



Versi Online tersedia di : <https://jurnal.buddhidharma.ac.id/index.php/algor/index>

JURNAL ALGOR

[|2715-0577 \(Online\)| 2715-0569 \(Print\)](https://jurnal.buddhidharma.ac.id/index.php/algor/index)



Artikel

Implementasi Speech Recognition untuk Asisten Virtual dengan Python

Yulius Setyawan¹, Yo Ceng Giap²

¹ Buddhi Dharma University, Faculty of Science & Technology, Banten, Indonesia

SUBMISSION TRACK

Received: August 17, 2022

Final Revision: September 10, 2022

Available Online: September 15, 2022

KEYWORD

Arduino, *Experimental*, Python, *Voice Command*

KORESPONDENSI

Phone: 082208244317

E-mail: yuliussetyawan@gmail.com

A B S T R A K

Cara berkomunikasi yang paling umum digunakan oleh manusia adalah suara atau ucapan. Banyak eksperimen yang dilakukan di aspek pemrosesan suara untuk menciptakan model mekanis yang mensimulasikan bagaimana cara manusia berkomunikasi secara verbal. Teknologi pengenalan suara bertujuan untuk mengembangkan sistem dan teknik tentang bagaimana mengenalkan perintah berbentuk suara agar dapat dimengerti oleh mesin, sehingga mesin mampu menjalankan perintah yang diberikan. Penelitian ini menggunakan teknologi pengenalan suara untuk diimplementasikan ke aplikasi asisten virtual pribadi berbasis desktop dengan tujuan membantu menyederhanakan tugas sehari-hari melalui komputer. Hasil penelitian ini menghasilkan sebuah aplikasi pengenalan suara (*speech recognition*) untuk menjalankan tugas umum tertentu di desktop dan menyalakan perangkat elektronik. Dari pengujian yang sudah dilakukan, aplikasi dapat mengenali perintah suara untuk menjalankan perintah yang diberikan dan dapat menyalakan ataupun mematikan perangkat listrik seperti yang diharapkan.

PENGANTAR

Cara berkomunikasi yang paling umum digunakan oleh manusia adalah suara atau ucapan. Banyak eksperimen yang sudah dilakukan di berbagai aspek pemrosesan suara untuk menciptakan model mekanis yang mensimulasikan bagaimana manusia berkomunikasi secara verbal, salah satunya adalah teknologi pengenalan suara. Teknologi pengenalan suara bertujuan mengembangkan sistem dan teknik tentang bagaimana mengenalkan perintah berbentuk suara agar dapat dimengerti oleh mesin, sehingga mesin mampu menjalankan perintah yang diberikan. *Speech recognition* merupakan prosedur merekam pembicaraan yang dituturkan lewat mikrofon atau telepon dan mengonversikan ke bentuk data yang bersifat digital. Di samping *speech recognition* ada juga sistem pengenalan suara lain yaitu *speaker recognition*. *Speaker recognition* adalah proses yang dilakukan oleh perangkat untuk mengetahui pembicara berdasarkan suara. Perbedaannya adalah *speech recognition* mengidentifikasi apa yang dikatakan, sedangkan *speaker recognition* mengenali siapa yang berbicara.

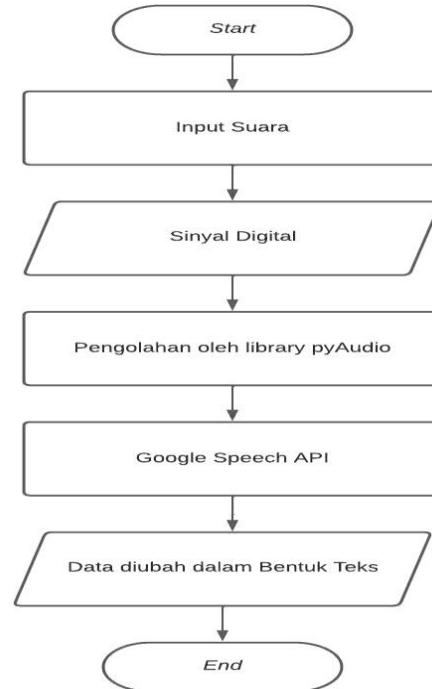
Implementasi pengenalan suara (*speech recognition*) dapat ditemukan pada perangkat Android berupa fitur *Google voice*. *Google voice* adalah contoh dari implementasi teknologi pengenalan suara yang mutakhir dan diyakini mampu menyaingi kemampuan jutaan syaraf yang dimiliki otak manusia. *Google voice* akan mengidentifikasi konsonan dan vokal penggunaannya, kemudian informasi tersebut digunakan untuk menaksir prediksi perintah menurut pola syaraf otak manusia. [1]. Salah satu bahasa pemrograman yang populer dan dikenal handal adalah Python. Berkat dukungan komunitas aktif di seluruh dunia, Python merupakan suatu bahasa untuk menulis kode yang mampu bertahan dan terus bertumbuh sampai saat ini.

