

Smart Home Berbasis IoT Menggunakan Suara Pada Google Assistant

Ajib Hanani¹, Mokhamad Amin Hariyadi²

^{1,2}UIN Maulana Malik Ibrahim Malang

¹ajib@uin-malang.ac.id, ²adyt2002@uin-malang.ac.id

ABSTRAK. Penelitian ini membangun sebuah *Smart Home* berbasis *IoT* menggunakan suara pada *Google Assistant*. Hal ini dibutuhkan sebagai solusi untuk orang sakit yang berada di kursi roda/tempat tidur atau orang disabilitas tetapi dapat berbicara atau orang lanjut usia yang tidak dapat mencapai saklar agar dapat menghidupkan/mematikan perangkat rumah. Selain itu, agar dapat mengontrol perangkat rumah dari jarak yang sangat jauh. Sistem yang dibangun menggunakan perintah suara pada aplikasi *Google Assistant* di *android*. *Google Assistant* mengubah perintah suara menjadi teks. Teks tersebut kemudian diteruskan dari *Google Assistant* ke *Webhooks* oleh *IFTTT*. *Webhooks* akan melakukan *request* ke *HTTP RESTful API*. Dengan *library phpMQTT* yang terdapat di *HTTP RESTful API*, perintah di *publish* ke *MQTT Broker*. *ESP32 Dev Kit* sebagai *microcontroller* yang terhubung dengan *internet* menerima perintah dari *MQTT Broker* untuk menyalakan atau mematikan lampu yang ada di rumah. Pada pengujian sistem telah berhasil menyalakan dan mematikan lampu dengan perintah suara menggunakan *Google Assistant*.

Kata Kunci: *Google Assistant*; *IFTTT (If This Than That)*; *IoT (Internet of Things)*; *Smart Home*; *Voice Control*

ABSTRACT. This research builds an IoT based Smart Home using voice command on Google Assistant. It is a solution for sick people who are in wheelchairs / beds or people with disabilities but can speak or elderly people who cannot reach the switch in order to turn on / turn off home devices. In addition, we able to control home devices from any where. The system is built using voice command on the Google Assistant application. Google Assistant converts the voice command to the text command. And then, the text command is forwarded from Google Assistant to Webhooks by IFTTT. Webhooks makes a request to the HTTP RESTful API. The text command is published to MQTT Broker by phpMQTT library available on the HTTP RESTful API. ESP32 Dev Kit as an internet connected microcontroller receives text command from MQTT Broker to turn on or turn off the lights in home. The system testing has succeeded in turning on and turning off the lights with voice commands using Google Assistant.

Keywords: *Google Assistant*; *IFTTT (If This Than That)*; *IoT (Internet of Things)*; *Smart Home*; *Voice Control*

1. PENDAHULUAN

Dalam kehidupan sehari-hari di dalam rumah tentunya kita selalu menggunakan perangkat rumah untuk melakukan beberapa hal untuk kebutuhan kita, misalnya menyalakan lampu untuk penerangan rumah, menyalakan kipas angin ketika ketika suhunya panas, menyalakan TV/radio, dll. Hal ini menjadi hal biasa bagi orang yang sehat untuk melakukannya. Akan tetapi, bagaimana dengan orang sakit yang berada di kursi roda/tempat tidur atau orang disabilitas atau orang lanjut usia yang tidak dapat mencapai saklar ketika ingin menghidupkan/mematikan perangkat rumah dan mereka tidak dapat melakukannya. Begitu juga dengan orang yang bepergian keluar kota, yang mana mereka tidak dapat mengontrol perangkat rumah dari jarak yang sangat jauh. Hal ini tentunya menjadi kebutuhan bagi mereka untuk membangun sebuah *smart home* berbasis *IoT (Internet of Things)* yang dapat membantu mereka mengontrol perangkat rumah dengan mudah dan dapat dikontrol dari mana saja.

