

**PENERAPAN JARINGAN SARAF TIRUAN
MENGUNAKAN ALGORITMA *LEARNING*
VECTOR QUANTIZATION UNTUK KLASIFIKASI
ARAH KURSI RODA**

SKRIPSI



Oleh:
Zefxson Vrima Sihombing
150210182

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2020**

**PENERAPAN JARINGAN SARAF TIRUAN
MENGUNAKAN ALGORITMA *LEARNING*
VECTOR QUANTIZATION UNTUK KLASIFIKASI
ARAH KURSI RODA**

SKRIPSI

**Untuk memenuhi salah satu syarat
memperoleh gelar Sarjana**



**Oleh:
Zefxson Vrima Sihombing
150210182**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
TAHUN 2020**

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini saya:

Nama : Zefxson Vrma Sihombing
NPM : 150210182
Fakultas : Teknik dan Komputer
Program Studi : Teknik Informatika

Menyatakan bahwa “Skripsi” yang saya buat dengan judul:

PENERAPAN JARINGAN SARAF TIRUAN MENGGUNAKAN ALGORITMA *LEARNING VECTOR QUANTIZATION* UNTUK KLASIFIKASI ARAH KURSI RODA

Adalah hasil karya sendiri dan bukan “duplikasi” dari karya orang lain. Sepengetahuan saya, didalam naskah Skripsi ini tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip didalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia naskah Skripsi ini digugurkan dan gelar akademik yang saya peroleh dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya tanpa ada paksaan dari siapapun

Batam, 20 Februari 2020



Zefxson Vrma Sihombing
150210182

**PENERAPAN JARINGAN SARAF TIRUAN
MENGUNAKAN ALGORITMA *LEARNING*
VECTOR QUANTIZATION UNTUK KLASIFIKASI
ARAH KURSI RODA**

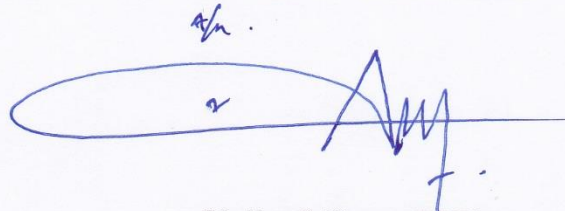
SKRIPSI

**Untuk memenuhi salah satu syarat
memperoleh gelar Sarjana**

**Oleh:
Zefxson Vrima Sihombing
150210182**

**Telah disetujui oleh Pembimbing pada tanggal
seperti tertera di bawah ini**

Batam, 20 Febuari 2020

A handwritten signature in blue ink, consisting of a large, stylized loop on the left and a series of sharp, vertical strokes on the right, all written over a horizontal line.

**Yulia, S.Kom., M.Kom.
Pembimbing**

ABSTRAK

Untuk dapat meningkatkan kualitas hidup Penyandang Disabilitas dengan gangguan mobilitas dapat dibangun sistem yang dapat mengendalikan kursi roda dengan gelombang otak. Dengan memanfaatkan kemajuan teknologi dan ilmu pengetahuan, pada penelitian ini akan dibangun sistem yang dapat mengendalikan kursi roda berdasarkan gelombang otak. Jaringan Saraf Tiruan (JST) dengan algoritma *Learning Vector Quantization* (LVQ) akan digunakan untuk mengklasifikasi arah kursi roda. Proses ini akan dijalankan di Android. Masukan yang digunakan untuk proses klasifikasi ini yaitu nilai *Attention* dan *Meditation* yang di dapatkan dari *MindWave Mobile 2* melalui sambungan *bluetooth*. Keluaran dari proses klasifikasi yang merupakan nilai arah dari kursi roda akan dikirim ke Arduino melalui sambungan Wifi. Nilai *Blink* yang juga didapatkan dari *MindWave Mobile 2* digunakan memicu Android untuk mengirimkan perintah ke Arduino untuk mengalihkan arus listrik yang akan menentukan pergerakan dari kursi roda. Nilai *Blink* akan mengalihkan nilai arus dari nol menjadi nilai arus sesuai dengan masukan yang diberikan oleh pengguna. Hasil pengujian menunjukkan JST dengan algoritma LVQ dapat digunakan untuk mengklasifikasikan arah kursi roda dengan akurasi sebesar 93.33% pada data yang diuji.

Kata kunci: JST; LVQ; Kursi Roda.

ABSTRACT

In order to improve the quality of life of people with disabilities with mobility impairments, a system that can control a wheelchair with brain waves can be built. By leveraging advances in technology and science, this research will build a system that can control a wheelchair based on brain waves. Artificial Neural Networks (ANN) with the Learning Vector Quantization (LVQ) algorithm will be used to classify the direction of the wheelchair. This process will run on Android. Inputs used for this classification process are Attention and Meditation values obtained from MindWave Mobile 2 via a Bluetooth connection. The output from the classification process will be sent to Arduino via a Wifi connection to change the direction value of the wheelchair. The output from the classification process which is the value of the direction of the wheelchair will be sent to Arduino via a Wifi connection. The Blink value also obtained from MindWave Mobile 2 is used to trigger Android to send commands to Arduino to switch electric current which will determine the movement of the wheelchair. Blink value will switch the voltage value from zero to the current value according to the input provided by the user. The test results show ANN with the LVQ algorithm can be used to classify the direction of a wheelchair with an accuracy of 93.33% in the tested data.

Keywords: ANN; LVQ; Wheelchair.

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Kuasa yang telah melimpahkan segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika Universitas Putera Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati. Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam;
2. Dekan Fakultas Teknik dan Komputer Universitas Putera Batam;
3. Ketua Program Studi Teknik Informatika;
4. Ibu Yulia, S.Kom., M.Kom. selaku dosen pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam;
5. Bapak Rahmat Fauzi, S.Kom., M.Kom. selaku dosen pembimbing akademik pada Program Studi Teknik Informatika Universitas Putera Batam;
6. Dosen dan Staff Universitas Putera Batam;
7. Kedua orang tua yang penulis sayangi Bapak L. Sihombing dan Ibu M. Lumban Gaol yang selalu memberikan doa serta dukungan kepada penulis;
8. Keluarga yang telah memberikan dukungan dan semangat kepada penulis;
9. Teman-teman penulis di Universitas Putera Batam, dan teman-teman yang telah membantu penulis;
10. Serta seluruh pihak yang telah membantu penelitian ini yang tidak bisa disebutkan satu persatu;

Semoga Tuhan Yang Maha Kuasa membalas kebaikan dan selalu mencurahkan bimbingan serta pertolongan-Nya, Amin.

Batam, 20 Februari 2020



Penulis

DAFTAR ISI

	Halaman
HALAMAN SAMPUL	i
HALAMAN JUDUL	ii
SURAT PERNYATAAN ORISINALITAS	iii
HALAMAN PENGESAHAN	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR RUMUS	xiii
BAB I PENDAHULUAN	
1.1. Latar Belakang.....	1
1.2. Identifikasi Masalah.....	5
1.3. Pembatasan Masalah.....	6
1.4. Rumusan Masalah.....	6
1.5. Tujuan Penelitian	6
1.6. Manfaat	7
1.6.1. Manfaat Teoritis.....	7
1.6.2. Manfaat Praktis	7
BAB II KAJIAN TEORI	
2.1. Teori Dasar	8
2.1.1. Kecerdasan Buatan.....	8
2.1.2. Jaringan Saraf Tiruan	9
2.2. Teori Khusus.....	9
2.2.1. Disabilitas	9
2.2.2. Arduino Uno Rev3.....	10
2.2.3. <i>MindWave Mobile 2</i>	13
2.2.4. ESP8266.....	19
2.2.5. L298N	21
2.2.6. Android	21
2.3. Model JST.....	22
2.4. <i>Software</i> Pendukung dan Perangkat Pemodelan.....	24
2.4.1. Java	24
2.4.2. Android Studio.....	25
2.4.3. Arduino IDE.....	26
2.4.4. <i>Fritzing</i>	26
2.4.5. <i>StarUML</i>	27
2.4.6. <i>Blender</i>	27
2.4.7. <i>Unified Modeling Language (UML)</i>	28

2.4.8. <i>Flowchart</i>	32
2.5. Penelitian Terdahulu	33
2.6. Kerangka Pemikiran	40
BAB III METODE PENELITIAN	
3.1. Desain Penelitian	41
3.2. Variabel Data Masukan dan Keluaran	42
3.3. Inisialisasi Parameter	44
3.4. Rancangan arsitektur Jaringan Saraf Tiruan	44
3.5. Perancangan Perangkat Keras dan Perangkat Lunak.....	44
3.5.1. Perancangan kerangka keras (<i>hardware</i>).....	45
3.5.2. Perancangan perangkat lunak (<i>software</i>).....	49
BAB IV HASIL DAN PEMBAHASAN	
4.1. Hasil Penelitian.....	64
4.1.1. Hasil Pengujian Klasifikasi Arah dengan LVQ.....	64
4.1.2. Hasil Perancangan Mekanik	66
4.1.3. Hasil Perancangan Elektrik.....	68
4.1.4. Hasil Perancangan Antarmuka Pengguna.....	69
4.1.5. Hasil Pengujian Perangkat Lunak	73
4.2. Pembahasan	75
4.2.1. Klasifikasi Arah dengan LVQ	75
4.2.2. Penerapan LVQ untuk Sistem Kontrol Kursi Roda.....	80
BAB V SIMPULAN DAN SARAN	
5.1. Simpulan.....	82
5.2. Saran	82
DAFTAR PUSTAKA	
LAMPIRAN	
Lampiran 1. Pendukung Penelitian	
Lampiran 2. Daftar Riwayat Hidup	
Lampiran 3. Surat Keterangan Penelitian	

DAFTAR TABEL

	Halaman
Tabel 2.1 Spesifikasi Arduino Uno Rev3.....	11
Tabel 2.2 ESP8266 <i>pinout</i>	20
Tabel 2.3 <i>node</i> dan <i>path</i> dalam <i>class diagram</i>	29
Tabel 2.4 <i>node</i> dan <i>edge class diagram</i>	30
Tabel 2.5 <i>VisibilityKind class diagram</i>	31
Tabel 2.6 <i>Node</i> dan <i>edge deployment diagram</i>	32
Tabel 2.7 Simbol-simbol <i>flowchart</i>	33
Tabel 3.1 Pemetaan kelas dan nilai numeriknya.....	43
Tabel 3.2 Deskripsi aktor.....	50
Tabel 3.3 Deskripsi <i>use case Training and clasification</i>	50
Tabel 3.4 Deskripsi <i>use case View about</i>	51
Tabel 3.5 Deskripsi <i>use case Adjust forward</i>	51
Tabel 3.6 Deskripsi <i>use case Adjust right</i>	52
Tabel 3.7 Deskripsi <i>use case Adjust left</i>	52
Tabel 3.8 Deskripsi <i>use case Train data</i>	53
Tabel 3.9 Deskripsi <i>use case Clear data set</i>	54
Tabel 3.10 Deskripsi <i>use case Connect</i>	54
Tabel 3.11 Deskripsi <i>use case Dissconnect</i>	55
Tabel 3.12 Deskripsi <i>use case Change wheel speed</i>	56
Tabel 3.13 Deskripsi <i>class diagram</i>	57
Tabel 3.14 Deskripsi <i>deployment diagram</i>	58
Tabel 4.1 Data masukan dengan kelas targetnya.....	64
Tabel 4.2 Data pelatihan.....	76
Tabel 4.3 Data pelatihan setelah normalisasi.....	77
Tabel 4.4 Parameter-parameter pelatihan.....	77
Tabel 4.5 Data pengujian.....	78
Tabel 4.6 Data pengujian setelah normalisasi.....	79

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Arduino Uno Rev3.....	11
Gambar 2.2 Neurosky <i>MindWave Mobile 2 EEG Headset</i>	14
Gambar 2.3 ESP8266-01	19
Gambar 2.4 ESP8266 <i>pinout</i>	20
Gambar 2.5 L298N	21
Gambar 2.6 Logo Android	22
Gambar 2.7 Arsitektur JST dengan LVQ	23
Gambar 2. 8 Antarmuka pengguna Android Studio	25
Gambar 2. 9 Antarmuka pengguna Arduino IDE.....	26
Gambar 2.10 Antarmuka pengguna Fritzing	27
Gambar 2.11 Logo blender	28
Gambar 2.12 Disabilitas pada Anak Umur 5-17 Tahun menurut Provinsi Tahun 2018.....	34
Gambar 2.13 Proporsi Disabilitas Dewasa umur 18-59 Tahun menurut Provinsi di Indonesia Tahun 2018	35
Gambar 2.14 Proporsi Disabilitas Lansia (umur ≥ 60 tahun) Menurut Provinsi di Indonesia Tahun 2018	36
Gambar 2.15 Kerangka pemikiran.....	40
Gambar 3.1 Desain penelitian	41
Gambar 3.2 Blok diagram perancangan perangkat keras	45
Gambar 3.3 Perancangan mekanik	46
Gambar 3.4 Blok diagram perancangan eletrik	47
Gambar 3.5 Rangkaian penggunaan pin.....	48
Gambar 3.6 Rangkaian penggunaan pin untuk mengunggah program ke ESP8266	48
Gambar 3.7 <i>Use case diagram</i>	50
Gambar 3.8 <i>Class diagram</i>	57
Gambar 3.9 <i>Deployment diagram</i>	58
Gambar 3.10 <i>Flowchart diagram Access Point dan Web server</i> untuk program yang di akan diunggah ke ESP8266.....	59
Gambar 3.11 <i>Flowchart diagram</i> untuk program yang di unggah ke Arduino... 60	60
Gambar 3.12 Perancangan antarmuka pengguna pemilihan menu.....	61
Gambar 3.13 Perancangan antarmuka pengguna <i>MindWave</i>	62
Gambar 3.14 Perancangan antarmuka pengguna <i>About</i>	63
Gambar 4.1 Hasil perancangan mekanik tampak depan.....	66
Gambar 4.2 Hasil perancangan mekanik tampak belakang.....	67
Gambar 4.3 Hasil perancangan mekanik tampak samping.....	67
Gambar 4.4 Hasil perancangan mekanik dan elektrik	68
Gambar 4.5 Hasil perancangan eletrik	68

Gambar 4.6 Hasil <i>screenshot</i> antarmuka pengguna pemilihan menu	69
Gambar 4.7 Hasil <i>screenshot</i> antarmuka pengguna <i>MindWave</i>	70
Gambar 4.8 Hasil <i>screenshot</i> bagian bawah antarmuka pengguna <i>MindWave</i> ...	71
Gambar 4.9 Hasil <i>screenshot</i> antarmuka pengguna <i>About</i>	72
Gambar 4.10 Hasil pengujian unit untuk kelas <i>Lvq1</i>	73
Gambar 4.11 Hasil pengujian antarmuka pengguna <i>MainActivity</i>	73
Gambar 4.12 Hasil pengujian antarmuka pengguna <i>MindwaveFragmentTest</i>	74
Gambar 4.13 Hasil pengujian antarmuka pengguna <i>AboutFragment</i>	74

DAFTAR RUMUS

	Halaman
Rumus 3.1 Normalisasi data	43

BAB I

PENDAHULUAN

1.1. Latar Belakang

Sekitar 15% dari populasi di dunia merupakan Penyandang Disabilitas, bahkan 2-4% di antaranya adalah Penyandang Disabilitas berat. Prevalensi disabilitas global lebih tinggi dari perkiraan *World Health Organization* (WHO) sebelumnya, yaitu pada tahun 1970-an dengan perkiraan yang ditetapkan WHO saat itu sebesar 10% (World Health Organization, 2011). Disabilitas merupakan keadaan yang membatasi kemampuan mental dan atau fisik seseorang (“Kamus Besar Bahasa Indonesia (KBBI) Daring,” n.d.). Disabilitas meliputi gangguan (masalah pada fungsi tubuh), keterbatasan aktivitas, dan pembatasan partisipasi. Penyandang Disabilitas adalah orang yang mempunyai kelainan fisik dan atau mental, yang dapat mengganggu atau merupakan rintangan dan hambatan baginya untuk melakukan secara selayaknya, yang terdiri dari penyandang cacat fisik, penyandang cacat mental serta penyandang cacat fisik dan mental (“Undang-Undang Republik Indonesia Nomor 8 Tahun 2016 Tentang Penyandang Disabilitas,” 2016). Dari hasil penelitian yang dilakukan oleh Infodatin Kemenkes RI yang membahas tentang Situasi Disabilitas di Indonesia, terdapat anak umur 5-17 tahun yang mengalami disabilitas di Indonesia adalah sebesar 3,3%. Proporsi disabilitas pada umur 18-59 tahun di Indonesia adalah sebesar 22,0%, sebanyak 74,3% lansia (umur ≥ 60 tahun) dapat beraktifitas sehari-hari secara mandiri, 22,0% mengalami ketergantungan ringan, 1,1% mengalami ketergantungan sedang, 1%

mengalami ketergantungan berat, dan 1,6% mengalami ketergantungan total (“Disabilitas,” 2019).

Recommendation 2 untuk menghindari hambatan yang dihadapi Penyandang Disabilitas dalam *World Report On Disability* menyatakan berinvestasi dalam program dan layanan khusus untuk penyandang disabilitas, dengan mengadakan program atau layanan-layanan yang dapat membantu para penyandang disabilitas baik dalam pemulihan maupun dalam melakukan aktivitasnya. Selain layanan umum, beberapa penyandang disabilitas mungkin memerlukan akses ke langkah-langkah tertentu, seperti rehabilitasi, layanan dukungan, atau pelatihan. Untuk rehabilitasi termasuk teknologi bantu seperti kursi roda, alat bantu dengar, dan tongkat (World Health Organization, 2011).

