

# Augmented Reality dalam Penggunaan Alat Rumah Tangga Berbasis *Internet Of Things*

Raja Chairul Jannah Wydmann<sup>1</sup>, Riki Mukhaiyar<sup>2</sup>

<sup>1,2</sup> Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Padang  
Jl. Prof Dr. Hamka Air Tawar, Padang, Indonesia

[chairulraja4@gmail.com](mailto:chairulraja4@gmail.com)<sup>1</sup>, [riki.mukhaiyar@yahoo.co.uk](mailto:riki.mukhaiyar@yahoo.co.uk)<sup>2</sup>

**Abstract**— *The development of this application is based on the minimum level of efficiency in the use of augmented reality technology, especially in the household sector. This is due to a lack of knowledge in this field and which is the basis for building augmented reality applications that are used to manage household appliances that are bridged by the Internet of Things. In the application, augmented reality will be embedded in an Android smartphone and can be controlled using finger gestures on the screen and displayed using the camera. The command will be sent to the Raspberry Pi which acts as the MQTT server and will send it to NodeMCU via the internet to control household appliances. The server is connected to the NodeMCU using the MQTT "mosquitto" protocol in sending commands. As a result, the augmented reality application created using Unity can run and connect all systems into one because of the one protocol that bridges it, namely MQTT (Message Queuing Telemetry Transport).*

**Keywords**— *Message Queuing Telemetry Transport, Augmented Reality, Raspberry Pi, Internet of Things, Unity.*

**Abstrak**— Pengembangan aplikasi ini didasari oleh tingkat efisiensi penggunaan teknologi *augmented reality* yang masih minimum, khususnya dibidang rumah tangga. Hal ini disebabkan kurangnya pengetahuan dibidang tersebut dan menjadi dasar dalam membangun aplikasi *augmented reality* yang digunakan untuk mengatur peralatan rumah tangga yang dijembatani oleh *Internet of Things*. Dalam aplikasinya, *augmented reality* akan ditanamkan didalam *smartphone android* dan dapat dikontrol menggunakan gestur jari tangan pada layar dan ditampilkan menggunakan kamera. Perintah akan dikirimkan ke *Raspberry Pi* yang berperan sebagai *server* MQTT dan akan mengirimkannya ke NodeMCU via *internet* untuk mengontrol alat rumah tangga. *Server* terhubung dengan NodeMCU menggunakan protokol MQTT "*mosquitto*" dalam mengirimkan perintah. Hasilnya aplikasi *augmented reality* yang dibuat menggunakan Unity dapat berjalan dan menghubungkan semua sistem menjadi satu karena adanya satu protokol yang menjembatannya yaitu MQTT (*Message Queuing Telemetry Transport*).

**Kata kunci**— *Message Queuing Telemetry Transport, Augmented Reality, Raspberry Pi, Internet of Things, Unity.*

## I. PENDAHULUAN

Perkembangan zaman menuntut teknologi yang ada ikut berkembang dalam menyesuaikan kebutuhan manusia semakin efisien. Bagitu pula dalam pengendalian alat rumah tangga, banyak sekali kemajuan yang telah diciptakan. Dari pengontrolan rumah tangga via *smartphone*[1] atau berbasis *web*[2], menggunakan komunikasi *Wi-Fi*[3] atau *bluetooth*[4], maupun yang menggunakan *arduino uno*[1] ataupun *raspberry pi*[5]. Namun belum ada antarmuka *augmented reality* yang digunakan dalam sistem yang digunakan. Hal inilah yang mendasari adanya pengembangan aplikasi ini guna meningkatkan efisiensi pengguna dalam mengendalikan alat rumah tangga.

Dalam pengamatan penulis, permasalahan yang terdapat pada pengembangan aplikasi ini adalah kompleksitas sistem yang akan menghubungkan antara mekanik, elektronika dan pembangunan sistem *database* berbasis dengan *web* yang terhubung dengan internet melalui protokol MQTT. Selain itu, pemodelan objek *virtual* yang ditempatkan pada dunia nyata, biaya masing-masing komponen yang mahal serta perancangan replica

rumah tangga menjadi permasalahan dalam pengembangan aplikasi ini.

Penelitian tentang *Internet of Things* (IoT) sudah ada dan banyak dilakukan oleh Somayya Madakam pada tahun 2015 yang berjudul *Internet of Things (IoT): A Literature Review* yang berkesimpulan bahwa IoT telah mambantu membawa perubahan teknologi dalam kehidupan sehari-hari. Dari sektor medis, manufaktur, industri, transportasi, pendidikan, tata kelola, pertambangan, dan lain-lain. Tetapi terdapat kelemahan dalam tata kelola IoT dan tingkat implementasi antara lain tidak ada definisi standar yang berlaku di seluruh dunia, standarisasi universal yang diperlukan di tingkat arsitektur, teknologi berbeda dari *vendor* pengembang sehingga perlu *interoperable* (operasi antar teknologi), dan untuk skala global perlu dibangun protokol standar. Namun pada penelitian Budi Artono pada tahun 2018, beliau memanfaatkan suatu *platform* IoT yaitu *Cayenne* yang sekaligus menjadi server yang mampu menyimpan *project* yang dibuat. Dalam judul penelitiannya yaitu "*Penerapan Internet of Things (IoT) untuk Kontrol Lampu menggunakan Arduino Berbasis Web*" berhasil mengendalikan lampu menjadi hidup dan mati secara otomatis dan termonitor dalam servernya. Maka

dari itu IoT cocok digunakan dalam mengatur perangkat rumah tangga yang dalam hal ini penulis menggunakan lampu dan kamera.