Pada tahun 2015, Satria mengungkapkan bahwa sebagian besar sistem otomasi rumah menggunakan *Bluetooth* di perangkat *mobile*. Ini memiliki berbagai kekurangan dalam setiap aspek karena *Bluetooth* memiliki jangkauan yang sangat pendek dan sistem yang dibangun hanya dapat bekerja di rumah-rumah kecil, tidak termasuk rumah-rumah dengan area yang lebih luas. *Bluetooth* tidak mendukung jarak jauh, perangkat tidak dapat dikontrol oleh *Bluetooth* dari wilayah geografis yang jauh. Penelitian juga dilakukan oleh Dani, Adriansyah dan Hermawan pada tahun 2016 tentang *Smartphone Android* yang dapat digunakan untuk melakukan kontrol terhadap perangkat lain menggunakan *Bluetooth* dan *Microcontroller Arduino Uno*. *Bluetooth* memiliki jarak kontrol lebih dari 10 meter, baik kondisi tidak ada halangan maupun terhalang oleh tembok. Sistem kontrol dibangun menggunakan *bluetooth*, yang mana dalam penggunaannya masih terbatas dengan jarak yang pendek dan tidak untuk jarak yang sangat jauh. Bagi mereka yang ingin mengontrol perangkat rumah dari jarak jauh tentunya menjadi sebuah kendala, maka dari pada penelitian yang dikembangkan menggunakan internet agar dapat diakses dari jarak jauh.

Penelitian yang terkait dengan sistem kontrol jarak jauh adalah Khedkar dan Malwatkar pada tahun 2016 yang telah membuat aplikasi otomasi rumah menggunakan *RPi* dan *GSM* dengan pemrograman *Python* pada sistem operasi *RPi*. Dalam hal ini komunikasi antara pengguna jarak jauh dan perangkat rumah diimplementasikan menggunakan *GSM* dan *SMS* yang menggunakan perintah *AT*. Menurut Sen (2015), sebagian besar implementasi sistem otomasi perangkat rumah bergantung pada teknologi *GSM* karena dapat digunakan secara jarak jauh. Hal ini berguna ketika pengguna dapat mengontrol perangkat rumah dari tempat yang jauh. Kodali dan Soratkal (2016) dalam penelitiannya mengungkapkan bahwa dengan adanya kemajuan di bidang industri otomasi dan *wireless*, semua perangkat yang ada di dalam rumah dapat terhubung. Perangkat yang berukuran kecil dapat digunakan untuk memonitor kondisi di dalam rumah dan mengontrol peralatan rumah dari jarak jauh. Perangkat berdaya rendah dibutuhkan untuk mentransmisikan pesan melalui protokol. Dalam penelitian tersebut mengimplementasikan *MQTT* sebagai protokol yang ringan pada *ESP8266*. Sensor dan aktuator terhubung ke *ESP8266* dan broker *MQTT* berbasis *Mosquitto* dibangun untuk pemantauan dan kontrol jarak jauh. Penelitian yang dilakukan oleh Pavithra dan Balakrishnan pada tahun 2015 mengimplementasikan *IoT* yang digunakan untuk memantau dan mengendalikan peralatan rumah tangga melalui *World Wide Web*. Sistem otomasi rumah menggunakan perangkat portabel sebagai *interface* pengguna. Perangkat portabel tersebut dapat berkomunikasi dengan jaringan otomatisasi rumah melalui internet dengan menggunakan protokol komunikasi berdaya rendah seperti *Zigbee*, *Wi-Fi*, dll. Perangkat yang dikembangkan memiliki kesamaan dalam mengontrol perangkat dari jarak jauh, yaitu menggunakan perangkat yang berukuran kecil, berdaya rendah dan ringan seperti *GSM*, *MQTT*, *Wi-Fi* dan *microcontroller*. Akan tetapi, memiliki perbedaan dimana perangkat yang dikembangkan menggunakan suara dalam mengontrol perangkat rumah sedangkan penelitian sebelumnya menggunakan aplikasi berbasis *web*.

Selain menggunakan tombol *On/Off* pada *interface* berbasis *web* maupun *mobile*, *smart home* dapat dikontrol dengan menggunakan perintah suara seperti yang diungkapkan oleh Caranica pada tahun 2017. Caranica menyatakan bahwa kemajuan terbaru dalam teknologi pengenalan suara telah membuat *smart home* dapat dikontrol dengan menggunakan suara. Paul Jasmin pada tahun 2016 membangun sistem menggunakan perintah suara untuk berinteraksi dengan peralatan rumah menggunakan *Arduino* dan *mobile*. Perintah suara ini membantu pengguna untuk lebih interaktif dengan teknologi. Kumar dan Shimi pada tahun 2015 juga membangun sistem pengenalan suara yang terdiri dari mikrokontroler *Arduino* dan *relay*. Perangkat yang dikembangkan memiliki kesamaan dalam mengontrol perangkat rumah menggunakan suara. Akan tetapi, memiliki perbedaan dimana perangkat yang dikembangkan menggunakan *Google Assistant* sedangkan penelitian sebelumnya tidak. Perbedaan lainnya terletak pada desain *software* dan *hardware*.

