Development of Scalable IoT-Based Smart Home Infrastructure Using ESP-Mesh

Syifaul Fuada[1] and Hendriyana[2]
[1] Program Studi Sistem Telekomunikasi, Universitas Pendidikan Indonesia, Bandung, Indonesia
[2] Program Studi Rekayasa Perangkat Lunak, Universitas Pendidikan Indonesia, Bandung, Indonesia
Email: syifaulfuada@upi.edu

Abstract**—**All household appliances in the Industrial 4.0 era such as lights, curtains, fans, and others are being connected to the internet to allow users to control them through a smartphone. This has led to several intense and continuous studies on IoT-based smart homes to determine the ideal platform and system to provide excellent performance in terms of functionality and electronics such as power consumption, data throughput, distance, interoperability, flexibility, reconfigurable protocols, costs, and others. Therefore, this study was conducted to develop an IoT-based smart home system's infrastructure using the ESP8266 module (Wemos D1 mini) which is a microcontroller integrated into a Wi-Fi module. Moreover, ESP-mesh was used for communication to make the smart home system more scalable and modular. This is necessary to ensure the home remains connected when one node is disconnected due to the fact that the node within the mesh topology is fully connected and has the ability to forward data. The infrastructure contains hardware (node), software, and user interface. The nodes were further divided into three parts which are mechanical-based such as electric door lock, electrical-based such as fan, power socket, or plug, and sensors such as humidity & temperature sensor. Furthermore, the system was evaluated for functionality, power consumption during idle/active processes, RSSI of each node, and mesh connection. The performances were later compared with the findings of previous study on IoT-based Smart Home.

Index Terms—Internet-of-Things (IoT), ESP-MESH, ESP8266, smart home

I. Introduction

Internet-of-Things (IoT) is a system used in connecting all devices commonly used in daily life to the internet and reported to have become a new paradigm in innovation [1], [2]. Moreover, Smart Home applications are designed to establish communications between household appliances and the internet to ensure they can be accessed and controlled anywhere and anytime through a Web-server or Android-based smartphone. This technology is observed to be a part of Machine-to-Machine (M2M) [3]. Meanwhile, the consideration of the connectivity aspect and mobile application features showed the use of Wi-Fi networks for smart home applications is the most practical option compared to NFC, Bluetooth, Zigbee, and several others [4]-[6]. It is also important to note that the most recent version of smartphones has Bluetooth and Wi-Fi features provided with an opportunity for future upgrades based on newer standards. The use of Wi-Fi allows Smart Home to interconnect several sensor devices which perform monitoring functions, actuators which are used as physical system movers or modifiers, devices applied as the controllers, and servers which are the central controllers or the intelligence system brain.

An IoT-based smart home is not a new technology anymore but still attracts the attention of several studies. This is indicated by the intensive and continuous study being conducted to determine the ideal platform or system for IoT and the strategies to improve its performance [7]-[9]. Moreover, several organizations and countries have also developed IoT-based smart homes with specific standards, thereby, leading to the use of different platforms. This is creating serious problems [10] such as interoperability between several home environment appliances [11], [12]. Another problem is the difference in identifiers, operating system platforms, and accepted programming languages for Smart Home systems [13].

Some essential issues required in designing a Smart Home infrastructure are stated as follows:

- *Functionality*: the device needs to cover the basic functions of a Smart Home such as the control and monitoring functions [14]. There is also the need to fulfill reasonable electronic functions and performance such as latency, data throughput, distance, and network coverage [15].
- *Low-power and easy installation*: the smart home infrastructure needs to be efficient with a straightforward interface and low power consumption [16]. The simplicity of the hardware and software design also has the ability to increase user-friendliness [17].
- *Interoperability*: the smart home infrastructure is required to ensure hardware uniformity and compatibility with alternating-current (AC) voltage in different countries (220 or 110 Volt, 50 Hz or 60 Hz) as an interoperability solution. This is needed to reduce the duration of system development and production [16].
- *Flexibility*: the smart home infrastructure design needs to have a good flexibility value for easy configuration. This means the installation of the device in various

Manuscript received September 26, 2021; revised April 12, 2022.
Corresponding author email: syifaulfuada@upi.edu.
doi:10.12720/jcm.17.5.373-385

buildings should not require other additional trinkets or even changing major electrical installations. This is important towards realizing Zhaojing's hierarchy [18].

- *Security*: the system is also required to have an adequate level of security in order to limit the access rights to the owners [19].

This study already produced a smart home platform named MINDS™ in previous work (http://www.pme.itb.ac.id/products/mind-meshed-and-internet-networked-devices-system/) and (https://www.xirkachipset.com/product/minds). This product was recognized globally based on several publications in reputable international journals which led to national and international awards including:

- The best paper in the 3rd and 4th International Conference on Intelligent Green Building and Smart Grid (IGBSG) 2018 & 2019, respectively.
- The best paper in the International Journal of Online Engineering (IJOE) which is a Scopus-indexed journal.
- Top-108 Indonesia Innovations in 2018 [20].

MINDS™ was also presented in several prestigious exhibitions including *Centrum der Büro-und Informationstechnik* (CeBIT) 2016 in Germany [21] and a product demo session in the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS) 2016 in Jeju, South Korea. These achievements indicate MINDS™ as a new start-up which has become one of Indonesia's best inventions.

The MINDS™ divides the smart home system into two environments and these include the *outdoor* which works based on an internet cloud-based system and *indoor* which is a wireless sensor network (WSN)-based system. These two environments are connected using a Gateway which further distributes internet protocol (IP) addresses of the devices to be connected. It is important to note that the previous platform involved complicated data communication scenarios with the use of a Bluetooth module as the smartphone interface to a Smart Home device, a Zigbee module to create a mesh network, and a Wi-Fi module to ensure access outside the home. Moreover, the Gateway on the MINDS™ was implemented using Raspberry Pi 3 [22], [23] while one of the nodes developed was the Temperature & Humidity sensor which requires users to request data first before obtaining temperature and humidity updates. There is also the need to create a local network, install the router, and set it to a Repeater mode to access the system.

This study was conducted to simplify the Smart Home platform designed in the previous study by replacing the Bluetooth and Zigbee modules with only the Wi-Fi module in the form of ESP8266 which combines a microcontroller with Wi-Fi in a single chip. It has a reasonably high specification and a lower cost than other Wi-Fi modules [24]. Moreover, the system was designed to have five end-devices considered to represent the "Smart Home" system as a whole as indicated in [25], [26] and these include: (1) temperature and humidity monitoring, (2) generic power socket, (3) solenoid lock for electric door lock, (4) fan controllers, and (5) lamps. The ESP8266 module was also configured as a gateway or node to replace the Raspberry Pi as used in [27]. Furthermore, the mesh network was formed using the ESP-mesh feature while the Temperature & Humidity sensor was designed to send data periodically every 5 seconds to ensure the users do not have to request data before monitoring the indoor air quality. The ESP8266 Wemos D1 mini-module was selected due to its smaller size when compared to NodeMCU series [28] in order to ensure a more minimal hardware packaging.

The mesh configuration was also selected for the IoT-based smart homes due to the following reasons: (1) high-flexibility to ensure a new node added to the network find the closest point and form a path autonomously, (2) an alternative route is immediately created when one of the nodes cannot be used (nonactive/off) in the network, (3) ability to expand the communication coverage area, and (4) each node only communicates with the closest node to minimize interference.

The contribution of this study compared to [25-26] are as follows:

1. *Seamless connection*: the system does not require a complicated configuration due to the ability of the network to automatically configure itself. This means the user only needs to connect the nodes to the power grid. Moreover, the Wi-Fi mesh provided by ESP8266 was used for the communication system.
2. *Full-mesh network:* the system sets a smartphone, which was used to control and monitor the nodes, into a mesh network
3. *Supports other networks*: it is possible to connect the system to other communication modules by installing a particular node-set to a repeater mode—for example, UART communication, RS-485, NB-IoT, GSM, and others.
4. *A gateway server is not required*: all the nodes are connected to the mesh network including the smartphones used as controllers but a gateway was included for specific purposes.
5. *Internet or local networks are not needed*: the nodes within the system form a mesh network autonomously. This means it is possible to use smart home devices even when they are not connected to the internet network due to the ESP-mesh's features.