Tujuan yang hendak dicapai dalam penelitian dan pengembangan alat ini adalah menambahkan suatu sistem baru yang terpisah dan saling terhubung menggunakan teknologi IoT yang sangat berguna dalam efisiensi pengontrolannya. Sistem itu adalah *Augmented Reality* dimana data komputer yang bersifat 3D (3 dimensi) digabungkan dengan dunia nyata. Jadi, dengan melakukan *interfacing* untuk menempatkan obyek *virtual* ke dalam dunia nyata melalui modul yang sesuai. Dengan begitu tidak memerlukan layar yang besar.

Dalam hal *monitoring*, tetap menggunakan *web services* yang dapat dipantau melalui 2 cara. Menggunakan *Augmented Reality* dan *Web Services*, tetapi yang lebih ditekankan adalah menggunakan *Augmented Reality*. Perbedaan cara ini bertujuan sebagai efisiensi penampilan informasi. Jika hanya informasi singkat, dapat di-*monitor* melalui *AR*. Tetapi, jika ingin melihat informasi yang kompleks, dapat diakses menggunakan *web services*. Dan dalam menghemat biaya penggunaan *hosting*, dapat membuat *web services* tersendiri menggunakan *mini-computer* semisal Raspberry Pi 3. Tidak hanya *web services*, Raspberry Pi 3 ini akan digunakan untuk membangun *MQTT Broker Server*.

Titik fokus dalam pengontrolan komponen rumah tangganya antara lain kontrol hidup dan mati lampu, dan notifikasi pemindai gerakan pada pintu masuk sebagai penanda ada tamu ataupun penyusup menggunakan kamera. Diharapkan dalam penelitian ini dapat memenuhi kebutuhan praktis dan efisiensi dalam penggunaan peralatan dirumah tangga.

### Microcomputer

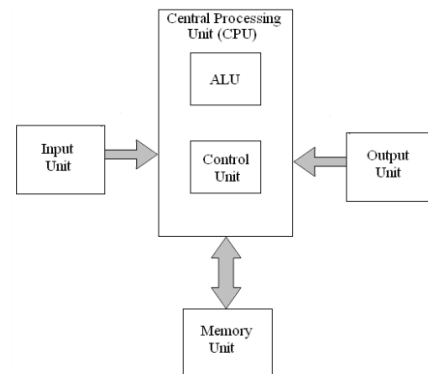
*Microprocessor* adalah sebuah perangkat digital berbentuk *chip* yang dapat mengambil instruksi dari memori, men-*decode* dan mengeksekusi dari hasil *decode* yaitu melakukan operasi aritmatika dan logika tertentu, menerima data dari perangkat *input*, dan mengirim hasil yang prosesnya ke perangkat *output*. Dalam hal ini, jika *microprocessor* yang dihubungkan dengan *memory* dan perangkat *input* dan *output* akan membentuk *microcomputer*.

Terdapat lima blok inti dari *microcomputer*, antara lain :

1. *Input Unit* : Tujuan dari unit ini adalah untuk membaca data ke dalam mesin pemroses.
2. *Memory Unit* : Tujuan utama dari unit ini adalah menyimpan informasi atau data.
3. *Arithmetic and Logical Unit (ALU)* : Tujuan utama unit ini adalah melakukan operasi aritmatika seperti penambahan, pengurangan, penggandaan, pembagian dan operasi logis lainnya pada data.
4. *Control Unit* : Tujuan unit ini adalah mengontrol semua unit lain dan juga mengontrol aliran data dari

satu unit ke unit lain untuk melakukan perhitungan. Unit ini juga dapat mengurutkan operasi dan menginstruksikan semua unit untuk melakukan tugas dalam urutan tertentu dengan bantuan *clock pulse*.

5. *Output Unit* : Tujuan unit ini adalah menampilkan hasil proses dari *Arithmetic and Logical Unit (ALU)* ke *output device* seperti CRT (*Cathode Ray Tubes*), LED (*Light Emitting Diodes*), printer, dan lain-lain.



Gambar 1. Diagram blok *microcomputer*  
(Sumber : Khausik, 2010)

### Raspberry Pi

Adalah sebuah SBC (*single board computer*) yang berukuran kecil yang dikembangkan dengan tujuan untuk mengenalkan sistem ilmu komputer di sekolah-sekolah untuk dipelajari. Raspberry Pi menggunakan rangkaian DDR2-SODIMM yang Kompatibel dengan *System on Modules (SoM)* yang berisi prosesor, memori, eMMC Flash (pada varian *non-Lite*) dan mendukung sirkuit daya[6].