Google Assistant menyediakan *interface* suara dan dapat diintegrasikan dengan *webhooks* pada *IFTTT* (*If This Than That*) yang dapat digunakan sebagai *voice control* dalam membangun sebuah *smart home* (*IFTTT*, 2020). *Smart home* yang dibangun diharapkan dapat diimplementasikan dan digunakan untuk orang sakit yang berada di kursi roda/tempat tidur atau orang disabilitas tetapi dapat berbicara atau orang lanjut usia yang tidak dapat mencapai saklar agar dapat menghidupkan/mematikan perangkat rumah. Selain itu, agar dapat mengontrol perangkat rumah dari jarak yang sangat jauh. Berdasarkan dari uraian tersebut, maka penelitian ini mengimplementasikan *smart home* berbasis *IoT* menggunakan suara pada *Google Assistant*.

2. METODE PENELITIAN

Pada metode penelitian ini membahas tentang Perancangan Rangkaian *ESP32 Dev Kit* dan *Relay Board*, *Library EspMQTTCClient* pada *Arduino IDE*, *MQTT Broker* dan *HTTP RESTful API*, Integrasi *Google Assistant* dan *Webhooks* pada *IFTTT*, Desain keseluruhan Sistem *Smart Home* berbasis *IoT*.

2.1. Perancangan Rangkaian *ESP32 Dev Kit* dan *Relay Board*

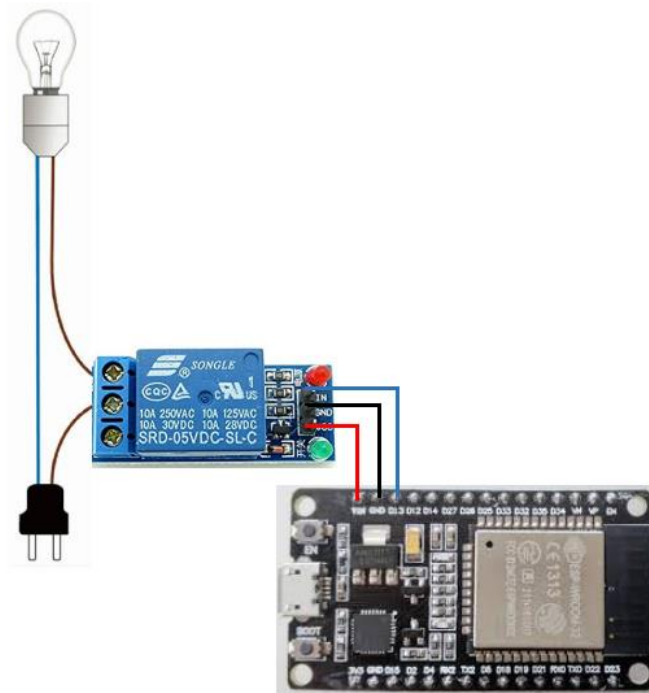
Microcontroller yang digunakan pada *smart home* ini adalah *ESP32 Dev Kit*. *ESP32* merupakan pengembangan dari *ESP266*. *ESP32 Dev Kit* terhubung dengan internet melalui jaringan *WiFi* dan menerima perintah dari *MQTT Broker* untuk menyalakan atau mematikan lampu yang ada di rumah melalui *relay board*. Konfigurasi pin *relay board* dengan pin *ESP32 Dev Kit* adalah sebagai berikut :

1. Pin VCC dihubungkan ke pin tegangan Positif (+) 5V (VIN).
2. Pin GND dihubungkan ke pin tegangan Negatif (-) atau *ground* (GND).
3. Pin IN dihubungkan ke pin D13.

Konfigurasi pin *relay board* dengan kabel lampu adalah sebagai berikut :

1. Pin C dihubungkan ke tegangan Positif (+) dari jalur arus AC.
2. Pin NO dihubungkan ke lampu.

Rangkaian *ESP32 Dev Kit* dan *Relay Board* terdapat pada gambar 1.