Semakin berkembangnya *Google voice* beriringan dengan bahasa pemrograman Python yang juga terus berkembang, penulis memiliki ide untuk membuat asisten suara virtual dengan Python berbasis desktop dan juga diimplementasikan ke sebuah mikrokontroler

Arduino dengan tujuan mengendalikan perangkat elektronik melalui perintah suara.

I. METODE

1.1 *Speech Processing*

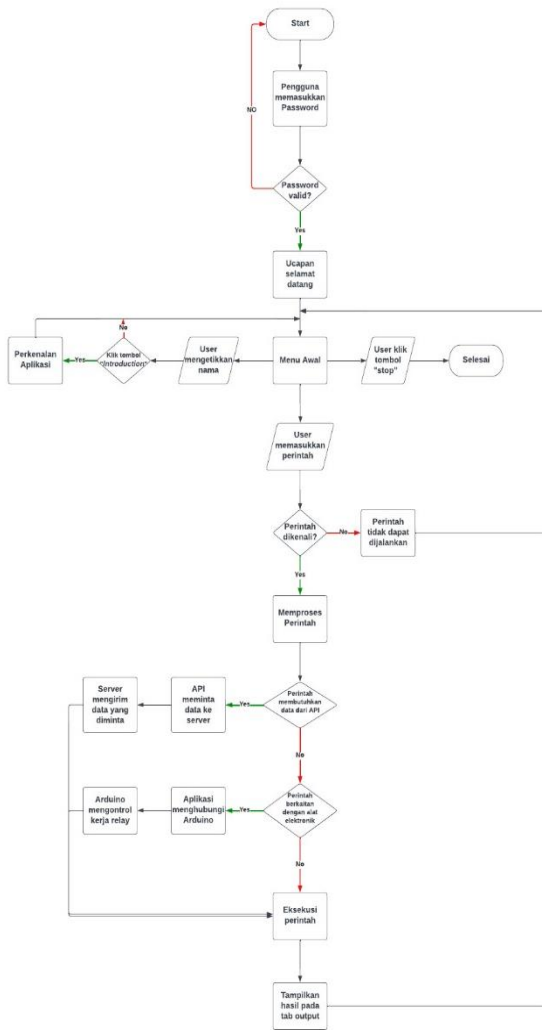


Gambar 1. Alur *Speech Processing*

Metode yang digunakan dalam proses perancangan aplikasi asisten virtual ini adalah *flowchart*. *Flowchart* adalah representasi simbolis dari algoritma atau prosedur yang digunakan untuk memecahkan masalah [3]. Alur tahapan *speech processing* ini adalah:

- Setelah dilakukan perekaman suara menggunakan mikrofon, sinyal analog dari suara manusia akan dirubah menjadi sinyal digital.
- Sinyal digital tersebut akan dikenali oleh *library pyaudio* dan selanjutnya dikirim via internet untuk diolah oleh *Google Speech API*.
- Google Speech API* berperan untuk mengubah sinyal digital tersebut ke dalam bentuk teks sehingga ucapan dapat dikenali oleh komputer.

1.2 Flowchart Aplikasi



Gambar 2. Flowchart Aplikasi

Alur *flowchart* dalam pembuatan aplikasi adalah sebagai berikut:

- Pengguna harus login menggunakan password untuk mulai menggunakan aplikasi.
- Setelah login, aplikasi akan mengucapkan salam (selamat pagi/ siang/ malam) kepada pengguna.
- Kemudian akan muncul tampilan menu awal yang dapat digunakan pengguna untuk mulai memberikan perintah.
- Jika perintah dikenali, aplikasi akan memproses perintah yang diberikan dan menampilkan hasilnya pada layar.
- Apabila perintah tidak dikenali, maka aplikasi akan meminta pengguna untuk

mengulang kembali dengan perintah yang berbeda.