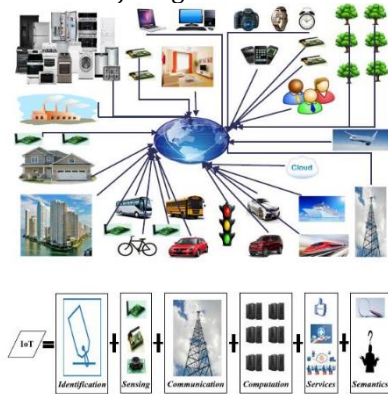
Gambar 2. Raspberry Pi : (a) tampak depan, (b) tampak belakang  
(Sumber : Lab Elektronika, 2018)

### Internet of Things (IoT)

Adalah struktur yang memberikan suatu objek identitas yang terbatas dengan kemampuan dalam memindahkan data atau informasi melalui jaringan tanpa memerlukan hubungan dua arah antara manusia ke manusia (sumber ke tujuan) atau interaksi manusia ke komputer [7].

IoT mempunyai cara kerja yang memanfaatkan suatu sistematisasi program, dimana tiap perintah-perintah dalam program jika dijalankan akan menghasilkan suatu interaksi antar mesin yang telah terhubung secara otomatis tanpa campur tangan manusia dan tanpa

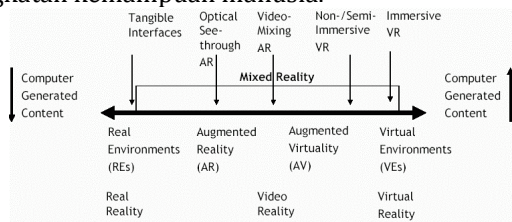
terbatas jarak. Hal ini menggunakan suatu protokol *computer* yang disebut IP (*Internet Protocol*). *Internet Protocol* (IP) adalah suatu identitas didalam jaringan yang membuat suatu perangkat bisa diperintahkan dari perangkat lain dalam jaringan yang sama atau mengirim data atau informasi dengan jarak yang jauh. Selanjutnya, alamat *Internet Protocol* (IP) dalam perangkat tersebut akan dikoneksikan ke jaringan *internet*.



Gambar 3. *Internet of Things* (IoT)  
(Sumber : google.com)

### Augmented Reality

Dalam komputer klasik, lingkungannya akan terisolasi dari lingkungan fisiknya. Komputer klasik membutuhkan daya listrik dan instruksi yang diberikan oleh manusia melalui *keyboard*. Sedangkan manusia dan hewan berkomunikasi dengan lingkungan fisik mereka dengan indera pendengaran, melihat, dan merasakan. Dalam sejarah teknologi, kemampuan manusia dapat diperluas oleh instrumen dan mesin. *Augmented Reality* melanjutkan pengembangan ini dengan *input* sensorik yang dihasilkan komputer seperti suara, video, grafik, dan segala jenis sinyal yang sensitif[8]. Augmentasi juga berarti peningkatan kemampuan manusia.



Gambar 4. *Reality-virtuality Continuum*  
(Sumber : Milgram and Kishino, 1994)

Milgram et al 1994[9], memperkenalkan kontinum realitas-virtualitas (*reality-virtuality continuum*) yang mendefinisikan istilah realitas campuran dan menggambarkan “hubungan” antara dunia nyata dan dunia maya (Gambar 4). Jika dunia nyata berada di salah satu ujung kontinum dan VR (yaitu dunia buatan, *artificial world*) berada di ujung yang lain, maka AR menempati ruang yang lebih dekat dengan dunia nyata. Semakin dekat suatu sistem ke ujung VR, semakin banyak elemen dunia nyata berkurang. Sebagai contoh, sistem AR menggunakan *Optical See-through Displays* yang ditempatkan lebih dekat ke dunia nyata dibandingkan dengan sistem AR dengan

*Video-mixing* dimana lebih mendekati dunia *virtual* (Gambar 4). Jika dunia nyata dapat ditambah dengan objek *virtual*, akan masuk akal jika sebaliknya, untuk berharap bahwa dunia *virtual* dapat ditambah dengan adegan yang nyata (tampilan, objek). Lingkungan seperti itu disebut *augmented virtuality*. Pada kontinum realitas-virtualitas, AR menempati ruang yang lebih dekat dengan lingkungan VR.

### MQTT Broker

MQTT (*Message Queuing Telemetry Transport*) adalah suatu *transport protocol* pada suatu jaringan yang besar *bandwidth header* sebesar 2 byte. MQTT bekerja pada TCP (*Transfer Control Protocol*) dan memastikan pengiriman dari *node* ke *server*. Pertukaran informasi yang berorientasi pesan, menjadikan MQTT sangat ideal diterapkan dalam sistem IoT yang memiliki kemampuan yang tinggi dengan sumber daya yang terbatas. Awalnya MQTT dikembangkan oleh IBM pada tahun 1999 dan telah diakui menjadi standar oleh *Organization for the Advancement of Structured Information Standards* (OASIS)[10].