II. Methods

A. Smart Home

Only a few end-devices were selected in this study to represent each hierarchy level defined by Zhaojing et al. [18] and as used in [25], [26] and indicated in Table I. The five nodes selected are also similar to those in [25], [26]. According to [18], the hierarchical division was based on the necessary level by considering the most crucial system with a significant impact on the people living in the house. Moreover, there are three categories of smart homes and these include Level I which are devices with the ability to facilitate people's work and improve their quality of life and classified as a primary need. Level II includes an IoT-based smart home with the ability to fulfill secondary needs while Level III is the one with the ability to satisfy tertiary needs.

Conventional lights, doors, and plugs are always present in most residential homes with the most important ones being door locks and power switches or plugs as indicated by the need for their nodes to have a continuous supply of electricity. For example, the door is conditioned to be active at night or when the user leaves the house while the lights are usually active between 17.00 – 05.00 WIB (GMT+7) or 5 PM – 5 AM. Some electronic devices also need to be active at certain hours for a long duration and are usually connected to the generic power socket node. These devices can be classified as Group I which represents the basic need.

TABLE I: HIERARCHY OF HOUSEHOLD DEVICES IN SMART HOME ENVIRONMENT

No.	Zhaojing, et al. [18]		Our work	
	Household devices	**Hierarchy**	**Household devices**	**Hierarchy**
1	General lighting	Group I	• Generic power socket (power plug) 220V, • Lamp 220V • Electric Door Lock	Group I
2	Control system			
3	Refrigerator			
4	Rice cooker			
5	Electric Pan			
6	Washing machine	Group II	Mini DC fan	Group II
7	Clothes dryer			
8	Microwave			
9	Hoods			
10	Air Conditioner	Group III	Temperature Monitoring	Group III
11	Decorative lights and RGB Lamp			
12	Television			
13	Computer		Humidity Monitoring	
14	Loudspeaker			

The devices in Group II which represent the secondary need include fans and others usually used to provide a more comfortable lifestyle. Their level of need is below those in Group I with some residential homes not having them. Moreover, Group III which represents the tertiary need includes sensors which are considered to be beyond necessity but normally used to improve the quality of life. It is not mandatory to install this node in a residential house.

B. System Limitation

The Wi-Fi mesh which is a feature of the ESP8266 Wemos D1 mini was used to establish communication between the nodes in this smart home system. Moreover, the android application developed was made of mesh and this means the user is also considered a node, therefore, there was no need for a server. The system did not require a Gateway but a Gateway node was provided in the system to have two ESP8266 which include the Gateway and TTL-to-USB. This gateway was used for the initial test which involved sending commands from PCs to mesh-connected Nodes [29] as explained in Section II.C.

The system was designed to be applied in residential houses in Indonesia with an average size of 10 m x 10 m and connection distance up to 100 m is considered reasonable between the nodes. It is important to note that the system developed was not connected to the Internet. Its connection to achieve remote coverage requires the users to install a special router configured as a repeater. Moreover, the humidity and temperature sensor data logs were also not provided in the android application considering the need for a special cloud server for the data to be stored. This study also highlights the smart home infrastructure.

C. System Architecture

Fig. 1 shows an IoT-based smart home system architecture designed using ESP-mesh which is a network protocol built on the Wi-Fi protocol. ESP-mesh allows multiple devices, referred to as nodes, spread over a certain large area both indoors and outdoors to be connected under one WLAN.

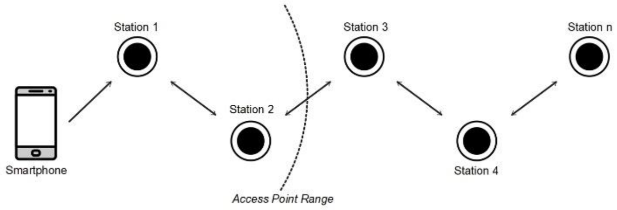


Fig. 1. System architecture for IoT-based smart home with an ESP-mesh

Wi-Fi networks typically use a point-to-multipoint network infrastructure with one master node or access point (AP) directly connected to all other nodes or stations. This AP node is responsible for the mediation and forwarding of transmissions between stations with some relaying transmissions to or from external IP networks via routers.

The Wi-Fi network infrastructure commonly used has the disadvantage of limited coverage area due to the need for each station to be within range to connect directly to the AP. Moreover, the network is prone to overload due to the fact that the maximum number of stations allowed in the network is limited by the capacity of the AP.

The ESP-Mesh differs from traditional infrastructure Wi-Fi networks where nodes are not required to connect to the master node but are allowed to connect with other nodes in order to relay transmissions. The ESP-mesh network has a wider coverage area due to the ability of the nodes to achieve interconnectivity without being within the reach of the master node. This means the nodes allowed on the network in ESP-mesh are no longer restricted by the master node.

The stations in Wi-Fi which are limited to one connection with the AP is known as an upstream connection while the simultaneous connection of AP to several stations is known as the downstream connection. However, the ESP-mesh allows nodes to simultaneously act as stations and APs. This means it is possible for a node in the ESP-mesh to have multiple downstream connections using the softAP interface and also simultaneously have a single upstream connection using its station interface. The condition naturally produced a tree network topology with a parent-child hierarchy consisting of several layers.

D. Software Design

The flowchart node of electric door lock, fan, lamp, generic power socket, temperature and humidity sensor is depicted in Fig. 2(a) to Fig. 2(e) respectively while the flowchart for the gateway is presented in Fig. 2(f). The flow diagram of the four end-devices has the same working process with each other due to their use of hard-control principle or condition the nodes in “1” (on state) and “0” (off state) only, wirelessly. At first, each node looks for the mesh network and goes through the reading process of the switch device status after obtaining the network.

The node for fan, lamp, and generic power socket executes the state of the relay commanded via the ESP8266’s GPIO port output while the electric door lock executes the status of the solenoid. Moreover, the read data for the sensor node were sent to the smartphone periodically without a first request.

The data format used on all nodes is JSON while the mesh program was created and configured to send data periodically every 5 seconds using Task. The data format for each node is as follows starting with the electric door lock and ending with the temperature & humidity.

Status data for door lock:
```
Status off: {"ID":"LOCK-001","STATUS":"OFF"}
Status on: {"ID":"LOCK-001","STATUS":"ON"}
```
Command data:
```
Command off: {"ID":"LOCK-001","CMD":"OFF"}
Command on: {"ID":"LOCK-001","CMD":"ON"}
```

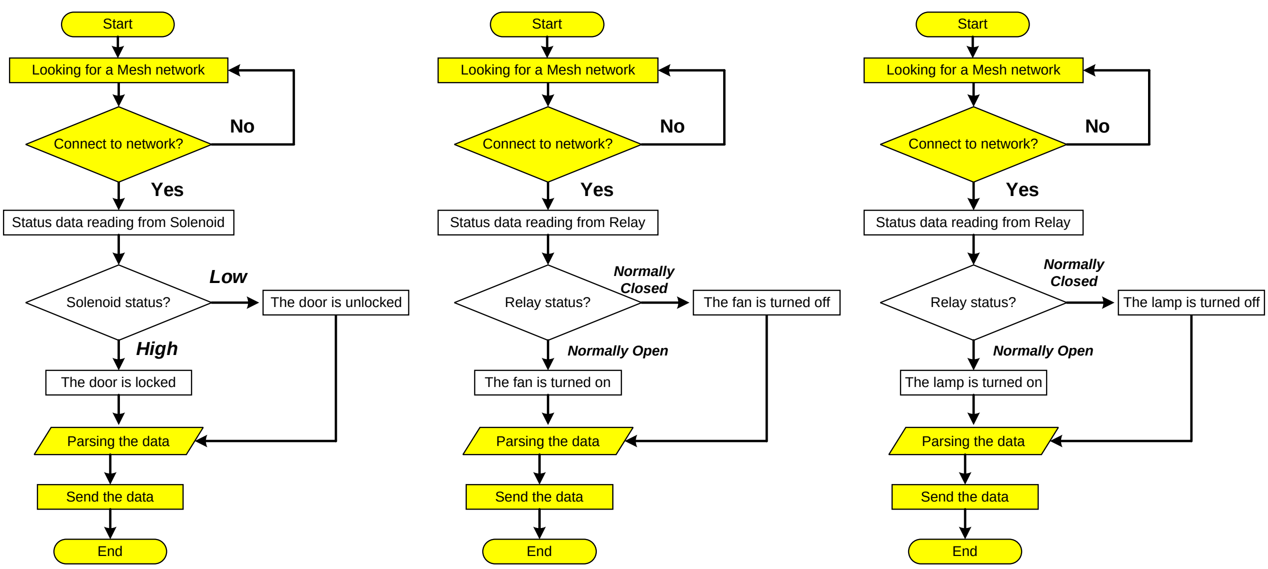
Status data for fan:
```
Status off: {"ID":"FANX-001","STATUS":"OFF"}
Status on: {"ID":"FANX-001","STATUS":"ON"}
```
Command data:
```
Command off: {"ID":"FANX-001","CMD":"OFF"}
Command on: {"ID":"FANX-001","CMD":"ON"}
```

