

Terakreditasi SINTA Peringkat 4

Surat Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti No. 28/E/KPT/2019 masa berlaku mulai Vol.3 No. 1 tahun 2018 s.d Vol. 7 No. 1 tahun 2022

Terbit online pada laman web jurnal:
<http://publishing-widyagama.ac.id/ejournal-v2/index.php/jointecs>



Vol. 5 No. 2 (2020) 121 - 128

JOINTECS

(Journal of Information Technology and Computer Science)

e-ISSN:2541-6448

p-ISSN:2541-3619

Aplikasi Personal Assistant Berbasis Voice Command Pada Sistem Operasi Android Dengan NLP

Yoshua Constantin¹, Ucuk Darusalam², Novi Dian Nathasia³

Program Studi Sistem Informasi, Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional

¹joshua_constantine@gmail.com, ²ucuk.darusalam@gmail.com, ³novidian@civitas.unas.ac.id

Abstract

Currently the use of smartphones is very massive throughout the world, especially the android. The features needed are very easy to use, but most people are not able to use the maximum features available because it takes time to support with these features, one of the technologies that can be used to optimize android, namely voice recognition but the technology that has been developed by the vendor requires an internet connection to be able to run, for some users that don't have an internet network cannot enjoy the technology, to be able to overcome this problem required a personal assistant application that can be used to compile no internet connection, the idea obtained for creating this application is to perfecting the shortcomings of previous studies, with natural language processing techniques (NLP) making it easy in the process of making this application, applications that have been approved by testing and the results of these tests show how the installation can be done on the updated version of android 10 and 10/12 of the commands that were carried out successfully and average respond time is 0.054 second. All features can be run without the need for an internet connection.

Keywords: voice command; android voice assistant; personal assistant android.

Abstrak

Saat ini penggunaan *smartphone* sangat masif di seluruh dunia, khusus nya *android*. Fitur yang dimiliki sangat mudah dipakai, namun sebagian orang tidak mampu untuk menggunakan secara maksimal fitur-fitur yang ada karena membutuhkan waktu untuk berinteraksi dengan fitur tersebut, salah satu teknologi yang dapat dimanfaatkan untuk mengoptimalkan *android*, yaitu *voice recognition* namun teknologi yang sudah dikembangkan oleh vendor membutuhkan koneksi internet untuk dapat berjalan, untuk sebagian *user* yang tidak terdapat jaringan internet tidak bisa menikmati teknologi tersebut, untuk dapat mengatasi masalah ini dibutuhkan aplikasi *personal assistant* yang dapat digunakan ketika tidak ada koneksi internet sekalipun, ide yang didapat dari penciptaan aplikasi ini adalah untuk menyempurnakan kekurangan dari penelitian sebelumnya, dengan ada nya teknik *Natural Language Processing (NLP)* mempermudah dalam proses pembuatan aplikasi ini, aplikasi sudah diuji dengan serangkaian tes dan hasil dari tes tersebut menunjukkan bahwa instalasi dapat dilakukan di versi terbaru yaitu *android 10* dan *10* dari 12 perintah yang diuji berhasil dilakukan dengan rata-rata respons selama 0.054 detik. Semua fitur dapat dijalankan tanpa perlu koneksi internet.

Kata kunci: perintah suara; asisten suara *android*; asisten pribadi *android*.

© 2020 Jurnal JOINTECS

1. Pendahuluan

Di era modern salah satu aplikasi *voice command* yang terkenal adalah *siri* aplikasi tersebut terdapat pada *smartphone* berbasis *ios* yang sangat fenomenal, aplikasi tersebut dapat membantu *user*

berkomunikasi dengan *smartphone* melalui suara dan aplikasi tersebut dapat merespons perintah suara *user*. Jenis aplikasi serupa sudah tersedia yang dapat bersaing dengan aplikasi *smartphone* berbasis *ios* tersebut, aplikasi tersebut merupakan aplikasi

Diterima Redaksi : 11-01-2020 | Selesai Revisi : 21-01-2020 | Diterbitkan Online : 30-05-2020

berbasis sistem operasi *android* yang dikembangkan oleh *Google* yaitu "*google voice search*". Tetapi untuk menjalankan aplikasi tersebut dibutuhkan koneksi internet[1].