Dengan perkembangan teknologi yang sangat pesat, baik teknologi perangkat keras dan perangkat lunak sudah banyak dikembangkan seperti sensor, mikrokontroler (*microcontroller*) bahkan sistem operasi untuk ponsel dan tablet. Sensor merupakan sesuatu yang digunakan untuk mengukur jenis kuantitas fisik atau kimia tertentu, seperti gaya yang bekerja pada perangkat, cahaya yang jatuh di permukaan, atau suhu di suatu ruangan (Nagpal, 2016). Salah satu sensor yaitu sensor *electroencephalography* (EEG) merupakan sensor yang dapat digunakan untuk mendeteksi perubahan medan listrik dari neuron otak melalui kulit kepala. Sensor EEG dirancang untuk mendeteksi gelombang otak di berbagai posisi kulit kepala manusia (Yoon, Kim, Han, Han, & Preda, 2015). *MindWave Mobile 2* merupakan perangkat yang dengan aman dapat mengukur dan mengeluarkan spektrum daya EEG (gelombang alfa, gelombang beta, dll), *NeuroSky eSense meter*

(*Attention* dan *Meditation*) dan kedipan mata. Mikrokontroler adalah komputer kecil yang terkandung dalam satu sirkuit terpadu atau *chip* komputer (Nussey, 2018). Salah satu mikrokontroler yang cukup dikenal yaitu Arduino. Nussey mendefinisikan Arduino sebagai papan sirkuit cetak (*Printed Circuit Board*) yang dirancang untuk menggunakan *chip* mikrokontroler serta input dan *output* lainnya (Nussey, 2018). Sedangkan menurut Dukish, Arduino merupakan platform elektronik sumber terbuka yang didasarkan pada perangkat keras dan lunak yang mudah digunakan (Dukish, 2018). Android adalah sistem operasi berbasis Linux yang mengandalkan Java untuk menyederhanakan pengembangan perangkat lunak dan menggunakan *Extensible Markup Language* (XML) untuk menyederhanakan tugas pengkodean (Rollins & Sandberg, 2013). Ponsel dengan sistem operasi android sudah banyak bermunculan di Indonesia (Umbara & Sulianta, 2013).

Tidak hanya teknologi saja, bidang lain juga mengalami kemajuan yaitu ilmu pengetahuan. Telah dilakukan penelitian-penelitian yang menerapkan ilmu pengetahuan khususnya bidang ilmu kecerdasan buatan dalam kehidupan sehari-hari. Kaplan & Haenlein mendefinisikan kecerdasan buatan sebagai kemampuan sistem untuk menafsirkan data eksternal dengan benar, untuk belajar dari data tersebut, dan menggunakan pembelajaran tersebut untuk mencapai tujuan dan tugas tertentu melalui adaptasi yang fleksibel (Haenlein & Kaplan, 2019). Jaringan Saraf Tiruan atau *Artificial Neural Network* (JST/ANN) yang merupakan bagian dari kecerdasan buatan menurut Bell pada dasarnya dimodelkan pada arsitektur paralel otak hewan, yang belum tentu manusia (Bell, 2015). Jaringan didasarkan pada bentuk input dan *output* yang sederhana. Salah satu metode atau algoritma JST yaitu

Learning Vector Quantization (LVQ). LVQ adalah suatu metode untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi (Kusumadewi, 2003). LVQ merupakan pengembangan dari metode kuantisasi vektor dimana setiap kelas sampel input vektor diwakili oleh set vektor *codebook* sendiri. Satu-satunya tujuan LVQ adalah untuk menggambarkan batas kelas dengan aturan tetangga terdekat (Kohonen, 1995). Siswoyo, Sulistijono, & Arief menggunakan JST dengan algoritma *backpropagation* untuk memprediksi arah kursi roda dengan masukan gelombang otak untuk disabilitas dengan gangguan mobilitas fisik. Hasil penelitian tersebut mendapatkan *Mean square Error* (MSE) pada akhir pelatihan sebesar 0,92495 pada *epoch* 9958, nilai koefisien korelasi sebesar 0,92804 menunjukkan bahwa akurasi hasil proses pelatihan adalah baik (Siswoyo, Sulistijono, & Arief, 2017). Kusumaningtyas & Asmara menggunakan JST dengan metode *perceptron* dan pengolahan citra digital untuk mengidentifikasi kematangan buah tomat dengan tingkat keberhasilan identifikasi kematangan buah tomat sebesar 43,33% (Kusumaningtyas & Asmara, 2016). Indrawan menggunakan JST dengan algoritma LVQ yang dapat mengidentifikasi jenis tanaman obat (Indrawan, 2010). Qur'ani & Rosmalinda menggunakan JST dengan metode LVQ yang dapat mengenali pola tanda tangan dengan ketepatan 98% pada data yang di uji (Qur'ani & Rosmalinda, 2010). Afriandi & Sutikno menggunakan JST dengan algoritma LVQ untuk identifikasi telapak tangan dan mendapatkan hasil dengan tingkat akurasi sebesar 74,66% (Afriandi & Sutikno, 2016).

Untuk dapat meningkatkan kualitas hidup Penyandang Disabilitas dengan gangguan mobilitas yang juga sesuai dengan *Recommendation 2* untuk menghindari

hambatan Penyandang Disabilitas (World Health Organization, 2011), maka pada penelitian ini akan dibangun alat bantu rehabilitasi yang berupa sebuah prototipe kursi roda yang dapat digerakkan dengan pikiran yang nantinya dapat digunakan oleh Penyandang Disabilitas, khususnya Penyandang Disabilitas dengan gangguan mobilitas (sulit dalam berpindah tempat). Didukung dari penelitian yang pernah dilakukan sebelumnya yang menerapkan ilmu pengetahuan dalam kehidupan sehari-hari, dalam penelitian ini akan digunakan JST dengan algoritma LVQ untuk menentukan arah dari kursi roda. Proses ini akan di jalankan di ponsel Android. Keluaran dari proses ini akan dikirim ke Arduino untuk menentukan arah dari kursi roda. Masukan yang diterima oleh ponsel Android berupa nilai keadaan pikiran yang diterima dari *MindWave Mobile 2*. Berdasarkan latar belakang dan penjelasan tersebut tersebut, maka penelitian ini diangkat dengan judul "PENERAPAN JARINGAN SARAF TIRUAN MENGGUNAKAN ALGORITMA *LEARNING VECTOR QUANTIZATION* UNTUK KLASIFIKASI ARAH KURSI RODA"..

1.2. Identifikasi Masalah

Berdasarkan latar belakang masalah diatas, maka permasalahan dalam penelitian ini dapat diidentifikasi sebagai berikut:

1. Meningkatnya populasi Penyandang Disabilitas global.
2. Keterbatasan dan ketergantungan Penyandang Disabilitas dalam melakukan aktifitas.
3. Penerapan alat bantu rehabilitasi mobilitas untuk Penyandang Disabilitas.

1.3. Pembatasan Masalah

Agar pembahasan penelitian tetap terfokus dan tidak menyimpang, maka dapat ditentukan batasan masalah dalam penelitian ini. Berikut batasan masalah dalam penelitian ini:

1. Menggunakan JST dengan algoritma LVQ1.
2. Data masukan untuk pelatihan dan pengujian berupa nilai *Attention* dan *Meditation* yang dihasilkan oleh *MindWave Mobile 2*.
3. Terdapat tiga arah yang dapat diklasifikasi yaitu maju, kanan, dan kiri.
4. Kursi roda berupa prototipe.

1.4. Rumusan Masalah

Adapun perumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana penggunaan JST dengan algoritma LVQ untuk klasifikasi arah kursi roda?
2. Bagaimana merancang dan membangun sistem yang dapat mengendalikan kursi roda dan menentukan arah kursi roda menggunakan JST dengan algoritma LVQ berdasarkan nilai keadaan pikiran?

1.5. Tujuan Penelitian

Tujuan dari penelitian ini diantaranya sebagai berikut:

1. Menguji JST dengan algoritma LVQ untuk klasifikasi arah kursi roda.

2. Merancang dan membangun sistem yang dapat mengendalikan kursi roda dan menentukan arah kursi roda menggunakan JST dengan algoritma LVQ berdasarkan nilai keadaan pikiran.

1.6. Manfaat

Penelitian ini diharapkan memberikan manfaat, baik yang bersifat teoritis ataupun praktis .

1.6.1. Manfaat Teoritis

Adapun manfaat teoritis dari penelitian ini antara lain:

1. Sebagai referensi atau bahan rujukan untuk penelitian selanjutnya.
2. Menambah wawasan pembaca tentang JST dengan algoritma LVQ.
3. Informasi yang diperoleh dari penelitian ini dapat dimanfaatkan oleh penyedia produk sebagai pertimbangan pada penerapan atau pembuatan produk akhir.

1.6.2. Manfaat Praktis

Adapun manfaat praktis dari penelitian ini adalah sistem yang dibangun dapat digunakan oleh Penyandang Disabilitas untuk mengendalikan kursi roda berdasarkan kondisi pikiran.

BAB II KAJIAN TEORI

2.1. Teori Dasar

2.1.1. Kecerdasan Buatan

Kecerdasan buatan (*Artificial intelligence*) merupakan ilmu yang mempelajari tentang bagaimana komputer melakukan penalaran berdasarkan pengetahuan atau pengalaman dan menarik kesimpulan sesuai dengan masukan yang diberikan. Kusumadewi mengemukakan bahwa kecerdasan buatan sebagai salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia (Kusumadewi, 2003). Menurut Nagy, Kecerdasan buatan adalah ilmu yang digunakan untuk membangun kecerdasan menggunakan solusi perangkat keras dan perangkat lunak (Nagy, 2018). Sedangkan Haenlein & Kaplan mendefinisikan kecerdasan buatan sebagai kemampuan sistem untuk menafsirkan data eksternal dengan benar, belajar dari data tersebut, dan menggunakan pembelajaran tersebut untuk mencapai tujuan dan tugas tertentu melalui adaptasi yang fleksibel (Haenlein & Kaplan, 2019).

2.1.2. Jaringan Saraf Tiruan

Kusumadewi dalam bukunya mendefinisikan Jaringan Saraf Tiruan (JST) sebagai salah satu representasi buatan dari otak manusia yang selalu mencoba menstimulasi proses pembelajaran pada otak manusia tersebut. JST dapat digambarkan sebagai model matematis dan komputasi untuk fungsi aproksimasi non-linear, klasifikasi data *cluster* dan regresi non-parametrik atau sebuah simulasi dari koleksi model jaringan saraf biologis (Kusumadewi, 2003).

Menurut Bell JST pada dasarnya dimodelkan pada arsitektur paralel otak hewan, yang belum tentu manusia. Jaringan didasarkan pada bentuk input dan *output* yang sederhana (Bell, 2015). Sedangkan Gurney mendefinisikan JST sebagai kumpulan elemen, unit atau *node* pemrosesan yang saling berhubungan, yang fungsinya didasarkan pada neuron hewan. Kemampuan pemrosesan jaringan disimpan dalam kekuatan koneksi interunit, atau bobot, yang diperoleh dengan proses adaptasi atau belajar dari serangkaian pola pelatihan (Gurney, 2004).

2.2. Teori Khusus

2.2.1. Disabilitas

Disabilitas merupakan keadaan yang membatasi kemampuan mental dan atau fisik seseorang (“Kamus Besar Bahasa Indonesia (KBBI) Daring,” n.d.). Disabilitas meliputi meliputi keterbatasan aktivitas, gangguan (masalah pada fungsi tubuh), dan pembatasan partisipasi. Penyandang Disabilitas adalah setiap orang yang

mengalami keterbatasan fisik, intelektual, mental, dan atau sensorik dalam jangka waktu lama yang dalam berinteraksi dengan lingkungan dapat mengalami hambatan dan kesulitan untuk berpartisipasi secara penuh dan efektif dengan warga negara lainnya berdasarkan kesamaan hak (“Undang-Undang Republik Indonesia Nomor 8 Tahun 2016 Tentang Penyandang Disabilitas,” 2016) .

Ragam Penyandang Disabilitas sesuai dengan yang tercantum dalam UU No. 8 Tahun 2016 diantaranya Penyandang Disabilitas fisik, Penyandang Disabilitas intelektual, Penyandang Disabilitas mental, Penyandang Disabilitas sensorik. Penyandang Disabilitas fisik yaitu terganggunya fungsi gerak, antara lain amputasi, lumpuh layuh atau kaku, *paraplegi*, *celebral palsy* (CP), akibat stroke, akibat kusta, dan orang kecil. Penyandang Disabilitas intelektual yaitu terganggunya fungsi pikir karena tingkat kecerdasan di bawah rata-rata, antara lain lambat belajar, disabilitas grahita dan *down syndrom*. Penyandang Disabilitas mental yaitu terganggunya fungsi pikir, emosi, dan perilaku, antara lain: psikososial di antaranya *skizofrenia*, bipolar, depresi, *anxietas*, dan gangguan kepribadian; dan disabilitas perkembangan yang berpengaruh pada kemampuan interaksi sosial di antaranya autisme dan hiperaktif. Penyandang Disabilitas sensorik yaitu terganggunya salah satu fungsi dari panca indera, antara lain disabilitas netra, disabilitas runtu, dan atau disabilitas wicara.

2.2.2. Arduino Uno Rev3

Arduino adalah papan tunggal mikrokontroler dengan perangkat keras, yang terdiri dari papan perangkat keras sumber terbuka (*open source*) yang dirancang

dengan Atmel AVR Mikrokontroler (Schwartz & Manickum, 2015). Pajankar mengemukakan bahwa Arduino adalah platform dan ekosistem prototipe elektronik sumber terbuka, yang didasarkan pada lingkungan perangkat keras dan perangkat lunak yang mudah digunakan (Pajankar, 2018).



Gambar 2.1 Arduino Uno Rev3
(Sumber : <http://bit.ly/2wjVbvd>)

Uno adalah bahasa Italia untuk nomor satu, Uno digunakan sebagai nama untuk rilis versi 1.0 dari perangkat lunak Arduino. R3 yang diartikan sebagai revisi ketiga, berkaitan dengan revisi fitur di papan Arduino itu sendiri, yang mencakup pembaruan, penyempurnaan, dan perbaikan.

Tabel 2.1 di menunjukkan ringkasan spesifikasi dari Arduino Uno Rev 3.

Tabel 2.1 Spesifikasi Arduino Uno Rev3

Spesifikasi	Nilai
Mikrokontroler	ATmega328P
Tegangan operasi	5V
Tegangan input (direkomendasikan)	7-12V
Tegangan input (batas)	6-20V
Pin I/O digital	14 (6 pin dapat digunakan sebagai <i>output</i> PWM)
PWM digital I/O pin	6

Tabel 2.1 (Lanjutan) Spesifikasi Arduino Uno Rev3

Spesifikasi	Nilai
Pin input analog	6
Arus DC per pin I/O	20 mA
Arus DC untuk 3.3V pin	50 mA
Memori <i>flash</i>	32 KB (ATmega328P) 0.5 KB digunakan untuk <i>bootloader</i>
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Kecepatan <i>clock</i>	16 MHz
LED_BUILTIN	13
Panjang	68.6 mm
Lebar	53.4 mm
Berat	25 g

(Sumber : <http://bit.ly/2wjVbvd>)

Berikut merupakan bagian-bagian dari Arduino Uno Rev3.

1. *Power*

Papan Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Daya eksternal dapat berasal dari adaptor AC ke DC atau baterai. Papan Arduino Uno dapat beroperasi dengan suplai eksternal dari 6 hingga 20V. Namun, jika diberikan kurang dari 7V, pin 5V dapat memasok kurang dari 5V dan papan mungkin menjadi tidak stabil. Jika menggunakan lebih dari 12V, pengatur tegangan bisa menjadi terlalu panas dan merusak papan. Kisaran yang disarankan adalah 7V hingga 12V. Pin *power* diantaranya adalah pin Vin yang digunakan untuk tegangan input ke papan Arduino ketika menggunakan sumber daya eksternal, pin 5V menghasilkan tegangan 5V, pin 3.3V menghasilkan tegangan 3,3V, pin GND digunakan sebagai *ground*, dan pin IOREF memberikan referensi tegangan yang digunakan mikrokontroler.

2. Memori

ATmega328 memiliki memori 32 KB (dengan 0,5 KB digunakan oleh *bootloader*). Selain itu juga memiliki 2 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis dengan *library* EEPROM).

3. Input dan *output*

Masing-masing dari 14 pin pada Arduino Uno dapat digunakan sebagai input atau *output*. Beberapa pin memiliki fungsi khusus. Pin 3, 5, 6, 9, 10, dan 11 memiliki fungsi khusus yaitu menyediakan *output* PWM 8-bit. Pin 0 (RX) dan pin 1 (TX) digunakan untuk menerima (RX) dan mengirimkan (TX) data serial *Transistor-Transistor Logic* (TTL).

2.2.3. MindWave Mobile 2

MindWave Mobile 2 merupakan perangkat yang dengan aman dapat mengukur dan mengeluarkan spektrum daya EEG (gelombang alfa, gelombang beta, dll), NeuroSky *eSense meter* (*Attention* dan *Meditation*) dan kedipan mata ("Mindwave," n.d.). Perangkat terdiri dari *headset*, klip telinga, dan lengan sensor. Referensi *headset* dan *ground* elektroda ada di klip telinga dan elektroda EEG ada di lengan sensor, ditempatkan di dahi di atas mata. Perangkat ini menggunakan baterai AAA tunggal dengan masa pakai baterai selama 8 jam.



Gambar 2.2 Neurosky *MindWave Mobile 2* EEG Headset
(Sumber : <http://bit.ly/2SBd1m8>)

MindWave Mobile 2 dirancang untuk aplikasi pendidikan dan hiburan dengan mengubah perubahan kondisi mental menjadi tindakan. *MindWave Mobile 2* melaporkan kondisi mental pengguna dalam bentuk algoritma kepemilikan NeuroSky, termasuk algoritma *Attention* dan *Meditation eSense*. NeuroSky *MindWave Mobile 2* dapat digunakan dengan video game yang didukung, perangkat lunak pendidikan, dan banyak aplikasi lainnya (*MindWave Mobile 2: User Guide*, 2018).

Electroencephalogram (EEG) adalah tes yang merekam aktivitas listrik di otak menggunakan cakram logam kecil (elektroda) yang ditempelkan di kulit kepala (“EEG (electroencephalogram),” n.d.). Sel-sel otak yang juga dikenal sebagai neuron, saling berkomunikasi dengan memancarkan impuls listrik dan aktif setiap saat bahkan ketika sedang tidur, mengelola cara berpikir dan cara berinteraksi. Aktivitas ini ditampilkan sebagai garis bergelombang pada rekaman EEG. Hasil EEG menunjukkan perubahan aktivitas otak yang mungkin berguna

dalam mendiagnosis kondisi otak (“Reading Your Brainwaves: Understanding the Basics of EEG,” n.d.).

