



UNIVERSITAS
Dinamika

***TALENT MAPPING IDENTIFIKASI MINAT BAKAT SESEORANG
MENGUNAKAN NATURAL LANGUAGE PROCESSING***

TUGAS AKHIR



Program Studi

SI TEKNIK KOMPUTER

UNIVERSITAS
Dinamika

Oleh :

Ilham Fahmi Amri Rosada

19410200028

FAKULTAS TEKNOLOGI DAN INFORMATIKA

UNIVERSITAS DINAMIKA

2023

***TALENT MAPPING IDENTIFIKASI MINAT BAKAT SESEORANG
MENGUNAKAN NATURAL LANGUAGE PROCESSING***

TUGAS AKHIR

**Diajukan sebagai salah satu syarat untuk menyelesaikan
Program Sarjana Teknik Komputer**



Oleh :
Nama : Ilham Fahmi Amri Rosada
NIM : 19410200028
Program Studi : S1 Teknik Komputer

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
UNIVERSITAS DINAMIKA**

2023

TUGAS AKHIR

TALENT MAPPING IDENTIFIKASI MINAT BAKAT SESEORANG MENGUNAKAN NATURAL LANGUAGE PROCESSING

Dipersiapkan dan disusun oleh

Ilham Fahmi Amri Rosada

NIM : 19410200028

Telah diperiksa, dibahas dan disetujui oleh Dewan Pembahas

Pada: 12 Januari 2023

Susunan Dewan Pembahas

Pembimbing :

I. Heri Pratikno, M.T., MTCNA., MTCRE.

NIDN : 0716117302

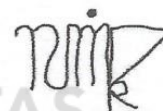
II. Weny Indah Kusumawati, S.Kom., M.MT.

NIDN : 0721047201

Pembahas :

Harianto, S.Kom., M.Eng.

NIDN : 0722087701



Digitally signed
by Heri Pratikno
Date: 2023.01.13
16:11:35 +07'00'



Universitas
Dinamika
2023.01.12
19:08:23 +07'00'



cn=Harianto Harianto,
o=Universitas Dinamika,
ou=Prodi S1 Teknik Komputer,
email=hari@dinamika.ac.id,
c=ID
2023.01.13 16:54:56 +07'00'

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana



Digitally signed by
Universitas Dinamika
Date: 2023.01.18
07:20:40 +07'00'

Tri Sagirani, S.Kom., M.MT.

NIDN: 0731017601

Dekan Fakultas Teknologi dan Informatika

UNIVERSITAS DINAMIKA



“Don't compare yourself with others, because the main character in your life is yourself.”

~ Ilham Fahmi AR ~

UNIVERSITAS
Dinamika



UNIVERSITAS

“Dipersembahkan kepada Bapak, Ibu, Keluarga saya atas dukungan, motivasi dan doa terbaik yang diberikan kepada saya. Beserta semua orang yang selalu membantu, mendukung, memberi masukan, dan memberi motivasi agar tetap berusaha dan belajar agar menjadi lebih baik.”

PERNYATAAN
PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa **Universitas Dinamika**, Saya :

Nama : **Ilham Fahmi Amri Rosada**
NIM : **19410200028**
Program Studi : **S1 Teknik Komputer**
Fakultas : **Teknologi dan Informatika**
Jenis Karya : **Laporan Tugas Akhir**
Judul Karya : **TALENT MAPPING IDENTIFIKASI MINAT
BAKAT SESEORANG MENGGUNAKAN NATURAL
LANGUAGE PROCESSING**

Menyatakan dengan sesungguhnya bahwa :

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, Saya menyetujui memberikan kepada **Universitas Dinamika** Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalty Free Right*) atas seluruh isi/sebagian karya ilmiah Saya tersebut diatas untuk disimpan, dialihmediakan, dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama Saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta.
2. Karya tersebut diatas adalah hasil karya asli Saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya, atau pendapat orang lain yang ada dalam karya ilmiah ini semata-mata hanya sebagai rujukan yang dicantumkan dalam Daftar Pustaka Saya.
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiasi pada karya ilmiah ini, maka Saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada Saya.

Demikian surat pernyataan ini Saya buat dengat sebenar-benarnya.

Surabaya, 10 Desember 2022

A 10,000 Indonesian Rupiah postage stamp is shown, featuring the Garuda Pancasila emblem and the text '10000', 'METERAI TEMPEL', and 'FF357AKX066361398'. A black ink signature is written over the stamp.

Ilham Fahmi Amri Rosada
NIM : 19410200028

ABSTRAK

Bakat adalah sebuah kemampuan yang dimiliki tiap diri seseorang yang masih belum disadari dan diketahui sejak awal. Akan tetapi jika seseorang bisa menemukan hal yang disukai dan diminati, maka mempermudah menemukan bakat minat yang dimilikinya. Di Perguruan Tinggi banyak mahasiswa dinyatakan salah memilih jurusan terutama dibidang IT, dan dampak terbesarnya menjadi pemicu peningkatan angka pengangguran. Penelitian Sebelumnya yang dilakukan oleh Wiryanto, Meneliti sebuah *talent* data perusahaan karyawan dengan memprediksi analisa menggunakan *Machine Learning* dengan hasil keluaran peta pembagian “*Star*” dan “*Talent Risk*”. Pada penelitian ini mengidentifikasi minat bakat seseorang dengan pengujian sebuah pertanyaan berdasarkan kesukaan, *personality* dan produktifitas seseorang menggunakan acuan tabel kuadran. Mengolah data *input* melalui *speech recognition*, kemudian di *tokenization* untuk memisah sebuah kalimat menjadi pecahan per kata. Lalu reduksi kembali dari pecahan kata menggunakan proses *Lemmatization* untuk mengolah reduksi bentuk kata awalan dan akhiran menjadi kata dasar, dari hasil tersebut di olah dan dilakukan pencocokan pada *dataset*. Sehingga memberikan persentase kecocokan pada *referensi job skill IT*. Pengujian menunjukkan bahwa hasil *Pre-trained model network* yang digunakan dapat berjalan dengan baik. Menghasilkan nilai akurasi paling tinggi hingga 73.06%, pada *epochs* 1000. Pengujian evaluasi *talent mapping* tiap usia, dan mendapatkan nilai tertinggi di usia dewasa dengan *persentase* rata-rata 42.67%.

Kata kunci: *Minat Bakat, Deep Learning, Pre-trained model network, Natural Language Processing, Android*

KATA PENGANTAR

Dengan mengucapkan puji dan syukur terhadap kehadiran Tuhan Yang Maha Esa yang telah memberikan berkah dan rahmat-Nya, serta segala karunia yang telah diberikan kepada penulis, sehingga penulis dapat menyelesaikan Laporan Tugas Akhir yang berjudul “*TALENT MAPPING IDENTIFIKASI MINAT BAKAT SESEORANG MENGGUNAKAN NATURAL LANGUAGE PROCESSING*”. Laporan Tugas Akhir ini disusun dalam rangka memenuhi salah satu prasyarat dalam menyelesaikan Program Sarjana Teknik Komputer di Universitas Dinamika.

Pada kesempatan yang telah diberikan ini, penulis mengucapkan rasa terima kasih terhadap individu-individu yang memberikan dukungan dan juga saran serta bimbingan dalam upaya untuk menyelesaikan laporan Tugas Akhir ini. Oleh karena itu penulis ingin mengucapkan terima kasih kepada:

1. Orang Tua yang telah memberikan kontribusi besar berupa dukungan penuh atas apa yang akan penulis lakukan dan juga atas apa yang telah penulis lakukan, sehingga dapat menyelesaikan Laporan Tugas Akhir ini.
2. Ibu Tri Sagirani, S.Kom., M.MT., selaku Dekan Fakultas Teknologi dan Informatika (FTI) Universitas Dinamika.
3. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi S1 Teknik Komputer Universitas Dinamika.
4. Bapak Heri Pratikno, M.T., MTCNA., MTCRE., selaku Dosen Pembimbing I yang selalu memberikan waktu dan bimbingan serta ilmu dalam menyelesaikan Tugas Akhir beserta laporan ini.
5. Ibu Weny Indah Kusumawati, S.Kom., M.MT., selaku Dosen Pembimbing II yang juga selalu memberi waktu dan bimbingan dalam menyelesaikan Tugas Akhir beserta laporan ini.
6. Bapak Harianto, S.Kom., M.Eng., selaku Dosen Pembahas yang juga memberikan saran dan kritik dalam menyelesaikan Tugas Akhir beserta laporan ini.
7. Seluruh Dosen pengajar Program Studi S1 Teknik Komputer Universitas Dinamika yang telah memberikan ilmu, dan juga bimbingan yang berharga dari semester 1 hingga sampai saat ini.

8. Aviva Salsabilah Harun yang telah memberikan dukungan serta bantuan dalam segala bentuk yang akhirnya penulis bisa menyelesaikan Laporan Tugas Akhir.
9. Seluruh rekan-rekan S1 Teknik Komputer angkatan 2019 yang telah memberikan dukungan dan semangatnya untuk membantu penulis untuk menyelesaikan Laporan Tugas Akhir.

Penulis memahami bahwa Laporan Tugas Akhir ini belum mencapai kata sempurna, dan masih banyak kekurangan dalam menyusun laporan ini. Oleh karena itu dalam kesempatan ini, penulis meminta maaf apabila dalam Laporan Tugas Akhir ini masih terdapat kesalahan, baik dalam penulisan maupun bahasa yang digunakan. Penulis juga menerima kritik dan saran dari para pembaca yang sifatnya membangun untuk kesempurnaan laporan yang telah penulis susun.

Surabaya, 12 Januari 2023



UNIVERSITAS
Dinamika Penulis

DAFTAR ISI

	Halaman
ABSTRAK	vi
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xiv
DAFTAR LAMPIRAN.....	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Manfaat	3
BAB II LANDASAN TEORI	4
2.1 Bakat Minat.....	4
2.2 <i>Talent Mapping</i>	4
2.3 <i>Speech Recognition</i>	4
2.4 <i>Android Studio</i>	5
2.5 <i>Deep Learning</i>	6
2.6 <i>Natural Language Processing</i>	6
2.7 <i>Flask RESTful API</i>	7
2.8 <i>TensorFlow</i>	8
2.9 <i>Keras</i>	8
BAB III METODOLOGI PENELITIAN.....	9
3.1 Klasifikasi Kalimat Menggunakan Metode <i>Natural Language Processing 3 Layer Dengan Library Nltk, Keras dan TensorFlow</i> ...	9
3.2 <i>Setting Virtual Environment</i>	10
3.3 <i>Dataset</i>	10
3.4 Metode <i>Talent Mapping</i>	11
3.5 <i>Persentase Output</i>	13
3.6 <i>Semantik Role Labeling (SRL)</i>	14

3.7 Training Server Process	14
3.8 Testing Server Process.....	15
3.9 Testing Client Process	15
3.10 Training Program Natural Language Processing.....	16
3.11 Bag of Word NLP Program	17
3.12 Pembuatan RESTful API	17
3.13 Pembuatan Desain Mockup Aplikasi	18
3. 14 Pembuatan Aplikasi di Android Studio.....	19
3.14.1 Konversi Data Suara Menggunakan Google API Speech Recognition	19
3.14.2 Perancangan Layanan REST API.....	20
3.14.3 Perancangan Request dan Response Data Talent RESTfull API	21
3.14.4 Sistem Pengambilan Data Client	22
3.14. 5 Function Slider Preview Tutorial	23
3.14.6 Function Main Menu	24
3.14.7 Function Layanan API Dari Retrofit2.....	25
3.14.8 Function API Service	26
BAB IV HASIL DAN PEMBAHASAN	27
4.1 Pengujian Pre-trained Model Network	27
4.1.1 Tujuan Pengujian Pre-trained Model Network.....	27
4.1.2 Prosedur Pengujian Pre-trained Model Network.....	27
4.1.3 Hasil Pengujian Pre-trained Model Network.....	27
4.2 Pengujian Akurasi Pengambilan Data Suara	28
4.2.1 Tujuan Pengujian Akurasi Pengambilan Data Suara.....	28
4.2.2 Prosedur Pengujian Akurasi Pengambilan Data Suara	28
4.2.3 Hasil Pengujian Akurasi Pengambilan Data Suara.....	29
4.3 Hasil Perolehan Uji Coba Evaluasi Talent Mapping	30
4.3.1 Tujuan Memperoleh Uji Coba Evaluasi Hasil Talent Mapping	30
4.3.2 Prosedur Memperoleh Uji Coba Evaluasi Talent Mapping....	30
4.3.3 Hasil Nilai Pengujian Evaluasi Talent Mapping.....	31

BAB V PENUTUP.....	34
5.1 Kesimpulan	34
5.2 Saran	34
DAFTAR PUSTAKA	36
LAMPIRAN.....	38
BIODATA PENULIS	62