Beberapa jurnal penelitian yang mengenai hal ini telah banyak dilakukan oleh penelitian sebelumnya, penulis mengambil beberapa acuan sebagai referensi. Pada jurnal penelitian pertama *personal assistant with voice recognition intelligence* dirancang untuk membantu masyarakat terutama penyandang tunanetra yang dapat membantu untuk mengoperasikan *smartphone* melalui perintah suara, aplikasi tersebut juga memiliki kemampuan mengenali perintah suara tanpa koneksi internet[1].

Pada jurnal penelitian selanjutnya aplikasi pemutar musik menggunakan *speech recognition* dirancang untuk memainkan musik berdasarkan perintah suara yang sesuai dengan judul lagu, terdapat kendala seperti ketika internet mengalami gangguan akan membuat *google speech* menjadi sedikit lambat dalam memproses perintah suara[2]. Pada jurnal penelitian selanjutnya *rancang bangun aplikasi messaging berbasis voice interaction bagi penderita tunanetra pada sistem operasi android* dirancang melalui perancangan *sequence diagram*, hasil pengujian aplikasi tersebut mendapat nilai *Mean Opinion Score (MOS)* sebesar 3.7[3]. Pada jurnal penelitian selanjutnya rancang bangun sistem pengendali *smarthome* menggunakan mikrokontroler dengan *speech command* pada *smarthome android* dirancang menggunakan mikrokontroler dengan *speech command*, hasil pengujian dari pengujian pertama sampai pengujian ke tiga mendapatkan presentase keberhasilan sebesar 100% [4].

Pada jurnal penelitian yang selanjutnya *smart home* dengan *speech recognition* melalui *bluetooth* berbasis *android* mendapatkan presentase keberhasilan dalam uji coba perintah dengan 3 responden sebesar 100%, dengan waktu respons rata-rata selama 5.33 detik[5]. Pada jurnal penelitian yang selanjutnya pemanfaatan *speech recognition* pada *smartphone android* sebagai sistem pengontrolan pintu berbasis mikrokontroler dapat menjangkau maksimal 70cm untuk jarak yang bisa terdeteksi dan mendapat hasil gagal untuk kehandalan aplikasi dalam perintah suara baik dari tingkat kebisingan 30 dB sampai 80 dB[6].

Pada jurnal penelitian yang selanjutnya *single and multispeaker voice recognition in android* dapat bekerja dengan baik dan mendapatkan hasil akurasi sebesar 90.3% persamaan dalam mengidentifikasi 10 dari 20 pria untuk *single speaker recognition* dan mendapat hasil akurasi sekitar 34% untuk *multi speaker recognition*, pada pengujian ke 2 mendapat hasil akurasi sebesar 72.1% persamaan untuk *single speaker recognition* dan 14% untuk *multispeaker recognition*[7].

Pada penelitian jurnal penelitian yang selanjutnya *voice recognition based call and notification android app* menghasilkan beberapa poin yang dapat dilakukan dari kontrol aplikasi *android* dan menambahkan fitur yang saat itu tidak ada seperti menjawab panggilan telepon melalui perintah suara dan menolak panggilan telepon melalui perintah suara, aplikasi tersebut dapat membantu masyarakat terutama penyandang tunanetra dan masyarakat yang tidak dapat dengan mudah menggerakkan tangannya untuk menggenggam *smartphone*[8]. Pada jurnal penelitian yang selanjutnya *a voice controlled pick and place robotic arm vehicle using android application* berhasil sepenuhnya setelah diuji coba namun hanya dapat mengangkat bobot maksimal 250g[9]. Pada jurnal penelitian yang selanjutnya *android phone speech recognition sensed notice board display* menunjukkan bahwa dengan diperkenalkannya konsep teknologi tanpa kabel di ladang komunikasi dapat membuat komunikasi lebih efisien dan cepat, dengan bertambahnya efisiensi dapat menampilkan pesan dengan sedikit *error* dan hanya membutuhkan sedikit perawatan[10]. Pada jurnal penelitian yang selanjutnya *intelligent android voice assistant - a future requisite* menunjukkan di dalam survei yang dilakukan untuk pengujian mendapatkan hasil bahwa *user* lebih membutuhkan fungsi panggilan dengan presentase sebesar 80% dan hasil yang terendah kurang dari 4%[11]. Pada jurnal penelitian yang selanjutnya *offline voice assistant app for android based devices* diharapkan dapat membantu penyandang tunanetra dalam mengoperasikan sistem operasi *android*[12]. Pada jurnal penelitian yang selanjutnya *smart voice assistant: a universal voice control solution for non-visual access to the android operating system* disimpulkan dengan dikembangkannya perintah suara dapat membantu orang buta dalam menjalankan fungsi dari perintah suara seperti memanggil panggilan suara, mengirim sms dan mempermudah dalam pengoperasian perangkat[13]. Pada jurnal penelitian yang selanjutnya *voice controlled wheelchair using android technology* menghasilkan tingkat efisiensi yang tinggi untuk sistem yang diusulkan dibandingkan dengan sistem yang sudah ada dengan presentase lebih dari 80%[14]. Pada jurnal penelitian selanjutnya *integrating google speech recognition with android home screen application for easy and fast multitasking* mendapatkan hasil tingkat efisiensi yang tinggi untuk *user* melakukan pekerjaan dengan sistem operasi *android*[15].