Gambar 1. Rangkaian ESP32 Dev Kit dan Relay Board.

2.2. Library EspMQTTClient pada Arduino IDE

Pada *Arduino IDE* menggunakan library *EspMQTTClient* untuk koneksi ke WiFi yang terhubung ke internet, koneksi ke MQTT broker server dan untuk mengontrol relay board pada pin D13. Source code pada *Arduino IDE* menggunakan library *EspMQTTClient* terdapat pada gambar 2.

```

File Edit Sketch Tools Help
VoiceMQTT
#include "EspMQTTClient.h"

EspMQTTClient client(
  "TP-Link_93AC", // SSID
  "79364926", // Password WiFi
  "103.xxx.xxx.xxx", // MQTT Broker Server IP
  "ajib", // MQTT User
  "xxxxxxxxxx", // MQTT Password
  "ajibhanani", // Client name
  1883 // The MQTT port
);

void setup()
{
  Serial.begin(115200);
  pinMode(13, OUTPUT);
  client.enableDebuggingMessages();
  client.enableHTTPWebUpdater();
}

void onConnectionEstablished()
{
  client.subscribe("IoT", [](const String & payload) {
    Serial.println(payload);

    if (payload=="L1") {
      digitalWrite(13, HIGH); // lampu nyala
    } else if (payload=="L0") {
      digitalWrite(13, LOW); // lampu mati
    }
  });
  client.publish("IoT", "Terkoneksi");
}

void loop()
{
  client.loop();
}
Done Saving.

```

Gambar 2. Source code pada *Arduino IDE* menggunakan library *EspMQTTClient*.

2.3. MQTT Broker dan HTTP RESTful API

Pada penelitian ini, server yang digunakan adalah server kampus yang memiliki IP Publik sehingga dapat diakses dimanapun kita berada. MQTT Broker menggunakan Mosquitto yang di-install dan dikonfigurasi pada Server. Secara default, MQTT berjalan pada port 1883. Untuk keamanan, kita dapat menggunakan autentikasi user. RESTful API menggunakan bahasa pemrograman PHP dan library phpMQTT yang terdapat pada Server. RESTful API ini sebagai service yang berfungsi untuk menjembatani antara IFTTT dengan MQTT Broker. phpMQTT adalah class php sederhana yang digunakan untuk membuat koneksi, subscribe dan publish ke MQTT Broker. Source code HTTP RESTful API menggunakan pemrograman PHP dan library phpMQTT terdapat pada gambar 3.

```

1  <?php
2  if (isset($_SERVER['HTTP_ORIGIN'])) {
3      header("Access-Control-Allow-Origin: {$_SERVER['HTTP_ORIGIN']}");
4      header('Access-Control-Allow-Credentials: true');
5      header('Access-Control-Max-Age: 86400'); // cache for 1 day
6  }
7  if ($_SERVER['REQUEST_METHOD'] == 'OPTIONS') {
8      if (isset($_SERVER['HTTP_ACCESS_CONTROL_REQUEST_METHOD']))
9          header("Access-Control-Allow-Methods: GET, POST, OPTIONS");
10     if (isset($_SERVER['HTTP_ACCESS_CONTROL_REQUEST_HEADERS']))
11         header("Access-Control-Allow-Headers: {$_SERVER['HTTP_ACCESS_CONTROL_REQUEST_HEADERS']}");
12     exit(0);
13 }
14
15 if (($_GET['user']=='ajib') and ($_GET['password']=='xxxxxx') and (empty($_GET['msg']))) {
16     mqtt_google_assistant($_GET['msg']);
17     unset($_GET['user'], $_GET['password'], $_GET['msg']);
18 }
19
20 function mqtt_google_assistant($msg)
21 {
22     require("phpMQTT.php");
23
24     $port = 1883;
25     $topik = "IoT";
26     $client_id = "ajibhanani";
27     $host = "103.xxx.xxx.xxx";
28     $username = "ajib";
29     $password = "xxxxxxxxxx";
30
31     $mqtt = new bluewin\phpMQTT($host,$port,$client_id);
32     if ($mqtt->connect(true,NULL,$username,$password))
33     {
34         $mqtt->publish($topik,$msg,0);
35         $mqtt->close();
36         echo json_encode(["respon:'Sukses'"]);
37     } else {
38         echo json_encode(["respon:'Gagal'"]);
39     }
40     unset($mqtt,$host,$port,$client_id,$username,$password,$topik,$msg);
41 }
42

```

Gambar 3. Source code HTTP RESTful API menggunakan pemrograman PHP dan library phpMQTT.

