

PEMBUATAN PROTOTYPE SMART HOME MENGGUNAKAN NODEMCU ESP8266 V3 DAN CHAT BOT PADA SMARTPHONE ANDROID

¹Renaldi Dewangga Sindhu, ²Ilmiyati Sari, ³Dewi Putrie Lestari

¹Fakultas Teknologi Industri Universitas Gunadarma, ^{2,3}Pusat Studi Komputasi Matematika Universitas Gunadarma

Jl. Margonda Raya No. 100, Depok, 16424, Jawa Barat

¹renaldidewanggasindhu@student.gunadarma.ac.id, ²ilmiyati@staff.gunadarma.ac.id,

³dewi_putrie@staff.gunadarma.ac.id

Abstrak

Internet of things adalah sebuah gagasan untuk membuat semua benda di dunia nyata dapat berkomunikasi satu sama lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. Pemanfaatan internet of things dapat digunakan untuk mengendalikan alat elektronik seperti pada pembuatan smart home. Smart home memadukan teknologi informasi dan teknologi komputasi yang diterapkan di dalam rumah ataupun bangunan yang dihuni oleh manusia dengan mengandalkan efisiensi, otomatisasi perangkat, kenyamanan, keamanan, dan penghematan perangkat elektronik rumah. Smart home yang dibuat dalam penelitian ini hanya dapat mengontrol lampu dan kipas angin dari jarak jauh serta dapat mengetahui status ketinggian air di rumah melalui smartphone. Tahap pembuatan aplikasi dimulai dari skema diagram rancangan aplikasi, flowchart, use case diagram smart home, rancangan rangkaian smart home, pembuatan chat bot, dan pengujian. Pembuatan smart home menggunakan NodeMCU ESP8266 V3 dan aplikasi chat bot pada smartphone Android. Setelah dilakukan uji coba, pengontrolan lampu dan kipas angin dari jarak jauh berhasil dilakukan dengan menggunakan smartphone berbasis Android yang terhubung ke internet. Selain itu, status ketinggian air dalam rumahpun dapat diketahui penghuni dimanapun berada.

Kata Kunci: aplikasi chat bot, internet of things, NodeMCU ESP8266, smart home

Abstract

Internet of things is an idea to make all objects in the real world able to communicate with each other as part of an integrated system using the internet network as a connector. The utilization of the internet of things can be used to control electronic devices such as in construction of smart homes. Smart home combines information and computing technology that is applied in homes or buildings inhabited by humans by relying on efficiency, device automation, convenience, security, and savings in home electronic devices. The smart home made in this study can only control lights and fans remotely and can find out the status of the water level at home via smartphone. The application development stage starts from the application design schematic diagram, flowchart, use case diagram of smart home, circuit design of smart home, chat bot creation, and testing. Making a smart home using NodeMCU ESP8266 V3 and chat bot application on Android. After testing, controlling the lights and fans remotely was successfully carried out using an Android-based smartphone connected to the internet. In addition, the status of the water level in the house can be known wherever residents are.

Keywords: chat bot application, internet of things, NodeMCU ESP8266, smart home

PENDAHULUAN

Perkembangan teknologi yang sangat berkembang pesat saat ini telah mampu menjadikan internet sebagai media komunikasi dan kontrol terhadap perangkat dari jarak jauh selama masih terkoneksi dan saling terhubung. *Internet of things* merupakan sebuah pengembangan komunikasi jaringan dari benda yang saling terkait, terhubung satu dengan yang lain lewat komunikasi internet serta dapat saling bertukar data yang kemudian mengubahnya menjadi informasi [1]. *Internet of things* atau dikenal juga dengan singkatan IoT merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus [2-4]. Pada dasarnya perangkat IoT terdiri dari sensor sebagai media pengumpul data, sambungan internet sebagai media komunikasi dan *server* sebagai pengumpul informasi yang diterima sensor dan untuk analisa [5].