Pemanfaatan teknologi *voice command* dengan teknik *Natural Language Processing (NLP)* pada *smartphone* sudah dapat digunakan lebih dari sekedar melakukan panggilan dan membalas pesan melalui perintah suara, namun hanya dapat digunakan jika terkoneksi dengan internet, maka

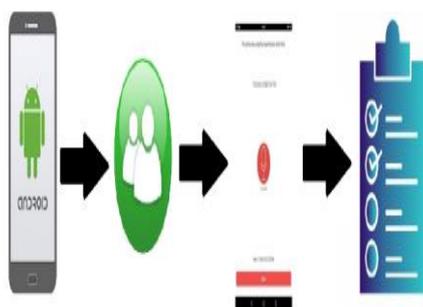
dibuatlah aplikasi *personal assistant* ini untuk menyempurnakan kekurangan dari penelitian sebelumnya. Tujuan dari penelitian ini dilakukan untuk membangun sebuah aplikasi yang dapat membantu *user* awam *android* dapat lebih mudah berinteraksi dengan perangkat sistem operasi *android* tanpa harus terkoneksi dengan internet.

2. Metode Penelitian

Teknik yang digunakan dalam penelitian ini adalah *Natural Language Processing (NLP)*. *NLP* merupakan salah satu bidang ilmu komputer, kecerdasan buatan dan bahasa (linguistik) yang berkaitan dengan interaksi antara komputer dan bahasa alami manusia, seperti bahasa Indonesia atau bahasa Inggris. Tujuan utama dari *NLP* adalah membuat mesin yang mampu mengerti dan memahami makna bahasa manusia lalu memberikan respons yang sesuai, seperti yang digunakan saat ini yaitu *speech recognition*, *text to speech*, *voice command* dan *machine translation*. Rancangan aplikasi ini terdiri dari beberapa tahapan antara lain: Analisa sistem, perancangan sistem, dan pengujian aplikasi.

2.1 Analisa Sistem

Pada tahap ini akan dijelaskan tentang rencana hasil penelitian yang dapat dikembangkan. Aplikasi yang dibangun akan diberi nama rancang bangun aplikasi *personal assistant* berbasis *voice command* bagi pengguna awam sistem operasi *android* dengan teknik *Natural Language Processing (NLP)*. Sistem yang sedang berjalan dapat dilihat pada gambar 1.



Gambar.1. Sistem yang sedang berjalan

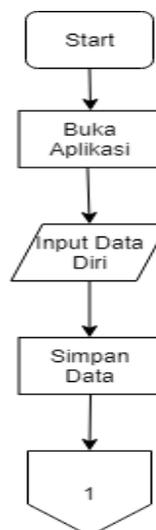
Dari Gambar 1 menjelaskan bahwa masyarakat di seluruh kalangan yang memiliki *smartphone* berbasis *android* sudah terinstal *google speech to text*, aplikasi *personal assistant* ini dapat digunakan dengan cara menekan *icon* mikrofon di tengah layar aplikasi lalu sistem akan memberi respons sesuai dengan perintah yang diberikan.

2.2 Perancangan Sistem

Perancangan aplikasi *personal assistant* dengan *voice command* berbasis *android* dirancang menggunakan 2 metode yaitu *software android studio* dan di gambarkan melalui *flowchart* sistem.