UNIVERSITAS
Dinamika

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Alur diagram blok penerimaan sinyal suara.....	5
Gambar 2.2 <i>Android Studio</i>	6
Gambar 3.1 Klasifikasi kalimat menggunakan metode NLP 3 <i>layer</i>	9
Gambar 3.2 Alur proses sistem aplikasi identifikasi minat bakat seseorang	9
Gambar 3.3 <i>Dataset IT Skill</i>	10
Gambar 3.4 Kuadran untuk klasifikasi.....	11
Gambar 3.5 <i>Flowchart Training Server Process</i>	14
Gambar 3.6 <i>Flowchart Testing Server Process</i>	15
Gambar 3.7 <i>Flowchart Testing Client Process</i>	16
Gambar 3.8 Program data <i>training Natural Language Processing</i>	16
Gambar 3.9 <i>Bag of Word NLP Program</i>	17
Gambar 3.10 Pembuatan <i>RESTful API</i>	18
Gambar 3.11 Pembuatan Desain <i>Mockup</i> Aplikasi.....	18
Gambar 3.12 Diagram Komunikasi <i>Client</i> dan Layanan <i>Google API</i>	19
Gambar 3.13 Diagram <i>endpoints</i> pada layanan <i>REST API</i>	20
Gambar 3.14 Diagram Komunikasi <i>Client</i> dan Layanan <i>server</i>	21
Gambar 3.15 <i>Flowchart</i> sistem pengambilan data <i>client</i>	22
Gambar 3.16 <i>Function Slider Preview Tutorial</i>	23
Gambar 3.17 <i>Function Main Menu</i>	24
Gambar 3.18 <i>Function Layanan API</i> Dari <i>Retrofit2</i>	25
Gambar 3.19 <i>Function API Service</i>	26

DAFTAR TABEL

	Halaman
Tabel 4.1 Hasil Pengujian <i>Pre-trained Network</i>	27
Tabel 4. 2 Hasil Uji Anak-anak.....	29
Tabel 4.3 Hasil uji Remaja.....	29
Tabel 4.4 Hasil uji Dewasa	30
Tabel 4.5 Hasil Pengujian <i>Talent Mapping</i> Anak-anak.....	31
Tabel 4.6 Hasil Pengujian <i>Talent Mapping</i> Remaja	32
Tabel 4.7 Hasil Pengujian <i>Talent Mapping</i> Dewasa.....	33



UNIVERSITAS
Dinamika

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. <i>Source Code Program Natural Language Processing</i>	38
Lampiran 2. <i>Source Code Program Communicate Flask REST API Server</i>	44
Lampiran 3. <i>Source Code Program Android Studio Retrofit2 API Client</i>	46
Lampiran 4. <i>Source Code Program Alent Apps System</i>	47
Lampiran 5. Hasil Keluaran Aplikasi.....	59
Lampiran 6. Hasil Plagiasi	60



UNIVERSITAS
Dinamika

BAB I

PENDAHULUAN

1.1 Latar Belakang

Bakat adalah kemampuan yang dimiliki setiap orang dan pada dasarnya tidak diketahui dan disadari sejak awal. Namun, jika seseorang dapat menemukan objek yang menarik dan di minati, maka lebih mempermudah untuk memunculkan bakat yang dimilikinya. Menurut Shaleh Abdul Rahman di bukunya “psikologi perspektif islam”, minat adalah suatu kecenderungan pada kepribadian yang memberikan perhatian dan tindakan terhadap orang, aktivitas maupun situasi menjadikan suatu objek minat disertai perasaan atau kesenangan (Anggraini, 2020).

Perguruan tinggi merupakan jenjang tertinggi yang mempersiapkan peserta didik dengan kemampuan akademik dan organisasi yang dapat menerapkan ilmu, mengembangkan pengetahuan, teknologi dan seni. Berbagai jurusan atau program studi ditawarkan oleh perguruan tinggi salah satunya di Indonesia. Banyak mahasiswa yang merasa mereka salah memilih jurusan setelah melewati beberapa semester kuliah, terutama dibidang *Information and technology*. Dikutip oleh pakar *Educational Psychologist* Irene Guntur, M.Psi, Psi, CGA, dari *Integrity Development Flexibility (IDF)* (Assauri, 2019) menyebutkan bahwa sebanyak 87% mahasiswa Indonesia yang masih salah memilih jurusan di Perguruan Tinggi. Inti dari masalah utama tersebut berdampak negatif bagi mahasiswa seperti kesulitan memahami materi, kurang motivasi belajar, rasa rendah diri, kurang percaya diri hingga kerusakan psikologis. dan pada akhirnya mereka bekerja semaksimal mungkin tanpa passion, serta buruknya lagi menjadi pemicu peningkatan pengangguran (Adji & Silvia, 2018).

Penelitian sebelumnya (Adji & Silvia, 2018), Meneliti *Talent Mapping*: menuju jenjang pendidikan yang lebih tinggi dengan menyebarkan kuisioner pada siswa SMA Muhammadiyah 4 dengan gaya pembelajaran *Student Active Learning (SAL)*. Menurut penelitian yang dilakukan (Wiryanto, 2017), Membahas *Reinventing Talent Management: Predictive Analytics with Machine Learning* dengan data simulasi dari perusahaan yang telah menggunakan *nine box*. Dengan

memasukkan unsur prediksi algoritma “Rendah”, “Menengah” dan “Tinggi” dimana proses prediksi ini menggunakan validasi *K-Folding* hingga 10 iterasi dan hasil prediksi dengan tingkat akurasi 93% tetapi yang diperoleh hanyalah peta siapa yang masuk kelompok “*STAR*” dan “*TALENT RISK*”.

Berdasarkan permasalahan sebagaimana tersebut diatas, maka pada Tugas Akhir ini membuat aplikasi identifikasi dan klasifikasi minat bakat seseorang berdasarkan kriteria usia anak-anak, remaja, dan dewasa, untuk membantu seseorang dalam menentukan minat bakatnya melalui jawaban atas 10 pertanyaan yang diberikan menggunakan *Deep Learning* dengan metode *Natural Language Processing (NLP)*. Adapun pengolahan data *inputan* melalui *Speech Recognition* yang nantinya menghasilkan aplikasi “*Talent Recommended*” dan “*Talent Not Recommended*”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat dirumuskan masalah pada Tugas Akhir ini sebagai berikut:

1. Bagaimana cara membedakan suara menurut umur dan jenis kelamin menggunakan *speech recognition* menjadi sebuah teks?
2. Bagaimana melakukan klasifikasi dan evaluasi sebuah data dari *user* menggunakan *NLP*?

1.3 Batasan Masalah

Dalam pembuatan Tugas Akhir ini, pembahasan masalah dibatasi pada beberapa hal berikut:

1. Kosakata dalam *NLP* masih terbatas hanya pengambilan Hobi yang terdaftar di *dataset*.
2. *Dataset* yang digunakan hanya berfokus pada *job talent IT*.
3. *Microphone* tidak ada *noise* dan *device* yang sama.
4. Tidak bisa menggunakan suara yang berdialek atau logat daerah.

1.4 Tujuan

Berdasarkan latar belakang dan rumusan masalah diatas, mendapatkan tujuan pada Tugas Akhir ini sebagai berikut:

1. Mengetahui cara membedakan suara menurut umur dan jenis kelamin menggunakan *speech recognition* menjadi sebuah teks.
2. Mampu melakukan klasifikasi dan evaluasi sebuah data dari *user* menggunakan *NLP*.

1.5 Manfaat

Adapun dari Tugas Akhir ini dapat diperoleh manfaat sebagai berikut:

1. Dapat menambahkan pengetahuan dan penerapan tentang sistem pencarian bakat seseorang menggunakan *Deep Learning*.
2. Referensi untuk mahasiswa jika melakukan penelitian tentang *NLP* dan *talent mapping*.
3. Mengurangi para remaja muda yang kesalahan memilih jurusan kuliah serta meningkatkan SDM dalam perusahaan.



UNIVERSITAS
Dinamika

BAB II

LANDASAN TEORI

2.1 Bakat Minat

Bakat merupakan anugerah yang diberikan kepada seseorang, namun juga menjadi tantangan bagi individu untuk mengenali dan mengembangkan bakat yang dimilikinya, dan pemicu untuk mengetahui bakat tersebut adalah minat. Hobi adalah aktivitas yang dicintai dan dihargai secara alami yang membantu mengembangkan bakat seseorang, tetapi bakat itu sendiri memiliki sebuah arti makna kemampuan bawaan serta potensi yang belum berkembang dan belum terlatih. Bakat seseorang dapat dipengaruhi oleh 2 faktor, yaitu: Faktor internal (minat, motivasi, keberanian atau resiko, ketekunan dalam menghadapi tantangan dan kegigihan dalam mengatasi kesulitan yang muncul). Sedangkan faktor eksternal (kesempatan maksimal untuk pengembangan diri, sarana dan prasarana, dukungan dan dorongan orang tua dan keluarga, lingkungan tempat tinggal). Pada Buku ajar Erni Murniarti yang mengutip dari Asrori dalam jurnalnya (Dr.Dra Erni Murniarti, 2020).

2.2 Talent Mapping

Talents Mapping menurut Abah Rama Royani, merupakan cara assessment atau menilai dan menggali bakat (karateristik produktif) dan potensi kekuatan kita dengan tampilan hasil yang detail, jelas, mudah dipahami dan menarik. Cara ini bisa mengidentifikasi potensi kekuatan individu yang mencakup pengukuran dan pernyataan kekuatan diri (Prio Kustanto, 2021). Talent Mapping juga dapat membantu menemukan bakat terpendam, dan mengarahkan orang agar tidak fokus pada kelemahannya saja (Karyaone, 2020).

2.3 Speech Recognition

Speech Recognition atau pengenalan suara merupakan sarana yang dominan untuk komunikasi antara manusia dan mesin, proses mengubah sinyal akustik menjadi satu set kata-kata. Aplikasi termasuk perintah dan kontrol suara, entri data, suara antarmuka *user*, mengotomatisasi pekerjaan operator telepon di telepon serta

berfungsi sebagai *input* untuk pemrosesan bahasa alami. Ada dua varian pengenalan suara berdasarkan durasi sinyal suara: Pertama yaitu pengenalan kata terisolasi, di mana setiap kata dikelilingi oleh semacam jeda. Kedua yaitu pengenalan kata berhubungan, dimana pengenalan ucapan terus menerus, di mana kata-kata dikelola satu serupa lain dan harus di kelompokkan. Pengenalan ucapan memiliki tugas yang sulit sebab dari banyak sumber variabilitas yang terkait dengan sinyal seperti akustik *realisasi fonem*, unit suara terkecil yang terdiri dari kata-kata, sangat tergantung pada konteksnya. Variabilitas dalam diri pembicara dapat diakibatkan oleh perubahan fisik pembicara dan keadaan emosi, kecepatan berbicara, atau kualitas suara. Linguistik, dialek, dan ukuran serta bentuk saluran vokal dapat berkontribusi pada variabilitas lintas pembicara (Ahmed, 2012).

Implementasi sistem mencakup *pre-processing* dan fitur umum modul ekstraksi, pemodelan dan klasifikasi ucapan independen pembicara. Dimulai dari penangkapan sinyal audio, ekstraksi fitur terdiri dari langkah-langkah berikut seperti yang ditunjukkan pada diagram blok di seperti contoh:



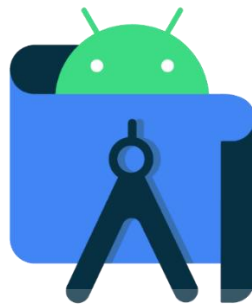
Gambar 2.1 Alur diagram blok penerimaan sinyal suara
(Sumber: Ahmed, 2012)

2.4 Android Studio

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) yang berfokus membangun aplikasi pada *platform android*. Memiliki basis pada IntelliJ IDEA. Bahasa pemrograman utama yang digunakan yaitu Java dan Kotlin. *Android Studio* juga terhubung dengan *Android Software Development Kit* (SDK) dan *Native Development Kit* (NDK) untuk *deploy* atau *render* ke perangkat *android*.

Berikut spesifikasi syarat minimum komputer untuk instalasi *android studio*:

No.	Keterangan
1.	RAM atau memori minimal 3 GB. Direkomendasikan RAM 8 GB.
2.	Minimal 2 GB penyimpanan <i>hardisk</i> yang tersedia. Direkomendasikan 4 GB (500 MB untuk penyimpanan IDE dan 1,5 GB untuk penyimpanan <i>Android</i> SDK dan sistem emulator).
3.	Resolusi layar minimum 1280 x 800.



Gambar 2.2 *Android Studio*
(Sumber: Android, 2021)

2.5 *Deep Learning*

Deep learning adalah salah satu bidang cabang dari *Machine Learning* yang memanfaatkan jaringan syaraf tiruan yang digunakan dalam berbagai aplikasi dalam kehidupan. *Deep Learning* mudah dipahami sebagai metode untuk meningkatkan hasil dan mengoptimalkan proses waktu dalam komputasi. Dengan banyaknya lapisan, maka model pembelajaran lebih baik dan hasil keluaran yang lebih presisi (Vargas, Mosavi, & Ruiz, 2017).

2.6 *Natural Language Processing*

Natural language processing atau bisa disebut pengolahan bahasa alami metode pemrosesan *input* teks di mana kata kunci dipakai, *library* yang dipakai sendiri yaitu *nltk tools*, dan *nlp_id* untuk pengolahan bahasa. Pengolahan bahasa alami merupakan cabang dari ilmu komputer dan *linguistik* yang membahas tentang memahami konsep tentang makna interaksi antara manusia dengan komputer menggunakan bahasa alami (bahasa manusia). Agar komputer dapat memahami bahasa alami, maka harus memiliki pengetahuan tentang bahasa alami itu sendiri

baik tentang kata yang digunakan seperti apa arti dari kata, fungsi kata-kata dalam sebuah kalimat dan bagaimana dari kata-kata dapat membentuk kalimat. Bahasa alami yang dimaksudkan yaitu pesan yang ingin dikomunikasikan oleh manusia baik secara lisan maupun tulisan (Muhammad, 2020).