II. HASIL

“*Speech recognition adalah kemampuan komputer untuk memahami dan mengidentifikasi suara seseorang, terlepas dari siapa yang berbicara. Implementasi speech recognition contohnya adalah mengeksekusi program yang ada di komputer memanfaatkan bantuan perintah suara.*” [3]

Asisten virtual adalah sistem yang bekerja layaknya asisten pribadi dan menjalankan bermacam kewajiban termasuk mencari informasi dan data beserta menjalankan tugas yang mengaitkan aplikasi ke sebuah perangkat, Misalnya, memanfaatkan perintah suara untuk mengecek tanggal atau waktu, menyalakan alarm, dan bergabung ke ruang diskusi [4].

Firmata adalah sebuah protokol untuk menerima dan mengirim pesan dari perangkat lunak komputer ke mikrokontroler. Protokol tersebut diimplementasikan ke perangkat bawaan mikrokontroler atau paket perangkat lunak. Pada dasarnya, semua jenis mikrokontroler dapat mengimplementasikan protokol Firmata, akan tetapi saat ini implementasi yang paling lengkap masih ditemukan pada mikrokontroler Arduino [5].

Python dikenal sebagai bahasa yang mengkombinasikan antara kemampuan, kecakapan, dilengkapi sintaks kode yang mudah, serta ditambah fungsionalitas pustaka standar yang lengkap. Perjalanan bahasa Python dimulai pada tahun 1990. Seorang pemrogram asal Amsterdam, Belanda bernama Guido Van Rossum menciptakan bahasa Python yang berlokasi di CWI (*Centrum Wiskunde & Informatica*). Hingga kini Python terus dibesarkan dan dijalankan tim pemrogram yang dipimpin oleh Guido bersama lembaga *Python Software Foundation* [6].

III. PEMBAHASAN

3.1 Analisis Kebutuhan

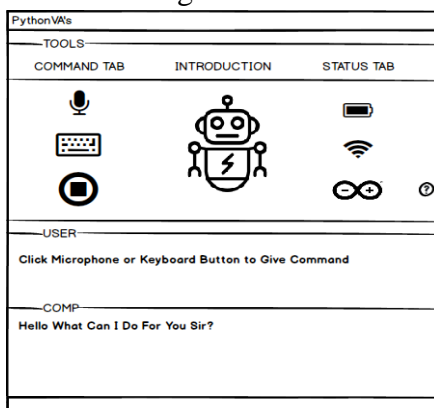
Berdasarkan informasi penelitian dari jurnal sebelumnya berjudul *The Voice Enabled Personal Assistant for PC using Python* [7] dan *Personal Virtual Assistant for Windows using Python* [8] umumnya pengguna membutuhkan asisten virtual untuk melakukan tugas-tugas berikut:

1. Memeriksa kondisi cuaca di suatu kota (*showing weather condition*).
2. Memutar musik, film, dan lainnya (*play music, movies, etc*).
3. Penjadwalan kegiatan (*scheduling activity*).
4. Menampilkan tanggal waktu (*showing datetime*).
5. Mengelola email (*managing email*).
6. Membacakan berita (*reading news*).
7. Membuka aplikasi (*open apps*).
8. Melakukan pencarian di Google (*Google search*).

3.2 Desain

A. Perancangan Tampilan Antar Muka

Diperlukan tampilan GUI (*Graphical User Interface*) untuk mempermudah pengoperasian aplikasi. Diperkirakan rancangan tampilan sederhana yang akan menjadi dasar pembuatan tampilan antar muka adalah sebagai berikut:



Gambar 3. Rancangan Tampilan Antar Muka

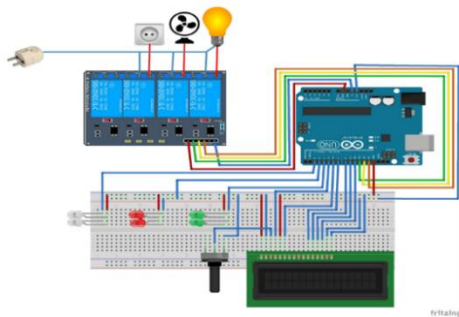
B. Perancangan Alat

Tahap selanjutnya adalah perancangan *prototype* alat untuk menunjang sistem yang akan dibangun.

