Rancang Bangun Smart Home berbasis IoT Menggunakan Telegram Messenger Bot Dan NodeMCU ESP 32

Smart Home Design based on IoT Using Telegram Messenger Bot And NodeMCU ESP 32

**Ali Nur Fathoni 1\*, Khusnul Khotimah 2**

1 Departemen Teknik Elektro dan Teknologi Informasi, Fakultas Teknik, Universitas Gadjah Mada

Bulaksumur, Caturtunggal, Depok, Sleman DIY 6491936

2 Departemen Pascasarjana, Universitas Negeri Surabaya

Lidah Wetan, Lakarsantri, Surabaya 99424002

[alinur2020@mail.ugm.ac.id](mailto:alinur2020@mail.ugm.ac.id)1\*, khusnul.19006@mhs.unesa.ac.id2

***Abstrak –*** *Sistem smart home banyak digunakan pada saat ini dikarenakan dapat memberikan keaamanan dan kenyamanan. Sistem ini bekerja secara otomatis pada sebuah rumah dan dapat menghemat energi sehingga mengurangi penggunaan daya listrik. Pada umumnya sebagian besar sistem kontrol peralatan listrik rumahan masih bersifat manual yaitu dengan menghubungkan peralatan listrik pada stop kontak atau menekan saklar secara langsung. Penerapan sistem kendali manual seringkali membuat pengguna lupa mematikan saklar atau peralatan elektronik lainya saat keluar rumah dikarenakan pengguna harus mematikannya secara manual atau terkadang dibiarkan menyala hingga pemilik rumah. Sistem kendali jarak jauh sangat diperlukan untuk mengatasi hal ini. Penggunaan ESP32 NodeMCU banyak dipakai untuk sistem pengendalian jarak jauh dikarenakan relatif terjangkau dan mudah diprogram. Selain diprogram menggunakan Arduino IDE modul NodeMCU ESP32 juga memungkinkan diprogram menggunakan Micropython. Micropython merupakan sub bagian bahasa pemrograman python yang digunakan untuk pemrograman mikrokontroler. Penggunaan media sosial untuk berkomunikasi sudah menjadi hal lumrah dewasa ini, selain itu media sosial messenger seperti aplikasi Telegram juga dapat digunakan sebagai kontrol jarak jauh. Pada penelitian ini prototipe sistem smart home dibuat dengan memanfaat fitur chat pada Telegram yang dipakai untuk mengontrol relay. Sistem bot pada aplikasi Telegram digunakan sebagai pengirim dan penerima pesan untuk kontrol otomatis sehingga pengguna tidak perlu mengontrol secara manual. Berdasarkan hasil pengujian menunjukkan bahwa hasil uji kelayakan mendapatkan hasil persentase nilai rata-rata keseluruhan adalah 84.37% yang meliputi aspek desain alat, aspek kinerja alat, aspek kemudahan pengoperasian dan implementasi. Penilaian tersebut dilakukan oleh ahli menunjukkan bahwa prototipe ini sudah cukup baik dan layak untuk diterapkan pada sebuah rumah.*

***Kata Kunci****: Bot Telegram, Internet of Things, Smart Home, Sistem Kontrol, Micropython.*

***Abstract –*** *Smart home systems are widely used today because they can provide security and comfort. This system works automatically in a house and can save energy thereby reducing the use of electrical power. In general, most home electrical equipment control systems are still manual, namely by connecting electrical equipment to a socket or pressing a switch directly. The application of a manual control system often makes users forget to turn off switches or other electronic equipment when leaving the house because users have to turn it off manually or sometimes leave it on until the home owner. Remote control system is needed to overcome this. The use of ESP32 NodeMCU is widely used for remote control systems because it is relatively affordable and easy to program. In addition to being programmed using the Arduino IDE, the NodeMCU ESP32 module also allows programming using Micropython. Micropython is a sub-section of the python programming language used for microcontroller programming. The use of social media to communicate has become commonplace nowadays, besides that, social media messengers such as Telegram can also be used as remote controls. In this research a smart home system prototype was made by utilizing the chat feature on Telegram which is used to control the relay. The bot system on Telegram messenger is used as a sender and recipient of messages for automatic control so that users do not need to control manually. Based on the test results show that the results of the feasibility test get the results of the percentage of the overall average value is 84.37% which includes aspects of tool design, aspects of tool performance, aspects of ease of operation and implementation. The assessment carried out by the expert showed that this prototype was good enough and feasible to be applied to a house.*