Dalam melakukan pengolahan bahasa alami, sistem membutuhkan beberapa komponen untuk menganalisis makna kalimat:

1. *Sistem Parser* mengambil kata kerja yang dimasukkan oleh *user* dan mengelompokkan secara tata bahasa.
2. *Sistem Representasi* yaitu Pengetahuan tentang sistem analisis kelompok kata untuk menentukan definisi.
3. *Output Translator* Setelah melakukan *scan*, sistem memberikan umpan balik kepada *user*. *Output* dapat berupa bahasa alami atau keluaran lainnya.
4. *Pattern Matching* adalah sebuah metode pencocokan pola kata yang digunakan dalam kecerdasan buatan. Dalam ilmu komputer, *pattern matching* digunakan untuk memeriksa urutan *tokenize* yang ada di dalam suatu kalimat. Salah satu kegunaan *pattern matching* yaitu untuk mencari komponen pola yang cocok, kemudian menggantinya dengan urutan *tokenize* lainnya. Contoh *pattern matching* adalah *sequence pattern*. Dalam *sequence pattern*, pencarian pola terjadi dengan menggunakan *reguler expression (regex)* dan algoritma *backtracking*. Selain *reguler expression*, pencarian pola kalimat dapat dilakukan dengan *tree patterns*. *Tree patterns* di gunakan dalam beberapa bahasa pemrograman sebagai alat umum untuk memproses data berdasarkan strukturnya, misalnya, *Haskell*, *ML*, *Scala* dan *Mathematical*. Bahasa pemrograman tersebut memiliki sintaks khusus untuk mengekspresikan *tree patterns* dan konstruksi bahasa untuk eksekusi bersyarat dan pengambilan nilai berdasarkan itu. Dalam hal kesederhanaan dan efisiensi, penggunaan *reguler expression (regex)* masih lebih bagus dari pada *tree patterns*.

2.7 Flask RESTful API

Flask-RESTful adalah sebuah *extension Flask* yang mendukung untuk membangun *REST API* secara cepat. *REST* sendiri adalah sebuah arsitektur desain pengembang *web* yang membagi secara logis sumber daya *API* mudah diakses,

dimanipulasi dan dikembangkan, sedangkan *Flask* sendiri merupakan *microweb framework* dari bahasa *python* dan tidak *membutuhkan tools* maupun *library* (Hudya, 2019).

2.8 TensorFlow

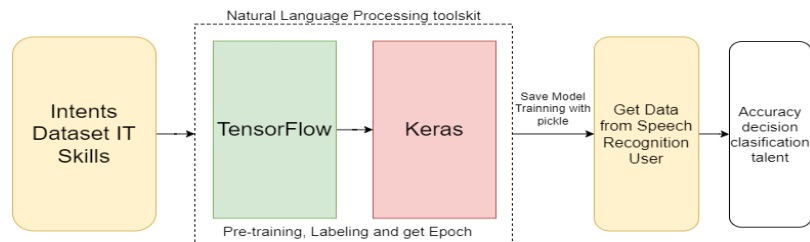
TensorFlow merupakan *framework* sebuah *library open source python* yang populer serta banyak digunakan untuk pengembangan dan penerapan *Machine Learning* serta algoritma lain yang memiliki metode tentang operasi matematika. *TensorFlow* bisa penggabungan banyak model dan algoritma *Machine Learning* termasuk *deep learning*. Ada 2 jenis model *TensorFlow* yang dapat melatih dan menjalankan. Yang pertama adalah *Neural Network* yang guna untuk keperluan mengklasifikasi tulisan tangan, pengenalan gambar atau objek. Sedangkan yang kedua adalah *Re-Current Neural Network*, yang merupakan model *Sequential* digunakan di *NLP (Natural Language Processing)* dan *PDE (Partial Differential Equation)* (Tensorflow.org, 2022).

2.9 Keras

Keras sebuah *interface library* yang mempunyai tujuan menyederhanakan algoritma *deep learning*. Fungsi *keras* sendiri mengurangi jumlah tindakan yang diperlukan dalam mengimplementasi kode umum juga dapat menjelaskan kesalahan *user*. *Keras* sendiri bisa berjalan diatas *TensorFlow* dan dikolaborasikan membuat pelatihan data lebih maksimal (Keras.io, 2022).

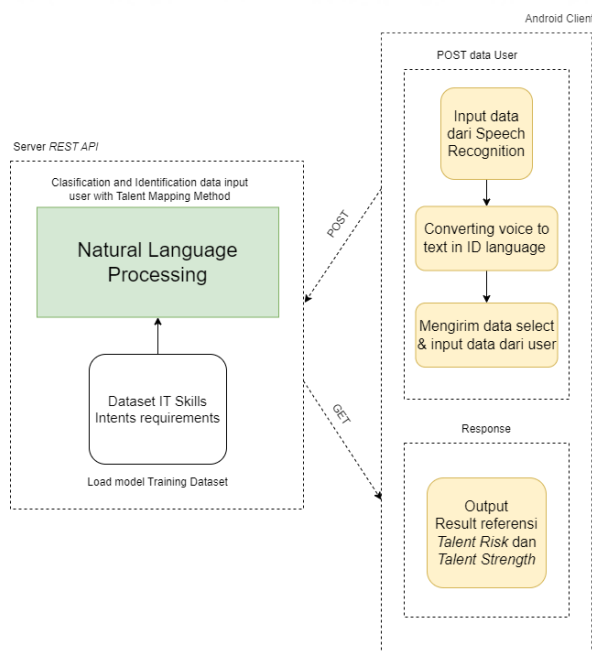
BAB III METODOLOGI PENELITIAN

3.1 Klasifikasi Kalimat Menggunakan Metode *Natural Language Processing* 3 Layer Dengan *Library Nltk, Keras dan TensorFlow*



Gambar 3.1 Klasifikasi kalimat menggunakan metode NLP 3 layer

Dari diagram gambar 3.1 diatas merupakan alur mekanisme program yang dimana dataset di *training* menggunakan *framework tensorflow* dan *keras*, Selanjutnya setelah selesai *training*, maka dilakukan penyimpanan *model data training* yang nantinya data *model training* tersebut digunakan untuk klasifikasi sebuah *talent mapping*, maka pada alur diagram proses *testing* sistem pada aplikasi *client* dan *server* menjadi seperti pada gambar 3.2 sebagai berikut:



Gambar 3.2 Alur proses sistem aplikasi identifikasi minat bakat seseorang

Pada gambar 3.2 menjelaskan alur sebuah sistem aplikasi identifikasi minat bakat seseorang, dimana ketika *client* memulai menjalankan aplikasi dan memulai menginputkan data melalui *google API speech recognition* yang secara bersamaan konversi suara ke *text* dengan bahasa Indonesia. Saat memulai pengiriman data ke layanan *server* dengan alamat source “*http://IP Address: 5000/chat*” dan *parameter* ‘*select*’ dan ‘*chatInput*’. Pada layanan *server* menerima data dari *client* dan memulai proses sistem *NLP*. Setelah selesai proses, hasil akhir dikirim kembali ke *client* dan *client* menampilkan data hasil tersebut.

3.2 Setting Virtual Environment

Setup keperluan untuk pembuatan *project*, disini penulis menggunakan *anaconda* sebagai *virtual env*. *Virtual env* adalah sebuah *directory* untuk menyimpan *library (package)* tiap *project* agar tidak tercampur dengan *library* utama. Dikarenakan pada *project NLP* diperlukan *library* khusus seperti *tensorflow*, *keras*, *flask*, *nlk*. Lalu untuk *library pickle* dan *yaml library* fungsinya untuk menyimpan *model* yang telah di *training*.

3.3 Dataset

```

{"intents": [
  {"tag": "softwareTest",
   "patterns": ["lulusan gelar sarjana dalam ilmu komputer, teknik komputer, sistem informasi dan informatika", "Pengetahuan terkini tentang desain pengujian perangkat lunak dan metodologi pengujian", "komunikasi yang baik dan keterampilan berpikir kritis", "bekerja santai, fleksibel dan serius", "Pengetahuan kerja tentang teknik pengujian dan kompatibilitas dengan berbagai program perangkat lunak", "Paham akan dasar matematika, ipa, fisika, tik", "mampu bekerja individu, kolaborasi dan tim", "passionate, kreatif, dan inovatif detail", "cepat beradaptasi", "paham akan dasar elektronika", "bisa mengoperasikan sistem operasi seperti linux, windows, As400 komputer", "disiplin, serius, jujur, dan bertanggung jawab", "Meninjau persyaratan perangkat lunak dan menyiapkan skenario pengujian", "mahir berbahasa inggris secara lisan dan tulisan", "suka akan menonton film / video tutorial", "Menjalankan tes pada kegunaan perangkat lunak", "Menganalisis hasil pengujian pada dampak database", "kesalahan / bug, dan kegunaan", "bisa menggunakan program perangkat lunak", "mampu bekerja dibawah tekanan", "rapi dan terorganisir", "rasa ingin tau yang tinggi", "paham akan tema,estetika, dan bug game", "Menyiapkan laporan pengujian perangkat lunak, pengujian dan memprioritaskan kegiatan pengujian", "menyukai bidang IT minimal 1 tahun pengalaman di posisi", "identifikasi permasalahan / cacat", "bisa memajemen waktu, dan proyek", "menyukai game dan sering bermain game", "menyukai semua genre game", "Keterampilan yang Diperlukan Pengujian Otomatis, Whitebox dan Blackbox, Penguji Perangkat Lunak, SDLC, SQL, RDBMS, Pemrograman", "Familiar dengan bahasa pemrograman Java, Golang, Python, Cobol, .Net, Ruby, Node.Js, Rpg"],
   "responses": ["Software Testing", "Game Testing"],
   "context_set": ""
  },
  {"tag": "programmer",
   "patterns": ["mampu mengembangkan perangkat lunak", "Gelar pendidikan sarjana kuliah dalam ilmu komputer, teknik komputer, sistem informasi dan informatika", "memiliki pengalaman dibidang IT 1 sampai 3 tahun", "Menguasai bahasa pemrograman api, sql, dan python", "smk minimal jurusan teknik komputer jaringan (TKJ) / rekayasa perangkat lunak (RPL)", "menguasai Database MySQL, PostgreSQL / SQL Server", "memiliki skill bahasa html, php, c, css", "cepat belajar dan proaktif", "mampu bekerja individu, kolaborasi dan tim", "passion, kreatif dan inovatif detail", "bisa mengolah sebuah data terstruktur", "tekun, kerja keras dan berhati-hati", "komunikasi yang baik dan keterampilan berpikir kritis", "bisa mengoperasikan sistem operasi seperti linux, windows komputer", "mampu mengoperasikan program editor seperti sublime, visual code, visual studio", "Paham akan dasar matematika, ipa, fisika dan tik", "ng dan menguji struktur komputer", "bisa mengatur manajemen waktu", "mampu menggunakan salah satu framework laravel, codeigniter", "menguasai server pos, get, remove", "pemecah masalah kesalahan sistem", "menulis instruktur komputer", "memahami alur flowchart dan cara kerja", "suka mempelajari teknologi baru", "memiliki kemampuan komunikasi"],
  }
]

```

Gambar 3.3 Dataset IT Skill

Pada gambar 3.3 *Dataset* yang digunakan dalam klasifikasi kalimat untuk *Talent Mapping* ini merupakan *dataset* yang diambil dari *website Jobs Requirement*. *Dataset* yang diambil ada 130 data *talent* dengan *requirements* hobi yang digunakan untuk pengujiannya, dari penulis mengambil *talent* khusus IT yang dikonversikan ke bahasa indonesia dan bentuk ke format *.json* yang nantinya dapat digunakan untuk *tag label* pembuatan *dataset* baru dengan kriteria yang diperlukan. Pada *dataset* 130 data *talent* dengan mengkategorikan *talent skill requirements* sama pengambilan data *user* 4 unsur: kepribadian, kemampuan, aktivitas dan hobi.

3.4 Metode *Talent Mapping*

Untuk metode klasifikasi bakat yang dipakai penulis mengambil dari buku Tasya Syavitri Pradani dengan judul “*Talent Mapping Assesment Result*”. Dengan metode *Talent Mapping* dengan pengambilan data kegiatan / hobi disukai dan tidak disukai serta produktif dan tidak produktif yang nantinya memulai klasifikasi dengan 4 kriteria dominan “*Relating*”, “*Thinking*”, “*Striving* dan “*Impact*”. Dalam pengerjaan *project* ini menambahkan *parameter* untuk *klasifikasi* 2 data yaitu “*Recomended*” dan “*Not Recomend*”. Untuk pembuatan pertanyaan diberikan penggambaran kuadran *klasifikasi*. Berikut pembagian tabel kuadran yang dipakai ditunjukkan pada gambar 3.4.

DISUKAI	KUADRAN DUA (II)	KUADRAN SATU (I)
TIDAK DISUKAI	KUADRAN EMPAT (IV)	KUADRAN TIGA (III)
	TIDAK PRODUKTIF	PRODUKTIF

Gambar 3.4 Kuadran untuk klasifikasi

Dari gambar 3.4 tabel kuadran tersebut diperlukan sebuah pengambilan data, maka untuk mendapatkan sebuah data yang bagus diberikan beberapa pertanyaan

dengan inti pokok tema masalah untuk memasukkan klasifikasi dalam tabel, contoh pertanyaan tersebut berisi:

1. Kegiatan / hobi apa yang disukai dan produktif?
2. Kegiatan / hobi apa yang disukai tetapi tidak produktif?
3. Kegiatan / hobi apa yang tidak disukai tetapi produktif?
4. Kegiatan / hobi apa yang tidak disukai dan tidak produktif?