Revolusi industri 4.0 telah melahirkan teknologi *internet of things* yang memungkinkan adanya berbagai usaha untuk memberikan kemudahan dan kenyamanan bagi masyarakat dalam kegiatan sehari-hari. Pemanfaatan *internet of things* dapat digunakan untuk mengendalikan alat elektronik seperti pada pembuatan *smart home* [5]. Perangkat *smart home* adalah sebuah perangkat yang memiliki sistem otomatisasi sangat canggih untuk

mengendalikan lampu dan suhu, perangkat multi media untuk memantau dan menghidupkan sistem keamanan yang terhubung dengan pintu atau jendela dan beberapa fungsi yang lainnya [6]. Rumah pintar (*smart home*) merupakan sebuah aplikasi yang dirancang dengan bantuan komputer yang akan memberikan kenyamanan, keamanan, dan penghematan energi yang berlangsung secara otomatis sesuai dengan kendali pengguna dan terprogram melalui komputer pada gedung atau tempat tinggal pengguna [7-10]. Pengguna dapat memantau dan mengendalikan perangkat-perangkat listrik di dalam rumah dari jarak jauh melalui suatu saluran komunikasi seperti melalui jaringan internet, Wi-Fi atau Bluetooth [11].

Sistem *smart home* biasanya terdiri dari perangkat kontrol, monitoring, dan otomatisasi beberapa perangkat atau peralatan rumah yang dapat diakses melalui sebuah komputer [12]. Salah satu perangkat yang terpenting dalam *smart home* adalah perangkat kontrol yang berfungsi pada sistem pengendalian jarak jauh. Sistem kendali dan pemantauan perangkat ruangan pada *smart home* merupakan sebuah bentuk kendali dan dipantau secara otomatis terhadap alat-alat listrik rumah tangga, sistem penerangan atau sistem keamanan rumah yang semuanya mampu dikendalikan dan dipantau secara langsung sesuai keinginan oleh pemilik [13]. Saat ini pengendalian pada *smart home* dapat dilakukan dari jarak jauh dengan

menggunakan *smartphone*. Namun, perangkat *smartphone* harus terhubung dengan internet yang berfungsi sebagai jembatan penghubung antara alat dan sistem kontrol yang digunakan. Beberapa penelitian telah membuat sistem *smart home*. Orpwood, Gibbs, Adlam, Faulkner, dan Meegahawatte tahun 2005 yang memberikan analisis kebutuhan pengguna, rancangan dan evaluasi *smart home* untuk penderita demensia [14]. Sistem ini menggunakan infra red sebagai sensor yang efektifitas hasil luaran sebanding dengan jumlah radiasi infra red yang diterima. Robles dan Kim tahun 2010 melakukan penelitian untuk membuat *smart home* dengan teknologi PCS (*Power Carries System*) [15]. Kekurangan sistem ini yaitu sistem kontrol harus tersambung ke PC berbasis *windows* selama penggunaannya sehingga kurang efisien. Arifiyanto, Syaferi, dan Somantri tahun 2013 telah menghasilkan pemantauan dan pengendalian perangkat listrik yang dapat dilakukan menggunakan *smartphone* melalui jaringan internet [16]. Komputer *server* sebagai pengendali pusat yang menghubungkan komunikasi data antara *smartphone* dengan perangkat mikrokontroler. Peran pengguna atau manusia masih terlihat pada model *smart home* ini.

Penelitian selanjutnya dilakukan oleh Dinata dan Sunanda pada tahun 2015 yang menghasilkan sistem *smart home* dengan *wireless* seperti internet *shield* untuk mengirimkan data energi listrik dari Arduino ke *web server* secara *real time* [17].

Parameter listrik yang dikirimkan antara lain Vrms, Irms, daya nyata, dan faktor daya. Kelebihan dari sistem ini adalah penggunaan energi listrik dapat dipantau melalui suatu *web* secara *real time*. Selain menggunakan komunikasi *wireless* dan jaringan internet untuk mengendalikan perangkat listrik jarak jauh, ada beberapa penelitian yang menggunakan jaringan TCP/IP dan komunikasi Bluetooth pada *smartphone* [18]. Kelemahan model komunikasi ini adalah hanya bersifat lokal saja atau terbatas jaraknya. Selain itu, Ramesh, Kumar, Vamsi dan Akarsh tahun 2017 telah menghasilkan sistem *smart home* untuk mengontrol perangkat berbasis IR melalui Wi-Fi [19].

Berdasarkan penelitian-penelitian tersebut maka dalam penelitian ini dibuat *smart home* yang dapat mengontrol kipas angin dan lampu dari jarak jauh melalui *smartphone* dengan *platform* Android. Selain itu, *smart home* yang dibuat juga memungkinkan penghuni rumah untuk memantau status ketinggian air dalam rumah walaupun penghuni sedang tidak berada di rumah dengan menggunakan sensor *water level*. *Water level* merupakan sensor yang berfungsi untuk mendeteksi ketinggian air dengan *output* analog kemudian diolah menggunakan mikrokontroler.