Basis MQTT adalah *publish* dan *subscribe*. Koneksi MQTT biasanya melibatkan 2 unsur, MQTT *client* dan MQTT *broker* (MQTT *server*). Data yang diangkut dalam protokol MQTT biasa disebut sebagai aplikasi. Dan perangkat atau program apapun yang bertukar pesan aplikasi dan terhubung ke jaringan dikenal sebagai MQTT *client*. MQTT *client* dapat berupa *publisher* atau *subscriber* tergantung pesan aplikasi seperti apa yang dikirimkan. Sedangkan MQTT *broker* adalah sebuah perangkat atau program yang menghubungkan antar MQTT *client* dengan menerima pesan aplikasi dari *subscriber* dan mengirimkannya kembali ke perangkat tujuan sebagai *publisher*. Ketika sebuah MQTT *client* memiliki sebuah data tertentu untuk disiarkan (*broadcast*), data tersebut akan diterbitkan ke MQTT *broker*. MQTT *broker* bertanggung jawab dalam pengumpulan data dan pengorganisasiannya. Pesan yang diterbitkan (*publish*) oleh satu MQTT *client* akan diteruskan kepada MQTT *client* yang memberikan perintah *subscribe* pada MQTT *broker*.

Terdapat beberapa paket kontrol yang terbagi antara *header* tetap, *header* variabel, dan *payload*. Beberapa diantaranya adalah *CONNECT*, *CONNACK*, *PUBLISH*, *PUBACK*, *PUBREC*, *PUBREL*, *SUBSCRIBE*, dan *SUBACK*. Terdapat istilah *topic* dalam MQTT yang merupakan informasi rute data. Setiap *topic* memiliki nama dan level sendiri.

## II. METODE

### A. Analisa Kebutuhan Sistem

#### 1. Software

Perangkat lunak yang dibutuhkan adalah sebagai berikut:

- Raspbian : adalah sistem operasi “resmi” dari Raspberry Pi. Raspbian adalah versi Linux yang dibuat khusus untuk Raspberry Pi. Perangkat ini dilengkapi dengan semua perangkat lunak yang diperlukan untuk setiap tugas dasar dengan komputer.
- Mosquitto : adalah program *open source* MQTT (*Message Queuing Telemetry Transport*) yang dibuat oleh Eclipse Foundation pada Tahun 2019. Kelebihannya adalah programnya yang ringan dijalankan secara terus menerus dengan klien yang terhubung ke server bias mencapai 100.000 klien. Selain itu, Mosquitto memungkinkan adanya jembatan dengan server MQTT lainnya.
- Node-Red : terbentuk pada awal Tahun 2013 yang merupakan proyek oleh Nick O’Leary dan Dave Conway-Jones dari grup *Emerging Technology Services* IBM. Node-Red memvisualisasikan dan memanipulasi pemetaan antara topik MQTT, yang ditampilkan dalam *flow diagram* (aliran diagram) berbasis *web browser*. *Runtime*-nya dibangun diatas *node.js* dan disimpan menggunakan JSON. Hal ini memudahkan pengguna untuk merancang sendiri sistem MQTT-nya.
- Blynk-Server : adalah Blynk Server adalah server Java berbasis *Open-Source* Netty, yang bertanggung jawab untuk meneruskan pesan antara aplikasi seluler Blynk dan berbagai papan mikrokontroler dan SBC (yaitu Arduino, Raspberry Pi, dll.). Terdapat server global blynk yang tersedia dengan mengakses webnya<sup>1</sup>. Dan dikarenakan berbayar, blynk menyediakan opsi untuk membangun sendiri server menggunakan SBC (*Single Boarded Computer*) seperti Raspberry Pi<sup>2</sup>.
- Apache : merupakan *software web server* yang bertanggung jawab pada *request response* HTTP dan *logging* informasi secara detail.
- No-IP<sup>3</sup> : adalah sebuah program DDNS (*Dynamic Domain Name System*) yang bertugas untuk menyediakan sebuah *domain*

*name* bagi server Raspberry Pi. Perbedaannya dengan DNS biasa adalah dimana DDNS akan mencegah kita untuk mengganti IP Publik yang diberikan oleh penyedia internet (ISP) secara berulang-ulang karena sifatnya yang dinamis (selalu berganti secara berkala). ISP yang digunakan oleh penulis adalah IndiHome yang IP Publik akan diganti secara berkala setiap seminggu atau sebulan sekali.