Status data for lamp:
```
Status off: {"ID":"LAMP-001","STATUS":"OFF"}
Status on: {"ID":"LAMP-001","STATUS":"ON"}
```
Command data:
```
Command off: {"ID":"LAMP-001","CMD":"OFF"}
Command on: {"ID":"LAMP-001","CMD":"ON"}
```

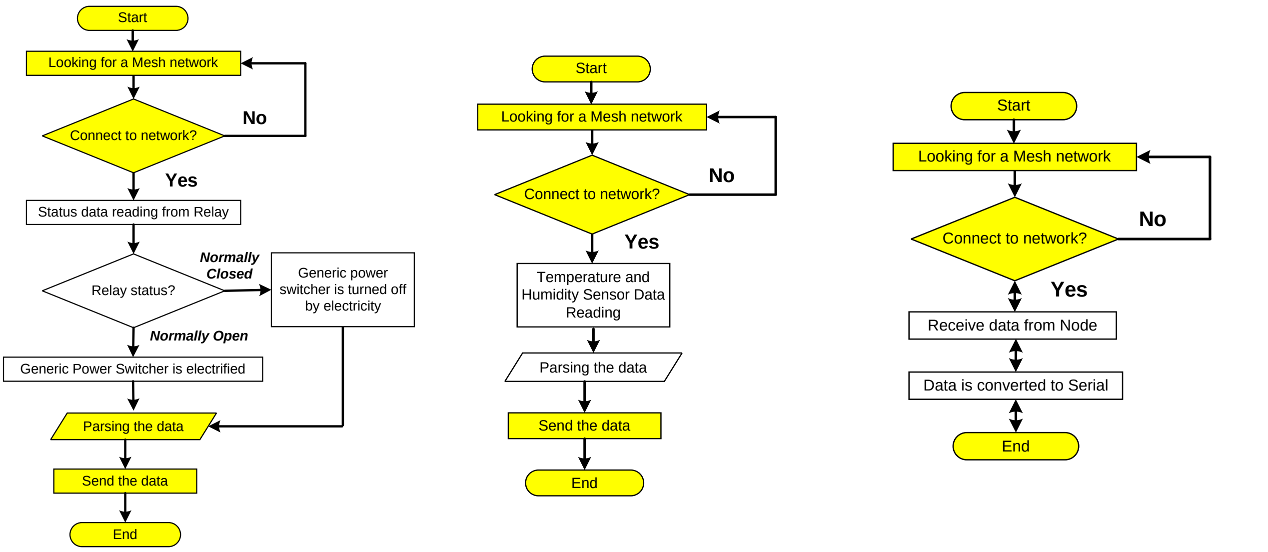
Status data for generic power switch:
```
Status off: {"ID":"STKR-001","STATUS":"OFF"}
Status on: {"ID":"STKR-001","STATUS":"ON"}
```
Command data:
```
Command off: {"ID":"STKR-001","CMD":"OFF"}
Command on: {"ID":"STKR-001","CMD":"ON"}
```

Data sent for humidity and temperature sensor:
```
{"ID":"TEMP-
001","TEMP":<nilai_temperature>,"HUM":<nilai
_humidity>}
```

For example:
```
{"ID":"TEMP-001","TEMP":28.4,"HUM":74}
```
It means, the recent temperature and Humidity are 28.4ºC and 74%, respectively.



(a) (b) (c)



(d) (e) (f)

Fig. 2. Flowchart of (a) electric door lock, (b) fan, (c) lamp, (d) generic power socket, (e) temperature and humidity, and (f) the gateway

The Gateway node was used to convert the data received from the mesh network into a serial data format which allowed sending the data back in other forms of communication by pairing a communication add-on or through a connection to a PC in order to view the data.

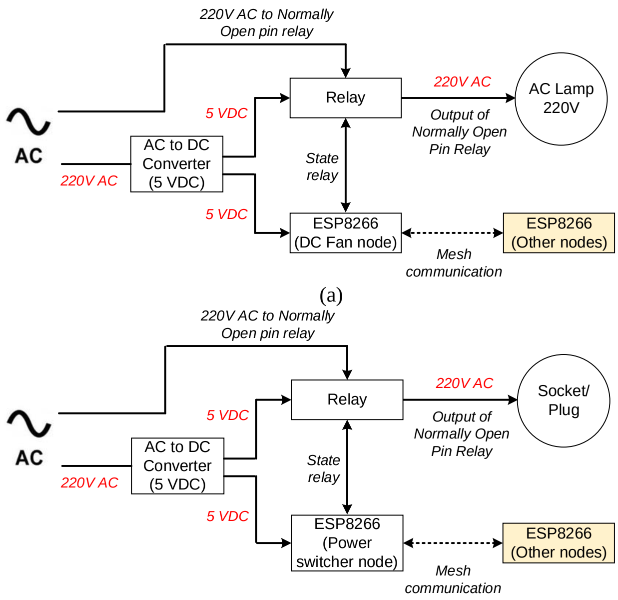
E. Hardware Design

The switch function embedded in the end device only performs an on or off function as previously explained. The main power source is ~220V$_{ac}$, therefore, a step-down AC-to-DC converter module was required (Hi-Link P/N: HLK-PM01 3W, input: 100-240 V$_{ac}$ 0.1A 50-60 Hz, output: 5 V$_{DC}$ 0.6 A). Moreover, the DC voltage was distributed to the ESP8266 and Relay device and the voltage requirements for the Solenoid supply ranged from ~220 V$_{ac}$ to 12 V$_{DC}$ which was converted through AC-to-DC (Hi-Link P/N: HLK-10M12 10W, input: 100-240 V$_{ac}$ 50-60 Hz, output: 12 V$_{DC}$ 0.83A).

The hardware block diagrams for lamps and generic power sockets nodes are the same with the difference observed in the output. The lamp node used a fitting which was later installed with a light-bulb as indicated in Figure 3a while the generic power socket node used a terminal as shown in Fig. 3b which was later used as a plug for household appliances. These nodes use an AC-to-DC converter module to convert the ~220V$_{ac}$ voltage into 5 V$_{DC}$ voltage supplied as the working voltage of the ESP8266 controller and 1 channel relay (Tong Ling model). Moreover, the ESP8266 controlled the relay to issue a 220 V$_{ac}$ voltage to turn the lamp on while there is a normally open (NO) and normally closed (NC) switch on the relay. The output voltage of the converter, 5 V$_{DC}$, was connected to the NO switch connected to household electronic devices or lamps.

Therefore, when the relay switch is in the NO position, the device becomes active and the connected lamp is switched on. Conversely, when in the NC position, the device is disabled and the lights are off. The ESP8266 was used to process and control the status of the relay by sending it through the mesh network to other nodes. The use of the same controlling function, on & off, makes this block diagram simpler compared to [30].

The function and hardware interface of the controller fan are the same as the electric door lock which requires a working voltage of 12 V$_{DC}$. Therefore, a functional approach technique was applied to speed up the design process and this involved adding an AC-to-DC converter module to convert from 220 V$_{ac}$ to 12 V$_{DC}$. This means the electric door lock and fan nodes used two AC-to-DC converter modules and these include the 5 V$_{DC}$ to supply the ESP8266 controller and 12 V$_{DC}$ to supply voltage to the switch device.