METODE PENELITIAN

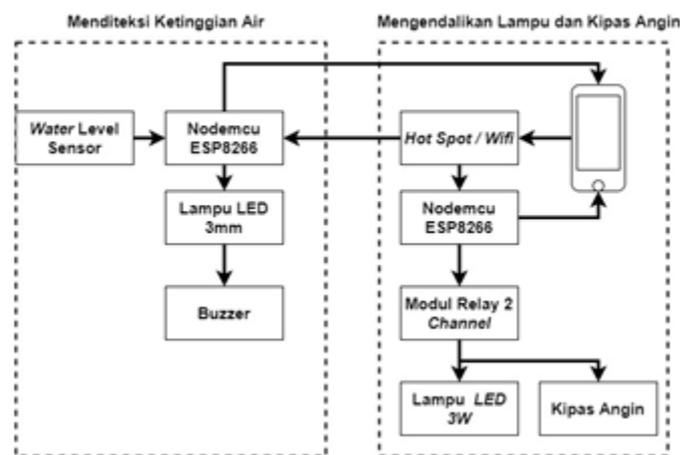
Alat yang diperlukan dalam pembuatan *smart home* pada penelitian ini antara lain

mikrokontroler, modul NodeMCU ESP8266 V3, *project board*, modul *relay 2 channel*, kabel *jumper*, lampu, *water level sensor*, *power bank*, dan *buzzer*. Pengontrolan *smart home* melalui *smartphone* dengan *platform* Android dilengkapi dengan aplikasi Telegram untuk membuat *chat bot*. *Chat bot* berfungsi untuk menuliskan perintah pengguna yang diberikan pada *smart home*. Skema diagram rancangan *smart home* yang dibuat dalam penelitian ini dapat dilihat pada Gambar 1.

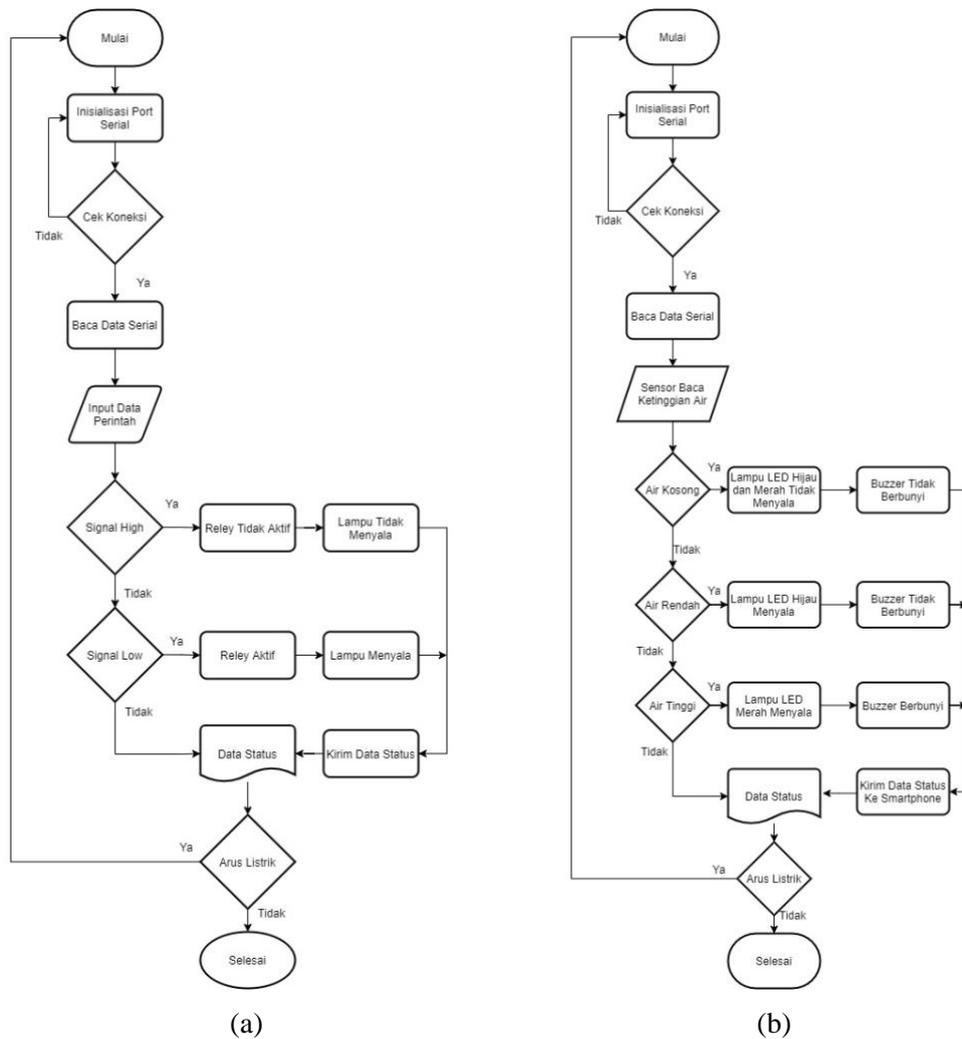
Langkah pertama pada Gambar 1 adalah memasang aplikasi Telegram pada *smartphone* dan membuat *chat bot* dalam aplikasi tersebut. *Chat bot* berfungsi untuk mendapatkan *IP address* yang digunakan pada rangkaian *smarthome*, sehingga *chat bot* dapat terhubung dengan rangkaian *smarthome*. Kemudian merangkai NodeMCU ESP8266 V3 dengan modul *relay 2 channel* dan *water level sensor*. Modul *relay 2 channel* selanjutnya dirangkaikan dengan