Beberapa jenis gelombang otak yang umum adalah sebagai berikut (Demos, 2019):

1. Gelombang *delta*

Gelombang *delta* dikaitkan dengan tidur, tidur tanpa mimpi, tidur *Non-Rapid Eye Movement* (N-REM), dan tidak sadar. Gelombang *delta* dominan muncul pada bayi. Gelombang *delta* yang terlalu tinggi mengindikasikan ketidakmampuan untuk meremajakan tubuh, ketidakmampuan untuk merevitalisasi otak, atau kurang tidur. Gelombang *delta* yang terlalu rendah mengindikasikan cedera otak atau *traumatic brain injury* (TBI), masalah belajar, ketidakmampuan untuk berpikir, *attention deficit hyperactivity disorder* (ADHD) berat.

2. Gelombang *theta*

Gelombang *theta* dikaitkan dengan kreativitas, koneksi emosional, intuisi, relaksasi. Anak-anak memiliki amplitudo *theta* yang lebih tinggi daripada orang dewasa. Gelombang *theta* yang terlalu tinggi mengindikasikan ADHD, depresi, hiperaktif, impulsif, dan kurang perhatian. Gelombang *theta* yang terlalu rendah mengindikasikan kecemasan, kesadaran emosional yang buruk, dan stres.

3. Gelombang *alpha*

Gelombang *alpha* dikaitkan dengan santai, tetapi tidak mengantuk, tenang, sadar. Gelombang *alpha* yang terlalu tinggi mengindikasikan melamun, ketidakmampuan untuk fokus, terlalu santai. Gelombang *alpha* yang terlalu rendah

mengindikasikan kecemasan, stres tinggi, susah tidur, *Obsessive compulsive disorder* (OCD).

4. Gelombang *beta*

Gelombang *beta* dengan nilai yang tepat dikaitkan dengan fokus sadar, ingatan, pemecahan masalah. Gelombang *beta* yang terlalu tinggi mengindikasikan kecemasan yang berlebihan, gairah tinggi, ketidakmampuan untuk rileks, dan stres. Gelombang *beta* yang terlalu rendah mengindikasikan ADHD, melamun, depresi, kognisi buruk.

5. Gelombang *gamma*

Gelombang *gamma* dengan nilai yang tepat dikaitkan dengan pengikatan indera, kognisi, pemrosesan informasi, pembelajaran (mempelajari materi atau hal baru), persepsi, tidur REM. Gelombang *gamma* yang terlalu tinggi mengindikasikan kecemasan, gairah tinggi, stres. Individu yang mengalami gangguan mental dan memiliki ketidakmampuan belajar cenderung memiliki aktivitas *gamma* yang lebih rendah daripada rata-rata. Gelombang *gamma* yang terlalu rendah mengindikasikan ADHD, depresi, dan ketidakmampuan belajar.

2.2.3.1. *ThinkGear*

ThinkGear adalah teknologi di dalam setiap produk atau mitra NeuroSky yang memungkinkan perangkat untuk berinteraksi dengan gelombang otak pemakai. Ini termasuk sensor yang menyentuh dahi, titik kontak dan referensi yang terletak di klip telinga, dan *chip on-board* yang memproses semua data. Baik

gelombang otak mentah atau *raw brainwaves* dan *eSense meter* (*Attention* dan *Meditasi*) dihitung pada *chip ThinkGear*.

2.2.3.2. eSense

eSense adalah algoritma kepemilikan NeuroSky untuk mengkarakterisasi keadaan mental. Untuk menghitung *eSense*, teknologi Neurosky *Thinkgear* memperkuat sinyal *raw brainwave* dan menghilangkan kebisingan sekitar dan gerakan otot. Algoritma *eSense* kemudian diterapkan pada sinyal yang tersisa, menghasilkan nilai meteran *eSense meter* yang dapat diinterpretasikan. Nilai-nilai *eSense meter* tidak menggambarkan angka yang pasti, melainkan menggambarkan rentang aktivitas (*MindWave Mobile 2: User Guide*, 2018).

eSense meter adalah cara untuk menunjukkan seberapa efektif pengguna melakukan perhatian (mirip dengan konsentrasi) atau meditasi (mirip dengan relaksasi). Untuk setiap jenis *eSense* (*Attention*, dan *Meditation*), nilai meter dilaporkan pada skala *eSense* relatif dari 1 hingga 100. Pada skala ini, nilai antara 40 hingga 60 pada saat tertentu dianggap "netral" dan serupa dengan "*baselines*" yang ditetapkan dalam teknik pengukuran gelombang otak konvensional (meskipun metode untuk menentukan *ThinkGear baselines* adalah hak milik dan mungkin berbeda dari metode lain). Nilai dari 60 hingga 80 dianggap "sedikit lebih tinggi", dan dapat ditafsirkan sebagai tingkat yang cenderung lebih tinggi dari normal (tingkat *Attention* atau *Meditation* yang mungkin lebih tinggi dari normal untuk orang tertentu). Nilai dari 80 hingga 100 dianggap "tinggi". Demikian pula, nilai antara 20 hingga 40 menunjukkan tingkat "rendah" dari *eSense*, sementara nilai

antara 1 hingga 20 menunjukkan tingkat "sangat rendah" tingkat *eSense*. Tingkat dapat menunjukkan keadaan gangguan, agitasi, atau kelainan, sesuai dengan kebalikan dari masing-masing *eSense* (*MindWave Mobile 2: User Guide*, 2018).

2.2.3.3. Attention eSense

Indikator *Attention eSense* menunjukkan intensitas tingkat "fokus" atau "perhatian" pengguna, seperti yang terjadi selama konsentrasi intens dan aktivitas mental terarah (tetapi stabil). Nilainya berkisar dari 0 hingga 100. Gangguan, pikiran yang berkeliaran, kurang fokus, atau kecemasan dapat menurunkan tingkat meteran *Attention* (*MindWave Mobile 2: User Guide*, 2018).

2.2.3.4. Meditation eSense

Indikator *Meditation eSense* menunjukkan tingkat "ketenangan" atau "relaksasi" mental pengguna. Nilainya berkisar dari 0 hingga 100. Nilai *Meditation* meningkat ketika ketika pengguna merilekskan pikiran dan berkurang ketika gelisah atau stres. Meditasi adalah ukuran keadaan mental seseorang, bukan tingkat fisik, jadi sekadar merelaksasikan semua otot tubuh mungkin tidak akan segera menghasilkan tingkat Meditasi yang tinggi. Namun, bagi kebanyakan orang dalam kebanyakan situasi normal, rileks tubuh sering membantu pikiran untuk rileks juga. Meditasi berhubungan dengan berkurangnya aktivitas oleh proses mental aktif di otak. Menutup mata dapat mematikan aktivitas mental yang memproses gambar dari mata. Jadi menutup mata sering kali merupakan metode yang efektif untuk

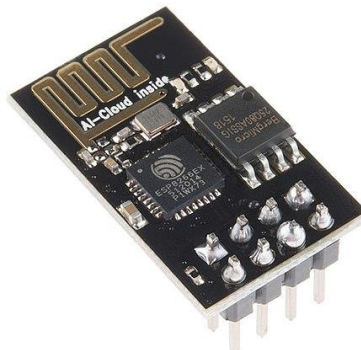
meningkatkan level meteran *Meditation*. Gangguan, pikiran yang berkeliaran, kecemasan, agitasi, dan rangsangan sensorik dapat menurunkan tingkat meteran *Meditation* (*MindWave Mobile 2: User Guide*, 2018).

2.2.3.5. *Blink Detection*

Blink Detection menunjukkan tingkat kedipan mata pengguna. Nilai yang lebih tinggi menunjukkan kedipan "kuat", sedangkan nilai kecil menunjukkan kedipan "lebih ringan" atau "lebih lemah". Nilai ini berkisar dari 0 hingga 255 ("EEG Algorithms," n.d.).

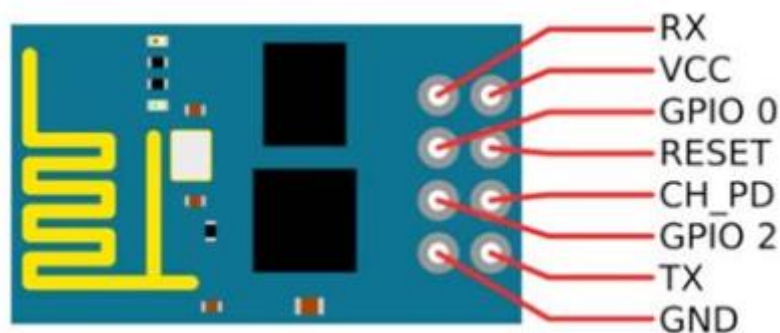
2.2.4. ESP8266

Pada dasarnya, ESP8266 merupakan modul Wifi (Towaha, 2018). ESP8266 memiliki dukungan untuk 802.11 b/g/n dan beroperasi pada frekuensi 2,4 GHz. Modul ini mendukung *open networks*, WPA dan WPA2 dan dapat digunakan sebagai *Station* atau *Access Point* atau gabungan keduanya.



Gambar 2.3 ESP8266-01
(Sumber : <http://bit.ly/39NcfZg>)

ESP8266 biasa digunakan untuk pembuatan IoT. Perangkat ini terintegrasi dengan prosesor 32-bit yang berjalan pada kecepatan 80 MHz (dapat di-*overclock* hingga 160 MHz).



Gambar 2.4 ESP8266 *pinout*
Sumber: (Towaha, 2018)

Tabel 2.2 berikut berisi keterangan *pinout* untuk ESP8266.

Tabel 2.2 ESP8266 *pinout*

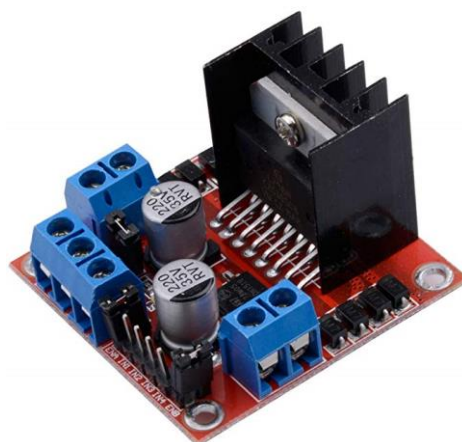
Nomor	Nama	Nama alternatif	Deskripsi/Kegunaan
1	GND	-	Sebagai <i>ground</i>
2	TX	GPIO-1	Mentransmisikan data
3	GPIO 2	-	<i>Input/output</i> pin untuk tujuan umum
4	CH_EN	-	<i>Chip enable</i>
5	GPIO 0	Flash	<i>Input/output</i> pin untuk tujuan umum
6	Reset	-	Mengatur ulang modul.
7	RX	GPIO-3	Menerima data
8	VCC	-	Pin <i>power</i> (3.3V)

(Sumber : (Towaha, 2018))

ESP8266 juga dapat diprogram menggunakan Arduino IDE atau AT-Command dan mendukung komunikasi *serial* sehingga kompatibel dengan banyak platform pengembangan seperti Arduino.

2.2.5. L298N

L298N adalah *motor driver H-Bridge* ganda yang memungkinkan pengendalian arah dan kecepatan sekaligus untuk dua motor DC. (Bi, Kapoor, & Bhatia, 2018). *Motor driver* digunakan untuk mengendalikan motor DC. Alasan utama diperlukannya *motor driver* yaitu karena komponen seperti motor tidak dapat dihubungkan langsung ke Arduino karena tidak akan mendapatkan arus yang cukup dari Arduino. Ketika motor berjalan maka motor akan mengkonsumsi arus tinggi. *Motor driver* memberikan keamanan pada mikrokontroler sehingga aliran tidak akan mengalir ke mikrokontroler.



Gambar 2.5 L298N

(Sumber : <https://amzn.to/3bN4zaW>)

2.2.6. Android

Android pada dasarnya adalah sistem operasi komputasi seluler, misalnya untuk perangkat seluler seperti ponsel pintar dan tablet (Vandome, 2017). Menurut Rollins & Sandberg Android adalah sistem operasi berbasis Linux yang

mengandalkan Java untuk menyederhanakan pengembangan perangkat lunak dan menggunakan *Extensible Markup Language* (XML) untuk menyederhanakan tugas pengkodean (Rollins & Sandberg, 2013). Android adalah sistem operasi *open source*, yang berarti pengembang dan produsen dapat bekerja dengan kode sumber untuk menyesuaikannya dengan kebutuhan dan perangkat mereka sendiri (selama memenuhi persyaratan dan standar tertentu) (Vandome, 2017).



Gambar 2.6 Logo Android
(Sumber : <http://bit.ly/323Hpsu>)

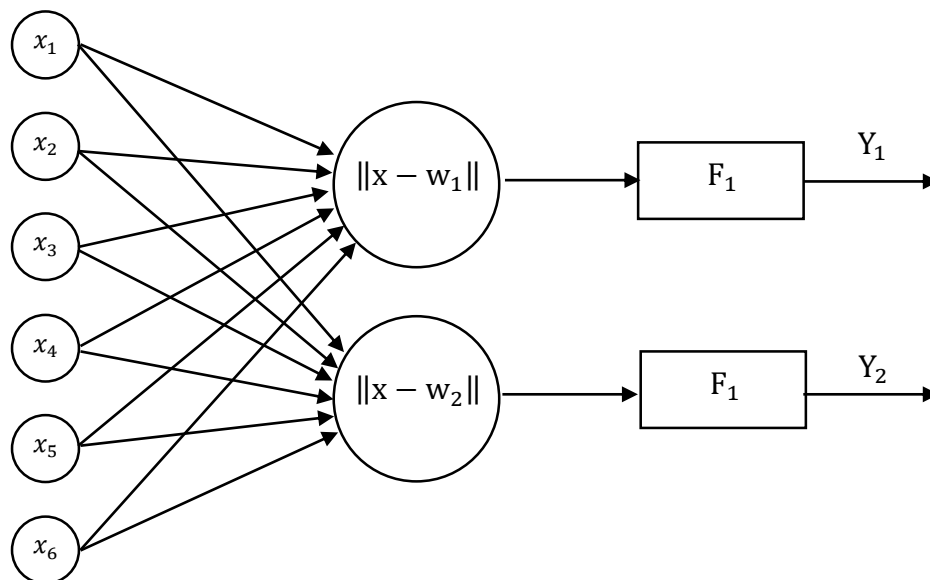
2.3. Model JST

Model JST yang digunakan dalam penelitian ini adalah *Learning Vector Quantization* (LVQ). LVQ adalah suatu metode untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi (Kusumadewi, 2003). LVQ merupakan pengembangan dari metode kuantisasi vektor dimana setiap kelas sampel input vektor diwakili oleh set vektor *codebook*. Satu-satunya tujuan LVQ adalah untuk menggambarkan batas kelas dengan aturan tetangga terdekat (Kohonen, 1995).

Lapisan kompetitif akan belajar untuk mengklasifikasikan vektor-vektor input. Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor-vektor input. Jika 2 vektor input mendekati

sama, maka lapisan kompetitif akan meletakkan kedua vektor input tersebut ke dalam kelas yang sama.

LVQ menggunakan arsitektur jaringan dengan lapisan kompetitif (*competitive layer net*). Pada JST dengan LVQ terdiri dari $x_1..x_n$ input yang terhubung dengan lapisan kompetitif dan menuju lapisan keluaran. Lapisan keluaran merupakan kelas atau target yang ditentukan selama proses pelatihan. Arsitektur JST dengan algoritma LVQ dapat dilihat pada gambar 2.7.



Gambar 2.7 Arsitektur JST dengan LVQ
(Sumber : (Kusumadewi, 2003))

Berikut merupakan algoritma dari LVQ (Kusumadewi, 2003):

1. Tetapkan nilai untuk bobot (w), maksimum *epoch* (MaxEpoch), *error* minimum yang diharapkan (ϵ) dan *learning rate* (α).
2. Masukkan Input : $x(m,n)$ dan Target : $(T1,n)$ dimana m merupakan jumlah input, n merupakan jumlah data.
3. Inialisasi kondisi awal $epoch = 0$, $\epsilon = 1$
4. Lakukan jika: ($epoch < \text{MaxEpoch}$) atau ($\alpha < \epsilon$)

1. $\text{Epoch} = \text{Epoch} + 1$
2. Kerjakan untuk $i = 1$ sampai n
 1. Tentukan J sedemikian hingga $\|x - w_j\|$ minimum (sebagai C_j)
 2. Perbaiki w_j dengan ketentuan:
 1. Jika $T = C_j$ maka: $w_j(\text{baru}) = w_j(\text{lama}) + \alpha (x - w_j(\text{lama}))$
 2. Jika $T \neq C_j$ maka: $w_j(\text{baru}) = w_j(\text{lama}) - \alpha (x - w_j(\text{lama}))$
 3. Kurangi nilai α :

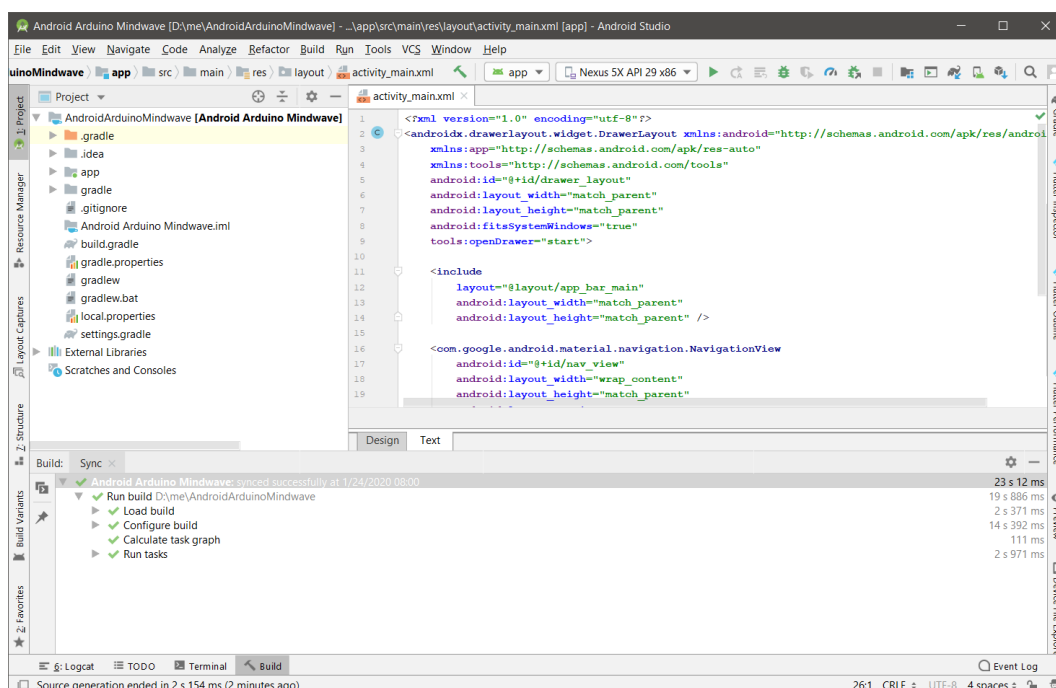
2.4. Software Pendukung dan Perangkat Pemodelan

2.4.1. Java

Java adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer yang berdiri sendiri (*standalone*) ataupun pada lingkungan jaringan. Java berjalan di atas sebuah mesin penerjemah (*interpreter*) yang diberi nama *Java Virtual Machine* (JVM). JVM akan membaca kode bit (*bytecode*) dalam *file* .class dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Oleh karena itu bahasa Java disebut sebagai bahasa pemrograman yang portabel karena dapat dijalankan pada berbagai sistem operasi, asalkan pada sistem operasi tersebut terdapat JVM. Alasan utama pembentukan bahasa Java adalah untuk membuat aplikasi-aplikasi yang dapat diletakkan di berbagai macam perangkat elektronik, sehingga Java harus bersifat tidak bergantung pada platform (*platform independent*) (S & Shalahuddin, 2015).