(a)

(b)

Fig. 3. Diagram block of (a) node lamp, (b) node generic power socket

The ESP8266 module on the electric door lock node was used to control the Solenoid to lock and open the door based on two states which are “High” for locking and “Low” for unlocking as indicated in Figure 4a. Meanwhile, the ESP8266 module at the fan node controlled the relay to turn the fan on and off as shown in Figure 4b. Moreover, the 12 V_{DC} voltage was connected to the NO switch of the fan to ensure the fan turns on when the relay switch is in the NO position and off when in the NC position. The ESP8266 was used to process and control the status of the relay and solenoid by sending it through the mesh network to other nodes.

The electric door lock device in a previous study used a transistor-based switch [31] and this caused heat problems in the semiconductor device [32]. However, the addition of a heatsink and fan is not the solution due to the fact that it has the ability to make the nodes bulky or increase in size. The electronic driver also wears out quickly when used for a long time. This means the device developed to perform the same function of locking and unlocking is safer from heat problems because it uses a relay.

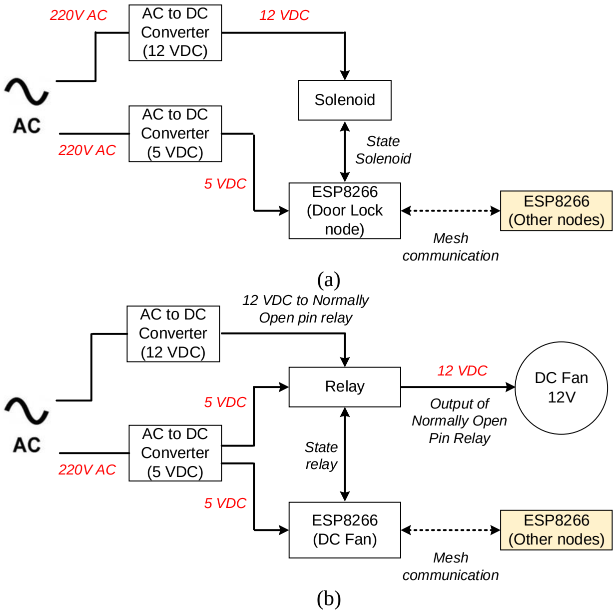


Fig. 4. Diagram block of (a) node electric door lock, (b) node fan

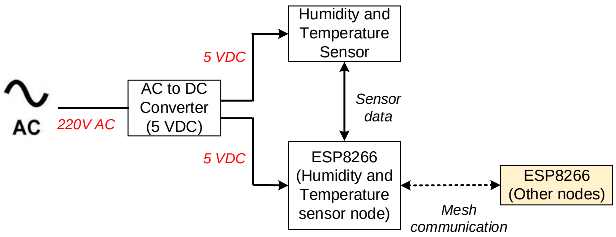


Fig. 5. Diagram block of node DHT11 sensor

There is also an AC-to-DC converter at the Temperature & Humidity node to convert voltage from 220V_{ac} to 5 V_{DC} required by the ESP8266 device and the DHT11 sensor. The sensor data was processed by the ESP8266 controller and the results of the data processing are sent through the mesh network to other nodes. Meanwhile, the block diagram of the DHT11 sensor node is presented in Fig. 5 and the sensor device was designed to require a 220 V_{ac} source compared to previous study which used a battery charger [33]. This block diagram for sensor nodes is observed to be simpler despite its low portability compared to [33].

F. Android Apps Design

The android application was used to display the data from temperature & humidity nodes as well as the "on" or "off" status of other nodes. In contrast to the Apps design in [34], [35] which performs functions outside the mesh network, the android applications in this study were included in the mesh network to ensure an easy connection between nodes in one network. Moreover, the library used for the mesh communication was Painlessmesh which works by sending data from the mesh node and resending it in the form of Broadcast to be received by the main activity. This method is often called a Broadcast Receiver which is a component or library in an Android application used in waiting for a Broadcast message or event from several sources, either the system or local network of the application. The block diagram of the android application created is presented in Fig. 6.

The SSID and password also need to be configured with the port used by the mesh network designed to be controlled wirelessly before the application is applied. This configuration was stored in Android shared preferences and this means there is no need to enter it again when using the next application.

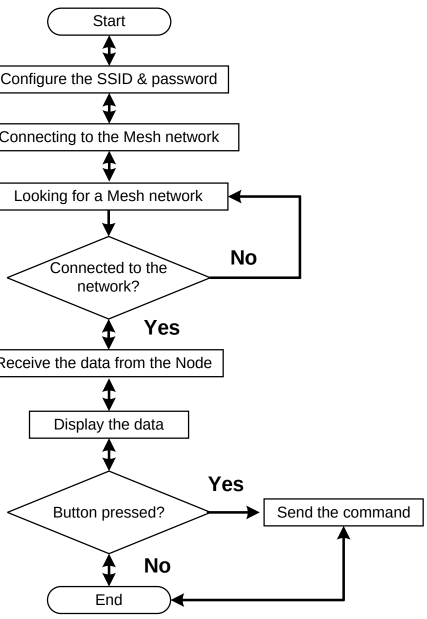


Fig. 6. Flowchart of android application

G. Security

The initial objective of this study was to prepare an IoT-based smart home infrastructure using hardware and software considered more efficient than those applied in previous study [25], [26]. The system was also proposed to be capable of supporting control and monitoring functions, both near and far through a Wi-Fi mesh (functional system aspect). It also has a self-reconfigurable network feature with a mesh network and designed in such a way to facilitate the installation and deployment process for Smart Home applications (flexibility & interoperability). The distance can be increased using the Internet with a Repeater or a Gateway connection. Moreover, the information from smartphones to nodes or vice versa is in the form of simple commands and monitor data, and this means the system is vulnerable to intrusion. It is important to note that security features are not the focus of this study and are recommended for future studies. Meanwhile, it was possible to apply the design protocol and regularly scheduled passwords as used in [36], [37] in this study. This is necessary due to the fact that smart homes require security standards in addition to the ease of integration to have a very good quality control process.

III. Results

A. Hardware and Software Implementation

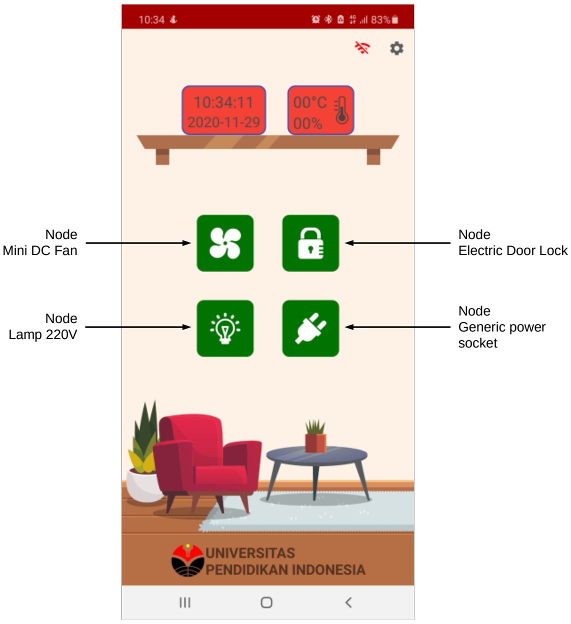


Fig. 7. UPISmartHome V.1.0 android application section

Fig. 7 shows the android application developed and named UPISmartHome V.1.0 consists of the following several parts:

1. The toolbar has a “Connect” and a “Setting SSID & Password” button. The first-time users need to input the SSID, Password, and the port used by the mesh network. This is to be followed by pressing the “Connect” button to connect the smartphone to the mesh network with the signal icon changing color and the strikethrough disappearing when it is connected.
2. The Body section contains buttons to operate the existing nodes with each designed to represent fan, electric door lock, lamp, and generic power socket. It is possible for the users to turn on or off the node by pressing the desired button when connected. The device is limited to being controlled through on/on and off/off only.
3. The Header section includes the Date & Time and Temperature & Humidity as indicated in Fig. 8.

