

**IMPLEMENTASI SPEECH RECOGNITION UNTUK ASISTEN  
VIRTUAL DENGAN PYTHON**

**SKRIPSI**



**TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS BUDDHI DHARMA**

**TANGERANG**

**2022**

# **IMPLEMENTASI SPEECH RECOGNITION UNTUK ASISTEN VIRTUAL DENGAN PYTHON**

## **SKRIPSI**

**Diajukan sebagai salah satu syarat untuk kelengkapan gelar kesarjanaan pada  
Program Studi Teknik Informatika  
Jenjang Pendidikan Strata 1**



**Disusun Oleh:**

**YULIUS SETYAWAN**

**20181000028**

**TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS BUDDHI DHARMA  
TANGERANG  
2022**

# LEMBAR PERSEMBAHAN

*"Education is the most powerful weapon you can use to change the world."*

(Nelson Mandela)

Dengan mengucap puji syukur kepada Tuhan Yang Maha Esa, Skripsi ini kupersembahkan untuk:

1. Bapak (Lie Siauw Khiong) dan Ibu (Hin Siu Pin) yang telah membesarkan aku dan selalu membimbing, mendukung, memotivasi, memberi apa yang terbaik bagiku serta selalu mendoakan aku untuk meraih suksesanku.
2. Kakak dan adikku yang telah memberikan dukungan semangat serta dorongan yang senantiasa diberikan.
3. Seluruh dosen Universitas Buddhi Dharma yang telah mengajarku, yang telah membimbing dan membagikan ilmunya untukku.
4. Teman-teman seperjuangan sejak PKK dan UKM BADMINTON UBD yang telah menjadi motivasi saya untuk terus melangkah maju, selalu memberi saya semangat, dukungan, dan rasa percaya diri terhadap keadaan kekurangan saya, serta mendorong saya untuk menggapai kebahagiaan saya.

# UNIVERSITAS BUDDHI DHARMA

## LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini,

NIM : 20181000028  
Nama : Yulius Setyawan  
Jenjang Studi : Strata 1  
Program Studi : Teknik Informatika  
Perminatan : Jaringan

Dengan ini saya menyatakan bahwa:

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Diploma/Sarjana) atau kelengkapan studi, baik di Universitas Buddhi Dharma maupun di Perguruan Tinggi lainnya.
2. Skripsi ini saya buat sendiri tanpa bantuan pihak lain, kecuali arahan dosen pembimbing.
3. Dalam Skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dan dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan daftar pustaka.
4. Dalam Skripsi ini tidak terdapat pemalsuan (kebohongan), seperti buku, artikel, jurnal, data sekunder, pengolahan data, dan pemalsuan tanda tangan dosen atau Ketua Program Studi di Universitas Buddhi Dharma yang dibuktikan dengan keasliannya.
5. Lembar Pernyataan ini saya buat dengan sesungguhnya, tanpa paksaan dan apabila dikemudian hari atau pada waktu lainnya terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah saya peroleh karena Skripsi ini, serta sanksi lainnya sesuai dengan peraturan dan norma yang berlaku.

Tangerang, 3 Agustus 2022

Penulis,



Yulius Setyawan

20181000028

# UNIVERSITAS BUDDHI DHARMA

## LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH

N I M : 20181000028  
Nama : Yulius Setyawan  
Jenjang Studi : Strata 1  
Program Studi : Teknik Informatika  
Peminatan : Jaringan

Dengan ini menyetujui untuk memberikan ijin kepada pihak Universitas Buddhi Dharma, Hak Bebas Royalti Non- Eksklusif (*Non-exclusive Royalti-Free Right*) atas karya ilmiah kami yang berjudul: “IMPLEMENTASI *SPEECH RECOGNITION* UNTUK ASISTEN VIRTUAL DENGAN PYTHON”, beserta perangkat yang diperlukan (apabila ada).

Dengan Hak Bebas Royalti Non-Eksklusif ini pihak Universitas Buddhi Dharma berhak menyimpan, mengalih-media atau format-kan, mengelolaannya dalam pangkalan data (*database*), mendistribusikannya, dan menampilkan atau mempublikasikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Buddhi Dharma, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Tangerang, 3 Agustus 2022

Penulis,



Yulius Setyawan

20181000028

**UNIVERSITAS BUDDHI DHARMA**

**LEMBAR PENGESAHAN PEMBIMBING**

**IMPLEMENTASI *SPEECH RECOGNITION* UNTUK  
ASISTEN VIRTUAL DENGAN PYTHON**

Dibuat Oleh:

NIM : 20181000028

Nama : Yulius Setyawan

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian  
Komprehensif

Program Studi Teknik Informatika

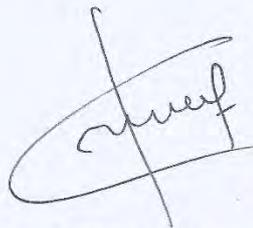
Peminatan Jaringan

Tahun Akademik 2021/2022

Disahkan oleh,

Tangerang, 3 Agustus 2022

**Pembimbing,**



Yo Ceng Giap, M.Kom., CPS

NIDN. 0412078003

# UNIVERSITAS BUDDHI DHARMA

## LEMBAR PENGESAHAN SKRIPSI

### IMPLEMENTASI *SPEECH RECOGNITION* UNTUK ASISTEN VIRTUAL DENGAN PYTHON

Dibuat Oleh:

NIM : 20181000028

Nama : Yulius Setyawan

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian  
Komprehensif

Program Studi Teknik Informatika

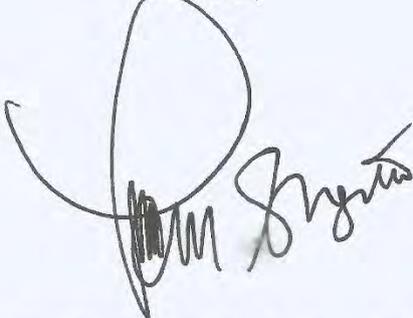
Peminatan Jaringan

Tahun Akademik 2021/2022

Disahkan oleh,

Tangerang, 3 Agustus 2022

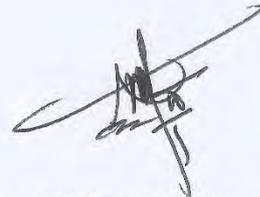
**Dekan,**



Dr. Eng. Ir. Amin Suyitno, M.Eng.

NIDK. 8826333420

**Ketua Program Studi,**



Hartana Wijaya, S.Kom., M.Kom.

NIDN. 0412058102

## LEMBAR PENGESAHAN TIM PENGUJI

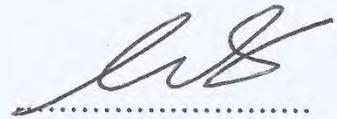
Nama : Yulius Setyawan  
NIM : 20181000028  
Fakultas : Sains dan Teknologi  
Judul Skripsi : **“IMPLEMENTASI *SPEECH RECOGNITION* UNTUK ASISTEN VIRTUAL DENGAN PYTHON”**

Dinyatakan LULUS setelah mempertahankan di depan Tim Penguji pada hari Rabu, 3 Agustus 2022.

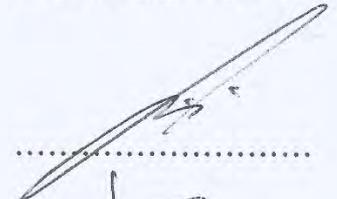
Nama penguji:

Tanda Tangan:

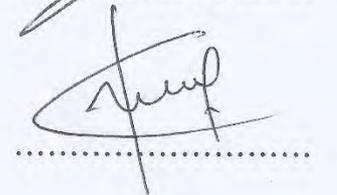
Ketua Sidang : **Edy, ST., M.Kom**  
NIDN. 0328128201



Penguji I : **Dram Renaldi, S.Kom., M.Kom**  
NIDN. 0411019001

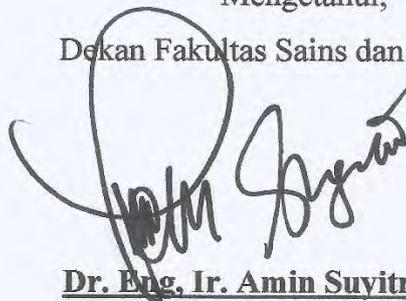


Penguji II : **Yo Ceng Giap, M.Kom., CPS**  
NIDN. 0412078003



Mengetahui,

Dekan Fakultas Sains dan Teknologi



**Dr. Eng, Ir. Amin Suyitno, M.Eng**

NIDK. 8826333420

## KATA PENGANTAR

Dengan mengucapkan Puji Syukur kepada Tuhan Yang Maha Esa, yang telah memberikan Rahmat dan karunia-Nya kepada penulis sehingga dapat menyusun dan menyelesaikan Skripsi ini dengan judul **Implementasi Speech Recognition untuk Asisten Virtual dengan Python**. Tujuan utama dari pembuatan Skripsi ini adalah sebagai salah satu syarat kelengkapan dalam menyelesaikan program pendidikan Strata 1 Program Studi Teknik Informatika di Universitas Buddhi Dharma. Dalam penyusunan Skripsi ini penulis banyak menerima bantuan dan dorongan baik moril maupun materiil dari berbagai pihak, maka pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. Suryadi Winata, SE, MM, M.Si., Ak., CA, CMA, CBV, ACPA, CPA (Aust.) sebagai Pelaksana Tugas Rektor Universitas Buddhi Dharma.
2. Bapak Dr. Eng, Ir. Amin Suyitno, M.Eng. sebagai Dekan Fakultas Sains dan Teknologi.
3. Bapak Rudy Ariyanto, S.Kom., M.Kom. sebagai Wakil Dekan Fakultas Sains dan Teknologi.
4. Bapak Hartana Wijaya, S.Kom., M.Kom. sebagai Ketua Program Studi Teknik Informatika.
5. Bapak Yo Ceng Giap, S.Kom., M.Kom., CPS. sebagai pembimbing yang telah membantu dan memberikan dukungan serta harapan untuk menyelesaikan penulisan Skripsi ini.
6. Orang tua dan keluarga yang selalu memberikan dukungan baik moril dan materiil.
7. Teman-teman yang selalu membantu dan memberikan semangat.

Serta semua pihak yang terlalu banyak untuk disebutkan satu-persatu sehingga terwujudnya penulisan ini. Penulis menyadari bahwa penulisan Skripsi ini masih belum sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan di masa yang akan datang.

Akhir kata semoga Skripsi ini dapat berguna bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Tangerang, 3 Agustus 2022

Penulis

## ABSTRAK

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 diimplemetasikan ke aplikasi asisten virtual pribadi berbasisan desktop dengan tujuan membantu menyederhanakan tugas sehari-hari melalui komputer. Asisten virtual ini juga terhubung dengan Arduino untuk mengoperasikan peralatan listrik yang terhubung. Metode yang digunakan di dalam penelitian ini bersifat eksperimental. Metode eksperimental dilakukan dengan cara memeriksa beragam studi literatur dan studi lapangan serta mempertimbangkan hasil dari eksperimen. Setelah melakukan eksperimen dan hasilnya memenuhi persyaratan, langkah selanjutnya adalah membuat perancangan dengan mengelompokkan berbagai eksperimen, sehingga tersusun sebuah sistem. Hasil penelitian ini menghasilkan sebuah aplikasi perintah 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.

Kata Kunci: *Arduino, Experimental, Python, Voice Command.*

## ***ABSTRACT***

*The most common way of communicating used by humans is by voice or speech. Many experiments were carried out in the aspect of creating sound to create a model mechanism that simulates how humans communicate verbally. Voice recognition technology aims to develop systems and techniques on how to introduce voice commands so that they can be understood by machines, so that machines are able to carry out commands given by humans. This study uses speech recognition technology to be implemented into a desktop-based personal virtual assistant application with the aim of helping daily tasks through a computer. This virtual assistant also connects with Arduino to operate connected electrical equipment. The method used in this research is experimental. The experimental method is carried out by examining various literature studies and field studies and considering the results of experiments. After conducting experiments and generating requirements, the next step is to make a design by grouping various experiments, so that a system is arranged. The output of this study resulted in a voice command application (speech recognition) to perform certain general tasks on desktop and electronic device applications. From the tests that have been carried out, the application able recognize voice commands to carry out the commands given and can use to turning on or off electrical devices as expected.*

*Keywords: Arduino, Experimental, Python, Voice Command*

# DAFTAR ISI

## LEMBAR JUDUL SKRIPSI

|   |       |
|---|-------|
| LEMBAR JUDUL DALAM SKRIPSI .....                | i     |
| LEMBAR PERSEMBAHAN .....                        | ii    |
| LEMBAR PERNYATAAN KEASLIAN SKRIPSI.....         | iii   |
| LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH ..... | iv    |
| LEMBAR PENGESAHAN PEMBIMBING.....               | v     |
| LEMBAR PENGESAHAN SKRIPSI.....                  | vi    |
| LEMBAR PENGESAHAN TIM PENGUJI .....             | vii   |
| KATA PENGANTAR.....                             | viii  |
| ABSTRAK .....                                   | ix    |
| ABSTRACT .....                                  | x     |
| DAFTAR ISI .....                                | xi    |
| DAFTAR GAMBAR.....                              | xiv   |
| DAFTAR TABEL .....                              | xvii  |
| DAFTAR LAMPIRAN .....                           | xviii |
| BAB I PENDAHULUAN .....                         | 1     |
| 1.1 Latar Belakang .....                        | 1     |
| 1.2 Identifikasi Masalah.....                   | 3     |
| 1.3 Rumusan Masalah.....                        | 3     |
| 1.4 Tujuan dan Manfaat Penulisan .....          | 4     |
| 1.4.1 Tujuan.....                               | 4     |
| 1.4.2 Manfaat.....                              | 4     |
| 1.5 Ruang Lingkup .....                         | 5     |
| 1.6 Metodologi Penelitian.....                  | 5     |
| 1.7 Sistematika Penulisan .....                 | 6     |

|  |    |
|--|----|
| BAB II LANDASAN TEORI.....                     | 8  |
| 2.1 Teori Umum.....                            | 8  |
| 2.1.1 Definisi Implementasi .....              | 8  |
| 2.1.2 Definisi Perancangan.....                | 9  |
| 2.1.3 Definisi <i>Speech Recognition</i> ..... | 9  |
| 2.1.4 Internet.....                            | 11 |
| 2.1.5 Relay.....                               | 13 |
| 2.1.6 Komputer Desktop.....                    | 14 |
| 2.2 Teori Khusus.....                          | 15 |
| 2.2.1 Python.....                              | 15 |
| 2.2.2 API.....                                 | 23 |
| 2.2.3 Kecerdasan Buatan (AI) .....             | 26 |
| 2.2.4 Asisten Pribadi Virtual .....            | 27 |
| 2.2.5 Mikrokontroler .....                     | 29 |
| 2.2.6 Arduino.....                             | 30 |
| 2.2.7 Visual Studio Code.....                  | 31 |
| 2.3 Teori Analisis dan Perancangan.....        | 32 |
| 2.3.1 <i>Black Box Testing</i> .....           | 32 |
| 2.3.2 <i>Firmata</i> .....                     | 34 |
| 2.3.3 <i>Flow Chart</i> .....                  | 35 |
| 2.4 Tinjauan Studi.....                        | 39 |
| 2.4.1 Penelitian .....                         | 39 |
| 2.4.2 Rangkuman Jurnal Penelitian .....        | 50 |
| 2.5 Kerangka Pemikiran .....                   | 56 |
| BAB III PERANCANGAN APLIKASI .....             | 57 |
| 3.1 Analisa Kebutuhan dan Fungsional .....     | 57 |

|   |   |     |
|---|---|-----|
| 3.1.1                                   | Analisis Kebutuhan .....                            | 57  |
| 3.1.2                                   | Analisis Fungsional .....                           | 59  |
| 3.2                                     | Perancangan .....                                   | 60  |
| 3.2.1                                   | Speech Processing .....                             | 61  |
| 3.2.2                                   | Pengambilan Data.....                               | 63  |
| 3.2.3                                   | Tampilan Antar Muka ( <i>User Interface</i> ) ..... | 64  |
| 3.2.4                                   | Pemrosesan Kueri .....                              | 66  |
| 3.2.5                                   | Perancangan Sistem dan Alat .....                   | 68  |
| BAB IV PENGUJIAN DAN IMPLEMENTASI ..... |   | 71  |
| 4.1                                     | Implementasi.....                                   | 71  |
| 4.1.1                                   | Implementasi Antar Muka .....                       | 71  |
| 4.1.2                                   | Spesifikasi Perangkat Lunak .....                   | 72  |
| 4.1.3                                   | Spesifikasi Perangkat Keras .....                   | 77  |
| 4.2                                     | Tampilan Aplikasi dan Alat.....                     | 89  |
| 4.2.1                                   | Tampilan Aplikasi .....                             | 89  |
| 4.2.2                                   | Tampilan Alat .....                                 | 94  |
| 4.3                                     | <i>Blackbox Testing</i> .....                       | 96  |
| 4.3.1                                   | Pengujian Fungsi Utama.....                         | 97  |
| 4.3.2                                   | Pengujian Fungsi Tambahan .....                     | 106 |
| 4.3.3                                   | Pengujian Alat .....                                | 113 |
| 4.4                                     | Pengolahan Data Kuesioner.....                      | 123 |
| BAB V KESIMPULAN DAN SARAN .....        |   | 130 |
| 5.1                                     | Kesimpulan .....                                    | 130 |
| 5.2                                     | Saran .....   | 131 |