2.4.2. Android Studio

Android Studio adalah lingkungan pengembangan terintegrasi/*integrated development environment* (IDE), mirip dengan ADT *Bundle* atau *Eclipse* dengan ADT *Plugin*, tetapi didasarkan pada IntelliJ IDEA, yang merupakan alat yang sangat populer untuk pengembangan Java (Jr., Darcey, & Conder, 2014).

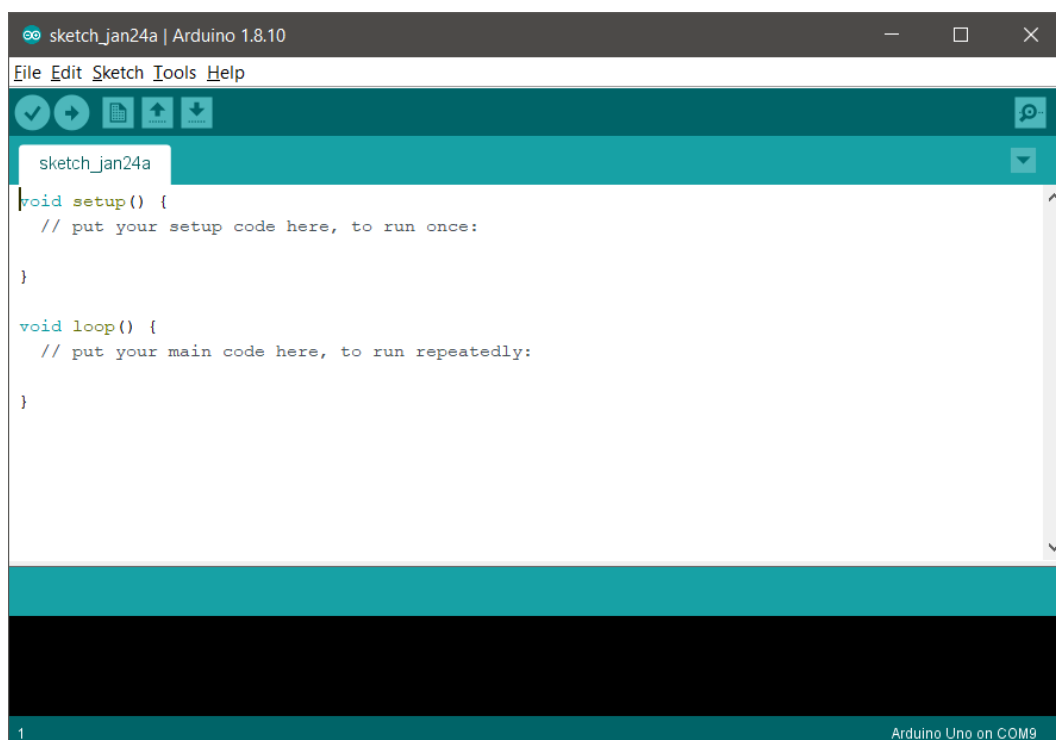


Gambar 2.8 Antarmuka pengguna Android Studio
(Sumber : Data Penelitian, 2020)

Menurut Drongelen Android Studio adalah IDE yang disarankan untuk mengembangkan aplikasi Android dan tersedia secara gratis bagi siapa saja yang mengembangkan aplikasi Android profesional (Drongelen, 2015). Horton mendefinisikan Android Studio sebagai alat yang menyederhanakan pengembangan Android dan menggunakan JDK untuk memungkinkan pengguna menulis dan membangun program Java (Horton, 2015).

2.4.3. Arduino IDE

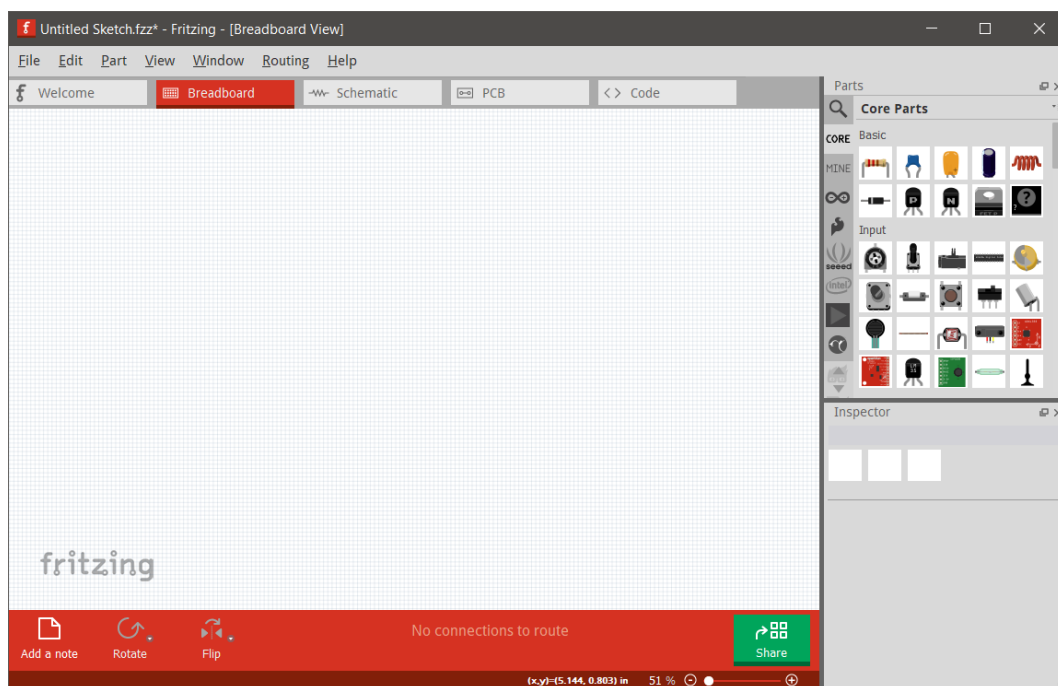
Arduino IDE adalah aplikasi *cross-platform* yang dikembangkan dengan Java yang dapat digunakan untuk mengembangkan, meng-*compile*, dan mengunggah program ke papan Arduino (Desai, 2015).



Gambar 2. 9 Antarmuka pengguna Arduino IDE
(Sumber : Data Penelitian, 2020)

2.4.4. Fritzing

Fritzing adalah program yang memungkinkan kita mengubah desain sirkuit menjadi representasi grafis langsung di komputer. Manfaat utamanya adalah sirkuit yang dirancang dalam *Fritzing* terlihat seperti aslinya. Ini akan memberikan representasi yang intuitif, langsung, dan segera dapat dikenali dari *breadboard* dan komponen yang dihubungkan (Allan & Bradford, 2013).



Gambar 2.10 Antarmuka pengguna *Fritzing*
(Sumber : Data Penelitian, 2020)

2.4.5. *StarUML*

StarUML merupakan alat pemodelan sumber terbuka (Lee, 2008) . *StarUML* mendukung pemodelan UML yang diadopsi sebagai standar oleh *Object Management Group* (OMG). Selain itu *StarUML* juga mendukung pemodelan diagram ER, *flowchart* diagram dan *Data Flow Diagram* (DFD).

2.4.6. *Blender*

Blender adalah perangkat lunak sumber terbuka yang menyediakan paket pembuatan grafik 3D paling lengkap, termasuk alat untuk pemodelan, penteksturan, *shading*, animasi, komposisi, *rendering*, pengeditan video, dan banyak lagi (Villar, 2014). Salah satu hal yang membuat *Blender* berbeda dibandingkan dengan

perangkat lunak 3D lain adalah karena *Blender* tersedia secara gratis dan merupakan perangkat lunak sumber terbuka. Menurut Gumster *Blender* adalah program grafis komputer yang mampu menghasilkan gambar diam dan animasi berkualitas tinggi menggunakan geometri tiga dimensi (Gumster, 2015). *Blender* juga merupakan perangkat lunak yang *cross-platform* dan dapat berjalan dengan baik di Linux, Windows, dan Macintosh.



Gambar 2.11 Logo *blender*
(Sumber : <http://bit.ly/37DhXvh>)

2.4.7. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah notasi dan semantik untuk visualisasi, konstruksi dan dokumentasi model untuk pengembangan perangkat lunak berorientasi objek yang telah distandarisi oleh *Object Management Group* (OMG) (Maier, 2004). UML bukan metodologi atau proses melainkan notasi (Hunt, 2003). UML tidak terikat dengan alat pengembangan spesifik, bahasa pemrograman khusus, atau platform target spesifik tempat sistem yang akan dikembangkan harus digunakan (Seidl et al., 2015).

Dalam UML, model direpresentasikan secara grafis dalam bentuk diagram. Diagram memberikan pandangan tentang bagian realitas yang dijelaskan oleh model. Ada diagram yang menyatakan pengguna mana yang menggunakan

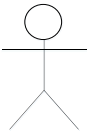
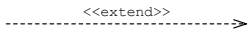
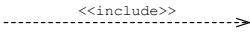
fungsionalitas dan diagram yang menunjukkan struktur sistem tetapi tanpa menentukan implementasi konkret.

Diagram UML yang digunakan dalam penelitian ini diantaranya use case diagram, class diagram, dan deployment diagram. Berikut merupakan ringkasan beberapa diagram uml tersebut.


1. *Use case diagram*

Use case menggambarkan fungsionalitas yang diharapkan dari sistem yang akan dikembangkan tetapi tidak membahas rincian spesifik implementasi (Seidl et al., 2015). *Use case* mencakup sejumlah fungsi yang dieksekusi ketika menggunakan sistem. Unit fungsionalitas yang disediakan sistem untuk penggunaannya disebut *use case*. Tabel 2.3 menunjukkan *node* dan *path* dalam *use case diagram*.

Tabel 2.3 *node* dan *path* dalam *class diagram*

Tipe <i>node/path</i>	Notasi	Deskripsi singkat
<i>Actor (default)</i>		Menentukan peran yang dimainkan oleh pengguna atau sistem lain yang berinteraksi dengan subjek. Aktor dapat mewakili peran yang dimainkan oleh pengguna manusia, perangkat keras eksternal, atau subjek lainnya.
<i>Extend</i>		Hubungan ini menentukan bahwa perilaku <i>use case</i> dapat diperluas oleh perilaku <i>use case</i> lain (biasanya tambahan).
<i>Include</i>		Hubungan <i>include</i> mendefinisikan bahwa <i>use case</i> berisi perilaku yang didefinisikan dalam <i>use case</i> lain.

Tabel 2.3 (Lanjutan) *node* dan *path* dalam *class diagram*

Tipe <i>node/path</i>	Notasi	Deskripsi singkat
<i>UseCase</i>		Spesifikasi dari serangkaian tindakan yang dilakukan oleh suatu sistem, yang menghasilkan hasil yang dapat diamati, yang biasanya, bernilai bagi satu atau lebih aktor atau pemangku kepentingan lain dari sistem.

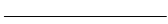



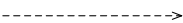
(Sumber : (“OMG Unified Modeling Language (OMG UML), Superstructure Version 2.4.1,” 2011))

2. *Class diagram*



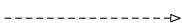
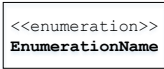
Class diagram digunakan untuk memodelkan struktur statis suatu sistem, sehingga menggambarkan elemen-elemen sistem dan hubungan di antara elemen-elemen tersebut yang tidak berubah seiring waktu (Seidl et al., 2015).

Tabel 2.4 menunjukkan *node* dan *edge* yang biasa digunakan dalam *class diagram*.

Tabel 2.4 *node* dan *edge class diagram*

Tipe <i>node/path</i>	Notasi	Deskripsi singkat
<i>Association</i>		Hubungan semantik yang dapat terjadi di antara instansi.
<i>Aggregation</i>		Bentuk asosiasi khusus yang digunakan untuk menyatakan bahwa instansi dari satu kelas adalah bagian dari instansi dari kelas lain.
<i>Class</i>		Serangkaian objek yang memiliki spesifikasi fitur, batasan, dan semantik yang sama. <i>Class</i> memiliki fitur atribut dan operasi.
<i>Composition</i>		Menunjukkan bahwa properti diagregasi secara komposit.
<i>Dependency</i>		Hubungan yang menandakan bahwa satu atau serangkaian elemen model memerlukan elemen model lain untuk spesifikasi atau implementasinya.

Tabel 2.4 (Lanjutan) *node* dan *edge class diagram*

Tipe node/path	Notasi	Deskripsi singkat
<i>Generalization</i>		Proses mengekstraksi karakteristik bersama dari dua kelas atau lebih, dan menggabungkannya menjadi <i>superclass</i> umum.
<i>Interface</i>		Pengelompokan yang mewakili deklarasi serangkaian fitur dan kewajiban publik yang koheren. Antarmuka menentukan kontrak, setiap instansi dari <i>classifier</i> yang menerapkan antarmuka harus memenuhi kontrak tersebut.
<i>InterfaceRealization</i>		<i>InterfaceRealization</i> adalah hubungan realisasi khusus antara <i>Classifier</i> dan <i>Interface</i> .
<i>Enumeration</i>		Tipe data yang nilainya disebutkan dalam model sebagai literasi enumerasi.

(Sumber : (“OMG Unified Modeling Language (OMG UML), Superstructure Version 2.4.1,” 2011))

Selain notasi diatas dalam dari *class diagram* juga terdapat *VisibilityKind*.

VisibilityKind adalah tipe enumerasi yang mendefinisikan literal untuk menentukan visibilitas elemen dalam suatu model. Tabel 2.5 berikut merupakan ringkasan deskripsi untuk *VisibilityKind*.

Tabel 2.5 *VisibilityKind class diagram*

Nama	Notasi	Deskripsi singkat
<i>public</i>	+	Terlihat oleh semua elemen yang dapat mengakses konten <i>namespace</i> yang memilikinya
<i>private</i>	-	Hanya terlihat di dalam <i>namespace</i> yang memilikinya.
<i>protected</i>	#	Terlihat oleh elemen yang memiliki hubungan generalisasi dengan <i>namespace</i> yang memilikinya
<i>package</i>	~	Terlihat oleh elemen-elemen yang ada dalam paket yang sama dengan <i>namespace</i> yang dimilikinya.


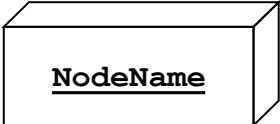
(Sumber : (“OMG Unified Modeling Language (OMG UML), Superstructure Version 2.4.1,” 2011))

3. *Deployment diagram*

Deployment diagram menunjukkan topologi perangkat keras yang digunakan dan sistem runtime. Perangkat keras memuat unit yang membentuk simpul serta hubungan komunikasi antar *node*. Sistem *runtime* berisi artefak atau *artifact* yang digunakan untuk node (Seidl et al., 2015).

Tabel 2.6 berikut menunjukkan *node* dan *edge* yang biasanya digunakan dalam *deployment diagram*:

Tabel 2.6 *Node dan edge deployment diagram*


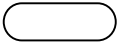
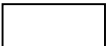
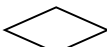


Tipe node/edge	Notasi	Deskripsi singkat
<i>Artifact</i>		Spesifikasi bagian informasi fisik yang digunakan atau diproduksi oleh proses pengembangan perangkat lunak, atau dengan <i>deployment</i> dan pengoperasian sistem.
<i>Node</i>		Node adalah sumber daya komputasi di mana artefak dapat digunakan untuk eksekusi.

(Sumber : (“OMG Unified Modeling Language (OMG UML), Superstructure Version 2.4.1,” 2011))

2.4.8. *Flowchart*

Flowchart adalah representasi grafis yang mewakili langkah-langkah yang spesifik, atau kegiatan dari suatu proses (Fryman, 2001). Tabel 2.7 menunjukkan simbol-simbol *flowchart* yang umum digunakan. *Flowchart* dapat digunakan sebagai komunikasi dan menunjukkan alur suatu algoritma atau proses serta mendokumentasikannya.

