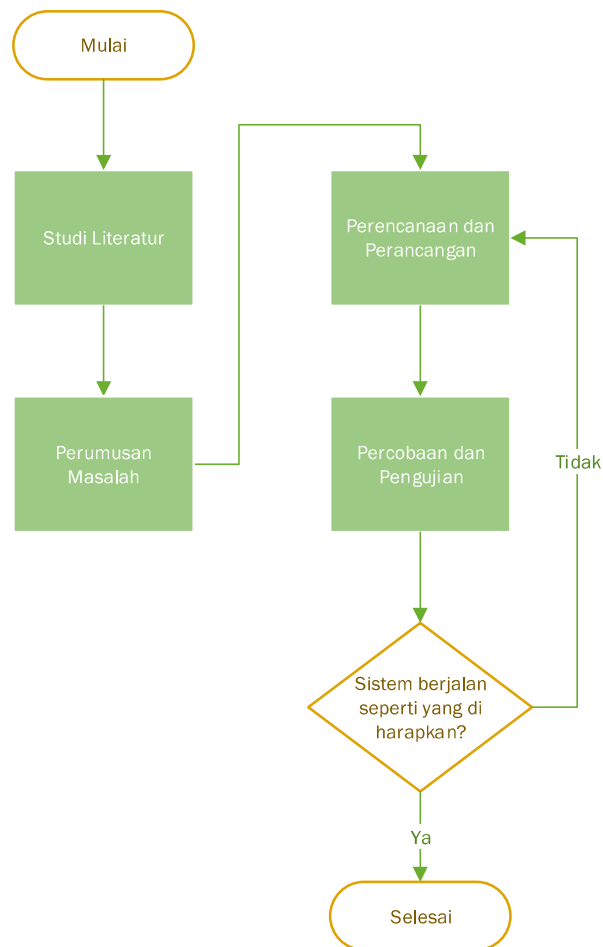


BAB III METODE PENELITIAN

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 khusus nya 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.

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

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.

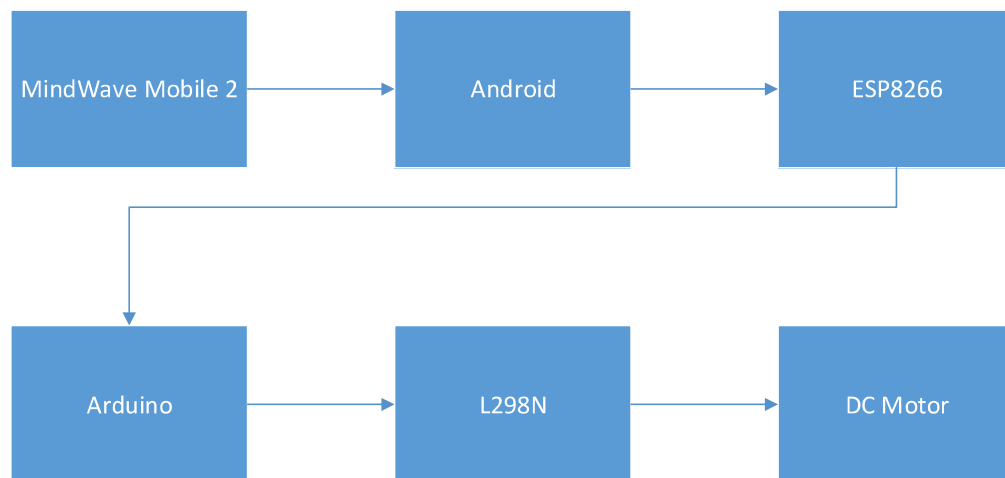
3.5. Perancangan Perangkat Keras dan Perangkat Lunak

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.

