

Aplikasi *Virtual Assistant* Berbasis Android Untuk Lansia



PROGRAM STUDI D-4 TEKNIK KOMPUTER DAN JARINGAN
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2022

HALAMAN PENGESAHAN

Skripsi dengan judul “*Aplikasi Virtual Assistant Berbasis Android untuk Lansia*” oleh **Muhammad Fathurrahman** Nomor Induk Mahasiswa **42517035** telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar Diploma IV (D-4/S1 Terapan) pada Program Studi Teknik Komputer dan Jaringan Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Makassar, 23 September 2022

Mengesahkan,

Pembimbing I



Muh. Fajri Raharjo, S.T., M.T.
NIP. 19700521 199601 1 001

Pembimbing II



Dr. Ir Hafsa Nirwana., M.T.
NIP. 19640405 1990032002

Mengetahui,

Ketua Program Studi

Teknik Komputer dan Jaringan

Politeknik Negeri Ujung Pandang



Eddy Tungadi, S.T., M.T.
NIP. 19790823 201012 1 001

HALAMAN PENERIMAAN

Pada hari ini, Selasa tanggal 27 September 2022. Tim Penguji Ujian Sidang Skripsi telah menerima dengan baik skripsi oleh mahasiswa : **Muhammad Fathurrahman (42517035)** dengan judul **Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia.**

Makassar, 27 September 2022

Tim Penguji Sidang Skripsi :

1. Irmawati, S.T., M.T.	Ketua	()
2. Muhammad Nur Yasir Utomo, S.ST, M.Eng.	Sekretaris	()
3. Ir. Dahlia, M.T.	Anggota	()
4. Meylanie Olivya, S.T., M.T.	Anggota	()
5. Muh. Fajri Raharjo, S.T., M.T.	Pembimbing I	()
6. Dr. Ir. Hafsah Nirwana, S.T., M.T.	Pembimbing II	()

KATA PENGANTAR

Puji syukur tidak hentinya penulis panjatkan kepada Allah Subhanahu Wa Ta'alah atas limpahan berkah, rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini. Sholawat dan salam kepada Rasulullah Muhammad Shallallahu 'Alaihi Wa Sallam, sang revolusioner umat Islam, yang diutus sebagai rahmat bagi semesta alam.

Skripsi ini disusun guna memenuhi salah satu syarat untuk menyelesaikan studi dan memperoleh gelar Diploma IV (D-4/S1 Terapan) pada Program Studi Teknik Komputer dan Jaringan Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Penulis menyadari bahwa keberhasilan penyusunan skripsi ini tidak lepas dari bantuan berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, melalui kesempatan ini penulis menyampaikan penghargaan dan ucapan terima kasih kepada:

1. Direktur selaku Direktur Politeknik Negeri Ujung Pandang.
2. Bapak Ahmad Rizal Sultan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.
3. Bapak Eddy Tungadi, S.T., M.T. selaku Koordinator Program Studi D-IV Teknik Komputer dan Jaringan.
4. Bapak Muh. Fajri Raharjo, S.T., M.T. selaku pembimbing I dan Ibu Dr. Ir. Hafsah Nirwana, M.T. selaku pembimbing II atas segala ilmu, masukan, arahan, nasehat, motivasi, dukungan, bantuan, waktu dan kesabarannya dalam membimbing penulis hingga penelitian ini selesai.
5. Seluruh Staf Akademik Politeknik Negeri Ujung Pandang, serta Dosen dan Staf Jurusan Teknik Elektro, khususnya pada Program Studi Teknik Komputer dan Jaringan.
6. Teman-teman angkatan 2017 di Program Studi D-IV Teknik Komputer dan Jaringan maupun di seluruh Program Studi pada Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang, khususnya orang-orang yang memberikan dukungan dan kebersamaan detik-detik pendaftaran ujian sidang.
7. Semua pihak yang telah memberikan bantuan, baik itu bantuan moril maupun materiil yang tidak dapat disebutkan satu per satu.

Ucapan terima kasih dan penghargaan yang sangat besar penulis berikan kepada kedua orang tua penulis, yaitu Ayahanda Drs. Saharuddin, M.Pd.I dan Ibunda A. Nurbayyinah, S.Pd., M.M. beserta keluarga besar yang tidak hentinya memberikan dukungan, semangat dan do'a kepada penulis sehingga penulis bisa menyelesaikan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih terdapat banyak kekurangan, sehingga penulis mengharapkan kritik dan saran agar penulis dapat berkembang juga penelitian ini bisa semakin dikembangkan pada penelitian lainnya. Semoga apa yang penulis lakukan dapat bermanfaat bagi pembaca sehingga menghasilkan sumber daya manusia yang berkualitas dan berguna bagi bangsa dan negara.

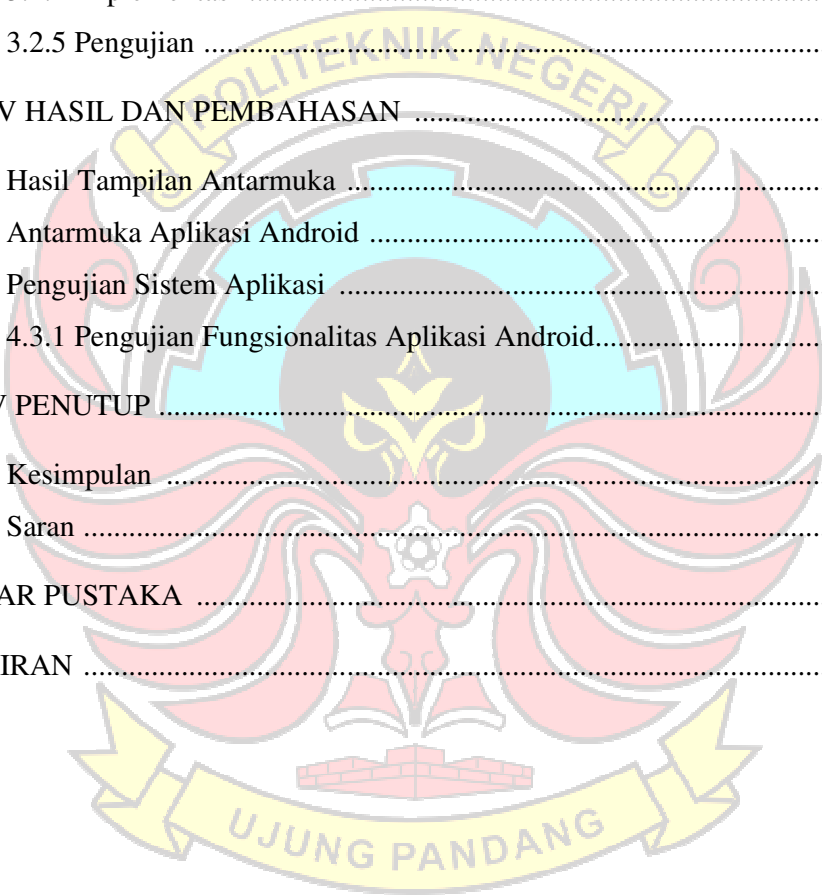
Makassar, 23 September 2022



DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PENERIMAAN	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR TABEL	viii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN	x
SURAT PERNYATAAN	xi
RINGKASAN	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Ruang Lingkup Penelitian	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA	5
2.1 <i>Artificial Intelligence</i>	5
2.2 <i>Visual Assistant</i>	5
2.3 <i>Natural Language Processing</i>	10
2.4 Android	10
2.5 Python	11
2.6 Kivy/KivyMD	11
2.7 Manula	11

BAB III METODE PENELITIAN	13
3.1 Tempat dan Waktu Penelitian	13
3.2 Metode Penelitian	13
3.2.1 Studi Literatur	14
3.2.2 Analisis Kebutuhan	15
3.2.3 Perancangan dan Desain	16
3.2.4 Implementasi	20
3.2.5 Pengujian	20
BAB IV HASIL DAN PEMBAHASAN	21
4.1 Hasil Tampilan Antarmuka	21
4.2 Antarmuka Aplikasi Android	21
4.3 Pengujian Sistem Aplikasi	29
4.3.1 Pengujian Fungsionalitas Aplikasi Android.....	29
BAB V PENUTUP	35
5.1 Kesimpulan	35
5.2 Saran	35
DAFTAR PUSTAKA	36
LAMPIRAN	39



DAFTAR TABEL

Tabel 3.1 Jadwal Penelitian	13
Tabel 3.2 Kebutuhan Perangkat Keras	15
Tabel 3.3 Kebutuhan Perangkat Lunak	16
Tabel 4.1 Pengujian Terhadap Tampilan Awal / Utama	29
Tabel 4.2 Pengujian Terhadap Tampilan Daftar Peningat	30
Tabel 4.3 Pengujian Terhadap Tampilan Tambahkan Peningat	31
Tabel 4.4 Pengujian Terhadap Tampilan Peningat Berbunyi	32



DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Metode Penelitian	14
Gambar 3.2 Diagram <i>Use Case</i>	16
Gambar 3.3 Diagram <i>Activity</i>	17
Gambar 3.4 <i>Mockup</i> Tampilan <i>Splash Screen</i>	18
Gambar 3.5 <i>Mockup</i> Tampilan Daftar Pengingat	18
Gambar 3.6 <i>Mockup</i> Tampilan Interaktif Berbicara	19
Gambar 3.7 <i>Mockup</i> Tampilan Pengingat Berbunyi	19
Gambar 4.1 Tampilan <i>Splash Screen</i>	22
Gambar 4.2 Tampilan Utama	23
Gambar 4.3 Tampilan Daftar Pengingat	24
Gambar 4.4 Tampilan Tambah Pengingat	25
Gambar 4.5 Tampilan Masukan Waktu Mode Geser	26
Gambar 4.6 Tampilan Masukan Waktu Mode Ketik	26
Gambar 4.7 Tampilan Masukan Tanggal Mode Sentuh	27
Gambar 4.8 Tampilan Masukan Tanggal Mode Ketik	27
Gambar 4.9 Tampilan Pengingat Berbunyi	28

DAFTAR LAMPIRAN

Lampiran 1 Tampilan <i>Splash Screen</i>	38
Lampiran 2 Perizinan Audio	38
Lampiran 3 Tampilan Utama	39
Lampiran 4 Memberikan perintah suara dan balasan dari aplikasi	39
Lampiran 5 Tampilan Daftar Pengingat	40
Lampiran 6 Menghapus salah satu item pengingat	40
Lampiran 7 Tampilan Tambahkan Pengingat	41
Lampiran 8 Menyimpan data pengingat	41
Lampiran 9 Kembali ke tampilan daftar pengingat dan menampilkan item pengingat yang ditambahkan	42
Lampiran 10 Tidak memasukkan nama pengingat pada kolom nama pengingat	42
Lampiran 11 Tampilan Pengingat Berbunyi	43
Lampiran 12 Pengkodean Python	43
Lampiran 13 Pengkodean Kivy	47
Lampiran 14 Konfigurasi Buildozer untuk <i>Deploy</i> ke Android	49
Lampiran 15 <i>Build and Deploy kivy file</i> menjadi .apk di Google Colab	51
Lampiran 16 Dokumentasi Uji Coba Aplikasi <i>Virtual Assistant</i> untuk Lansia	53

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Muhammad Fathurrahman

NIM : 425 17 035

menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul **“APLIKASI *VIRTUAL ASSISTANT* BERBASIS ANDROID UNTUK LANSIA”** merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 23 September 2022



Muhammad Fathurrahman

NIM. 425 17 035

APLIKASI *VIRTUAL ASSISTANT* BERBASIS ANDROID UNTUK LANSIA

RINGKASAN

Keterbatasan lansia dalam mengingat bukanlah sebuah penyakit melainkan kondisi manusia yang mulai memasuki penurunan fungsi organ, salah satunya yaitu penurunan daya ingat. Pada kondisi tersebut lansia membutuhkan bantuan orang terdekatnya dalam melakukan kesehariannya. Namun keluarga atau orang terdekat dari lansia tidak bisa menemani lansia seharian penuh dikarenakan orang terdekat atau keluarga mempunyai aktivitas lain yang harus dikerjakan.

Penelitian ini bertujuan untuk membuat suatu aplikasi android yang bisa digunakan lansia untuk mengingat rutinitas kegiatan, sehingga dengan adanya aplikasi android ini bisa membantu lansia untuk mengingat kegiatan kesehariannya, seperti mengingatkan untuk makan, minum obat dan olahraga.

Aplikasi android dirancang dan dibangun dengan menggunakan bahasa pemrograman Python, *library* Kivy dan KivyMD. Dengan menggunakan Python dan Kivy dibuat sistem pengingat dan *speech recognition* sederhana untuk dijalankan pada rancangan aplikasi android. Aplikasi android yang dibangun selanjutnya diimplementasikan dan diujikan terhadap lansia yang dibantu dengan keluarganya dengan menggunakan metode *Black Box Testing*, yang merupakan metode pengujian fungsional terhadap aplikasi.

Hasil pengujian menunjukkan bahwa aplikasi *virtual assistant* berbasis android untuk lansia mampu berjalan dengan baik. Berjalan dengan tampilan sederhana, menjalankan pengingat makan, pengingat minum obat dan pengingat olahraga untuk lansia. Sehingga dari hasil pengujian *Black Box Testing* menunjukkan bahwa aplikasi *virtual assistant* dapat digunakan dengan baik.

Kata kunci : Android, Python, Kivy, *Speech Recognition*, Lansia

BAB I PENDAHULUAN

1.1 Latar Belakang

Artificial Intelligence yang biasa disingkat dengan AI

atau dalam bahasa Indonesia disebut dengan kecerdasan buatan.

”Kecerdasan buatan dapat didefinisikan sebagai cabang ilmu komputer yang mempelajari otomisasi tingkah laku cerdas” (Adityawarman dkk. 2010 dalam Setiawan, 1993). AI pertama kali diperkenalkan pada konferensi di Dartmouth College pada tahun 1956 oleh Jhon McCarthy bersama peneliti yang lain. Definisi AI pertama kali dikemukakan oleh Jhon McCarthy (1956) menyatakan bahwa “*the science and engineering of making intelligent machines*”. Artinya ilmu dan teknik untuk membuat mesin yang cerdas. Sampai saat ini, AI memiliki berbagai jenis pengembangan salah satunya yaitu *Virtual Assistant*. Kelahiran *Virtual Assistant* dimulai pada tahun 1990an ketika muncul teknologi pengenalan suara digital hingga peluncuran ponsel cerdas pertama IBM Simon pada tahun 1994.