lampu LED 3W. Selain sebagai pemisah antara tegangan AC dan DC, *relay* juga berfungsi sebagai komponen utama untuk mematikan dan menghidupkan lampu serta kipas angin. *Water level sensor* berfungsi untuk mendeteksi ketinggian air di rumah. Jika air menyentuh *water level sensor* maka *buzzer* akan berbunyi, kemudian notifikasi status ketinggian air akan dikirim ke *smartphone*.

Flowchart pengendalian lampu dan kipas angin diberikan pada Gambar 2(a) dan *flowchart* deteksi status ketinggian air di rumah diberikan pada Gambar 2(b). Langkah pertama pada Gambar 2(a) dan 2(b) adalah melakukan inisialisasi *port* serial. Selanjutnya periksa koneksi. Jika tidak berhasil terkoneksi, maka lakukan inisialisasi *port* serial kembali. Jika *port* serial berhasil terkoneksi, maka akan dilakukan proses baca data serial. Kemudian pengguna dapat meng-*input* data perintah.



Gambar 1. Skema Diagram Rancangan *Smart Home*



Gambar 2. Alur Kerja Pada Smart Home (a) Flowchart Pengendalian Lampu dan Kipas Angin (b) Flowchart Deteksi Ketinggian Air

Pada Gambar 2(a), setelah perintah di-input oleh pengguna maka akan ada 4 kondisi: (1) jika perintah nyalakan lampu maka proses selanjutnya *relay* aktif sehingga lampu menyala setelah itu mengirim data status kemudian menampilkan data status ke *smartphone* bahwa lampu menyala, (2) jika perintah matikan lampu maka proses selanjutnya *relay* tidak aktif sehingga lampu tidak menyala setelah itu mengirim data status kemudian menampilkan data status pada *smartphone* bahwa lampu sudah mati

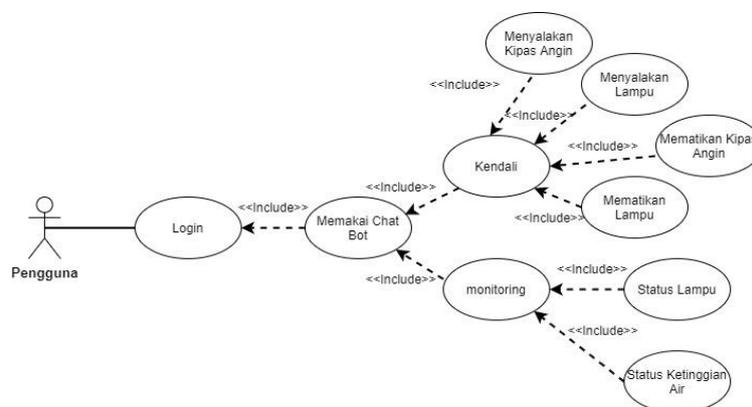
menyala, (3) jika perintah nyalakan kipas maka proses selanjutnya *relay* aktif sehingga kipas menyala setelah itu mengirim data status kemudian menampilkan data status ke *smartphone* bahwa kipas sudah menyala, dan (4) jika perintah matikan kipas maka proses selanjutnya *relay* tidak aktif sehingga kipas tidak menyala setelah itu mengirim data status kemudian menampilkan data status pada *smartphone* bahwa kipas sudah mati. Jika perintah tidak berada pada 4 kondisi yang telah disebutkan, maka akan menampilkan