- a. Pembuatan menggunakan *breadboard* sebagai konduktor listrik dari satu komponen ke komponen lainnya.
- b. Module LCD 16x2 digunakan untuk menampilkan hasil *output* dari komputer ke dalam layar. Urutan pemasangan pin LCD adalah sebagai berikut:
 - Pin GND dihubungkan dengan jalur GND pada *breadboard*.
 - Pin VDD dihubungkan dengan jalur 5V pada *breadboard*.
 - Pin V0 dihubungkan ke pin tengah potentiometer 10K ohm.
 - Pin RS dihubungkan ke pin ~5 Arduino.
 - Pin RW dihubungkan dengan jalur GND pada *breadboard*.
 - Pin E dihubungkan ke pin ~6 Arduino.
 - Pin D4-D7 dihubungkan ke pin ~7 sampai pin ~10 Arduino secara berurutan.
 - Pin BLA dihubungkan dengan jalur tegangan 5V pada *breadboard*.
 - Pin BLK dihubungkan dengan jalur GND pada *breadboard*.
- c. Pemasangan LED dimana setiap kaki anoda diletakkan pada jalur GND. Pada LED putih kaki katoda dihubungkan ke pin ~2 Arduino, LED merah disambungkan ke pin ~3 Arduino, LED hijau disambungkan menuju pin ~4 Arduino.
- d. Setiap warna LED melambangkan proses ketika program sedang berjalan. LED warna putih akan menyala saat program sudah siap menerima perintah dari pengguna. LED warna hijau menyala ketika program berhasil

mengeksekusi perintah user. Terakhir LED warna merah akan menyala jika program tidak mengenal perintah yang diberikan atau terjadi kesalahan.

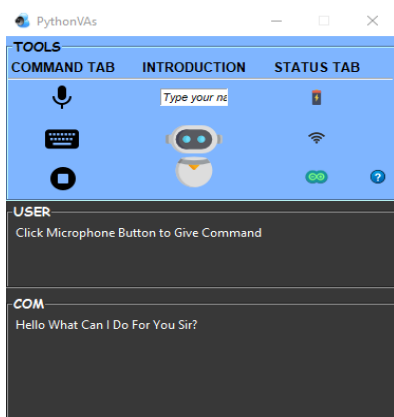
- e. Relay digunakan untuk membantu menyalakan dan mematikan perangkat elektronik. Jalur IN1 (Input 1), IN2, dan IN3 masing-masing dihubungkan ke pin nomor ~11, ~12, dan 13 secara berurutan.
- f. Jalur GND relay disambungkan ke pin Arduino GND. Sedangkan VCC relay dihubungkan ke pin Arduino power 5V.



Gambar 4. Skema Rancangan Alat

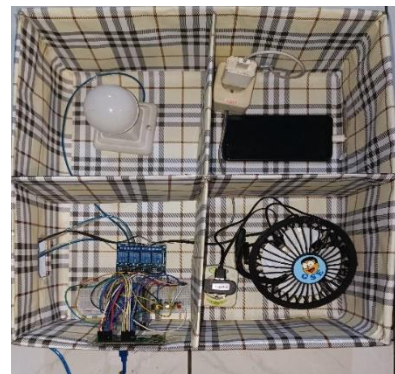
3.3 Implementasi

Berdasarkan rancangan tampilan antar muka pada gambar 2, maka *layout* dibuat menggunakan module tkinter yang sudah termasuk di dalam standar *library* Python. Berikut hasil implementasi tampilan aplikasi:



Gambar 5. Tampilan Aplikasi

Gambar dibawah ini memperlihatkan komponen alat yang sudah dipasang dan siap untuk digunakan. Komponen alat terdiri dari Arduino UNO R3, LCD 16x2, LED, relay, potentiometer, dan kabel jumper. Perangkat elektronik penunjang seperti lampu, fitting, kabel serabut, stop kontak, kabel charger, telepon genggam, dan kipas angin juga termasuk ke dalam komponen penunjang.