***Keywords****: Telegram Bot, Internet of Things, Smart Home, Control System, Micropython.*

**1. Pendahuluan**

Internet telah menjadi kebutuhan mendasar pada saat ini dikarenakan internet dapat dimanfaatkan untuk banyak keperluan. Pada mulanya internet dipakai untuk keperluan komunikasi jarak jauh baik itu *chatting* dan ataupun sekadar *browsing.* Dewasa ini, perkembangan teknologi semakin berkembang membuat internet tidak hanya sebatas sebagai media berkomunikasi namun juga dapat digunakan sebagai sistem kontrol jarak jauh [1]. Penggunaan internet saat ini membuat sistem automasi pada sistem kontrol menjadi lebih efisien dimana sebelumnya sistem kontrol masih dilakukan secara manual melalui banyak kabel. Pada penerapannya, kebanyakan orang seringkali lupa untuk mematikan peralatan listrik yang digunakan ketika sedang berada di luar ruangan sehingga harus kembali untuk memastikan peralatan listrik telah dimatikan setelah selesai digunakan atau membiarkannya tetap menyala sampai kembali ke rumah. Sistem kontrol manual masih dirasa kurang efisien baik dari segi waktu ataupun daya listrik yang dipakai karena apabila peralatan listrik dinyalakan terus menerus mengakibatkan pemakaian daya listrik meningkat. Penerapan sistem kontrol jarak jauh sangat dibutuhkan masyarakat dalam menunjang kehidupan. Hal ini membuat membuat sistem *smart home* banyak digunakan saat ini.

Sistem *Internet of Things* yang diterapkan suatu rumahadalah menjadikan perangkat dapat tersinkronisasi dengan sebuah modul melalui internet, sehingga memungkinkan penghuninya untuk mengontrol beragam peralatan listrik seperti pencahayaan dan sistem keamanan rumah dari jarak jauh [2]. Pembuatan sistem tersebut harus dirancang agar fleksibel dan mudah digunakan. Saat ini sistem *smart home* sudah banyak tersedia di pasaran mulai dari sistem yang sudah terintegrasi dengan internet hingga yang memanfaatkan teknologi koneksi *bluetooth, wi-fi* dan *cloud* [3]. Sistem *smart home* juga bisa dibuat dengan memanfaatkan teknologi biometrik manusia seperti suara manusia sebagai proses verifikasi untuk menjalankan perintah sesuai yang diinginkan [4]. Penerapan sistem *smart home* dapat diterapkan dengan menggunakan beberapa modul sistem kontrol seperti Arduino, NodeMCU, Raspberry, PLC dan modul lainya serta dapat menggabungkan beberapa sensor tambahanuntuk menunjang kebutuhan sistem sesuai dengan kebutuhan. Penelitian [5] menambahkan beberapa sensor tambahan seperti sensor accelerometer, sensor api dan sensor magnetik untuk proses monitoring dan controling pada sebuah industri kecil rumah tangga, sensor tersebut berfungsi untuk mengetahui status alat listrik yang sedang digunakan. Penelitian [6] menggunakan PLC pada sistem *water level control* untuk mengetahui status ketinggian level air pada sebuah wadah sehingga dapat dimonitor secara jarak jauh. Penelitian [7] menggunakan media sosial Whatsapp untuk membuat sebuah *smart garage,* penelitian ini menggunakan modul Arduino Uno R3 dan Raspberry Pi utuk berkomunikasi secara nirkabel dan proses kontrol sistem garasi dilakukan dengan menggunakan pesan pada aplikasi Whatsapp. Sistem komunikasi menggunakan media sosial seperti Whatsapp ataupun Telegram tidak bisa jauh dari penggunaan *bot.*