Virtual Assistant basisnya menggunakan metode *Natural Language Programming* (NLP). *Virtual Assistant* bisa diterapkan dalam berbagai bentuk seperti yang menggunakan audio, teks, gambar, *User Interface*, kontrol, dan konten web sebagai media (Beaver dan Cynthia Freeman, 2016). *Virtual Assistant* biasanya digunakan untuk membantu pekerjaan manusia seperti menyalakan atau mematikan lampu dengan suara, memutar musik, *chatbot*, dan bisnis. *Virtual Assistant* berbasis pengenalan suara banyak dikembangkan, seperti Alexa milik Amazon. Namun, pengembangannya masih kurang untuk

kebutuhan manula. Manula adalah singkatan dari manusia lanjut usia yang berarti manusia yang berumur tua. Menurut UU nomor 13 tahun 1998 tentang Kesejahteraan Lanjut Usia mengatakan bahwa “Lanjut usia adalah seseorang yang telah mencapai usia 60 tahun (enam puluh) ke atas.”

Manula bukanlah suatu penyakit melainkan kondisi seorang manusia memasuki fase penurunan fungsi organ, yang salah satu penurunan fungsi organ adalah daya ingat atau pikun. Pikun sendiri pada umumnya merupakan kondisi berkurangnya daya ingat atau memori seseorang dan kebanyakan lansia mengalami pikun. Direktur Regional Alzheimer Asia Pasifik sekaligus Penggagas Alzheimer's Indonesia (ALZI), DY Suharya mengatakan, masyarakat seharusnya tidak maklum dengan pikun, karena bisa jadi kondisi tersebut merupakan pertanda bahwa mereka telah mengalami demensia atau Alzheimer (Pranita, 2020). Dengan kondisi tersebut, manula memerlukan perhatian lebih dari keluarga atau pun masyarakat sekitar. Biasanya manula yang memiliki keluarga akan dirawat oleh keluarganya sedangkan yang tidak memiliki keluarga akan dirawat di Panti Jompo.

Melihat dari keterbatasan lansia, pernah dilakukan penelitian di tahun 2018 dengan judul Aplikasi Caring Assistance Untuk Manula Berbasis Android Dan Raspberry Pi (Chan dkk., 2018). Aplikasi ini mampu melakukan *live streaming*

antara lansia dan perawat jika lansia berada pada kondisi darurat dengan menekan tombol *switch*, serta aplikasinya juga memiliki daftar rumah sakit terdekat, apotek terdekat dan memiliki jadwal minum obat untuk lansia (Chan dkk., 2018). Pada penelitian lainnya di tahun 2019 yang berjudul Aplikasi

Asisten Untuk Lansia Dengan Memanfaatkan Smartphone Berbasis Android, membantu lansia untuk melakukan aktivitasnya seperti berolahraga dengan memperagakan gerakan olahraga atau senam dan juga aplikasi ini mampu melacak lokasi lansia berada (Sastriya dkk., 2019).

Virtual assistant merupakan salah satu teknologi yang dikembangkan untuk memudahkan pekerjaan manusia sebagai asisten virtual dengan *machine learning*, *voice recognizer*, *natural language processing* sebagai teknologi yang ada dibalik *virtual assistant*. Salah satu penelitian mengenai *virtual assistant* dengan metode *voice recognizer* pernah dilakukan untuk implementasi kendali perangkat listrik rumah tinggal. Dengan *virtual assistant* metode *voice recognizer* perangkat listrik seperti lampu, kipas dan pintu otomatis yang digunakan pada rumah tinggal dapat dikendalikan dengan mengucapkan kalimat tertentu (Gerung dkk., 2019).

Dari beberapa penelitian tersebut belum ada penelitian aplikasi android yang bisa membantu lansia untuk berdiskusi atau berkomunikasi guna mencegah pikun terhadap lansia, juga mengingatkan rutinitias. Untuk itu pada penelitian ini akan dibuatkan *virtual assistant* dengan metode *voice recognizer* berbasis aplikasi android yang dapat digunakan lansia untuk berdiskusi, mengingatkan jadwal minum obat, dan rutinitasnya.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah bagaimana membuat aplikasi android yang bisa membantu lansia seperti mengingatkan rutinitas dan berkomunikasi sederhana.

1.3 Ruang Lingkup Penelitian

Ada pun ruang lingkup penelitian yaitu aplikasi ini digunakan untuk membantu lansia mengingatkan rutinitas seperti waktu minum obat, waktu olahraga, waktu istirahat dan bisa berkomunikasi sederhana dengan lansia. Aplikasi ini nantinya akan dibuat dengan menggunakan bahasa pemrograman python.

1.4 Tujuan Penelitian

Ada pun tujuan dari penelitian ini yaitu :

1. Mengidentifikasi teknologi *Virtual Assistant* yang banyak diterapkan khususnya berbasis android.
2. Membangun teknologi *Virtual Assistant* berbasis android untuk Lansia.
3. Menganalisis penerapan *Virtual Assistant* berbasis android pada Lansia.

1.5 Manfaat Penelitian

1. Teridentifikasinya teknologi *Virtual Assistant* yang banyak diterapkan khususnya berbasis android.
2. Membantu lansia dengan teknologi *Virtual Assistant* berbasis android.
3. Terbangun teknologi *Virtual Assistant* berbasis android untuk lansia.

BAB II TINJAUAN PUSTAKA

2.1 *Artificial Intelligence*

Artificial intelligence yang biasa disingkat dengan AI atau dalam bahasa Indonesia disebut dengan kecerdasan buatan. "Kecerdasan buatan dapat didefinisikan sebagai cabang ilmu komputer yang mempelajari otomisasi tingkah laku cerdas" (Setiawan, dalam Adityawarman dkk. 2010). AI pertama kali diperkenalkan pada tahun 1956 oleh Jhon McCarthy dalam konferensi di Dartmouth College. Jhon McCarthy (1956) menyatakan bahwa AI adalah "*the science and engineering of making intelligent machines*". AI atau kecerdasan buatan berarti membuat mesin bisa bekerja secara otomatis sesuai dengan tingkah laku makhluk hidup. Dalam perkembangannya AI banyak membantu manusia, diantaranya yaitu penerapan AI dalam bidang kesehatan, seni, pendidikan, dll. Salah satu perkembangan dari AI adalah *Virtual Assistant*.

2.2 *Virtual Assistant*

Virtual Assistant merupakan salah satu pengembangan dari AI yang membantu pekerjaan manusia. Bernard dan Alexandre Arnold (2019) menyatakan bahwa "*virtual assistants are intended to interact with human users in a natural way, which is by essence cognitive, linguistic and collaborative*".

Artinya *virtual assistant* dimaksudkan untuk berinteraksi dengan pengguna manusia secara natural, yang pada dasarnya berbentuk kognitif, linguistik dan kolaboratif.

Teknologi *virtual assistant* menggunakan data yang banyak sehingga menggunakan *machine learning, voice recognizer, natural language processing*

(NLP) sebagai teknologi yang ada dibalik *virtual assistant*. *Virtual assistant* banyak digunakan untuk metode seperti teks, suara atau gambar. Penerapan *virtual assistant* dengan metode teks banyak digunakan dalam bentuk *chatbot*, sedangkan suara dalam bentuk *voice recognizer*.

Perusahaan teknologi yang sudah memiliki teknologi *virtual assistant* berbasis *voice recognizer* diantaranya Amazon dengan Alexa, Microsoft dengan Cortana, Google dengan Google Assistant, Apple dengan Siri. Beberapa *virtual assistant* yang bisa digunakan pada *smartphone* seperti Google Assistant dan Siri khusus pengguna Apple. Dengan *virtual assistant* berbasis *voice recognizer*, manusia bisa mengendalikan perangkat seperti *speaker* rumah, otomasi rumah, pemutaran media atau pun tugas dasar lainnya hanya dengan perintah suara.

Di Indonesia sudah ada beberapa penelitian tentang *virtual assistant*, baik itu dengan teknik teks atau pun suara. Pengimplementasian *virtual assistant* ini diantaranya digunakan sebagai media informasi lembaga, pelayanan perusahaan, kebutuhan personal.

Chatbot yang digunakan sebagai *virtual assistant* diimplementasikan pada Balai Penelitian Tanaman Pemanis dan Serat (BALITTAS). *Chatbot virtual assistant* dengan nama ALITTA berperan sebagai pusat informasi BALITTAS yang dapat membantu masyarakat dalam mencari informasi secara flexibel dan

tidak terikat waktu yang terdapat di BALITTAS (Yuniar dan Heri Purnomo, 2019).

Chatbot diimplementasikan sebagai *virtual assistant* untuk *live chat*. *Chatbot* ini diimplementasikan sebagai *customer service live chat* di PT. Garena

Indonesia dikarenakan pandemi covid-19 yang mengharuskan untuk merumahkan seluruh karyawan *customer service* untuk mencegah penularan virus. Tanpa *chatbot*, *user* atau pemain dari PT. Garena Indonesia harus antri menggunakan tiket laporan untuk menggunakan layanan *customer service*.

Natural Language Processing digunakan pada *chatbot* untuk memahami dan merespon pesan yang dikirim oleh *user* saat terhubung dengan layanan *Live Chat* PT. Garena Indonesia (Tarigan dan Rizki Ramadhan, 2020).

Penerapan *virtual assistant* dengan teknik *chatbot* sebagai media informasi pada ekowisata Taman Buru Masigit Kareumbi (TBMK). Penerapan *chatbot* ini dilakukan karena pengelola ekowisata TBMK yang kesulitan dalam menginformasikan kegiatan, fasilitas-fasilitas, serta reservasi ekowisata TBMK kepada pengunjung dan calon pengunjung. Pembangunan *virtual assistant chatbot* dilakukan dengan aplikasi Line. *Chatbot* ini menggunakan metode *forward chaining* untuk memberikan alur percakapan antara pengguna dengan bot, jaro-winkler untuk mengatasi kesalahan pengetikan dan singkatan (Wafa, 2020).

Virtual assistant metode *voice recognizer* diimplementasikan untuk kendali perangkat listrik rumah tinggal. Dengan *virtual assistant voice recognizer* perangkat listrik seperti lampu, lampu dimmer, kipas, dan pintu otomatis yang digunakan pada rumah tinggal bisa dikendalikan dengan mengucapkan kalimat tertentu. Pemantauan perangkat listrik dilakukan pada Blynk dan Scada (Gerung dkk., 2019).

Voice command dengan teknik NLP pada *smartphone* dapat digunakan lebih dari pada hanya untuk melakukan panggilan dan perpesanan dengan perintah suara, namun harus terhubung dengan internet. Dibuatlah *personal assistant* pada sistem android yang bisa digunakan dengan *voice command* untuk menjalankan fitur pada *smartphone* tanpa harus terhubung dengan internet (Constantin dkk., 2020).

Google Assistant digunakan sebagai *virtual assistant* pada sistem kendali kompor pintar. Kompor pintar bisa dikendalikan dengan suara atau dengan metode *voice recognizer* dengan bantuan Google Assistant pada *smartphone* dan tambahan aplikasi berupa *Auto Voice*, *Auto Arduino*, *Tasker*, *Auto App*, *Join*, dan yang terakhir IFTTT. Kompor pintar dikendalikan dengan perintah suara seperti kompor menyala, kompor mati, api besar, api sedang, api kecil (Setiawan, 2019).

Dalam pembuatan *virtual assistant* berbasis *voice recognizer* dapat berinteraksi kognitif dengan manusia, *virtual assistant* perlu untuk mengenali tindakan yang dilakukan oleh pengguna. *Virtual assistant* perlu memahami emosi, menampilkan emosi, atau norma yang dapat dipahami manusia

Virtual assistant berbasis gerak pada *smartphone* android membantu lansia untuk melakukan aktivitasnya seperti berolahraga dengan memperagakan gerakan olahraga atau senam. Aplikasi asisten ini menggunakan sensor *accelerometer* dan *gyroscope* untuk mengetahui pergerakan dari lansia apakah sudah sesuai dengan gerakan yang dicontohkan. Aplikasi ini juga mampu melacak lokasi lansia berada (Sastriya dkk., 2019).

Salah satu perusahaan yang memiliki *virtual assistant* adalah Amazon yang memiliki Alexa. Alexa bisa digunakan pada perangkat amazon atau pun perangkat pihak ketiga yang bekerja sama dengan amazon. Seperti yang dikatakan oleh Tulshan (2019) bahwa Alexa terintegrasi dengan teknologi-teknologi dari Amazon, bisa berjalan pada perangkat android dan iOS termasuk pengeras suara, *headphone*, *smart watches*, perangkat *smarthome* dan yang lainnya.

2.3 Natural Language Processing

Natural language processing atau biasa disingkat NLP adalah salah satu bidang kecerdasan buatan. Mesin atau komputer dapat menganalisis atau memahami makna dari bahasa manusia dengan cara yang cerdas dan berguna. NLP dapat dimanfaatkan oleh pengembang untuk mengatur dan menyusun pengetahuan agar mesin atau komputer mampu melakukan tugas-tugas yang berkaitan dengan kalimat atau bahasa. NLP memungkinkan komputer membaca teks, mendengar ucapan, menafsirkannya, mengukur sentimen, dan menentukan bagian mana yang penting sesuai dengan yang dipahami oleh manusia.

NLP adalah pengolahan otomatis atau semi-otomatis bahasa manusia. NLP erat kaitannya dengan linguistik, kognitif, psikologi, fisiologi, dan matematika. Dalam domain ilmu komputer khususnya, NLP berkaitan dengan teknik *compiler*, teori 10 bahasa formal, interaksi antar manusia dan komputer, *machine learning*, dan teorema pembuktian (Soeriawidjaya, 2019).

2.4 Android

Android adalah sistem operasi berbasis linux untuk perangkat *mobile* atau *smartphone*. Pengembangan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Serangkaian aplikasi inti Android antara lain klien email, program SMS, kalender, peta, browser, kontak, dan lain-lain.

Android menyediakan platform *open source* sehingga pengembang Android menawarkan kemampuan untuk membangun aplikasi yang lebih baik dan inovatif. Pengembang bebas untuk mengambil keuntungan dari perangkat keras, akses informasi lokasi, menjalankan *background services*, mengatur alarm, tambahkan pemberitahuan ke status bar, dan banyak lagi (Rasjid, 2010).