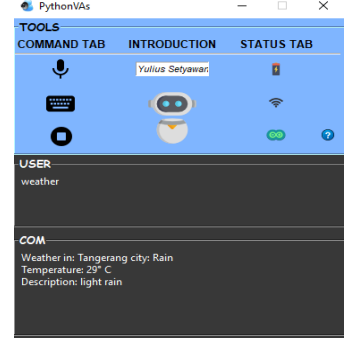
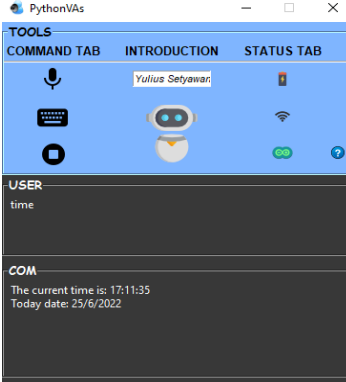
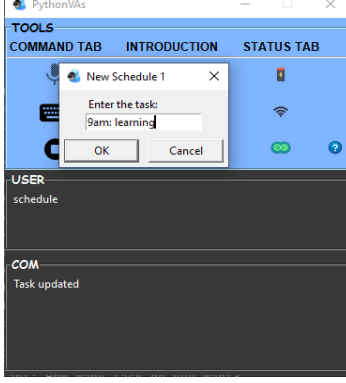


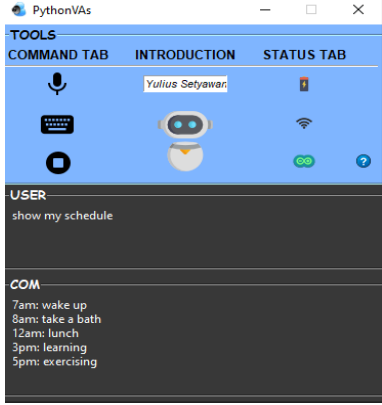
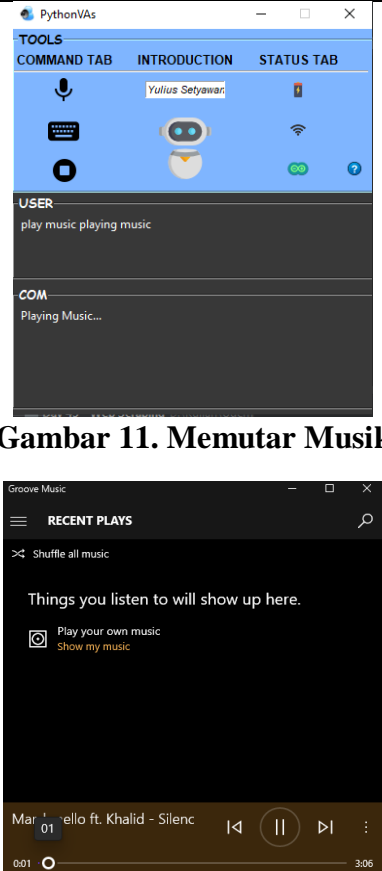
Gambar 6. Tampilan Alat

3.4 Pengujian

Penulis menggunakan metode pengujian *blackbox* untuk mengetahui fungsi dan komponen alat apakah dapat berjalan dengan semestinya. Pengujian yang dilakukan meliputi pengujian fungsi utama dan pengujian alat. Setiap fungsi dan fitur akan diuji coba berdasarkan sudut pandang pengguna untuk mengetahui *output* yang dihasilkan. Berikut ini hasil pengujian 5 dari 8 fitur utama yang ada:


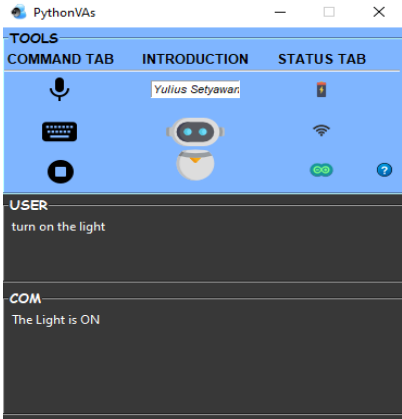

Tabel 1. Pengujian Kotak Hitam (Blackbox)