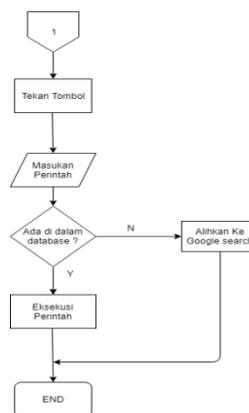
A. Diagram alur (*flowchart*)

Diagram alur atau *flowchart* adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program, diagram alur ini digunakan untuk menggambarkan proses kerja dari aplikasi *speech to text* dari awal *input* sampai akhir eksekusi, dapat dilihat pada Gambar 2 dan Gambar 3.



Gambar.2. *Flowchart* sistem aplikasi *personal assistant* bagian 1

Dari Gambar 2 merupakan alur sistem dimulai dari membuka aplikasi lalu *user* di minta untuk memasukkan data diri dan sistem akan merespons dengan menyimpan *data* diri yang sudah dimasukan.



Gambar.3. *Flowchart* sistem aplikasi *personal assistant* bagian 2

Dari Gambar 3 merupakan alur *flowchart* lanjutan proses dari Gambar 2, *user* merespons dengan menekan *icon* mikrofon lalu *user* memberikan perintah melalui suara dan sistem akan merespons dengan mengecek apakah perintah tersebut tersedia di dalam database atau tidak, lalu jika perintah tersedia di dalam *database* maka sistem akan mengeksekusi perintah yang diberikan, jika perintah tidak tersedia di dalam *database* maka sistem akan mengalihkan *user* ke *google search*.

2.3 Pengujian Aplikasi

2.3.1 Pengujian sistem aplikasi

Pada pengujian ini dilakukan dengan cara menguji kehandalan beberapa perintah yang diuji dan menghitung kecepatan respons aplikasi berdasarkan perintah yang diberikan.

2.3.2 Pengujian fungsional aplikasi

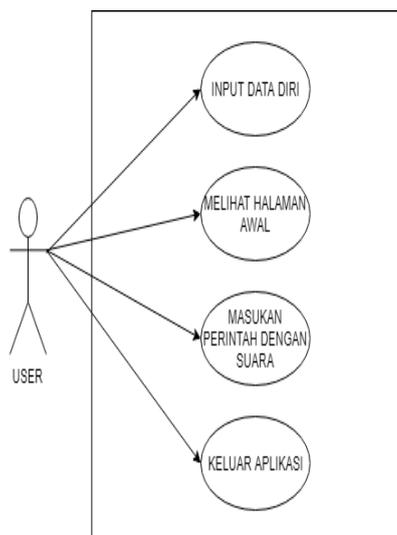
Pada pengujian ini dilakukan dengan instalasi aplikasi ke dalam perangkat yang memiliki versi *android* berbeda dimulai dari 7, 8, 9, 10.

3. Hasil dan Pembahasan

Perancangan sistem dalam penelitian menggunakan *Unified Modelling Language (UML)*. *UML* merupakan suatu cara yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak[1]. Penelitian ini menggunakan *UML* sebagai gambaran sistem yang meliputi *use case diagram*, *activity diagram* dan *class diagram*.

3.1 Use Case Diagram System

Use case diagram System merupakan diagram yang bersifat statis. Diagram ini menunjukkan interaksi antara *user* dan sistem, *user* yang terlibat di dalam sistem adalah masyarakat yang awam terhadap *android*. Berikut ini merupakan *use case diagram* aplikasi *personal assistant* berbasis *voice command* bagi pengguna awam sistem operasi *android* yang dirancang, dapat dilihat pada Gambar 4 dan dapat dilihat juga pada Tabel 1 sampai Tabel 4 untuk deskripsi dari *use case diagram* pada Gambar 4.