2.5 Python

Python adalah bahasa pemrograman tingkat tinggi dengan semantik dinamis yang terinterpretasi dan berorientasi objek. Python mendukung modul dan paket, yang mendorong modularitas program dan penggunaan kembali kode. Python tidak memiliki atribut keamanan (seperti: *public/private/protected*), sehingga programnya menjadi lebih sederhana dan mudah dimengerti. Python juga sangat dinamis sehingga bidang atau atribut bisa dibuat dengan cepat, yang tidak dapat dilakukan di JAVA atau C ++ (Bogdanchikov *et al.*, 2013).

2.5.1 Kivy / KivyMD

Kivy adalah *library open-source* dari Python yang bisa digunakan untuk membangun aplikasi *cross platform*. Kivy bisa membuat aplikasi Python dapat berjalan pada Windows, Linux, macOS, iOS dan Android. Kivy *framework* tidak perlu memerlukan spesifikasi komputer yang tinggi untuk

mengoperasikannya serta dapat berjalan *multi platform*, baik itu pada *desktop* ataupun *mobile* (Tobing dan Magdalena Ariance Ineke Pakereng, 2021). Selain Kivy, terdapat pula KivyMD (*Kivy Material Design*) yang merupakan *framework* yang memudahkan dalam mendesain tampilan aplikasi yang akan dibangun.

2.6 Manula

Manula adalah singkatan dari manusia lanjut usia yang berarti manusia yang berumur tua. Manula juga biasa disebut dengan lansia. Menurut UU nomor 13 tahun 1998 tentang Kesejahteraan Lanjut Usia mengatakan bahwa “Lanjut usia adalah seseorang yang telah mencapai usia 60 tahun (enam puluh) ke atas.” Manula bukanlah suatu penyakit melainkan kondisi seorang manusia memasuki fase penurunan fungsi organ. Penurunan fungsi organ diantaranya penurunan fungsi motorik yang umum dialami oleh manula seperti susah berjalan. Penurunan fungsi sensorik yang umum dialami oleh manula seperti gangguan pendengaran, gangguan penglihatan dan penurunan daya ingat alias pikun.

Pikun sendiri pada umumnya merupakan kondisi berkurangnya daya ingat atau memori seseorang dan kebanyakan lansia mengalami pikun. Direktur Regional Alzheimer Asia Pasifik sekaligus Penggagas Alzheimer's Indonesia (ALZI), DY Suharya mengatakan, masyarakat seharusnya tidak maklum dengan pikun, karena bisa jadi kondisi tersebut merupakan pertanda bahwa mereka telah mengalami demensia atau Alzheimer (Pranita, 2020).

Melihat dari keterbatasan lansia, pernah dilakukan penelitian di tahun 2018 dengan judul Aplikasi Caring Assistance Untuk Manula Berbasis Android Dan Raspberry Pi. Aplikasi ini mampu melakukan *live streaming* antara lansia dan perawat jika lansia berada pada kondisi darurat dengan menekan tombol *switch*, serta aplikasinya juga memiliki daftar rumah sakit terdekat, apotek terdekat dan memiliki jadwal minum obat untuk lansia (Chan dkk., 2018).



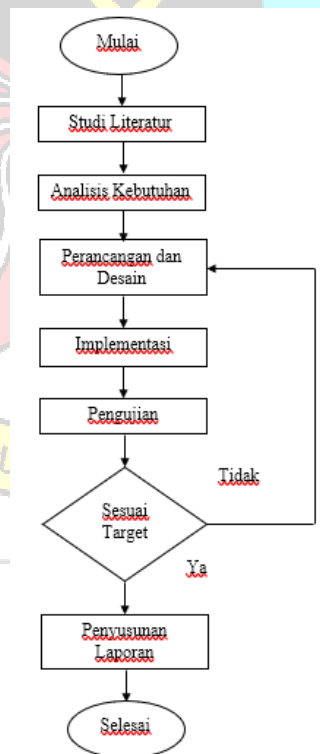
BAB III METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Laboratorium Tugas Akhir, Program Studi D4 Teknik Komputer dan Jaringan, Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang. Waktu penelitian ini dimulai dari bulan September 2021 sampai Februari 2022.

3.2 Metode Penelitian

Metode penelitian ini dilakukan dengan bertahap dan sistematis untuk mendapatkan hasil sesuai dengan tujuan penelitian. Berikut adalah gambar diagram alir dari metode penelitian yang akan dilakukan.



Gambar 3.1 Diagram Alir Metode Penelitian

3.2.1 Studi Literatur

Penelitian ini diawali dengan studi literatur, yaitu mencari dan membaca informasi baik itu yang bersumber dari jurnal penelitian, tugas akhir, atau internet yang bisa menjadi referensi atau dasar teori untuk melakukan penelitian.

Dengan mendapatkan informasi dari sumber yang disebutkan, diharapkan bisa memudahkan dalam penelitian.

3.2.2 Analisis Kebutuhan

Pada tahapan ini, dibutuhkan analisis kebutuhan untuk memenuhi kebutuhan lansia dan kebutuhan fungsional perancangan aplikasi. Tahapan ini dilakukan agar hasil perancangan aplikasi sesuai dengan identifikasi masalah yang telah ditentukan.

Ada pun kebutuhan lansia yaitu :

1. Aplikasi dengan volume suara yang besar.
2. Aplikasi dengan tampilan yang sederhana, *simple* (tidak menggunakan banyak kalimat, desain, dan warna).

Pada penelitian ini dibutuhkan perangkat keras (*hardware*) dan perangkat lunak (*software*). Ada pun perangkatnya di bawah ini :

- a. Kebutuhan perangkat keras

Tabel 3.1 Kebutuhan Perangkat Keras

Nomor	Perangkat Keras	Deskripsi
1.	Laptop Asus - Processor intel core i5 8 th Gen	Digunakan untuk menjalankan perangkat lunak untuk membangun aplikasi

	<ul style="list-style-type: none"> - Memori 8192MB RAM - <i>Hard disk</i> 1000 GB 	
2.	<i>Smartphone</i> <ul style="list-style-type: none"> - Sistem operasi android 11 - RAM 8 GB 	Digunakan untuk menjalankan aplikasi yang dibangun

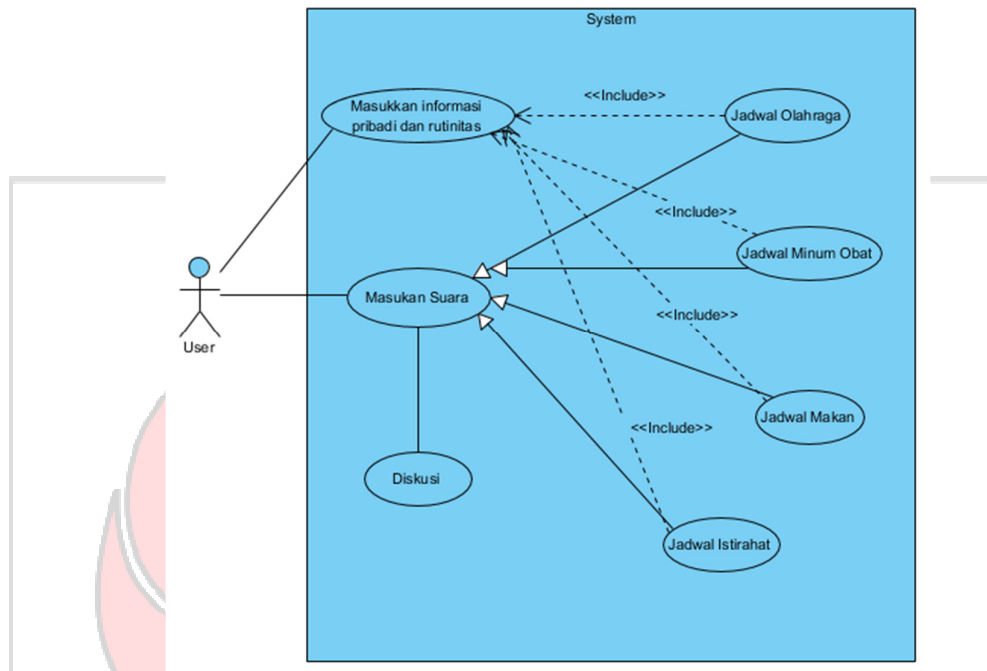
b. Kebutuhan perangkat lunak

Tabel 3.2 Kebutuhan Perangkat Lunak

Nomor	Perangkat Lunak	Deskripsi
1.	Sistem operasi Windows 10	Sistem operasi yang digunakan pada laptop penelitian.
2.	PyCharm / Visual Studio Code	Merupakan <i>software</i> yang digunakan untuk menulis kode dalam membangun aplikasi
3.	Python	Bahasa pemrograman yang akan digunakan untuk membangun aplikasi
4.	Kivy/KivyMD	<i>Library</i> Python digunakan untuk membangun dan menjalankan aplikasi dengan menggunakan bahasa pemrograman Python
4.	Microsoft Office Word versi 2103	Digunakan untuk menyusun laporan penelitian

3.2.3 Perancangan dan Desain

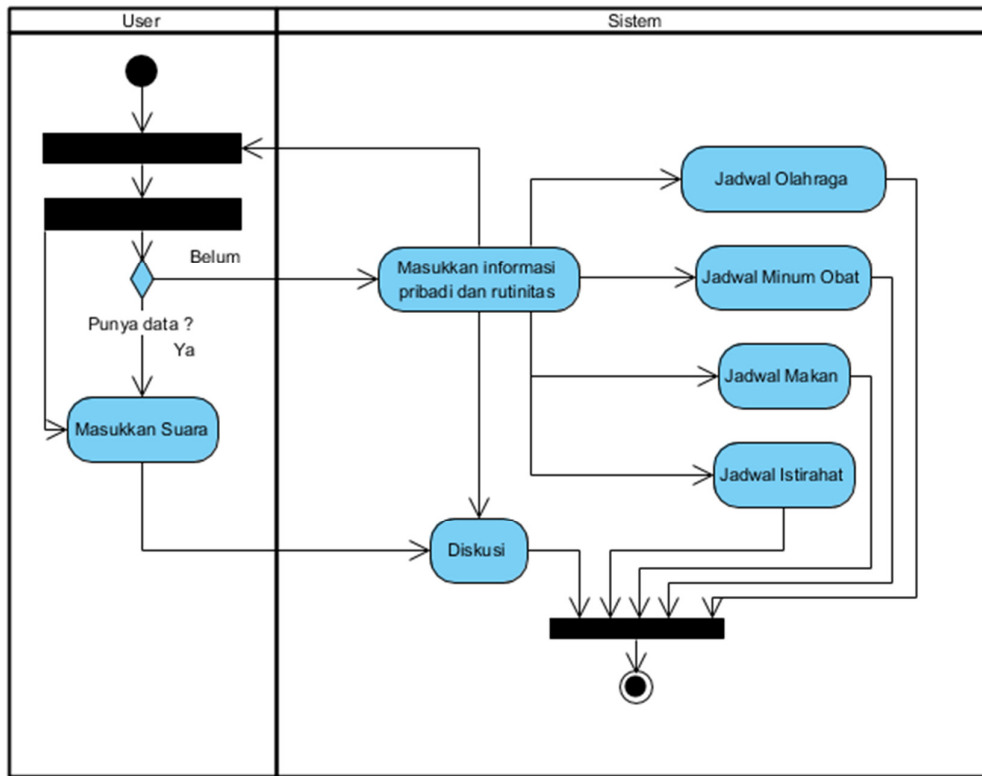
a. Use Case Diagram



Gambar 3.2 Use Case Diagram

Pada gambar 3.2 menampilkan diagram *use case*. Pada diagram tersebut menampilkan 1 aktor berupa user dan sebuah sistem. Di dalam sistem terdapat *use case* Masukkan informasi pribadi dan rutinitas, masukan suara, diskusi, jadwal olahraga, jadwal minum obat, jadwal makan dan jadwal istirahat.

b. Acitivity Diagram

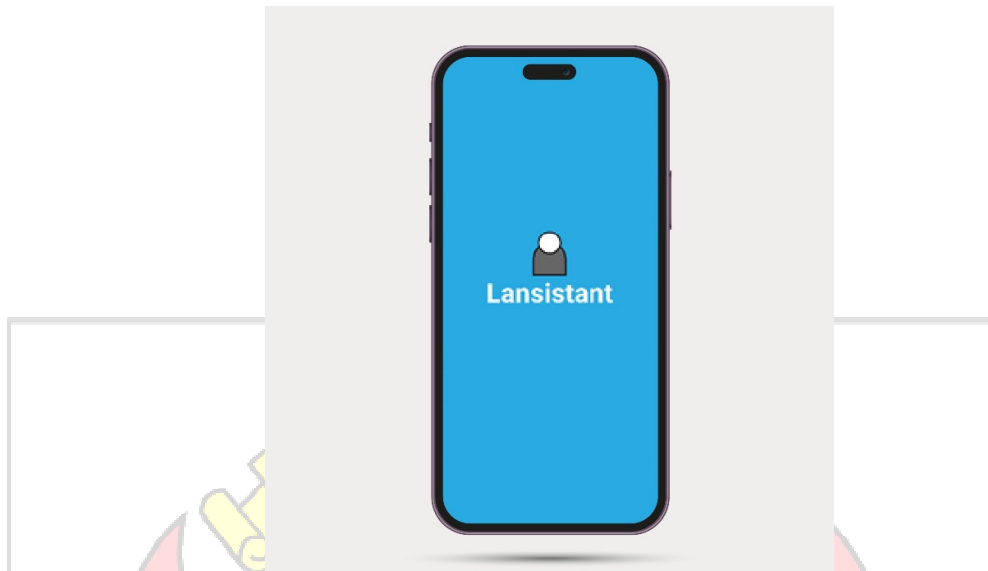


Gambar 3.3 Activity Diagram

Pada gambar 3.3 di atas terdapat *Activity diagram*. Diagram tersebut memperlihatkan relasi dari *user* dan sistem. *User* menggunakan aplikasi kemudian memasukkan informasi pribadi jika belum ada dan bisa langsung memulai interaksi suara dengan aplikasi.