*Bot* internet merupakan suatu aplikasi yang menjalankan semua perintah secara otomatis melalui internet. Pada aplikasi Telegram tersedia 2 metode API (*Aplication Programing Interface*) yang dapat digunakan yaitu mode IM (*Instant Messaging*) yang *source code-* nyadisediakan oleh aplikasi sehingga pengguna dapat mengembangkan aplikasi sesuai keinginan. Sedangkan mode API yang kedua adalah Telegram Bot yang memungkinkan semua pengguna dapat membuat bot yang secara otomatis membalas pesan ketika seseorang mengirim pesan [8]. dikembangkan secara bebas oleh siapa saja yang memiliki hak akses kepada bot tersebut. Penelitian ini bertujuan untuk membuat prototipe *smart home* berbasis IoT menggunakan *bot* Telegram dan modul NodeMCU ESP 32. Alat ini berupa saklar otomatis yang dapat dikendalikan jarak jauh dengan memanfaatkan fitur *chat* pada *bot* Telegram. Modul ini diintegrasikan melalui jaringan wifi yang tersedia pada sebuah rumah untuk mendapatkan akses internet. Alat ini dapat diterapkan pada peralatan rumah tangga seperti lampu, kipas angin, AC atau peralatan elektronik lainya agar dapat dikontrol melalui aplikasi Telegram. Setelah selesai dibuat kemudian sistem ini diuji untuk mengetahui kinerja alat yang dibuat serta pengujian tingkat kelayakan alat.

**2. Metode Penelitian**

Metode yang diterapkan pada penelitian ini yaitu metode *Research and Development*. Metode ini digunakan untuk mengembangkan suatu produk dalam menghasilkan produk baru dan menguji keefektifan produk tersebut [9]. Metode ini bersifat longitudinal dan bertahap serta dapat dikembangkan lagi sesuai dengan tujuan dari penelitian. Langkah yang dilakukan pada penelitian ini meliputi proses penggalian potensi dari suatu masalah, proses pengumpulan data, desain sistem, validasi desain, pembuatan, uji coba sistem dan hasil pembuatan prototipe *smart home* berbasis IoT. Alur penelitian ini ditunjukan pada Gambar 1.



Gambar 1. Metode penelitian *Research and Development*

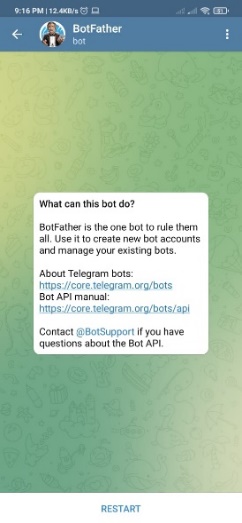
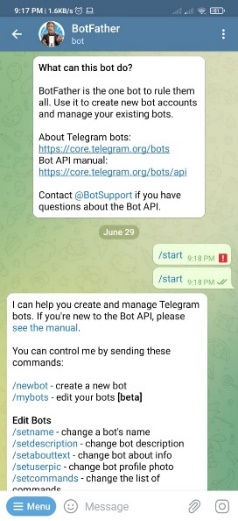
Produk yang dihasilkan adalah prototipe sistem *smart home* yang berupa saklar otomatis yang dapat dikontrol jarak jauh melalui internet menggunakan aplikasi Telegram. Aplikasi Telegram merupakan aplikasi yang umum digunakan sebagai media untuk berkomunikasi melalui *chat* dan panggilan. Aplikasi ini mudah ditemukan dan dapat diunduh secara gratis melalui *playstore.* Pengguna diminta untuk memasukkan nomor telepon untuk memulai menggunakan aplikasi. Selain digunakan sebagai media untuk berkomunikasi aplikasi ini juga dapat digunakan sebagai kontrol jarak jauh pada sistem *smart home* dengan memanfaatkan fitur *chat.* Pada proses ujicoba sistem, pengujian dilakukan melalui dua tahapan yaitu ujicoba tahap awal dan ujicoba tahap akhir. Pada tahap uji coba tahap awal dilakukan dengan menguji masing-masing komponen yang digunakan untuk mengetahui kendala dari sistem yang dirancang. Sedangkan tahap uji coba akhir dilakukan dengan menguji tingkat kelayakan yang dilakukan oleh pakar/ahli untuk mendapatkan penilaian dari prototipe yang telah dibuat.