Dari hasil data jawaban pertanyaan tersebut diolah dan mengeluarkan *result* referensi berupa “*Recomended*” dan “*Not Recomend*ed” dengan *persentase* diakhir. Pertanyaan yang dipakai oleh penulis dari hasil riset dan validasi pembimbing disimpulkan untuk kuadran 1 dan 3 sebagai “*Recomended*” sebab meski pada kuadran 3 tidak suka tetapi produktif dan kerja keras. Hal itu menjadikan dari hal tidak suka menjadi suka yang membuat masuk pada pekerjaan “*Recomended*”, sedangkan untuk kuadran 2 dan 4 sebagai “*Not Recomend*ed” sebab meski pada kuadran 2 yang diawal suka tapi tidak produktif karena malas dan membuat tidak konsisten dalam mengerjakan sesuatu dan akhirnya melupakan hal suka menjadi tidak suka. Karena tidak konsisten dan kemalasan mengubah hal suka menjadi tidak suka.

Berikut 10 pertanyaan tiap usia yang telah dibuat berdasarkan tiap kuadran:

1. Usia anak-anak
 - a. 2 Hal apa yang kamu sukai saat dirumah dan disekolah?
 - b. 3 hal yang membuatmu bahagia?
 - c. 3 pelajaran sekolah yang kamu sukai dan membuatmu bersemangat?
 - d. Sebutkan apa cita-cita kamu?
 - e. 2 hobi apa yang kamu sukai dan sering kamu lakukan?
 - f. 2 hal apa yang lakukan jika ada pelajaran kosong disekolah?
 - g. 2 hal apa yang tidak kamu sukai saat dirumah dan disekolah?
 - h. 3 pelajaran sekolah apa yang tidak kamu sukai?
 - i. 2 hal apa yang membuat kamu marah?
 - j. 2 hal apa yang membuat kamu takut?
2. Usia Remaja
 - a. 2 hal apa yang kamu sukai saat dirumah dan saat disekolah?
 - b. 3 pelajaran yang kamu sukai dan membuatmu bersemangat saat bersekolah?

- c. 2 hal apa yang kamu sukai dari diri kamu?
 - d. Sebutkan hal apa saja yang menarik dari teknologi?
 - e. 3 hobi apa yang kamu sukai dan sering kamu lakukan di bidang teknologi?
 - f. 3 hal apa yang kamu lakukan jika ada pelajaran kosong ketika bersekolah?
 - g. 3 pelajaran apa yang tidak kamu sukai ketika bersekolah?
 - h. 3 hal apa yang tidak kamu sukai dari diri kamu?
 - i. 3 hal apa saja yang membuat kamu tidak suka dari teknologi?
 - j. 3 bidang IT apa saja yang tidak kamu sukai?
3. Dewasa
- a. 3 kata yang menggambarkan diri kamu?
 - b. 3 hobi apa yang paling banyak menghabiskan waktumu di bidang teknologi?
 - c. Sebutkan apa saja yang membuat kamu tertarik belajar teknologi?
 - d. 3 tujuan yang ingin kamu capai dimasa depan?
 - e. Apa yang bisa kamu lakukan untuk bisa mencapai tujuan tersebut?
 - f. 3 hal apa yang kamu suka lakukan pada waktu senggang?
 - g. 3 hal apa yang tidak kamu sukai dari diri kamu?
 - h. 3 hal apa saja yang kamu tidak sukai dari teknologi?
 - i. 3 bidang IT apa saja yang tidak kamu sukai?
 - j. Alasan kamu tidak menyukai 3 bidang IT tersebut?

3.5 Persentase *Output*

Persentase *output* adalah sebuah nilai dari kombinasi banyaknya kata kerja dasar tanpa ada sambungan dan imbuhan kata data *user* yang sesuai dengan banyaknya *pattern* dari *dataset* yang kemudian dipersentasekan. Nilai persentase *output* di dapatkan dengan menggunakan rumus persamaan (1) sebagai berikut:

$$\text{Persentase } Output = \frac{\sum_i^n True\ Input}{\sum_i^n Pattern} * 100\% \dots\dots\dots (1)$$

Persamaan (1) merupakan perhitungan dari banyaknya kecocokan kata dari inputan user yang dibagi dengan banyaknya *pattern* didalam *dataset*, tiap *variable*-nya dijelaskan sebagai berikut.

$$\sum_i^n True\ Input \quad : \text{ Banyaknya inputan data yang benar.}$$

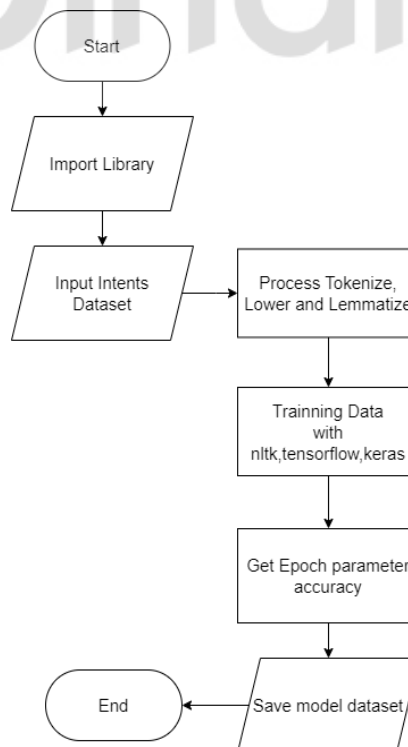
$$\sum_i^n Pattern \quad : \text{ Banyaknya requirement pada jobs dataset.}$$

3.6 Semantik Role Labeling (SRL)

SRL merupakan *type labeling* yang ada pada *NLP* yang dapat digunakan untuk mengidentifikasi argumen dari predikat dalam kalimat yang biasanya disebut “*bag of word*” pada sistem *NLP*, maka dengan hasil *training dataset* memulai pengambilan data suara dari *user* menggunakan *speech recognition*. Hasil dari pengambilan tadi akan di ubah bentuk teks kemudian di *labeling* agar bisa memulai klasifikasi perbandingan dari *dataset model* yang sudah di *training* dengan data *user*.

3.7 Training Server Process

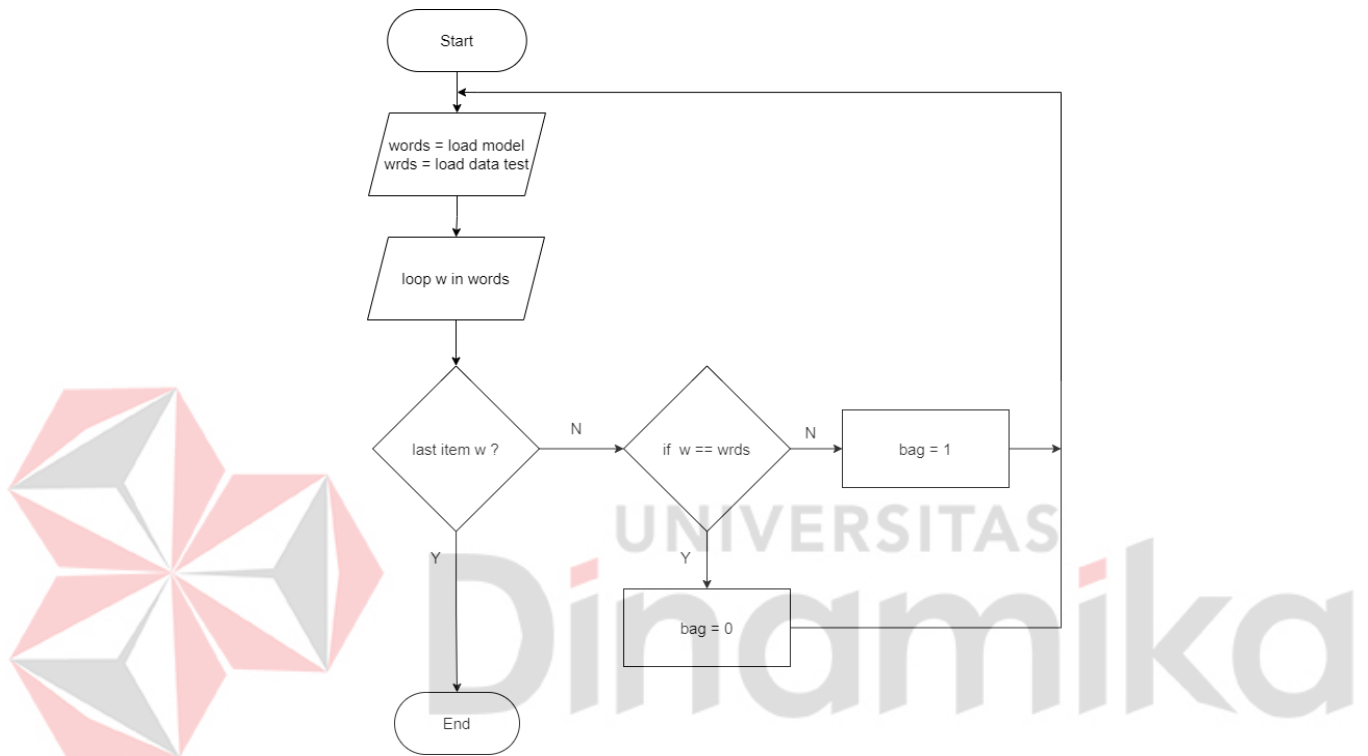
Pada proses ini, Mulai dari menjalankan program lalu *import library* yang digunakan, kemudian memasukkan *dataset* yang telah di *customize* hingga proses *labeling*. Pada proses ini pengecekan data kalimat yang diambil hobi, kemampuan, kepribadian, dan aktivitas untuk *tokenize* kemudian di masukkan ke proses *training*, pada proses *training* dengan 1000 *epoch* dan 3 *layer neuron*. Hasil dari *training* memunculkan nilai *loss*, dan *accuracy* disimpan data model *training* dengan format *YAML*. Kemudian mengubah format ke *HDF5* untuk memperkecil ukuran, agar ketika pembacaan *load* model lebih cepat untuk mengolah data.



Gambar 3.5 Flowchart Training Server Process

3.8 Testing Server Process

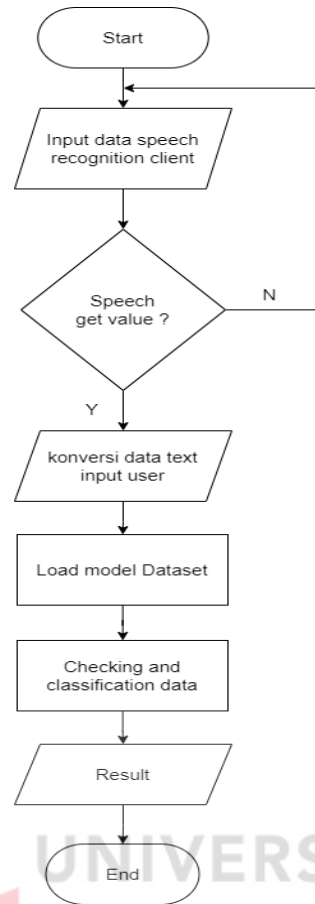
Pada proses ini, pengecekan data menggunakan data pattern yang sudah *ditokenize* dan data hasil dari yang telah di *training*, kemudian memulai pencocokan data. Pada proses akhir memberikan data jika data testing pattern dengan data hasil training cocok, maka menyimpan data nilai 1 dan jika tidak cocok, maka menyimpan data nilai 0. Dapat dilihat pada gambar 3.6 berikut.



Gambar 3.6 Flowchart Testing Server Process

3.9 Testing Client Process

Pada proses ini, pengambilan data dilakukan secara langsung menggunakan *speech recognition* dan data suara tersebut diubah menjadi teks lalu memulai *SRL* tetapi ketika tidak mendengar sebuah suara atau tidak ada data sesuai, maka mengulang program pengambilan suara. Dari hasil pertanyaan dan jawaban tersebut, maka mendapatkan data *input client*, kemudian *load* model, kemudian memulai pencocokan data. Pada proses akhir mendapatkan hasil *talent mapping* sesuai dengan apa yang ada didata. Isi dalam *talent mapping* nanti ada “*Talent Strength*”, “*Talent Risk*”, yang telah di klasifikasikan hasilnya. Dapat dilihat pada gambar 3.7 berikut.