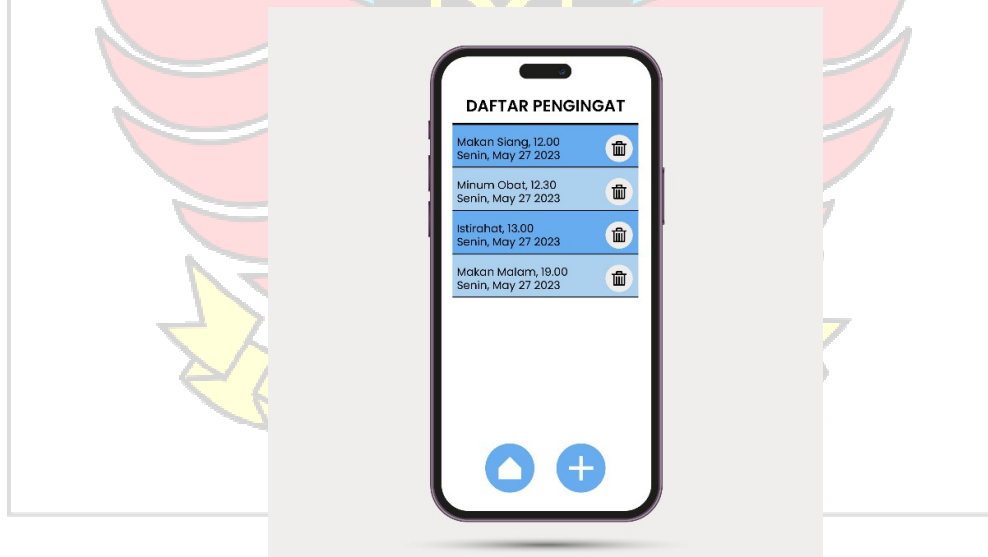
c. *Mockup* Aplikasi

Berikut adalah beberapa rancangan *mockup* tampilan aplikasi yang akan dibuat.



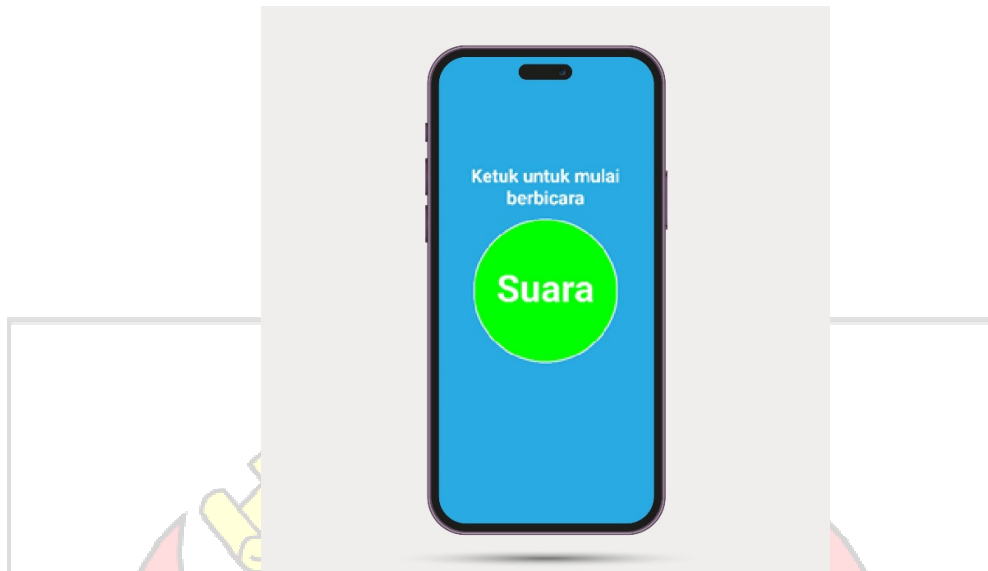
Gambar 3.4 *Mockup Tampilan Splash Screen*

Pada gambar 3.4 menampilkan logo dari aplikasi (*splash screen*) saat mulai menjalankan aplikasi.



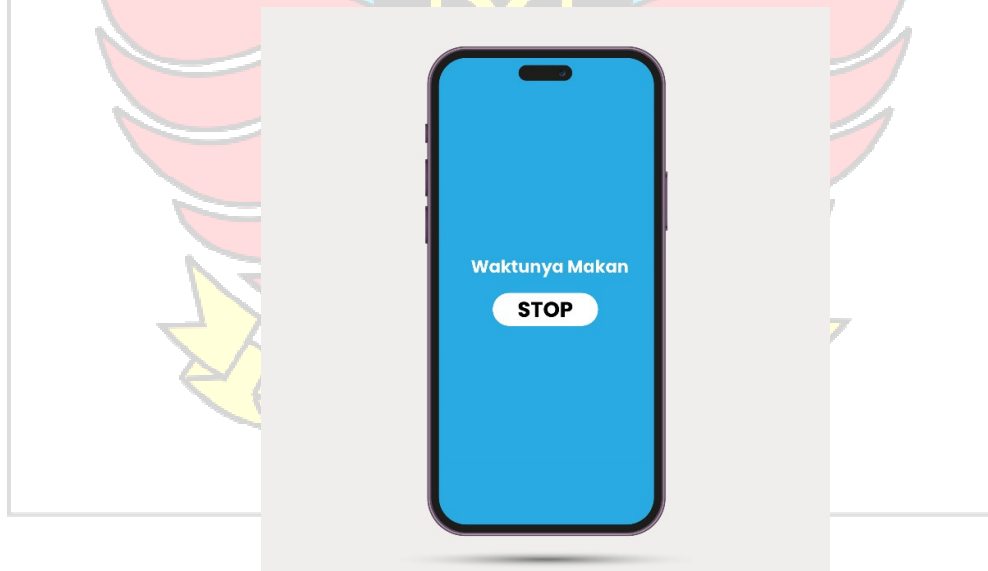
Gambar 3.5 *Mockup Tampilan Daftar Pengingat*

Pada gambar 3.5 menampilkan daftar Pengingat yang bisa dimasukkan sesuai kebutuhan lansia.



Gambar 3.6 *Mockup* Tampilan Interaktif Berbicara

Pada gambar 3.6 menampilkan lingkaran besar sebagai tombol untuk memulai berbicara dengan aplikasi.



Gambar 3.7 *Mockup* Tampilan Peningat Berbunyi

Pada gambar 3.7 menampilkan nama pengingat yang diingatkan juga tombol menon-aktifkan pengingat dari aplikasi sesuai yang telah dijadwalkan.

3.2.4 Implementasi

Ada pun pengimplementasian pada aplikasi ini diimplementasikan pada lansia. Sesuai dari usia kategori lansia, pengujian ini dilakukan pada orang yang berusia 60 tahun ke atas. Implementasi ini berupa penggunaan aplikasi sesuai fungsinya yang dilakukan oleh lansia.

3.2.5 Pengujian

Pengujian aplikasi dilakukan dengan metode *Black box*. *Black box testing* adalah pengujian yang dilakukan untuk menilai kebutuhan dan spesifikasi perangkat lunak. *Black box testing* cukup meninjau *input* dan *output* sistem perangkat lunak aplikasi tanpa pengetahuan tentang struktur pengkodean programnya. *Black box testing* juga dikenal sebagai *behavioral testing*. Metode ini merupakan pengujian terhadap fungsionalitas atau kegunaan sebuah aplikasi. Pengujian dengan metode *black box testing* ini dilakukan dengan memberikan serangkaian skenario pengujian yang akan dilakukan oleh para penguji, apakah sistem aplikasi yang sudah dibangun sudah sesuai dengan tujuan penelitian.

Pengujian metode *black box* ini akan dilakukan dengan *inputan* berupa ketukan pilihan dan suara dari manusia yang kemudian akan direspon oleh aplikasi dengan *output* berupa tampilan dan suara dari aplikasi.

BAB IV HASIL DAN PEMBAHASAN

Hasil dari penelitian ini adalah Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia. Aplikasi ini dibuat untuk membantu lansia dalam mengingat jadwal kesehariannya. Dengan menggunakan bahasa pemrograman python, aplikasi ini nantinya juga bisa melakukan diskusi dengan lansia.

4.1 Hasil Tampilan Antarmuka

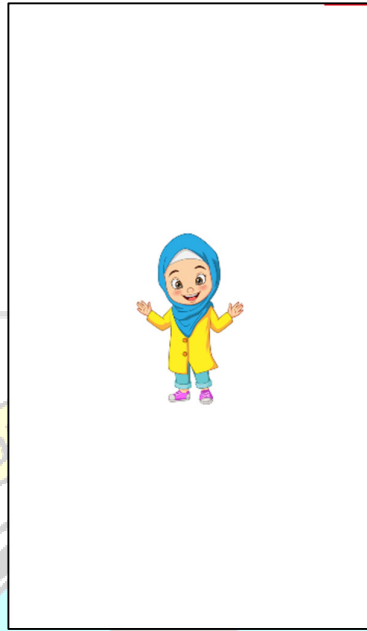
Tampilan antarmuka atau *user interface* merupakan desain tampilan yang dibuat agar memudahkan penggunaan aplikasi. Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia ini dikembangkan dengan menggunakan bahasa pemrograman Python dengan *framework* Kivy pada Visual Studio Code.

Berikut ini merupakan tampilan keseluruhan rancangan aplikasi.

4.2 Antarmuka Aplikasi Android

1. Tampilan awal (*Splash Screen*)

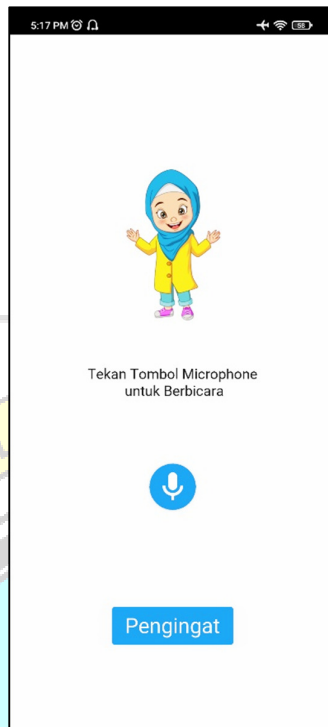
Splash screen merupakan tampilan yang pertama kali muncul saat aplikasi baru dibuka atau dijalankan.



Gambar 4.1 Tampilan *Splash Screen*

2. Tampilan utama

Tampilan utama ini merupakan tampilan yang akan muncul setelah *splash screen*. Pada tampilan ini terdapat satu gambar dengan dua tombol. Tombol yang pertama yaitu tombol mulai, berfungsi untuk menjalankan fungsi *speech recognition* pada aplikasi. Tombol yang kedua yaitu tombol pengingat, berfungsi untuk berpindah tampilan dan menampilkan daftar pengingat yang ada.

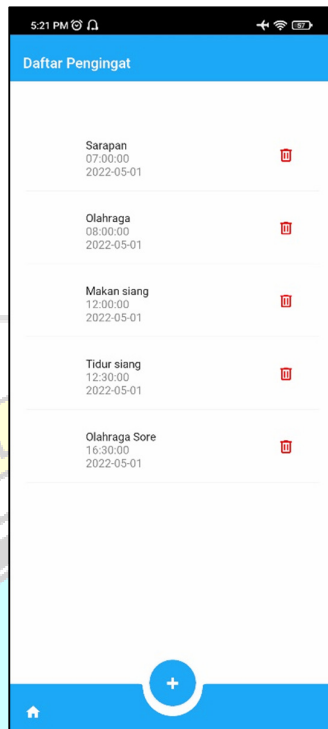


Gambar 4.2 Tampilan Utama

Untuk menjalankan fungsi *speech recognition* cukup dengan menekan tombol *microphone* maka aplikasi akan memulai untuk menerima masukan suara, kemudian membalasnya dengan data yang dimiliki.

3. Tampilan Daftar Peningat

Pada tampilan daftar peningat akan memunculkan daftar peningat berbentuk *list*. Terdapat dua tombol pada tampilan daftar peningat ini, yaitu tombol tambah peningat dan tombol kembali. Tombol kembali berfungsi untuk mengembalikan tampilan ke tampilan utama. Tombol tambah peningat berfungsi untuk berpindah ke halaman tampilan untuk menambahkan peningat. Di bawah ini merupakan tampilan peningat yang masih kosong.



Gambar 4.3 Tampilan Daftar Peningat

4. Tampilan Tambah Peningat

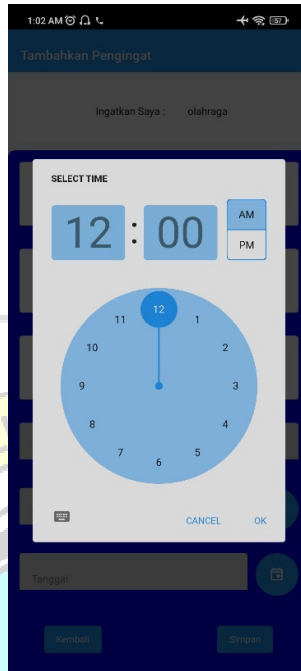
Pada tampilan tambahkan pengingat, digunakan untuk memasukkan data-data yang akan menjadi pengingat nantinya. Data ini dimasukkan ke *local database* menggunakan *sqlite3* database.



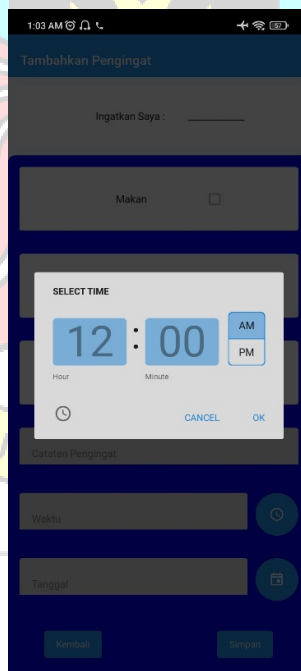
Gambar 4.4 Tampilan Tambah Peningat.

Pada tampilan ini akan diperlihatkan 3 kolom isian yang digunakan untuk memasukkan data, berupa judul, waktu dan tanggal. Untuk masukan ketiga data ini menggunakan tipe data string yang akan disimpan ke dalam *local database*. Untuk masukan waktu dan tanggal, menggunakan modul dari KivyMD yaitu MDTimePicker dan MDDatePicker.

Tampilan masukan waktu dan tanggal ini masing-masingnya memiliki dua mode, mode pertama yaitu memilih waktu dan tanggal dengan tampilan UI, sedangkan mode kedua yaitu memasukkan waktu dan tanggal dengan mengetikkannya. Tampilan masukan waktu dapat dilihat pada gambar 4.5 dan gambar 4.6.



Gambar 4.5 Tampilan Masukan Waktu Mode Geser.



Gambar 4.6 Tampilan Masukan Waktu Mode Ketik.

Tampilan masukan tanggal dapat dilihat pada gambar 4.7 dan gambar 4.8.