Gambar 2. Blok diagram sistem *smart home* berbasis IoT

Penelitian ini memanfaatkan modul NodeMCU ESP32 yang diintegrasikan melalui jaringan wifi pada suatu rumah untuk mendapatkan akses internet.. Modul ini mempunyai fungsi seperti halnya modul mikrokontroler lainya seperti Arduino Uno atau Raspberry yang biasa digunakan sebagai sistem kontrol sederhana, yang membedakannya adalah modul ini sudah terdapat modul internal yang mendukung akses internet sedangkan pada modul Arduino Uno harus menambahkan board eksternal agar dapat terhubung dengan internet [10]. Modul NodeMCU ESP32 menerima masukan berupa perintah yang dikirimkan melalui fitur *chat* pada aplikasi Telegram untuk memberikan *trigger* pada relay yang berfungsi sebagai saklar otomatis. Blok diagram sistem *smart home* berbasis Iot ditunjukan pada Gambar 2.

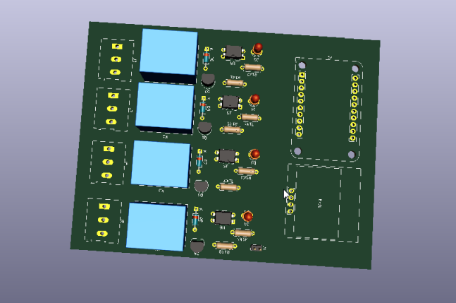
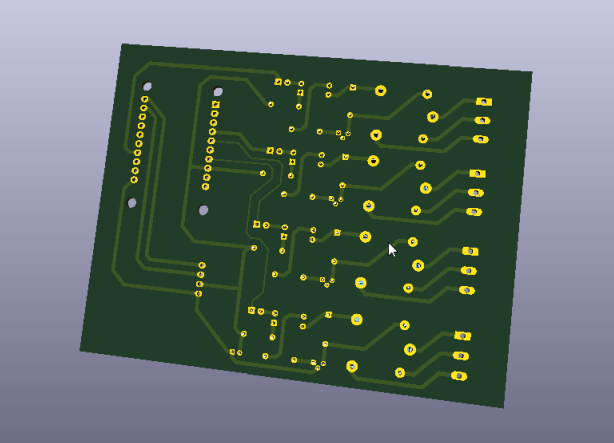
Pada dasarmya *bot* merupakan pengguna umum seperti biasa namun tidak mempunyai nomer telepon, *bot* hanya dapat merespon suatu perintah sesuai dengan parameter yang diatur oleh pembuat *bot*. Pengguna dapat memulai membuat *bot* Telegram dengan menggunakan BotFather. *Bot* ini sudah terdapat pada aplikasi Telegram dan dapat dicari dengan mengetikan nama *bot* pada kolom pencarian. BotFather juga disebut sebagai bapaknya semua *bot* dikarenakan semua *bot* yang ada di aplikasi Telegram diatur oleh *bot* ini, dengan kata lain disediakan menu untuk mengatur *bot* yang telah dibuat oleh pengguna [11]. Langkah untuk membuat bot Telegram adalah sebagai berikut: 1)Mengetikkan “BotFather” pada kolom pencarian dan memilihnya, 2) Menekan *start* untuk menjalankan BotFather, 3) Mengetik */newbot* untuk membuat memulai pembuatan *bot* baru, 4) Menuliskan nama untuk *bot* baru, 5) Membuat *username* pada *bot* yang telah dibuat, *username* pada *bot* diakhiri dengan kata “bot” sebagai contoh adalah SBMku\_bot, 6) Setelah *bot* berhasil dibuat maka pengguna akan mendapat kode API oleh Telegram. Kode tersebut digunakan sebagai sistem konfigurasi untuk menghubungkan NodeMCU ESP32 pada aplikasi Telegram. Proses pembuatan *bot* Telegram untuk *smart home* berbasis IoT ditunjukan pada Gambar 3.