2.4. Integrasi Google Assistant dan Webhooks pada IFTTT

Tahap integrasi Google Assistant dan Webhooks pada IFTTT adalah sebagai berikut:

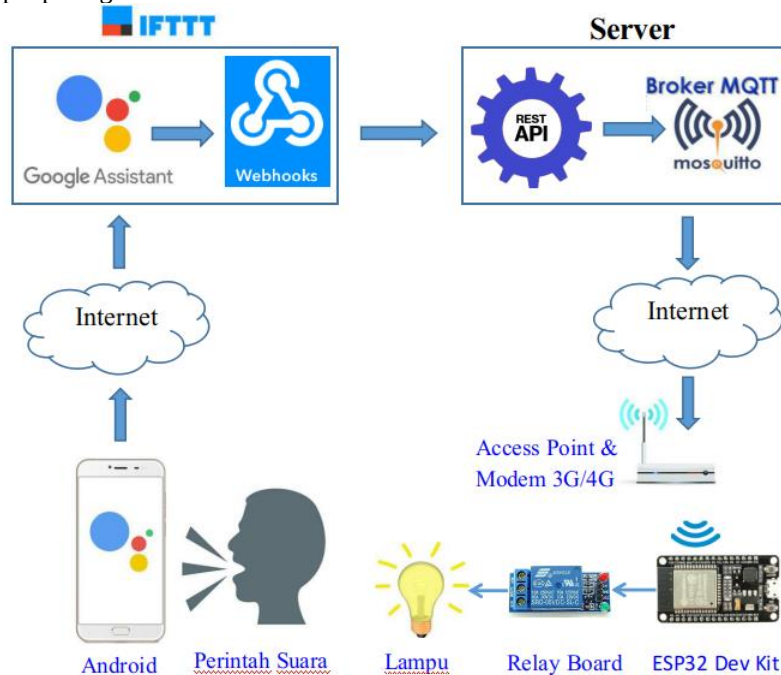
- Kita dapat membuat perintah pada Google Assistant menggunakan IFTTT untuk menyalakan dan mematikan lampu. Pada ponsel Android, kita buka *ifttt.com*. Login atau daftar jika kita belum melakukannya. Pada tap "Search" di bagian atas, kita telusuri "Google Assistant". Tap "Google Assistant > Connect". Pilih Akun Google agar IFTTT dapat mengaksesnya. Kita harus memilih akun yang digunakan untuk menyiapkan Google Assistant di ponsel Android. Selanjutnya adalah membuat Applet. Di ponsel Android, kita buka *ifttt.com*. Di kanan atas, tap "foto kita > New Applet". Tap "this" dan kita telusuri "Google Assistant" lalu tap "Google Assistant". Kita pilih salah satu "trigger", misalnya adalah "Turn on the lamp". Kolom trigger tersebut harus kita lengkapi dan maksimal tiga kalimat perintah yang ingin kita gunakan untuk memicu tindakan dari Google Assistant. Di kolom terakhir, kita masukkan apa yang kita ingin Google Assistant ucapkan untuk merespon perintah kita tersebut. Untuk saat ini, bahasa yang didukung IFTTT adalah bahasa Inggris dan belum mendukung bahasa Indonesia. Selanjutnya tap "Create trigger". Kemudian kita buat tindakan yang kita ingin Google Assistant lakukan saat kita mengucapkan perintah. Tap "that" lalu kita telusuri "Webhooks". Jika webhooks meminta kita menghubungkan akun, maka ikuti langkah-langkahnya. Kita pilih "Make a web request", lalu kita isi kolom yang wajib diisi. Pada kolom URL diisi dengan url HTTP RESTful API. Kemudian tap "Create action" dan setelah selesai, tap "Finish". Inti membuat Applet ini adalah ketika kita mengucapkan perintah pada Google Assistant, maka akan memicu layanan Webhooks untuk melakukan web request ke HTTP RESTful API. Form Applet Google Assistant dan Webhooks pada IFTTT terdapat pada gambar 4.