data status di *smartphone* bahwa perintah yang dimasukan salah. Pada Gambar 2(b), setelah proses baca data serial berhasil dilakukan, langkah selanjutnya adalah melakukan proses sensor baca ketinggian air. Pada proses ini terdapat 3 kondisi yang dimungkinkan sebagai hasil baca *water level sensor*: (1) jika ketinggian air berada di keadaan air kosong maka lampu led berwarna hijau dan merah tidak akan menyala dan *buzzer* tidak berbunyi setelah itu status ketinggian air akan dikirim ke *smartphone*, (2) jika ketinggian air berada di keadaan air aman maka lampu led berwarna hijau akan menyala dan *buzzer* tidak berbunyi setelah itu status ketinggian air akan dikirim ke *smartphone*, dan (3) jika ketinggian air berada di keadaan air berbahaya maka lampu led berwarna merah akan menyala dan *buzzer* akan berbunyi setelah itu status ketinggian air akan dikirim ke *smartphone*. Gambar 3 memperlihatkan interaksi antara pengguna dan *smart home* yang dibuat. Pengguna harus *login* terlebih dahulu pada aplikasi Telegram

di *smartphone* untuk menggunakan *smart home*. Pengguna dapat menggunakan *chat bot* pada aplikasi Telegram setelah melakukan *login*. Ketika pengguna memakai *chat bot*, pengguna dapat mengendalikan lampu dan kipas angin yaitu menyalakan dan mematikan. Selain itu, pengguna dapat melakukan pemantauan terhadap kondisi lampu menyala atau tidak dan kondisi ketinggian air dengan cara mendapatkan pesan balasan dari *chat bot*.

HASIL DAN PEMBAHASAN

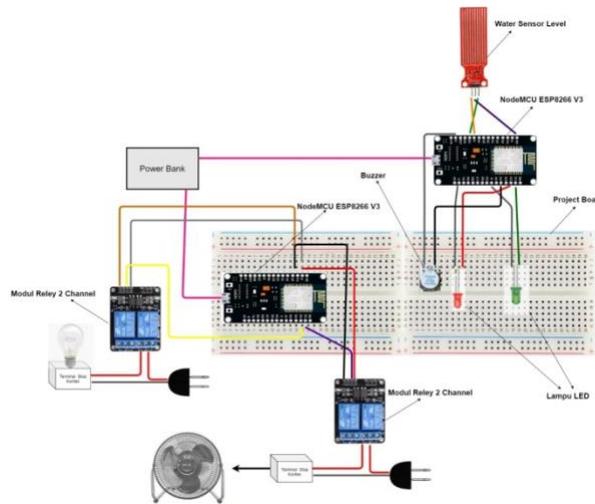
Penelitian ini menghasilkan sebuah sistem *smart home* yang dapat menghidupkan dan mematikan lampu dan kipas angin serta mendeteksi status ketinggian air melalui *smartphone*. Secara garis besar terdapat tiga langkah dalam pembuatan *smart home* pada penelitian ini, yaitu pembuatan rangkaian perangkat keras *smart home*, pembuatan *chat bot*, dan pengujian.



Gambar 3. Use Case Diagram Smart Home

Rancangan rangkaian *smart home* diberikan oleh Gambar 4(a) dan hasil rancangan rangkaian *smart home* ini dapat dilihat pada Gambar 4(b). Ada 10 tahap yang dibutuhkan untuk menghasilkan rangkaian *smart home* pada Gambar 4(b), antara lain:

1. Hubungkan pin VIN pada NodeMCU ESP8266 ke pin VCC modul *relay* menggunakan kabel *jumper*.
2. Hubungkan pin G pada NodeMCU ESP8266 ke pin GND modul *relay* menggunakan kabel *jumper*.
3. Hubungkan pin D2 pada NodeMCU ESP8266 ke pin IN 1 modul *relay* menggunakan kabel *jumper* yang berfungsi untuk menyalakan lampu.
4. Hubungkan pin D1 pada NodeMCU ESP8266 ke pin IN 2 modul *relay* menggunakan kabel *jumper* yang berfungsi untuk menyalakan kipas.
5. Hubungkan pin NodeMCU ESP8266 ke pin modul *relay*, kemudian masukan kabel lampu ke *port* NO 1 dan COM 1 untuk menyalakan lampu dan masukan kabel ke *port* NO 2 dan COM 2.
6. Hubungkan (kutub +) ke pin D1 pada NodeMCU ESP8266 dan (kutub -) ke pin G pada NodeMCU ESP8266 untuk menyalakan lampu LED berwarna hijau yang digunakan sebagai indikator deteksi ketinggian air aman.
7. Hubungkan (kutub +) ke pin D2 pada NodeMCU ESP8266 dan (kutub -) ke pin G pada NodeMCU ESP8266 untuk menyalakan lampu LED berwarna kuning yang digunakan sebagai indikator deteksi ketinggian air rendah.
8. Hubungkan (kutub +) ke pin D3 pada NodeMCU ESP8266 dan (kutub -) ke pin G pada NodeMCU ESP8266 untuk menyalakan lampu LED berwarna merah yang digunakan sebagai indikator deteksi ketinggian air tinggi.
9. Hubungkan *buzzer* (kutub +) ke pin D4 NodeMCU ESP8266, lalu hubungkan (kutub -) ke pin G pada NodeMCU ESP8266.
10. Hubungkan (kutub +) *water level sensor* ke pin tegangan 3V pada NodeMCU ESP8266 untuk menggunakan *water level sensor*, kemudian sambungkan (kutub -) dengan G lalu hubungkan pin S dengan sinyal analog yaitu A0 pada NodeMCU ESP8266.



(a)



(b)

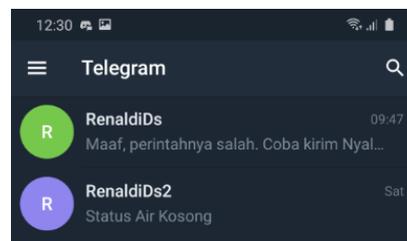
Gambar 4. Sistem *Smart Home* (a) Rancangan Rangkaian *Smart Home* (b) Hasil Rangkaian *Smart Home*

Langkah selanjutnya adalah pembuatan *chat bot* pada aplikasi Telegram. Langkah-langkah pembuatan *chat bot* ini sebagai berikut:

1. Buka perangkat lunak Telegram.
2. Ketik *botfather* pada kontak pencarian sehingga muncul banyak kontak bertuliskan *botfather*.
3. Pilih kontak paling atas sehingga muncul halaman baru.
4. Pilih *start* pada halaman tersebut.
5. Ketik perintah */newbot* sehingga *botfather* mengirim pesan balasan untuk memulai pembuatan *bot*.
6. Ketik nama *bot* yang akan dibuat dengan format penulisan nama *bot* yaitu *namabot_bot*.
7. Proses pembuatan *chat bot* berhasil dilakukan jika mendapatkan API token yang digunakan untuk mengakses API *bot* dan untuk mengontrol NodeMCU ESP8266.

Chat bot pada aplikasi Telegram dapat digunakan jika *smartphone* terhubung ke jaringan internet. *Chat bot* berfungsi untuk menuliskan perintah dari pengguna kepada *smart home*. Pada Gambar 5 terdapat *chat bot* RenaldiDs yang digunakan untuk mengirim pesan atau perintah menyalakan atau mematikan lampu maupun kipas, sedangkan *chat bot* RenaldiDs2 digunakan untuk mendeteksi status ketinggian air. Pada langkah terakhir dilakukan proses pengujian. Pengujian dilakukan pada *smartphone* dan *chat bot*. Hasil pengujian ini dapat dilihat pada Gambar 6, Gambar 7, dan Gambar 8. Pada Gambar 6, ketika pengguna menuliskan perintah “Nyalakan Lampu”, maka perintah akan diterima oleh NodeMCU ESP8266 V3. Kemudian *relay* akan aktif dan lampu akan

menyala. Selanjutnya akan muncul pesan balasan pada *chat bot* bahwa “Lampu Menyala”. Jika pengguna menuliskan perintah “Nyalakan Kipas”, maka perintah akan diterima oleh NodeMCU ESP8266 V3. Kemudian *relay* akan aktif dan kipas akan menyala. Selanjutnya akan muncul pesan balasan pada *chat bot* bahwa “Kipas Sudah Menyala”. Gambar 7 adalah hasil yang terjadi pada *smart home* jika pengguna memberikan perintah menyalakan lampu dan kipas angin. Hal yang sama juga dapat dilakukan untuk mematikan lampu atau kipas angin. Ketika perintah yang dikirim salah maka *chat bot* akan mengirim pesan bahwa “Maaf, perintahnya salah. Coba kirim Nyalakan Lampu/Nyalakan Kipas atau Matikan Lampu/Matikan Kipas”.