|  |  |  |  |
| --- | --- | --- | --- |
| (a) | (b) | (c) | (d) |

Gambar 2. (a) Tampilan utama BotFather, (b) Memulai menggunakan BotFather, (c) Menentukan nama *bot*  yang akan digunakan dan mendapatkan kode API, (c) *Bot* telah selesai dibuat dan siap digunakan

Sistem *bot* pada Telegram memiliki fitur *autoreply* yang secara otomatis membalas pesan saat perintah dikirim, hal ini digunakan untuk mengetahui status dari modul NodeMCU ESP32. Saat pengguna memasukkan suatu perintah melalui *chat* misalnya untuk mengaktifkan relay 1 maka *bot* akan secara otomatis membalas pesan yang dikirimkan yaitu berupa pesan yang menyatakan status relay 1 saat ini, apabila relay belum aktif maka pesan ini akan memberikan perintah pada NodeMCU ESP32 untuk mengaktifkan relay 1. Selain itu apabila perintah yang sama dikirimkan ulang maka sistem *bot* akan membalas pesan tersebut dan menyatakan bahwa status relay 1 sudah aktif, hal ini juga berlaku untuk perintah lain misalnya ketika menonaktikan relay. Desain PCB dari prototipe *smart home* berbasis Iot digambar dengan menggunakan *software* KiCad ditunjukkan pada Gambar 4.

|  |  |
| --- | --- |
| (a) | (b) |

Gambar 4. (a) Desain PCB *smart home* berbasis IoT tampak atas, (b) Desain PCB tampak bawah

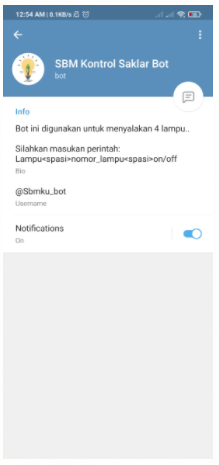
**3. Hasil dan Pembahasan**

Penelitian ini memanfaatkan NodeMCU ESP 32 sebagai akses untuk menghubungkan ke internet melalui jaringan *wifi* yang tersedia dalam rumah. Penerapan sistem kendali saklar dapat berfungsi dengan baik ketika *smartphone* dan perangkat NodeMCU ESP 32 terkoneksi dengan internet. Sistem pengiriman dan penerimaan perintah dilakukan dengan memanfaatkan fitur *chat* yang tersedia dalam *bot* Telegram. Dalam *bot* tersebut terdapat panduan dalam memberikan perintah kepada sistem dan secara otomatis akan memberikan balasan saat perintah yang diberikan. Sistem akan mengirimkan pesan *auto reply* ke pengguna sebagai notifikasi bahwa sistem sudah merespon perintah untuk mengaktifkan dan menonaktifkan relay dengan baik. Mode pengontrolan *bot* Telegram memiliki 2 metode yaitu metode pengetikan secara langsung dan melalui *inline* pada *keyboard*. Untuk melakukan pengetikan secara langsung dapat dilakukan secara langsung melalui *keyboard* dan untuk memuculkan *inline* pada *keyboard* hal yang dilakukan adalah mengatur pada bagian /option pada telegram. Daftar perintah dan balasan *bot* pada aplikasi Telegram *smart home* berbasis IoT ditunjukkan pada Tabel 1.