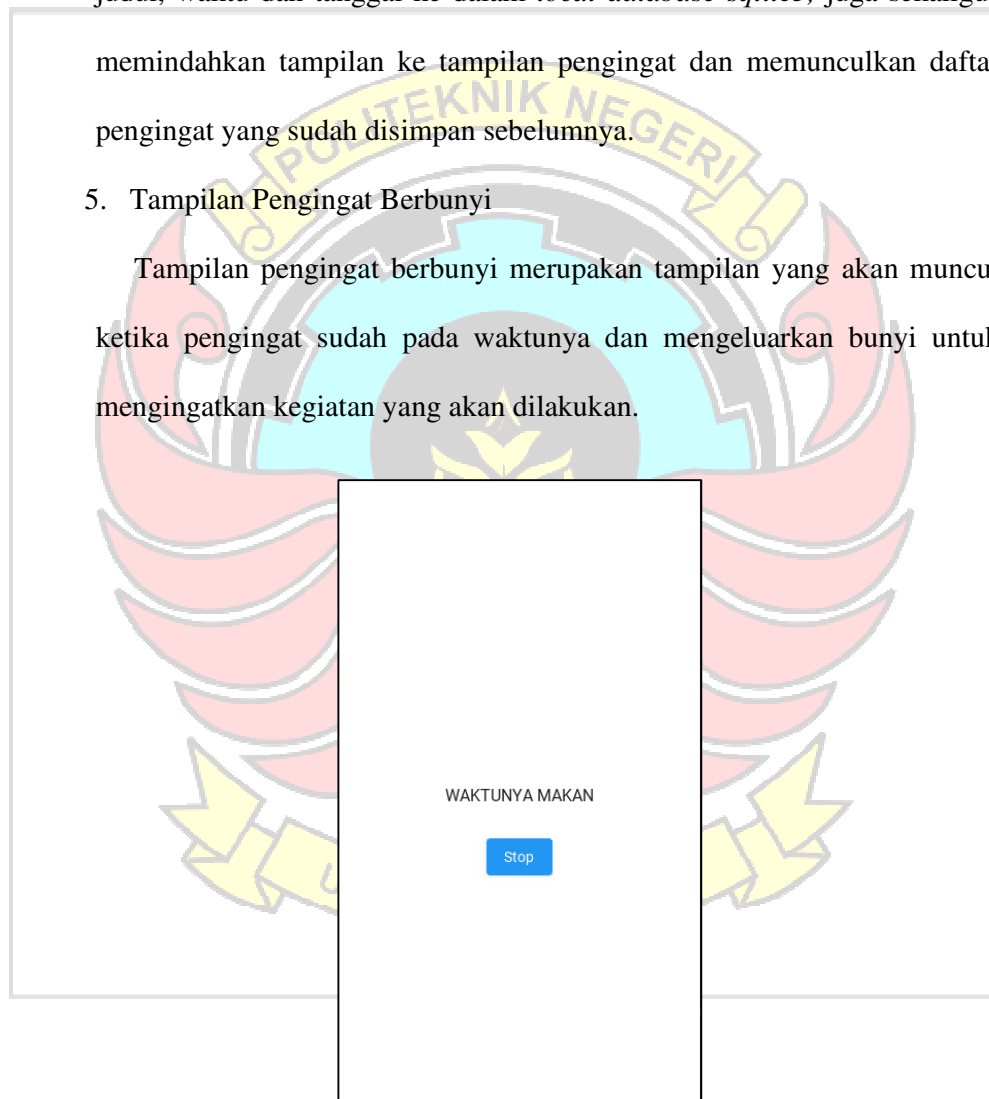
Gambar 4.7 Tampilan Masukan Tanggal Mode Sentuh

Gambar 4.8 Tampilan Masukan Tanggal Mode Ketik

Pada bagian bawah tampilan ini, terdapat dua tombol, yaitu tombol kembali dan tombol simpan. Tombol kembali digunakan untuk kembali pada tampilan sebelumnya, yaitu Tampilan Pengingat. Sedangkan, tombol simpan berfungsi untuk menyimpan data yang ada pada tiap kolom masukan judul, waktu dan tanggal ke dalam *local database sqlite3*, juga sekaligus memindahkan tampilan ke tampilan pengingat dan memunculkan daftar pengingat yang sudah disimpan sebelumnya.

5. Tampilan Pengingat Berbunyi

Tampilan pengingat berbunyi merupakan tampilan yang akan muncul ketika pengingat sudah pada waktunya dan mengeluarkan bunyi untuk mengingatkan kegiatan yang akan dilakukan.



Gambar 4.9 Tampilan Pengingat Berbunyi

4.3 Pengujian Sistem Aplikasi

Pengujian yang dilakukan terhadap aplikasi ini mengutamakan pada metode *Black Box Testing*. Pengujian jenis ini menitik beratkan pada fungsi sistem untuk melihat apakah hasil yang dicapai sesuai dengan yang diharapkan.

4.3.1 Pengujian Fungsionalitas Aplikasi Android

Pengujian ini adalah pengujian yang dilakukan terhadap aplikasi android dengan membuat beberapa kumpulan kondisi masukan untuk melihat respon aplikasi android apakah berjalan sesuai fungsi atau tidak sehingga dapat ditarik kesimpulan.

Tabel 4.1 Pengujian Terhadap Tampilan Awal / Utama

Skenario	Yang diharapkan	Pengamatan	Hasil
Menampilkan <i>Splash Screen</i>	<i>Splash screen</i> berhasil tampil	Aplikasi menampilkan <i>Splash screen</i> saat membuka aplikasi	Berhasil (Lampiran 1)
Menampilkan <i>dialog box</i> meminta perizinan <i>record audio</i>	Muncul <i>dialog box</i> meminta perizinan <i>record audio</i>	Aplikasi memunculkan <i>dialog box</i> meminta perizinan audio	Berhasil (Lampiran 2)
Menampilkan gambar kartun perempuan sebagai maskot, tombol <i>microphone</i> , dan tombol pengingat	Maskot kartun perempuan, tombol <i>microphone</i> , dan tombol pengingat berhasil tampil	Aplikasi menampilkan gambar kartun perempuan, tombol <i>microphone</i> dan tombol pengingat	Berhasil (Lampiran 3)
Menekan tombol <i>micrphone</i> untuk memberikan perintah suara	Tombol <i>microphone</i> merespon dan mengeluarkan suara <i>beep</i> untuk	Aplikasi berhasil mendengarkan dan merespon perintah suara	Berhasil (Lampiran 4)

	memasukkan perintah suara.		
Menekan tombol <i>microphone</i> dan memberikan perintah suara membuka daftar pengingat	Aplikasi merespon dan berganti tampilan menjadi tampilan daftar pengingat	Aplikasi berhasil merespon dengan suara dan berganti tampilan menjadi daftar pengingat	Berhasil (Lampiran 5)
Menekan tombol pengingat untuk berpindah tampilan dari tampilan utama ke tampilan daftar pengingat	Tampilan utama berganti menjadi tampilan daftar pengingat	Aplikasi berganti tampilan, dari tampilan utama menjadi tampilan daftar pengingat	Berhasil (Lampiran 4)

Berdasarkan Tabel 4.1 Pengujian Terhadap Tampilan Awal / Utama, menyajikan keberhasilan fungsionalitas aplikasi terhadap tampilan awal / utama. Skenario pertama yaitu aplikasi pertama kali dibuka, aplikasi berhasil menampilkan *splash screen*. Setelah aplikasi menampilkan *splash screen*, selanjutnya akan muncul *dialog box* yang berisi perizinan *record audio* sebagai fungsi dari *microphone* untuk menangkap *inputan* suara dari pengguna. Pada tampilan utama aplikasi, terdapat kartun perempuan sebagai maskot, tombol *microphone*, dan tombol pengingat. Tombol *microphone* berjalan sesuai fungsinya untuk memberikan *inputan* suara. Tombol pengingat berfungsi untuk berpindah tampilan dari tampilan utama ke tampilan daftar pengingat.

Tabel 4.2 Pengujian Terhadap Tampilan Daftar Pengingat

Skenario	Yang diharapkan	Pengamatan	Hasil
Menampilkan daftar pengingat yang sudah didaftarkan, tombol	Daftar pengingat ditampilkan dalam bentuk <i>list</i> dengan	Aplikasi berhasil	Berhasil (Lampiran 5)

<i>home</i> , tombol ikon tambah dan tombol ikon tempat sampah	masing-masing daftar memiliki ikon tempat sampah	menampilkan daftar pengingat, ikon hapus, tombol kembali dan tombol tambahkan pengingat	
Menghapus pengingat dengan menekan tombol ikon tempat sampah	Pengingat pada <i>list</i> terhapus	Aplikasi berhasil menghapus pengingat pada <i>list</i>	Berhasil (Lampiran 6)
Menekan tombol <i>home</i>	Tampilan berpindah dari tampilan daftar pengingat ke tampilan awal / utama	Aplikasi berhasil berpindah tampilan dari tampilan daftar pengingat ke tampilan awal / utama	Berhasil (Lampiran 3)
Menekan tombol ikon tambah untuk berpindah tampilan dari tampilan daftar pengingat ke tampilan tambahkan pengingat	Tampilan berpindah dari tampilan daftar pengingat ke tampilan tambahkan pengingat	Aplikasi berhasil berpindah tampilan dari tampilan daftar pengingat ke tampilan tambahkan pengingat	Berhasil (Lampiran 7)

Berdasarkan Tabel 4.2 Pengujian Terhadap Tampilan Daftar Pengingat, menyajikan keberhasilan fungsionalitas terhadap tampilan daftar pengingat.

Pada tampilan ini berhasil untuk menampilkan data daftar pengingat yang telah didaftarkan, tombol *home*, tombol ikon tempat sampah dan tombol ikon tambah. Tombol *home* berhasil berfungsi untuk kembali dari tampilan Daftar Pengingat ke tampilan utama aplikasi. Tombol ikon tempat sampah berhasil berfungsi untuk menghapus data pengingat yang telah didaftarkan. Tombol

ikon tambah berhasil berfungsi untuk berpindah tampilan dari tampilan Daftar Pengingat ke tampilan Tambahkan Pengingat.

Tabel 4.3 Pengujian Terhadap Tampilan Tambahkan Pengingat

Skenario	Yang diharapkan	Pengamatan	Hasil
Menampilkan kolom masukan untuk nama pengingat, waktu dan tanggal	Kolom nama pengingat, waktu dan tanggal kosong dapat ditampilkan	Aplikasi berhasil menampilkan form pengisian data pengingat	Berhasil (Lampiran 7)
Memasukkan judul / nama pengingat pada <i>text field</i> , mengatur waktu dan tanggal yang akan diingatkan serta menyimpannya dengan menekan tombol simpan	Data yang dimasukkan berhasil tersimpan di <i>database</i> dan ditampilkan pada tampilan daftar pengingat	Aplikasi berhasil menyimpan dan menampilkan pengingat yang telah ditambahkan	Berhasil (Lampiran 8)
Menekan tombol kembali untuk berpindah tampilan dari tampilan daftar tambahkan pengingat menjadi tampilan daftar pengingat	Tampilan berpindah dari tampilan tambahkan pengingat ke tampilan daftar pengingat	Aplikasi berhasil berpindah tampilan dari tampilan tambahkan pengingat ke tampilan daftar pengingat	Berhasil (Lampiran 9)
Tidak memasukkan nama pengingat pada kolom nama pengingat	Aplikasi tidak berpindah dan muncul <i>toast message</i>	Aplikasi tidak berpindah tampilan dan muncul <i>toast message</i> yang berisikan <i>Kolom nama pengingat kosong, harap diisi</i>	Berhasil (Lampiran 10)

Berdasarkan Tabel 4.3 Pengujian Terhadap Tampilan Tambahkan Pengingat, menyajikan keberhasilan fungsionalitas untuk tampilan Tambahkan Pengingat. Pada tampilan ini berhasil menampilkan *checkbox* makan, *checkbox* minum obat, *checkbox* olahraga, kolom keterangan, tombol jam, tombol kalender, tombol kembali dan juga tombol simpan. Pada tampilan ini, berhasil menjalankan fungsi untuk memilih salah satu *checkbox* item yang akan diingatkan, kemudian memasukkan keterangan terkait pengingat, memilih waktu pada tombol jam juga tombol kalender. Kemudian untuk tombol kembali berhasil berfungsi untuk berpindah tampilan dari tampilan Tambahkan Pengingat ke tampilan Daftar Pengingat. Tombol Simpan berhasil berfungsi untuk menyimpan pengingat yang didaftarkan ke database yang disediakan dan berpindah tampilan dari tampilan Tambahkan Pengingat ke tampilan utama.

Tabel 4.4 Pengujian Terhadap Tampilan Pengingat Berbunyi

Skenario	Yang diharapkan	Pengamatan	Hasil
Menampilkan item pengingat dan tombol <i>stop</i> untuk menghentikan bunyi pengingat	Menampilkan item pengingat dan menekan tombol <i>stop</i> . Bunyi pengingat mati dan tampilan kembali ke halaman utama	Aplikasi berhasil menampilkan item pengingat dan menghentikan bunyi pengingat	Berhasil (Lampiran 11)

Berdasarkan Tabel 4.4 Pengujian Terhadap Tampilan Pengingat Berbunyi, menyajikan keberhasilan fungsionalitas aplikasi pada tampilan Pengingat Berbunyi. Pada tampilan ini berhasil menampilkan item pengingat yang telah

ditentukan sebelumnya dan tombol stop. Tombol stop berhasil berfungsi untuk menghentikan bunyi pengingat yang sedang aktif dan mengganti tampilan ke tampilan utama.



BAB V PENUTUP

5.1 Kesimpulan

Dalam proses perancangan, implementasi dan pengujian dari penelitian Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia dapat ditarik kesimpulan bahwa Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia berhasil dibangun dan dapat berfungsi berdasarkan hasil pengujian fungsionalitas dengan menggunakan metode *black box testing*. Aplikasi dapat digunakan sebagai pengingat makan, pengingat minum obat dan pengingat olahraga. Aplikasi juga dapat melakukan komunikasi sederhana dengan masukan suara.

5.2 Saran

Aplikasi *Virtual Assistant* Berbasis Android untuk Lansia ini tentunya masih memiliki banyak kekurangan sehingga sangat bisa dikembangkan pada penelitian lainnya yang akan datang. Untuk penelitian selanjutnya dapat dilakukan penambahan fitur-fitur seperti mengirim pesan dari *smartphone* lansia ke *smartphone* kerabat lansia, melengkapi fitur *speech recognition* dengan menambahkan fungsi fungsi yang bisa dilakukan dengan *inputan* suara.