- Aplikasi *Google Assistant* di-install di ponsel *Android* dan login menggunakan akun *gmail*.

Gambar 4. Form Applet *Google Assistant* dan *Webhooks* pada *IFTTT*.

2.5. Desain keseluruhan Sistem *Smart Home* berbasis *IoT*

Desain sistem *Smart Home* berbasis *IoT* menggunakan suara pada *Google Assistant* secara keseluruhan terdapat pada gambar 5.



Gambar 5. Desain sistem *Smart Home* berbasis *IoT* menggunakan suara pada *Google Assistant*.

Perintah suara dijalankan dengan menggunakan bahasa Inggris pada aplikasi *Google Assistant* di *android* yang terkoneksi dengan *internet*. *Google Assistant* mengubah perintah suara menjadi teks. Teks tersebut kemudian akan diteruskan dari *Google Assistant* ke *Webhooks* oleh *IFTTT (If This Than That)*. *Webhooks* akan melakukan *request* ke *HTTP RESTful API*. Dengan library *phpMQTT* yang terdapat di *HTTP RESTful API*, perintah di *publish* ke *MQTT Broker*. *ESP32 Dev Kit* sebagai *microcontroller* yang terhubung dengan *internet* menerima perintah dari *MQTT Broker* untuk menyalakan atau mematikan lampu yang ada di rumah melalui *relay board*.

3. HASIL DAN PEMBAHASAN

Pada hasil dan pembahasan ini akan menjelaskan tentang Pengujian Sistem dan Analisa Hasil pada sistem yang telah dibangun.

3.1. Pengujian Sistem

Pada pengujian ini dilakukan perintah suara bahasa Inggris pada aplikasi *Google Assistant* di *android* untuk menyalakan lampu terdapat pada tabel 1.

Tabel 1. Pengujian menyalakan lampu

No	Perintah	Kondisi Lampu
1	Turn on the lamp	Menyala
2	Turn on the lamp please	Menyala
3	Please turn on the lamp	Menyala

Kondisi lampu menyala setelah diperintah untuk menyalakan lampu dengan menggunakan *Google Assistant* terdapat pada gambar 6. *Google Assistant* juga memberi respon “Ok Mr. Ajib, the lamp is on now”.



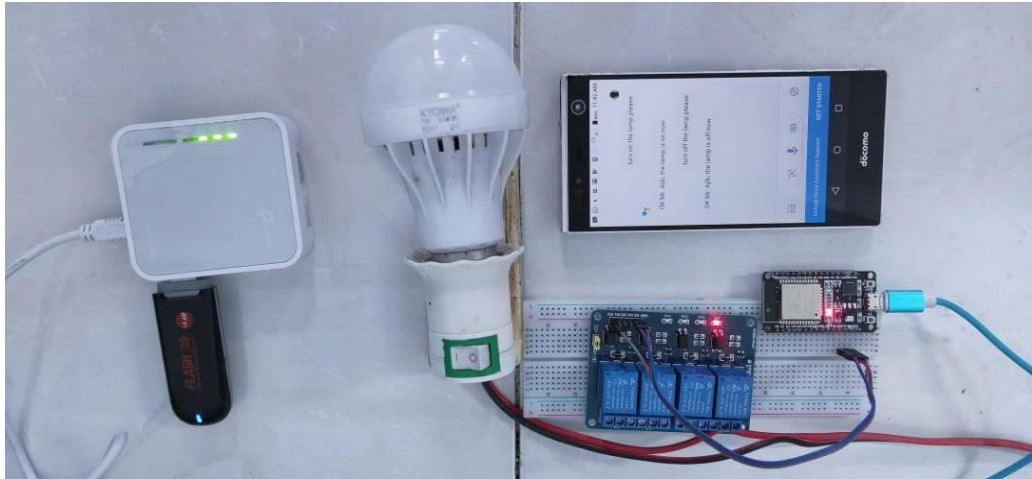
Gambar 6. Menyalakan lampu dengan perintah suara pada *Google Assistant*.

Pada pengujian selanjutnya dilakukan perintah suara bahasa Inggris pada aplikasi *Google Assistant* di *android* untuk mematikan lampu terdapat pada tabel 2.