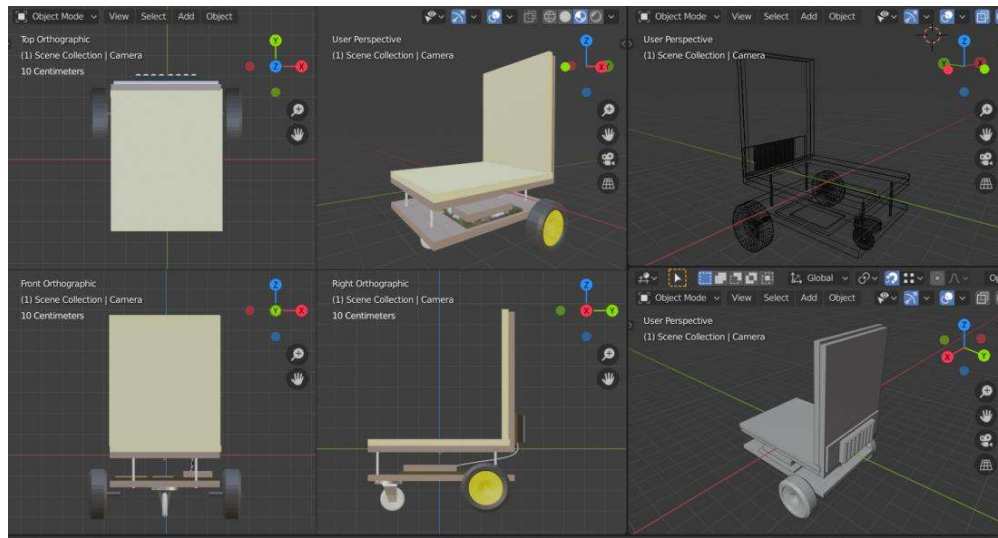


Gambar 3.2 Blok diagram perancangan perangkat keras
(Sumber : Data Penelitian, 2020)

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 ditempatkan di bagian bawah dudukan kursi

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

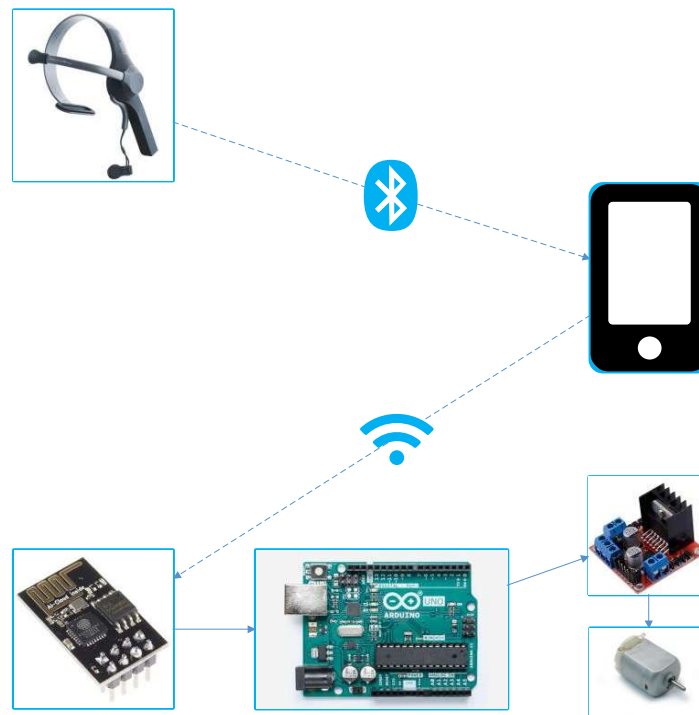


Gambar 3.3 Perancangan mekanik
(Sumber : Data Penelitian, 2020)

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.