DAFTAR PUSTAKA

Adityawarman, Dicky dkk. 2010. Pengembangan Sistem Pakar Berbasis Web Untuk Pemilihan Program Studi Di Universitas Bina Nusantara. Skripsi. Teknik Informatika Universitas Bina Nusantara.

Beaver, Ian and Cynthia Freeman. 2016. Analysis of User Behavior with Multimodal Virtual Customer Service Agents. IEEE : Spoken Language Technology Workshop.

Bernard, D., & Arnold, A. 2019. Cognitive interaction with virtual assistants: From philosophical foundations to illustrative examples in aeronautics. Computers in Industry, 107, 33-49.

Bogdanchikov, A et al. 2013. Python to learn programming. In Journal of Physics: Conference Series (Vol. 423, No. 1, p. 012027). IOP Publishing.

Chan dkk. 2018. Aplikasi Caring Assistance Untuk Manula Berbasis Android Dan Raspberry Pi. Skripsi. Surabaya: Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra.

Constantin, Yoshua dkk. 2020. Aplikasi Personal Assistant Berbasis Voice Command Pada Sistem Operasi Android Dengan NLP. JOINTECS (Journal of Information Technology and Computer Science), 5(2), 121-128.

Gerung, Raymond Anthony dkk. 2019. Aplikasi Asisten Virtual Sebagai Perintah Suara Pada Sistem Otomatisasi Rumah Tinggal. ELECTRICES, 1(1), 9-14.

Hoy, M. B. 2018. Alexa, Siri, Cortana, and more: an introduction to voice assistants. Medical reference services quarterly, 37(1), 81-88.

McCarthy, John et al. 1955. What is Artificial Intelligence.. Proposal. Hanover : Dartmouth College.

Pranita, E. (2020, September 21). Ahli Minta Jangan Maklumi Lansia yang Pikun, Ini 10 Gejala Demensia (Halaman Web). Diakses dari <https://www.kompas.com/sains/read/2020/09/21/072800723/ahli-minta-jangan-maklumi-lansia-yang-pikun-ini-10-gejala-demensia?page=all>.

Rasjid, Fadjar Efendy. (2010, September 02). Android: Sistem Operasi Pada Smartphone (Halaman Web). Diakses dari https://ubaya.ac.id/2018/content/articles_detail/7/Android--Sistem-Operasi-pada-Smartphone.html

Sastriya dkk. 2019. Aplikasi Asisten Untuk Lansia Dengan Memanfaatkan Smartphone Berbasis Android. Badung: Teknik Elektro Universitas Udayana.

Schirmer, Heike. (2020, Juni 01). Alexa uses Amazon Translate to Reach More International Customers (Halaman Web). Diakses dari <https://aws.amazon.com/blogs/machine-learning/alex-a-uses-amazon-translate-to-reach-more-international-customers/>

Setiawan, M. M., & Hendriyawan A, M. S. 2019. Rancang Bangun Sistem Kendali Kompor Pintar Berbasis Aplikasi Google Assistant Menggunakan Metode

Voice Recognition (Doctoral dissertation, University of Technology Yogyakarta).

Soeriawidjaya, P. 2019. Implementasi Chatbot Aplikasi Chatting Berbasis Web Studi Kasus Pemesanan Jasa Fotografer (Doctoral dissertation, STMIK AKAKOM YOGYAKARTA).

Tarigan, M., & Ramadhan, R. 2020. Implementasi Fitur Virtual Assistance Pada Layanan Live Chat di PT. Garena Indonesia. *Budi Luhur Information Technology*, 17(2).

Tobing, Prihantoro Manahan & Magdalena Ariance Ineke Pakereng. Migrasi Aplikasi Stock Opname Platform Desktop Ke Android Menggunakan Kivy Framework. Skripsi. Salatiga : Program Studi Teknik Informatika Universitas Kristen Satya Wacana.

Tulshan, A. S., & Dhage, S. N. 2018. Survey on virtual assistant: Google assistant, siri, cortana, alexa. In *International symposium on signal processing and intelligent recognition systems* (pp. 190-201). Springer, Singapore.

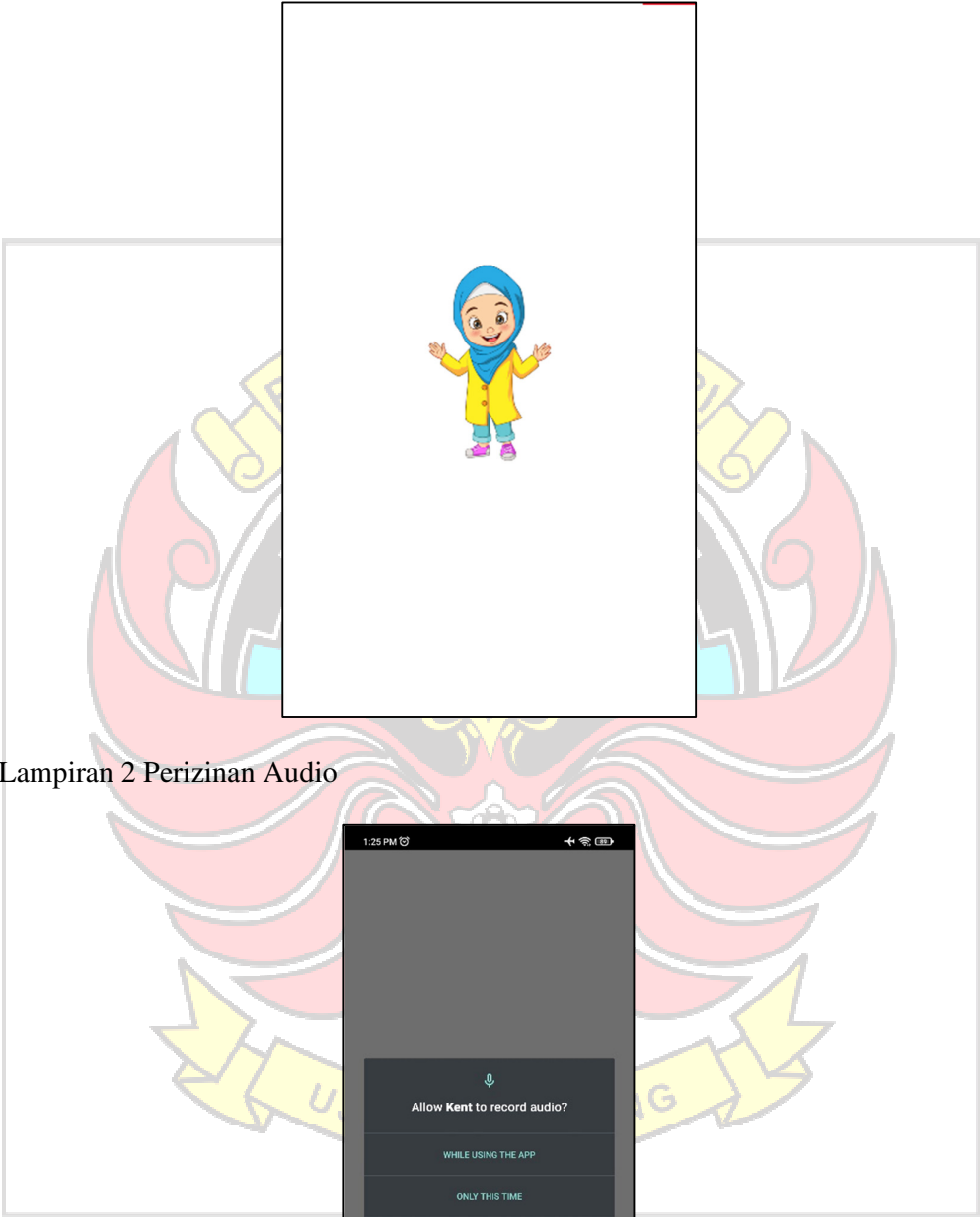
Wafa, R. S. 2020. Pembangunan Virtual Asisten Sebagai Media Informasi Ekowisata Dan Wali Pohon Di Kawasan Konservasi Taman Buru Masigit Kareumbi (Doctoral dissertation, Universitas Komputer Indonesia)..

Yuniar, E., & Purnomo, H. 2019. Implementasi Chatbot "Alitta" Asisten Virtual Dari Balittas Sebagai Pusat Informasi Di Balittas. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 13(1), 24-35.

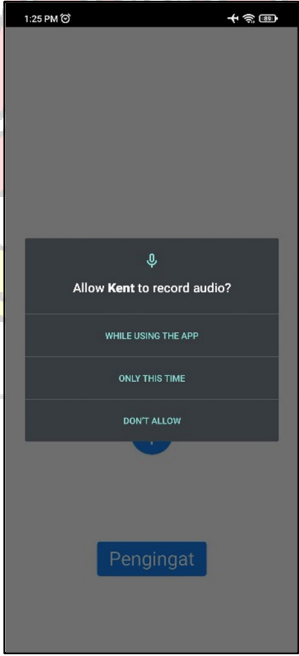


LAMPIRAN

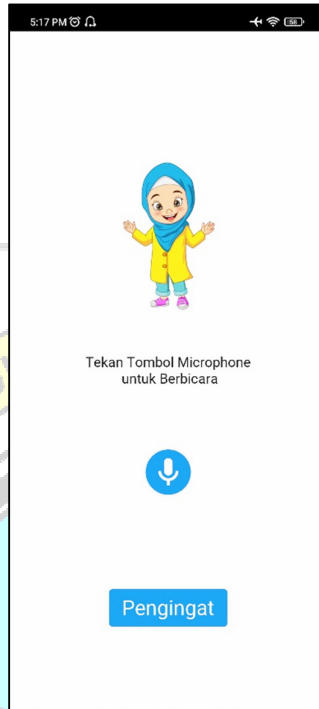
Lampiran 1 Tampilan *Splash Screen*



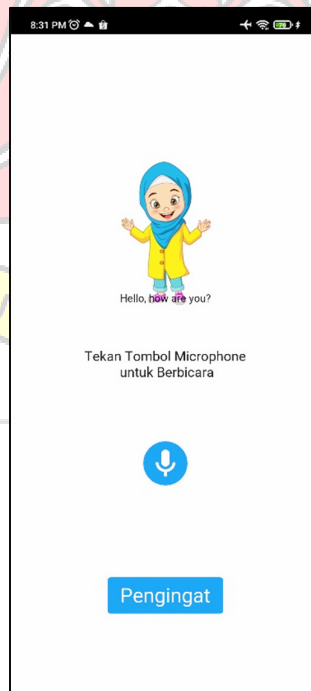
Lampiran 2 Perizinan Audio



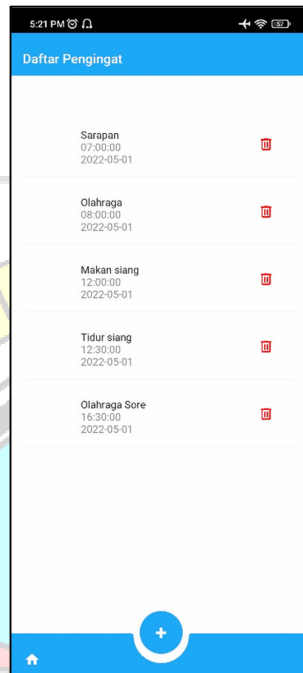
Lampiran 3 Tampilan Utama



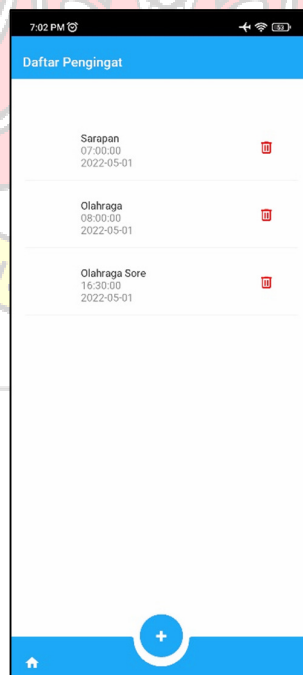
Lampiran 4 Memberikan perintah suara dan balasan dari aplikasi



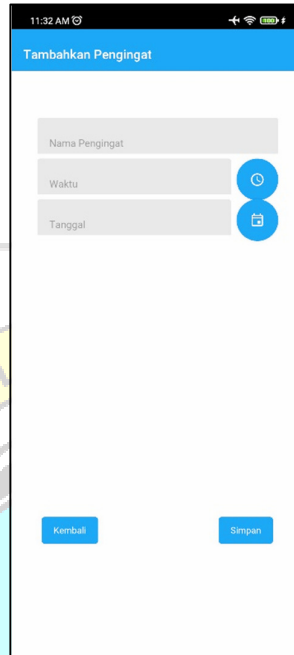
Lampiran 5 Tampilan Daftar Peningat



Lampiran 6 Menghapus salah satu item peningat



Lampiran 7 Tampilan Tambahkan Pengingat



11:32 AM

Tambahkan Pengingat

Nama Pengingat

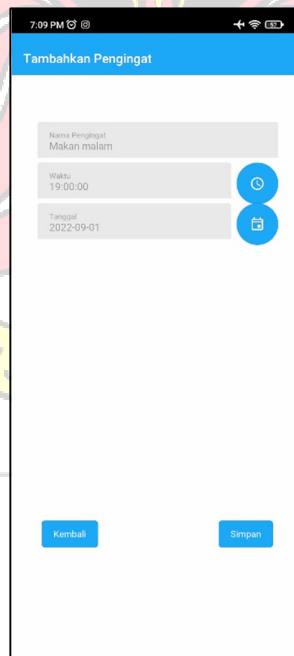
Waktu

Tanggal

Kembali Simpan

This screenshot shows the 'Tambahkan Pengingat' (Add Reminder) form. The form has three input fields: 'Nama Pengingat' (Reminder Name), 'Waktu' (Time), and 'Tanggal' (Date). The 'Waktu' and 'Tanggal' fields have circular icons to the right, likely for opening a time or date picker. At the bottom, there are two buttons: 'Kembali' (Back) and 'Simpan' (Save).