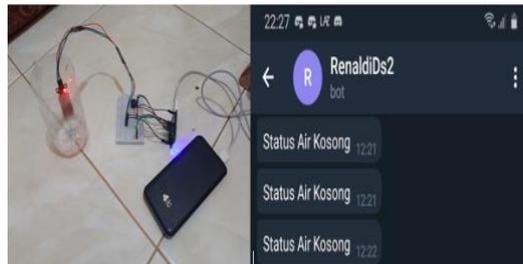
Gambar 5. Tampilan Awal *Chat Bot*



Gambar 6. Tampilan Perintah Pada *Chat Bot*



Gambar 7. Rangkaian *Smart Home* Jika Diberikan Perintah Hidupkan Lampu dan Kipas Angin



Gambar 8. Keadaan *Smart Home* dan Tampilan *Chat Bot* Saat Status Air Kosong



Gambar 9. Keadaan *Smart Home* dan Tampilan *Chat Bot* Saat Status Air Berbahaya

Gambar 8 memperlihatkan keadaan *smart home* dan tampilan *chat bot* saat status air kosong. Ketika ketinggian air belum menyentuh *water level sensor*, maka sensor mendeteksi bahwa status air kosong yang ditampilkan pada serial monitor pada Arduino IDE. Setelah 30 detik akan terkirim pesan pada *chat bot* bahwa status air kosong dan di indikator lampu LED tidak menyala serta

buzzer tidak berbunyi. Pada Gambar 9, ketika ketinggian air sudah menyentuh *water level sensor* dengan kondisi ketinggian air tinggi, maka sensor mendeteksi bahwa status air tinggi yang ditampilkan pada serial monitor pada Arduino IDE. Setelah 30 detik akan terkirim pesan pada *chat bot* bahwa status air berbahaya dan indikator lampu LED menyala dengan warna merah serta *buzzer* berbunyi.

```

unsigned long waktusekarang kirim=millis();
if ( waktusekarang kirim-
waktusebelum kirim>=30000){
    digitalWrite(Buzzer,LOW);
robot.sendMessage(id, "Status Air
Berbahaya");
    Serial.println("Pesan Terkirim");
    waktusebelum kirim=millis();
}
}
delay(30000);
}

```

Gambar 10. Pengaturan Pengiriman Pesan Status Ketinggian Air

Gambar 10 memperlihatkan blok program yang berfungsi untuk mengirim pesan status ketinggian air ke *smartphone* berdasarkan hasil deteksi dari *water level sensor*. Pada bagian waktu kirim pesan diberi *input 30000 millisecond*, dimana jika diubah ke dalam *second* akan menjadi 30 *second* atau 30 detik. Hal ini menandakan bahwa pesan status ketinggian air baru masuk ke *smartphone* setelah 30 detik sensor mendeteksi adanya air atau dapat dikatakan bahwa proses pengiriman pesan status ketinggian air ke *smartphone* mengalami *delay* 30 detik.

KESIMPULAN DAN SARAN

Berdasarkan hasil dari perancangan dan pengujian penerapan *internet of things* dalam pembuatan *smart home* menggunakan NodeMCU ESP8266 V3 dan *chat bot* pada *smartphone* Android dapat diambil

kesimpulan yaitu *smarhome* yang sudah dibuat dapat digunakan untuk mengendalikan lampu, kipas angin, dan mendeteksi status ketinggian air di rumah sesuai dengan yang diharapkan. Perangkat NodeMCU ESP8266 V3 merupakan sebuah mikrokontroler yang terjangkau tetapi benar-benar baik digunakan secara *standalone* (berdiri sendiri).

Penggunaan mikrokontroler NodeMCU ESP8266 V3 ini sangat bergantung dengan keadaan sinyal jaringan internet yang diterima, sehingga menyebabkan cepat atau lambatnya proses pengiriman perintah dari *smartphone* yang menggunakan *chat bot* dalam mematikan atau menghidupkan lampu dan kipas angin. Pada pendeteksian status ketinggian air, *water level sensor* hanya dapat membaca ketinggian air berdasarkan ketinggian alat tersebut. *Water level sensor* ini hanya bekerja ketika air sudah mengenai lempengan yang bergaris-garis pada sensor. Ketika *water level sensor* membaca status

ketinggian air seperti status air kosong, status air aman atau status air berbahaya, maka status ketinggian air tersebut langsung dikirim ke *smartphone* dengan proses pengiriman pesan mengalami *delay* 30 detik.