Tabel 2. Pengujian mematikan lampu

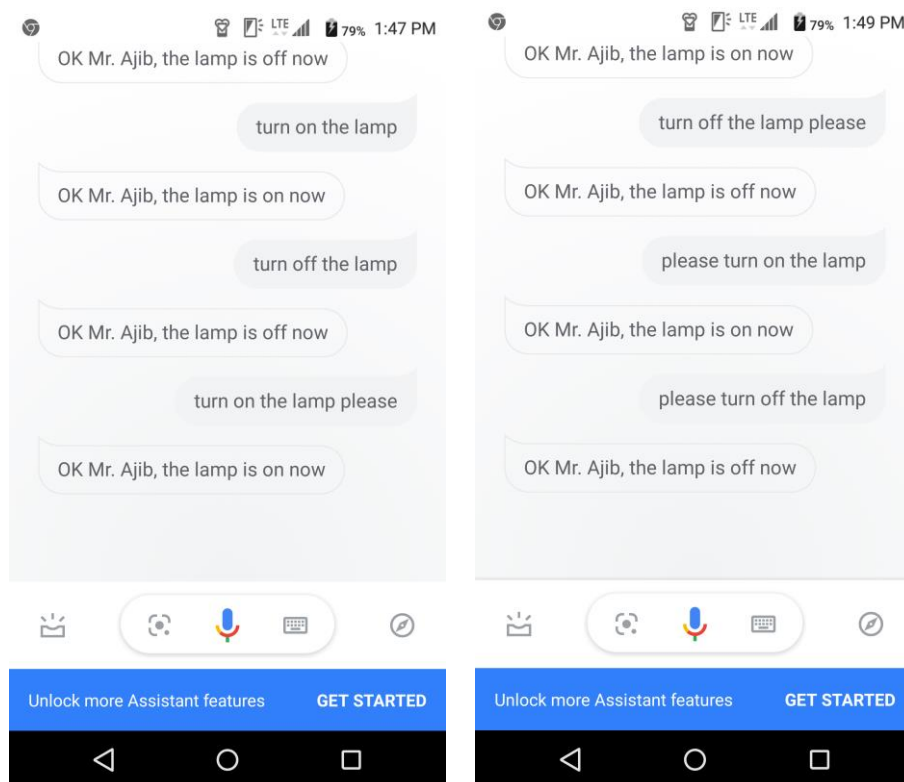
No	Perintah	Kondisi Lampu
1	Turn off the lamp	Mati
2	Turn off the lamp please	Mati
3	Please turn off the lamp	Mati

Kondisi lampu mati setelah diperintah untuk mematikan lampu dengan menggunakan *Google Assistant* terdapat pada gambar 7. *Google Assistant* juga memberi respon “Ok Mr. Ajib, the lamp is off now”.



Gambar 7. Mematikan lampu dengan perintah suara pada *Google Assistant*.

Pada pengujian sistem telah berhasil menyalakan dan mematikan lampu dengan perintah suara pada *Google Assistant* melalui *internet*. *Google Assistant* juga memberi respon ketika lampu dalam keadaan menyala maupun mati seperti pada gambar 8.



Gambar 8. Respon *Google Assistant* ketika lampu dalam keadaan menyala maupun mati.

3.2. Analisa Hasil

Pada penelitian ini, perangkat rumah dapat dikontrol dengan menggunakan perintah suara sehingga lebih interaktif bagi *user* daripada dengan menggunakan tombol *On/Off*. Untuk menggunakan *Google Assistant* harus terkoneksi dengan *internet*. *Google Assistant* mampu mengubah perintah suara menjadi teks. *Google Assistant* juga dapat memberi respon setelah diperintah. Untuk perintah suara maksimal tiga kalimat dalam satu *applet* pada *IFTTT*. Bahasa yang didukung *IFTTT* adalah bahasa Inggris. *IFTTT* mampu mengintegrasikan antara *Google Assistant* dan *Webhooks*. *Webhooks* selanjutnya melakukan *request* ke

HTTP RESTful API. Dengan library *phpMQTT* yang terdapat di *HTTP RESTful API*, perintah di *publish* ke *MQTT Broker*. *MQTT Broker* sebagai server meneruskan perintah ke *microcontroller* yang terhubung dengannya. *ESP32 Dev Kit* sebagai *microcontroller* yang terhubung dengan *MQTT Broker* mampu berkomunikasi dengan menggunakan *wireless*. *ESP32 Dev Kit* menerima perintah dari *MQTT Broker* untuk menyalakan maupun mematikan lampu yang ada di rumah melalui *relay board*. *Relay board* digunakan sebagai saklar untuk memutus atau menyambung arus listrik yang menuju ke lampu sehingga lampu dapat menyala maupun mati.