Gambar.4. Use Case Diagram

Deskripsi *use case*

1. Nama *use case* : *Input data diri*
 Aktor : *User*
 Tujuan *use case* : Memasukkan *data diri user*

Tabel 1. *Use case* input data diri

No	Actor Action	System Action
1	Aktor memasukkan <i>data diri</i>	Aplikasi menyimpan <i>data diri</i> dan beralih ke halaman awal

2. Nama *use case* : *Melihat halaman awal*
 Aktor : *User*
 Tujuan *use case* : *User* dapat melihat halaman awal aplikasi setelah memasukkan *data diri*

Tabel 2. *Use case* melihat halaman awal

No	Actor Action	System Action
1	Menekan <i>ok</i> setelah menginput <i>data diri</i>	Aplikasi akan beralih ke halaman awal

3. Nama *use case* : *Masukkan perintah dengan suara*
 Aktor : *User*
 Tujuan *use case* : *User* bisa memulai memberi perintah untuk menjalankan suatu fungsi dengan perintah suara

Tabel 3. *Use case* masukkan perintah dengan suara

No	Actor Action	System Action
1	Menekan <i>icon</i> mikrofon	<i>Google Speech To Text</i> akan keluar dan meminta <i>user</i> memberi perintah

4. Nama *use case* : *Keluar aplikasi*
 Aktor : *User*
 Tujuan *use case* : *User* dapat keluar dari aplikasi dengan memasukkan perintah “keluar”

Tabel 4. *Use case* keluar aplikasi

No	Actor Action	System Action
1	<i>User</i> memberi perintah untuk keluar	Aplikasi bertindak sepenuhnya berhenti

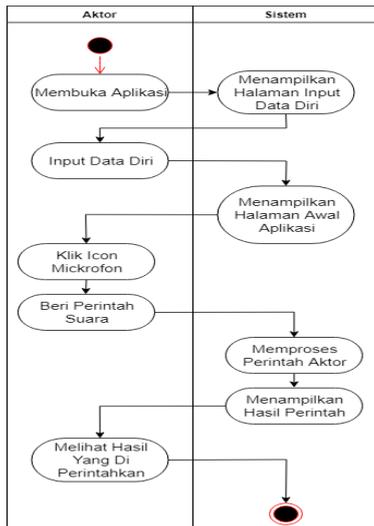
3.2 Activity Diagram

Activity diagram merupakan *diagram* yang menggambarkan alur kerja (*workflow*), Pada pembahasan ini terdapat beberapa *activity diagram* yang digunakan dalam membangun aplikasi *personal assistant* berbasis *voice command* bagi pengguna awam sistem operasi *android* dengan teknik *Natural Language Processing (NLP)*, antara lain :

- 3.2.1 *Activity diagram* melihat konten aplikasi
 Pada *activity* ini menjelaskan alur kerja aplikasi disaat pengguna mulai berinteraksi dengan sistem hingga

sistem menampilkan hasil proses perintah berupa menjalankan fungsi atau aplikasi, dapat dilihat pada Gambar 5.

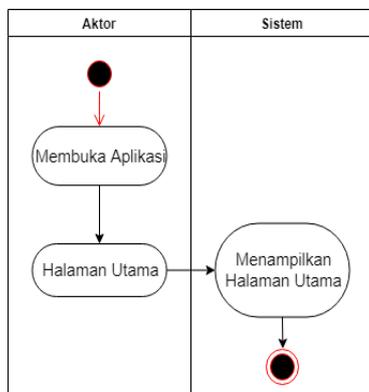
Pada Gambar 5 merupakan alur *activity* sistem dengan *user*, dimulai dengan *user* membuka aplikasi lalu sistem akan menampilkan halaman *input data* diri, lalu *user* di haruskan memasukkan *data* diri, lalu sistem akan merespons dengan menampilkan halaman awal aplikasi setelah itu *user* menekan *icon* mikrofon lalu memasukkan perintah suara dan sistem akan merespons dengan memproses perintah *user* lalu setelah diproses sistem akan menampilkan hasil dari perintah yang sudah dimasukan *user* dan *user* akan melihat hasil yang di perintahkan.



Gambar.5. Activity diagram melihat konten aplikasi

3.2.2 Activity diagram melihat halaman utama

Pada *activity* ini digambarkan proses untuk pengguna melihat informasi yang ditampilkan oleh sistem. *Activity diagram* untuk melihat halaman utama dapat dilihat pada Gambar 6.