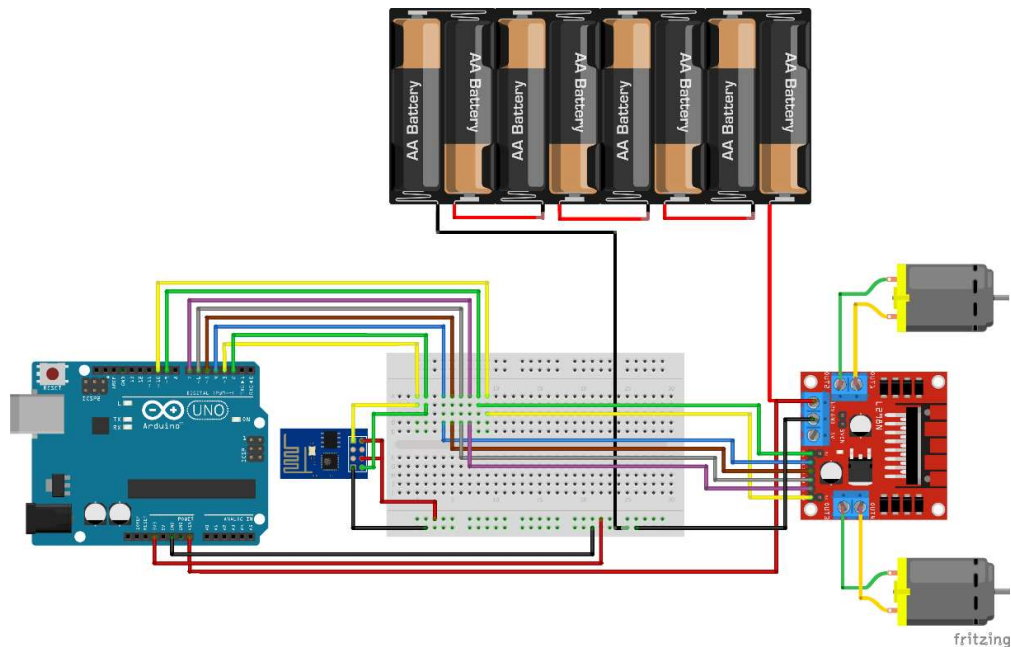
3.5.1.2. Perancangan Elektrik

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



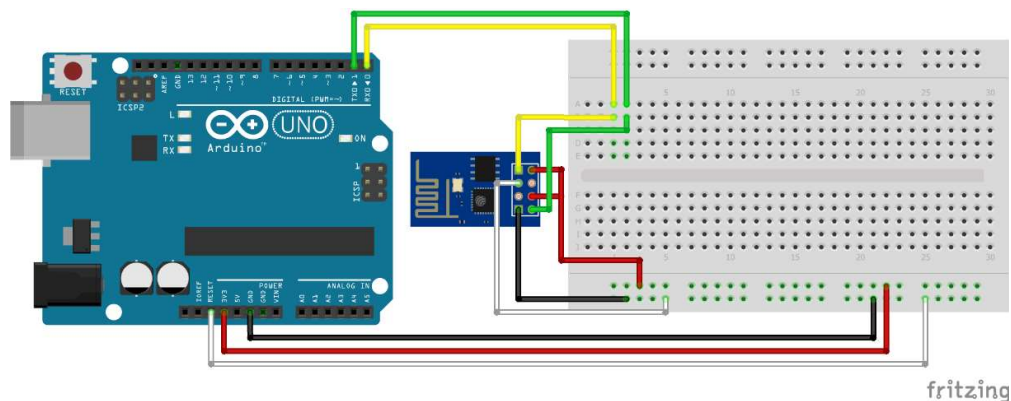
Gambar 3.4 Blok diagram perancangan elektrik
(Sumber : Data Penelitian, 2020)

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)