DAFTAR PUSTAKA

LAMPIRAN

## DAFTAR GAMBAR

|  |    |
|--|----|
| Gambar 2.1 Transformasi Sinyal Suara menjadi Informasi .....       | 11 |
| Gambar 2.2 Relay .....   | 14 |
| Gambar 2.3 Contoh <i>Layout</i> Tkinter.....                       | 22 |
| Gambar 2.4 Ilustrasi Kerja API .....                               | 25 |
| Gambar 2.5 Arduino Uno .....                                       | 31 |
| Gambar 2.6 Kerangka Pemikiran .....                                | 56 |
| Gambar 3.1 Arsitektur Kerangka Kerja.....                          | 61 |
| Gambar 3.2 <i>Flow Chart Speech Processing</i> .....               | 62 |
| Gambar 3.3 Alur Pengambilan Data .....                             | 63 |
| Gambar 3.4 Rancangan Tampilan Antar Muka.....                      | 65 |
| Gambar 3.5 Flowchart Pemrosesan Kueri.....                         | 67 |
| Gambar 3.6 Flowchart Sistem .....                                  | 68 |
| Gambar 3.7 Skema Alat Arduino .....                                | 70 |
| Gambar 3.8 LCD 16x2 .....  | 82 |
| Gambar 4.1 Tampilan Antar Muka Aplikasi.....                       | 71 |
| Gambar 4.2 Arduino Uno R3 .....                                    | 78 |
| Gambar 4.3 Relay .....   | 79 |
| Gambar 4.4 Bentuk LED .....  | 80 |
| Gambar 4.5 Kabel Jumper .....                                      | 83 |
| Gambar 4.6 Kabel Serabut .....                                     | 85 |
| Gambar 4.7 Stop Kontak <i>In Bow</i> dan <i>Out Bow</i> .....      | 86 |
| Gambar 4.8 Fitting Lampu Langit.....                               | 87 |
| Gambar 4.9 Steker .....  | 88 |
| Gambar 4.10 Tampilan Form Login.....                               | 89 |
| Gambar 4.11 Implementasi Antar Muka Aplikasi.....                  | 90 |
| Gambar 4.12 Tampilan Aplikasi tanpa Koneksi .....                  | 91 |
| Gambar 4.13 Tampilan Aplikasi Setelah Diberikan Perintah .....     | 92 |
| Gambar 4.14 Tampilan Aplikasi Setelah diberikan Perintah Lain..... | 93 |
| Gambar 4.15 Fungsi LCD .....                                       | 94 |

|  |     |
|--|-----|
| Gambar 4.16 Implementasi Alat.....   | 95  |
| Gambar 4.17 Tampilan Ruangan dengan Alat dan Perangkat Elektronik dalam Keadaan Tidak Menyala..... | 95  |
| Gambar 4.18 Tampilan Ruangan dengan Semua Perangkat Elektronik yang Sedang Menyala .....           | 96  |
| Gambar 4.19 Pengujian Fungsi Cuaca. ....   | 97  |
| Gambar 4.20 Pengujian Tanggal dan Waktu.....   | 98  |
| Gambar 4.21 Form Jadwal Kegiatan. ....   | 98  |
| Gambar 4.22 Pengujian Menampilkan Jadwal Kegiatan.....   | 99  |
| Gambar 4.23 Pengujian Aplikasi Memutar Musik.....  | 99  |
| Gambar 4.24 Aplikasi berhasil Memutar Musik. ....  | 100 |
| Gambar 4.25 Form Pengisian Konten Email.....   | 100 |
| Gambar 4.26 Form Pengisian Alamat Email.....   | 101 |
| Gambar 4.27 Aplikasi Berhasil Mengirim Email.....  | 101 |
| Gambar 4.28 Form Topik Berita. ....  | 102 |
| Gambar 4.29 Tampilan Hasil Pengujian Membacakan Berita. ....                                       | 102 |
| Gambar 4.30 Pengujian Aplikasi Membuka Microsoft Word.....   | 103 |
| Gambar 4.31 Microsoft Word Berhasil Dibuka.....  | 103 |
| Gambar 4.32 Tampilan Pengujian Membuka Aplikasi Zoom.....  | 104 |
| Gambar 4.33 Aplikasi Zoom Berhasil Dibuka. ....  | 104 |
| Gambar 4. 34 Tampilan Aplikasi Setelah Diberi Perintah Penelusuran di Google.....                  | 105 |
| Gambar 4.35 Web Browser Berhasil Dibuka sesuai dengan Topik Pencarian. ....                        | 105 |
| Gambar 4.36 Tampilan Aplikasi setelah Suara Asisten Diubah. ....                                   | 106 |
| Gambar 4.37 Form Pencarian Topik Video di Youtube.....   | 107 |
| Gambar 4.38 Aplikasi Membuka Youtube sesuai Topik .....  | 107 |
| Gambar 4.39 Perintah untuk Mencari Info Tokoh di Wikipedia.....                                    | 108 |
| Gambar 4.40 Aplikasi Membacakan dan Menampilkan Informasi pada Tab <i>Com</i> .....                | 108 |
| Gambar 4.41 Tampilan Aplikasi saat Digunakan Mencari Lokasi Tempat .....                           | 109 |
| Gambar 4.42 Aplikasi Berhasil Memunculkan Lokasi Tempat .....                                      | 109 |
| Gambar 4.43 Form untuk Mengisi Kontak Penerima Pesan .....   | 110 |
| Gambar 4.44 Form Menulis Isi Pesan .....   | 110 |
| Gambar 4.45 Aplikasi Mengirim Pesan Whatsapp via Web.....  | 111 |
| Gambar 4.46 Aplikasi akan Mengingat Sesuatu yang Diucapkan Pengguna.....                           | 111 |

|   |     |
|---|-----|
| Gambar 4.47 Aplikasi Berhasil Mengingat dan Membacakan Ingatannya .....                         | 112 |
| Gambar 4.48 Aplikasi Membacakan Lelucon Ringan tentang Pemrograman. ....                        | 112 |
| Gambar 4.49 Tampilan Awal LCD .....   | 113 |
| Gambar 4.50 Contoh Tampilan LCD Setelah Diberi Perintah.....                                    | 113 |
| Gambar 4.51 Tampilan LCD saat Digunakan untuk Mengoperasikan Alat Elektronik....                | 114 |
| Gambar 4.52 Tampilan Aplikasi saat Menyalakan Lampu .....                                       | 114 |
| Gambar 4.53 Alat Berhasil Menyalakan Lampu.....   | 115 |
| Gambar 4.54 Tampilan Aplikasi saat Mematikan Lampu.....   | 115 |
| Gambar 4.55 Alat Berhasil Mematikan Lampu.....  | 116 |
| Gambar 4.56 Tampilan Aplikasi saat Menyalakan <i>Charger</i> .....                              | 116 |
| Gambar 4.57 Alat Berhasil Menyalakan Charger .....  | 117 |
| Gambar 4.58 Tampilan Aplikasi saat Mematikan <i>Charger</i> . ....                              | 117 |
| Gambar 4.59 Alat Berhasil Mematikan Charger .....   | 118 |
| Gambar 4.60 Tampilan Aplikasi ketika Kipas Dinyalakan.....                                      | 119 |
| Gambar 4.61 Alat Berhasil Menyalakan Kipas.....   | 119 |
| Gambar 4.62 Tampilan Aplikasi ketika Kipas Dimatikan .....                                      | 120 |
| Gambar 4.63 Alat Berhasil Mematikan Kipas.....  | 120 |
| Gambar 4.64 Tampilan Aplikasi saat Semua Perangkat Elektronik Dinyalakan secara Bersamaan ..... | 121 |
| Gambar 4.65 Alat Menyalakan Semua Perangkat Elektronik secara Bersamaan .....                   | 122 |
| Gambar 4.66 Tampilan Aplikasi saat Semua Perangkat Elektronik Dimatikan secara Bersamaan .....  | 122 |
| Gambar 4.67 Alat Mematikan Semua Perangkat Elektronik secara Bersamaan.....                     | 123 |
| Gambar 4.68 Grafik Hasil Kuesioner Pertanyaan 1 .....   | 124 |
| Gambar 4.69 Grafik Hasil Kuesioner Pertanyaan 2 .....   | 125 |
| Gambar 4.70 Grafik Hasil Kuesioner Pertanyaan 3 .....   | 126 |
| Gambar 4.71 Grafik Hasil Kuesioner Pertanyaan 4 .....   | 127 |
| Gambar 4.72 Grafik Hasil Kuesioner Pertanyaan 5 .....   | 128 |
| Gambar 4.73 Grafik Hasil Kuesioner dari 5 Pertanyaan.....                                       | 129 |

## DAFTAR TABEL

|   |     |
|---|-----|
| Tabel 2.1 Simbol <i>Flow Chart</i> .....  | 37  |
| Tabel 2.2 Jurnal I Komang Setia Buana .....   | 39  |
| Tabel 2.3 Jurnal Mohamad Ali Sadikin, dan Dedy Septono C. P .....   | 41  |
| Tabel 2. 4 Jurnal Dr.V. Geetha, Dr. C, K, Gomathy, Kottamasu Manasa Sri Vardhan, dan Nukala Pavan Kumar ..... | 44  |
| Tabel 2.5 Jurnal Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, dan Megha Raule .....         | 46  |
| Tabel 4.1 Daftar <i>Library</i> yang digunakan .....  | 73  |
| Tabel 4.2 Spesifikasi <i>Hardware</i> .....   | 77  |
| Tabel 4.3 Pin LCD 16x2 .....  | 82  |
| Tabel 4. 4 <i>Blackbox Testing</i> Fungsi Utama .....   | 97  |
| Tabel 4.5 <i>Blackbox Testing</i> Fitur Tambahan .....  | 106 |
| Tabel 4.6 <i>Blackbox Testing</i> Alat .....  | 113 |
| Tabel 4.7 Hasil Kuesioner Pertanyaan 1 .....  | 124 |
| Tabel 4.8 Hasil Kuesioner Pertanyaan 2 .....  | 125 |
| Tabel 4.9 Hasil Kuesioner Pertanyaan 3 .....  | 126 |
| Tabel 4.10 Hasil Kuesioner Pertanyaan 4 .....   | 127 |
| Tabel 4.11 Hasil Kuesioner Pertanyaan 5 .....   | 128 |

## DAFTAR LAMPIRAN

|  |    |
|--|----|
| Lampiran 1 Tabel Usia Responden .....            | L1 |
| Lampiran 2 Hasil Kuesioner Pertanyaan ke-1 ..... | L1 |
| Lampiran 3 Hasil Kuesioner Pertanyaan ke-2 ..... | L2 |
| Lampiran 4 Hasil Kuesioner Pertanyaan ke-3 ..... | L2 |
| Lampiran 5 Hasil Kuesioner Pertanyaan ke-4 ..... | L3 |
| Lampiran 6 Hasil Kuesioner Pertanyaan ke-5 ..... | L3 |
| Lampiran 7 Kuesioner Evaluasi Penelitian .....   | L4 |
| Lampiran 8 Tabel Dataset Kuesioner .....         | L5 |



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

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. Berbicara merupakan bentuk dasar dari komunikasi manusia dan pengenalan suara adalah suatu hal yang dipelajari dalam aspek komunikasi. 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.

*Speech recognition* adalah prosedur merekam ucapan menggunakan mikrofon atau telepon dan mengoversikan menjadi data yang berbentuk digital. Selain *speech recognition* ada juga sistem pengenalan suara lain yaitu *speaker recognition*. *Speaker recognition* adalah proses yang dilakukan oleh perangkat untuk mengetahui pembicara berdasarkan suaranya. Perbedaannya adalah *speech recognition* mengidentifikasi apa yang dikatakan, sedangkan *speaker recognition* mengenali siapa yang berbicara (Prayoga, 2019).

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. Ketika pengguna

menggunakan *Google voice*, spektogram suara akan dipecah lalu dikirim ke 8 buah server Google. Sebelum *Google voice* dapat mengidentifikasi perintah suara oleh pengguna, suara vokal dan konsonan akan terlebih dahulu diidentifikasi oleh sistem, kemudian informasi tersebut digunakan untuk menaksir perintah yang diprediksi menurut pola syaraf otak manusia.

Python adalah bahasa untuk melakukan penulisan kode di komputer yang dikenal handal dan mudah untuk dipahami karena penulisan kodenya memiliki kemiripan dengan bahasa manusia. Python pertama kali dikembangkan pada awal 1990-an oleh Guido Van Rossum, pemrogram asal Belanda di *Centrum Wiskunde and Informatica* (CWI). Berkat dukungan komunitas aktif di seluruh dunia, bahasa pemrograman Python dapat bertahan dan terus bertumbuh sampai saat ini. Banyak forum ataupun *blogger* berbagi pengalaman mereka ketika menggunakan Python. Hal ini memudahkan penggunaannya dari pemula sampai pengembang untuk saling berbagi dan bertukar ide mengenai bahasa pemrograman ini.

Semakin berkembangnya *Google voice* beriringan dengan bahasa pemrograman Python yang terus berkembang, penulis memiliki ide untuk membuat asisten suara virtual dengan Python berbasis desktop dan juga diimplementasikan ke mikrokontroler papan tunggal untuk mengoperasikan alat listrik dengan bantuan perintah suara. Mikrokontroler papan tunggal (*single board*) adalah perpaduan antara rangkaian papan tunggal bersama RAM (*Random Access Memory*), penyimpanan (*storage*) dan unit I/O (*input/output*). Mikrokontroler memanfaatkan chip mikro untuk otaknya. Arduino merupakan model mikrokontroler yang sangat terkenal dan banyak dipakai di seluruh dunia. Arduino populer karena terkenal mudah dipelajari dan banyak pemula yang mengenal elektronika dan robotik menggunakan Arduino.

Dengan dasar pemikiran tersebut, penulis terdorong untuk meneliti dan menyusun penelitian berjudul “**Implementasi Speech Recognition untuk Asisten Virtual dengan Python**”. Penulis memiliki ide memanfaatkan teknologi pengidentifikasi suara untuk membantu mempermudah pekerjaan dan menyalakan atau mematikan peralatan listrik yang terkoneksi Arduino dan *relay*, sehingga mengoperasikan peralatan bertenaga listrik menjadi lebih efisien dan praktis.

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang yang sudah diuraikan, ditemukan beberapa identifikasi masalah:

1. Mengetik dan menggerakkan kursor membutuhkan waktu yang lebih lama dibandingkan menggunakan perintah suara.
2. Membutuhkan aplikasi yang bisa menjalankan berbagai perintah dasar seperti layaknya asisten virtual.
3. Kesulitan untuk mematikan atau menyalakan peralatan elektronik apabila pemilik rumah mengalami perbatasan pergerakan, misalnya sedang sakit atau cidera.

## 1.3 Rumusan Masalah

Berdasarkan latar belakang masalah dan mengidentifikasi masalah yang ada, maka rumusan masalah yang ditemukan adalah sebagai berikut:

1. Bagaimana cara membuat aplikasi desktop yang dapat menjalankan berbagai tugas berdasarkan ucapan pengguna?
2. Bagaimana pengguna dapat mengetahui apakah aplikasi sukses menjalankan perintah yang diberikan?

3. Bagaimana menghubungkan alat elektronik dengan arduino agar dapat dikontrol melalui *speech recognition*?

## 1.4 Tujuan dan Manfaat Penulisan

### 1.4.1 Tujuan

Berdasarkan susunan riset yang dibuat, tujuan pembuatan aplikasi berikut adalah:

1. Merancang aplikasi asisten virtual yang dapat berkomunikasi dua arah dengan manusia.
2. Merancang program yang dapat menjalankan tugas-tugas sederhana di perangkat desktop berdasarkan perintah dari si pengguna.
3. Mempermudah mematikan dan menyalakan alat elektronik melalui perintah suara dengan bantuan aplikasi asisten virtual.

### 1.4.2 Manfaat

Manfaat yang dapat diperoleh dari penelitian pembuatan aplikasi ini adalah:

1. Menghemat waktu yang diperlukan untuk mengetik dan menggerakkan kursor ketika ingin menjalankan tugas sederhana pada perangkat desktop.
2. Dapat meminimalkan penggunaan perangkat input seperti keyboard, mouse, pena sentuh, dll. yang akan berdampak pada pengurangan biaya perangkat keras dan ruang yang dibutuhkan.
3. Dapat mematikan dan menyalakan alat elektronik tanpa membutuhkan banyak tenaga.

## 1.5 Ruang Lingkup

Penelitian hanya difokuskan dalam ruang lingkup sebagai berikut:

1. Pola bahasa yang digunakan adalah bahasa Inggris yang dibuat dari bentuk suara ke bentuk teks.
2. Aplikasi berjalan pada sistem operasi Windows 10.
3. Mikrokontroler yang digunakan pada penelitian ini adalah Arduino UNO R3.
4. Pengembangan aplikasi menggunakan bahasa pemrograman *Python* dengan versi minimal 3.8.5.
5. Suara yang diproses adalah suara pria dan wanita yang normal.

## 1.6 Metodologi Penelitian

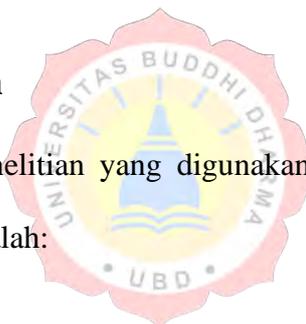
Tahapan metode penelitian yang digunakan untuk mengembangkan aplikasi *Python Voice Assistant* adalah:

### 1. Studi Literatur

Metode ini digunakan sebagai langkah pencarian sumber referensi ataupun teori yang memiliki kaitan dengan judul “IMPLEMENTASI *SPEECH RECOGNITION* UNTUK ASISTEN VIRTUAL DENGAN PYTHON”.