The hardware realization process started from defining the required modules, designing electronic circuits, laying out the PCB, printing double layer PCBs, mounting, soldering, checking connections, and finally packaging. The hardware of each node developed is presented in Fig. 9. Moreover, the concept of interoperability or uniformity of the hardware design was based on the methods applied in previous study [25], [26]. This allows reducing the time for hardware realization without compromising the function and performance of each node [17]. It is also important to note that the hardware was made to be minimal and portable with the photos of the node parts presented in Fig. 10.

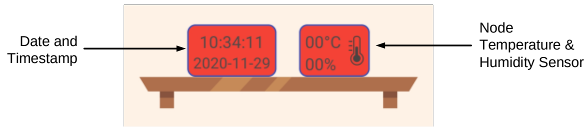


Fig. 8. The header section of the UPISmartHome V.1.0 application which displays temperature & humidity data

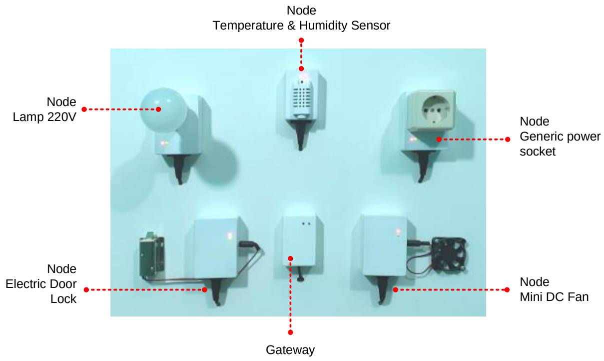
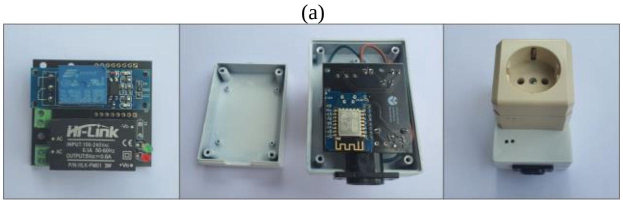
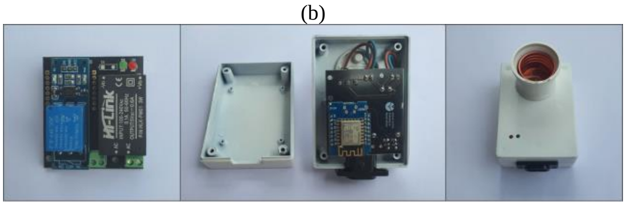


Fig. 9 Hardware of the nodes developed

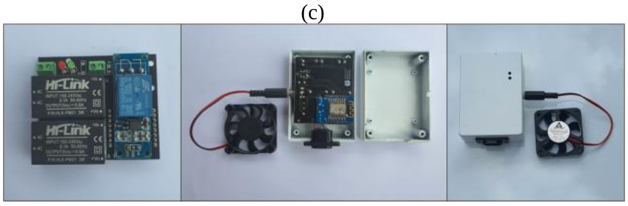
(a)



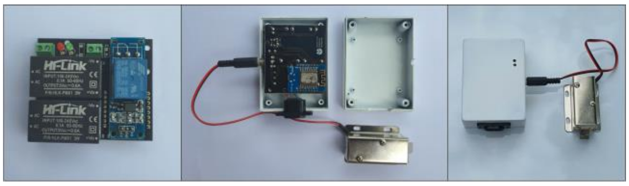
(b)



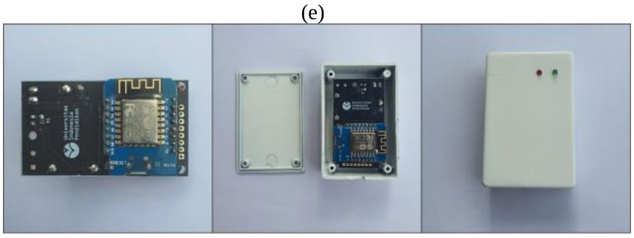
(c)



(d)



(e)



(f)

Fig. 10. The node packaging process: (a) temperature and humidity sensors, (b) generic power sockets, (c) lamps, (d) fans, (e) electric door locks, and (f) gateway

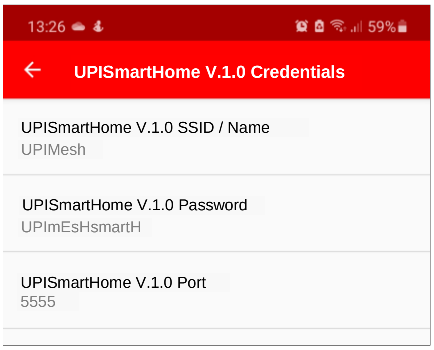
Each node was equipped with an AC power socket, a reset button, and two indicator lights such that the LED1 is for the power which indicates the node is connected to a voltage source and LED2 is to indicate the node is connected to the mesh network. There is an additional 220 V_{ac} output socket on the node for the lamp and generic power socket while those for the fan and electric door lock have additional 12 V_{DC} output sockets.

B. Functional Test

The development of the hardware and software was followed by the verification stage. The process was conducted on a personal computer connected via a cable to each node with an intermediary gateway before they were integrated wirelessly into the smartphone. This test has been successfully conducted in previous study [29].

The method to operate the first version of the android application was very simple and easy due to the fact that it only involves a brief press of the available button (when it is connected to the mesh network). The general operating stages of the UPISmartHome V.1.0. are presented as follows:

1) The first-time usage of the application requires entering the required SSID, password, and the port used by the network mesh as indicated in Fig. 11(a).
2) Then press the “Connect” button and the icon will change color to after the smartphone is connected to the mesh network as in Fig. 11(b).
3) Press the desired button when the connection has been established to turn the node “on” or “off”.



(a)

(b)

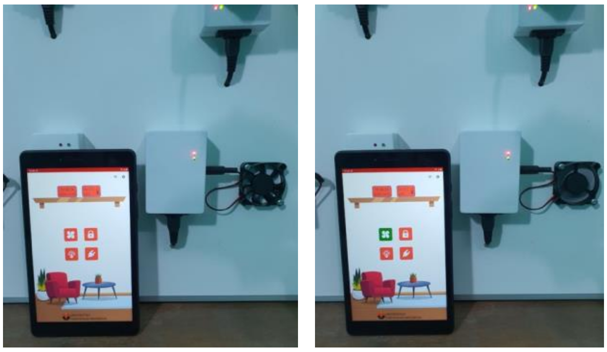
Fig. 11. Setting SSID, password, and port on UPISmartHome V.1.0; (b) Connect button

Fig. 12 to Fig. 16 show the functional aspect of the device has been properly verified. It was observed to have the ability to perform the monitoring and control functions (on/off) wirelessly using a special smartphone device, UPISmartHome V.1.0. All the node buttons were red at the initial or off state and turned green when the button was pressed to indicate the node condition is “on”.

Fig. 12. Temperature and humidity readings (39.8 degree Celsius of temperature & 47% of humidity)

Fig. 12 shows the temperature and humidity data recorded on the smartphone with the room temperature at the time found to be 39.8 Celsius while the humidity was 47%. The nodes in this study automatically send data periodically every 5 seconds or semi real-time unlike the previous study [33] which requires an advance request to determine the recorded data. Moreover, the fan node was verified to be working as expected as demonstrated with the power on in Fig. 13a and off in Fig. 13b. The key device (door lock) was also confirmed to have the ability to perform functions according to the designed specifications. The solenoid is always in the lock position under normal circumstances as indicated in Fig. 14a but it changed to the unlock position when the smartphone lock button was pressed as indicated in Fig. 14b. Furthermore, the generic power socket node in Fig. 15 was tested in a conventional light control scenario as indicated in Fig. 16. Both nodes were confirmed to be able to conduct the control function of turning on and off the household electronic equipment which was represented by lights using the same hardware.

The UPISmartHome V.1.0 android application was also able to control and monitor the household devices packaged in the application as depicted in Fig. 9.