Gambar 3.7 Flowchart Testing Client Process

3.10 Training Program Natural Language Processing

Penggunaan metode *Pre-Processing Training* pada *Natural Language Processing* ini menggunakan *Framework TensorFlow* dan *keras* dengan melalui 3 *layer* dengan penggunaan aktivasi *ReLU* untuk *input* dan prosesnya. Sedangkan untuk *output* menggunakan *Softmax*, berikut pada gambar 3.8 program *training Natural Language Processing*:

```

# pembuatan layer neural network
myChatModel = Sequential()
myChatModel.add(Dense(128, input_shape=[len(words)], activation='relu'))
myChatModel.add(Dense(64, activation='relu'))
myChatModel.add(Dense(len(labels), activation='softmax'))

# optimize model
myChatModel.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])

epochs = 1000
# train model
history = myChatModel.fit(training, output, epochs=epochs, batch_size=8,
verbose=1)
  
```

Gambar 3.8 Program data training Natural Language Processing

3.11 Bag of Word NLP Program

Bag of Word sebuah teknik teks *modeling* pada *NLP*, dengan metode tersebut bisa mengekstraksi sebuah teks bentuk kalimat menjadi per kata. Metode ini digunakan untuk mempermudah mengambil data yang di *training* dan identifikasi, berikut program *bag of word* pada yang dipakai oleh penulis:

```
def bag_of_words(s, word):
    list_check = 0
    bags = [0 for _ in range(len(word))]
    # pemisahan kalimat mejadi per kata dari input suara
    s_words = tokenizer.tokenize(s)
    s_words = [stopwords.remove_stopword(txt.lower()) for txt in s_words]
    s_words = [lemmatizer.lemmatize(word) for word in s_words if word not
in delete_word]

    for se in s_words:
        for i, w in enumerate(words):
            if w == se:
                list_check += 1
                bags[i] = 1

    # kata yang valid dan cocok dengan dataset
    list_error = ((len(s_words)) - list_check) / (len(s_words)) * 100
    list_acc = (list_check / (len(s_words))) * 100
    print("Data Sukses : {:.2f}%".format(list_acc))
    print("Data Loss : {:.2f}%".format(list_error))
    return numpy.array(bags), s_words
```

Gambar 3.9 *Bag of Word NLP Program*

Pada gambar 3.9 merupakan sebuah program dengan cara kerjanya menggunakan *function* model *bag_of_words*. Didalam *function bag_of_words* dengan parameter hasil data *input* dan *parameter* hasil *training words*. Kemudian membaca *range* dari hasil nilai *training* yang nantinya memulai identifikasi kecocokan dengan *input*. Fungsi *tokenizer.tokenize* merupakan sebuah fungsi untuk memecah kalimat menjadi *tag label* lalu diubah ke format *array*. Fungsi *lemmatizer.lemmatize* merupakan sebuah fungsi yang menjadikan kata dasar, menghilangkan kata imbuhan seperti awalan kata dan akhiran kata. Setelah itu diubah ke nilai bentuk angka menggunakan fungsi *enumerate* yang kemudian memulai pencocokan sebuah hasil *training* dengan masukkan *input*.

3.12 Pembuatan RESTful API

RESTful API sebuah *application interface* yang digunakan sebagai perantara komunikasi dari *server* ke *client*. Penulis menggunakan *API* karena untuk pengiriman sebuah hasil keluaran indetifikasi *talent* ke sebuah *client* lewat *device*

android, pada *android* pembuatan aplikasi *interface* yang memudahkan *user* untuk berinteraksi sebuah *server* di ibaratkan sebagai *chatbot* yang memberikan pertanyaan dan hasil keluaran dari pertanyaan *user*. Berikut *Package program* pembuatan *API*:

```
from flask import Flask, request, jsonify
from flask_restful import Resource, Api
from main import chatWithBot
import re

app = Flask(__name__)
api = Api(app)
```

Gambar 3.10 Pembuatan *RESTful API*

Pada gambar 3.10 *Flask* atau *Flask RESTful* sebuah *package API* yang digunakan sebagai aplikasi komunikasi antar *server* dan *client*, *Request* adalah sebuah *package* yang meminta data ke *client* dengan *Id Address* yang diberikan dan *jsonify* sebuah *package* yang mengubah sebuah data ke format *json string* agar mempermudah pengiriman antar *server* dan *client*.

3.13 Pembuatan Desain *Mockup* Aplikasi



Gambar 3.11 Pembuatan Desain *Mockup* Aplikasi

Pada gambar 3.11, merupakan hasil desain *interface mockup aplikasi*. Pembuatan desain *mockup* dari aplikasi sendiri mempertimbangkan dari *responsibility* dan kenyamanan *user* pada *device*. *Mockup* sendiri adalah konsep

yang lebih dalam untuk menyampaikan desain visual sebuah aplikasi, seperti warna, gambar, dan tipografi dengan secara detail dan terperinci.

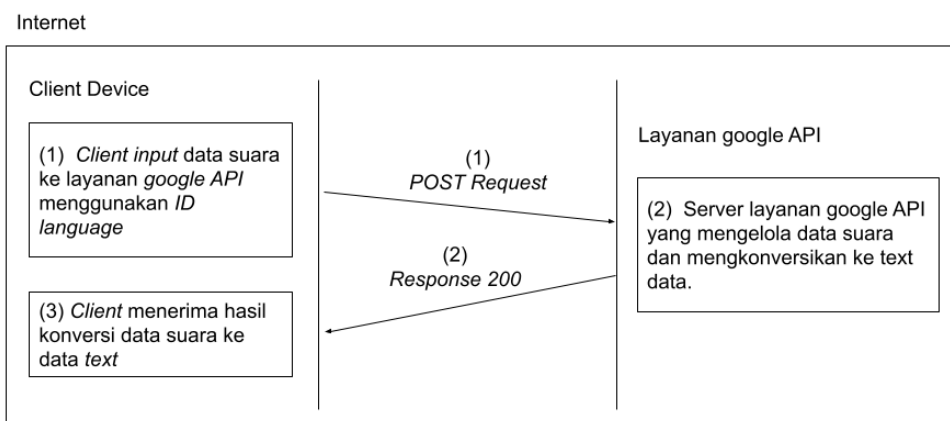
3.14 Pembuatan Aplikasi di *Android Studio*

Pembuatan aplikasi diperlukan persiapan sebuah bahan material yang telah dibuat pada desain *mockup* dengan minimal *SDK 21* dan target *SDK 33* dan beberapa *package library dependencies*. Berikut *package* penting yang diperlukan dalam pembuatan:

1. *Retrofit2* yaitu *package* yang memudahkan untuk terhubung ke layanan *web Apps* dengan menerjemahkan *API* ke *interface client* dengan *server*.
2. *Circle Indicator* yaitu *package* yang menampilkan sebuah indikator tiap halaman *interface (slider circle)* antar *activity*.
3. *CardView* yaitu *package* yang berfungsi untuk membungkus sebuah data program *layout activity*.
4. *RecyclerView* yaitu *package* untuk menampilkan data ke *list* secara *dinamis*.

3.14.1 Konversi Data Suara Menggunakan *Google API Speech Recognition*

Pada masukan dari user yaitu menggunakan *speech recognition* dari *Google API*. *Google API* sendiri yaitu sebuah layanan *API* untuk mikrophone yang ada pada tiap *smartphone android* maupun *ios*, dan juga terdapat sebuah fitur yang mana pekonversian sebuah ‘data suara ke *text*’ dan ‘data *text* ke suara’. Dalam hal ini bisa dilihat pada gambar 3.12 untuk alur pengiriman data suara sebagai berikut.

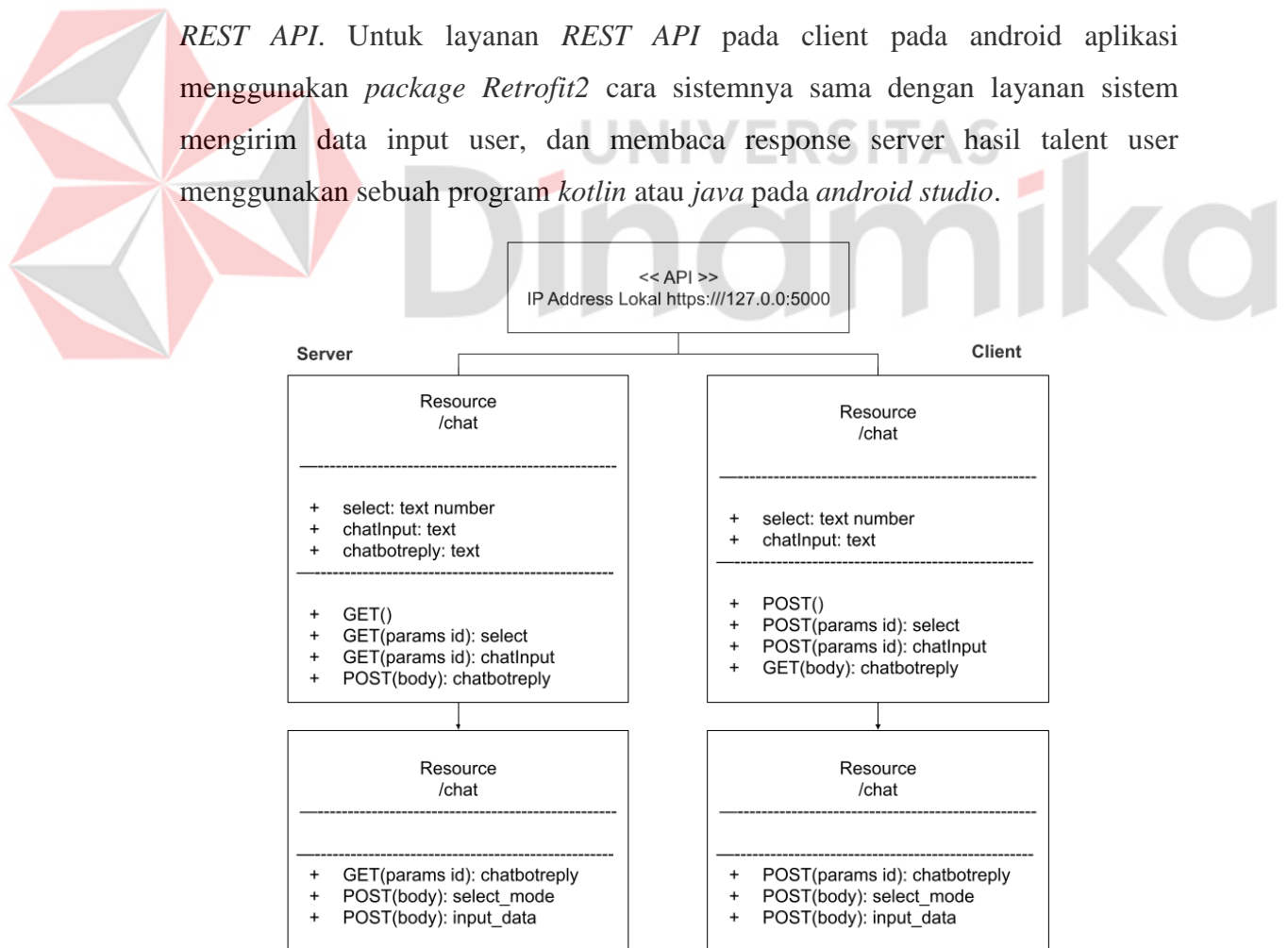


Gambar 3.12 Diagram Komunikasi *Client* dan Layanan *Google API*

Pada gambar 3.12 menjelaskan sebuah alur pengiriman data suara dari *google API* ke *client* yang dimana saat *client* menjalankan proses untuk memulai menginputkan suara, maka *google API speech recognition* memulai merekam suara secara *real-time* yang kemudian masukkan suara diteruskan ke *server google API* secara bersamaan memberikan hasil konversi dan koreksi berupa *text* yang pastinya menggunakan internet. Dari hasil *text* tersebut nantinya masuk ke data *client* sebagai simpanan data *input* yang siap dikirim ke *server local*.

3.14.2 Perancangan Layanan *REST API*

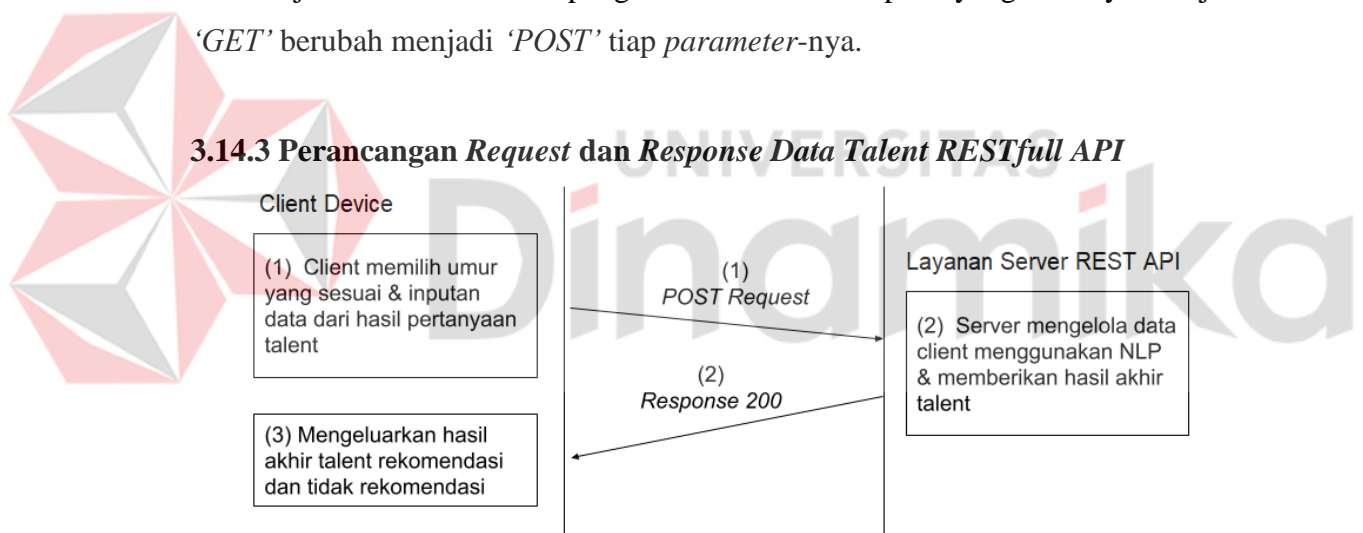
Perancangan layanan *REST API* meliputi beberapa sebuah layanan, diantaranya sistem pembacaan data input user dan sistem pengiriman hasil data talent yang telah diolah oleh sistem *NLP*. Server *REST API* dibuat dengan bahasa *python* dengan tambahan beberapa *package Jsonify* yang guna membantu layanan *REST API*. Untuk layanan *REST API* pada client pada android aplikasi menggunakan *package Retrofit2* cara sistemnya sama dengan layanan sistem mengirim data input user, dan membaca response server hasil talent user menggunakan sebuah program *kotlin* atau *java* pada *android studio*.