- IDE Arduino : *Software processing* yang digunakan sebagai media pemrograman NodeMCU dan ESP32 Cam yang terintegrasi untuk menulis program, mengkompilasi, dan *men-debug* jika ada kesalahan pemrograman, kemudian *meng-upload*-nya ke papan tersebut. *Processing*-nya menggunakan penggabungan antara bahasa C++ dan Java.
- Visual Studio : adalah program *developer* multifungsi yang akan digunakan untuk mendesain suatu *web server* dari *front-end* sampai dari *back-end*-nya.
- Unity : adalah aplikasi yang digunakan untuk mengembangkan *game multiplatform*. Tetapi dalam waktu dekat ini, terdapat banyak *developer* yang bergerak pada sektor *Virtual Reality* dan *Augmented Reality* yang berfokus menggunakan aplikasi ini dengan platform yang berbeda-beda, dari Android, iOS, HoloLens, Samsung Gear VR, dan lain-lain.
- Mixed Reality Toolkit : adalah sebuah plugin yang dibuat oleh Microsoft untuk membantu *developer* dalam mengembangkan antarmuka *Augmented Reality* agar bisa digunakan dalam kehidupan sehari-hari.
- MQTT Dash : merupakan program penghubung antara *user* sebagai client dengan *user* MQTT. Aplikasi ini dikembangkan oleh Routix Software.
- Blynk : merupakan program yang digunakan sebagai penghubung antara *client blynk* dengan *blynk server*.

#### 2. Hardware

- Raspberry Pi 3 Model B
- ESP32-Cam
- Sensor PIR
- USB to TTL YP-01
- NodeMCU ESP8266 with WiFi
- Relay Module
- Lampu
- Router/Access Point Huawei HG8245H

---

<sup>1</sup> <https://blynk.io/>

<sup>2</sup> <https://github.com/blynkkk/blynk-server>

<sup>3</sup> <https://www.noip.com/>

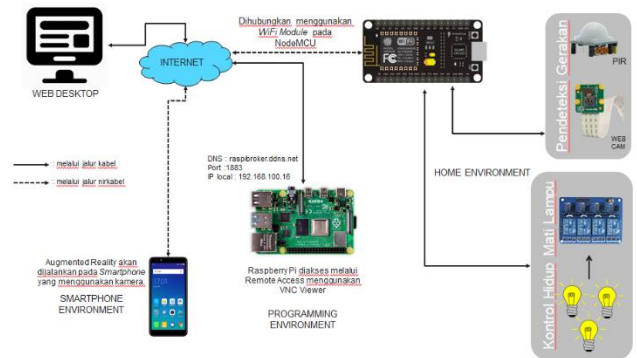
B. Prinsip Kerja dan Blok Diagram

Konsep dari Proyek yang dibuat adalah penggabungan *Home Appliances Control* dengan *Internet of Things* yang dapat dimonitor menggunakan teknologi *Augmented Reality*. Dalam pengontrolan peralatan rumah, sensor yang dibutuhkan antara lain Sensor gerakan PIR yang akan diprogram dan dipasangkan pada ESP32 Cam, sebagai pemindai dari makhluk hidup yang bergerak. Jika PIR sensor ini mendeteksi ada gerakan, maka ESP32 Cam yang ditempatkan akan memotret atau merekam objek tersebut.

Selain itu, kontrol hidup dan mati lampu menggunakan relay elektromagnetik yang dihubungkan ke NodeMCU ESP8266. Mikrokontroler ini akan dikoneksikan ke *internet* guna menerima perintah masuk dari Raspberry Pi 3. Hal yang menghubungkan kedua perangkat ini adalah protokol MQTT. Saat Raspberry Pi 3 mengirim sinyal LOW untuk NodeMCU, maka NodeMCU akan menerima paket data tersebut dan lampu akan menyala karena saklar relay akan berpindah dari *Normally Open* menjadi *Close*. Begitupun saat mengirim sinyal HIGH, lampu akan mati karena saklar akan kembali menjadi *Normally Open*. Hal ini mengikuti indikator LED yang ada pada *relay module*. Pengontrolan ini bersifat *manual* melalui *smartphone AR* yang akan dirancang.

Pengontrolan dan *monitoring* yang dirancang menggunakan 2 sistem, antara lain yang pertama berbasis *Website* yang dibangun didalam Raspberry Pi 3 yang dapat diakses melalui internet pada *web browser*. Pada *website* akan ditampilkan menu-menu untuk mengontrol dan memonitoring peralatan rumah tersebut. Hal ini bisa dilakukan karena *interface* yang tampil di *web browser* menggunakan Node-Red yang dapat mengatur MQTT pada server, serta menampilkannya dalam bentuk web.