Tabel 1. Daftar perintah dan balasan *bot* pada aplikasi Telegram

|  |  |  |
| --- | --- | --- |
| **No** | **Perintah** | **Balasan *autoreply*** |
| 1 | Tampilan Awal | Halo, Selamat Datang, Silahkan masukkan perintah : Lampu<spasi>nomor<>on/off |
| 2 | “Lampu 1 On” | * Menyalakan Lampu 1 * Lampu 1 Sudah Menyala |
| 3 | “Lampu 2 On” | * Menyalakan Lampu 2 * Lampu 2 Sudah Menyala |
| 4 | “Lampu 3 On” | * Menyalakan Lampu 3 * Lampu 3 Sudah Menyala |
| 5 | “Lampu 4 On” | * Menyalakan Lampu 4 * Lampu 4 Sudah Menyala |
| 6 | “Lampu 1 Off” | * Mematikan Lampu 1 * Lampu 1 Sudah Mati |
| 7 | “Lampu 2 Off” | * Mematikan Lampu 2 * Lampu 2 Sudah Mati |
| 8 | “Lampu 3 Off” | * Mematikan Lampu 3 * Lampu 3 Sudah Mati |
| 9 | “Lampu 4 Off” | * Mematikan Lampu 4 * Lampu 4 Sudah Mati |
| 10 | Selain yang ditetapkan | “Perintah tidak ditemukan, Silahkan masukkan perintah : Lampu<spasi>nomor<>on/off'” |

Alur program *smart home* pada penelitian ini dimulai mengetikan nama *bot* yang dipakai dalam penelitian ini yaitu SBMku\_bot. Untuk memulai menggunakan program adalah dengan menekan *start* pada *chat bot* Telegram yang telah dinamakan sebelumnya sebagai SBMku\_bot. Saat pertama kali memasuki *bot* maka akan terdapat pesan “Halo, Selamat Datang, Silahkan masukkan perintah: Lampu<spasi>nomor<>on/off". Setelah itu untuk meyalakan saklar yang diinginkan, dapat menulis perintah pada *inline keyboard*. Penelitian ini menggunakan modul relay sebagai saklar otomatis yang dapat dikendalikan melalui aplikasi Telegram, terdapat empat buah saklar yang dapat dipakai untuk mengendalikan peralatan listrik rumah tangga. Sebagai contoh untuk menyalakan saklar pertama adalah dengan mengetikkan “lampu 1 on”, maka secara otomatis *bot* akan mengirimkan pesan balasan yaitu “lampu 1 on”. Perintah ini diterima oleh NodeMCU ESP32 untuk memberikan *trigger* kepada modul relay yang berfungsi sebagai saklar otomatis maka saklar pertama akan aktif. Setelah saklar pertama telah aktif maka NodeMCU ESP32 mengirimkan notifikasi pada *bot* dan menyatakan bahwa lampu telah menyala. Saat lampu menyala maka *bot* akan mengirimkan pesan “Menyalakan Lampu 1”. Apabila perintah diulangi dengan cara yang sama maka *bot* akan membalas dengan mengirimkan pesan “Lampu 1 Sudah Menyala”. Hal ini juga berlaku sebaliknya, apabila pengguna mengirimkan pesan “lampu 1 off” maka lampu akan padam dan NodeMCU akan mengirimkan notifikasi yang menyatakan bahwa lampu telah padam. Apabila pesan yang ditulis diluar dari kata yang telah ditentukan maka *bot* akan mengirimkan pesan yaitu “Perintah tidak ditemukan, Silahkan masukkan perintah: Lampu < spasi> nomor <spasi> on/off”, yang menandakan bahwa penulisan pesan masih belum tepat. Pada tiap relay terdapat sebuah led yang berguna sebagai indikator saat relay dalam kondisi aktif. Proses penggunaan *bot* Telegram untuk menghidupkan saklar ditunjukan pada Gambar 5.

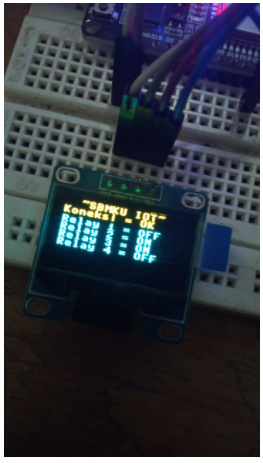
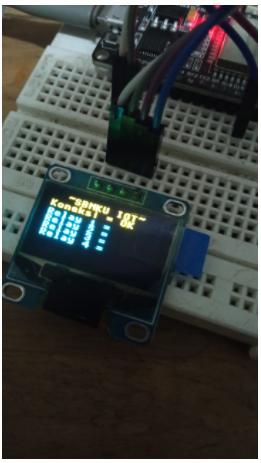


|  |  |  |  |
| --- | --- | --- | --- |
| (a) | (b) | (c) | (d) |

Gambar 5. (a) Tampilan utama aplikasi *smart home* SBMku\_bot, (b) Tampilan untuk memulai *bot*,