Tabel 2.7 Simbol-simbol *flowchart*

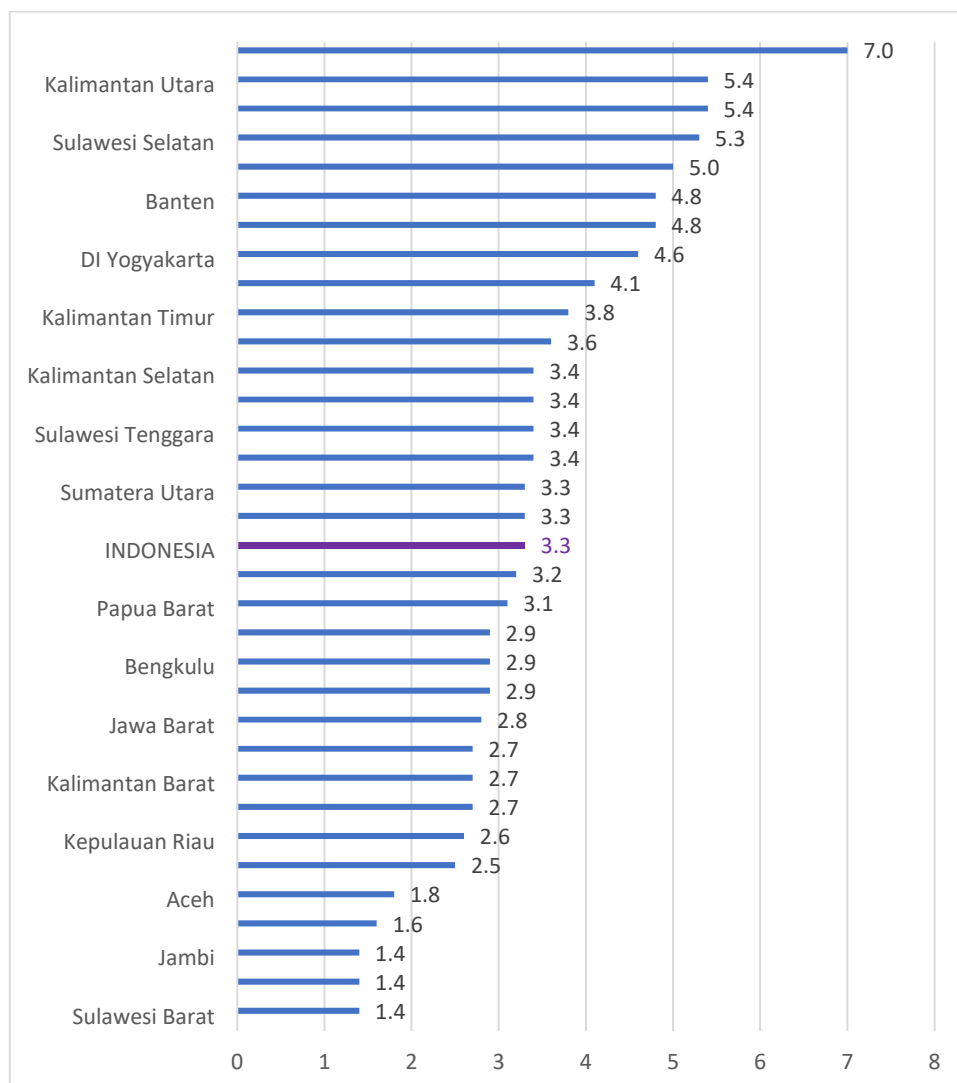
Simbol	Nama	Deskripsi
	<i>Flow</i>	Menunjukkan arah aliran proses.
	<i>Terminator</i>	Merepresentasikan awal dan akhir suatu proses.
	<i>Process</i>	Merepresentasikan langkah dalam proses
	<i>Decision</i>	Merepresentasikan percabangan atau pertanyaan dalam aliran proses
	<i>Predefined Process</i>	Merepresentasikan proses lain atau serangkaian langkah aliran proses yang didefinisikan di tempat lain.
	<i>Delay</i>	Mendeskripsikan masa tunggu yang merupakan bagian dari suatu proses.

2.5. Penelitian Terdahulu

Sebelum menetapkan judul dalam penelitian ini, terlebih dahulu dilakukan penelusuran terhadap penelitian-penelitian sebelumnya yang terkait dengan penelitian ini. Berikut beberapa penelitian terdahulu yang terkait dengan penelitian ini:

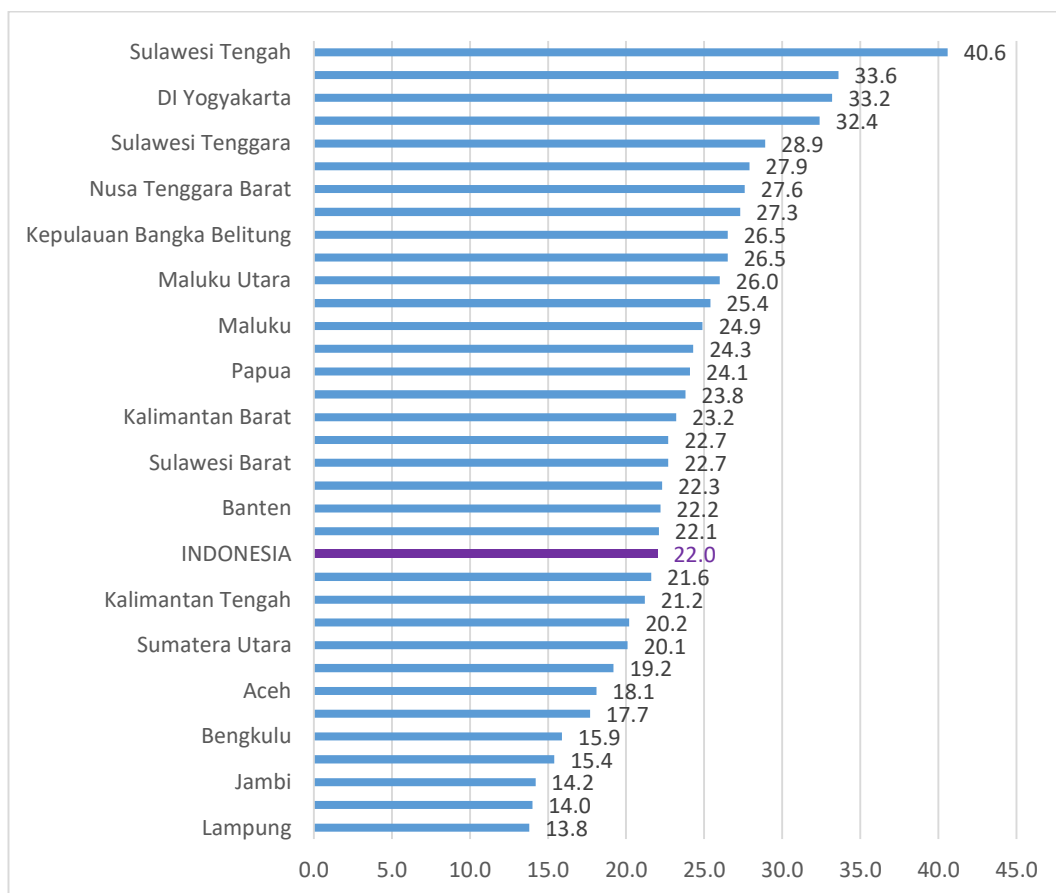
1. Disabilitas (“Disabilitas,” 2019).

Jurnal ini memberikan ringkasan data penyandang disabilitas yang ada di Indonesia berdasarkan Survei Penduduk Antar Sensus (SUPAS) tahun 2015 yang dikumpulkan oleh Badan Pusat Statistik (BPS) dan hasil Riset Kesehatan Dasar (Riskesdas) tahun 2018 yang telah dikumpulkan oleh Kementerian Kesehatan. Berikut ringkasan data Penyandang Disabilitas dalam Riskesdas.



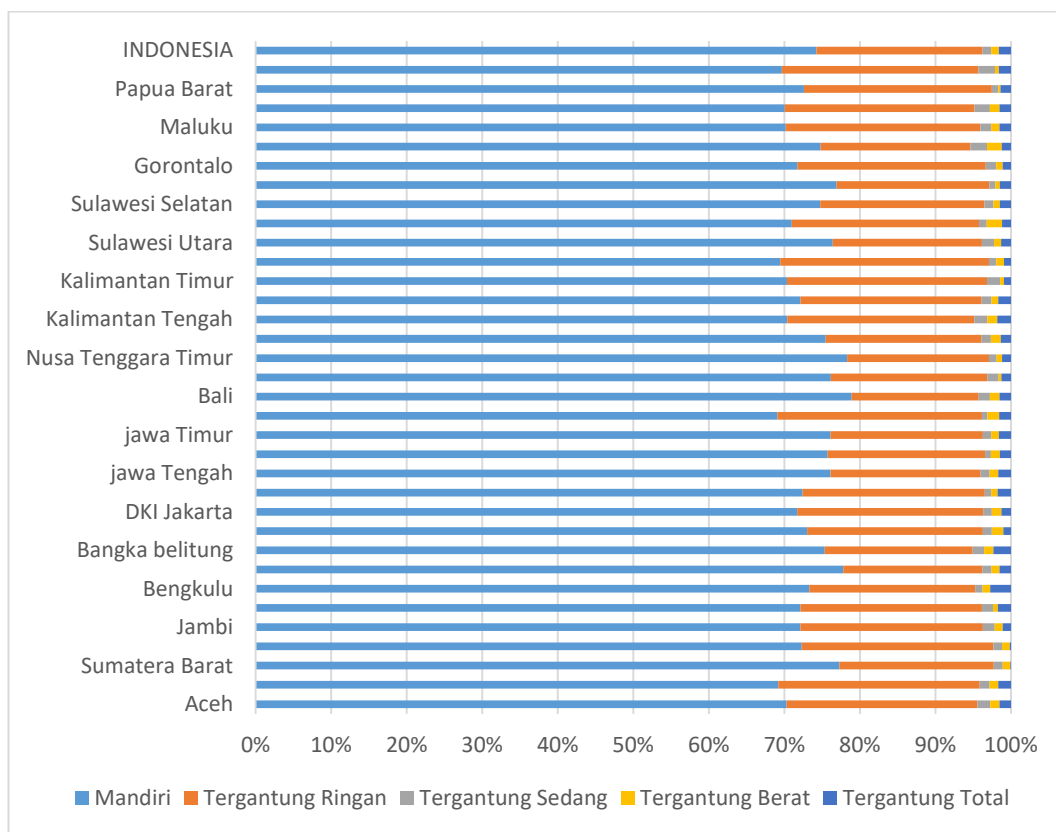
Gambar 2.12 Disabilitas pada Anak Umur 5-17 Tahun menurut Provinsi Tahun 2018
(Sumber :("Disabilitas," 2019))

Dari hasil tersebut anak umur 5-17 tahun yang mengalami disabilitas di Indonesia adalah sebesar 3,3%. Provinsi dengan proporsi disabilitas anak tertinggi adalah Sulawesi Tengah (7,0%), Kalimantan Utara, dan Gorontalo (masing-masing 5,4%), sedangkan proporsi terendah di Provinsi Sulawesi Barat, Lampung dan Jambi (masing-masing 1,4%).



Gambar 2.13 Proporsi Disabilitas Dewasa umur 18-59 Tahun menurut Provinsi di Indonesia Tahun 2018
(Sumber :("Disabilitas," 2019))

Proporsi disabilitas pada umur 18-59 tahun di Indonesia adalah sebesar 22,0%, proporsi terendah terdapat di Provinsi Lampung yaitu sebesar (13,8%), Kepulauan Riau (14,0%) dan Jambi (14,2%), proporsi tertinggi di Provinsi Sulawesi Tengah (40,6%), Sulawesi Selatan (33,6%), dan DI Yogyakarta (33,2%).



Gambar 2.14 Proporsi Disabilitas Lansia (umur ≥ 60 tahun) Menurut Provinsi di Indonesia Tahun 2018
(Sumber : (“Disabilitas,” 2019))

Didapatkan sebanyak 74,3% lansia (umur ≥ 60 tahun) dapat beraktifitas sehari-hari secara mandiri, 22,0% mengalami ketergantungan ringan, 1,1% mengalami ketergantungan sedang, 1% mengalami ketergantungan berat, dan 1,6% mengalami ketergantungan total. Provinsi dengan persentase lansia mandiri tertinggi adalah Bali (78,9%) dan terendah Banten (69,1%). Provinsi dengan proporsi lansia yang mengalami ketergantungan sedang, berat dan ketergantungan total tertinggi masing-masing adalah Sulawesi Barat (5,4%), Kepulauan Bangka Belitung (5,0%) dan Maluku Utara (4,9%), sedangkan provinsi dengan proporsi terendah adalah Papua Barat (2,6%) dan Kalimantan Utara, Sulawesi Tenggara, serta Nusa Tenggara Timur (masing-masing 2,9%).

2. *Application of Artificial Neural Networks in Modeling Direction Wheelchairs Using Neurosky Mindset Mobile (EEG) Device* (Siswoyo et al., 2017)

Penelitian ini dilakukan dengan latar belakang untuk membantu disabilitas dengan gangguan mobilitas. Penelitian ini menerapkan JST dengan algoritma *backpropagation* untuk memprediksi arah kursi roda listrik dengan masukan gelombang otak untuk disabilitas dengan gangguan mobilitas fisik. Perangkat yang digunakan dalam penelitian ini adalah Arduino Uno, *Neurosky Mindset sensor*, dan dengan perangkat lunak Matlab. Hasil percobaan didapatkan *Mean square Error* (MSE) pada akhir pelatihan sebesar 0,92495 pada epoch 9958, nilai koefisien korelasi sebesar 0,92804 menunjukkan bahwa akurasi hasil proses pelatihan sudah baik.

3. Identifikasi Kematangan Buah Tomat Berdasarkan Warna Menggunakan Metode Jaringan Saraf Tiruan (JST) (Kusumaningtyas & Asmara, 2016)

Penelitian ini dilatar belakangi oleh identifikasi kematangan buah dengan pengamatan manusia yang memiliki beberapa kelemahan diantaranya adalah waktu yang dibutuhkan relatif lama serta menghasilkan produk yang beragam karena adanya keterbatasan visual manusia, tingkat kelelahan dan perbedaan persepsi tentang mutu buah. Dalam penelitian ini JST dengan metode *perceptron* dan pengolahan citra digital diterapkan untuk mengidentifikasi kematangan buah tomat. Tampak visual buah didapatkan menggunakan webcam, yang kemudian kematangan buah tersebut diidentifikasi menggunakan JST. Tingkat keberhasilan identifikasi kematangan buah tomat dalam penelitian ini adalah sebesar 43,33%.

Dari hasil identifikasi yang diperoleh menghasilkan 3 output yaitu Mentah 10%, Setengah Matang 6,66%, dan Matang 26,66%.

4. Aplikasi Pengenalan Pola Daun Menggunakan Jaringan Saraf *Learning Vector Quantification* Untuk Penentuan Tanaman Obat (Indrawan, 2010)

Penelitian ini berangkat dari banyaknya jenis tanaman obat sehingga membuat sulit untuk mengidentifikasinya. Untuk menghindari pengonsumsi tanaman obat yang salah, penelitian ini kemudian dilakukan. Metode jaringan saraf yang digunakan dalam penelitian ini adalah LVQ. Pada penelitian ini JST digunakan untuk identifikasi pola daun tanaman obat, khususnya tanaman obat di daerah tropis. Pola yang akan diuji kemudian dibandingkan dengan pola hasil pelatihan. Hasil dari pengujian berupa informasi tanaman obat diantaranya nama tanaman, jenis, khasiat serta gambar daun dan pohon dari tanaman yang dicek.

5. Jaringan Saraf Tiruan *Learning Vector Quantization* untuk Aplikasi Pengenalan Tanda Tangan (Qur'ani & Rosmalinda, 2010)

Pada penelitian ini, digunakan metode deteksi tepi (*edge detection*) untuk mengolah citra yang digunakan sebagai masukan JST. Ini dilakukan untuk memperoleh garis tepi dari citra tanda tangan. Hasil pengujian menunjukkan bahwa JST dengan algoritma LVQ dapat mengenali pola tanda tangan dengan ketepatan 98% pada data yang di uji.

6. Identifikasi Telapak Tangan Menggunakan Jaringan Syaraf Tiruan *Learning Vector Quantization* (LVQ) (Afriandi & Sutikno, 2016).

Latar belakang penelitian ini adalah sistem pengenalan diri seperti kata sandi (*password*), ID card, atau PIN memiliki beberapa kelemahan yaitu dapat dicuri dan

mudah diduplikasi, memiliki kemungkinan seseorang untuk lupa dan beberapa password dapat diperkirakan sehingga dapat dimanfaatkan oleh orang-orang yang tidak bertanggung jawab. Penelitian ini menerapkan JST dengan metode LVQ untuk mengenali identitas seseorang melalui telapak tangan. Penelitian ini menggunakan model proses pengembangan perangkat lunak *Waterfall*. Perangkat lunak yang digunakan dalam penelitian ini diantaranya Matlab dan Microsoft Access sebagai basis datanya. Penelitian ini menggunakan JST dengan algoritma LVQ. Tingkat akurasi aplikasi hasil penelitian ini adalah sebesar 74,66%.

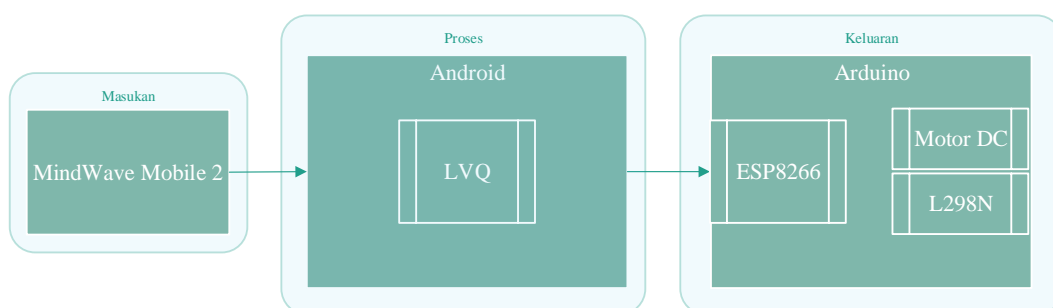
7. *Prediction of the Number of Airport Passengers Using Fuzzy C-Means and Adaptive Neuro Fuzzy Inference System* (Sitohang, Girsang, & Suharjito, 2017)

Penelitian yang dilakukan oleh Sitohang, Girsang, & Suharjito ini di latar belakang oleh masalah transportasi yang umum terjadi hampir di semua kota besar di dunia. Masalah transportasi tersebut diantaranya fasilitas transportasi yang terbatas, infrastruktur yang tidak memadai, tingkat disiplin yang rendah, urbanisasi yang cepat, dan perencanaan yang buruk. Sehubungan dengan masalah transportasi tersebut, diperlukan prediksi sebagai referensi untuk perencanaan. Prediksi jumlah penumpang merupakan hal yang penting untuk mempersiapkan fasilitas dalam mengantisipasi meningkatnya jumlah penumpang, seperti menyiapkan jadwal penerbangan tambahan, fasilitas ruang, tempat parkir yang lebih luas dan sebagainya. Dalam jurnal ini *Fuzzy C-Means* dan *Adaptif Neuro Fuzzy Inference System* digunakan untuk memprediksi jumlah penumpang bandara. Hasilnya menunjukkan bahwa “*Mean Absolute Percentage Errors (MAPE)* yang mewakili

kesalahan selama 4 tahun adalah 4,20%, 5,70%, 5,36% dan 4,47% dengan rata-rata 4,93%. Berdasarkan hasil ini, FCM dan ANFIS dapat dikombinasikan untuk memprediksi deret waktu data.