### 2. Perancangan Program

Bagian ini menentukan fungsi dan tugas dari program untuk menjalankan perintah yang sesuai dengan hasil yang dituju, perancangan ini dilakukan untuk menyusun atau membentuk struktur utama program memanfaatkan bahasa pemrograman Python.



### 3. Pengujian Program

Pengujian aplikasi dilaksanakan untuk membuktikan bahwasanya aplikasi yang dibangun dapat melakukan tugas berdasarkan fungsi yang sudah ditentukan sebelumnya.

### 4. Analisis

Metode terakhir adalah analisis dari program untuk memastikan bahwa hasil yang diperoleh cocok dengan tujuan yang diinginkan, serta mengubah atau menambahkan fitur agar dapat berjalan dengan semestinya dan tanpa kendala.

## 1.7 Sistematika Penulisan

Dalam skripsi ini menggunakan sistematika penulisan yang tersusun dari lima bab dengan detail sebagai berikut:

### BAB I Pendahuluan

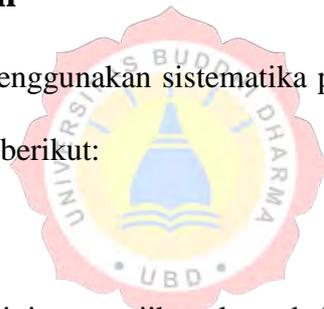
Bab pendahuluan ini menyajikan latar belakang, topik yang dikaji, manfaat & tujuan dari penulisan, batasan atau ruang lingkup, metodologi penelitian, dan prosedur penulisan sebagai gambaran umum dari penelitian.

### BAB II Landasan Teori

Landasan teori berisikan penggunaan konsep dan ide pendukung yang memiliki kaitan dengan penelitian, seperti *Speech Recognition*, asisten virtual, kecerdasan buatan, bahasa pemrograman Python, dan mikrokontroler Arduino, serta tinjauan studi dari beberapa jurnal pendukung, dan kerangka pemikiran.

### BAB III Perancangan Aplikasi

Bab ini memaparkan tentang analisis kebutuhan dan fungsional beserta penjelasan mengenai perancangan, prosedur dan langkah-langkah penyelesaian



masalah dalam merancang asisten virtual ke dalam aplikasi desktop yang dituangkan ke dalam suatu *flowchart*.

#### BAB IV Hasil dan Pembahasan

Bab ini akan menunjukkan hasil perencanaan aplikasi dan menampilkan skema alat. Selanjutnya dilakukan pengujian untuk memberikan gambaran kelebihan dan kekurangan hasil pembuatan aplikasi dan alat.

#### BAB V Kesimpulan dan Saran

Bab penutup yang berisikan kesimpulan atas program yang telah dibuat mampu menyelesaikan masalah penggunaan *Speech Recognition* untuk diterapkan pada prototipe asisten virtual dan memberikan gambaran tentang bagaimana program ini dapat dikembangkan untuk penelitian lebih lanjut.



## BAB II

### LANDASAN TEORI

#### 2.1 Teori Umum

##### 2.1.1 Definisi Implementasi

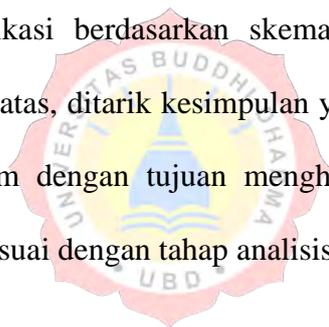
Secara etimologis implementasi berasal dari serapan penggunaan bahasa Inggris, yakni *to implement* bermakna mengimplementasikan. Implementasi adalah media dalam mewujudkan hal yang mampu membuat suatu efek atau hasil. Implementasi menurut Nurdin Usman, bermuara pada tindakan, aksi, maupun aktivitas kepada prosedur suatu sistem. Sedangkan menurut Guntur Setiawan implementasi adalah aktivitas yang diperluas dan bagaimana cara untuk mengoordinasikan harapan dengan aksi untuk menggapainya memerlukan pelaksanaan birokrasi yang mulus. (Saepulloh & Adeyadi, 2019)

Berdasarkan definisi dari Kamus Besar Bahasa Indonesia (KBBI), implementasi adalah pelaksanaan dan penerapan yang bermaksud untuk menelusuri bentuk mengenai hal yang ingin disepakati. Menurut Oxford *Advance Learner Dictionary* diuraikan implementasi berarti “*put something into effect*” (menerapkan hal-hal untuk menghasilkan pengaruh dan berdampak). Dari berbagai sudut pandang para ahli, pengertian implementasi adalah aktivitas yang direncanakan secara matang dan dilaksanakan dengan sungguh-sungguh dan berpedoman pada standar sampai memperoleh tujuan kegiatan.

### 2.1.2 Definisi Perancangan

Menurut (Setiyanto Rudi et al., 2019) perancangan merupakan tahapan pertama dalam membangun sebuah sistem. Perancangan adalah metode untuk menentukan apa yang hendak dikerjakan dengan berbagai macam proses yang menyertakan penjabaran tentang struktur beserta faktor dan juga keterikatan yang dijalani selama proses penyusunan. Berdasarkan pendapat Sommerville dari buku Agus Mulyanto (2009: 259) proses perancangan melibatkan peningkatan separuh sistem dan model dalam tahapan abstraksi beragam.

Keuntungan dari tahap perancangan sistem adalah menyediakan gambaran cetak biru yang lengkap sebagai panduan bagi tim pemrogram untuk mengembangkan aplikasi berdasarkan skema sistem yang terkomputerisasi. Berdasarkan uraian diatas, ditarik kesimpulan yaitu perancangan adalah tahapan selepas analisis sistem dengan tujuan menghasilkan konsep yang menjawab persyaratan standar sesuai dengan tahap analisis.

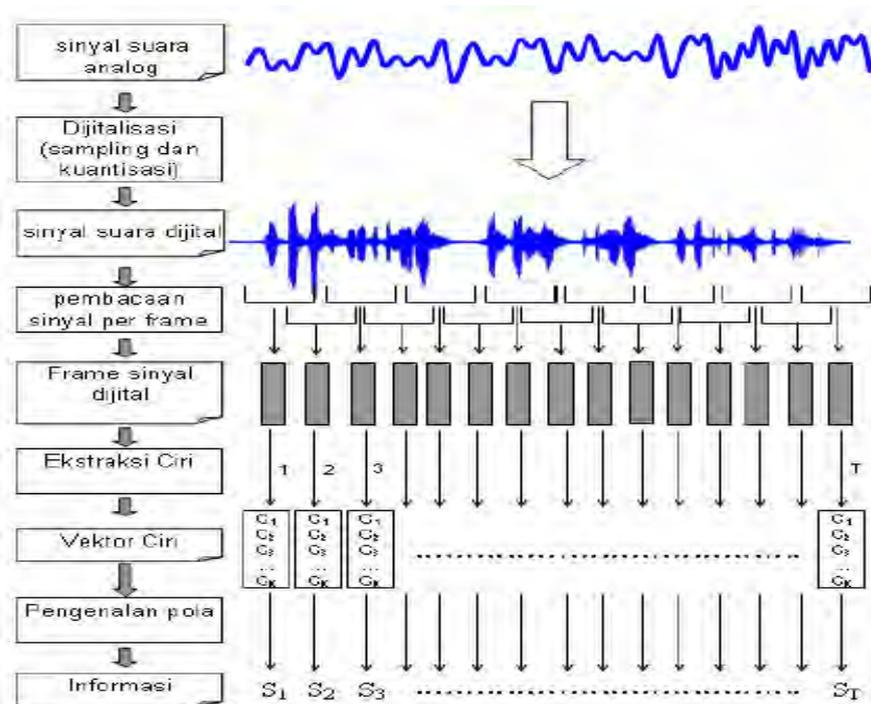


### 2.1.3 Definisi *Speech Recognition*

. *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. Kriteria yang diukur adalah durasi pengucapan dan tingkat tekanan suara terutama suara vokal dan selanjutnya parameter tersebut akan dibandingkan dengan basis data yang sudah tersedia.

Menurut (Andriana et al., 2016, p. 2) secara umum *speech recognition* memproses cara seseorang mengucapkan kata, bagaimana penekanan pada vokal tertentu, dan menyimpan data suara yang dikonversi menjadi data digital. Hasil selanjutnya diubah menjadi gelombang suara untuk dianalisis dan dibandingkan dengan sampel suara yang ada di *database* sistem. Sistem pengenalan pola suara menggunakan mikrofon untuk memasukkan data suara, perangkat lunak untuk mengidentifikasi pola suara orang, dan kartu suara (*soundcard*) untuk *input* dan *output*.

Dikutip dari web [www.realPython.com](http://www.realPython.com) (Amos, 2018) pengenalan suara dimulai saat Bell Labs yang melakukan penelitian di tahun 1950-an. Pengenalan suara yang dikenal saat ini terinspirasi dari zaman kuno. Manusia zaman kuno dapat mengenali ucapan dari beberapa pembicara dan memiliki kosakata yang banyak sekali dalam berbagai bahasa. Kebanyakan sistem pengenalan suara saat ini memanfaatkan pendekatan *Hidden Markov Model* (HMM). Pendekatan HMM berfungsi berdasarkan hipotesis jika sinyal ucapan terlihat dalam periode waktu yang sebentar (misalnya, lima belas milidetik) bisa diprediksi dengan normal seperti metode stasioner yaitu, metode yang sifat statistiknya tidak akan berubah seiring waktu.



**Gambar 2.1 Transformasi Sinyal Suara menjadi Informasi**

(Sumber: Aisuwarya et al., 2017, p. 67)



### 2.1.4 Internet

Menurut (Babbar, 2017, p. 39) Internet (*Interconnection-Networking*) pada dasarnya adalah jaringan komputer yang bersifat global dan terkoneksi satu sama lain melalui protokol TCP/IP untuk meladeni miliaran penggunanya yang berada di berbagai belahan benua. Komputer yang awalnya berdiri sendiri-sendiri (*stand-alone*) saat ini berkomunikasi secara langsung satu sama lain dengan *host* komputer yang berbeda di seluruh dunia.

Dilansir dari web [www.windowssku.com](http://www.windowssku.com) (Hernawan, 2021) cara kerja internet secara singkat akan dijelaskan sebagai berikut:

1. Terkoneksi dengan Modem/Router

Komputer atau ponsel pintar dapat terhubung ke jaringan internet dengan bantuan perangkat *modulator-demodulator* dan *router*. Secara serentak, perangkat ini tersambung menuju jaringan di tempat lainnya. Router menyambungkan komputer ke jaringan dan modem bertugas untuk mengarahkan ke ISP atau *Internet Service Provider* melalui medium kabel. ISP kemudian menyediakan akses internet.

## 2. Mengetik Alamat Website

Disaat pengguna mengetikkan alamat domain situs web di peramban (*browser*), selanjutnya alamat akan diteruskan oleh ISP dan ISP juga terhubung ke DNS (*Domain Name Server*). Kemudian peramban memeriksa alamat IP berdasarkan alamat yang diketik lewat DNS. DNS dapat dianalogikan layaknya buku yang berisi kumpulan situs dan alamatnya, contohnya “facebook.com” dan dilengkapi angka yang terhubung dengan alamat IP. Contohnya 157.240.217.35 yang merupakan alamat IP dari facebook. Setiap situs memiliki alamat IP yang beragam dan unik. Sebagai informasi, keberadaan lokasi server dapat berdampak pada kecepatan akses suatu situs.

## 3. *Hypertext Transfer Protocol* (HTTP)

Peramban kemudian menugaskan HTTP *request* ke peladen (*server*) target dengan protokol TCP/IP. Apabila tidak ada kendala, peladen target meneruskan tanda dengan kode “200 OK”. Kemudian seluruh elemen berupa teks, gambar, maupun video dikirimkan sebagai data.

## 4. *Browser* Menampilkan Data

Data yang didapat dari peladen digunakan peramban untuk menampilkan semua informasi tersebut agar kita bisa mengakses dan

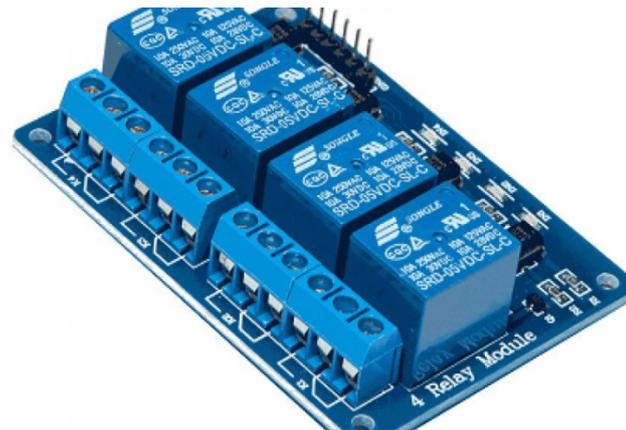
melihat web yang dikunjungi. Untuk bisa mengakses internet diperlukan protokol. Protokol bisa dikatakan sebagai aturan. Contoh dari protokol adalah HTTP (*HyperText Transfer Protocol*). HTTP berfungsi untuk melihat tampilan web di peramban. Jadi apabila menyambangi suatu situs, akan menampilkan alamat "*http*" di depannya.

Selain HTTP, diperlukan protokol internet lain yaitu TCP/IP (*Transmission Control Protocol IP/ Internet Protocol*) yang berperan penting dalam memastikan bahwa setiap paket data yang dikirim bisa sampai ke server tujuan tanpa ada kehilangan. Apabila ada data yang tidak terkirim atau lenyap, protokol TCP akan terus mengirim kembali data tersebut.

### 2.1.5 Relay



Relay merupakan suatu benda elektronika berbentuk sakelar yang dinyalakan melalui aliran listrik. Pada dasarnya, relay adalah penyungkit sakelar berupa kawat yang sekelilingnya dililit ke gagang metal (solenoid). Saat solenoid dilewati arus listrik, penyungkit bakal tertarik disebabkan oleh gaya tarik magnet pada gagang besi mengakibatkan sakelar tertutup. Ketika aliran listrik berhenti, kontak sakelar kembali terbuka dan penyungkit balik ke posisi awal karena gaya magnet yang menghilang. Relay secara umum berfungsi mengendalikan aliran/ tekanan listrik yang kuat (contohnya peranti elektronik 4A / AC 220 V) atau menggunakan aliran/ tekanan yang rendah (0.1 A / 12 Volt DC).



**Gambar 2.2 Relay**

(Sumber: <https://wasiswa.com/relay/>)

### 2.1.6 Komputer Desktop

Menurut (Shodikin, 2020) Komputer desktop adalah semacam personal komputer dan dibuat untuk penggunaan yang umum. Komputer desktop diletakkan di suatu tempat sendiri atau di meja yang sesuai kebutuhan ukuran dan kapasitasnya. Komputer Desktop terdiri dari perangkat keras (*hardware*) penyusunnya. Perangkat keras tersebut diantaranya:

1. Papan ketik (*Keyboard*).
2. Papan sirkuit utama (*Motherboard*).
3. Monitor.
4. Kartu *Video Graphics Array* (VGA).
5. *Random Access Memory* (RAM).
6. Sumber daya listrik (*Power Supply*).
7. Tetikus.
8. CPU (*Central Processing Unit*).

9. Penyimpanan data berupa HDD (*Harddisk Drive*) atau SSD (*Solid State Drive*).

## 2.2 Teori Khusus

### 2.2.1 Python

#### 2.2.1.1 Sejarah Python

Python dikenal sebagai bahasa pemrograman serbaguna yang diciptakan Guido Van Rossum, pemrogram asal Amsterdam, Belanda pada tahun 1990 di CWI (*Centrum Wiskunde & Informatica*), yang ditujukan sebagai penerus dari bahasa pemrograman ABC. CWI merilis versi terakhir 1.2. Pada tahun 1995, Guido pergi ke CNRI di Virginia Amerika Serikat sembari melanjutkan menyempurnakan Python. Guido bersama dengan pemrogram lain bekerja sama dengan BeOpen.com, sebuah industri komersial dan membangun BeOpen PythonLabs. BeOpen akhirnya merilis Python versi 2.0. Sesudah Python 2.0 dirilis, Guido dan anggota pemrogram lain di PythonLabs bertolak ke DigitalCreations. Hingga kini Python terus dibesarkan dan dijalankan tim pemrogram yang dipimpin oleh Guido bersama lembaga Python *Software Foundation* (Setia Buana, 2018).

Menurut (Hamim Zajuli Al Faroby, 2018, p. 30) Python *Software Foundation* adalah badan nirlaba yang telah didirikan sebagai pemilik hak kekayaan intelektual Python sejak versi 2.1 dan dengan begitu melarang Python dibeli oleh institusi komersial. Ketika penelitian ini ditulis, Python telah berada di terbitan 3.9.0. Istilah Python digunakan

oleh Guido untuk sebutan bahasa pemrograman buatannya sebab terinspirasi dari program acara kesukaanya bernama *Monty Python's Flying Circus*. Python dikenal sebagai bahasa yang mengkombinasikan kemampuan, kecakapan, dilengkapi sintaks kode yang mudah, serta ditambah fungsionalitas pustaka standar yang lengkap.