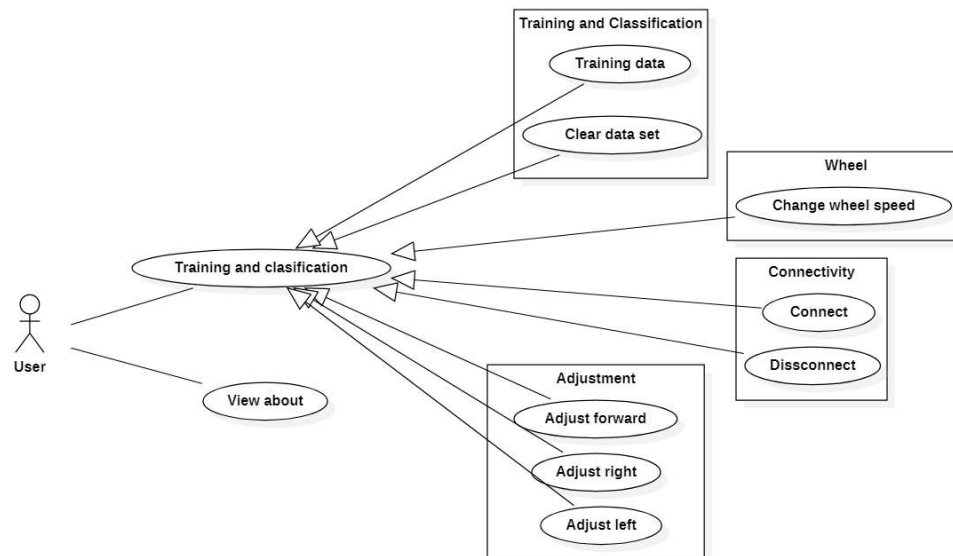
3.5.2. Perancangan perangkat lunak (*software*)

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 Disconnect* pada sistem yang akan dibangun adalah seperti pada Tabel 3.11 berikut.

Tabel 3.11 Deskripsi *use case Disconnect*

Nama	<i>Disconnect</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>Disconnect</i>
Proses standar:	<i>User</i> menekan tombol <i>Disconnect</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.

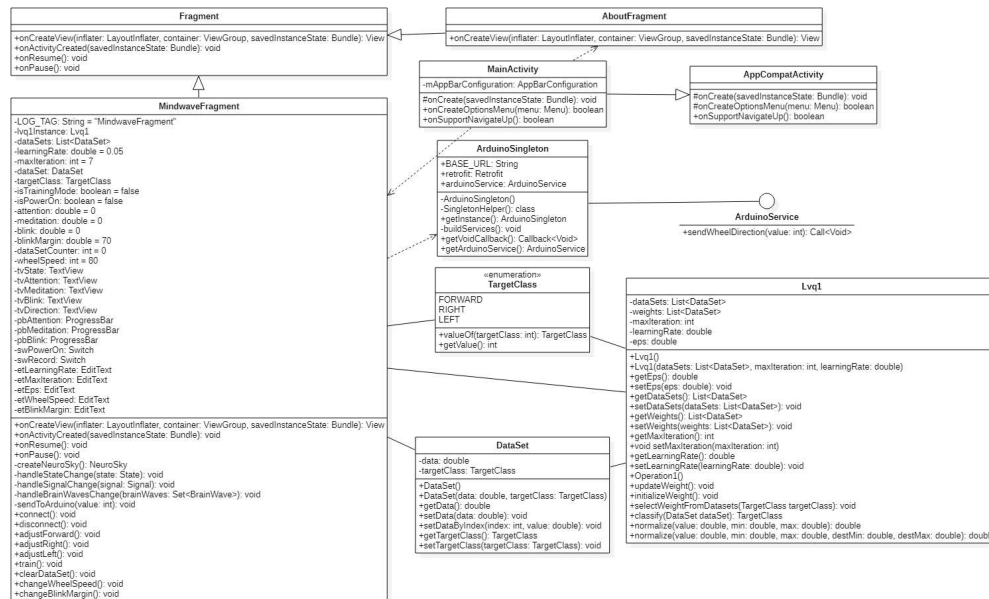
Tabel 3.12 Deskripsi *use case Change wheel speed*