Gambar 3.13 Diagram *endpoints* pada layanan *REST API*

Berdasarkan gambar 3.13 memiliki *IP Address* pemanggilan *server* yaitu *IP* lokal dengan *port 5000*. Kemudian terdapat satu *endpoints* yang sama yaitu */chat* tetapi memiliki cara pemanggilan layanan berbeda tiap *client* dan *server* pada bagian *endpoints* di *server* terdapat 3 *parameter* dengan 2 status metode *'POST'* dan *'GET'*, untuk *parameter 'select'* dengan *type data text number* guna untuk perpindahan *mode* pemanggilan dalam *server* seperti *"skill recommended"*, *"skill not recommended"* dan *"dominan"*. Fungsi *parameter 'select'* cuman menerima saja dalam bagian *server*, selanjutnya untuk *parameter 'chatInput'* dengan *type data text* guna untuk menerima inputan atau kiriman dari user yang nantinya diolah pada tiap *mode* dengan *parameter 'select'* yang diberikan. Untuk *parameter 'chatbotreply'* dengan *type data text* guna hanya mengirim hasil sebuah olah data pada *parameter 'chatInput'* ke *client*. Dari semua 3 *parameter* tersebut untuk bagian *client device* cara kerja sama cuman cara pengambilan berbeda seperti yang awalnya menjadi *'GET'* berubah menjadi *'POST'* tiap *parameter*-nya.

3.14.3 Perancangan *Request* dan *Response Data Talent RESTfull API*

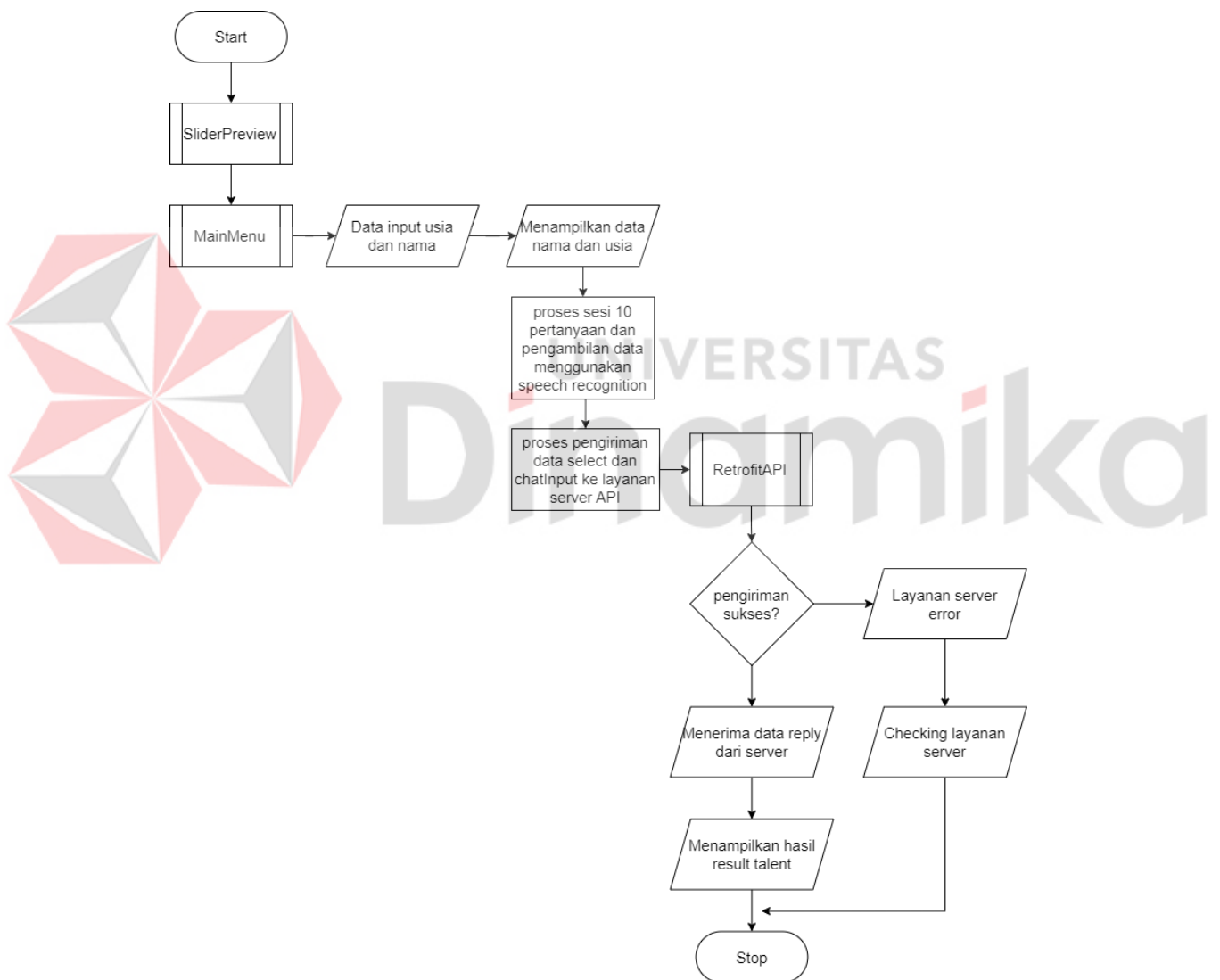


Gambar 3.14 Diagram Komunikasi *Client* dan Layanan *server*

Gambar 3.14, menunjukkan diagram komunikasi antara *client device* dan *server RESTfull API*. Komunikasi dimulai dari *Client Device* yang mengirimkan *params select* dan *chatInput* dengan metode *POST request*. Kemudian *server RESTfull API* mengecek *params "select"* dan *"chatInput"* yang telah terdaftar pada *server* dan merespon keluaran hasil olahan sistem *NLP* yang diberikan melalui *input client*. Setelah itu *client device* menerima *output data* dari *server RESTfull API*.

3.14.4 Sistem Pengambilan Data *Client*

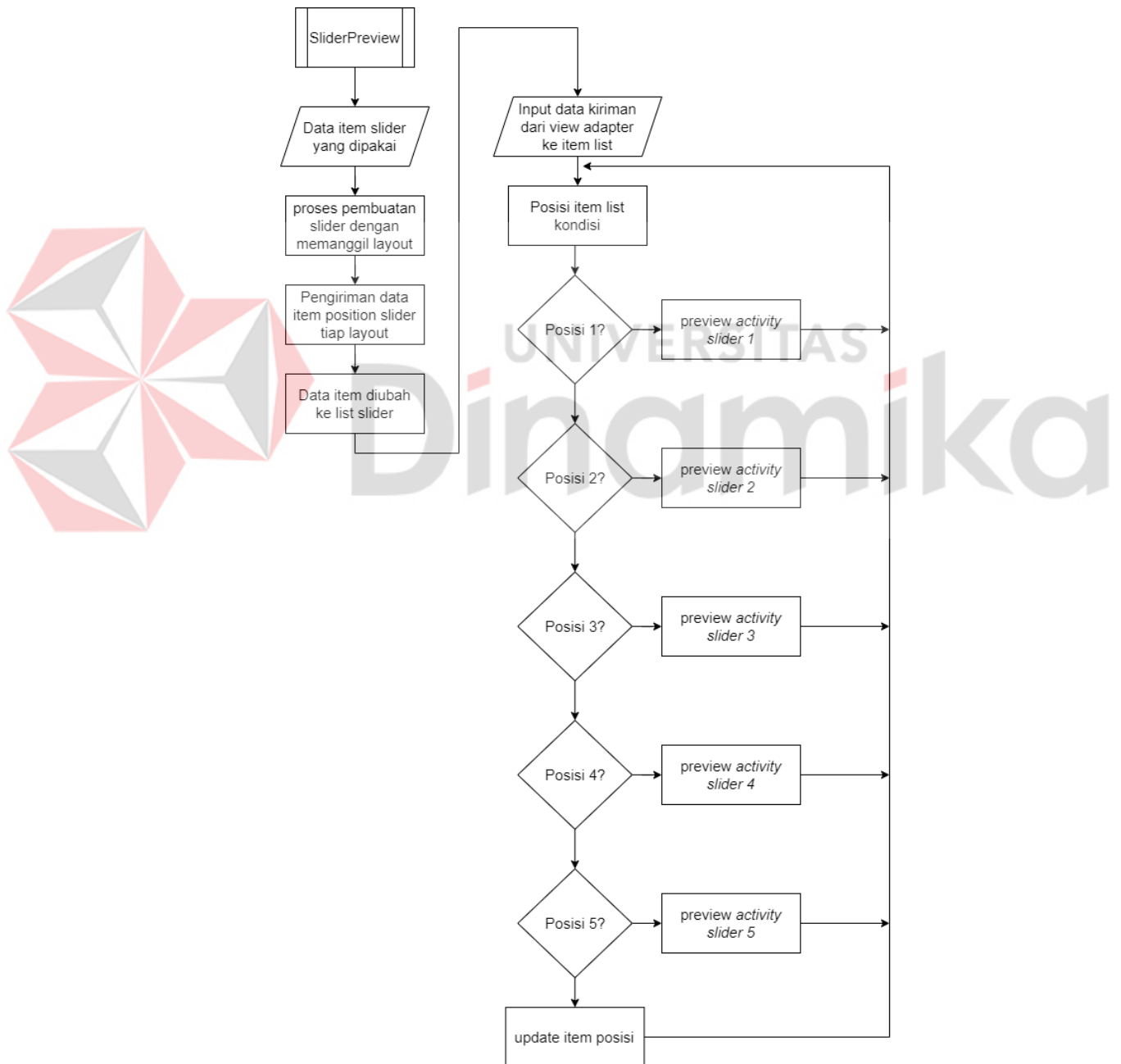
Pada proses ini, pengambilan data inputan *user* pada aplikasi *client*, sistem meminta *client* untuk memasukkan data usia dan nama untuk memulai sesi *activity* selanjutnya, setelah selesai pada *activity* sistem akan memberikan 10 pertanyaan yang telah dibuat dan diklasifikasikan. Kemudian setelah selesai menjawab pertanyaan, maka selanjutnya memulai pengiriman data *input* dan data *select* ke layanan *API* menggunakan *retrofit2 library* dan memberikan hasil akhir *talent* “*Recomended*” dan “*Not Recomendaded*”.



Gambar 3.15 *Flowchart* sistem pengambilan data *client*

3.14.5 Function Slider Preview Tutorial

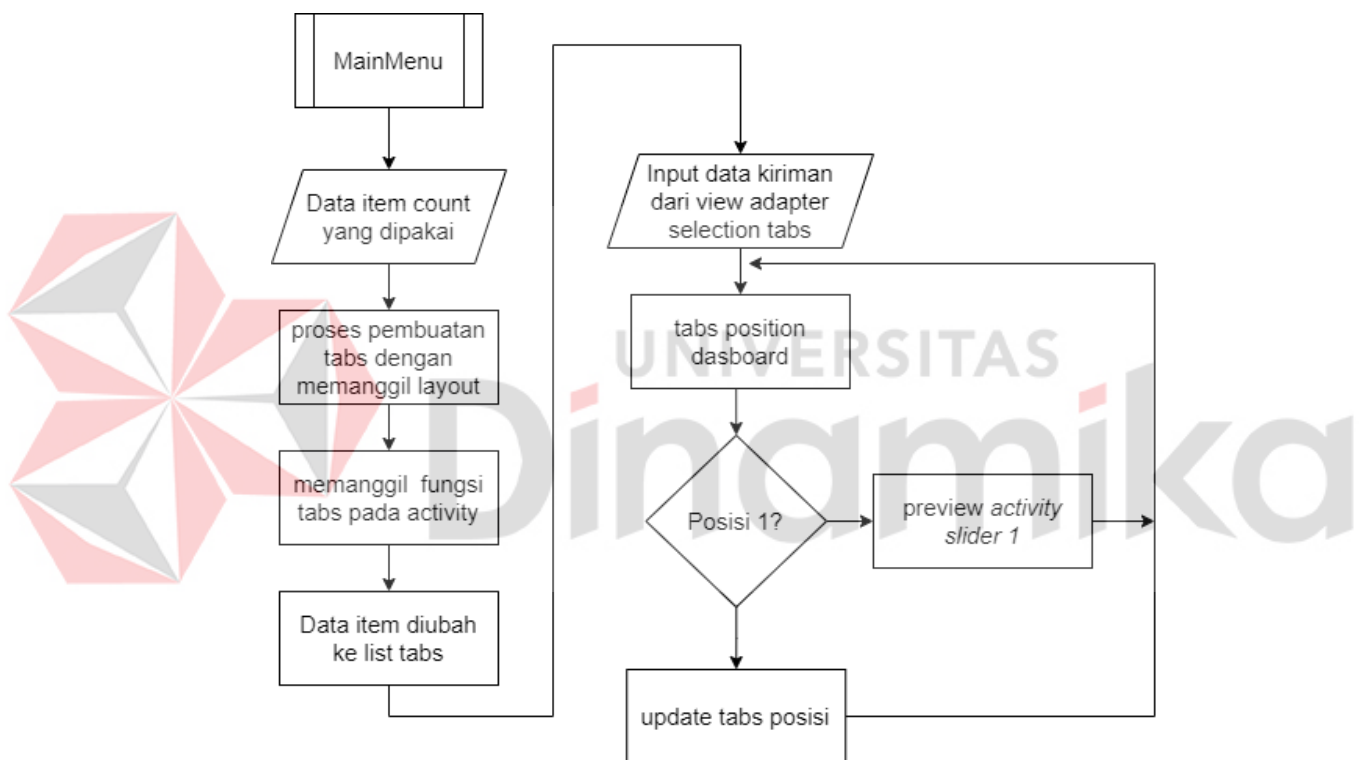
Pada proses ini, menjelaskan tentang alur kerja *circle slider* dengan penggunaan *library circle indicator* dan *recycle view* yang dimana sebuah *list item* data sebagai *counter* posisi pergeseran *indicator circle* tiap *activity layout* dan dibungkus dengan *function* nama adapter sebagai sistem pergeseran *slider*. Kemudian pada *activity* lain sebagai *main slider* dibuat pemanggilan *function adapter* dengan dibuatkan *switch case* sebagai percabangan pergantian *activity layout*. Bisa dilihat pada gambar 3.16 untuk alur *flowchart* berikut:



Gambar 3.16 *Function Slider Preview Tutorial*

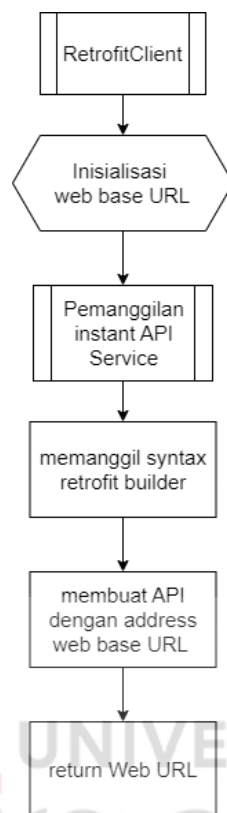
3.14.6 Function Main Menu

Pada *function* sistem *main menu* menjelaskan sebuah alur dari pengambilan *data activity layout* ke *activity* utama dimana sebuah data list yang nantinya digunakan sebagai petunjuk posisi pada *slider* dari *tabs slider*. Dengan mengetahui posisi dari *slider*, maka bisa menampilkan *layout activity dashboard main menu* tiap *layout* dan pada *project* ini untuk *activity tabs* hanya dibuat *single layout* tanpa ada *slider activity multiple layout* lain. Untuk alur flowchartnya bisa dilihat pada gambar 3.17 berikut:



Gambar 3.17 *Function Main Menu*

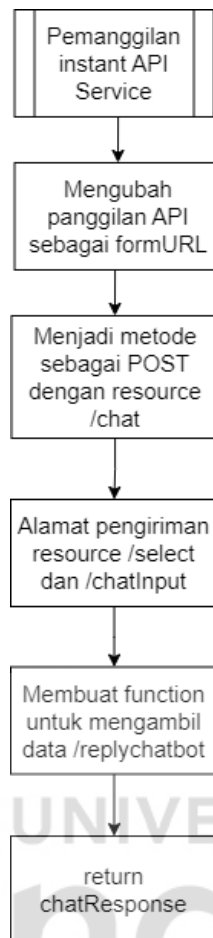
3.14.7 Function Layanan API Dari Retrofit2



Gambar 3.18 *Function Layanan API Dari Retrofit2*

Pada gambar 3.18, menjelaskan sebuah *flowchart function* pemanggilan sebuah layanan *API* menggunakan *library retrofit2* yang dimana membuat sebuah *value absolut* yaitu *URL server* atau *broker server* yang digunakan sebagai pengalamatan untuk komunikasi data. Kemudian memanggil sebuah *function API* yang dimana didalamnya berisi alamat *source parameter* tertentu seperti *"/chat"* guna sebagai kode alamat *chatbot server* *"/select"* sebagai menentukan metode dan *"/chatInput"* guna sebagai memasukkan sebuah data dari *input client* yang mau diolah dari sebuah layanan *server* utama. Kemudian memulai build sebuah Layanan *API* dengan *address dari value URL*, setelah selesai dibuat, maka memberikan data return untuk *WEB URL* yang digunakan.

3.14.8 Function API Service



Gambar 3.19 Function API Service

Pada gambar 3.19 menjelaskan sebuah *flowchat function* sistem *API Service* yang dimana cara pengiriman sebuah data *API* sebagai *form URL* data dengan menggunakan metode *POST* pada *address source* pada panggilan *chatbot* yaitu “/chat” , “/select”, “/chatInput” serta sebuah *function* untuk penerimaan data replay dengan alamat *source* “/replychatbot” data dari *reply* akan mengembalikan data dari panggilan *function*.

BAB IV HASIL DAN PEMBAHASAN

4.1 Pengujian *Pre-trained Model Network*

4.1.1 Tujuan Pengujian *Pre-trained Model Network*

Tujuan pengujian ini untuk mengetahui hasil akurasi dari *pre-trained model network*, serta menentukan *epoch* yang memiliki tingkat akurasi paling tinggi.

4.1.2 Prosedur Pengujian *Pre-trained Model Network*

Prosedur pengujian ini untuk mengetahui pengujian nilai akurasi tiap epoch, untuk langkah pengujian seperti berikut:

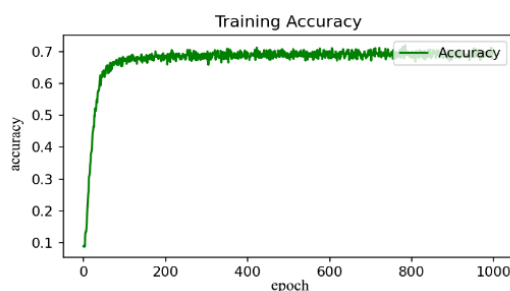
1. Memberikan *input* nilai *epoch* dengan nilai 100, 500, 1000 pada *pre-trained model network* digunakan.
2. Menjalankan proses *training* pada *pre-trained model network*.
3. Mengambil data *accuracy* serta data *loss* pada *epoch* dengan nilai 100, 500, 1000.

4.1.3 Hasil Pengujian *Pre-trained Model Network*

Tabel 4.1 Hasil Pengujian *Pre-trained Network*

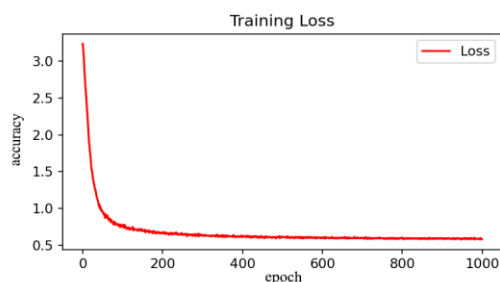
No	Epoch	Hasil Keluaran	
		Loss	Accuracy
1.	100	0.6450	0.7009
2.	500	0.6850	0.6812
3.	1000	0.5094	0.7306

Berdasarkan dari hasil evaluasi nilai *epoch* yang telah diuji coba didapat pada *epoch* ke-1000 mendapatkan *accuracy* tinggi dengan 0.7306 dan hasil didapatkan dari grafik memberikan informasi sebagai berikut:



Gambar 4.1 Hasil Akurasi

Pada gambar 4.1, diketahui hasil *training* mendapat nilai *accuracy* meningkat secara stabil pada titik *vertical* 0.7 selama bertambah *epoch*.



Gambar 4.2 Hasil *Loss*

Pada gambar 4.2, menjelaskan nilai hasil *training loss* mendapatkan nilai 0.5094 yang artinya bahwa validasi dari hasil *epoch* 1000 mendapatkan angka *loss* stabil.

4.2 Pengujian Akurasi Pengambilan Data Suara

4.2.1 Tujuan Pengujian Akurasi Pengambilan Data Suara

Tujuan pengujian ini adalah untuk *testing* sebuah keluaran suara dari berbeda jenis *gender* dan usia dengan jarak suara normal, serta menentukan *gender* dan usia mana yang memiliki tingkat akurasi paling tinggi dengan tiap keseluruhan uji coba.

4.2.2 Prosedur Pengujian Akurasi Pengambilan Data Suara

Prosedur pengujian ini untuk mengetahui akurasi dari pengambilan data suara tiap umur dan *gender*, untuk langkah pengujian seperti berikut:

1. Memberikan *device* yang serupa pada tiap usia dan *gender* berbeda serta dengan jarak suara normal.
2. Menjalankan aplikasi untuk memulai *test talent* dengan masukan dari *speech recognition*.
3. Mengambil data akurasi tiap pelafalan kata yang benar kemudian di konversikan ke *persentase* lalu memberikan keterangan tiap uji coba, pengujian ini di dilakukan selama 30 kali uji coba.

4.2.3 Hasil Pengujian Akurasi Pengambilan Data Suara

Tabel 4. 2 Hasil Uji Anak-anak

No	Keluaran Suara				Keterangan
	Laki-laki		Perempuan		
	Sukses (%)	Gagal (%)	Sukses (%)	Gagal (%)	
1.	66.67	33.33	69.23	30.77	kata yang diterima tidak sesuai Rata-Rata 32.05% di dataset.
2.	57.32	42.68	65.21	34.79	kata yang diterima tidak sesuai Rata-Rata 38.74% di dataset.
3.	64.32	35.68	51.32	48.68	kata yang diterima tidak sesuai Rata-Rata 42.18% di dataset.
4.	53.83	46.18	47.31	52.69	kata yang diterima tidak sesuai Rata-Rata 49.44% di dataset.
5.	40.79	59.21	57.32	42.68	kata yang diterima tidak sesuai Rata-Rata 50.95% di dataset.
6.	65.32	34.68	45.34	54.66	kata yang diterima tidak sesuai Rata-Rata 44.67% di dataset.
7.	63.82	36.18	65.65	34.35	kata yang diterima tidak sesuai Rata-Rata 35.27% di dataset.
8.	57.35	42.65	63.22	36.78	kata yang diterima tidak sesuai Rata-Rata 39.72% di dataset.
9.	69.23	30.77	52.44	47.56	kata yang diterima tidak sesuai Rata-Rata 39.17% di dataset.
10.	42.96	57.04	61.32	38.68	kata yang diterima tidak sesuai Rata-Rata 47.86% di dataset.
rata	58.16	41.84	57.84	42.16	kata yang diterima tidak sesuai Rata-Rata 42.00% di dataset.

Tabel 4.3 Hasil uji Remaja

No	Keluaran Suara				Keterangan
	Laki-laki		Perempuan		
	Sukses (%)	Gagal (%)	Sukses (%)	Gagal (%)	
1.	54.55	45.45	50.00	50.00	kata yang diterima tidak sesuai Rata-Rata 47.73% di dataset.
2.	71.13	28.87	46.67	53.33	kata yang diterima tidak sesuai Rata-Rata 41.10% di dataset.
3.	64.00	36.00	60.00	40.00	kata yang diterima tidak sesuai Rata-Rata 38.00% di dataset.
4.	69.23	30.77	60.87	39.13	kata yang diterima tidak sesuai Rata-Rata 34.95% di dataset.
5.	56.25	43.75	41.18	58.82	kata yang diterima tidak sesuai Rata-Rata 51.29% di dataset.
6.	62.07	37.93	51.28	48.72	kata yang diterima tidak sesuai Rata-Rata 43.33% di dataset.
7.	50.00	50.00	59.09	40.91	kata yang diterima tidak sesuai Rata-Rata 45.46% di dataset.
8.	48.39	51.61	73.33	26.67	kata yang diterima tidak sesuai Rata-Rata 39.14% di dataset.
9.	60.00	40.00	50.00	50.00	kata yang diterima tidak sesuai Rata-Rata 45.00% di dataset.
10.	48.28	51.72	51.52	48.48	kata yang diterima tidak sesuai Rata-Rata 50.10% di dataset.
rata	58.39	41.61	54.39	45.61	kata yang diterima tidak sesuai Rata-Rata 43.61% di dataset.

Tabel 4.4 Hasil uji Dewasa

No	Keluaran Suara				Keterangan
	Laki-laki		Perempuan		
	Sukses (%)	Gagal (%)	Sukses (%)	Gagal (%)	
1.	75.00	25.00	40.74	59.26	kata yang diterima tidak sesuai Rata-Rata 42.13% di dataset.
2.	80.00	20.00	63.88	36.12	kata yang diterima tidak sesuai Rata-Rata 28.06% di dataset.
3.	66.67	33.33	58.33	41.67	kata yang diterima tidak sesuai Rata-Rata 37.50% di dataset.
4.	55.00	45.00	72.76	27.24	kata yang diterima tidak sesuai Rata-Rata 36.12% di dataset.
5.	72.55	27.45	69.68	30.32	kata yang diterima tidak sesuai Rata-Rata 28.89% di dataset.
6.	83.87	16.13	55.56	44.44	kata yang diterima tidak sesuai Rata-Rata 30.29% di dataset.
7.	55.34	44.66	56.48	43.52	kata yang diterima tidak sesuai Rata-Rata 44.09% di dataset.
8.	52.09	47.91	63.33	36.67	kata yang diterima tidak sesuai Rata-Rata 42.29% di dataset.
9.	64.04	35.96	65.34	34.66	kata yang diterima tidak sesuai Rata-Rata 35.31% di dataset.
10.	56.01	43.99	55.21	44.79	kata yang diterima tidak sesuai Rata-Rata 44.39% di dataset.
rata	66.06	33.94	60.13	39.87	kata yang diterima tidak sesuai Rata-Rata 36.91% di dataset.