2.6. Kerangka Pemikiran

Berikut kerangka pemikiran yang penulis susun dalam penelitian ini.



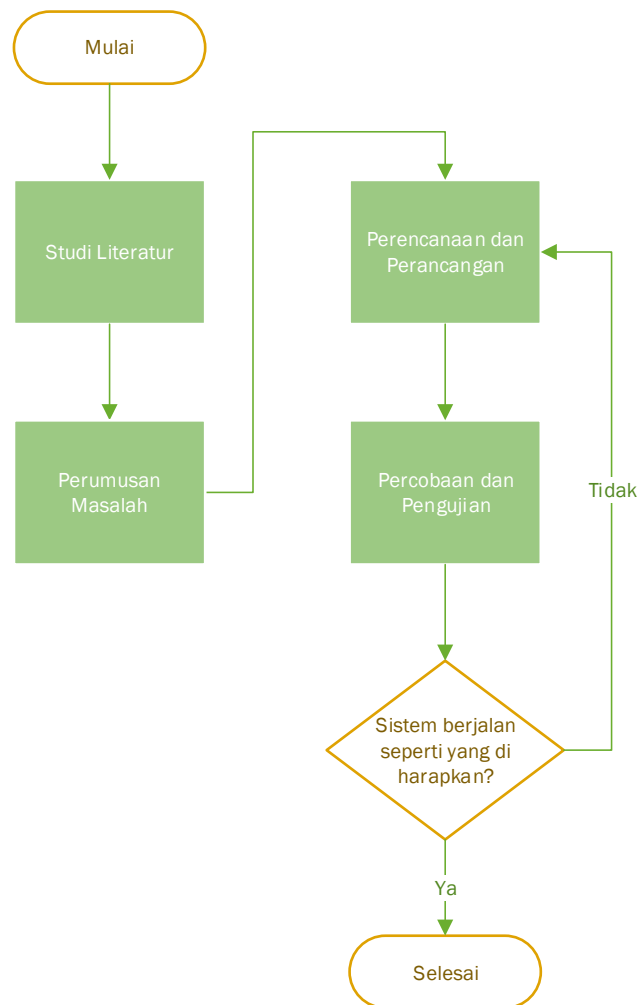
Gambar 2.15 Kerangka pemikiran
(Sumber : Data Penelitian, 2020)

MindWave Mobile 2 akan bertindak sebagai masukan (sumber data) dengan mengirimkan data berupa gelombang otak ke Android melalui *bluetooth*. Data tersebut kemudian di proses di Android menggunakan algoritma LVQ untuk menghasilkan data keluaran berupa kelas yang menentukan arah kursi roda yang dapat menggerakkan motor. Setelah melakukan klasifikasi Arduino kemudian mengirimkan data hasil proses ke ESP8266 melalui Wifi, kemudian Arduino akan menerima dari ESP8266 dan menentukan arah pergerakan dari kursi roda dan menggerakkan motor DC melalui L298N.

BAB III METODE PENELITIAN

3.1. Desain Penelitian

Tahapan proses pelaksanaan yang disusun dalam penelitian ini ditunjukkan pada Gambar 3.1.



Gambar 3.1 Desain penelitian
(Sumber : Data Penelitian, 2020)

Penelitian ini dimulai dari studi literatur tentang hal-hal yang berkaitan dengan penelitian ini seperti penyandang disabilitas, JST dengan algoritma LVQ,

Arduino, ESP8266, *MindWave Mobile 2*, L298N, dan Android. Studi literatur dilakukan di perpustakaan dan melalui situs-situs daring yang menyediakan literatur terkait dengan penelitian ini khususnya situs daring resmi dari penyedia teknologi atau alat yang digunakan dalam penelitian ini. Perumusan masalah kemudian ditentukan berdasarkan latar belakang dari penelitian ini setelah melakukan studi literatur. Pada tahap perencanaan dan perancangan akan dipilih teknologi apa saja yang akan digunakan dan melakukan perancangan untuk perangkat keras dan perangkat lunak yang akan digunakan.

Tahap percobaan dan pengujian dilanjutkan setelah tahap perencanaan dan perancangan. Jika hasil dari percobaan dan pengujian belum sesuai dengan yang diharapkan, maka akan kembali pada langkah perencanaan dan perancangan untuk melakukan perubahan atau perbaikan seperlunya hingga sesuai dengan tujuan penelitian. Jika hasil dari percobaan dan pengujian sudah sesuai dengan yang diharapkan, maka penelitian sudah dapat di akhiri.

3.2. Variabel Data Masukan dan Keluaran

Variabel data masukan yang akan digunakan dalam penelitian ini adalah nilai *Attention* dan *Meditation* yang didapatkan dari *MindWave Mobile 2*. Rentang nilai untuk *Attention* dan *Meditation* adalah 0 sampai 100. Variabel masukan akan terlebih dahulu di normalisasi kedalam rentang 0.1 sampai 0.9. Rumus yang digunakan untuk normalisasi rentang nilai data masukan ditunjukkan pada Rumus 3.1.

$$x_{norm} = (b - a) \frac{(x_i - x_{min})}{(x_{max} - x_{min})} + a \quad \text{Rumus 3.1 Normalisasi data}$$

Dimana x_{norm} adalah variabel hasil normalisasi, x_i adalah nilai yang akan di normalisasi, x_{min} dan x_{max} adalah nilai minimum dan maksimum dari variabel x_i , b dan a adalah nilai maksimum dan minimum rentang normalisasi.

Keluaran berupa kelas yang mendekati nilai bobot terdekat setelah melakukan penghitungan. Kelas keluaran dari JST merupakan nilai yang menentukan arah dari kursi roda. Kelas keluaran dalam penelitian ini ada tiga, yaitu *FORWARD*, *RIGHT*, dan *LEFT*. Kelas *FORWARD* yang memiliki nilai pengenalan 0 akan mengarahkan kursi roda ke arah depan. Kelas *RIGHT* yang memiliki nilai pengenalan 1 akan mengarahkan kursi roda ke arah kanan. Kelas *LEFT* yang memiliki nilai pengenalan 3 akan mengarahkan kursi roda ke arah kiri. Tabel 3.1 menunjukkan pemetaan arah dan kelas dengan nilai numeriknya.

Tabel 3.1 Pemetaan kelas dan nilai numeriknya

Nama kelas (label)	Nilai numerik	Arah
<i>FORWARD</i>	0	Depan
<i>RIGHT</i>	1	Kanan
<i>LEFT</i>	2	Kiri

(Sumber : Data Penelitian, 2020)

Meskipun hasil uji atau target kelas telah ditentukan dari ketiga kelas tersebut, motor tidak akan bergerak sebelum nilai arus diatur ke nilai selain 0. Nilai *blink* yang di dapatkan dari *MindWave Mobile 2* digunakan untuk mengontrol nilai arus untuk roda, akan tetapi nilai ini tidak digunakan dalam perhitungan JST. Nilai *blink* digunakan untuk mengalihkan nilai arus dari 0 ke nilai yang di tetapkan oleh pengguna atau sebaliknya. Nilai *blink* dapat diubah dengan pemicu tingkat kedipan mata.

3.3. Inisialisasi Parameter

Untuk inisialisasi bobot awal akan diambil data masukan sebanyak jumlah kelas target yaitu tiga, sedangkan untuk pelatihan digunakan sisa dari data masukan yang tidak digunakan untuk inisialisasi bobot. Beberapa parameter dapat ditetapkan oleh pengguna sebelum proses pelatihan. Parameter yang dapat ditetapkan oleh pengguna diantaranya *learning rate* (α), *maximum epoch*, dan nilai *error* minimum yang diharapkan (*eps*).

3.4. Rancangan arsitektur Jaringan Saraf Tiruan

Kedua nilai masukan terlebih dahulu di normalisasi menjadi ke dalam rentang 0.1 sampai 0.9 yang kemudian digunakan untuk proses pelatihan. Pada lapisan kompetitif, nilai bobot akan diperbarui saat proses pelatihan atau pembelajaran. Nilai bobot akhir dari proses pembelajaran akan digunakan untuk mendapatkan hasil keluaran. Keluaran berupa kelas yang mendekati nilai bobot terdekat setelah melakukan penghitungan. Kelas keluaran dari JST merupakan nilai yang menentukan arah dari kursi roda.

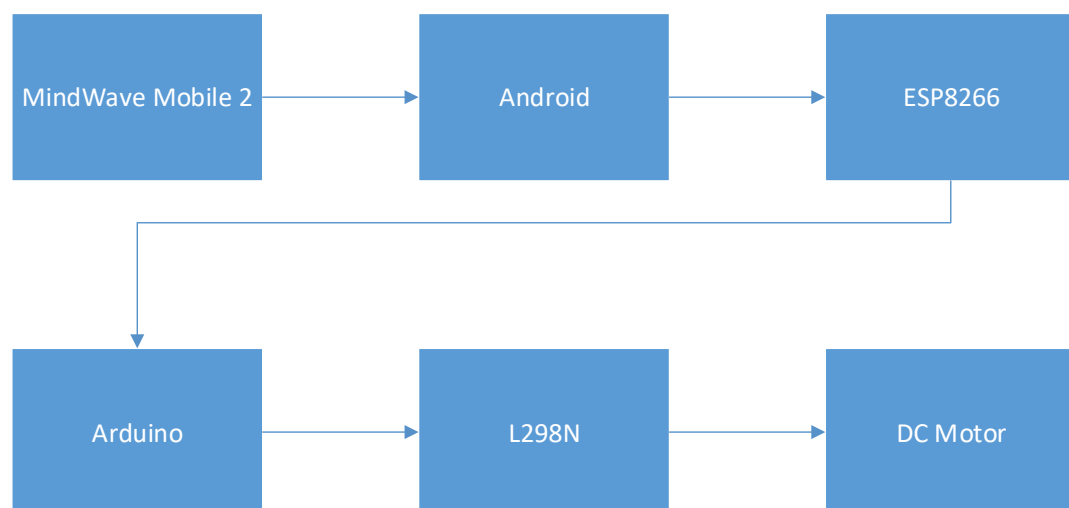
3.5. Perancangan Perangkat Keras dan Perangkat Lunak

Perancangan sistem pada penelitian ini meliputi perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Perangkat keras dan

perangkat lunak harus berjalan selaras agar sistem dapat bekerja seperti yang diharapkan.

3.5.1. Perancangan kerangka keras (*hardware*)

Perancangan perangkat keras dilakukan untuk mendapatkan gambaran bagaimana perangkat keras dapat bekerja baik secara mekanik maupun elektrik. Gambar 3.2 berikut merupakan diagram blok dari sistem yang akan dibangun dalam penelitian ini.

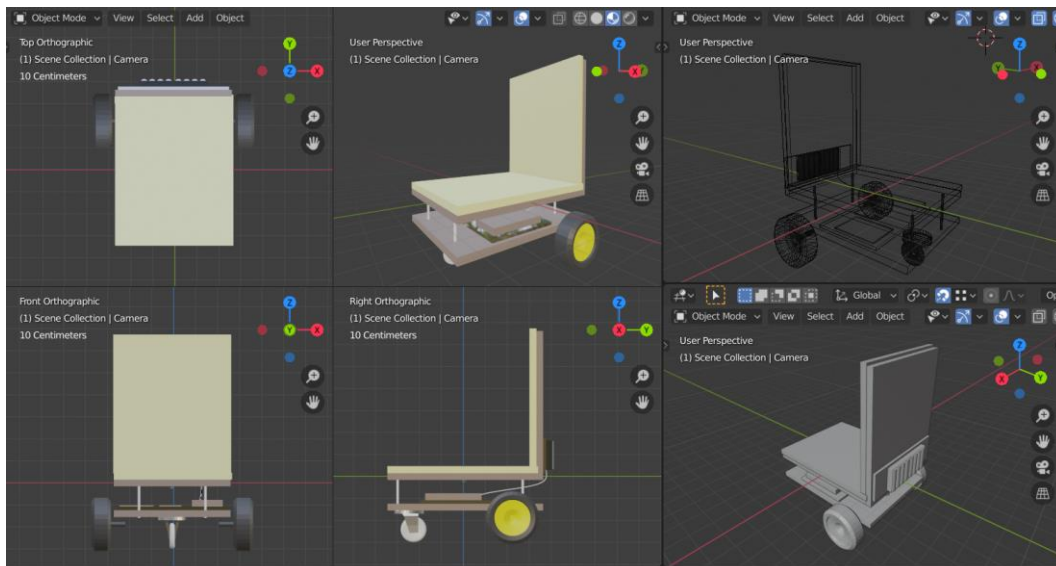


Gambar 3.2 Blok diagram perancangan perangkat keras
(Sumber : Data Penelitian, 2020)

3.5.1.1. Perancangan Mekanik

Dua buah motor DC ditempatkan di bagian belakang bawah rangka kursi roda, dan satu roda ditempatkan di depan bawah rangka kursi roda. Arduino, ESP8266 dan perangkat keras lain di tempatkan di bagian bawah dudukan kursi

roda. *MindWave Mobile 2* digunakan oleh pengguna dengan dipasangkan di bagian kepala.

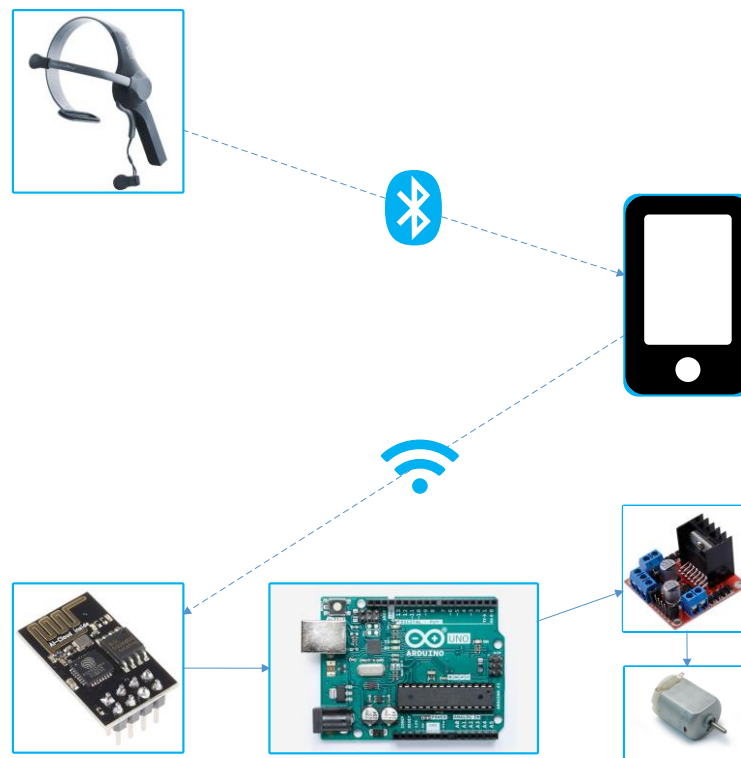


Gambar 3.3 Perancangan mekanik
(Sumber : Data Penelitian, 2020)

Ponsel akan menerima masukan dari *MindWave Mobile 2* yang kemudian diolah dan di kirim ke Arduino menggunakan perangkat ESP8266. Berdasarkan masukan yang di terima Arduino tidak akan menggerakkan atau menggerakkan motor DC melalui *motor driver* L298N.

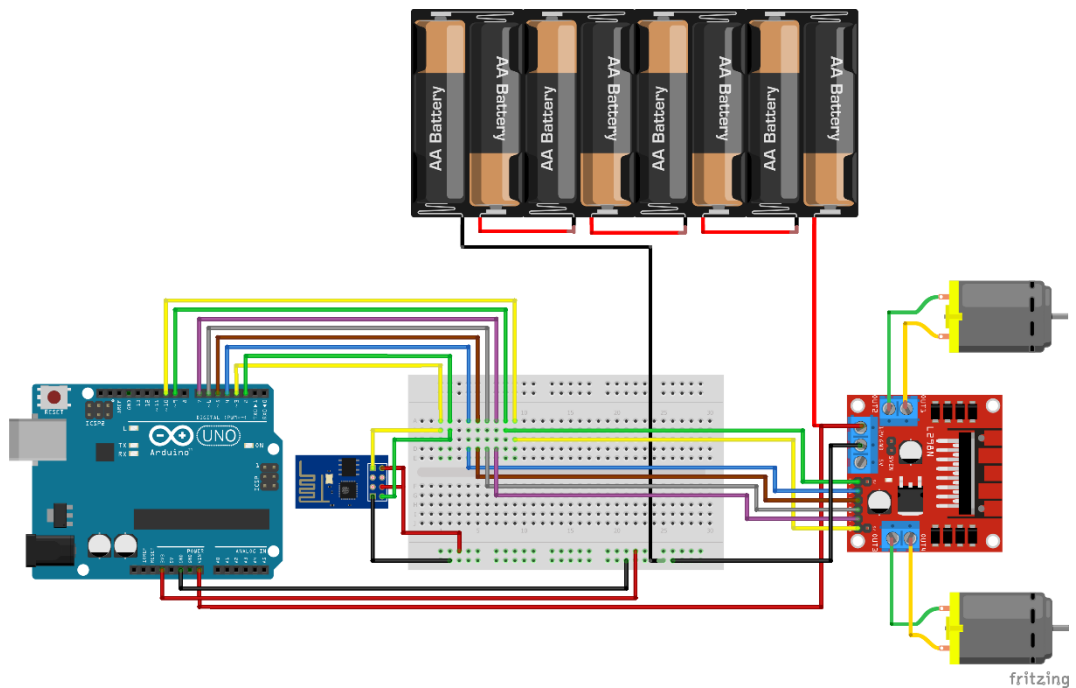
3.5.1.2. Perancangan Elektrik

ESP8266, motor DC dan L298N dihubungkan ke Arduino seperti pada Gambar 3.4.