### 2.2.1.2 Manfaat Python

Berikut beberapa manfaat Python yang dapat ditelaah:

#### 1. Mengembangkan Website

Dalam merintis dunia bisnis, memiliki website adalah bagian penting. Selain memberikan profil bisnis kepada calon pelanggan, website juga memperlihatkan kualitas dan reputasi bisnis. Tidak sedikit bermacam situs web perusahaan diberikan beraneka ragam fitur agar calon pelanggan makin tertarik dengan bisnis yang sedang dikelola. Python dapat dimanfaatkan untuk meningkatkan laman website agar lebih intuitif, menarik, dan prosesnya juga menjadi lebih mudah

#### 2. Pengembangan IoT

Pentingnya penggunaan Python selain untuk pengembangan website adalah pengembangan *Internet of Things* (IoT). *Internet of Things* adalah sebuah sistem yang memungkinkan objek berbeda atau perangkat yang berbeda untuk saling berkomunikasi melalui jaringan internet. Python dipilih untuk mengembangkan IoT karena dikenal akan fleksibilitas dan kemudahannya.

### 3. Penambangan Data

Fungsi lain dari Python adalah manajemen dan merapikan data. Python dipandang sebagai bahasa pemrograman yang handal pada bidang ini. Selain itu, Python digunakan untuk *machine learning* dan memudahkan penguraian data dengan bantuan algoritma.

### 4. Pengembangan *Machine Learning*

Masih berhubungan mengenai penambangan data, *machine learning* membutuhkan beraneka ragam jenis data yang dimasukkan. Data berbagai jenis selanjutnya diproses dan menjelma ke tindakan apa yang dijalankan mesin. Dalam operasinya, bahasa pemrograman Python lah yang digunakan untuk menyederhanakan aktivitas ini.

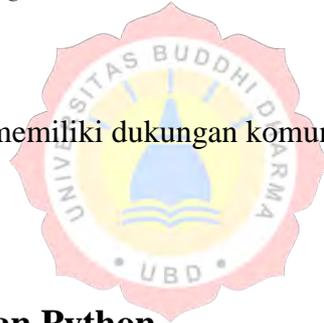
### 5. *Big Data* dan Otomasi

Python adalah bahasa pemrograman yang mendukung manipulasi *Big Data*. Berkat kehandalannya, Python mampu menangani berbagai jenis informasi. Selain itu, Python juga menunjang komputasi paralel. Salah satunya adalah penggunaan Python dalam *Big Data*, yang membuatnya lebih mudah untuk digunakan. Python juga memiliki peran penting dalam otomatisasi karena dapat dimanfaatkan sebagai bahasa skrip.

## 2.2.1.3 Kelebihan Python

Berikut beberapa kelebihan Python sebagai bahasa pemrograman (Khoirom et al., 2020):

1. Python mudah dipelajari, dibaca dan ditulis. Menjadikan Python bahasa pemrograman yang ramah bagi pemula.
2. Python gratis dan bersifat *open-source*. Python secara luas digunakan untuk berbagai fungsi.
3. Python menyediakan beberapa pola pemrograman, contohnya OOP (*Object Oriented Programming*), pemrograman imperative dan fungsional, pemrograman prosedural, dan juga pemrograman terstruktur.
4. Python adalah bahasa pemrograman *interpreted*. Kode tidak perlu dikompilasi, kode akan dieksekusi baris demi baris, membuat *debugging* kode lebih mudah dibandingkan bahasa pemrograman lainnya.
5. Python memiliki dukungan komunitas yang luas.



#### **2.2.1.4 Kekurangan Python**

Selain memiliki kelebihan, Python juga memiliki kekurangan dibanding bahasa pemrograman lainnya, seperti:

1. Karena Python adalah bahasa pemrograman *interpreted*, eksekusinya lebih lambat daripada bahasa pemrograman lainnya.
2. Python tidak menguntungkan untuk pengembangan aplikasi seluler.
3. Kesederhanaan Python menjadi kelemahan ketika programmer ingin beralih ke bahasa lain seperti Java yang memiliki struktur ketat.

### 2.2.1.5 Python *Library*

Menurut situs [www.dqlab.id](http://www.dqlab.id), *library* Python adalah kode program tambahan yang digunakan untuk maksud dan tujuan khusus. Python memiliki > 140 ribu *library* yang terus ditingkatkan lewat proyek sumber terbuka (*open source*). Penggunaan *library* pada Python membantu mencetak kode yang efektif dan mempersingkat waktu tanpa harus menetik kembali segala skrip. *Library* pada Python sifatnya *reusable*, yang berarti bisa dipergunakan kembali kapanpun saat dibutuhkan. (*Library Python Kenali Perbedaan Module, Package, Dan Library Pada Python*, 2021).

Contoh *library* Python yang umum digunakan diantaranya:

#### 1. Pyttsx3

Pyttsx3 adalah *library* untuk konversi *text-to-speech* (sistem yang mengubah teks menjadi sebuah ucapan). Tidak serupa *library* substitusi, pyttsx3 bekerja secara luring (*offline*) dan cocok digunakan oleh Python versi 2 maupun 3. Aplikasi memanggil fungsi bawaan `pyttsx3.init()` untuk mendapatkan referensi ke pyttsx3. Contoh penggunaan *library* pyttsx3 seperti berikut:

```
import pyttsx3

engine = pyttsx3.init()

engine.runAndWait()
```

#### 2. Pandas

Pandas adalah *library* yang didasarkan pada sistem *dataframe* dan mampu memproses dokumen ke bentuk tabel secara virtual seperti halnya aplikasi pengolah angka (*spreadsheet*). Pandas juga

berfungsi mengolah suatu data seperti teknik *join*, *distinct*, *group by*, agregasi, dan teknik lainnya seperti pada SQL. Perbedaannya adalah *pandas* melakukannya pada tabel. Keuntungan dari *library* ini juga dapat membaca fail dari berbagai format seperti *.txt*, *.csv*, dan *tsv*. Contoh penggunaan *library Pandas* seperti berikut,

```
import pandas as pd

d={'Nama':pd.Series(['Alwick','Randy','Willy','Pearl','Dustin','Graham','Jason']),'Tahun
Pengalaman':pd.Series([7,4,2,1,3,5,2]),'Bahasa
Pemrograman':pd.Series(['Python','Golang','PHP','C#','Java
Script','Scala','React'])}

df = pd.DataFrame(d)
```

**Output:**

|   | Nama   | Bahasa Pemrograman | Tahun Pengalaman |
|---|--------|--------------------|------------------|
| 0 | Alwick | Python             | 7                |
| 1 | Randy  | Golang             | 4                |
| 2 | Willy  | PHP                | 2                |
| 3 | Pearl  | C#                 | 1                |
| 4 | Dustin | Java Scirpt        | 3                |
| 5 | Graham | Scala              | 5                |
| 6 | Jason  | React              | 2                |

### 3. OS

Tujuan utama dari *library OS* adalah untuk berinteraksi dengan sistem operasi pada komputer. Penggunaan utama OS adalah

untuk membuat folder, menghapus folder, memindahkan folder, dan terkadang mengubah direktori kerja. OS juga dapat mengakses nama fail dalam jalur fail dengan melakukan *listdir()*. Contoh dari pemanfaatan OS *library*:

```
import os

cwd = os.getcwd()

print("Direktori saat ini:", cwd)
```

**Output:**

*Direktori saat ini: /home/yulius/Desktop/Downloads*

#### 4. Tkinter

Tkinter merupakan salah satu GUI (*Graphical User Interface*) yang dapat mempermudah pembuatan program berbasis grafis. Modul tkinter merupakan standar GUI *library* yang ditautkan dengan Python dengan tujuan untuk menyediakan cara cepat dan mudah untuk membangun aplikasi dengan tampilan yang membuat pengguna bisa melakukan interaksi lewat gambar dan ikon yang ada. Tkinter berfungsi sebagai penyusun *user interface* pada aplikasi yang akan dibuat untuk memudahkan para pengguna dalam mengakses aplikasi *Python Voice Assistant*.

Contoh penggunaan library tkinter:

```
import tkinter as tk

window = tk.Tk()

window.title("PythonVA's")
```

```
window.geometry("250x250")
```

```
window.resizable(0, 0)
```

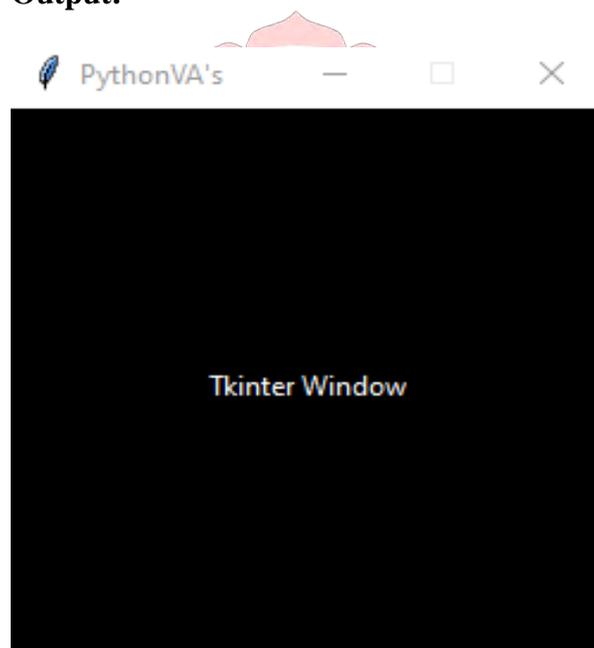
```
window.configure(bg="black")
```

```
label = tk.Label(text="Tkinter Window", bg="black", fg="white",  
width=50, height=50)
```

```
label.pack()
```

```
window.mainloop()
```

**Output:**



**Gambar 2.3 Contoh *Layout* Tkinter**

## 2.2.2 API

### 2.2.2.1 Pengertian API

API merupakan kependekan dari *Application Programming Interface*. Menurut (Gite, 2017) API adalah seperangkat prosedur, fungsi, protokol, dan aturan yang menentukan bagaimana komponen perangkat lunak yang berbeda dapat berinteraksi satu sama lain. API seperti perantara antara dua orang yang berinteraksi satu sama lain menggunakan bahasa yang sama sekali berbeda. Dalam perangkat lunak, data dapat didistribusikan melalui API sehingga dua paket perangkat lunak bisa saling berkomunikasi.

### 2.2.2.2 Jenis API

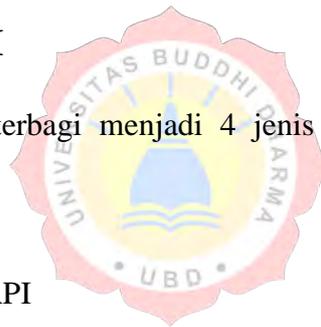
API terbagi menjadi 4 jenis yang digolongkan menurut hak aksesnya:

#### 1. *Public* API

*Public* API dikenal juga dengan API terbuka. API public adalah jenis API yang bisa diakses tanpa membutuhkan akses khusus dan bisa digunakan di berbagai sistem operasi. Penggunaan API terbuka cukup mudah hanya dengan mendaftar dan bisa langsung digunakan. Contoh *Public* API adalah API untuk memantau kondisi cuaca (*Open Weather API*) dan API yang memperbarui berita terkini (*News API*).

#### 2. *Private* API

*Private* API adalah jenis API yang tertutup untuk pemakaian publik. Fungsi *Private* API adalah memenuhi kebutuhan tim internal



dalam mengembangkan aplikasi tertentu. Contohnya, API dari *back end* berguna mengirim informasi ke *front end* lewat website.

### 3. *Partner API*

*Partner API* bisa dimanfaatkan oleh khalayak publik tapi memiliki akses terbatas hanya kepada pengguna yang sudah memiliki izin. Pengguna *Partner API* diharuskan mendaftar ke pemilik layanan API dan menyetujui ketentuan yang berlaku. Penggunaanya juga terbatas di aplikasi tertentu sesuai dengan ketentuan. Contohnya, API Pinterest.

### 4. *Composite API*

*Composite API* adalah API yang menjaga data dari peladen yang berbeda di suatu lokasi. Tujuannya adalah untuk mempersingkat waktu yang dibutuhkan oleh pengakses API karena pengakses dapat memperoleh beragam model data dari satu kali kunjungan.

## 2.2.2.3 Cara Kerja API

API bekerja dengan 4 tahapan berikut:

### 1. Aplikasi Mengakses API

Bagian pertama dari cara kerja API adalah saat pengguna mengakses aplikasi. Misalnya dalam aplikasi travel, ketika pengguna mau membeli tiket kereta api, aplikasi tersebut akan meneruskan ke API stasiun terkait.

### 2. API Meminta *Request* ke Server

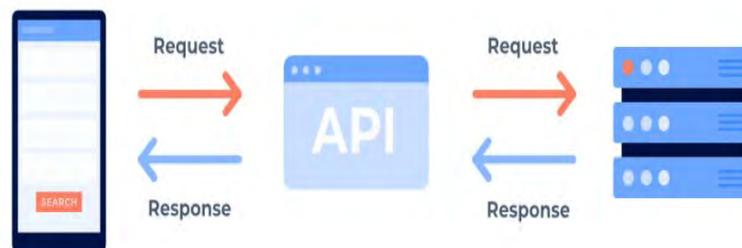
Sesudah aplikasi dinyatakan sukses mendapat alamat API, permintaan data kemudian dilanjutkan ke peladen stasiun kereta. API akan memperlihatkan data ke aplikasi travel yang memerlukan data perjalanan sesuai tujuan dan tanggal yang diinginkan.

### 3. Server Memberi Respon ke API

Saat mendapatkan informasi yang cocok dengan kebutuhan, peladen selanjutnya mengontak API. Informasi ini berisikan pukul keberangkatan, sisa ketersediaan kursi, daftar stasiun, dan lain-lain.

### 4. API Meneruskan Jawaban menuju Aplikasi

Langkah terakhir, API menyampaikan keterangan dari peladen menuju aplikasi perjalanan dari stasiun tersebut. Metode ini sejalan untuk permintaan ke stasiun kereta api lain.



**Gambar 2.4 Ilustrasi Kerja API**

(Sumber: [https://www.niagahoster.co.id/blog/wp-content/uploads/2020/11/Ilustrasi-Mengenal-API\\_Pengertian-Fungsi-dan-Cara-Kerjanya-1024x345.jpg](https://www.niagahoster.co.id/blog/wp-content/uploads/2020/11/Ilustrasi-Mengenal-API_Pengertian-Fungsi-dan-Cara-Kerjanya-1024x345.jpg))

### 2.2.3 Kecerdasan Buatan (AI)

Menurut (Rachmad & Widyastuti, 2020, p. 225) kecerdasan buatan didasarkan pada istilah *Artificial Intelligence*, yaitu bermakna tiruan atau kecerdasan. Kecerdasan buatan merupakan suatu disiplin bidang yang mempelajari tentang bidang sains dan komputer yang memberikan kecerdasan pada komputer, dan kecerdasan tersebut digunakan untuk membantu menyelesaikan tugas dan aktivitas manusia

Kecerdasan buatan ditafsirkan bermacam-macam menurut pendapat para ahli akan tetapi terdapat keseragaman makna, Menurut para ahli, definisi dari AI adalah:

- a. H.A Simon (1987) berpendapat bahwa kecerdasan buatan seumpama lingkup analisis, aplikasi yang berhubungan dengan sejumlah perintah yang dimaksudkan dengan fungsi dan tujuan yang baik bagi manusia dikombinasikan bantuan bahasa pemrograman pada komputer atau disebut cerdas.
- b. Menurut Rick and Knight memaparkan bahwa kecerdasan buatan (*Artificial Intelligence*) adalah ilmu mengenai cara mewujudkan komputer menunaikan pekerjaan yang umumnya dilaksanakan oleh manusia agar bertambah sempurna. Bagaimana menciptakan mesin menjadi lebih pandai adalah tujuan utama dari ilmu kecerdasan buatan.

Menurut (Putri & Pratama, 2017, p. 199) istilah Kecerdasan Buatan “*Artificial Intelligence*” diperkenalkan di konferensi *Darhmouth* pada tahun 1956. Para pakar dan ahli telah mencoba mendalami kecerdasan buatan hasil pemikiran manusia selama bertahun-tahun. Berdasarkan gagasan itulah muncul

AI sebagai bidang studi yang berupaya mendalami dan mencontoh kecerdasan manusia. Semenjak itu kelompok ilmuwan mendiskusikan kemajuan AI yang kemudian filosofi dan prinsipnya terus bertumbuh sampai hari ini.

Di era sekarang kebutuhan kecerdasan buatan terus meningkat ke berbagai bidang modern. Seperti diantaranya industri pabrik, perbankan, perjalanan, media konvensional atau elektronik, rumah sakit, dan industri hiburan. Menurut Stuart J. Russel & Peter Norvig, kecerdasan buatan seperti layaknya komputer yang mempunyai kapabilitas untuk mempelajari lingkungan dan bereaksi seharusnya. Prendergast dan Winston (1984) berpendapat kecerdasan buatan dibuat untuk memenuhi 3 tujuan tertentu:

- a. Menjadikan mesin pintar yang berarti mesin mampu memperkirakan, menjumlahkan dan menuntaskan perintah yang pengguna berikan.
- b. Tujuan dibuatnya kecerdasan buatan adalah mengerti tentang kewajiban dan bisa menuntaskan pekerjaan, serta cepat, efektif, dan teliti.
- c. Mesin yang berguna, mesin bekerja sesuai ketelitian pemrosesan untuk mencapai hasil paling maksimum.

#### **2.2.4 Asisten Pribadi Virtual**

Menurut (McTear & Callejas, 2013) istilah asisten pribadi virtual (APV) merupakan 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.