Dengan dibangunnya sebuah *Smart Home* berbasis *IoT* menggunakan Suara pada *Google Assistant* ini tentunya dapat membantu bagi orang sakit yang berada di kursi roda/tempat tidur atau orang disabilitas tetapi dapat berbicara atau orang lanjut usia yang tidak dapat mencapai saklar mampu mengontrol semua perangkat yang ada di dalam rumah dengan mudah dan lebih interaktif. Selain itu, agar dapat mengontrol perangkat rumah dari jarak yang sangat jauh dan tidak terbatas dengan jarak selama terkoneksi dengan internet, kapanpun dan dimanapun dia berada.

4. KESIMPULAN DAN SARAN

Smart Home berbasis *IoT* menggunakan suara pada *Google Assistant* dapat diimplementasikan dan digunakan untuk membantu orang sakit yang berada di kursi roda/tempat tidur atau orang disabilitas tetapi dapat berbicara atau orang lanjut usia yang tidak dapat mencapai saklar agar mampu menghidupkan/mematikan perangkat rumah dengan mudah. Selain itu, agar dapat mengontrol perangkat rumah dari jarak yang sangat jauh. *Interface* pada aplikasi menggunakan perintah suara sebagai *voice control* pada *Google Assistant* sehingga lebih interaktif bagi *user* daripada dengan menggunakan tombol *On/Off*. Sistem telah berhasil menyalakan dan mematikan perangkat rumah dengan perintah suara menggunakan *Google Assistant* melalui *internet*. *Google Assistant* juga memberi respon ketika perangkat rumah dalam keadaan menyala maupun mati.

Pada penelitian selanjutnya disarankan untuk mengontrol perangkat rumah dengan perintah suara menggunakan bahasa Indonesia atau bahasa daerah dan diproses menggunakan *Natural Language Processing*.

DAFTAR RUJUKAN

- Caranica, A., Cucu, H., Burileanu, C., Portet, F. and Vacher, M. (2017). Speech recognition results for voice-controlled assistive applications. IEEE conference In Speech Technology and Human-Computer Dialogue (SpeD) on 2017, pp. 1-8.
- Dani, A. W., Adriansyah, A. and Hermawan, D. (2016). Perancangan Aplikasi Voice Command Recognition Berbasis Android dan Arduino UNO. Jurnal Teknologi Elektro, Universitas Mercu Buana, Vol 7, No. 1, hal. 11-19.
- IFTTT. (2020, Januari 10). Connect your Google Assistant to Webhooks with IFTTT. IFTTT (If This Than That). (Online). https://ifttt.com/connect/maker_webhooks/google_assistant.
- Khedkar, S. and Malwatkar, G.M. (2016). Using raspberry Pi and GSM survey on home automation. IEEE International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), pp. 758-761.
- Kodali, R. K. and Soratkal, S. (2016). MQTT based home automation system using ESP8266. 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Agra, pp. 1-5.
- Kumar, M. and Shimi, S.L. (2015). Voice Recognition Based Home Automation System for Paralyzed People. International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE), Vol. 4.
- Paul Jasmin, R., Jason, B., Praveen, K., Santhosh, K. (2016). Voice Controlled Home Automation System Using Natural Language Processing (NLP) and Internet Of Things (IoT). International Journal of Control Theory and Applications, vol. 9, issue 40.
- Pavithra, D. and Balakrishnan, R. (2015). IoT based monitoring and control system for home automation. IEEE Global Conference Communication Technologies (GCCT) on 2015, pp 169-173.
- Satria, A., Priadi, M.L., Wulandhari, L.A. and Budiharto, W. (2015). The Framework of Home Remote Automation System Based on Smartphone. International journal of smart home, vol. 9, issue 1, pp.53-60.
- Sen, S., Chakrabarty, S., Toshniwal, R. and Bhaumik, A. (2015). Design of an intelligent voice-controlled home automation system. International Journal of Computer Applications, vol. 121, issue 15.