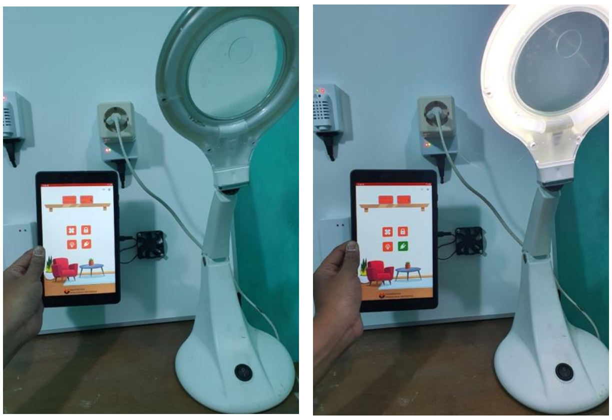


(a) (b)

Fig. 13. Fan test results: (a) off condition (b) on condition

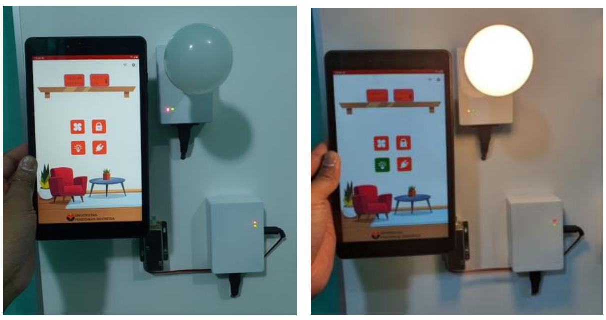
(a) (b)

Fig. 14. Test results for electric door lock nodes: (a) solenoid-push or lock condition, (b) solenoid-pull or unlock condition



(a) (b)

Fig. 15. Test results of generic power socket node: (a) off (b) on



(a) (b)

Fig. 16. ON/OFF test results for the lamp node.

C. Power Measurement

This test was conducted to determine the power consumption of each node before and after the mesh connection was formed. The voltage and current of each node were measured in two conditions which are the idle state when they are not connected to other nodes and the communication state. Table II shows that the current increased more in the active condition compared to the idle condition where it was not more than 10 mA. It is important to note that this test has also been conducted in previous studies.

TABLE II: RESULTS OF VOLTAGE AND CURRENT MEASUREMENT DURING IDLE AND ACTIVE CONDITIONS

node	*Idle*		*Active*	
	Voltage (Volt)	Current (mA)	Voltage (Volt)	Current (mA)
Mini DC fan	5.02	79.5	5.02	80.3
Electric Door Lock	5.04	84.3	5.04	87.4
Generic power socket	5.10	83.6	5.09	85.6
Lamp	5.07	82.5	5.07	86.9
Temperature & Humidity	5.05	79.1	5.06	83.8
Gateway	5.06	79.0	5.07	86.9

The power consumption values recorded in this study were compared with those from previous study as shown in Table III. Power is defined as the product of current and voltage and the results showed the device developed consumes lower power compared to [25], [26] when it is active for all nodes except for the sensor node which has more active status by sending data periodically, thereby, making its power consumption higher than in [25], [26]. For the record, this study did not use some of the nodes applied in previous studies such as RGB lamp [38], [39], horizontal-type curtain [40], [41], and fans with PWM control [44] which are tagged as Not Available (NA) in Table III.

TABLE III: COMPARISONS OF POWER CONSUMPTION

Node	Idle (mWatt)		Active (mWatt)	
	[25-26]	This work	[25-26]	This work
Mini DC fan	517.2	399.09	1741.2	403.106
Electric Door Lock	507.6	424.872	7050.0	440.496
Generic power socket	249.5	426.36	618.5	435.704
Lamp	N/A	418.275	N/A	440.583
Temperature & Humidity sensor	213.0	399.455	246.0	424.028
Gateway	N/A	399.74	N/A	440.583
RGB Lamp	214.0	N/A	3015.5	N/A
Curtain	210.5	N/A	1586.0	N/A

D. RSSI Measurement

The RSSI test was used to measure the received signal strength indicator between ESP8266 nodes [27] and also to determine the response provided between nodes in relation to the maximum distance. It was conducted outdoors by setting each node in a straight line (Line-of-sight configuration) without obstruction to ensure immediate receival of the signal from the sender.

Moreover, the nodes were placed at a certain predetermined distance of 10 to 200 meters with a range of 10 meters. The process was repeated five times on each node and the average RSSI value was obtained in the form of dBm as indicated in Table IV.

The results showed the nodes cannot be controlled and monitored at distances beyond 200 meters and the RSSI value of each node is not the same even though they used the same ESP8266 module. For example, the maximum average value for the sensor node at a distance of 120 meters was -91 dBm while the fan, electric door lock, and generic power socket nodes had ~94 dBm at 150 meters and the same value was recorded for the lamp node at 180 meters. This means the ESP8266 module has a good quality with some of the nodes having the ability to be placed far and some near [42]. Meanwhile, a distance of 120 meters was found to be reasonable to be used due to the fact that an average house in Indonesia is 10 m × 10 m in dimension. It is important to note that this test was not conducted in previous studies [25], [26].

TABLE IV: RSSI TEST BETWEEN NODES

node	**Distance(m)**																			
	10	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**	**100**	**110**	**120**	**130**	**140**	**150**	**160**	**170**	**180**	**190**	**200**
Gateway	-68	-70	-77	-72	-75	-79	-80	-81	-82	-83	-85	-87	-88	-91	-93	-93	-93	-93	-94	-94
Lamp	-63	-65	-68	-69	-80	-81	-82	-83	-83	-83	-85	-87	-88	-87	-90	-91	-91	-93	-	-
Generic power socket	-64	-65	-67	-79	-80	-83	-84	-85	-84	-86	-88	-91	-90	-92	-93	-	-	-	-	-
Lock	-56	-70	-72	-76	-82	-84	-92	-86	-88	-88	-87	-87	-90	-90	-94	-	-	-	-	-
Fan	-62	-64	-66	-68	-68	-71	-78	-79	-81	-83	-85	-86	-87	-87	-94	-	-	-	-	-
Temperature and Humidity	-65	-68	-70	-77	-80	-83	-87	-91	-91	-91	-90	-91	-	-	-	-	-	-	-	-

E. *Mesh Connection Test*

Another observed test which is not conducted in previous studies is the connection of mesh in an outdoor environment [25], [26]. This involved using field conditions which are not Line-of-Sight due to the presence of obstacles in the form of trees, grass, walls, and others to determine the distance each node can communicate with each other. The test scenario involved keeping the nodes away from each other in random positions up to the moment the wireless connection was lost. This was used to determine the maximum communication distance between nodes in a condition where they are unable to self-reconfigure.

TABLE V: MESH CONNECTION TEST BETWEEN NODES

Node	Gateway	Lamp	Fan	Lock	Switch	Sensor
Gateway	-	80 m	60 m	59 m	43.5 m	96 m
Lamp	-	-	166 m	160 m	87 m	131 m
fan	-	-	-	80 m	144 m	80 m
Lock	-	-	-	-	151 m	144 m
Switch	-	-	-	-	-	92 m

The results summarized in Table V showed the maximum communication distances for the gateway with light nodes, fans, electric door locks, generic power sockets, and sensors were 80 m, 60 m, 59 m, 43 m, and 96 m, respectively. Moreover, the furthest distance of 166 m was obtained between the light node and the fan while the shortest, 43.5 m, was between the gateway and a switch. The distances above 40 m are considered reasonable to be used for node control and monitoring.

F. *Comparison with Other Competitors*

Table VI compares the UPISmartHome developed with other competitors circulating in the country such as the Den Smart HomeTM and several others. The Den Smart Home Complete Package includes a smart remote, smart bulb, smart socket, smart switch, smart sensor, and smart camera and requires a gateway and special router to access. It is required to be connected to the internet due to the fact that the software from Tuya Smart used is connected to Google Home (Assistant) and Amazon Alexa. Moreover, other products also require that the nodes are connected to Wi-Fi except MINDS^{TM} which can be accessed without a Wi-Fi connection [26] due to its Zigbee mesh base. However, the system has a simpler communication mechanism caused by using only the ESP8266.

The IoT-Based Smart Home developed offers a variety of nodes which can be used without the internet and this means the device does not need a special gateway and router. This means the node control range is very limited within a distance of approximately 120 meters while routers and cloud servers are required to be accessible over a distance of 150 meters. Moreover, it is possible to connect the device to other communication modules such as NB-IoT, GSM, and LoRa but the mesh connection is still based on the ESP8266.