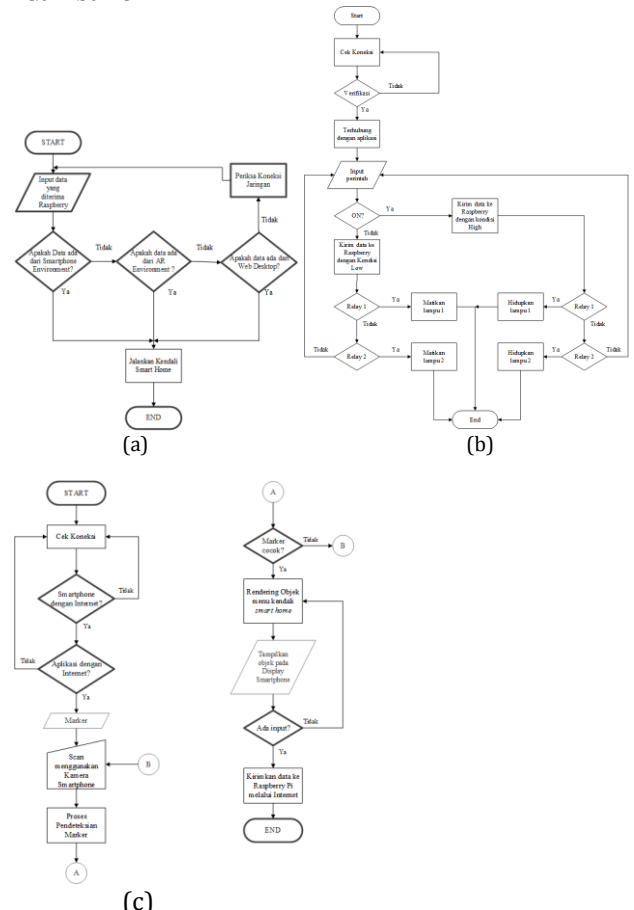
Yang kedua menggunakan aplikasi yang dibangun berbasis android untuk ditempatkan di *smartphone*. *Smartphone* dihubungkan ke *internet*, lalu tersambung pada Raspberry Pi sebagai *Gate* untuk sistem *Home Appliances Control*. Dalam aplikasi tersebut juga akan muncul menu-menu untuk mengontrol dan memonitor peralatan rumah tangga. Aplikasi ini juga dibuat sebagai *interface* dari *augmented reality*. Menggunakan kamera, sistem akan mendeteksi gerakan atau *input* kita pada layar untuk memasukkan perintah ke *server* Raspberry Pi 3.



Gambar 5. Blok diagram sistem

C. Diagram Alir (Flowchart) Sistem

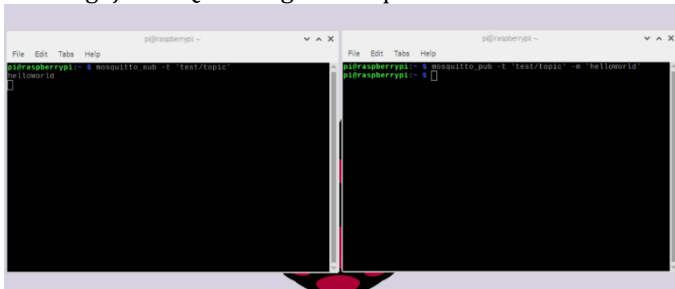
Adalah sebuah jenis diagram yang mewakili algoritma dan proses dan dikemas untuk menampilkan langkah-langkah secara simbolis dan berurutan menggunakan panah. Diagram alir yang dibutuhkan dalam sistem ini ada 4 (empat) dan dapat dilihat pada Gambar 6.



Gambar 6. Diagram Alir : (a) Diagram kerja keseluruhan, (b) Kontrol Hidup dan Mati Lampu, dan (c) Sistem Augmented Reality dalam Smartphone

### III. HASIL DAN PEMBAHASAN

#### A. Pengujian MQTT dengan Mosquitto



Gambar 7. Pengujian Mosquitto *publish* dan *subscribe*

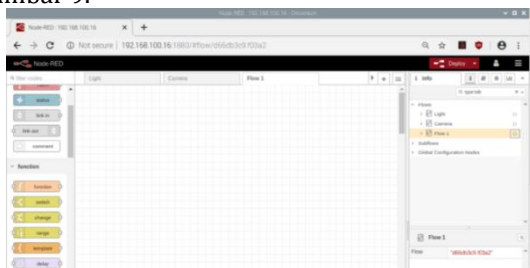
Dalam pengujian menggunakan mosquitto, setelah penginstalannya, buka 2 terminal pada Raspberry Pi. Lalu ketikkan hal ini untuk menguji *publish* dan *subscribe* pada mosquitto. Penjelasan adalah :

- mosquitto\_sub : adalah perintah untuk *subscribe*
- mosquitto\_pub : adalah perintah untuk *publish*
- t 'test/topic' : adalah argumen untuk topik "test/topik"
- m hello world : adalah argumen pesan (*message*) hello world

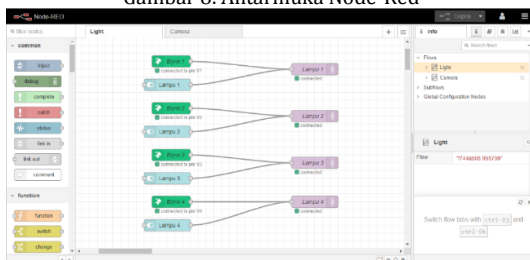
Dalam pengujian berhasil. MQTT server telah berhasil dibangun.