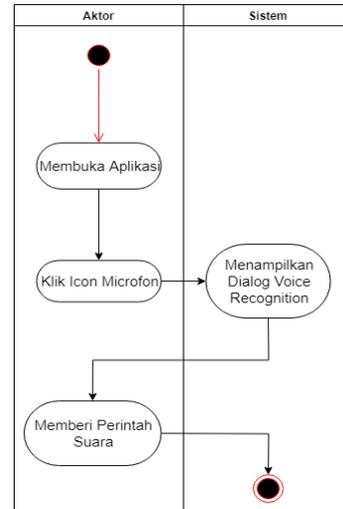


Gambar.6. Activity diagram melihat halaman utama

Pada Gambar 6 merupakan alur *activity* sistem dengan *user*, dimulai dengan *user* membuka aplikasi lalu *user* akan menuju ke halaman utama dan sistem akan merespons dengan menampilkan halaman utama.

3.2.3 Activity diagram memberi perintah

Pada *activity* ini digambarkan alur sistem *activity diagram* untuk memberi perintah melalui suara, dimulai dari membuka aplikasi sampai dengan memberi perintah suara.

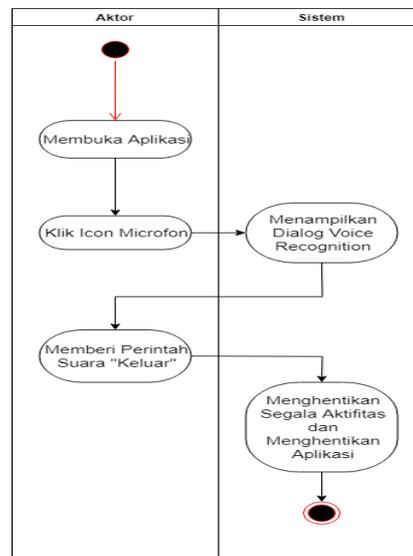


Gambar.7. Activity diagram memberi perintah

Pada Gambar 7 merupakan alur *activity* sistem dengan *user*, dimulai dengan *user* membuka aplikasi lalu *user* menekan *icon* mikrofon dan sistem akan merespons dengan menampilkan *dialog voice recognition* setelah itu *user* dapat memberi perintah melalui suara.

D. Activity diagram keluar dari aplikasi

Pada *activity* ini digambarkan alur sistem *activity diagram* untuk keluar dari aplikasi, dimulai dari membuka aplikasi kemudian memberi perintah “keluar” sehingga sistem menghentikan segala aktivitas dan menutup aplikasi.



Gambar.8. Activity diagram keluar aplikasi

Pada Gambar 8 merupakan alur sistem dengan *user*, dimulai dari *user* membuka aplikasi lalu *user* akan mengetuk *icon* mikrofon lalu sistem akan menampilkan

dialog *voice recognition* dan *user* akan memberi perintah suara berupa “keluar” lalu sistem akan merespons dengan menghentikan aplikasi.

pengujian ini menggunakan perangkat Zenfone Max Pro M1 dengan versi *android* terbaru yaitu *android* 10. hasil pengujian ini dapat dilihat pada Tabel 5.

3.1 Implementasi *Natural Language Processing (NLP)* Pada *Speech To Text*

```

Implementasi NLP
private void promptspeechInput() {
Intent intent = new
Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault());
intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
getString(R.string.speech_prompt));
try {
startActivityForResult(intent, REQ_CODE);
} catch (ActivityNotFoundException a) {
Toast.makeText(getApplicationContext(),
getString(R.string.speech_not_supported),
Toast.LENGTH_SHORT).show();
}
}
    
```

Gambar 9. Source Code *Speech to text* dengan Teknik *Natural Language Processing (NLP)*

Pada Gambar 9 merupakan sumber kode *text to speech* dengan teknik *natural language processing (NLP)*, dengan teknik *NLP voice command* dapat dijalankan dengan bahasa bebas sesuai dengan bahasa yang digunakan oleh perangkat sistem *android*.

3.4 Pembahasan

Aplikasi ini telah diuji dengan menggunakan beberapa perangkat *smartphone* dengan sistem operasi *android* yang berbeda untuk hasil implementasi, terdapat pengujian fungsional aplikasi untuk menguji setiap sistem yang tersedia di dalam aplikasi dan terdapat pengujian sistem aplikasi yang bertujuan untuk mengetahui keberhasilan dari instalasi aplikasi di berbagai perangkat *android* yang berbeda-beda dan versi *android* yang berbeda-beda.