4.3 Hasil Perolehan Uji Coba Evaluasi *Talent Mapping*

4.3.1 Tujuan Memperoleh Uji Coba Evaluasi Hasil *Talent Mapping*

Tujuan untuk memperoleh uji coba hasil *talent mapping* adalah untuk mengevaluasi hasil *training* dengan inputan yang diberikan dari objek serta kecocokan hasil keluaran.

4.3.2 Prosedur Memperoleh Uji Coba Evaluasi *Talent Mapping*

Prosedur pengujian ini untuk mengetahui keluaran yang didapat pada uji coba *talent mapping*, untuk langkah pengujian seperti berikut:

1. Menjalankan aplikasi Alentapps lalu masuk ke mode *talent*.
2. Memulai *test* pertanyaan pada *talent mapping* untuk mendapatkan nilai *parameter* hasil.
3. Menampilkan nilai *parameter* hasil.

4.3.3 Hasil Nilai Pengujian Evaluasi *Talent Mapping*

Terdapat 3 parameter uji yang digunakan, yaitu: Hasil *Predict Talent Recommended*, Hasil *Predict Talent Not Recommended*, *Karakteristik*. Tiap aplikasi test pengujian memberikan 3 parameter diatas.

Tabel 4.5 Hasil Pengujian *Talent Mapping* Usia Anak-anak

Object	Uji Coba talent Mapping ke -	Hasil <i>Predict Recommended</i>		Hasil <i>Predict Not Recommended</i>		Karakteristik	
		Jobs	Cocok (%)	Jobs	Cocok (%)	Dominan	Cocok (%)
Musafaq	1	Game Testing	24.55	IT Support	23.62	Striving (Pekerja Keras)	20.06
		Software Testing	24.55	Technical Support	23.62		
		Software Quality	24.55	Technical Hardware	23.62		
		Net-Software	24.57	Software Quality	22.27		
		Software Develop	24.57	Software Develop	22.27		
	2	Game Testing	36.36	UX Research	31.03	Striving (Pekerja Keras)	33.33
		Software Testing	36.36	Aisstant Researcher	30.18		
		Creator Youtube	32.76	UI Research	30.18		
		Sound Engineer	32.00	UX Designer	30.18		
		Customer Services	32.00	UI Designer	30.46		
1	Data Security	25.06	Android Develop	23.50	Thinking (Pemikir)	21.28	
	Cyber Security	25.06	Java Develop	23.50			
	Database Security	25.06	Electric Engineer	23.33			
	Copywriter	24.10	Customer Service	23.33			
	Content Writer	24.10	Sound Engineer	23.33			
Vinanda	2	Social Media	34.43	Game Testing	32.47	Relating (Sosialisasi)	30.33
		Content Writer	34.43	Software Testing	32.47		
	Copywriter	34.43	Software Quality	32.58			
	UX Designer	33.77	Net-Softwawre	32.58			
	Graphic Design Arts	33.77	Software Develop	32.58			

Tabel 4.6 Hasil Pengujian *Talent Mapping* Usia Remaja

Object	Uji Coba talent Mapping ke -	Hasil <i>Predict Recomended</i>		Hasil <i>Predict Not Recomended</i>		Karakteristik	
		Jobs	Cocok (%)	Jobs	Cocok (%)	Dominan	Cocok (%)
Arya	1	Game Testing	32.67	Maintenance Engineer	32.20	Striving (Pekerja Keras)	33.33
		Software Testing	32.67	Computer Hardware Engineer	32.20		
		Software Quality	32.58	IT Research	32.20		
		Net-Software	32.58	IT Support	32.61		
		Software Develop	32.58	Technical Support	32.61		
	2	Game Testing	33.77	UX Designer	31.13	Thinking (Pemikir)	31.76
		Software Testing	33.77	Graphic Design Art	31.13		
		Software Quality	32.58	UI Designer	31.13		
		Net-Software	32.58	Maintenance Engineer	31.07		
		Software Develop	32.57	Computer Hardware Engineer	31.07		
Cahreza	1	Java Develop	32.17	Computer System Analyst	35.04	Striving (Pekerja Keras)	36.51
		Android Develop	32.17	Computer Engineer	35.04		
		Creator Youtube	31.03	IT Support	34.78		
		Maintenance Engineer	31.64	Technical Support	34.78		
		Computer Hardware Engineer	31.64	Technical Hardware	34.78		
	2	Creator Youtube	31.03	Maintenance Engineer	32.20	Thinking (Pemikir)	34.12
		UX Research	31.36	Computer Hardware Engineer	32.20		
		Assistant Research	31.36	IT Research	32.20		
		UI Research	31.36	Software Quality	32.58		
		UX Designer	31.13	Net-Software	32.58		

Tabel 4.7 Hasil Pengujian *Talent Mapping* Usia Dewasa

Object	Uji Coba talent Mapping ke -	Hasil <i>Predict Recomended</i>		Hasil <i>Predict Not Recomended</i>		Karakteristik	
		Jobs	Cocok (%)	Jobs	Cocok (%)	Dominan	Cocok (%)
Yoga	1	SCADA System Engineer	42.04	Sound Engineer	34.00	Thinking (Pemikir)	52.94
		HMI Develop	42.04	Customer Service	34.00		
		PLC Automation Engineer	42.04	Electrical Engineer	34.00		
		Software Quality	41.10	Machine Learning Engineer	34.09		
		Net-Software	41.10	Machine Learning Infra Eng	34.09		
	2	PLC Automation Engineer	42.67	Machine Learning Scientist	34.09	Thinking (Pemikir)	53.53
		HMI Develop	42.67	Machine Learning Infra Eng	34.09		
		IoT Engineer	39.17	Machine Learning Engineer	34.09		
		IoT Hardware Specialist	39.17	IoT Engineer	34.01		
		IoT Network Engineer	39.17	IoT Hardware Specialist	34.01		
1	Creator Youtube	26.44	Software Quality	31.82	Thinking (Pemikir)	51.76	
	Software Quailty	26.44	Net-Software	31.82			
	Net-Software Develop	26.52	Software Develop Engineer	31.82			
	Software Develop Engineer	26.52	Software Testing	30.46			
	UX Research	25.44	Game Testing	30.52			
Aviva	2	UX Designer	36.42	Software Quality	32.58	Striving (Pekerja Keras)	38.10
		Graphic Design Art	36.42	Net-Software	32.58		
	UI Designer	36.42	Software Develop	32.58			
	Social Media	36.07	Sound Engineer	32.00			
	Content Writer	36.07	Customer Service	32.00			

BAB V PENUTUP

5.1 Kesimpulan

Dari hasil penelitian tersebut, penulis mendapatkan kesimpulan sebagai berikut diantara lain:

1. Dari hasil testing keluaran suara pada laki-laki untuk usia anak-anak didapat nilai rata-rata data diterima 58.16%, untuk usia remaja didapat rata-rata data diterima 58.39%, dan untuk usia dewasa rata-rata data diterima 66.06%.
2. Dari hasil testing keluaran suara pada perempuan untuk usia anak-anak didapat nilai rata-rata data diterima 57.84%, untuk usia remaja didapat rata-rata data diterima 54.39%, dan untuk usia dewasa rata-rata data diterima 60.13%.
3. Dari uji coba ke beberapa orang mendapatkan hasil evaluasi pengujian *talent mapping* untuk usia anak-anak didapat kecocokan tertinggi pada jobs rekomendasi dengan *persentase* 36.36%, untuk usia remaja didapat kecocokan tertinggi pada jobs rekomendasi dengan *persentase* 33.77%, untuk usia dewasa didapat kecocokan tertinggi pada jobs rekomendasi dengan *persentase* 42.67%.
4. Dari hasil percobaan yang telah dilakukan, dapat diambil kesimpulan bahwa hasil *Pre-trained model network* yang digunakan dengan akurasi 73.06% dengan *epochs* 1000 dapat berjalan dengan baik dan lancar.
5. Dari hasil uji coba yang telah dilakukan ke beberapa orang mendapatkan hasil 58 *jobs talent* yang keluar atau terdeteksi dari 130 *jobs talent* pada *dataset*.

5.2 Saran

Pada penelitian selanjutnya diharapkan dapat menyempurnakan kekurangan dari Tugas Akhir ini dan dapat mengimplementasikan beberapa petunjuk yang diberikan oleh penulis, seperti berikut:

1. *Dataset* lebih kompleks dan *global* lagi karena pada penelitian penulis hanya berfokus pada pekerjaan IT.
2. Pengembangan metode *model training* menjadi *Reinforcement* agar tidak perlu mengelolah *dataset* dan *training* ulang lagi.
3. Mencoba metode *Talent Search* lain yang lebih baik dan lebih akurat.

4. Pembuatan survei pertanyaan mungkin bisa bekerjasama dengan *psychology* agar mendapatkan hasil keluaran *Talent mapping* lebih tinggi.
5. Penggunaan *loss function* yang berbeda, disarankan untuk mencoba *binary cross entropy*.
6. Perubahan *server web apps* masih lokal disarankan *deploy* atau *publish web apps API* agar mempermudah penggunaan *user* dan dapat diakses dimanapun secara jarak jauh.



UNIVERSITAS
Dinamika

DAFTAR PUSTAKA

- Adji, F., & Silvia, A. (2018). TALENT MAPPING: MENUJU JENJANG PENDIDIKAN YANG LEBIH TINGGI. *TALENT MAPPING: MENUJU JENJANG PENDIDIKAN YANG LEBIH TINGGI*, 115-124.
- Ahmed, M. (2012). Speech Recognition System. *ResearchGate*, 3-27.
- Android. (2021). *Android Studio*. Retrieved from Developer Android Documentation: <https://developer.android.com/studio>
- Anggraini. (2020). MENGIDENTIFIKASI MINAT BAKAT SISWA SEJAK USIA DINI DI SD ADIWIYATAMA. *MENGIDENTIFIKASI MINAT BAKAT SISWA SEJAK USIA DINI DI SD ADIWIYATAMA*, 161-169.
- Assauri, S. (2019). Rancang Bangun Sistem Pendukung Keputusan Untuk Pemilihan Jurusan Di Perguruan Tinggi Dengan Metode Fuzzy ELECTRE I Berbasis Web. *Rancang Bangun Sistem Pendukung Keputusan Untuk Pemilihan Jurusan Di Perguruan Tinggi Dengan Metode Fuzzy ELECTRE I Berbasis Web*.
- Asy'ari, M. Z. (2020). *Apa itu tensorflow? 3 Hal Penting Untuk Dipahami*. Diambil kembali dari auftechnique: <https://auftechnique.com/apa-itu-tensorflow/>
- Data, D. (2018, Oktober 02). *Implementasi Deep Learning Sederhana Menggunakan Keras*. Retrieved from Medium: <https://danau-data.medium.com/implementasi-deep-learning-sederhana-menggunakan-keras-3f5726f007e7>
- Dicoding. (2020, Oktober 14). *Android Studio*. Retrieved from Dicoding: <https://www.dicoding.com/blog/cara-install-android-studio-dalam-15-menit/>
- Dr.Dra Erni Murniarti, M. (2020). *PENGERTIAN BAKAT, CIRI-CIRI ANAK BERBAKAT, DAN IMPLIKASI PENDIDIKAN*. JAKARTA: UNIVERSITAS KRISTEN INDONESIA.
- Hudya. (2019, Desember 4). *Tutorial RESTful API dengan Flask Python Part 1 — Pengenalan & Instalasi*. Retrieved from Medium: <https://perogeremmer.medium.com/tutorial-restful-api-dengan-flask-python-part-1-pengenalan-instalasi-4836478ce651>
- Keras.io. (2022). *Documentation*. Retrieved from KERAS: <https://keras.io/>
- Kustanto, P., Fadjriya, A., & Rakhmat, P. (2021). Membaca Bakat Dengan Aplikasi Talents Mapping untuk Tenaga Pendidik dan Kependidikan SMK Catur Global Kota Bekasi. *Membaca Bakat Dengan Aplikasi Talents*

Mapping untuk Tenaga Pendidik dan Kependidikan SMK Catur Global Kota Bekasi.

Muhammad, A. (2020). NATURAL LANGUAGE PROCESSING : SENI OLAH BAHASA DALAM SAINS DATA. *ResearchGate*, 30-31.

Pradani, T. S. (2011). *TALENTS MAPPING ASSESSEMENT RESULT*.

Tensorflow.org. (2022). *Documentation*. Retrieved from TENSORFLOW: <https://www.tensorflow.org/>

Vargas, R., Mosavi, A., & Ruiz, R. (2017). DEEP LEARNING: A REVIEW. *Advances in Intelligent Systems and Computing. DEEP LEARNING: A REVIEW. Advances in Intelligent Systems and Computing*, 5.

Wiryanto, H. (2017, Agustus 14). *Reinventing Talent Management : Predictive Analytics Talent Management*. Retrieved from Rpubs: https://rpubs.com/heruwiryanto/Talent_analytics



UNIVERSITAS
Dinamika