Sistem *smart home* dan *chat bot* pada penelitian ini selanjutnya dapat dikembangkan dengan memberikan status ketinggian air yang lebih lengkap serta dapat digunakan untuk alat elektronik yang memiliki daya listrik yang lebih besar. Tampilan rangkaian *smart home* yang dibuat pada penelitian ini juga masih sangat sederhana sehingga masih perlu dikembangkan.

DAFTAR PUSTAKA

- [1] B. Artono and F. Susanto, "Wireless smart home system menggunakan internet of things," *Jurnal Teknologi Informatika dan Terapan*, vol. 05, no. 01, pp. 17-24, 2017.
- [2] A. Arafat, "Sistem pengamanan pintu rumah berbasis Internet Of Things (IoT) dengan ESP8266," *Jurnal Ilmiah Technologia*, vol. 7, no. 4, pp. 262-268, 2016.
- [3] Y. Efendi, "Internet of Things (IOT) sistem pengendalian lampu menggunakan Raspberry PI berbasis mobile," *Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al Asyariah Mandar*, vol. 4, no. 1, pp. 19-26, 2018.
- [4] A. Kusumaningrum, A. Pujiastuti, and M. Zeny, "Pemanfaatan internet of things pada kendali lampu," *Compiler*, vol. 6, no. 1, pp. 53-59, 2017.
- [5] R. Y. Endra, A. Cucus, and M. B. Syahputra, *Smart room menggunakan internet of things untuk efisiensi biaya dan keamanan ruangan*. Bandar Lampung: Anugrah Utama Raharja, 2019.
- [6] D. Bregman, R. Blvd, and R. Lezion, "Smart home intelligence -the ehome that learns," *Int. J. Smart Home Smart*, vol. 4, no. 4, pp. 35-46, 2010.
- [7] F. Masykur and F. Prasetyowati, "Aplikasi rumah pintar (smart home) pengendali peralatan elektronik rumah tangga berbasis web," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 3, no. 1, pp. 51-58, 2016.
- [8] H. Setiadi and Munadi, "Desain model smarthome system berbasis mikrokontroler ATmega8535," *Jurnal Teknik Mesin*, vol. 3, no. 2, pp. 138-142, 2015.
- [9] F. G. Aditya, Hafidudin, and A. G. Permana, "Analisis dan perancangan prototype smart home dengan sistem client server berbasis platform Android melalui komunikasi wireless," *eProceedings of Engineering*, vol. 2, no. 2, pp. 3070-3077, 2015.
- [10] S. Kumar, "Ubiquitous smart home system using android application,"

- IJCNC, vol. 6, no. 1, pp. 33-43, 2014.
- [11] D. Kurnianto, A. M. Hadi, and E. Wahyudi, "Perancangan sistem kendali otomatis pada *smart home* menggunakan modul Arduino Uno," *Jurnal Nasional Teknik Elektro*, vol. 5, no. 2, pp. 260-270, 2016.
- [12] T. F. Yurmama and N. Azman, "Perancangan *software* aplikasi *pervasive smart home*," *Seminar Nasional Aplikasi Teknologi Informasi*, 2009.
- [13] F. Z. Rachman, "Smart home berbasis IOT," *Seminar Nasional Inovasi Teknologi Terapan*, pp. 369-374, 2017.
- [14] R. Orpwood, C. Gibbs, T. Adlam, R. Faulkner, and D. Meegahawatte, "The design of smart homes for people with dementia—user-interface aspects," *Universal Access in The Information Society*, vol. 4, no. 2, pp. 156-164, 2005.
- [15] J. R. Rosslin and K. Tai-hoon, "Applications, systems and methods in smart home technology: a review," *International Journal of Advanced Science and Technology*, vol. 15, pp. 37-47, 2010.
- [16] F. Arifiyanto, W. A. Syafei, and M. Somantri, "Perancangan *prototype web-based online smart home controlled by smartphone*," *TRANSIENT*, vol. 2, no. 4, 2013.
- [17] I. Dinata and W. Sunanda, "Implementasi *wireless* monitoring energi listrik berbasis *web database*," *JNTE*, vol. 4, no. 1, pp. 83-88, 2015.
- [18] A. S. Rafika, M. S. H. Putra, and W. Larasati, "Smart home automatic menggunakan media Bluetooth berbasis mikrokontroler ATmega328," *CCIT*, vol. 8, no. 3, pp. 215-222, 2015.
- [19] N. V. K. Ramesh, S. T. Kumar, V. Vamsi, and S. Akarsh, "Wi-Fi controlled universal remote using ESP8266," *ARPJ Journal of Engineering and Applied Sciences*, vol. 12, pp. 7233-7238, 2017.