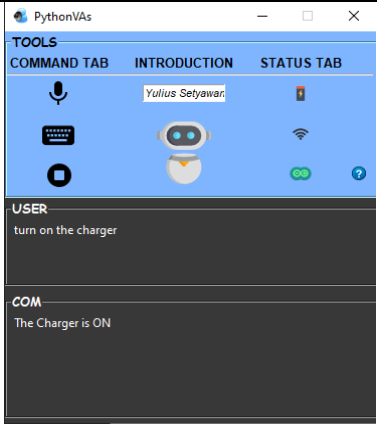

| No. | Bentuk Pengujian | Tindakan Aplikasi | Tampilan Pengujian |
|-----|---------------------------------------|---|---|
| 1. | Memeriksa kondisi cuaca di suatu kota | Aplikasi membacakan kondisi cuaca saat ini dan hasilnya akan muncul pada tab <i>com</i> . |  <p>Gambar 7. Menampilkan Kondisi Cuaca</p> |
| 2. | Menampilkan tanggal dan waktu | Aplikasi membacakan dan menampilkan waktu saat ini. |  <p>Gambar 8. Menampilkan Tanggal & Waktu</p> |
| 3. | Penjadwalan kegiatan | Aplikasi dapat menerima input user untuk penjadwalan kegiatan dan memberikan info jadwal yang sudah dibuat. |  <p>Gambar 9. Form Jadwal</p> |


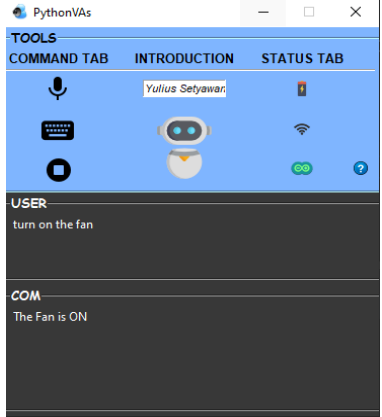

| | | | |
|-----------------------------------|------------------------|--|--|
| | | |  <p>Gambar 10. Menampilkan Jadwal</p> |
| <p>4. Memutar musik dan film.</p> | <p>musik dan film.</p> | <p>Aplikasi dapat memutar musik dan film yang berada dalam <i>file</i> komputer.</p> |  <p>Gambar 11. Memutar Musik</p> <p>Gambar 12. Musik Diputar</p> |

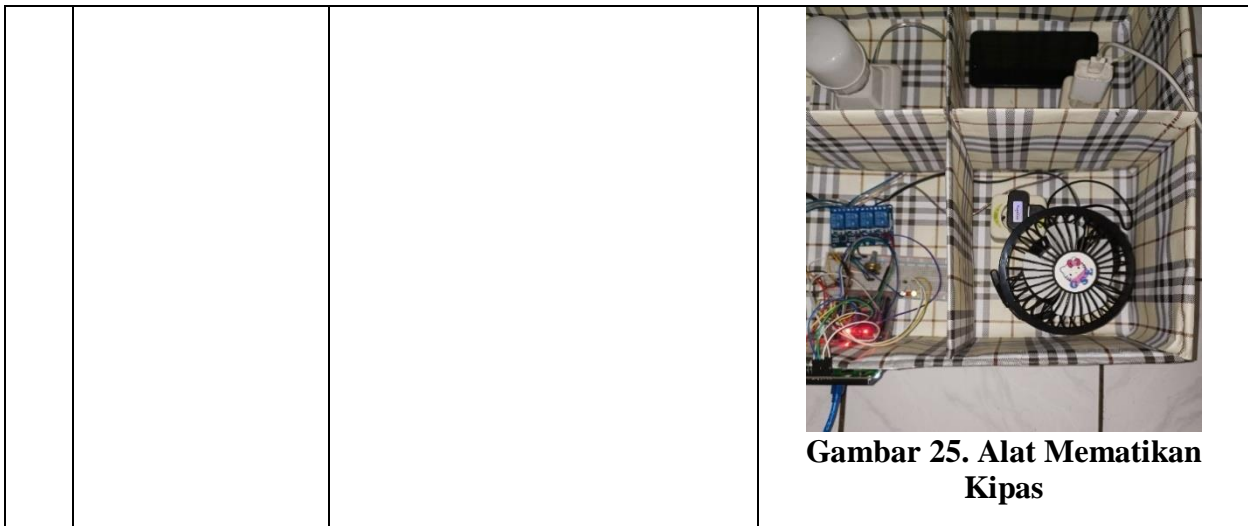
| | | | |
|-----------|-------------------------|---|--|
| <p>5.</p> | <p>Mengelola email.</p> | <p>Aplikasi dapat mengirim email dengan isi dan tujuan yang ditentukan oleh pengguna.</p> | <div data-bbox="979 188 1337 586"> </div> <p style="text-align: center;">Gambar 13. Mengisi Pesan Email</p> <div data-bbox="970 654 1347 1093"> </div> <p style="text-align: center;">Gambar 14 Mengisi Alamat Email</p> <div data-bbox="944 1236 1378 1572"> </div> <p style="text-align: center;">Gambar 15. Email Berhasil Dikirim</p> |
|-----------|-------------------------|---|--|