Lampiran 8 Menyimpan data pengingat



7:09 PM

Tambahkan Pengingat

Nama Pengingat
Makan malam

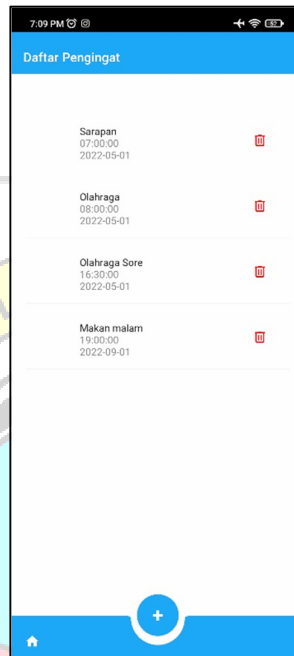
Waktu
19:00:00

Tanggal
2022-09-01

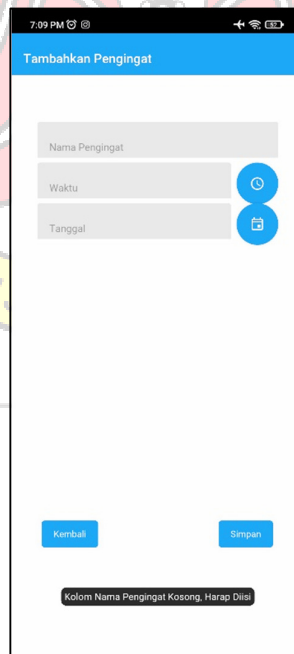
Kembali Simpan

This screenshot shows the 'Tambahkan Pengingat' form with the following data entered: 'Makan malam' in the 'Nama Pengingat' field, '19:00:00' in the 'Waktu' field, and '2022-09-01' in the 'Tanggal' field. The 'Kembali' and 'Simpan' buttons are visible at the bottom.

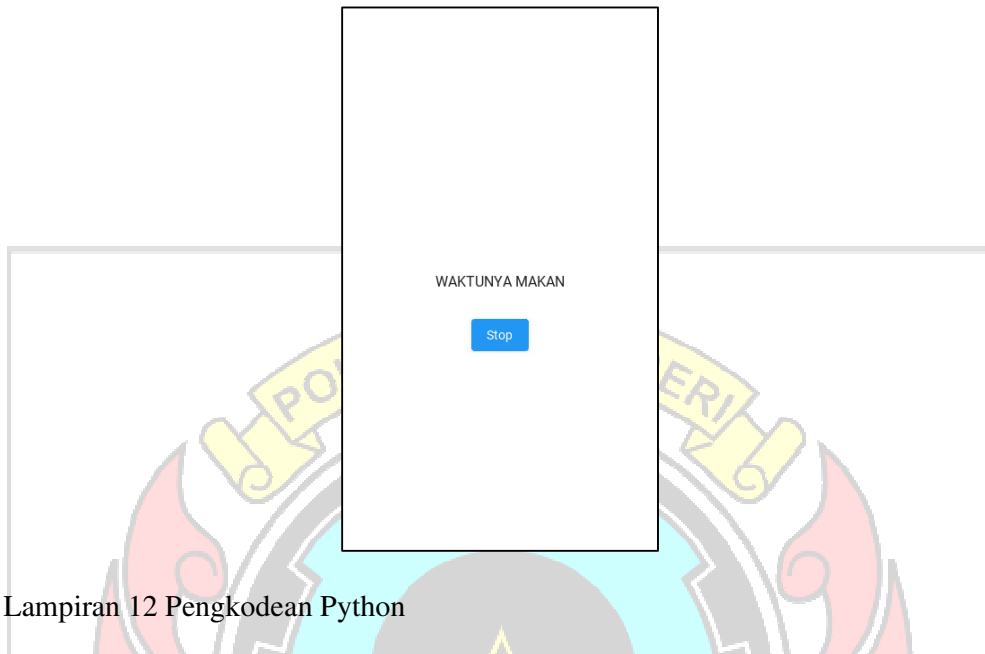
Lampiran 9 Kembali ke tampilan daftar pengingat dan menampilkan item pengingat yang ditambahkan



Lampiran 10 Tidak memasukkan nama pengingat pada kolom nama pengingat



Lampiran 11 Tampilan Peningat Berbunyi



Lampiran 12 Pengkodean Python

```
main.py 9+ x reminder.db
main.py
1 from kivy.app import MDApp
2 from kivy.lang import Builder
3 from kivy.uix.screen import MDScreen
4 from kivy.uix.screenmanager import MDScreenManager
5 from kivy.uix.pickers import MDTIMEPicker, MDDatePicker
6 from kivy.uix.list import ThreeLineAvatarIconListItem
7 from kivy.clock import Clock
8 from kivy.core.audio import SoundLoader
9 from kivy.uix.toast.kivytoast.kivytoast import toast
10 import sqlite3
11 import datetime
12 from plyer import stt
13 from plyer import tts
14
15 from android.permissions import request_permissions, Permission
16
17
18
19 class ListItemWithCheckBox(ThreeLineAvatarIconListItem):
20     """Check Box and Trash Icon function"""
21     def __init__(self, pk=None, **kwargs):
22         super().__init__(**kwargs)
23         # state a pk which we shall use link the list items with the database primary keys
24         self.pk = pk
25
26     def removeitem(self, taskid):
27         conn = sqlite3.connect('reminder.db')
28         c = conn.cursor()
29         c.execute("DELETE FROM reminder WHERE id=?", (taskid,))
30         conn.commit()
31         conn.close()
32
```

```

main.py
33 def delete_item(self, the_list_item):
34     '''Delete the task'''
35     self.parent.remove_widget(the_list_item)
36     self.removeitem(the_list_item.pk)# Here
37
38
39 class HomeScreen(MDScreen):
40     name = "homescreen"
41
42     def start_listening(self):
43         if stt.listening:
44             self.stop_listening()
45             return
46
47         stt.start()
48
49         Clock.schedule_interval(self.check_state, 1 / 5)
50
51     def stop_listening(self):
52         stt.stop()
53         self.update()
54
55         Clock.unschedule(self.check_state)
56
57     def check_state(self, dt):
58         if not stt.listening:
59             self.stop_listening()
60
61     def update(self):
62         query = stt.results
63         if "hello" in query:
64             tts.speak("Hello. How are you?")

```

```

61     def update(self):
62         query = stt.results
63         if "hello" in query:
64             tts.speak("Hello, How are you?")
65             self.ids.labelteks.text = str("Hello, how are you?")
66         elif "I am good" in query:
67             tts.speak("Great, Keep it Up!")
68             self.ids.labelteks.text = str("Great, keep it up!")
69         elif "open reminder" in query:
70             tts.speak("opening reminder")
71             self.manager.transition.direction = "left"
72             self.manager.current = "reminderscreen"
73         elif "add reminder" in query:
74             tts.speak("add reminder")
75             self.manager.transition.direction = "left"
76             self.manager.current = "addreminderscreen"
77
78
79 class ReminderScreen(MDScreen):
80     name = "reminderscreen"
81
82     def __init__(self, **kwargs):
83         super().__init__(**kwargs)
84         Clock.schedule_once(self.show_item)
85
86     def show_item(self, *args):
87         conn = sqlite3.connect('reminder.db')
88         c = conn.cursor()
89         records = c.execute("SELECT * FROM reminder").fetchall()
90         conn.commit()

```

```

main.py
85
86 def show_item(self, *args):
87     conn = sqlite3.connect('reminder.db')
88     c = conn.cursor()
89     records = c.execute("SELECT * FROM reminder").fetchall()
90     conn.commit()
91     conn.close()
92
93     if records != []:
94         for i in records:
95             reminder_items = ListItemWithCheckBox(
96                 pk=i[0], text = i[1],
97                 secondary_text = i[2],
98                 tertiary_text = i[3]
99             )
100             self.manager.get_screen("reminderscreen").ids.container.add_widget(reminder_items)
101
102 def home_screen(self, *args):
103     self.manager.transition.direction = "right"
104     self.manager.current = "homescreen"
105
106 def move_add_screen(self, *args):
107     self.manager.transition.direction = "left"
108     self.manager.current = "addreminderscreen"
109
110 def delete_all(self, *args, the_list_item):
111     deleteall = ListItemWithCheckBox()
112     deleteall.delete_all_item(the_list_item)
113
114
115 class AddReminderScreen(MDScreen):
116     name = "addreminderscreen"

```

```

main.py
114
115 class AddReminderScreen(MDScreen):
116     name = "addreminderscreen"
117
118     sound = SoundLoader.load('alarm.wav')
119     volume = 0
120
121     time_dialog = None
122
123     def __init__(self, *args, **kwargs):
124         super().__init__(*args, **kwargs)
125
126         conn = sqlite3.connect('reminder.db') # Hubungkan database
127         c = conn.cursor()
128         c.execute("CREATE TABLE if not exists reminder(id integer PRIMARY KEY AUTOINCREMENT, title TEXT, alarm TEXT, date TEXT)")
129         conn.commit()
130         conn.close()
131
132         self.time_dialog = MDTimePicker() # inialisasi
133         self.date_dialog = MDDatePicker() # inialisasi
134         Clock.schedule_interval(self.alarm, 1) # schedule the alarm function for every one second
135
136     def time_picker(self):
137         self.time_dialog.bind(on_save = self.schedule)
138         self.time_dialog.open()
139
140     def schedule(self, *args):
141         self.ids.alarm_timed.text = str(self.time_dialog.time)
142
143     def save_date(self, instance, value, date_range):
144         self.ids.date_time.text = str(value)
145

```

```

main.py
146     def date_picker(self):
147         self.date_dialog.bind(on_save = self.save_date)
148         self.date_dialog.open()
149
150     def save_reminder(self, *args):
151         self.ids.title_add.text = str(self.ids.title_add.text)
152         self.ids.alarm_timed.text = str(self.time_dialog.time)
153         self.ids.date_time.text = str(self.ids.date_time.text)
154
155         if self.ids.title_add.text == "":
156             toast("Kolom Nama Pengingat Kosong, Harap Diisi")
157         else:
158             conn = sqlite3.connect("reminder.db")
159             c = conn.cursor()
160             c.execute("INSERT INTO reminder (title, alarm, date) VALUES (:first, :second, :third)",
161                 {
162                     'first':self.ids.title_add.text,
163                     'second':self.ids.alarm_timed.text,
164                     'third':self.ids.date_time.text,
165                 })
166
167             conn.commit()
168             conn.close()
169
170             self.manager.get_screen('reminderscreen').ids['container'].add_widget(
171                 ListItemWithCheckBox(
172                     text = self.ids.title_add.text,
173                     secondary_text = self.ids.alarm_timed.text,
174                     tertiary_text = self.ids.date_time.text
175                 )
176             )

```

```

179
180 self.manager.transition.direction = "right" # set the direction of the transition before moving to next
181 self.manager.current = "reminderscreen" # move to the home screen
182
183 self.ids.title_add.text = "" # Reempty the add title text field
184 self.ids.alarm_timed.text = "" # Reempty the add time text field
185 self.ids.date_time.text = "" # Reempty the add time text field
186
187 def alarm(self, *args):
188     current_time = datetime.datetime.now().strftime("%H:%M:%S %Y-%m-%d")
189
190     conn = sqlite3.connect("reminder.db")
191     c = conn.cursor()
192     c.execute("SELECT alarm || ' ' || date AS today FROM reminder")
193     records = c.fetchall()
194     conn.commit()
195     conn.close()
196
197     alarms = [record[0] for record in records]
198
199     if str(current_time) in alarms:
200         self.start()
201
202 def start(self, *args):
203     self.manager.transition.direction = "left"
204     self.manager.current = "ringingscreen"
205     self.sound.play()
206     self.sound.loop = True

```

```

215 def stop_ring(self):
216     selfstop = AddReminderScreen()
217     selfstop.stop()
218     self.manager.transition.direction = "right"
219     self.manager.current = "homescreen"
220
221
222 class CustomScreenManager(MDScreenManager):
223     """Screen Manager"""
224
225
226 class GiantApp(MDApp):
227     def build(self):
228         screen = Builder.load_file('ui.kv')
229         return screen
230
231     def on_start(self):
232         request_permissions([Permission.RECORD_AUDIO])
233
234     def on_pause(self):
235         return True
236
237
238 if __name__ == "__main__":
239     GiantApp().run()
240

```

Lampiran 13 Pengkoedan Kivy

```
ui.kv
1 CustomScreenManager:
2     HomeScreen:
3     ReminderScreen:
4     AddReminderScreen:
5     RingingScreen:
6
7 <ListItemWithCheckbox>:
8     id: the_list_item
9     markup: True
10
11     IconRightWidget:
12         icon: 'trash-can-outline'
13         theme_text_color: "Custom"
14         text_color: 1, 0, 0, 1
15         on_release:
16             root.delete_item(the_list_item)
17
18 <HomeScreen>:
19     MDFloatLayout:
20         md_bg_color: 1, 1, 1, 1
21
22     MDLabel:
23         id: labelteks
24         text: ""
25         halign: "center"
26         pos_hint: {'center_y': .6}
27         color:
28
29     Image:
30         source: 'putri.png'
31         size_hint: .35, .35
32         pos_hint: {'center_x' : .5, "center_y" :.55}
```

```
34 MDLabel:
35     text: "Tekan Tombol Microphone\n untuk Berbicara"
36     font_size: "20sp"
37     halign: "center"
38     pos_hint: {'center_y': .45}
39
40 MDFloatingActionButton:
41     id: btn_speak
42     icon: "microphone"
43     icon_size: "50sp"
44     # size_hint: .10, .10
45     pos_hint: {'center_x': .5, 'center_y': .35}
46     on_release: root.start_listening()
47
48 MDRaisedButton:
49     id: btn_remindersc
50     text: "Pengingat"
51     font_size: "30sp"
52     pos_hint: {'center_x': .5, 'center_y': .15}
53     on_release:
54         root.manager.transition.direction = "left"
55         root.manager.current = "reminderscreen"
56
57 <ReminderScreen>:
58     MDFloatLayout:
59         md_bg_color: 1, 1, 1, 1
60
61     MDTopAppBar:
62         title: "Daftar Pengingat"
63         pos_hint : {"top" :1}
64
```

```

ScrollView:
  pos_hint: {'center_y': .5, 'center_x': .5}
  size_hint: .9, .78

  MDList:
    id : container

~
MDBoxLayout:
  MDBottomAppBar:
    MDTopAppBar:
      icon: "+"plus-thick"
      type: "bottom"
      left_action_items: [{"home", lambda x: root.home_screen()}]
      on_action_button: root.move_add_screen(self.icon)
      # right_action_items: [{"trash-can"}]

<RingingScreen>:
  MDFloatLayout:
    md_bg_color: 1, 1, 1, 1

    MDLabel:
      id : label_ring
      # text: root.alarmsaja
      halign : "center"
      pos_hint: {'center_y': .5}

    MDRaisedButton:
      id : stop_button
      text : "Stop"
      pos_hint: {'center_x': .5, 'center_y': .4}
      on_release: root.stop_ring()