(McTear & Callejas, 2013) menjelaskan tiga buah keunikan APV, diantaranya:

### 1. Sistem Percakapan

Sistem ini dijalankan berdasarkan pandangan untuk meningkatkan sistem yang mampu berinteraksi dengan manusia dalam bahasa yang natural (dahulu menggunakan teks dan berkembang secara linguistik). Ketika sistem pertama kali dibuat dan langsung ditujukan untuk hal yang berkaitan dengan mendapat informasi, misalnya, memperoleh informasi harga saham atau penerbangan. Generasi selanjutnya mengizinkan manusia bisa terlibat ke berbagai wujud transaksi, menjalankan kegiatan keuangan atau kegiatan perjalanan ekspedisi. Selain itu, AI terus diperbarui untuk mengerjakan pekerjaan lainnya, seperti memandu manusia yang mendapati kesusahan mengonfigurasi berbagai perangkat.

### 2. *Voice User Interface* (VUIs)

VUIs serupa sistem dialog namun lebih fokus ke penggunaan yang bersifat profitabel. Fokusnya adalah lebih ke hal khusus, contohnya perutean undangan, pertolongan pemisahan kalalog, dan dialog transaksi seperti perjalanan, motel, navigasi, sewa kendaraan, dan saldo bank. Secara umum VUIs dibuat berbasis *VoiceXML*, bahasa komputer yang menggunakan elemen dan tag berbasis XML. Skrip *VoiceXML* selanjutnya diterjemahkan oleh peramban suara yang memiliki kemampuan mengeluarkan suara.

### 3. *Chatbots*

*Chatbots* atau bot obrolan biasa dipakai untuk meniru dialog antar individu. *Chatbots* awalnya berupa program ELIZA yang dibuat oleh Joseph

Weizenbaum pada tahun 1960-an. Program ini secara menakutkan mampu mereplikasi psikoterapis Rogerian. Belakangan ini *chatbots* banyak digunakan di berbagai bidang, seperti bisnis, pendidikan, jual beli di *marketplace*, dan melayani pertanyaan pengguna. *Chatbots* memakai proses pencocokan pola untuk menyesuaikan masukan perintah dan mendapatkan respon sesuai dan tepat. Biasanya *chatbots* berdasarkan bentuk teks, tetapi belakangan ini *chatbot* berbasis suara semakin bermunculan.

### 2.2.5 Mikrokontroler

Menurut (Sanjaya & Karna, 2021) Mikrokontroler adalah sebuah komputer dalam chip yang dapat diprogram dengan kode untuk melakukan tugas atau operasi tertentu, mikrokontroler berasal dari dua kata yaitu mikro berarti kecil dan *controller* yang berarti perangkat yang dapat mengontrol suatu objek, proses, atau peristiwa. Mikrokontroler tidak hanya sekadar berbentuk mikroprosesor sebab didalamnya ada terpasang ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), port *input & output*, dan beberapa perangkat *timer*, ADC (konverter sinyal analog ke digital), DAC (konverter sinyal digital ke analog) dan komunikasi bersambungan.

Keunggulan dari mikrokontroler yaitu tersedia perangkat I/O dan memori baca/ tulis yang mendukung, dan berukuran kecil atau sederhana. Sistem penyimpanan pada mikrokontroler adalah CMOS 8-bit dilengkapi 4 KB *Flash PEROM* yang dapat ditulis dan dihapus sekitar 1000 kali (Novitasari, 2018)

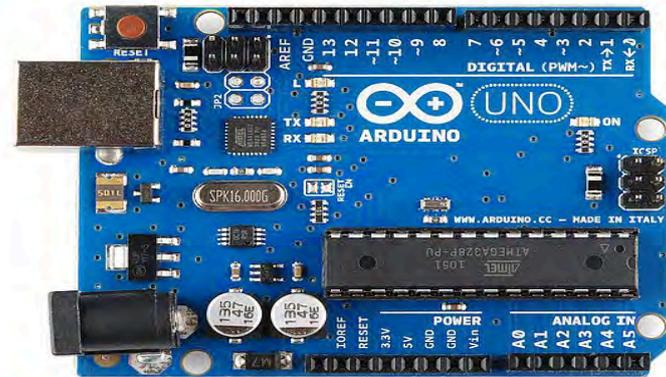
## 2.2.6 Arduino

Menurut (Ardiyanto & Edy Supriyadi, 2021, p. 12-13), Arduino adalah mikrokontroler papan tunggal dengan sumber terbuka, dibuat dengan tujuan mempermudah pembelajaran penggunaan elektronik untuk beragam fungsi. Arduino ditenagai oleh prosesor Atmel AVR sebagai perangkat keras andalannya dan menggunakan bahasa pemrogramannya khusus. Arduino dikenal sebagai arsitektur perangkat keras terbuka untuk pengembangan *prototype* perangkat elektronik interaktif dan mudah digunakan. Arduino diprogram dengan bahasa pemrograman sendiri dan *syntax* yang serupa seperti bahasa pemrograman C.

Arduino diproduksi dalam berbagai varian. Varian yang umum dari Arduino adalah Arduino Uno, Arduino Leonardo, Arduino Nano, dan lainnya. Arduino Uno adalah papan mikrokontroler berbasis prosesor ATmega328, ditambah 14 pin I/O dimana 6 pin untuk output PWM, 6 analog *input*, *crystal* sinyal elektronik 16 MHz, port USB, colokan daya, dan tombol pengaturan ulang. Mikrokontroler Arduino dapat dikoneksikan ke komputer menggunakan kabel USB.

Arduino memiliki fungsi memudahkan penggunaannya dalam berbagai alat berbasis elektronik seperti perakitan aplikasi LED berjalan, lampu lalu lintas, *mobile* robot, pemberi pakan otomatis, dan lain-lain. Dengan bantuan mikrokontroler Arduino, lebih nyaman, lebih mudah, dan lebih murah dalam membuat proyek-proyek sederhana. Proyek Arduino dapat bekerja secara independen, atau berinteraksi melalui perangkat lunak yang beroperasi di PC (misalnya Pengerjaan, *MaxMSP*, dan *Flash*). *Board* dapat dibeli atau dirakit secara manual dan IDE diunduh secara gratis. Mikrokontroler Arduino terus

berkembang dan memiliki banyak variasi, menjadikannya salah satu mikrokontroler yang paling populer untuk dipakai di seluruh dunia.



**Gambar 2.5 Arduino Uno**

(Sumber: <https://ndoware.com/apa-itu-arduino-uno.html>)

## 2.2.7 Visual Studio Code

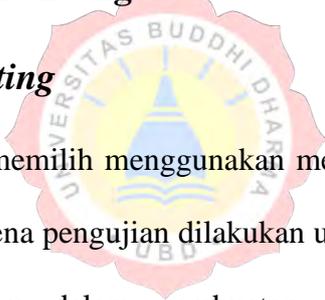
Dikutip dari buku *Seri Belajar ASP.NET: ASP.NET Core & PostgreSQL* (Faisal, 2017, p. 13–15), aplikasi desktop visual studio code merupakan sebuah *Integrated Development Environment* (IDE) yang dibuat oleh Microsoft bertujuan memudahkan para pengembang perangkat lunak mengembangkan aplikasinya. Aplikasi ini mampu dimanfaatkan untuk membuat berbagai aplikasi dari aplikasi ponsel, situs web, aplikasi komputer hingga komputasi awan. Bahasa yang diakomodasi oleh Visual Studio mulai dari Visual Basic, C#, C++, Python, Javascript dan lain-lain. Semenjak tahun 2015 Visual Studio hanya bisa dijalankan di Microsoft Windows.

Fitur-fitur utama dari Visual Studio Code adalah:

1. *Intelligent code completion*, fitur ini akan membantu pengembang perangkat lunak untuk menyelesaikan variabel, metode dan modul yang tertulis.
2. *Streamlined debugging*, fitur yang digunakan untuk melakukan debug terhadap kode yang diketik.
3. *Linters, multi-cursor editing, parameter hints*.
4. Navigasi kode.
5. Pemfaktoran ulang (*Refactoring*).
6. Adanya dukungan akses Git.

## 2.3 Teori Analisis dan Perancangan

### 2.3.1 *Black Box Testing*



Penulis memilih menggunakan metode *Black Box Testing* atau tes kotak hitam karena pengujian dilakukan untuk menguji fungsionalitas fitur pengenalan suara dalam pembuatan asisten virtual dengan bahasa pemrograman Python. Pengujian *Black Box* digunakan untuk memastikan bahwa semua alur proses yang ditentukan telah dijalankan sesuai dengan fungsinya.

Tes kotak hitam adalah metode pengujian yang berpusat di spesifikasi juga fitur perangkat yang akan diuji. *Black Box* diimplementasikan dengan melihat kesimpulan kinerja lewat uji coba dan memverifikasi fungsionalitas aplikasi yang telah dibuat.

Pengujian *Black Box* digunakan untuk menemukan kesalahan pada kategori berikut:

1. Patokan kinerja yang tidak sesuai atau berhenti.
2. Ketidaksesuaian tampilan.
3. Struktur data dan basis data eksternal yang tidak sesuai.
4. Performa atau reaksi tidak sesuai yang diharapkan.
5. Inaugurasi dan akhir yang tidak cocok.

Dalam pengujian *black box* umumnya digunakan 3 teknik berikut:

1. *Equivalence Partitioning*

Menurut (Krismadi et al., 2019, p. 156), Pengujian *Equivalence Partitioning* adalah melakukan pengujian keunggulan aplikasi untuk dibuahkan pengarsipan pengecekan perangkat lunak dengan cara mencari dan menemukan kekurangan di masing-masing *form* yang dipilah kedalam 3 model kekurangan, yaitu ketidaktepatan tujuan, figur data, dan antarmuka aplikasi.

2. *Boundarie Value Analysis*

Berdasarkan Jurnal Pengembangan IT (Snadhika Jaya, 2018), *Boundary Value Analysis* yaitu pengujian kotak hitam dengan memprioritaskan di cara kerja *input* dan menguji perhitungan garis atas dan perhitungan garis bawah. Prinsip kerja *Boundary Value Analysis* yaitu: (1) Kekurangan yang sering kedatangan terjadi selama kegiatan entri data; (2) Aktivitas saat masukan diproses.

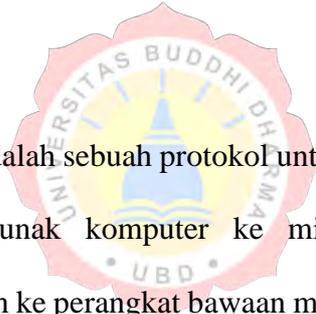
3. *Cause Effect Relationship*

Teknik desain pengujian perangkat lunak meliputi mengidentifikasi kasus (kondisi *input*) dan efek (kondisi *output*). *Cause Effect Relationship* adalah teknik pengujian yang mewakili kondisi logika dan dapat ditindaklanjuti. (Babbar, 2017)

Manfaat dari menggunakan metode pengujian Black Box Testing adalah:

1. Penguji tidak harus memerlukan wawasan khusus mengenai bahasa pemrograman yang digunakan.
2. Pengetesan didasarkan dari perspektif pengguna, dengan maksud mencari ketidakjelasan atau pertentangan dalam persyaratan detail.
3. Pemrogram memiliki ketergantungan dengan penguji.
4. Pemeriksaan dapat langsung dilakukan segera setelah spesifikasi perangkat lunak selesai dibuat.

### 2.3.2 *Firmata*

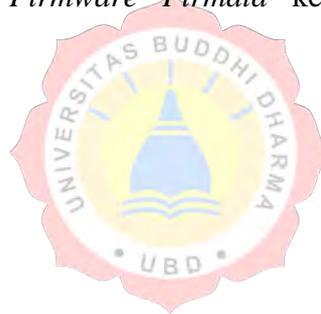


*Firmata* adalah sebuah protokol untuk menerima dan mengirim pesan dari perangkat lunak komputer ke mikrokontroler. Protokol tersebut diimplementasikan ke perangkat bawaan mikrokontroler atau paket *software*. Pada dasarnya, semua jenis mikrokontroler dapat mengimplementasikan protokol *Firmata*, akan tetapi saat ini implementasi yang paling lengkap masih ditemukan pada mikrokontroler Arduino. (*Firmata Protocol Documentation*, 2022)

Pada perancangan dan pembuatan aplikasi *Python Voice Assistant* diperlukan penggunaan protokol *Firmata* untuk menghubungkan kondisi pin Arduino Uno agar bisa dioperasikan melalui bahasa pemrograman Python dengan *library* Pyfirmata. Firmata perlu diunduh terlebih dahulu ke memori Arduino. Arduino IDE sudah menyediakan akses ke modul tersebut.

Berikut langkah pemasangan protokol *Firmata* ke dalam mikrokontroler Arduino:

1. Sambungkan mikrokontroler dengan komputer, kemudian jalankan program *Arduino IDE*. *Installer* aplikasi Arduino IDE bisa didapatkan dari situs asli Arduino di laman <https://www.arduino.cc/en/Main/Software>.
2. Selanjutnya buka menu *file* -> *example* -> *Firmata* -> *StandardFirmata*. Lalu muncul kode asli dari *StandardFirmata*. Unggah kode sumber ke Arduino dengan mengklik simbol panah unggah.
3. Instalasi *Firmware Firmata* ke mikrokontroler Arduino telah berhasil.



### 2.3.3 Flow Chart

Menurut (Santoso, 2017, p. 86), *Flow chart* adalah representasi simbolis dari algoritma atau prosedur yang digunakan untuk memecahkan masalah, penggunaan *flow chart* akan membantu pengguna dalam memeriksa bagian yang terlewatkan saat analisis masalah, selain itu *flow chart* juga berguna sebagai memfasilitasi komunikasi antara *programmer* yang bekerja dalam sebuah proyek tim.

Urutan logika yang panjang dan rumit dapat dijelaskan menggunakan *flow chart*. *Flow chart* memudahkan bagaimana komunikasi alur program kepada orang awam (selain penulis kode) menjadi kian sederhana. Dikutip

dari [www.pakdosen.co.id](http://www.pakdosen.co.id) (pakdosen, 2021) jenis-jenis *flow chart* antara lain adalah sebagai berikut:

1. *Flow chart* Sistem

*Flow chart* sistem adalah skema yang menunjukkan seluruh jalan kerja atau hal yang dilakukan oleh suatu metode dan menggambarkan serangkaian langkah metode tersebut.

2. *Flow chart* Dokumen

*Flow chart* dokumen adalah bagan alir untuk mewakili penjelasan dan laporan termasuk duplikatnya. Rancangan alir salinan ini memiliki penanda yang mirip sesuai rancangan alir sistem.

3. *Flow chart* Skematik

*Flow chart* skematik merupakan desain proses yang mirip dengan rangka sistem alir, yaitu pencitraan bagan ke sistem. Komparasinya selain menggunakan simbol rangka sistem alir, alir skematik juga menggunakan ilustrasi komputer dan kelengkapan lain yang digunakan.

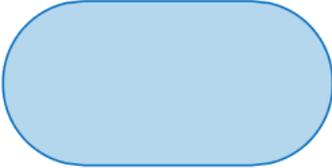
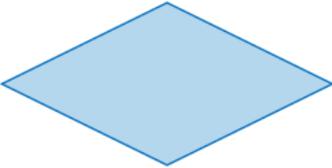
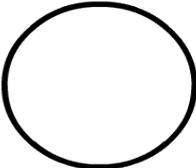
4. *Flow chart* Program

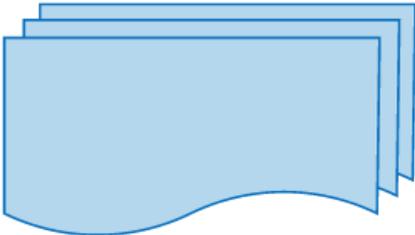
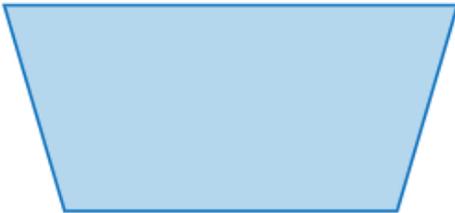
*Flow chart* program digunakan untuk mendefinisikan langkah-langkah dari metode program secara jelas. Perancangan alur program dilakukan dari konsepsi skema alir.

5. *Flow chart* Proses

*Flow chart* proses adalah metode penggambaran teknik industri yang memecah dan memisahkan proses selanjutnya ke bentuk sistem atau metode.