Tabel 2. Pengujian Alat

| No | Skenario Pengujian | Hasil yang diharapkan | Pengujian |
|----|--------------------------------|--|--|
| 1. | Pengujian fungsi LCD | LCD dapat menampilkan teks dan <i>output</i> dari perintah user. |  <p style="text-align: center;">Gambar 16. Fungsi LCD</p> |
| 2. | Menyalakan dan mematikan lampu | Aplikasi yang dibantu dengan alat dapat menyalakan dan mematikan lampu sesuai perintah pengguna. |  <p style="text-align: center;">Gambar 17. Aplikasi Menyalakan Lampu</p>  <p style="text-align: center;">Gambar 18. Alat Menyalakan Lampu</p> |

| | | | |
|-----------|---|--|--|
| | | |  <p>Gambar 19. Alat Mematikan Lampu</p> |
| <p>3.</p> | <p>Menyalakan dan mematikan <i>charger</i>.</p> | <p>Aplikasi yang dibantu dengan alat dapat menyalakan dan mematikan <i>charger</i> sesuai perintah pengguna.</p> |  <p>Gambar 20. Aplikasi Menyalakan Charger</p>  <p>Gambar 21. Alat Menyalakan Charger</p> |

| | | | |
|-----------|--|---|---|
| | | |  <p>Gambar 22. Alat Mematikan Charger</p> |
| <p>4.</p> | <p>Menyalakan dan mematikan kipas angin.</p> | <p>Aplikasi yang dibantu dengan alat dapat menyalakan dan mematikan kipas angin sesuai perintah pengguna.</p> |  <p>Gambar 23. Aplikasi Menyalakan Kipas</p>  <p>Gambar 24. Alat Menyalakan Kipas</p> |



Dalam penelitian ini, beberapa pertanyaan diajukan kepada pengguna aplikasi yang terdiri dari 20 responden. Berikut pertanyaan yang diajukan:

Tabel 3 Kuesioner Kepuasan

| No. | Rating Element | SS | S | KS | TS |
|-----|--|----|---|----|----|
| 1. | Cara kerja dari aplikasi dan alat mudah untuk dipahami | ✓ | | | |
| 2. | Aplikasi dan alat mudah digunakan | ✓ | | | |
| 3. | Fitur yang ada sudah sesuai dengan yang dibutuhkan | ✓ | | | |
| 4. | Tampilan dari aplikasi ini menarik | | ✓ | | |
| 5. | Secara keseluruhan Anda merasa terbantu dengan adanya aplikasi ini | ✓ | | | |

Kuesioner penilaian memiliki empat skala tanggapan, yakni SS (Sangat Setuju), S (Setuju), KS (Kurang Setuju), dan TS (Tidak Setuju). 20 pengguna dipilih berdasarkan orang-orang yang sudah familiar dengan penggunaan komputer dalam kehidupan sehari-hari. Dari hasil jawaban pengguna, data diolah

untuk setiap pertanyaan dengan menggunakan skala seperti berikut:

Tabel 4. Peringkat Nilai

| Opsi | Nilai |
|---------------|-------|
| Sangat Setuju | 4 |
| Setuju | 3 |
| Kurang Setuju | 2 |
| Tidak Setuju | 1 |

Tabel 5. Hasil Kuesioner

| Elemen Peringkat | Hasil |
|------------------|-------|
| Q1 | 77 |
| Q2 | 75 |
| Q3 | 75 |
| Q4 | 73 |
| Q5 | 76 |
| Total | 376 |
| Skor Maksimum | 400 |

Berdasarkan hasil jawaban dari 20 responden tingkat kepuasan pengguna adalah $(376 / 400) * 100\% = 94\%$.