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	Tipe	Deskripsi singkat
MindwaveFragment	<i>Class</i>	Antarmuka pengguna untuk melakukan pelatihan dan pengklasifikasian JST, konfigurasi kecepatan motor dan pengaturan konektivitas dengan <i>MindWave Mobile 2</i> .
MainActivity	<i>Class</i>	<i>Activity</i> utama sistem.
ArduinoService	<i>Interface</i>	Menyediakan antarmuka untuk mengakses layanan Arduino.
ArduinoSingleton	<i>Class</i>	Memberikan akses langsung ke ArduinoService.
DataSet	<i>Class</i>	Model untuk JST.
TargetClass	<i>Enumeration</i>	Berisi label dan nilai numerik untuk target kelas dari JST.
Lvl1	<i>Class</i>	Berisi algoritma LVQ dan operasi lain yang dibutuhkan untuk melakukan perhitungan JST.

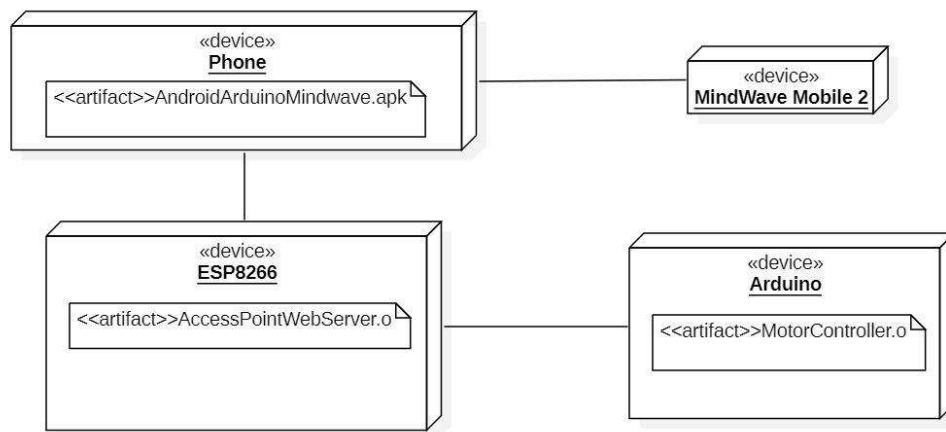
Tabel 3.13 (Lanjutan) Deskripsi class diagram

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

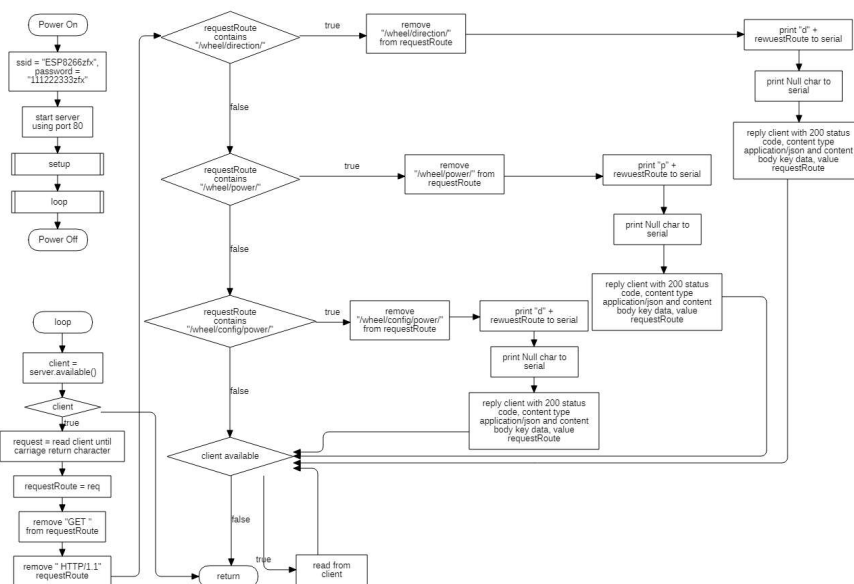
Tabel 3.14 (Lanjutan) Deskripsi *deployment diagram*