Tabel 2.1 Simbol *Flow Chart*

| Nama                        | Bentuk Simbol   | Deskripsi   |
|-----------------------------|---|---|
| Proses                      |    | Menampilkan proses atau langkah tindakan                                    |
| Garis alir                  |    | Menunjukkan arah aliran proses  |
| Garis alir<br>(putus-putus) |    | Menunjukkan arah aliran alternatif dari aliran informasi                    |
| Mulai/<br><i>terminator</i> |   | Titik awal atau akhir aliran dari proses.                                   |
| Keputusan                   |  | Mewakili titik pengambilan keputusan  |
| Konektor                    |  | Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama. |
| Persiapan                   |  | Penyiapan awal dan langkah persiapan lainnya sebelum memulai aliran proses. |

|                     |   |   |
|---------------------|---|---|
| <i>Input-output</i> |    | Mewakili proses masukan dan keluaran terlepas dari apa jenis peralatannya.              |
| Dokumen             |    | Mencetak keluaran ke bentuk dokumen   |
| Banyak dokumen      |   | Memiliki fungsi yang sama dengan dokumen, tetapi dokumen yang digunakan lebih dari satu |
| Layar               |  | Mencetak keluaran ke layar monitor  |
| Manual              |  | Menunjukkan proses yang tidak dilakukan oleh komputer                                   |

## 2.4 Tinjauan Studi

### 2.4.1 Penelitian

#### 1. Penelitian I Komang Setia Buana

**Tabel 2.2 Jurnal I Komang Setia Buana**

| No. | Data Jurnal        | Keterangan   |
|-----|--------------------|--|
| 1   | Judul              | Implementasi Aplikasi <i>Speech to Text</i> untuk Memudahkan Wartawan Mencatat Wawancara dengan Python   |
| 2   | Jurnal             | Jurnal Sistem dan Informatika (JSI)  |
| 3   | Volume dan Halaman | Volume 14 Nomor 2, 135-142   |
| 4   | Tanggal dan Tahun  | 1 Juli 2020  |
| 5   | Penulis            | I Komang Setia Buana   |
| 6   | Penerbit           | Politeknik Siber Dan Sandi Negara  |
| 7   | Tujuan Penelitian  | Membantu wartawan merekam wawancara dengan bahasa pemrograman Python menggunakan fitur <i>Speech to Text</i>                                     |
| 8   | Lokasi dan Subjek  | Lokasi: Indonesia<br>Subjek: wartawan  |
| 9   | Perancangan Sistem | a. Laptop Lenovo Y50 Intel Core i7<br>b. Software Python versi 3.8.1<br>c. OPPO A5 (Telepon pintar berbasis Android)<br>d. Windows 10 Enterprise |

|    |                      |   |
|----|----------------------|---|
| 10 | Hasil Penelitian     | Aplikasi membantu wartawan dalam merekam dan mencatat wawancara dengan fitur <i>speech to text</i> memanfaatkan bahasa pemrograman Python berhasil dibuat.  |
| 11 | Kekuatan Penelitian  | <ul style="list-style-type: none"> <li>a. Mempersingkat waktu. Umumnya mengetik membutuhkan waktu lebih lambat daripada berbicara.</li> <li>b. Tingkat keberhasilan dalam konversi suara cukup tinggi dengan rata-rata 94,75 %.</li> </ul>  |
| 12 | Kelemahan Penelitian | <ul style="list-style-type: none"> <li>a. Tingkat keberhasilan dalam merekam suara dari Youtube masih terbilang rendah hanya 87,5 % dari total durasi 20 detik.</li> <li>b. Aplikasi tidak dapat mengidentifikasi tanda baca.</li> </ul>  |
| 13 | Kesimpulan           | <ul style="list-style-type: none"> <li>a. Aplikasi untuk mencatat wawancara dengan fitur <i>speech to text</i> berhasil diimplementasikan.</li> <li>b. Modul <i>speech recognition</i> dan algoritma <i>FastICA</i> pada bahasa pemrograman Python, dapat merubah suara dalam bentuk rekaman berformat .wav ke bentuk teks atau tulisan, tetapi tanda baca tidak bisa dikenali aplikasi ini.</li> <li>c. Dari hasil uji coba yang sudah dilakukan tingkat keberhasilan mencapai 94,75 %.</li> </ul> |

## 2. Penelitian Mohamad Ali Sadikin dan Dedy Septono C. P

**Tabel 2.3 Jurnal Mohamad Ali Sadikin, dan Dedy Septono C. P**

| No. | Data Jurnal        | Keterangan   |
|-----|--------------------|--|
| 1   | Judul              | <i>Secure Personal Assistant Dengan Perintah Suara Berbasis Internet of Things (IoT) untuk Smart Office</i>  |
| 2   | Jurnal             | Seminar Nasional Aplikasi Teknologi Informasi (SNATi)  |
| 3   | Volume dan Halaman | -  |
| 4   | Tanggal dan Tahun  | 11 Agustus 2018  |
| 5   | Penulis            | Mohamad Ali Sadikin, Dedy Septono C. P   |
| 6   | Penerbit           | Sekolah Tinggi Sandi Negara  |
| 7   | Tujuan Penelitian  | Mengimplementasikan purwarupa <i>secure personal assistant</i> ke perangkat Raspberry pi   |
| 8   | Lokasi dan Subjek  | Lokasi: Bogor, Indonesia<br>Subjek: <i>Smart Office</i>  |
| 9   | Perancangan Sistem | <ul style="list-style-type: none"> <li>a. <i>Raspberry Pi 3 Model B</i></li> <li>b. DHT11 (Sensor temperatur dan kelembaban)</li> <li>c. MQ-2 Sensor (Sensor pendeteksi api)</li> <li>d. USB Mikrofon</li> <li>e. Speaker</li> <li>f. Kamera</li> <li>g. <i>Motion detection sensor</i></li> </ul> |

|    |                      |  |
|----|----------------------|--|
| 10 | Hasil Penelitian     | Perangkat asisten personal dapat diimplementasikan ke perangkat <i>Raspberry Pi</i> . Alat ini diperintah dan memberikan balasan juga melalui suara. Perangkat mampu beroperasi berdasarkan perintah pengguna. Dari hasil uji coba, perintah suara dikenali dengan memakai Google API.   |
| 11 | Kekuatan Penelitian  | <p>a. Kelebihan dari perangkat ini yakni dapat melakukan komputasi yang cukup, menghemat penggunaan komponen, lebih irit, lebih mudah, dan tidak boros energi karena menggunakan <i>Raspberry Pi 3 Model B</i>.</p> <p>b. Sistem pengamatan mampu mengantarkan video secara langsung dari perangkat <i>personal assistant</i> ke perangkat pengguna, contohnya komputer atau telepon pintar.</p> <p>c. Enkripsi pada paket data yang dikirim, dibuktikan dengan <i>software</i> Wireshark melalui <i>sniffing</i>.</p> |
| 12 | Kelemahan Penelitian | a. Membutuhkan API dan layanan eksternal seperti Google atau <i>Wolfram</i> untuk beroperasi secara penuh. Membutuhkan waktu rata-rata sekitar satu menit untuk membalas. Bahkan di beberapa percobaan bergantung pada koneksi internet atau Google <i>service</i> , membutuhkan waktu sampai 8 detik.   |

|    |            |  |
|----|------------|--|
|    |            | <p>b. Apabila perintah suara terlalu panjang, asisten pribadi memerlukan waktu banyak untuk mengeksekusi perintah.</p>   |
| 13 | Kesimpulan | <p>a. Perangkat asisten pribadi telah berhasil dijalankan pada <i>Raspberry Pi</i>. Perangkat ini diperintah dan membalas respon dengan suara. Perangkat bekerja berdasarkan perintah pengguna.</p> <p>b. VPN SSL/TLS berhasil diterapkan untuk mengamankan informasi dan data yang dikirimkan dari perangkat asisten pribadi ke perangkat <i>user</i>.</p> <p>c. Hasil uji keamanan memperlihatkan jika paket data dikirim ke dalam bentuk yang sudah dienkripsi, hal ini dapat dilihat dari paket yang dicek melalui aplikasi Wireshark dengan metode <i>sniffing</i>.</p> |

### 3. Penelitian Dr.V. Geetha, Dr. C, K, Gomathy, Kottamasu Manasa

Sri Vardhan, dan Nukala Pavan Kumar

**Tabel 2. 4 Jurnal Dr.V. Geetha, Dr. C, K, Gomathy, Kottamasu Manasa Sri Vardhan, dan Nukala Pavan Kumar**

| No. | Data Jurnal        | Keterangan  |
|-----|--------------------|---|
| 1   | Judul              | <i>The Voice Enabled Personal Assistant for Pc using Python</i>   |
| 2   | Jurnal             | <i>International Journal of Engineering and Advanced Technology</i>   |
| 3   | Volume dan Halaman | Volume-10 Issue-4, 162-165  |
| 4   | Tanggal dan Tahun  | April 2021  |
| 5   | Penulis            | Dr.V. Geetha, Dr. C, K, Gomathy, Kottamasu Manasa Sri Vardhan, dan Nukala Pavan Kumar   |
| 6   | Penerbit           | <i>Blue Eyes Intelligence Engineering &amp; Sciences Publication</i>  |
| 7   | Tujuan Penelitian  | Menyajikan gambaran tentang desain dan pengembangan dari asisten pribadi berbasis suara untuk PC menggunakan bahasa pemrograman Python. |
| 8   | Lokasi dan Subjek  | Lokasi: India<br>Subjek: Asisten Pribadi Virtual  |
| 9   | Perancangan Sistem | a. Komputer desktop<br>b. Bahasa pemrograman Python<br>c. Microsoft Visual Studio Code (IDE)<br>d. Pyttsx3 library                      |

|    |                      |  |
|----|----------------------|--|
|    |                      | <p>e. <i>Google Speech Recognition engine</i></p> <p>f. <i>Microsoft Bing Voice Recognition engine</i></p>   |
| 10 | Hasil Penelitian     | Asisten virtual berfungsi dengan baik untuk melakukan beberapa tugas yang diberikan oleh pengguna. Hal lainnya yang mampu dijalani, seperti mematikan PC, menyalakan ulang, atau membacakan berita terbaru, hanya dengan satu perintah suara.                      |
| 11 | Kekuatan Penelitian  | <p>a. Aplikasi sudah bisa berfungsi dengan baik untuk memenuhi beberapa tugas sederhana yang diberikan.</p> <p>b. Penyajian data sudah dalam bentuk kalimat dan tangkapan layar program sehingga mempermudah pembaca memahami isi dari hasil penelitian</p>        |
| 12 | Kelemahan Penelitian | <p>a. Pembuatan aplikasi baru menggunakan tampilan <i>Command Line Interface (CLI)</i> saja tanpa <i>Graphical User Interface (GUI)</i>.</p> <p>b. Aplikasi baru bisa berjalan pada platform desktop saja.</p>   |
| 13 | Kesimpulan           | <p>a. Paper ini menyajikan ikhtisar komprehensif tentang desain dan pengembangan asisten pribadi berbasis suara untuk PC menggunakan bahasa pemrograman Python.</p> <p>b. Asisten Pribadi ini telah dirancang dengan kemudahan penggunaan sebagai fitur utama.</p> |

|  |  |  |
|--|--|--|
|  |  | <p>Asisten berfungsi dengan baik untuk melakukan beberapa tugas yang diberikan oleh pengguna.</p> <p>c. Hal lainnya yang mampu dijalani asisten ini adalah mematikan PC, menyalakan ulang, atau membacakan berita terbaru, hanya dengan satu perintah suara.</p> |
|--|--|--|

**4. Penelitian Jurnal Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, dan Megha Raule**

**Tabel 2.5 Jurnal Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, dan Megha Raule**

| No. | Data Jurnal        | Keterangan  |
|-----|--------------------|---|
| 1   | Judul              | <i>Personal Virtual Assistant for Windows using Python</i>  |
| 2   | Jurnal             | <i>International Research Journal of Modernization in Engineering Technology and Science</i>                      |
| 3   | Volume dan Halaman | Volume-03 Issue-07, 485-491   |
| 4   | Tanggal dan Tahun  | Juli 2021   |
| 5   | Penulis            | Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, dan Megha Raule                                    |
| 6   | Penerbit           | Department of Computer Science & Engineering, JSPM's Bhagwant Institute of Technology, Barshi, Maharashtra, India |

|   |                    |  |
|---|--------------------|--|
| 7 | Tujuan Penelitian  | <p>Membuat rancangan dan tampilan asisten virtual berbasis windows menggunakan Python yang dapat menyediakan beberapa layanan, meliputi</p> <ol style="list-style-type: none"> <li>a. Menunjukkan kondisi cuaca.</li> <li>b. Penjadwalan janji.</li> <li>c. Membuat pengaturan perjalanan.</li> <li>d. Memutar musik, film, dll.</li> <li>e. Menampilkan tanggal waktu.</li> <li>f. Mengelola email.</li> <li>g. Buka aplikasi.</li> </ol> |
| 8 | Lokasi dan Subjek  | <p>Lokasi: India</p> <p>Subjek: Asisten Pribadi Virtual</p>  |
| 9 | Perancangan Sistem | <ol style="list-style-type: none"> <li>a. Subprocess</li> <li>b. WolframAlpha</li> <li>c. Pyttsx3</li> <li>d. JSON</li> <li>e. Speech Recognition</li> <li>f. Gtts (<i>Google Text to Speech</i>)</li> <li>g. Tanggal dan Waktu</li> <li>h. Wikipedia</li> <li>i. Web Browser</li> <li>j. Winshell</li> <li>k. Pyjokes</li> <li>l. Pyaudio</li> <li>m. Smtplib</li> </ol>  |

|    |                     |   |
|----|---------------------|---|
|    |                     | <p>n. Ctypes</p> <p>o. Request</p>  |
| 10 | Hasil Penelitian    | <p>Penggunaan asisten virtual dapat menghemat waktu. Asisten virtual adalah perangkat lunak yang dapat mengerti lisan atau tulisan dan melakukan tugas yang diperintah oleh si pengguna. Asisten virtual menggunakan pemrosesan bahasa alami (<i>Natural Language Processing</i>) untuk mencocokkan input suara atau teks pengguna dengan perintah yang dapat dieksekusi. Dengan bantuan asisten virtual Anda dapat menjalankan perangkat seperti laptop atau PC atas perintah Anda sendiri. Prosesnya cepat, sehingga menghemat waktu.</p> |
| 11 | Kekuatan Penelitian | <p>a. Jurnal dilengkapi tangkapan layar cara penginstalan modul-modul yang diperlukan dalam pembuatan aplikasi.</p> <p>b. Aplikasi sudah bisa berfungsi dengan baik untuk memenuhi beberapa tugas sederhana yang diberikan.</p> <p>c. Penyajian data sudah dalam bentuk kalimat dan tangkapan layar program sehingga mempermudah pembaca memahami isi dari hasil penelitian.</p>  |

|    |                      |  |
|----|----------------------|--|
| 12 | Kelemahan Penelitian | <ol style="list-style-type: none"><li>1. Pembuatan aplikasi baru menggunakan tampilan <i>Command Line Interface</i> (CLI) saja tanpa <i>Graphical User Interface</i> (GUI).</li><li>2. Aplikasi baru bisa berjalan pada platform desktop saja.</li></ol>   |
| 13 | Kesimpulan           | <ol style="list-style-type: none"><li>a. Asisten virtual membuat hidup lebih mudah bagi manusia.</li><li>b. Asisten pribadi virtual juga dapat diandalkan daripada asisten pribadi manusia karena, asisten pribadi virtual lebih portabel, setia, dan tersedia untuk digunakan kapan saja.</li><li>c. Proyek ini telah menggunakan teknologi <i>Artificial Intelligence</i> (kecerdasan buatan) sehingga dapat mengolah input yang diberikan oleh klien.</li></ol> |

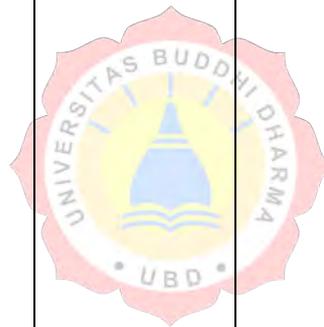
## 2.4.2 Rangkuman Jurnal Penelitian

Tabel 2.1 Rangkuman Jurnal Penelitian

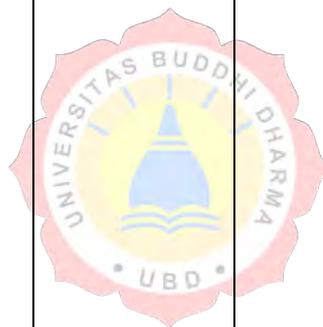
| Peneliti             | Nama Jurnal                         | Tahun | Institusi                         | Judul dan Metode yang digunakan   | Kesimpulan   |
|----------------------|-------------------------------------|-------|-----------------------------------|---|--|
| I Komang Setia Buana | Jurnal Sistem dan Informatika (JSI) | 2020  | Politeknik Siber Dan Sandi Negara | Implementasi Aplikasi <i>Speech to Text</i> untuk Memudahkan Wartawan Mencatat Wawancara dengan Python. | <p>1. Aplikasi untuk mencatat wawancara dengan fitur <i>speech to text</i> berhasil diimplementasikan.</p> <p>2. Modul <i>speech recognition</i> dan algoritma <i>FastICA</i> pada bahasa pemrograman Python, dapat merubah suara dalam bentuk rekaman berformat .wav ke bentuk teks atau tulisan, tetapi tanda baca tidak bisa dikenali aplikasi ini.</p> |

|  |   |      |                             |   |   |
|--|---|------|-----------------------------|---|---|
|  |   |      |                             |   | 3. Dari hasil uji coba yang sudah dijalankan tingkat keberhasilan mencapai 94,75 %.   |
| Mohamad Ali Sadikin, Dedy Septono C. P | Seminar Nasional Aplikasi Teknologi Informasi (SNATi) | 2018 | Sekolah Tinggi Sandi Negara | <i>Secure Personal Assistant</i> dengan Perintah Suara Berbasis <i>Internet of Things (IoT)</i> untuk <i>Smart Office</i> . | <p>1. Perangkat asisten pribadi telah berhasil dijalankan pada Raspberry Pi. Perangkat ini diperintah dan membalas respon dengan suara.</p> <p>Perangkat bekerja berdasarkan perintah pengguna.</p> <p>2. VPN SSL/TLS berhasil diterapkan untuk</p> |

|                  |                                 |      |                               |                          |  |
|------------------|---------------------------------|------|-------------------------------|--------------------------|--|
|                  |                                 |      |                               |                          | <p>mengamankan informasi dan data yang dikirimkan dari perangkat asisten pribadi ke perangkat user.</p> <p>3. Hasil uji keamanan memperlihatkan jika paket data dikirim ke dalam bentuk yang sudah dienkripsi, hal ini dapat dilihat dari paket yang dicek melalui aplikasi Wireshark dengan metode <i>sniffing</i>.</p> |
| Dr.V.<br>Geetha, | <i>International Journal of</i> | 2021 | <i>Blue Eyes Intelligence</i> | <i>The Voice Enabled</i> | 1. Paper ini menyajikan ikhtisar   |



|   |   |  |  |  |   |
|---|---|--|--|--|---|
| <p>Dr. C, K,<br/>Gomathy,<br/>Kottamas<br/>u Manasa<br/>Sri<br/>Vardhan,<br/>Nukala<br/>Pavan<br/>Kumar</p> | <p><i>Engineering<br/>and<br/>Advanced<br/>Technology</i></p> |  | <p><i>Engineering<br/>&amp; Sciences<br/>Publication</i></p> | <p><i>Personal<br/>Assistant for<br/>PC using<br/>Python</i></p> | <p>komprehensif tentang desain dan pengembangan asisten pribadi berbasis suara untuk PC menggunakan bahasa pemrograman Python.</p> <p>2. Asisten Pribadi ini telah dirancang dengan kemudahan penggunaan sebagai fitur utama. Asisten berfungsi dengan baik untuk melakukan beberapa tugas yang diberikan oleh pengguna.</p> <p>3. Asisten ini juga dapat mengeksekusi bermacam tugas, seperti mematikan PC, menyalakan ulang, atau</p> |
|---|---|--|--|--|---|