IV. SIMPULAN DAN SARAN

A. Simpulan

Berdasarkan hasil implementasi pembuatan aplikasi “*PythonVAs*” (kependekan dari *Python Voice Assistant*), aplikasi ini dapat membantu pengguna untuk mempermudah pekerjaan mereka sehari-hari di komputer desktop. *Library Speech Recognition* dari bahasa pemrograman Python dapat mengenali ucapan Bahasa Inggris dengan menggunakan bantuan Google dan Microsoft *Speech API*. Perpaduan bahasa pemrograman Python dengan protokol *pyfirmata* untuk mengendalikan mikrokontroler Arduino UNO menghasilkan aplikasi yang dapat mengoperasikan peralatan elektronik dan mampu menjalankan perintah agar sesuai dengan harapan berdasarkan ucapan penggunanya. Aplikasi asisten virtual ini dibangun berbasis desktop agar pengguna dapat lebih mudah dalam menjalankan aplikasi, terbukti dari hasil kuesioner yang

menyatakan tingkat kepuasan pengguna adalah 94% dari skor maksimum 100%, persentase ini berada pada kategori skala yang sangat bagus.

B. Saran

Terdapat beberapa aspek yang masih bisa diperbaiki. Berikut adalah saran dari peneliti:

1. Tampilan antar muka masih kurang menarik dan kurang interaktif. Diperlukan penambahan animasi dan ikon untuk membuatnya lebih *user-friendly*.
2. Aplikasi dikembangkan kembali agar dapat dioperasikan oleh perangkat dengan sistem operasi selain windows, seperti Android dan IOS.
3. Aplikasi sebaiknya dapat dijalankan dengan lancar dalam keadaan tanpa koneksi internet (*offline*).
4. Menambahkan pilihan bahasa selain bahasa Inggris ke dalam aplikasi.

REFERENSI

- [1] I Komang Setia Buana, “Implementasi Aplikasi Speech to Text untuk Memudahkan Wartawan Mencatat Wawancara dengan Python,” *Jurnal Sistem dan Informatika (JSI)*, vol. 14, no. 2, hlm. 135–142, Agu 2020, doi: 10.30864/jsi.v14i2.293.
- [2] R. N. Santoso, “Perencanaan dan Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas (Studi Kasus Politeknik Negeri Tanah Laut),” 2017.
- [3] N. F. I. Prayoga, “Analisis Speaker Recognition Menggunakan Metode Dynamic Time Warping (DTW) Berbasis Matlab,” *AVITEC*, vol. 1, no. 1, Agu 2019, doi: 10.28989/avitec.v1i1.492.
- [4] M. F. McTear dan Z. Callejas, *Voice Application Development for Android*. Packt Publishing, 2013.
- [5] “Firmata Protocol Documentation,” *Github*, Mar 03, 2022. <https://github.com/firmata/protocol> (diakses Jul 18, 2022).
- [6] Moh. Hamim Zajuli Al Faroby, “Identifikasi Jenis Kanker Darah (Leukemia) terhadap Pengaruh Parameter Kernel Support Vector Machine dan Ekstraksi Ciri dengan Rantai Markov Orde 2,” Surabaya, 2018.
- [7] G. V., G. C K, M. S. V. Kottamasu, dan N. P. Kumar, “The Voice Enabled Personal Assistant for Pc using Python,” *Int J Eng Adv Technol*, vol. 10, no. 4, hlm. 162–165, Apr 2021, doi: 10.35940/ijeat.d2425.0410421.
- [8] A. Fapal, T. Kanade, B. Janrao, M. Kamble, dan M. Raule, “Personal Virtual Assistant for Windows Using Python,” *www.irjmets.com @International Research Journal of Modernization in Engineering*, vol. 485, no. 07, hlm. 485–491, 2021, [Daring]. Available: www.irjmets.com

BIOGRAFI

Yulius Setyawan, Lulus dari Program Studi Teknik Informatika Universitas Buddhi Dharma pada tahun 2022. Saat ini tertarik dengan penelitian mengenai bahasa pemrograman Python dan Golang.

Yo Ceng Giap, Lulus dari Jurusan Teknologi Informasi STMIK Buddhi (S1) tahun 2003, Program Studi Teknik Informatika STMIK Eresha (S2) tahun 2010. Saat ini sebagai dosen tetap di Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Buddhi Dharma.