```

```

97 <AddReminderScreen>:
98   MDBoxLayout:
99     md_bg_color : 1, 1, 1, 1
100
101   MDTopAppBar:
102     title: "Tambahkan Pengingat"
103     pos_hint: {"top": 1}
104
105   MDBoxLayout:
106     orientation: 'vertical'
107     padding: "35dp"
108     spacing: 150
109     md_bg_color : 1, 0, 1, 0
110     pos_hint: {'center_y': 0.4}
111
112     MDBoxLayout:
113       spacing: 70
114       MDTextField:
115         id: title_add
116         hint_text: "Nama Pengingat"
117         mode: "fill"
118         fill_color: 0, 0, 0, .4
119         pos_hint: {'top': 1.0}
120
121     MDBoxLayout:
122       orientation: "horizontal"
123       spacing : 20
124
125     MDTextField:
126       id: alarm_timed
127       hint_text: "Waktu"

```

```

134         MDFloatingActionButton:
135             icon: 🕒"clock-outline"
136             pos_hint: {'top': 2.0}
137             on_release: root.time_picker()
138
139         MDBoxLayout:
140             orientation: "horizontal"
141             spacing : 20
142
143             MDTextField:
144                 id: date_time
145                 hint_text: "Tanggal"
146                 mode: "fill"
147                 fill_color: 0, 0, 0, .4
148                 pos_hint: {'top': 3.0}
149
150             MDFloatingActionButton:
151                 icon: 📅"calendar-outline"
152                 pos_hint: {'top': 3.0}
153                 on_release: root.date_picker()
154
155     MDFloatLayout:
156
157         MDRaisedButton:
158             id: btn_remindersc2
159             text: "Kembali"
160             pos_hint: {'center_x': 0.2,'center_y': 0.2}
161             on_release:
162                 root.manager.transition.direction = "right"
163                 root.manager.current = "reminderscreen"
164
165         MDRaisedButton:
166             id: btn_save
167             text: "Simpan"
168             pos_hint: {'center_x': 0.8,'center_y': 0.2}
169             on_release: root.save_reminder()

```

Lampiran 14 Konfigurasi Bulldozer untuk *Deploy* ke Android

```

1  [app]
2
3  # (str) Title of your application
4  title = Lyntt
5
6  # (str) Package name
7  package.name = lyntt
8
9  # (str) Package domain (needed for android/ios packaging)
10 package.domain = org.test
11
12 # (str) Source code where the main.py live
13 source.dir = .
14
15 # (list) Source files to include (let empty to include all the files)
16 source.include_exts = py,png,jpg,kv,atlas,wav
17

```

```

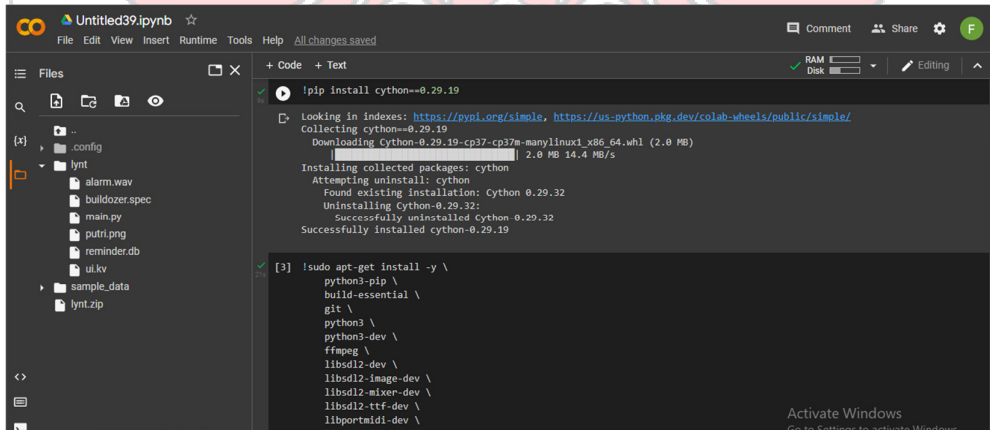
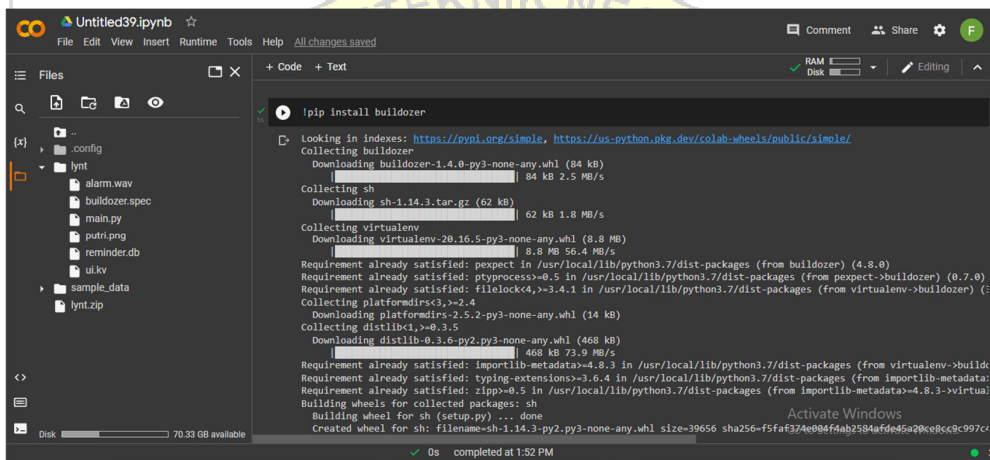
31 # (str) Application versioning (method 1)
32 version = 0.1
33
34 # (str) Application versioning (method 2)
35 # version.regex = __version__ = ['](.*')[']
36 # version.filename = %(source.dir)s/main.py
37
38 # (list) Application requirements
39 # comma separated e.g. requirements = sqlite3,kivy
40 requirements = python3,kivy==2.1.0,pillow,plyer,sqlite3,kivymd
41
42 # (str) Custom source folders for requirements
43 # Sets custom source for any requirements with recipes
44 # requirements.source.kivy = ../../kivy
45
46 # (str) Presplash of the application
47 presplash.filename = %(source.dir)s/putri|.png
48
49 # (str) Icon of the application
50 #icon.filename = %(source.dir)s/data/icon.png
51
52 # (str) Supported orientation (one of landscape, sensorLandscape,
53 orientation = portrait
54
55
56
57
58
59
60
61
62
63
64
65 # change the major version of python used by the app
66 osx.python_version = 3
67
68 # Kivy version to use
69 osx.kivy_version = 1.9.1
70
71 #
72 # Android specific
73 #
74
75 # (bool) Indicate if the application should be fullscreen or not
76 fullscreen = 0
77

```



```
94
95 # (list) Permissions
96 android.permissions = INTERNET, RECORD_AUDIO
97
98 # (list) features (adds uses-feature -tags to manifest)
99 #android.features = android.hardware.usb.host
100
101 # (int) Target Android API, should be as high as possible.
102 #android.api = 27
```

Lampiran 15 *Build and Deploy* kivy file menjadi .apk di Google Colab



```
Untitled39.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
RAM 100%
Disk 100%
Editing

!sudo apt-get install -y \
libgstreamer1.0 \
gstreamer1.0-plugins-base \
gstreamer1.0-plugins-good

Preparing to unpack .../43-libmail-sendmail-perl_0.80-1_all.deb ...
Unpacking libmail-sendmail-perl (0.80-1) ...
Selecting previously unselected package libgstreamer1.0-dbg:amd64.
Preparing to unpack .../44-libgstreamer1.0-dbg_1.14.5-0ubuntu1-18.04.2_amd64.deb ...
Unpacking libgstreamer1.0-dbg:amd64 (1.14.5-0ubuntu1-18.04.2) ...
Setting up libpam-cap:amd64 (1:2.25-1.2) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be used. at /usr/share/perl5/DebianConfig/Backend.pm line 102.)
debconf: falling back to frontend: Readline
Setting up libarchive-zip-perl (1:0-1ubuntu0.1) ...
Setting up libtime-date-perl (2.3000-2) ...
Setting up libsigsegv2:amd64 (2.12-1) ...
Setting up libgpm2:amd64 (1.20.7-5) ...
Setting up libcap2-bin (1:2.25-1.2) ...
Setting up libltdlconvert0:amd64 (1:24-2-1) ...
Setting up libarchive-cpio-perl (0.10-1) ...
Setting up gettext-base (0.19.8.1-6ubuntu0.3) ...
Setting up m4 (1.4.18-1) ...
Setting up libccparanoia0:amd64 (3.10.2-2debian-13) ...
Setting up libmagic-mgc (1:5.32-2ubuntu0.4) ...
Setting up libmagic1:amd64 (1:5.32-2ubuntu0.4) ...
Setting up libsys-hostname-long-perl (1.5-1) ...
Setting up libmail-sendmail-perl (0.80-1) ...
```

```
Untitled39.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
RAM 100%
Disk 100%
Editing

!l-dev openssl libgdbm-dev libgdbm-compat-dev liblzma-dev libreadline-dev libncursesw5-dev libffi-dev uuid-dev libffi6

Reading package lists... Done
Building dependency tree
Reading state information... Done
libffi-essential is already the newest version (12.4ubuntu1).
libffi6 is already the newest version (3.2.1-8).
libreadline-dev is already the newest version (7.0-3).
libreadline-dev set to manually installed.
bzlib2 is already the newest version (1.0.6-8.1ubuntu0.2).
libbz2-dev is already the newest version (1.0.6-8.1ubuntu0.2).
libbz2-dev set to manually installed.
liblzma-dev is already the newest version (5.2.2-1.3ubuntu0.1).
liblzma-dev set to manually installed.
libsqlite3-dev is already the newest version (3.22.0-1ubuntu0.6).
libsqlite3-dev set to manually installed.
libssl-dev is already the newest version (1.1.1-1ubuntu2.1~18.04.20).
libssl-dev set to manually installed.
libssl-dev is already the newest version (2.31.1-0.4ubuntu3.7).
libssl-dev set to manually installed.
zlib1g-dev is already the newest version (1:1.2.11.dfsg-0ubuntu2.2).
The following package was automatically installed and is no longer required:
  libnvidia-common-460
Use 'sudo apt autoremove' to remove it.
Suggested packages:
  ncurses-doc sqlite3-doc
The following NEW packages will be installed:
  libffi-dev libgdbm-compat-dev libgdbm-dev libncursesw5-dev sqlite3
```

```
Untitled39.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
RAM 100%
Disk 100%
Editing

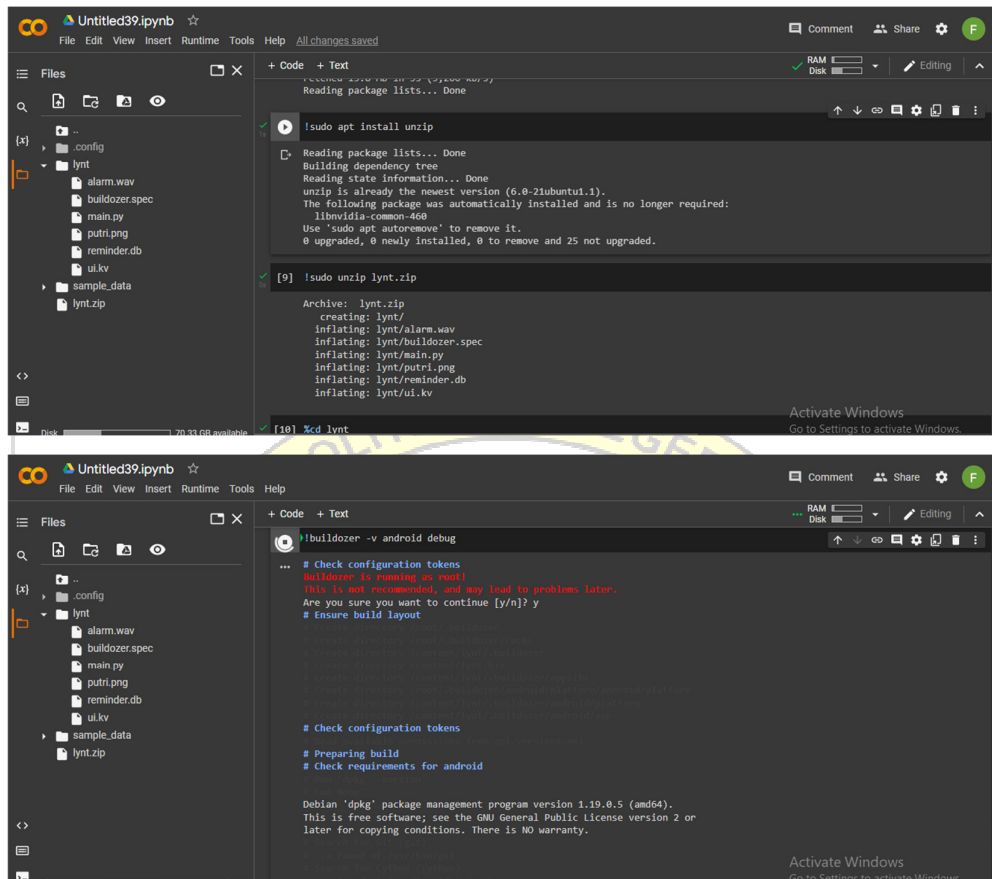
Setting up openssl (1.1.1-1ubuntu2.1-18.04.20) ...
[5] Setting up libgdbm-compat-dev (1.14.1-6) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...

!sudo apt-get install libffi-dev

Reading package lists... Done
Building dependency tree
Reading state information... Done
libffi-dev is already the newest version (3.2.1-8).
The following package was automatically installed and is no longer required:
  libnvidia-common-460
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 19 not upgraded.

[7] !apt-get update

Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:2 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran40/ InRelease [3,626 B]
Hit:3 http://archive.ubuntu.com/ubuntu bionic InRelease
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:5 https://ppa.launchpad.net/c2d4u.team/c2d4u4/ubuntu bionic InRelease [15.9 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [83.3 kB]
Ign:7 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x86_64 InRelease
Hit:8 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64 InRelease
```



Lampiran 16 Dokumentasi Uji Coba Aplikasi Vistual Assistan untuk Lansia.