(c) Fitur *autoreply* berisi petunjuk penggunaan, (d) Tampilan ujicoba kontrol saklar lampu

Penelitian ini menggunakan bahasa pemrograman Micropython. Selain diprogram menggunakan Arduino IDE modul NodeMCU ESP32 juga memungkinkan diprogram menggunakan Micropython. Micropython merupakan sub bagian bahasa pemrograman python yang digunakan untuk pemrograman mikrokontroler. Modul ini diprogram melalui aplikasi Upyfrat *Integrated Development Environment* (IDE). Upycraft IDE merupakan software yang digunakan untuk melakukan penulisan program, meng*compile* serta untuk meng*upload* program ke NodeMCU ESP32. Pada penerapan *smart home* berbasis IoT, sistem menggunakan beberapa *file* yang dicompile kedalam modul diantaranya adalah config.py, main.py, urequest.py dan utelegram.py. Module tersebut merupakan gabungan dari beberapa *library* dan program yang dituliskan kedalam modul. Urequests adalah modul python yang bisa digunakan untuk mengirim berbagai request HTTP. Sedangkan utelegram adalah modul python yang diggunakan untuk mengirim berbagai request HTTP ke server Telegram. Selanjutnya utelegram dipakai untuk server berkomunikasi dengan *bot* Telegram yang telah dibuat dan sehingga dapat meneruskan perintah ke modul Node MCU ESP32. Selain menampilkan balasan status saklar yang terpakai prototipe ini juga memiliki modul LCD untuk mengetahui status relay yang dipakai. Selain itu juga terdapat indikator yang menyatakan bahwa modul telah terhubung jaringan interne. Proses pengkoneksian modul dengan internet dan status penggunaan saklar ditunjukkan pada Gambar 6.



|  |  |  |
| --- | --- | --- |
| (a) | (b) | (c) |

Gambar 6. (a) Tampilan saat prototipe *smart home* gagal terkoneksi internet, (b) Tampilan saat prototipe *smart home* berhasil terkoneksi internet, (c) Tampilan status saklar saat prototipe *smart home* digunakan

Pengujian kelayakan prototipe *smart home* berbasis IoT ini dianalisis berdasarkan hasil pengisian instrumen kuesioner yang diajukan pada pakar atau ahli. Pakar menilai beberapa aspek yang diajukan, beberapa aspek yang dinilai antara lain adalah dari segi desain prototipe yaitu desain telah praktis dan memenuhi nilai ergonomi, aspek kedua yang dinilai adalah kinerja dari prototipe yang dibangun yaitu dapat berjalan normal seperti yang diharapkan, aspek ketiga yang dinilai adalah kemudahan dalam pengoperasian, aspek terakhir yang dinilai adalah implementasi *smart home* berbasis IoT ini apakah dapat diterapkan dalam kehidupan sehari-hari. Hasil uji kalayakan ditunjukan pada diagram pada Gambar 7.

Gambar 7. Persentase hasil uji kelayakan

Penilaian uji kelayakan prototipe tersebut ditujukan kepada tiga ahli di bidangnya dan memiliki wewenang untuk menilai aspek tersebut. Uji kelayakan dilakukan untuk mendapatkan tanggapan dan penilaian dari pakar untuk mendapatkan rekomendasi dan saran pengembangan. Berdasarkan hasil penilaian kelayakan menunjukkan bahwa persentase aspek desain alat adalah 85%, persentase aspek kinerja alat adalah 87.5%, persentase aspek kemudahan pengoperasian memiliki adalah 85%, dan persentase implementasi alat adalah 80%. Dari keempat aspek yang diuji memiliki nilai rata-rata keseluruhan adalah 84.37% dan dapat disimpulkan bahwa prototipe smart home berbasis IoT masuk kedalam kategori layak untuk diimplementasikan.