Several nodes with the same system architecture such as a lamp (RGB), curtain (horizontal & vertical types), indoor sensors (LPG gas & smoke), fan with speed control, IP camera, IR remote, human detector (PIR sensor), and others are proposed to be added in future studies. Power saver mode will also be included as presented in a previous study work [43].

TABLE VI: COMPARISON OF THIS STUDY WITH OTHER SMART HOME PRODUCTS

Commercial smart home products	Available nodes	Accessed by Internet	Accessed on "No Internet"
Den Smart Home™ (https://www.den-smarthome.com/smarthome-terbaik-indonesia)	Remote, bulbs, socket, switch, sensor, camera	Yes	
Bardi™ (https://bardi.co.id/products/)	Various nodes	Yes	
Igloohome smart locks™ (https://www.igloohome.co/) (https://kuncirumahku.com/)	Door locks	Yes	
ARBIT™ (https://rumah-pintar.id/product/)	Various nodes with voice control	Yes	
Hannoch™ (https://www.hannochs.com/futura/produk/)	Lamps	Yes	
Ezviz™ (https://www.ezvizlife.com/id/category/smart-home)	Various nodes	Yes	
Jon Powel™ (https://www.jonpowel.com/)	Curtain, switch, sockets, lamps, door lock, remote	Yes	
Bosman™ (http://www.bosman.id/Produk.htm)	Remote, lamps, camera, adaptor, towel dryer, coffee maker	Yes	
MINDS™ (https://www.pme.itb.ac.id/products/mind-meshed-and-internet-networked-devices-system/) (https://www.xirkachipset.com/product/minds)	Socket [30], RGB lamp [38], door lock [31], curtain [40], fan [44], remote [45-46], temperature and humidity sensor [33]	Yes	Yes
This product	Lamp, socket, door lock, fan (on/off), temperature and humidity sensor, involving the gateway	Yes	Yes

IV. CONCLUSION

An ESP8266- and IoT-based Smart Home System infrastructure was successfully developed. This involved using the ESP-mesh feature built on top of the Wi-Fi as a network protocol which combines several Wi-Fi networks into one WLAN. The system can be used within a considerable range because the nodes are connected and have the ability to forward data to the system. Moreover, the nodes were divided into three parts which include the mechanical such as door locks, electrical such as fans and generic power sockets, and sensors including the temperature & humidity sensors. It is possible to control the nodes through the hard-control method which consists of the on/off only. The device can be used to control and monitor nodes "with" or "without" the internet network due to the features provided by ESP-Mesh unlike those in the previous studies. Furthermore, the nodes can be automatically mesh-configured and the role of the actual gateway was eliminated because the smartphone functions as both the node and gateway. The device also excels in terms of system architecture simplicity due to its application of full Wi-Fi, hardware simplicity, and lower power consumption.

Security issues are recommended to be the focus of further studies. Moreover, some other nodes such as horizontal-type curtain, vertical-type curtain, RGB lamp, Toxic gas sensor, LPG sensor, PIR sensor, DC fan with soft-control (PWM-based control) are recommended to be added into the system to ensure more complete smart home devices.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHORS CONTRIBUTIONS

Syifaul Fuada and Hendriyana studied the existing literature and designed the experiment scenarios; Hendriyana designed the UI, programmed, performed the functional tests of the Android application; Hendriyana observed and compared the advantages of the proposed system against the available competitors; Syifaul Fuada designed the system architecture, designed the hardware structure, programmed the ESP-Mesh, layouted, mounted, and tested the PCB, performed the software/hardware integration and its functionality tests, observed the system performance, and contributed to the writing of the manuscript, drawn all of the figures. All authors had approved the final version.

ACKNOWLEDGMENT

This study was supported by the Ministry of Education, Culture, Study, and Technology of Republic Indonesia through grant number 788/UN40.D/PT/2020 under LPPM-Universitas Pendidikan Indonesia. This study is one of the top-112 Indonesia innovations awarded in 2020 by the Business Innovation Center (BIC) (http://bic.web.id/general/view/Menuju+Rumah+Cerdas+Nasional).

REFERENCES

[1] R. Steve. (2020). What is the IoT? Everything you need to know about the Internet of Things right now. ZDNet'. [Online]. Available: https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/

[2] A. Čolaković and M. Hadžialić, "Internet of Things (IoT): A review of enabling technologies, challenges, and open

study issues," *Computer Networks*, vol. 144, pp. 17–39, Oct. 2018.

[3] M. U. Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, "A Review on Internet of Things (IoT)," *IJCA*, vol. 113, no. 1, pp. 1–7, Mar. 2015.

[4] D. Thomas, E. Wilkie, and J. Irvine, "Comparison of power consumption of WiFi inbuilt internet of things device with bluetooth low energy," *International Journal of Computer and Information Engineering*, vol. 10, no. 10, p. 4, 2016.

[5] S. J. Danbatta and A. Varol, "Comparison of zigbee, Z-Wave, Wi-Fi, and bluetooth wireless technologies used in home automation," in *Proc. 7th International Symposium on Digital Forensics and Security*, Barcelos, Portugal, Jun. 2019, pp. 1–5.

[6] V. A. Mardiana, T. Adiono, S. Harimurti, M. M. M. Dinata, A. Mitayani, and G. N. Nurkahfi, "APSoC architecture design of 2.4 GHz zigbee baseband transceiver for IoT application," *Proc. of ICRAMET*, 2019.

[7] F. Javed, M. K. Afzal, M. Sharif, and B. S. Kim, "Internet of Things (IoT) operating systems support, networking technologies, applications, and challenges: A comparative review," *IEEE Commun. Surv. Tutorials*, vol. 20, no. 3, pp. 2062–2100, 2018.

[8] S. Tang, D. R. Shelden, C. M. Eastman, P. Pishdad-Bozorgi, and X. Gao, "A review of building information modeling (BIM) and the internet of things (IoT) devices integration: Present status and future trends," *Automation in Construction*, vol. 101, pp. 127–139, May 2019.

[9] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, "A review of smart home applications based on Internet of Things," *Journal of Network and Computer Applications*, vol. 97, pp. 48–65, Nov. 2017.

[10] K. Shafique, B. A. Khawaja, F. Sabir, S. Qazi, and M. Mustaqim, "Internet of Things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios," *IEEE Access*, vol. 8, pp. 23022–23040, 2020.

[11] S. Taj, U. Asad, M. Azhar, and S. Kausar, "Interoperability in IOT based smart home: A review," *RCES*, vol. 5, no. 3, pp. 50–55, Sep. 2019.

[12] M. O. Farooq, I. Wheelock, and D. Pesch, "IoT-Connect: An interoperability framework for smart home communication protocols," *IEEE Consumer Electron. Mag.*, vol. 9, no. 1, pp. 22–29, Jan. 2020.

[13] C. Yang, B. Yuan, Y. Tian, Z. Feng, and W. Mao, "A smart home architecture based on resource name service," in *Proc. IEEE 17th International Conference on Computational Science and Engineering*, Chengdu, China, Dec. 2014, pp. 1915–1920.

[14] V. Ricquebourg, D. Menga, D. Durand, B. Marhic, L. Delahoche, and C. Loge, "The smart home concept: Our immediate future," in *Proc. 1ST IEEE International Conference on E-Learning in Industrial Electronics*, Hammamet, Tunisia, Dec. 2006, pp. 23–28.

[15] M. Y. Fathany, "Perancangan dan implementasi infrastruktur sistem rumah cerdas," M.T. thesis, Dept. Elect. Eng., School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia, 2016.

[16] S. S. I. Samuel, "A review of connectivity challenges in IoT-smart home," in *Proc. 3rd MEC International Conference on Big Data and Smart City*, Muscat, Oman, Mar. 2016, pp. 1–4.

[17] S. Fuada and T. Adiono, "The importance of lightweight implementation concept for nodes in smart home system," *Electrical & Electronic Technology Open Access Journal*, vol. 2, no. 2, pp. 266-267, November 2018.

[18] Z. Yin, Y. Che, and W. He, "A hierarchical group control method of electrical loads in smart home," in *Proc. 6th International Conference on Power Electronics Systems and Applications*, Hong Kong, Hong Kong, Dec. 2015, pp. 1–6.