#### B. Pengujian Node-Red

Sesudah penginstalannya, saat menyetikkan alamatnya pada *web browser* (<http://192.168.100.16:1880>) akan langsung ditampilkan oleh Gambar 8. Dan pengaturan aliran diagram pada Node-Red dapat dilihat pada Gambar 9.



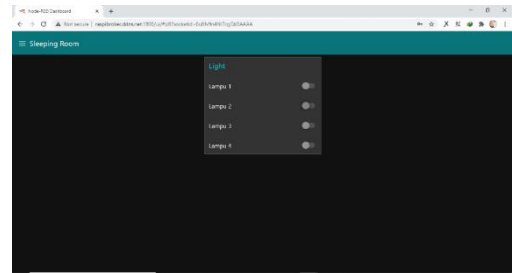
Gambar 8. Antarmuka Node-Red



Gambar 9. Aliran Diagram Node-Red

Setelah pengaturan selesai, klik Deploy pada bagian kanan atas dan Node-Red dapat diuji. Dengan cara pilih Menu pada bagian paling kanan. Lalu pilih View dan klik Dashboard. Pada submenunya, terdapat icon berikut. Klik dan anda akan masuk ke Dashboard yang

menampilkan *switch* yang telah dibuat diatas (Gambar 10).



Gambar 10. Dashboard Node-Red

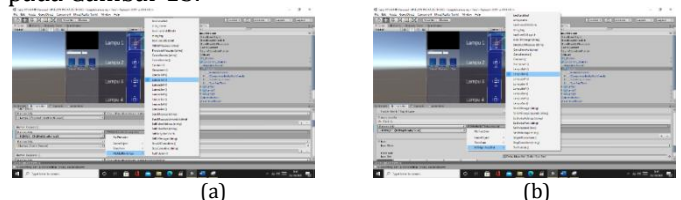
#### C. Pengujian Augmented Reality

Setelah mendaftar pada Website No-IP, nama domain yang dipakai adalah raspibroker.ddns.net. Maka atur dalam pengaturan MQTT pada Unity seperti Gambar 11.

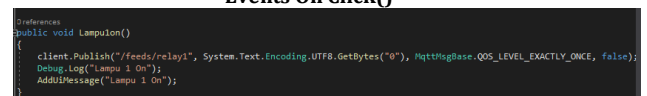


Gambar 11. Pengaturan MQTT pada Unity

Lalu atur setiap tombol MRTK (*Mixed Reality Toolkit*) agar mengarah kepada MQTT. Contohnya adalah pada tombol Lampu 1 ON, maka pilih Lampu1On() seperti Gambar 12. Dan baris kode Lampu1On() dapat dilihat pada Gambar 13.



Gambar 12. Pengaturan Lampu1On() : (a) Button Pressed(), (b) Events On Click()



Gambar 13. Baris kode Lampu1On()

Dan hasil dari pengujian *Augmented Reality* di *Smartphone* dapat dilihat pada Gambar 14 dan rinciannya pada Tabel 1.



Gambar 14. Hasil Pengujian Augmented Reality

Tabel 1. Rincian pengujian

No	Tombol	Status	Output Console
1	Connect	Program akan menyambungkan ke server Raspberry Pi dengan DNS raspibroker.ddns.net	Connected to broker on raspibroker.ddns.net
2	Disconnect	Program akan memutuskan koneksi ke server Raspberry Pi	Connected to broker on raspibroker.ddns.net Disconnected.
3	Clear	Program akan menghapus log pada output console.	Ready..
4	ON pada Lampu 1	Program akan mengirim sinyal ON dengan topic /feeds/relay1 dan lampu menyala.	Connected to broker on raspibroker.ddns.net Lampu 1 On
5	OFF pada Lampu 1	Program akan mengirim sinyal OFF dengan topic /feeds/relay1 dan lampu mati.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off
6	ON pada Lampu 2	Program akan mengirim sinyal ON dengan topic /feeds/relay3 dan lampu menyala.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On
7	OFF pada Lampu 2	Program akan mengirim sinyal OFF dengan topic /feeds/relay3 dan lampu mati.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On Lampu 2 Off
8	ON pada Lampu 3	Program akan mengirim sinyal ON dengan topic /feeds/relay2 dan lampu menyala.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On Lampu 2 Off Lampu 3 On
9	OFF pada Lampu 3	Program akan mengirim sinyal OFF dengan topic /feeds/relay2 dan lampu mati.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On Lampu 2 Off Lampu 3 On Lampu 3 Off
10	ON pada Lampu 4	Program akan mengirim sinyal ON dengan topic /feeds/relay4 dan lampu menyala.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On Lampu 2 Off Lampu 3 On Lampu 3 Off Lampu 4 On

11	OFF pada Lampu 4	Program akan mengirim sinyal OFF dengan topic /feeds/relay4 dan lampu mati.	Connected to broker on raspibroker.ddns.net Lampu 1 On Lampu 1 Off Lampu 2 On Lampu 2 Off Lampu 3 On Lampu 3 Off Lampu 4 On Lampu 4 Off
----	------------------	---	---