**4. Kesimpulan**

Berdasarkan hasil penelitian rancang bangun *smart home* berbasis IoT menggunakan Telegram Messenger Bot dan NodeMCU ESP 32 dapat disimpulkan bahwa tujuan penelitian ini adalah menerapkan sistem kendali otomatis yang dapat dikendalikan jarak jauh menggunakan fitur chat. Alat ini dibangun dengan menggunakan pemrograman *micropython* yang mana bahasa pemrograman ini masih jarang diterapkan dibanding program yang umum digunakan seperti Arduino IDE*.* Berdasarkan hasil penilaian oleh pakar atau ahli menunjukkan bahwa hasil uji kelayakan mendapatkan hasil persentase nilai rata-rata keseluruhan adalah 84.37% dan dapat disimpulkan bahwa prototipe *smart home* berbasis IoT masuk kedalam kategori layak untuk diimplementasikan. Penelitian ini dapat diimplementasikan secara langsung dikarenakan sudah terdapat desain fisiknya sehingga dalam penerapannya mudah dalam merangkainya. Penelitian ini sangat fleksibel dan terjangkau dikarenakan komponen yang digunakan cukup sedikit dan hanya terdiri dari beberapa komponen saja. Untuk pengembangan selanjutnya dapat dilakukan dengan menambahkan fitur *click-chat* pada *inline keyboard* yaitu fitur memilih menu yang disediakan agar pengguna tidak perlu mengetik perintah sehingga kesalahan pengetikan dapat dihindari

**Referensi**

1. Eryawan, B., Jayati, A. E., & Heranurweni, S.., “Rancang bangun prototype smart home dengan konsep internet of things (iot) menggunakan raspberry pi berbasis web”, *Elektrika*, 11(2), 1-5, 2019.
2. Malche, T., & Maheshwary, P., “Internet of Things (IoT) for building smart home system”, *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 65-70, 2017.
3. Taiwo, O., Ezugwu, A. E., Rana, N., & Abdulhamid, S. I. M., “Smart home automation system using zigbee, bluetooth and arduino technologies”, *International Conference on Computational Science and Its Applications,* 587-597, 2020.
4. Khotimah, K., Santoso, A. B., Ma’arif, M., Azhiimah, A. N., Suprianto, B., Sumbawati, M. S., & Rijanto, T., “Validation of voice recognition in various Google voice languages using voice recognition module V3 based on microcontroller”, *2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE),* 1-6, 2020.
5. Wadhwani, S., Singh, U., Singh, P., & Dwivedi, S., “Smart home automation and security system using Arduino and IOT”, *International Research Journal of Engineering and Technology (IRJET)*, 5(2), 1357-1359, 2018.
6. Ali, M. A., Miry, A. H., & Salman, T. M., “IoT Based Water Tank Level Control System Using PLC”, *2020 International Conference on Computer Science and Software Engineering (CSASE)*, 7-12, 2020, IEEE.
7. Munadi, R., Rakhman, A., & Perdana, D., “Smart Garage Implementation and Design Using Whatsapp Communication Media”. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 16(3), 1107-1113, 2018.
8. Dargahi Nobari, A., Reshadatmand, N., & Neshati, M., “Analysis of Telegram, an instant messaging service”, *2017 ACM on Conference on Information and Knowledge Management*, 2035-2038, 2017.
9. Fathoni, A. N., Hudallah, N., Putri, R. D. M., Khotimah, K., Rijanto, T., & Ma’arif, M., “Design automatic dispenser for blind people based on Arduino Mega using DS18B20 Temperature Sensor”. *2020 Third International Conference on Vocational Education and Electrical Engineering (ICVEE)*, 1-5, 2020, IEEE.
10. Thakare, S., & Bhagat, P. H., “Arduino-based smart irrigation using sensors and ESP8266 WiFi module”, *2018 Second International Conference on intelligent computing and control systems (ICICCS)*, 1-5, 2020, IEEE.
11. Kurniawan, A. D., “Design and Implementation of Home Security Using Telegram Botfather”, *FIDELITY: Jurnal Teknik Elektro*, *2*(1), 11-15, 2020.