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

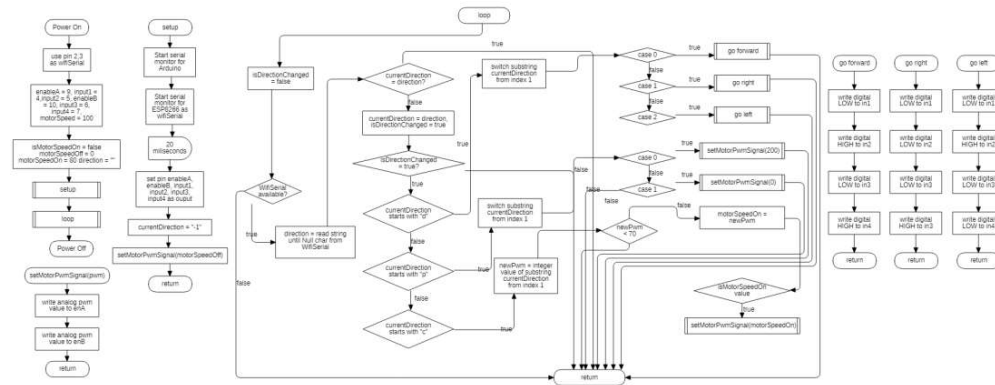
Untuk *flowchart diagram* dalam penelitian ini dibagi menjadi dua bagian, yaitu *flowchart diagram* untuk EPS8266 dan *flowchart diagram* Arduino. *Flowchart diagram* yang dirancang untuk ESP8266 ditunjukkan pada Gambar 3.10.



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.



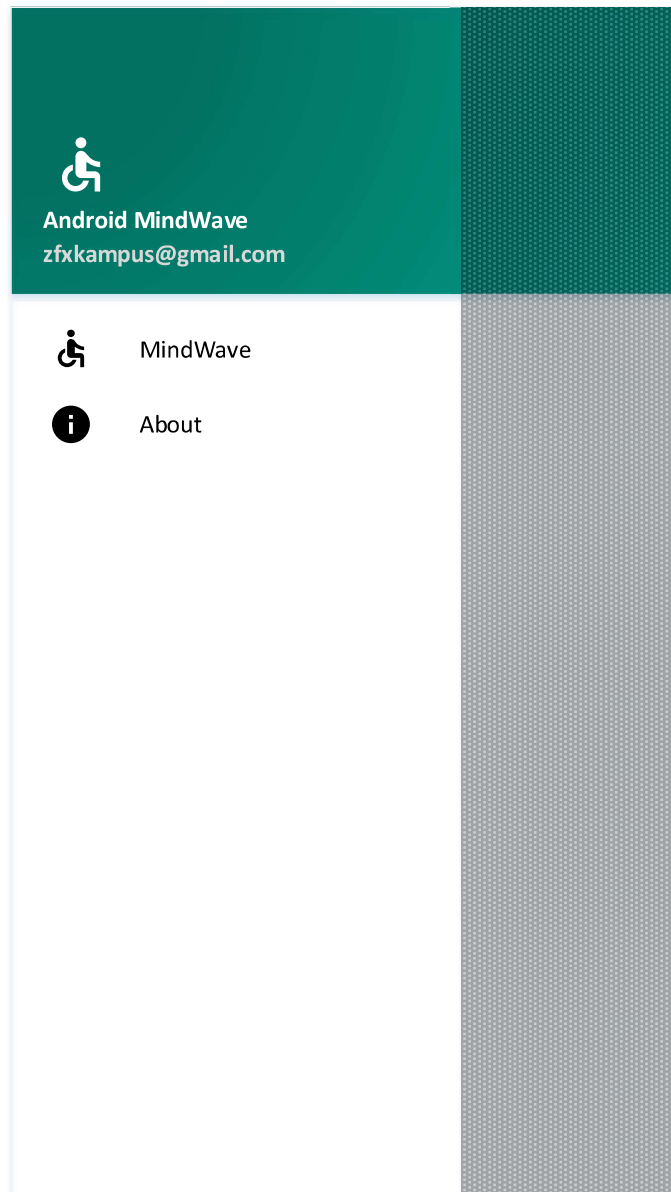
Gambar 3.11 *Flowchart diagram* untuk program yang di unggah ke Arduino
(Sumber : Data Penelitian, 2020)

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.

