	The Final State of The Lamp
1	0.85	4.19	Closed	Open
2	0.8	3.84	Closed	Open
3	0.6	4.47	Closed	Open
4	0.61	3.36	Closed	Open
5	0.53	3.8	Closed	Open
6	0.74	6.15	Open	Closed
7	0.6	3.95	Open	Closed
8	0.66	3.95	Open	Closed
9	0.54	3.39	Open	Closed
10	0.6	3.98	Open	Closed

Table 6. Response Time Client 1 and android on the solenoid door lock when online

No.	Response Time of Client (seconds)	Response Time of Android (seconds)	The Initial State of The Lamp	The Final State of The Lamp
1	7.85	14.73	Closed	Open
2	10.19	16.71	Closed	Open
3	6.8	13.79	Closed	Open
4	7.85	14.39	Closed	Open
5	6.87	12.95	Closed	Open
6	5.88	10.28	Open	Closed
7	8.24	15.03	Open	Closed
8	6.94	11.44	Open	Closed
9	8.82	12.1	Open	Closed
10	8.58	13.81	Open	Closed

The existence of a delay in response time because when android data sending instructions to a server, the server is still processing the data from the cloud and 300ms for listening span client and when the server sends a broadcast data to android. Android application to receive data every 1000ms and server processes the data to the cloud conditions firebase. When the client next sends the feedback data to the server, the server is still processing the data from the cloud and the server has a span of 300ms for listening client. When a server sends a broadcast data to android, android receives data every 1000ms.

Table 6 shows data of the response time of a client and android on client 1, when the solenoid door lock in execution with conditions Open and Closed on the internet network or online. The response time on the client is faster than the Android response time because the response time on the client is the response where the time span between android writes instruction data to the Firebase cloud, the server reads the Firebase cloud, and the server broadcasts the instruction data to the client via the WiFi server network. While the response time on Android is the response time after the client response time process, the server time process writes instruction data to the Firebase cloud, and the Android time process reads the Firebase cloud. The average response time of the client is 7.802 seconds and the average response time of the android is 13.523 seconds.

3.2 Client 2

The client 2 is a controller for living room lights and monitoring of moving objects in the house. Table 7 and Table 8 shows the data for the response time of a client and android on client 2, when executed with the final State of a particular lamp through the local network or offline and internet network or online.

Table 7. Response Time to Client 2 and Android on the living room light when offline

No.	Response Time of Client (seconds)	Response Time of Android (seconds)	The Initial State of The Lamp	The Final State of The Lamp
1	1.19	3.44	Off	On
2	0.93	2.62	Off	On
3	0.73	2.87	Off	On
4	1.12	4.61	Off	On
5	0.79	3.1	Off	On
6	1	4.83	On	Off
7	1.19	3.26	On	Off
8	0.73	4.71	On	Off
9	1.12	4.99	On	Off
10	1.06	2.76	On	Off

Table 8. Response Time to Client 2 and Android on the living room light when online

No.	Response Time of Client (seconds)	Response Time of Android (seconds)	The Initial State of The Lamp	The Final State of The Lamp
1	8.57	11.58	Off	On
2	7.19	9.38	Off	On
3	7.46	9.56	Off	On
4	6.74	9.12	Off	On
5	6.67	8.77	Off	On
6	5.96	8.51	On	Off
7	8.82	11.07	On	Off
8	8.25	10.33	On	Off
9	5.69	8.16	On	Off
10	6.22	8.47	On	Off

The average client response time for the living room lights is 0.986 seconds and the average value of the android response time is 3.719 seconds on the local or offline network. While the average client response time for a living room lamp is 7.157 seconds and the average value of android response time is 9.495 seconds on the internet or online.

Table 9 shows the android data response time on client 2, when the motion sensor (HC-SR505) found the condition of a moving object when used on a local network and the internet. Response time of the android on the local network is the time response when the client sends data or sensor feedback results to the server and the server data or sensor results broadcast feedback to Android WiFi AP via a network. The value of the Android response time on the local network is 1.411 seconds.

Android response time on internet network is response time where the client sends data or sensor feedback results to the server and the server is writing data to the sensor or the results of the feedback to the Firebase network through cloud WiFi AP. On the server the data results of the sensors or feedback received from the client are processed in the form of JSON data type and, if the data changes, then the server writes data the results of sensor or feedback to the

cloud through a network of WiFi AP Firebase. The value of the response time averages android using the internet network is 3.898 seconds

Table 1. Response Time to Client 2 and Android on the sensor motion (HC-SR505) when offline and online

No	Response Time of Android on local network/offline (seconds)	Response Time of Android on Internet network/online (seconds)
1	1.58	3.74
2	1.19	4.83
3	1.33	3.35
4	1.06	3.68
5	1.45	3.37
6	0.92	3.73
7	1.85	3.89
8	1.57	4.4
9	1.39	3.99
10	1.77	4

4. CONCLUSION

On a system designed, client response time on offline usage is faster than online. The Offline use does not require an internet network only through WiFi AP communication with Android mobile while online requires an internet network to communicate with Firebase which takes longer. The average response time of the magnetic sensor on client 1 for offline usage is only 2.521 seconds while online usage takes 4.373 seconds. The average response time of the motion sensor on client 1 for offline usage is only 2.3 seconds while online usage takes 4.753 seconds. The average response time of the solenoid on the client-side for offline usage is only 0.653 seconds and on Android side is 4.108 seconds while online usage takes 7.802 seconds on the client-side and 13.523 on the Android-side. For lamp control on client 2, the average response time takes 0.986 seconds on the client-side and 3,719 seconds on the Android side on offline use, while online usage takes 7,157 on the client-side and 9,495 on the Android side. The average response time of the motion sensor on client 2 for offline usage is only 1.411 seconds while online usage takes 3.898 seconds.

For further research, more sensors such as gas sensors, visual sensors (cameras), temperature sensors and voltage sensors need to be added to provide more safety for the user.

iium ej 1596

ORIGINALITY REPORT

11%	8%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	digilib.unila.ac.id Internet Source	6%
2	I Made Agus Wirawan, Ida Bagus Made Ludy Paryatna. "Implementation of the String Matching Method on Anggah-Ungguhing Balinese Language Dictionary", International Journal of Interactive Mobile Technologies (iJIM), 2020 Publication	2%
3	manualzz.com Internet Source	1%
4	Submitted to University of Exeter Student Paper	1%
5	Submitted to Victoria University Student Paper	<1%
6	studi.ristekdikti.go.id Internet Source	<1%
7	Sri Ratna Sulistiyanti, FX Arinto Setyawan. "Smart Home Controlling and Monitoring	<1%

System using Multiboard Client-Server Internet of Things (IoT)", Journal of Engineering and Scientific Research, 2020
Publication

8	Muhamed Alarbi, Hanan Lutfiyya. "Sensing as a Service Middleware Architecture", 2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud), 2018 Publication	<1%
9	Submitted to HELP UNIVERSITY Student Paper	<1%
10	Menzi Skhosana, Absalom E ezugwu. "Irenbus: A Real-Time Public Transport Management System", 2020 Conference on Information Communications Technology and Society (ICTAS), 2020 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		