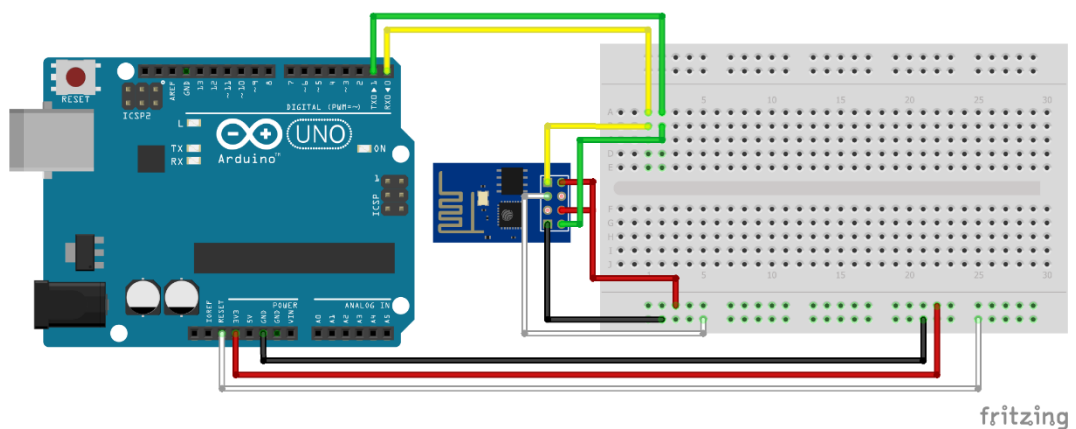
Gambar 3.4 Blok diagram perancangan elektrik
(Sumber : Data Penelitian, 2020)

Perangkat lunak diunggah ke ESP8266 dan Arduino sesuai dengan kegunaan masing-masing perangkat. Perangkat ESP8266 akan bertugas sebagai *access point* untuk *client* yang dalam hal ini adalah Android, dan sebagai *web server* dengan beberapa *end-point* untuk menerima permintaan dari *client* kemudian memproses data tersebut. Sebagai sumber tenaga listrik diberikan baterai yang dihungkan ke papan Arduino.



Gambar 3.5 Rangkaian penggunaan pin
(Sumber : Data Penelitian, 2020)

Gambar 3.5 merupakan rangkaian penggunaan pin untuk perancangan elektrik. Arduino akan menggerakkan motor DC melalui *motor driver* berdasarkan data yang diterima dari ESP8266 sesuai dengan aturan yang telah diberikan. Untuk mengunggah program ke ESP8266 digunakan rangkaian seperti pada Gambar 3.6 berikut.



Gambar 3.6 Rangkaian penggunaan pin untuk mengunggah program ke ESP8266
(Sumber : Data Penelitian, 2020)

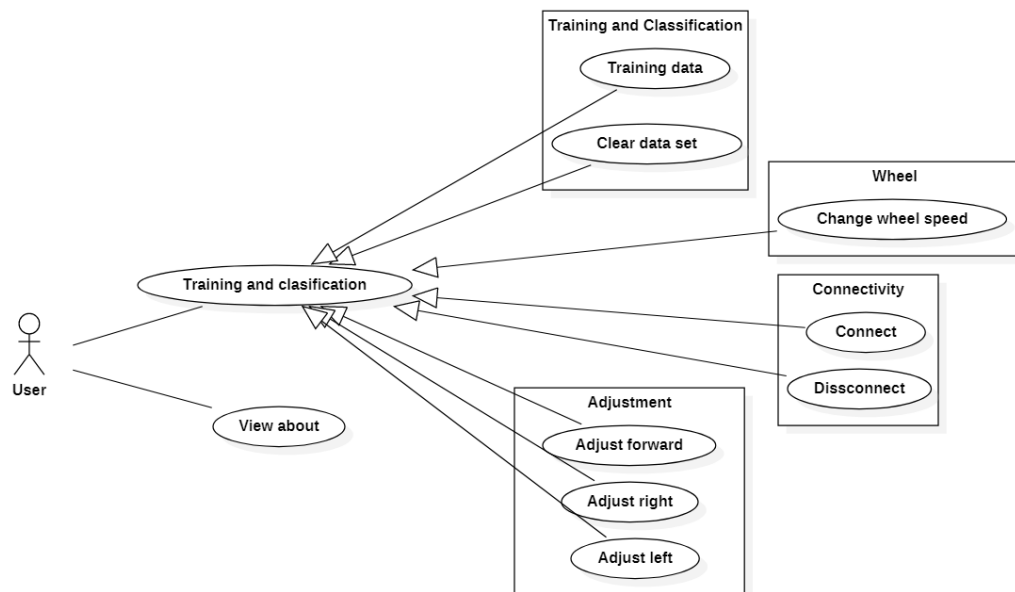
3.5.2. Perancangan perangkat lunak (*software*)

Dalam penelitian ini perangkat lunak yang akan dibuat ada tiga, yaitu perangkat lunak untuk ponsel Android, Arduino, dan untuk ESP8266. Perangkat lunak yang akan dipasang ke Android menggunakan paradigma Pemrograman Berorientasi Objek atau *Object Oriented Programming* (PBO/OOP), dan menggunakan spesifikasi UML untuk menspesifikasikan dan mendokumentasikannya. Diagram UML yang akan digunakan dalam penelitian ini diantaranya *usecase diagram*, *class diagram* dan *deployment diagram*. Bahasa pemrograman yang digunakan adalah Java dan Android Studio sebagai *Integrated Development Environment* (IDE) yang digunakan.

Perangkat lunak lainnya yaitu yang akan diunggah ke mikrokontroler Arduino dan ESP8266, ditulis dengan bahasa pemrograman C, dengan menggunakan Arduino IDE. Untuk perancangan dan dokumentasi program ini digunakan *flowchart diagram*.

3.5.2.1. *Use case diagram*

Berikut merupakan *use case diagram* dari sistem yang akan dibangun dalam penelitian ini.



Gambar 3.7 Use case diagram
(Sumber : Data Penelitian, 2020)

Deskripsi aktor pada sistem yang akan dibangun ditunjukkan pada Tabel 3.2.

Tabel 3.2 Deskripsi aktor

Aktor	Deskripsi
User	Orang yang menggunakan sistem

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Training and clasification* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.3.

Tabel 3.3 Deskripsi *use case Training and clasification*

Nama	<i>Training and clasification</i>
Deskripsi Singkat	Menampilkan antarmuka untuk pelatihan, pengujian dan konektivitas
Kondisi awal	User masuk ke sistem.
Kondisi akhir	User masuk ke menu <i>MindWave</i>
Situasi kesalahan	-
Status sistem jika terjadi kesalahan:	-
Aktor	User
Pemicu	User menekan tombol <i>MindWave</i> pada menu.

Tabel 3.3 (Lanjutan) Deskripsi *use case Training and clasification*

Proses standar:	<i>User</i> masuk ke sistem <i>User</i> menekan tombol <i>MindWave</i> pada menu.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case View about* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.4 berikut.

Tabel 3.4 Deskripsi *use case View about*

Nama	<i>View about</i>
Deskripsi Singkat	Melihat informasi tentang sistem.
Kondisi awal	<i>User</i> masuk ke sistem.
Kondisi akhir	<i>User</i> masuk ke menu About
Situasi kesalahan	-
Status sistem jika terjadi kesalahan:	-
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>About</i> pada menu.
Proses standar:	<i>User</i> masuk ke sistem <i>User</i> menekan tombol <i>About</i> pada menu.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Adjust forward* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.5 berikut.

Tabel 3.5 Deskripsi *use case Adjust forward*

Nama	<i>Adjust forward</i>
Deskripsi Singkat	Menambahkan data untuk pelatihan dengan target kelas <i>FORWARD</i>
Kondisi awal	<i>User</i> masuk ke menu <i>MindWave</i> . Koneksi ke <i>MindWave Mobile 2</i> telah terhubung.
Kondisi akhir	Data untuk pelatihan dengan target kelas <i>FORWARD</i> di tambah
Situasi kesalahan	Nilai <i>Attention</i> atau <i>Meditation</i> sama dengan nol.
Status sistem jika terjadi kesalahan:	Melakukan pengambilan data ulang hingga Nilai <i>Attention</i> atau <i>Meditation</i> tidak nol.
Aktor	<i>User</i>

Tabel 3.5 (Lanjutan) Deskripsi *use case Adjust forward*

Pemicu	<i>User</i> menekan tombol <i>Adjust forward</i> pada antarmuka <i>MindWave</i> .
Proses standar:	<i>User</i> menekan tombol <i>Adjust forward</i> pada antarmuka <i>MindWave</i> . <i>User</i> menetapkan kondisi pikiran untuk arah maju.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Adjust right* pada sistem ditunjukkan pada Tabel 3.6.

Tabel 3.6 Deskripsi *use case Adjust right*

Nama	<i>Adjust right</i>
Deskripsi Singkat	Menambahkan data untuk pelatihan dengan target kelas <i>RIGHT</i>
Kondisi awal	<i>User</i> masuk ke menu <i>MindWave</i> . Koneksi ke <i>MindWave Mobile 2</i> telah terhubung.
Kondisi akhir	Data pelatihan dengan target kelas <i>RIGHT</i> di tambah
Situasi kesalahan	Nilai <i>Attention</i> atau <i>Meditation</i> sama dengan nol.
Status sistem jika terjadi kesalahan:	Melakukan pengambilan data ulang hingga Nilai <i>Attention</i> atau <i>Meditation</i> tidak nol.
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Adjust right</i> pada antarmuka <i>MindWave</i> .
Proses standar:	<i>User</i> menekan tombol <i>Adjust right</i> pada antarmuka <i>MindWave</i> . <i>User</i> menetapkan kondisi pikiran untuk arah kanan.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Adjust left* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.7 berikut.

Tabel 3.7 Deskripsi *use case Adjust left*

Nama	<i>Adjust left</i>
Deskripsi Singkat	Menambahkan data untuk pelatihan dengan target kelas <i>LEFT</i>

Tabel 3.7 (Lanjutan) Deskripsi *use case Adjust left*

Kondisi awal	<i>User</i> masuk ke menu <i>MindWave</i> . Koneksi ke <i>MindWave Mobile 2</i> telah terhubung.
Kondisi akhir	Data untuk pelatihan dengan target kelas <i>LEFT</i> di tambah
Situasi kesalahan	Nilai <i>Attention</i> atau <i>Meditation</i> sama dengan nol.
Status sistem jika terjadi kesalahan:	Melakukan pengambilan data ulang hingga Nilai <i>Attention</i> atau <i>Meditation</i> tidak nol.
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Adjust left</i> pada antarmuka <i>MindWave</i> .
Proses standar:	<i>User</i> menekan tombol <i>Adjust left</i> pada antarmuka <i>MindWave</i> . <i>User</i> menetapkan kondisi pikiran untuk arah kiri.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Train data* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.8 berikut.

Tabel 3.8 Deskripsi *use case Train data*

Nama	<i>Train data</i>
Deskripsi Singkat	Melakukan pelatihan terhadap data yang telah di masukkan sebelumnya.
Kondisi awal	<i>User</i> masuk ke menu <i>MindWave</i> . Koneksi ke <i>MindWave Mobile 2</i> telah terhubung. Data masukan untuk setiap kelas telah mencukupi.
Kondisi akhir	Didapatkan bobot baru dari hasil pelatihan sesuai dengan masukan yang diberikan sebelumnya.
Situasi kesalahan	Data masukan tidak mencukupi.
Status sistem jika terjadi kesalahan:	Menampilkan " <i>Insufficient data sample</i> "
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>train</i> pada antarmuka <i>MindWave</i> .

Tabel 3.8 (Lanjutan) Deskripsi *use case Train data*

Proses standar:	<i>User</i> menekan tombol <i>Adjust stop</i> pada antarmuka <i>MindWave</i> . Sistem melakukan perhitungan untuk mendapatkan bobot baru.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Clear data set* pada sistem yang akan dibangun ditunjukkan seperti pada Tabel 3.9 berikut.

Tabel 3.9 Deskripsi *use case Clear data set*

Nama	<i>Clear data set</i>
Deskripsi Singkat	Mengosongkan masukan yang telah dikumpulkan sebelumnya.
Kondisi awal	<i>User</i> masuk ke menu <i>MindWave</i> .
Kondisi akhir	Data yang telah dikumpulkan sebelumnya dikosongkan.
Situasi kesalahan	-
Status sistem jika terjadi kesalahan:	-
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Clear data set</i> pada antarmuka <i>MindWave</i> .
Proses standar:	<i>User</i> menekan tombol <i>Clear data set</i> pada antarmuka <i>MindWave</i> . Sistem mengosongkan data yang telah dikumpulkan sebelumnya.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Connect* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.10 berikut.

Tabel 3.10 Deskripsi *use case Connect*

Nama	<i>Connect</i>
Deskripsi Singkat	Menghubungkan koneksi dengan <i>MindWave Mobile 2</i>
Kondisi awal	Koneksi dengan <i>MindWave Mobile 2</i> tidak terhubung. <i>User</i> masuk ke antarmuka <i>MindWave</i>

Tabel 3.10 (Lanjutan) Deskripsi *use case Connect*

Kondisi akhir	Koneksi dengan <i>MindWave Mobile 2</i> terhubung
Situasi kesalahan	Koneksi <i>MindWave Mobile 2</i> tidak terhubung
Status sistem jika terjadi kesalahan:	Sistem menampilkan “ <i>Not found</i> ” pada <i>field state</i>
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Connect</i>
Proses standar:	<i>User</i> menekan tombol <i>Connect</i> . Sistem menghubungkan konektivitas dengan <i>MindWave Mobile 2</i> .
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Dissconnect* pada sistem yang akan dibangun adalah seperti pada Tabel 3.11 berikut.

Tabel 3.11 Deskripsi *use case Dissconnect*

Nama	<i>Dissconnect</i>
Deskripsi singkat	Memutuskan koneksi dengan <i>MindWave Mobile 2</i>
Kondisi awal	Koneksi dengan <i>MindWave Mobile 2</i> terhubung. <i>User</i> masuk ke menu <i>MindWave</i>
Kondisi akhir	Koneksi dengan <i>MindWave Mobile 2</i> terputus
Situasi kesalahan	Koneksi <i>MindWave Mobile 2</i> tidak terhubung
Status sistem jika terjadi kesalahan:	Sistem tidak memberikan reaksi
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Dissconnect</i>
Proses standar:	<i>User</i> menekan tombol <i>Dissconnect</i> . Sistem memutuskan konektivitas dengan <i>MindWave Mobile 2</i> .
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

Deskripsi *use case Change wheel speed* pada sistem yang akan dibangun ditunjukkan pada Tabel 3.12 berikut.

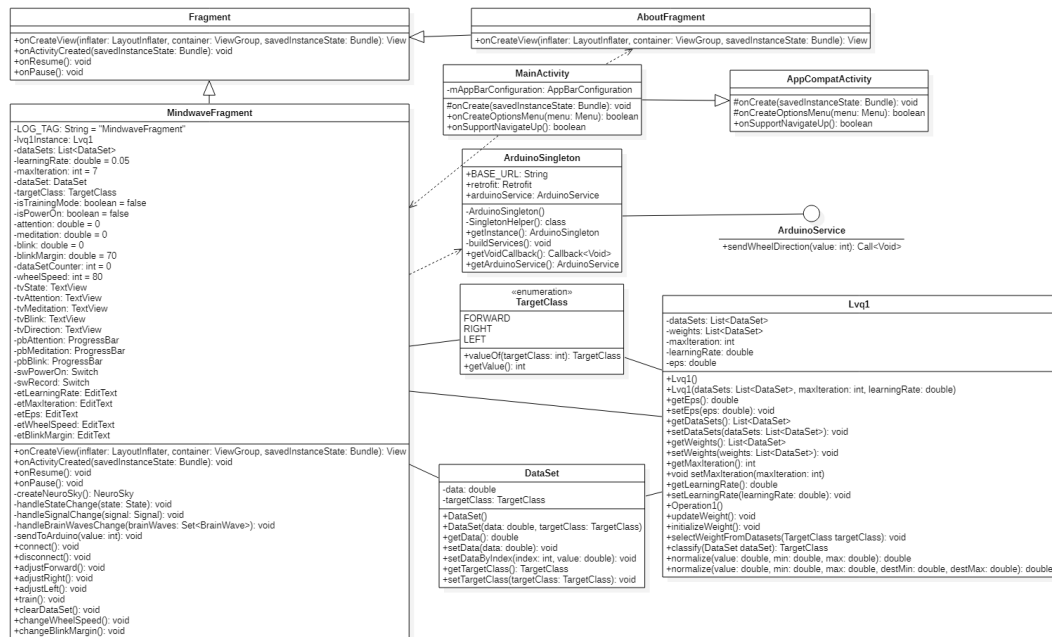
Tabel 3.12 Deskripsi use case *Change wheel speed*

Nama	<i>Change wheel speed</i>
Deskripsi Singkat	Mengubah kecepatan motor pada kursi roda.
Kondisi awal	<i>User</i> masuk ke sistem <i>User</i> masuk ke menu <i>MindWave</i>
Kondisi akhir	Kecepatan motor pada kursi roda berubah.
Situasi kesalahan	Masukan untuk kecepatan tidak dapat di <i>parse</i> ke <i>integer</i> Masukan yang diberikan lebih kecil dari nilai minimum, dan lebih besar dari nilai maksimum yang di izinkan yang masing-masing bernilai 70 dan 255
Status sistem jika terjadi kesalahan:	Menampilkan “ <i>Please check the input you provided!</i> ”
Aktor	<i>User</i>
Pemicu	<i>User</i> menekan tombol <i>Change</i>
Proses standar:	<i>User</i> memasukkan nilai kecepatan yang di inginkan pada kotak isian <i>Motor speed</i> . <i>User</i> menekan tombol <i>Change</i> Sistem memperbarui nilai kecepatan dan mengirimkan nilai kecepatan yang baru ke Arduino.
Proses alternatif:	-

(Sumber : Data Penelitian, 2020)

3.5.2.2. Class Diagram

Berikut merupakan *class diagram* keseluruhan dari sistem yang akan dibangun dalam penelitian ini.