#### IV. PENUTUP

Berdasarkan hasil pengujian, dapat disimpulkan Augmented Reality tidak hanya bisa digunakan dalam permainan, tetapi dapat digunakan untuk kehidupan sehari-hari yaitu mengendalikan alat-alat rumah tangga, yang dicontohkan dengan 4 (empat) buah lampu. Hal ini dikarenakan MQTT yang bertugas menghubungkan antara sesama *environment* yang menggunakan protokol universal, dan menggunakan port yang sama yaitu 1883. Dari hasil pengujian juga didapatkan bahwa program yang telah dibuat, dapat dijalankan dimanapun dengan syarat terhubung dengan *internet*, karena DDNS yang dibuat telah terhubung dengan *internet* yang membuatnya mendunia dan dapat diakses dimana saja. Dalam *monitoring*-nya dapat dilakukan menggunakan *web browser* dengan memasukkan alamat DDNS yang sudah dibuat.

Kelebihan dari sistem yang telah dibuat adalah fleksibilitas untuk menambah komponen apapun yang dibutuhkan oleh pengguna dan sifatnya secara gratis. Dikarenakan server yang kita pakai berasal dari Raspberry Pi dan domain yang dipakai adalah domain gratis yang tersedia di *internet*. Namun, bersama kelebihan terdapat kekurangannya. Kekurangan dari sistem ini adalah, tidak semua *smartphone* dipasaran mendukung *services ARCore*. Hanya beberapa *smartphone* tertentu saja yang dapat menggunakannya.

#### REFERENSI

- [1] D. Kurniadi and L. Amelia, "Sistem Kendali Perangkat Elektronik Rumah Berbasis Android dan Arduino," *J. Algoritma*, vol. 15, no. 2, pp. 37-42, 2019, doi: 10.33364/algoritma/v.15-2.37.
- [2] M. Muharam, M. Latif, and M. Saputra, "Sistem Kendali Jarak Jauh Berbasis Web untuk Sistem Rumah Pintar," *J. Nas. Tek. Elektro*, vol. 7, no. 3, p. 203, 2018, doi: 10.25077/jnte.v7n3.502.2018.
- [3] Firdaus, A. A. Nuryono, and A. Sahroni, "Monitoring dan Kendali Lampu Berbasis Jaringan WiFi untuk Mendukung Smart Home," *Semin. Nas. ke - 9 Rekayasa Teknol. Ind. dan Inf.*, vol. 6, no. 2, pp. 51-58, 2015.
- [4] K. A. Pamungkas *et al.*, "Aplikasi Android Dan Mikrokontroler Arduino Pada," *J. Ilm. NERO*, vol. 2, no. 3, pp. 197-203, 2016.
- [5] F. Panduardi and E. S. Haq, "Wireless Smart Home

System Menggunakan Raspberry PI Berbasis Android,” *J. Teknol. Inf. dan Terap.*, vol. 03, no. 01, pp. 320–325, 2016, [Online]. Available: <https://pdfs.semanticscholar.org/402a/ce8d6629211519bc524830408a5c9c825574.pdf>.

- [6] Raspberry Pi Foundation, “Raspberry Pi Hardware.” <https://www.raspberrypi.org/documentation/hardware/raspberrypi/> (accessed Nov. 20, 2019).
- [7] A. W. Burange and H. D. Misalkar, “Review of Internet of Things in development of smart cities with data management & privacy,” *Conf. Proceeding - 2015 Int. Conf. Adv. Comput. Eng. Appl. ICACEA 2015*, pp. 189–195, 2015, doi: 10.1109/ICACEA.2015.7164693.
- [8] R. T. Azuma, “A Survey of Augmented Reality,” *Presence*, vol. 6, no. 4, pp. 355–385, 1997, doi: 10.1016/j.chaos.2009.03.056.
- [9] P. Milgram and F. Kishino, “A Taxonomy of Mixed Reality Visual Displays,” vol. E77-D, no. 12, pp. 1–15, 1994.
- [10] M. Sikandar *et al.*, “MQTT based Home Automation

System Using,” vol. 365565, no. Gcct, pp. 807–813, 2015.

#### ***Biodata Penulis***

**Raja Chairul Jannah Wydmann**, lahir di Lhokseumawe, 17 Desember 1997. Sedang dalam masa studi di Program Studi Teknik Elektro Industri, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Padang.

**Riki Mukhaiyar**, dilahirkan di Padang, 25 Juni 1978. Menyelesaikan S1 pada jurusan Teknik Elektro Universitas Bung Hatta tahun 2000 dan pendidikan Pascasarjana (S2) Magister Teknik bidang Teknik Elektro Institut Teknologi Bandung pada tahun 2003. Dan menyelesaikan pendidikan Doktorat (S3) pada Tahun 2015 di University of Newcastle Upon Tyne jurusan Teknik Elektro. Menyelesaikan pendidikan keprofesian di Institut Teknologi Bandung pada Tahun 2016. Sejak tahun 20xx menjadi staf pengajar tetap di jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Padang.