3.4.1 Pengujian fungsional aplikasi

Pengujian ini dilakukan untuk menguji setiap sistem yang ada pada aplikasi, dengan memberi perintah berupa: Mengambil foto, mengambil *video*, menyalakan *bluetooth*, mematikan *bluetooth*, menyalakan *wi-fi*, mematikan *wi-fi*, membuka aplikasi *email*, membuka aplikasi musik, membuka web *amazon*, mode berdering, mode senyap, dan konversi suara ke dalam bentuk *file text* yang memungkinkan menyimpan *data* suara jika perintah tidak tersedia di dalam *database* aplikasi dan terdapat respons yang tercatat di dalam setiap eksekusi perintah yang diberikan tersimpan dalam bentuk satuan detik,

Tabel 5. Pengujian fungsional aplikasi

Aplikasi	Perintah yang diberikan	Hasil pengujian		
		Berhasil	Gagal	Respons kecepatan (detik)
<i>Personal assistant</i>	Mengambil foto	–	√	<i>Null</i>
	Mengambil <i>video</i>	–	√	<i>Null</i>
	Menyalakan <i>bluetooth</i>	√	–	0.044
	Mematikan <i>bluetooth</i>	√	–	0.093
	Menyalakan <i>wi-fi</i>	√	–	0.042
	Mematikan <i>wi-fi</i>	√	–	0.059
	Membuka aplikasi <i>email</i>	√	–	0.109
	Membuka aplikasi musik	√	–	0.008
	Membuka web <i>amazon</i>	√	–	0.031
	Mode berdering	√	–	0.089
	Mode senyap	√	–	0.117
	Konversi suara ke dalam bentuk <i>file text</i>	√	–	0.058

Dari Tabel 5 hasil yang didapat aplikasi *personal assistant* berbasis *voice command* ini mampu menjalankan hampir semua perintah pengujian, terdapat 2 perintah yang tidak dapat dijalankan yaitu mengambil foto dan mengambil *video*, respons kecepatan rata-rata di pengujian kurang dari 1 detik, menunjukkan respons aplikasi setelah diuji sangat cepat.

3.4.2 Pengujian sistem aplikasi

Pengujian aplikasi yang dilakukan dengan beberapa perangkat *android* berbeda, bertujuan untuk mengetahui keberhasilan dalam instalasi aplikasi terhadap perangkat *mobile android* yang memiliki spesifikasi yang berbeda. Pengujian ini dapat dilihat pada Tabel 6.

Tabel 6. Pengujian sistem aplikasi

No	Perangkat yang digunakan	Versi <i>android</i>	Hasil Pengujian	
			Berhasil	Gagal
1	Zenfone Max Pro M1	<i>Android 10</i>	√	–
2	Realme 2 Pro	<i>Android 9 Pie</i>	√	–
3	Zenfone Max Pro M1	<i>Android 8 Oreo</i>	√	–
4	Nvidia Shield K1	<i>Android 7.1.1 Nougat</i>	√	–

Dari Tabel 6 hasil yang didapatkan aplikasi *personal assistant* berbasis *voice command* dapat di instal dan dibuka di berbagai perangkat dengan versi *android* terbaru seperti 9(*pie*) dan 10.

3.4.3 Hasil Pengujian

Berdasarkan hasil pengujian yang telah dilakukan baik pengujian fungsional aplikasi dan pengujian sistem aplikasi, maka dapat rangkum sebagai berikut:

1. Respons kecepatan waktu untuk mengeksekusi suatu perintah suara sangat cepat apabila suara sudah sepenuhnya berhasil ditangkap oleh aplikasi dengan respons rata-rata kurang dari 1 detik.
2. Terdapat 2 perintah yang tidak berhasil dijalankan dengan perangkat yang diuji.
3. Gangguan akibat kebisingan dapat membuat aplikasi menjadi lambat dalam memproses suara yang masuk.
4. Aplikasi dapat di instal dan dibuka di versi *android* terbaru seperti 9 (*Pie*) dan 10, walau tidak semua fitur dapat dijalankan menggunakan versi *android* terbaru.