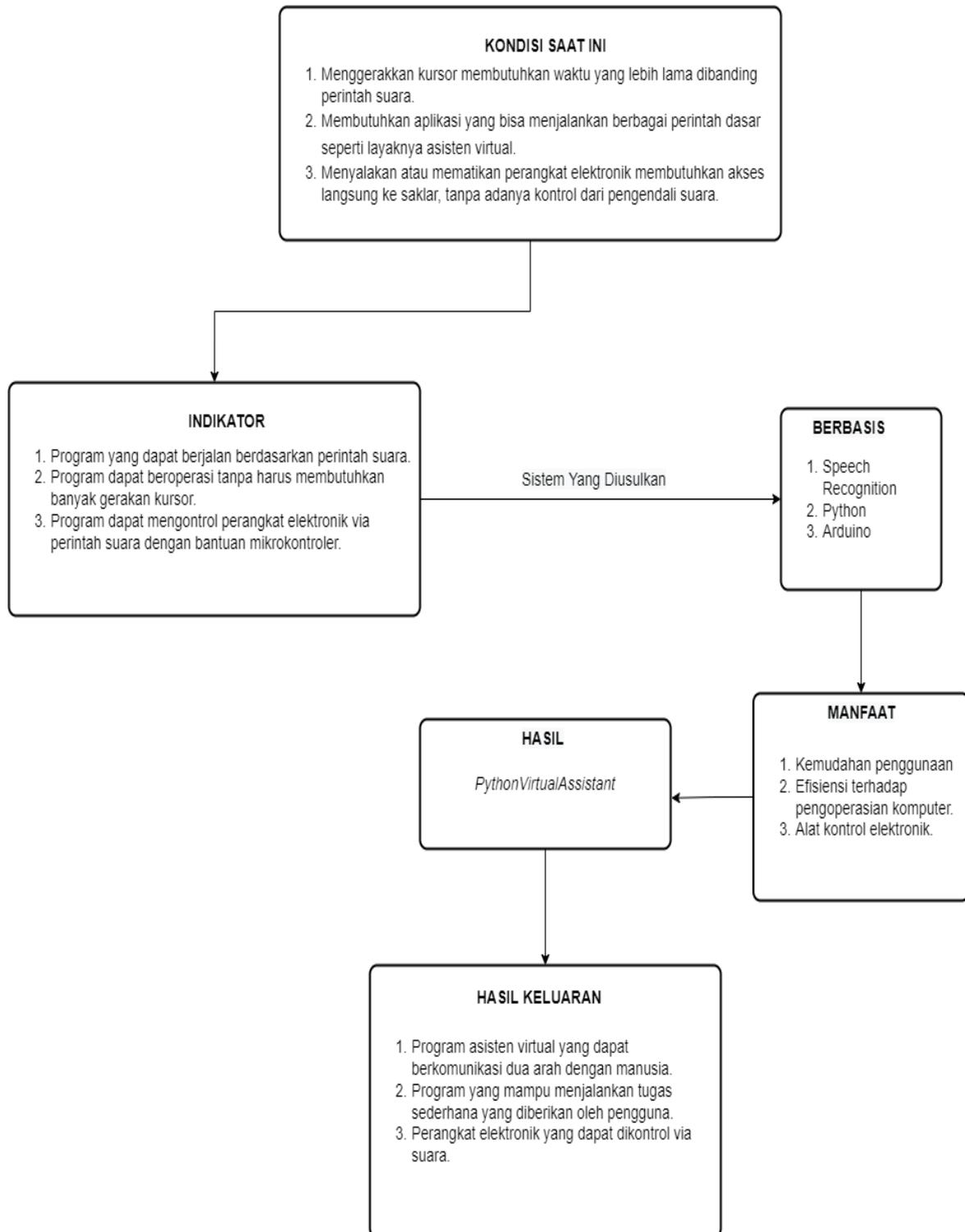


|  |  |      |  |  |   |
|--|--|------|--|--|---|
|  |  |      |  |  | membacakan berita terbaru, hanya dengan satu perintah suara.  |
| Anjali Fapal, Trupti Kanade, Bharati Janrao, Mrunalini Kamble, dan Megha Raule | <i>International Research Journal of Modernization in Engineering Technology and Science</i> | 2021 | <i>Department of Computer Science &amp; Engineering, JSPM's Bhagwant Institute of Technology, Barshi, Maharashtra, India</i> | <i>Personal Virtual Assistant for Windows using Python</i> | <p>1. Virtual asisten membuat hidup lebih mudah bagi manusia.</p> <p>2. Asisten pribadi virtual juga dapat diandalkan daripada asisten pribadi manusia karena, asisten pribadi virtual lebih portabel, setia, dan tersedia untuk digunakan kapan saja.</p> <p>3. Proyek ini telah menggunakan teknologi Artificial Intelligence (kecerdasan</p> |

|  |  |  |  |  |   |
|--|--|--|--|--|---|
|  |  |  |  |  | buatan) sehingga dapat mengolah <i>input</i> yang diberikan oleh klien. |
|--|--|--|--|--|---|



## 2.5 Kerangka Pemikiran



**Gambar 2.6 Kerangka Pemikiran**

## BAB III

### PERANCANGAN APLIKASI

#### 3.1 Analisa Kebutuhan dan Fungsional

##### 3.1.1 Analisis Kebutuhan

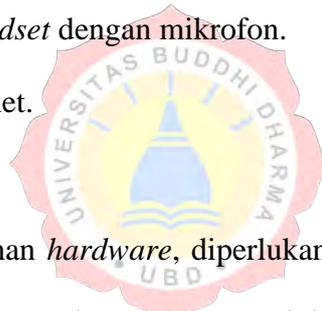
Analisis kebutuhan adalah bagian mengumpulkan fakta dan informasi untuk mendukung terciptanya suatu sistem demi merespon suatu rumusan masalah yang ada. Studi Pustaka adalah metode yang digunakan untuk analisis kebutuhan. Pada studi pustaka, fakta-fakta diperoleh melalui buku bacaan, jurnal ilmiah, penelitian terdahulu dan pembelajaran melalui dunia maya. Informasi yang diperlukan untuk membuat program pengenalan suara adalah proses pengambilan data ucapan menggunakan mikrofon kemudian mengkonversi ucapan tersebut menjadi teks ataupun sebaliknya menggunakan layanan daring *Google Speech Recognition* dan *Microsoft Speech API*. Teks tersebut kemudian akan diproses menjadi perintah untuk menjalankan sintaks kueri menggunakan logika *else-if*.

Asisten virtual seperti Google Now, Siri, dan Cortana dibuat oleh perusahaan teknologi terkemuka seperti Google, Apple dan Microsoft. Tujuan utama dari adanya asisten virtual adalah untuk membantu penggunanya berinteraksi dan mengoperasikan mesin dengan hanya memberikan *input* suara. Asisten virtual ini sangat berguna untuk lansia, orang dengan gangguan penglihatan, anak-anak, dll dengan membuat yakin bahwa interaksi dengan mesin bukanlah hal yang sulit untuk zaman sekarang.

Asisten virtual dibutuhkan untuk mengotomatisasi tugas dan dapat meningkatkan efisiensi.

Penelitian dilakukan dengan perangkat keras (*hardware*) yang sesuai agar program dapat berjalan dengan lancar. Adapun kebutuhan perangkat keras yang direkomendasikan adalah:

1. AMD Ryzen 5 2500U @ 2.0GHz.
2. RAM 4 GB.
3. Harddisk 512 GB.
4. Monitor 1366 x 768.
5. *Mouse* dan *Keyboard*.
6. *Earphone Headset* dengan mikrofon.
7. Koneksi Internet.



Selain kebutuhan *hardware*, diperlukan juga perangkat lunak (*software*) sebagai penunjang pembuatan program ini. Berikut ini perangkat lunak yang penulis gunakan untuk membuat aplikasi:

1. Microsoft Windows 10 Home.
2. Python dengan versi 3.8.5.
3. Microsoft Visual Studio Code.
4. Arduino IDE.

Berdasarkan informasi penelitian dari jurnal sebelumnya berjudul “*The Voice Enabled Personal Assistant for PC using Python*” dan “*Personal Virtual Assistant for Windows using Python*” biasanya 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*).

Sedangkan masukan (*input*) yang diperlukan berupa:

1. Suara dari pengguna.
2. Klik menggunakan mouse untuk menekan tombol pada tampilan.

Keluaran (*output*) yang diperoleh berupa:

1. Teks dari ucapan yang diberikan ditampilkan pada tampilan antar muka aplikasi
2. Hasil dari perintah yang diberikan dijalankan melalui aplikasi desktop.

### 3.1.2 Analisis Fungsional

Analisis fungsional menjelaskan tentang kebutuhan fungsional yang dikembangkan oleh sistem sebagai jawaban dari rumusan masalah. Adapun kebutuhan fungsional sistem ini diantaranya:

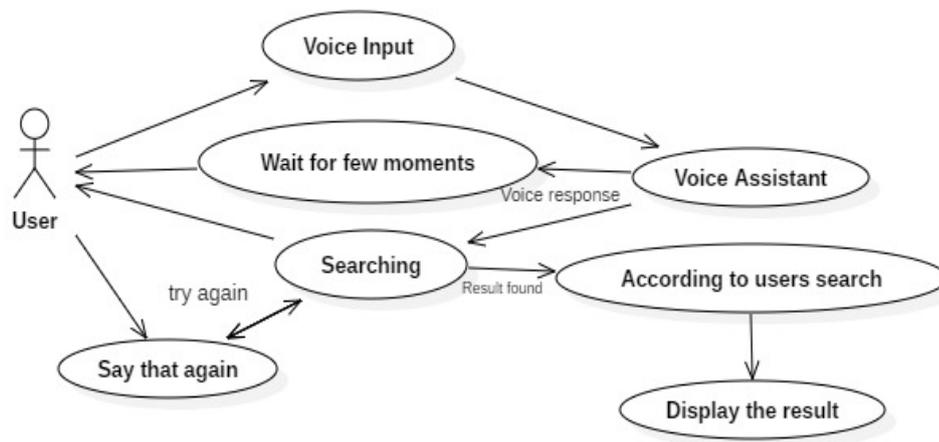
1. Program mampu memproses data suara yang ada.

2. Program dapat merespon inputan dari pengguna dengan *feedback* berupa suara dan tampilan yang diinginkan.
3. Program dapat menjalankan perintah dari kueri dan mengontrol alat elektronik.

### 3.2 Perancangan

Dalam penelitian ini dibuat sejumlah rancangan untuk membuat aplikasi asisten virtual dengan *speech recognition*. Rancangan pertama dilakukan sebelum membangun asisten virtual adalah pengumpulan data suara. Tahap ini disiapkan untuk membangun asisten virtual menggunakan *speech recognition*. Data suara akan digunakan untuk menjalankan perintah dan melakukan pengujian aplikasi.

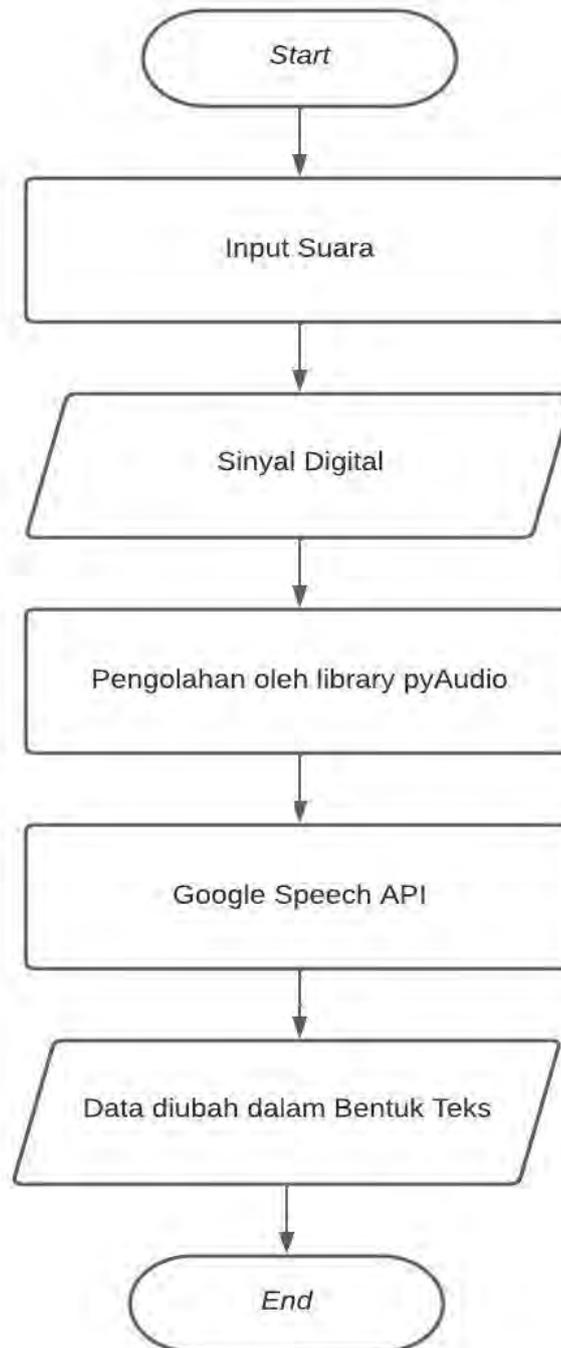
Asisten virtual dengan *speech recognition* pada penelitian ini mempunyai dua faktor penting untuk memperoleh data berbentuk suara dan merancang tampilan aplikasi yang diinginkan. Tahapan tersebut yaitu *speech recognition* dan pemrosesan kata kunci (*keyword*). Fitur *speech recognition* pada Python dapat diakses dengan menggunakan *library pytsx3*. Pada tahap implementasi nanti akan dijabarkan *library* apa saja yang dibutuhkan untuk menunjang aplikasi *Python Voice Assistant*.



**Gambar 3.1** Arsitektur Kerangka Kerja

### 3.2.1 Speech Processing

Pada tahap ini *speech processing* menggunakan bahasa pemrograman Python dengan versi minimal 3.8.5. Untuk tahap ini, setelah dilakukan perekaman suara menggunakan mikrofon sehingga sinyal analog dari suara manusia akan diolah menjadi sinyal digital dan dikirimkan melalui internet untuk diolah oleh *Google Speech API* secara daring untuk menghasilkan teks atau menyimpan data-data suara. Berikut tampilan flowchart dari *speech processing*.

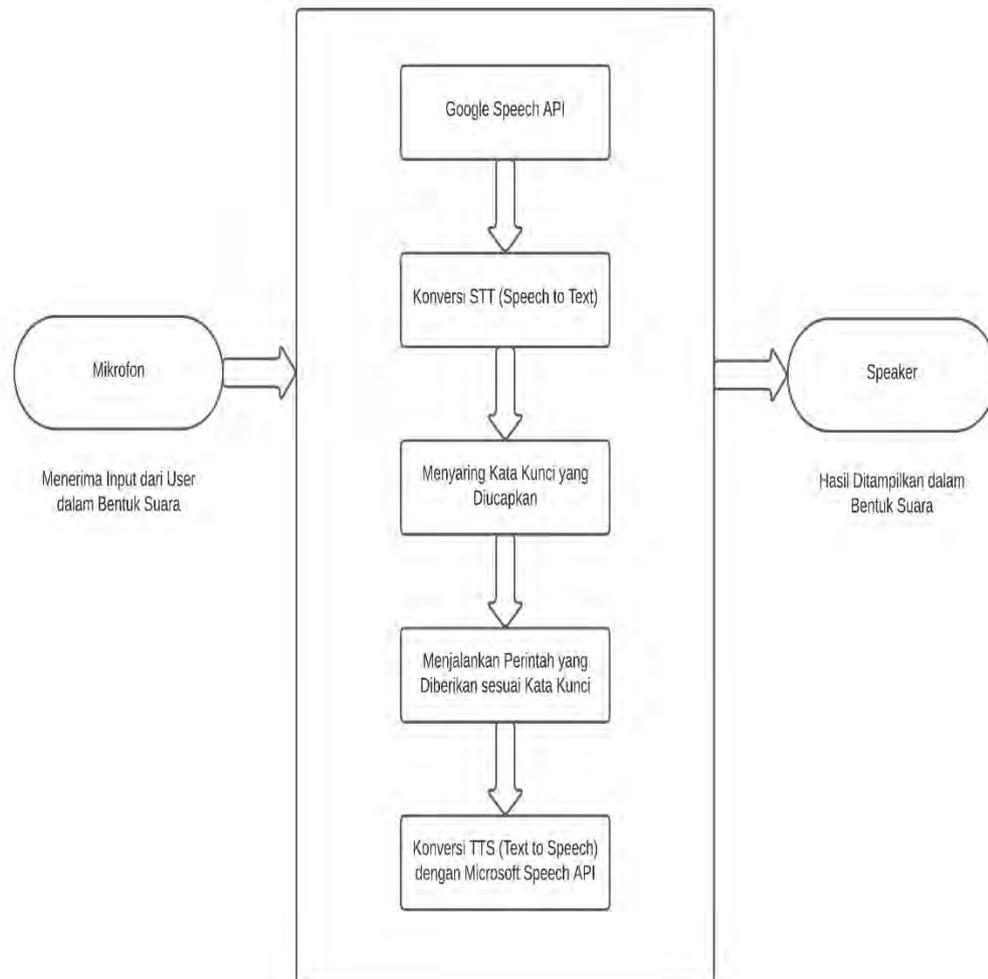


**Gambar 3.2** *Flow Chart Speech Processing*

PyAudio digunakan untuk mendapatkan input suara dari mikrofon. Untuk menginstallnya cukup mudah yaitu dengan mengetikkan *pip install pyAudio* pada terminal cmd (*command prompt*) Visual Studio Code.