[19] S. Hameed, F. I. Khan, and B. Hameed, "Understanding security requirements and challenges in Internet of Things (IoT): A review," *Journal of Computer Networks and Communications*, vol. 2019, pp. 1–14, Jan. 2019.

[20] 'IoT' Sudah Merambah. Rumah. [Online]. Available: http://bic.web.id/general/view/%22IoT%22+Sudah+Merambah+Rumah

[21] CEBIT Events Worldwide [Online]. Available: http://www.cebit.de/product/minds-meshed-and-internetnetwork-devi/2048320/H681699

[22] T. Adiono, R. Muttaqin, R. M. Rahayu, and R. V. W. Putra, "Cloud system design using AMQP protocol for smart devices system applications," INA-Rxiv, preprint, Jun. 2020.

[23] S. Fuada, R. Muttaqin, M. Y. Fathany, T. Adiono, and M. H. Santriaji, "Raspberry Pi as a gateway for smart home system," *Unpublished.*

[24] S. Fuada, T. Adiono, and L. Siregar, "Internet-of-Things for smart street lighting system using ESP8266 on mesh network," *Int. J. of Recent Contributions from Engineering, Science, & IT*, vol. 9, no. 2, pp. 73-78, 2021.

[25] T. Adiono, R. V. W. Putra, M. Y. Fathany, B. L. Lawu, K. Afifah, M. H. Santriaji, and S. Fuada, "Rapid prototyping methodology of lightweight electronic drivers for smart home appliances," *Int. J. of Electrical and Computer Engineering*, vol. 6, no. 5, pp. 2114-2124, October 2016.

[26] T. Adiono, M. Y. Fathany, R. V. W. Putra, K. Afifah, M. H. Santriaji, B. L. Lawu, and S. Fuada, "Live demonstration: MINDS-Meshed and internet networked devices system for smart home," in *Proc. 13th IEEE Asia Pacific Conference on Circuits and Systems*, October 2016, pp. 736-737.

[27] S. Fuada, T. Adiono, and L. Siregar, "Development of wireless control system for smart street lighting using ESP8266," *Int. J. of Interactive Mobile Technologies*, vol. 14, no. 15, pp. 68-80, September 2020.

[28] L. Siregar, "Sistem penerangan lampu jalan pintar berbasis komunikasi MESH," M.T. thesis, Dept. Elect. Eng., School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia, 2019.

[29] S. Fuada, Hendriyana, H. E. Putri, J. Caturiasari, and F. R. Jannah, "Preliminary design of Simple IoT-based smart home trainer for kids," *Journal of Physics: Conference Series*, 1987.

[30] T. Adiono, M. Y. Fathany, S. F. Anindya, S. Fuada, I. G. Purwanda, "Using a smart plug based on consumer electronics to support low power smart home," in *Proc. of*

the 4th Int. Conf. on Intelligent Green Building Smart Grdi, 2019, pp. 376-379.

[31] T. Adiono, S. Fuada, S. F. Anindya, I. G. Purwanda, and M. Y. Fathany, "IoT-Enabled door lock system," *Int. J. of Advanced Computer Science and Applications*, vol. 5, no. 2, pp. 445-449, 2019.

[32] T. Adiono, S. Fuada, A. P. Putra, and Y. Aska, "Desain awal analog front-end optical transceiver untuk aplikasi visible light communication," *J. Nasional Teknik Elektro dan Teknologi Informasi*, vol. 5, no. 4, pp. 319-327, November 2016.

[33] T. Adiono, M. Y. Fathany, S. Fuada, I. G. Purwanda, and S. F. Anindya, "A portable node of humidity and temperature sensor for indoor environment monitoring," in *Proc. 3rd Int. Conf. on Intelligent Green Building and Smart Grid*, 2018, pp. 1-5.

[34] K. Afifah, S. Fuada, R. V. W. Putra, T. Adiono, and M. Y. Fathany, "Design of low power mobile application for smart home," in *Proc. Int. Symposium on Electronics and Smart Devices*, November 2016, pp. 127-131.

[35] T. Adiono, S. F. Anindya, S. Fuada, K. Afifah, and I. G. Purwanda, "Efficient android software development using MIT App inventor 2 for bluetooth-based smart home," *Wireless Pers Commun*, vol. 105, no. 1, pp. 233-256, March 2019.

[36] T. Adiono, B. Tandiawan, and S. Fuada, "Device protocol design for security on internet of things based smart home," *Int. J. of Online Engineering*, vol. 14, no. 7, pp. 161-170, 2018.

[37] T. Adiono, R. Marthensa, R. Muttaqin, S. Fuada, S. Harimurti, and W. Adijarto, "Design of database and secure communication protocols for internet-of-things-based smart home system," in *Proc. of IEEE Region 10 Conf.*, November 2017, pp. 1273-1278.

[38] T. Adiono, M. Y. Fathany, S. F. Anindya, S. Fuada, and I. G. Purwanda, "Wirelessly control for RGB lamp end-device: Design and implementation," in *IEEE Region 10 Conf.*, October 2018, pp. 2066-2070.

[39] T. Adiono, R. V. W. Putra, M. Y. Fathany, K. Afifah, M. H. Santriaji, B. L. Lawu, and S. Fuada, "Prototyping design of electronic end-devices for smart home applications," in *Proc. IEEE Region 10 Symposium*, May 2016, pp. 1-5.

[40] T. Adiono, S. F. Anindya, S. Fuada, and M. Y. Fathany, "Curtain control systems development on mesh wireless network of the smart home," *Bulletin of Electrical Engineering and Informatics*, vol. 7, no. 4, pp. 615-625, December 2018.

[41] B. L. Lawu, M. Y. Fathany, K. Afifah, M. H. Santriaji, R. V. W. Putra, S. Fuada, and T. Adiono, "Prototyping design of mechanical based end-devices for smart home applications," in *Proc. 4th Int. Conf. on Information and Communication Technology*, May 2016, pp. 1-5.

[42] Yoppy, R. H. Arjadi, E. Setyaningsih, P. Wibowo, and M. I. Sudrajat, "Performance evaluation of ESP8266 mesh networks," *J. of Physics: Conference Series*, vol. 1230, 2018.

[43] N. A. Prasetyo, A. G. Prabawati, and S. Suyoto, "Smart home: Power electric monitoring and control in Indonesia," *Int. J. Interact. Mob. Technol.*, vol. 13, no. 3, pp. 143-151, Mar. 2019.

[44] S. Fuada, M. Y. Fathany, T. Adiono, and K. Afifah, "Controlling mini exhaust fan through android-based smartphone for IoT-Based smart home system," *TEM Journal*, vol. 10, no. 3, pp. 1301-1306, August 2021.

[45] T. Adiono, B. Tandiawan, M. Y. Fathany, W. Adijarto, and S. Fuada, "Prototyping design of IR remote controller for smart home applications," in *Proc. IEEE Region 10 Conf.*, November 2017, pp. 1304-1308.

[46] T. Adiono, S. F. Anindya, S. Fuada, and M. Y. Fathany, "Developing of general IrDa remote to wirelessly control IR-Based home appliances," in *Proc. IEEE 7th Global Conf. on Consumer Electronics*, 2018, pp. 461-463.

Syifaul Fuada obtained a B.A. in Electrical Engineering Education from Universitas Negeri Malang (UM), Indonesia and M.Sc. in Microelectronics option of Electrical Engineering from the School of Electrical Engineering and Informatics, Institut Teknologi Bandung (ITB), Indonesia. He was with the University Center of Excellence at Microelectronics ITB from 2016-2018 as a main studyer. He is presently with the Program Studi Sistem Telekomunikasi Universitas Pendidikan Indonesia (UPI) as a Lecturer. His study interests include analog circuit design and instrumentation, circuit simulation, engineering education, IoT, multimedia learning development, and Visible Light Communication.

Hendriyana obtained a Diploma degree from Padjadjaran University (UNPAD) in 2007, Bachelor degree from Pasundan University (UNPAS) in 2013, and Master degree from Telkom University (TEL-U), Bandung in 2017. He worked as an analyst on Industrial Banking from 2008 – 2014 and now working in the Program Studi Rekayasa Perangkat Lunak (Software Engineering) Universitas Pendidikan Indonesia (UPI), Bandung. His study interests include ICT business, information system, database, decision support system, multimedia, and datamining.