Gambar 3.8 Class diagram (Sumber : Data Penelitian, 2020)

Deskripsi masing-masing elemen pada class diagram dalam penelitian ini ringkas dalam Tabel 3.13 berikut.

Tabel 3.13 Deskripsi class diagram

Nama elemen	Typ	Deskripsi singkat
MindwaveFragment	Class	Antarmuka pengguna untuk melakukan pelatihan dan pengklasifikasian JST, konfigurasi kecepatan motor dan pengaturan konektivitas dengan MindWave Mobile 2.
MainActivity	Class	Activity utama sistem.
ArduinoService	Interface	Menyediakan antarmuka untuk mengakses layanan Arduino.
ArduinoSingleton	Class	Memberikan akses langsung ke ArduinoService.
DataSet	Class	Model untuk JST.
TargetClass	Enumeration	Berisi label dan nilai numerik untuk target kelas dari JST.
Lvq1	Class	Berisi algoritma LVQ dan operasi lain yang dibutuhkan untuk melakukan perhitungan JST.

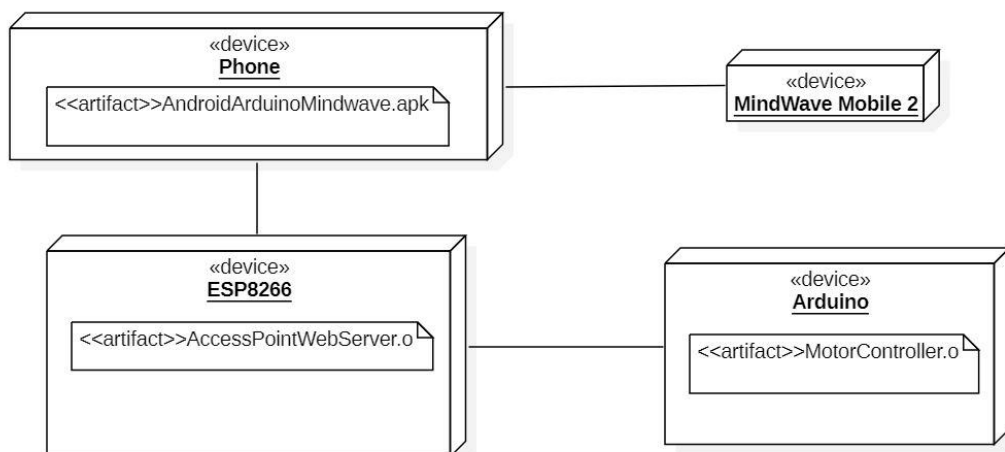
Tabel 3.13 (Lanjutan) Deskripsi class diagram

Nama elemen	Tipe	Deskripsi singkat
AboutFragment	<i>Class</i>	Antarmuka pengguna <i>About</i> , untuk menampilkan informasi tentang aplikasi.
AppCompatActivity	<i>Class</i>	Kelas ini dari Android SDK, berperan sebagai parent class untuk <i>MainActivity</i> .

(Sumber : Data Penelitian, 2020)

3.5.2.3. Deployment Diagram

Deployment diagram keseluruhan dari sistem yang akan dibangun dalam penelitian ini ditunjukkan pada Gambar 3.9.



Gambar 3.9 *Deployment diagram*
(Sumber : Data Penelitian, 2020)

Deskripsi singkat untuk elemen pada *deployment diagram* seperti pada Gambar 3.9 ditunjukkan pada Tabel 3.14.

Tabel 3.14 Deskripsi *deployment diagram*

Nama	Spesifikasi <i>deployment</i>	<i>Artifact</i>	Deskripsi singkat
Phone	<i>device</i>	AndroidArduino Mindwave.apk	Aplikasi yang dibangun untuk Android akan di deploy ke ponsel dengan sistem operasi Android

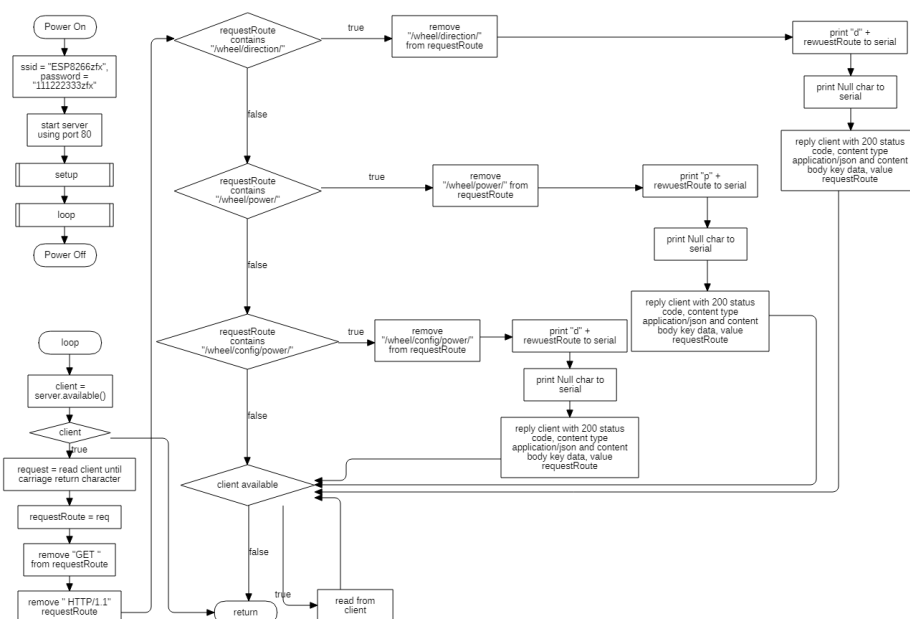
Tabel 3.14 (Lanjutan) Deskripsi *deployment diagram*

Nama	Spesifikasi <i>deployment</i>	Artifact	Deskripsi singkat
MindWave Mobile 2	<i>device</i>	-	Perangkat ini berperan sebagai masukan utama.
ESP8266	<i>device</i>	AccessPointWeb Server.o	Perangkat ini berperan sebagai penengah antara Arduino dan Android.
Arduino	<i>device</i>	MotorController.o	Perangkat ini akan mengontrol DC Motor sesuai dengan masukan yang diterima dari ESP8266.

(Sumber : Data Penelitian, 2020)

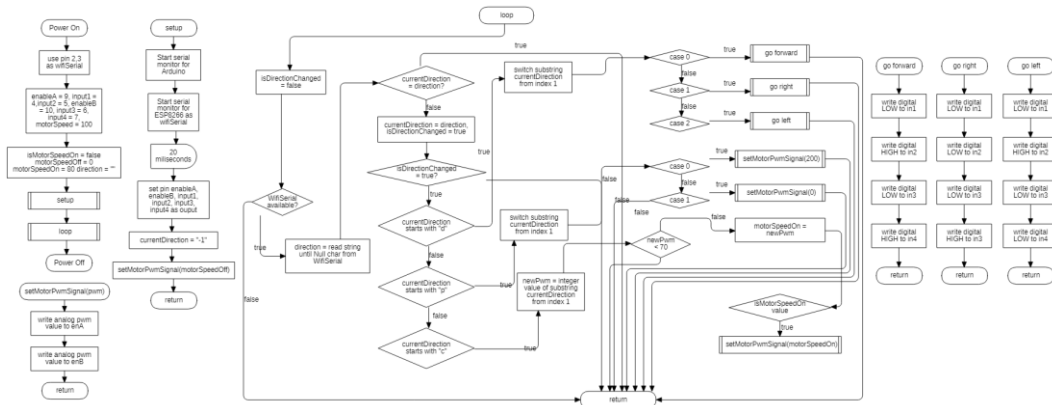
3.5.2.4. Flowchart Diagram

Untuk *flowchart diagram* dalam penelitian ini dibagi menjadi dua bagian, yaitu *flowchart diagram* untuk ESP8266 dan *flowchart diagram* Arduino. *Flowchart diagram* yang dirancang untuk ESP8266 ditunjukkan pada Gambar 3.10.



Gambar 3.10 Flowchart diagram Access Point dan Web server untuk program yang di akan diunggah ke ESP8266
(Sumber : Data Penelitian, 2020)

Flowchart diagram yang dirancang dalam penelitian ini untuk program yang akan di unggah ke Arduino ditunjukkan pada Gambar 3.11.



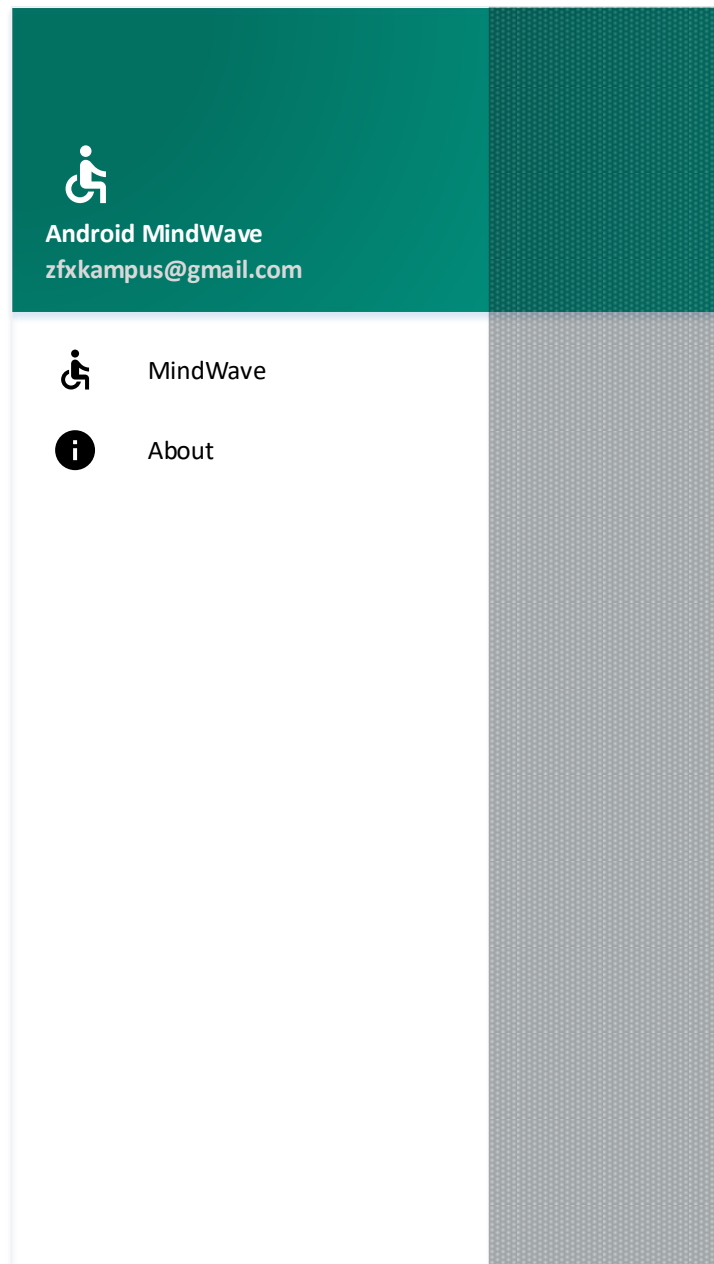
Gambar 3.11 Flowchart diagram untuk program yang di unggah ke Arduino (Sumber : Data Penelitian, 2020)

3.5.2.5. Antarmuka Pengguna

Terdapat tiga rancangan antarmuka pengguna yang di rancang oleh peneliti, diantaranya rancangan antarmuka menu, antarmuka *Mindwave*, dan antarmuka *About*.

1. Antarmuka pengguna untuk pemilihan menu.

Rancangan antarmuka pengguna untuk memilih menu pada sistem yang akan dibangun ditunjukkan pada Gambar 3.12.

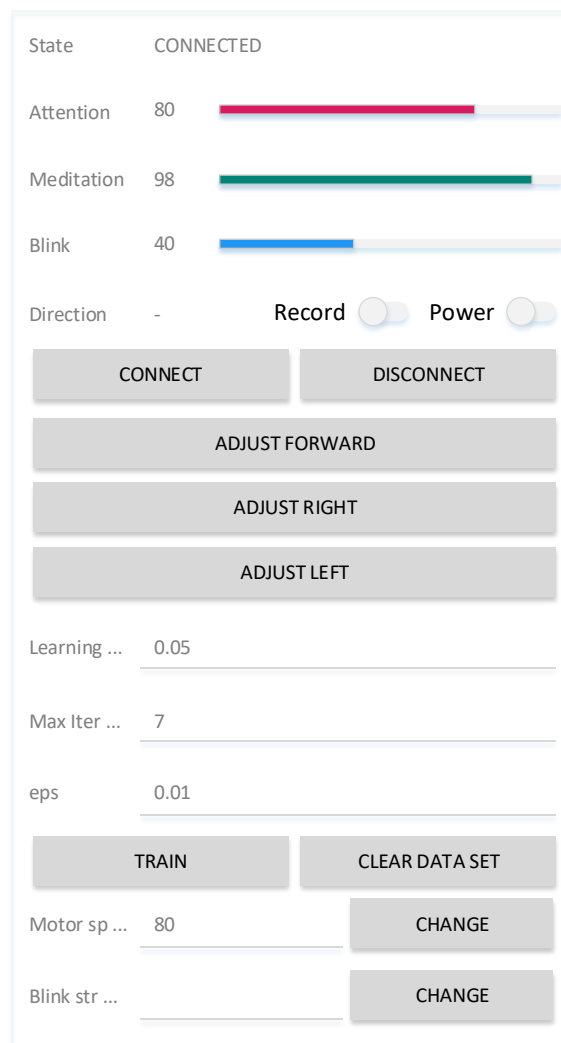


Gambar 3.12 Perancangan antarmuka pengguna pemilihan menu
(Sumber : Data Penelitian, 2020)

Terdapat dua pilihan menu yang tersedia pada sistem yaitu *MindWave* dan *About*. Menu *MindWave* akan menampilkan antarmuka *MindWave* dan *About* akan menampilkan antarmuka *About*.

2. Antarmuka pengguna *MindWave*.

Antarmuka pengguna *MindWave* digunakan oleh pengguna untuk mengatur konektivitas, melakukan pengumpulan data, pelatihan dan pengklasifikasian arah kursi roda. Rancangan antarmuka pengguna *MindWave* pada sistem yang akan dibangun ditunjukkan pada Gambar 3.13.

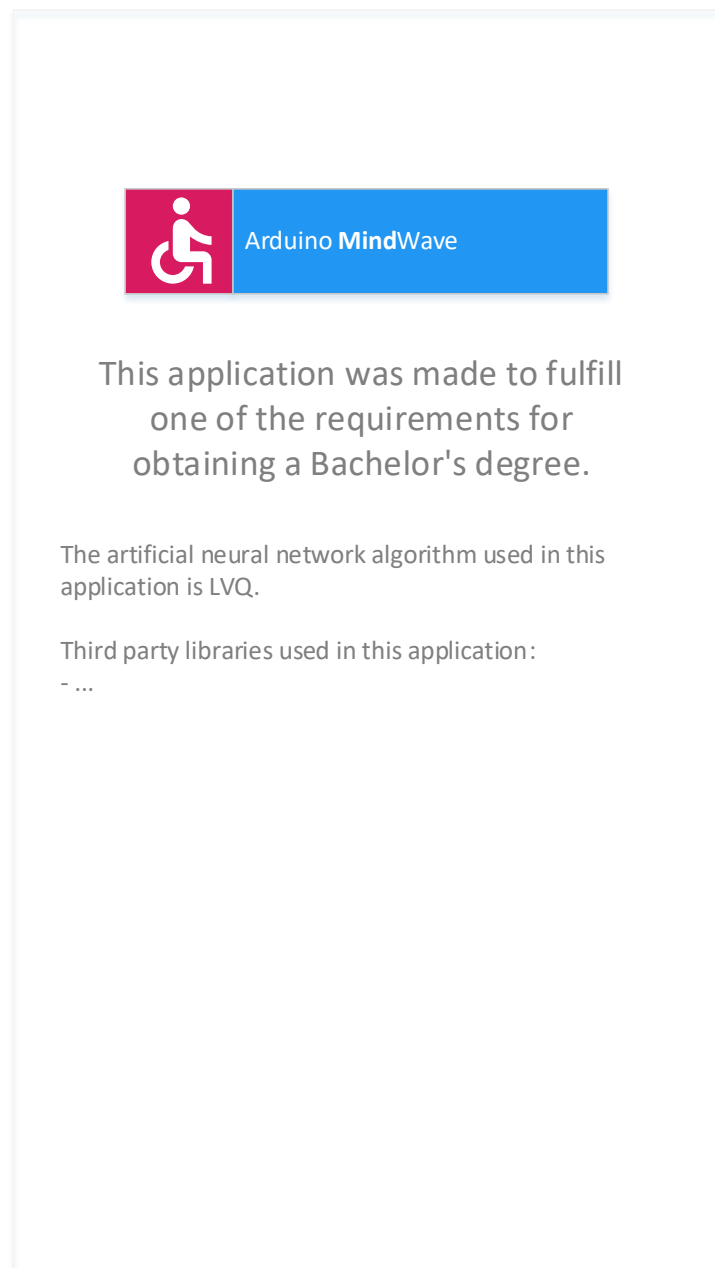


Gambar 3.13 Perancangan antarmuka pengguna *MindWave*
(Sumber : Data Penelitian, 2020)

4. Antarmuka pengguna *About*.

Antarmuka pengguna ini berisi informasi tentang sistem yang akan dibangun.

Rancangan antarmuka pengguna *About* ditunjukkan pada Gambar 3.14.



Gambar 3.14 Perancangan antarmuka pengguna *About*
(Sumber : Data Penelitian, 2020)