The screenshot displays the MindWave user interface with the following elements:

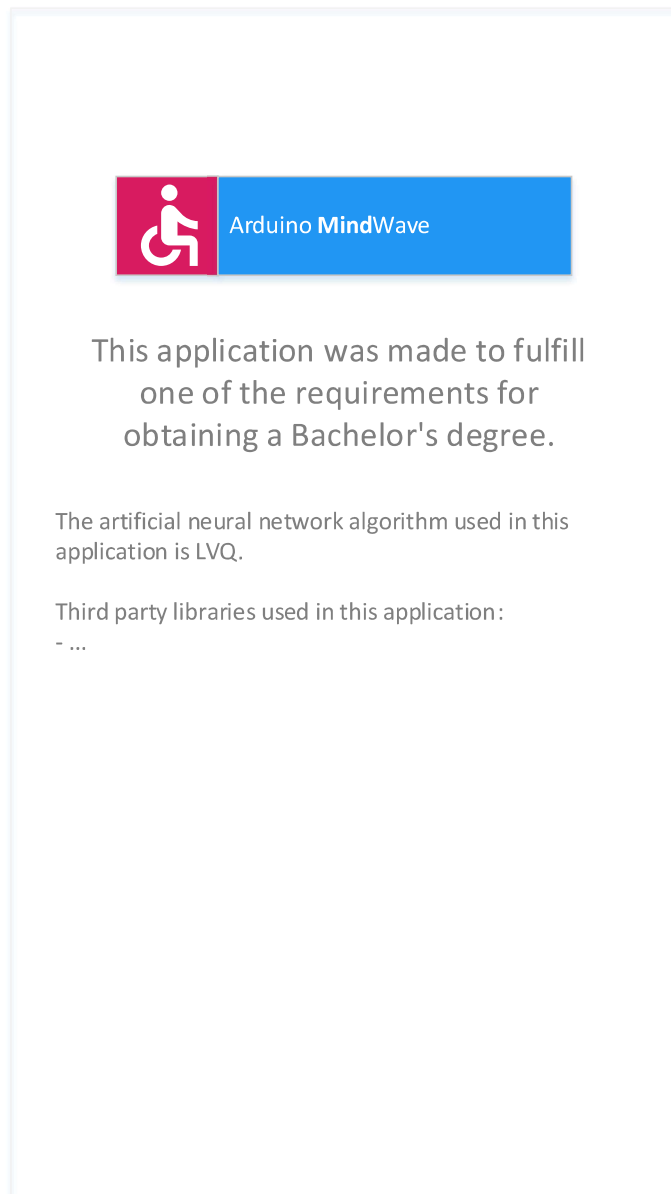
- State:** CONNECTED
- Attention:** 80 (represented by a pink progress bar)
- Meditation:** 98 (represented by a green progress bar)
- Blink:** 40 (represented by a blue progress bar)
- Direction:** -
- Record:** ☐
- Power:** ☐
- CONNECT** button
- DISCONNECT** button
- ADJUST FORWARD** button
- ADJUST RIGHT** button
- ADJUST LEFT** button
- Learning ...:** 0.05
- Max Iter ...:** 7
- eps:** 0.01
- TRAIN** button
- CLEAR DATA SET** button
- Motor sp ...:** 80
- CHANGE** button (next to Motor sp ...)
- Blink str ...:**
- CHANGE** button (next to Blink str ...)

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)