4. Kesimpulan

Berdasarkan hasil pengujian pada pembuatan aplikasi *personal assistant berbasis voice command* bagi pengguna awam sistem operasi *android* dengan teknik *Natural Language Processing (NLP)*, maka dapat diambil kesimpulan bahwa aplikasi dapat membantu pengguna awam *android* dengan tampilan antar muka yang sederhana, instalasi menggunakan perangkat dengan versi *android* terbaru 10 dapat dilakukan dan aplikasi dapat berjalan tanpa adanya koneksi internet.

Daftar Pustaka

- [1] K. V Kulhalli, K. Sirbi, M. A. J. Patankar, And Research, "Personal Assistant With Voice Recognition Intelligence," *Int. J. Eng. Res. Technol.*, Vol. 10, No. 1, Pp. 416–419, 2017.
- [2] K. Khairunizam, D. Danuri, And J. Jaroji, "Aplikasi Pemutar Musik Menggunakan Speech Recognition," *Inovtek Polbeng - Seri Inform.*, Vol. 2, No. 2, P. 97, 2017, Doi: 10.35314/Isi.V2i2.196.
- [3] F. A. Leo Tiofan Justicia, Herman Tolle, "Rancang Bangun Aplikasi Messaging Berbasis Voice Interaction Bagi Penderita Tunanetra Pada Sistem Operasi Android," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, Vol. 1, No. 7, Pp. 620–627, 2017.
- [4] A. R. Azka, E. D. Marindani, And R. D. Nyoto, "Rancang Bangun Sistem Pengendali Smarthome

- Menggunakan Mikrokontroler Dengan Speech Command Pada Smarthome Android," *J. Sist. Dan Tekn. Ol. Inf.*, Vol. 6, No. 2, P. 82, 2018, Doi: 10.26418/Justin.V6i2.24643.
- [5] P. Saputra, "Smart Home Dengan Speech Recognition Melalui Bluetooth Berbasis Android," *J. Elektron. Pendidik. Tek. Elektron.*, Vol. 7, No. 2, Pp. 38–55, 2018.
- [6] P. W. N. Banamtuan, H. Djahi, And A. A. Maggang, "Pemanfaatan Speech Recognition Pad Smartphone Android Sebagai Sistem Pengontrolan Pintu Berbasis Mikrokontroler."
- [7] Bosubabu Sambana, "Internet Of Things: Applications And Future Trends," *Int. J. Innov. Res. Comput. Commun. Eng. Internet Things Appl. Futur. Trends*, Vol. 5, No. 3, Pp. 5194–5202, 2017, Doi: 10.15680/Ijircce.2017.
- [8] R. Shyam, K. Vignesh, N. Vignesh, And C. Balakrishnan, "Voice Recognition Based Call And Notification Android App International Journal Of Modern Trends In Engineering And Research (Ijmeter)," Pp. 92–96, 2017, Doi: 10.21884/Ijmeter.2017.4058.Wni40.
- [9] N. U. Alka, A. A. Salihu, Y. S. Haruna, And I. A. Dalyop, "A Voice Controlled Pick And Place Robotic Arm Vehicle Using Android Application," No. 7, Pp. 207–215, 2017.
- [10] R. Agarwal, S. Singh, S. Yadav, And S. Bansal, "Android Phone Speech Recognition Sensed Voice Operated Notice Board Display," *Ijarccce*, Vol. 6, No. 4, Pp. 206–211, 2017, Doi: 10.17148/Ijarccce.2017.6439.
- [11] N. V. Vidyannagar, "Intelligent Android Voice Assistant - A Future Requisite," *Ijedr*, Vol. 5, No. 3, Pp. 337–339, 2017.
- [12] H. Om, "Offline Voice Assistant App For Android Based Devices," Vol. 6, No. 5, Pp. 54–59, 2017.
- [13] A. Bhalerao, S. Bhilare, A. Bondade, And M. Shingade, "Smart Voice Assistant : A Universal Voice Control Solution For Non-Visual Access To The Android Operating System," *Int. Res. J. Eng. Technol.*, Vol. 4, No. 2, 2017.
- [14] S. H. Wani, A. Hilal, And O. F. Khan, "Voice Controlled Wheelchair Using Android Technology," Vol. 6, No. 6, Pp. 420–424, 2017.
- [15] I. Satish And L. V. Kiran, "Integrating Google Speech Recognition With Android Home Screen Application For Easy And Fast Multitasking," No. May, Pp. 0–3, 2018.

Halaman ini sengaja dikosongkan