### 3.2.2 Pengambilan Data

Proses pengambilan data dilakukan melalui rekaman suara pengguna. Pengguna mengucapkan perintah dalam Bahasa Inggris yang diinginkan untuk dijalankan pada komputer. Pengguna diharapkan berbicara dengan lantang dan jelas agar komputer dapat mengenali perintah yang akan dieksekusi. Proses pengambilan data dilakukan dengan menggunakan mikrofon dengan kondisi yang cukup sunyi agar suara yang direkam tidak ada kebisingan (*noise*).



**Gambar 3.3 Alur Pengambilan Data**

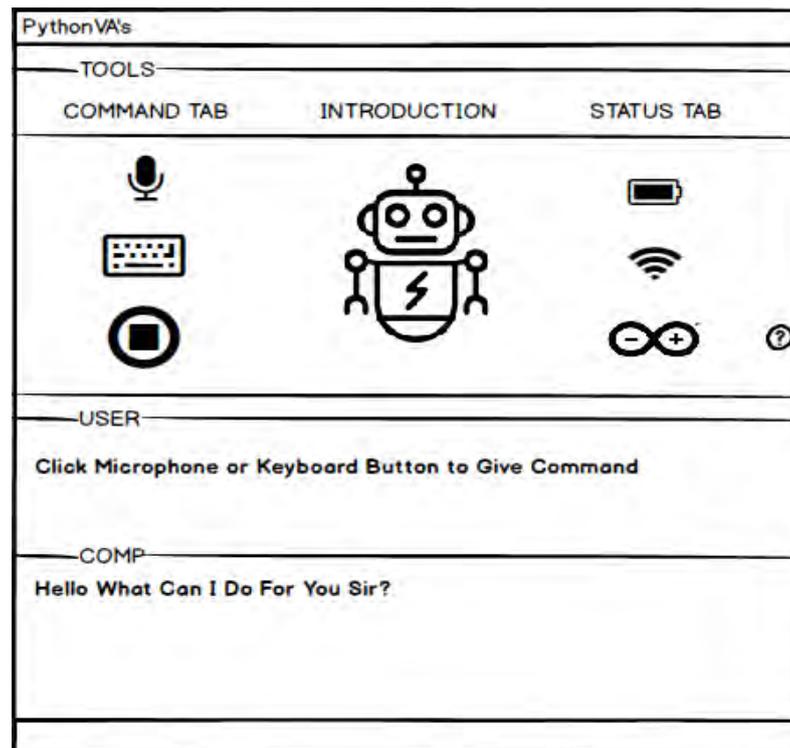
Penjelasan alur pengambilan data:

1. Saat pengguna berbicara, ucapan akan direkam oleh mikrofon komputer.
2. Ucapan diubah menjadi teks melalui *Google Speech API* yang merupakan atribut dari *library pyttsx3* untuk selanjutnya diproses oleh aplikasi.
3. Teks yang sudah dikonversi kemudian diproses untuk mendapatkan hasil yang dibutuhkan.
4. Teks berisi satu atau dua kata kunci yang menentukan kueri/ perintah apa yang akan dieksekusi. Jika kata kunci tidak cocok dengan salah satu kueri dalam kode, maka aplikasi akan meminta pengguna untuk mencoba lagi.
5. Hasil yang berupa teks akan diubah kembali menjadi suara dan hasilnya akan disuarakan menggunakan *Microsoft Speech API*.



### 3.2.3 Tampilan Antar Muka (*User Interface*)

Pada tahap rancangan ini akan ditemukan elemen apa saja yang akan ditampilkan pada *layout* tkinter. Diperkirakan gambar sederhana yang akan menjadi dasar pembuatan tampilan antar muka sebagai berikut:



**Gambar 3.4 Rancangan Tampilan Antar Muka**

Keterangan:

1. Nama program terletak pada bagian atas tampilan antar muka yaitu "PythonVAs" (singkatan dari *Python Voice Assistant*).
2. Tombol berbentuk mikrofon digunakan untuk merekam ucapan *user* setelah ditekan. Setelah tombol ditekan, aplikasi akan menunggu perintah dari *user* untuk mengeksekusi kueri. Jika perintah yang diberikan tidak dikenal, maka aplikasi akan memberi feedback berupa suara "*Say that again, please!*" dan *user* dapat mencoba kembali dengan memberi perintah yang lain.
3. Tombol berbentuk keyboard memiliki fungsi yang sama seperti tombol mikrofon. Bedanya tombol mikrofon digunakan untuk menerima perintah via suara, sedangkan tombol keyboard untuk menerima perintah dengan cara diketik.

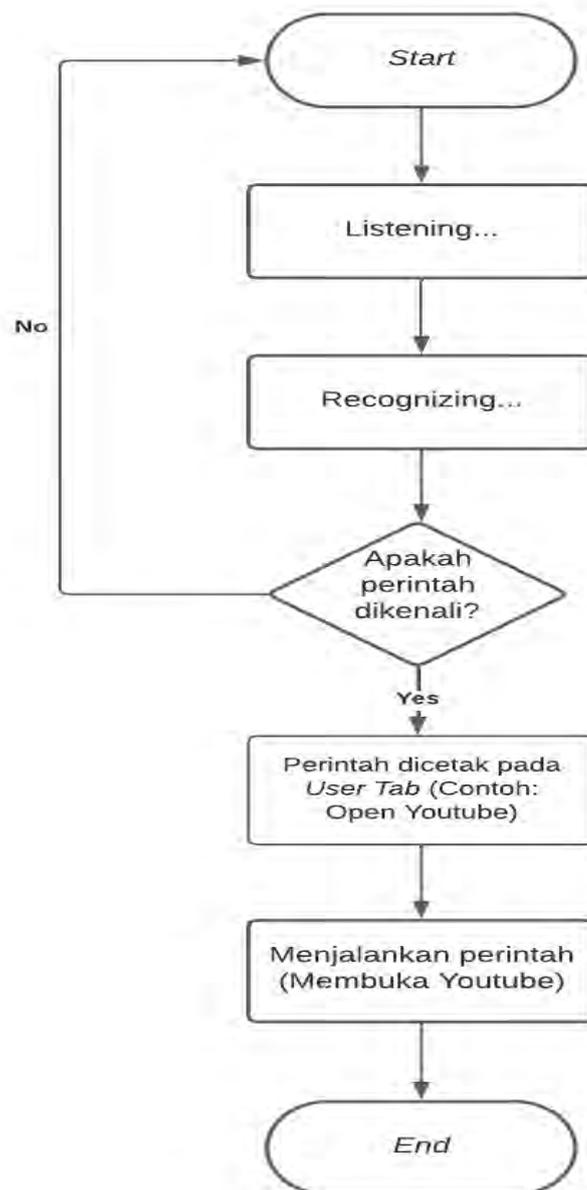
4. Tombol berbentuk robot digunakan untuk menginfokan fungsi program secara singkat.
5. *Status bar* dengan gambar baterai, sinyal, dan logo Arduino sebagai indikator persentase baterai, sinyal, dan konektivitas Arduino.
6. Tab *USER* digunakan sebagai petunjuk bagaimana cara menggunakan aplikasi dan akan menunjukkan hasil inputan user.
7. Tab *COMP* akan digunakan untuk menunjukkan hasil output dari perintah yang diberikan.

### 3.2.4 Pemrosesan Kueri

Pada tahap pemrosesan kueri, ucapan perintah dari *user* akan menjadi kata kunci dari pemrograman *else-if*. Asisten virtual menggunakan *Natural Language Processing* (NLP) atau pemrosesan bahasa secara alami untuk mencocokkan teks pengguna atau *input* suara dengan yang dapat dieksekusi perintah. Ketika pengguna mengajukan pertanyaan kepada *virtual assistant* untuk melakukan tugas, audio dalam bentuk sinyal diubah menjadi perintah yang dapat dieksekusi atau data digital yang dapat dianalisis oleh perangkat lunak. Data yang masuk kemudian dibandingkan dengan data *else-if* pada perangkat lunak untuk menemukan jawaban atau respon yang sesuai.

Pemrosesan kueri menggunakan beberapa paket penginstal Python seperti *Speech Recognition*, *pipwin*, dan *pyaudio*. Python menyediakan *library* yang disebut *SpeechRecognition* untuk memungkinkan mengubah perintah suara atau audio menjadi teks untuk diproses lebih lanjut sesuai dengan logika pemrograman *else-if*. Setelah perintah yang diberikan cocok,

*output* dihasilkan dalam bentuk teks dan suara. Jika perintah yang diberikan tidak sesuai atau tidak dikenal, maka aplikasi akan memberikan umpan balik (*feedback*) berupa suara yang menyarankan *user* untuk mengulang kembali ucapan atau perintah yang sudah diberikan sebelumnya.

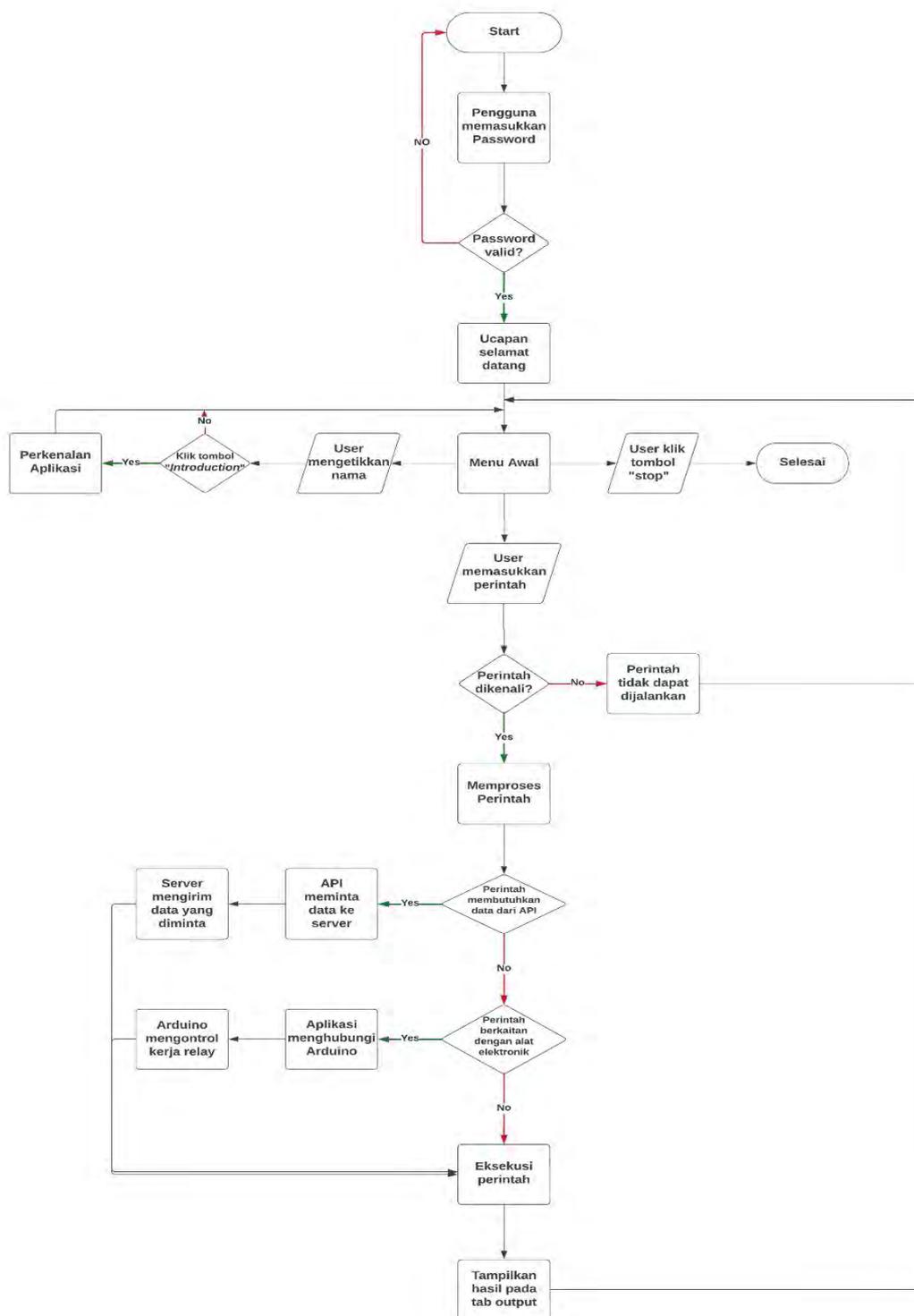


**Gambar 3.5 Flowchart Pemrosesan Kueri**

### 3.2.5 Perancangan Sistem dan Alat

Berikut alur perancangan sistem untuk mengoperasikan aplikasi

“Python Voice Assistant”:

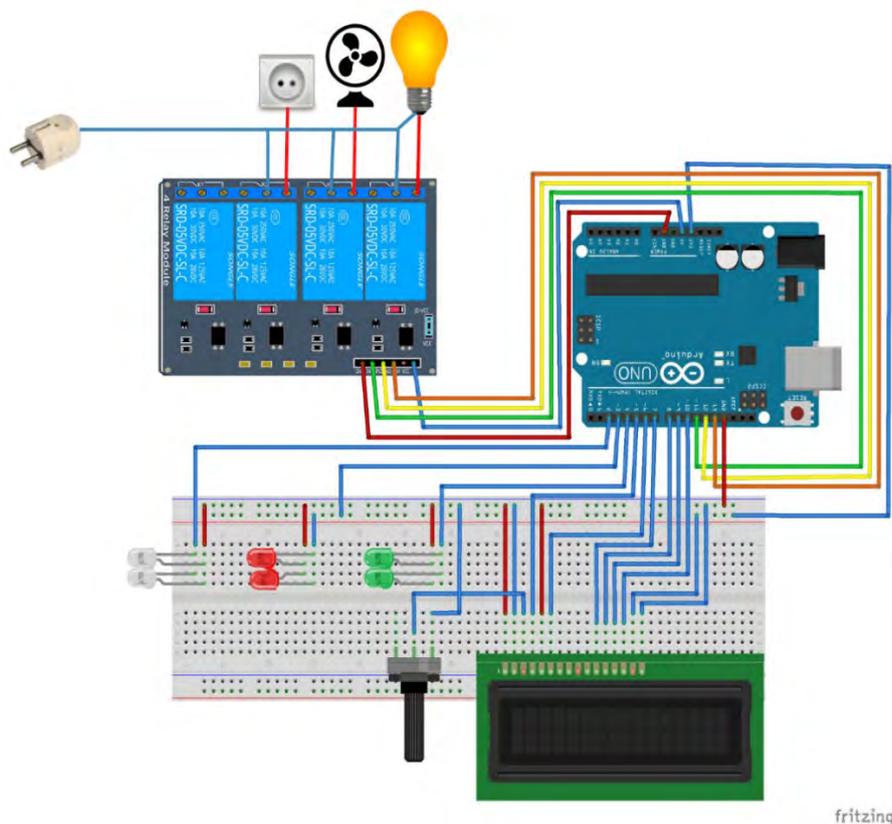


Gambar 3.6 Flowchart Sistem

Dalam penelitian ini dibutuhkan perancangan *protoype* alat untuk menunjang sistem yang akan dibangun agar dapat berfungsi seperti yang diinginkan.

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

- h. Pin BLA dihubungkan dengan jalur tegangan 5V pada *breadboard*.
  - i. Pin BLK dihubungkan dengan jalur GND pada *breadboard*.
5. Pemasangan potentiometer 10K ohm digunakan untuk mengatur tingkat kecerahan layar LCD. Kaki anoda potentiometer dihubungkan ke jalur GND *breadboard*, kaki katoda dihubungkan ke jalur positif *breadboard*, dan pin tengah dihubungkan ke pin V0 pada LCD.
  6. Pemasangan relay untuk menyalakan dan mematikan perangkat elektronik menggunakan jenis relay 4 channel dengan tegangan 5V.
  7. Jalur GND relay disambungkan ke pin Arduino GND. Sedangkan VCC relay dihubungkan ke pin Arduino *power* 5V.
  8. Sedangkan untuk jalur IN1 (*Input* 1), IN2, dan IN3 masing-masing dihubungkan ke pin nomor ~11, ~12, dan 13 secara berurutan.



**Gambar 3.7 Skema Alat Arduino**