

BAB 3

ANALISIS DAN PERANCANGAN

3.1. Analisis Sistem

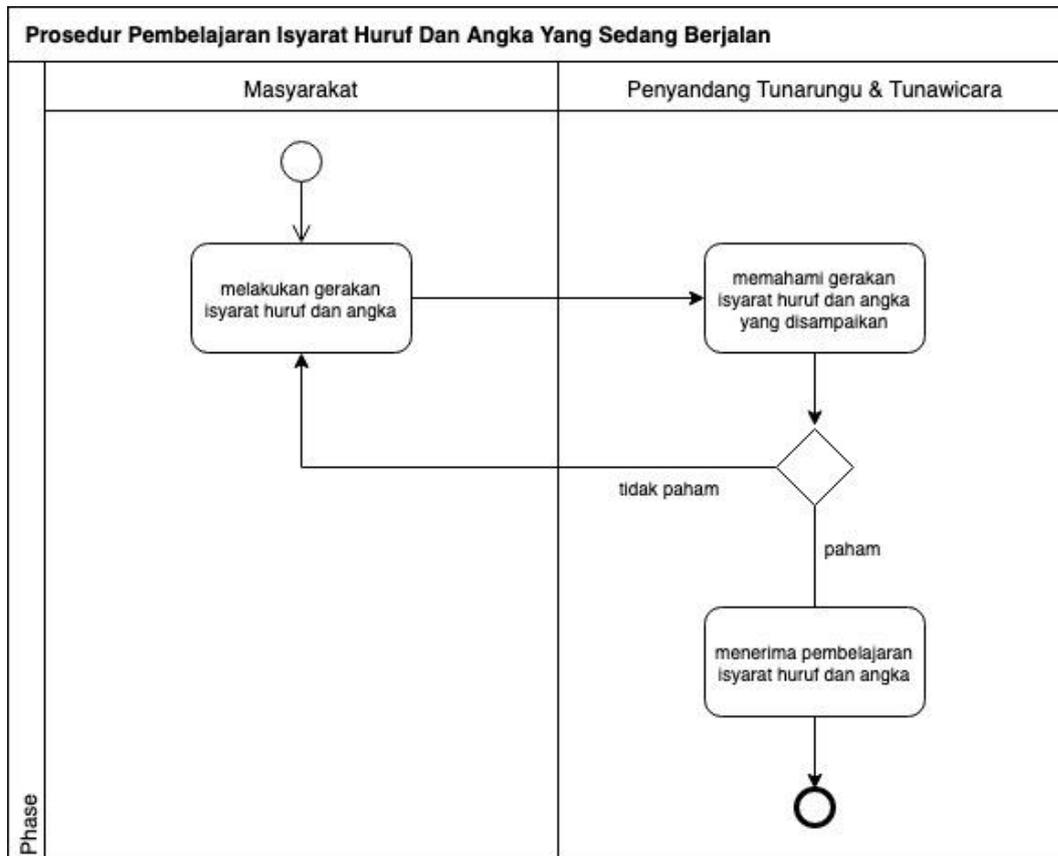
Sebelum memasuki tahapan perancangan sistem, dilakukan tahapan analisis sistem terlebih dahulu yang bertujuan untuk mempelajari prosedur yang sedang berjalan saat ini dan kebutuhan dari user nantinya. Dalam analisis sistem dilakukan penguraian dari suatu sistem informasi yang utuh menjadi bagian-bagian yang lebih terperinci dengan maksud agar proses evaluasi dan identifikasi masalah dapat lebih terarah dan sesuai dengan kebutuhan yang ada.

3.1.1. Analisis Masalah

Analisis masalah adalah tahapan mengumpulkan dan menentukan masalah yang merupakan ide dari pembangunan perangkat lunak. Analisis masalah yang ada yaitu bagaimana membangun sebuah alat penerjemah huruf dan angka bahasa indonesia yang dapat membantu proses pembelajaran penyandang tunarungu dan tunawicara.

3.1.2. Analisis Prosedur Yang Sedang Berjalan

Analisis sistem yang sedang berjalan dilakukan dengan metode observasi untuk mendapatkan gambaran tentang sistem yang sedang berjalan saat ini. Berikut adalah prosedur pembelajaran huruf dan angka terhadap penyandang tunarungu dan tunawicara yang sedang berjalan saat ini, dapat dilihat pada **Gambar 3.1**.



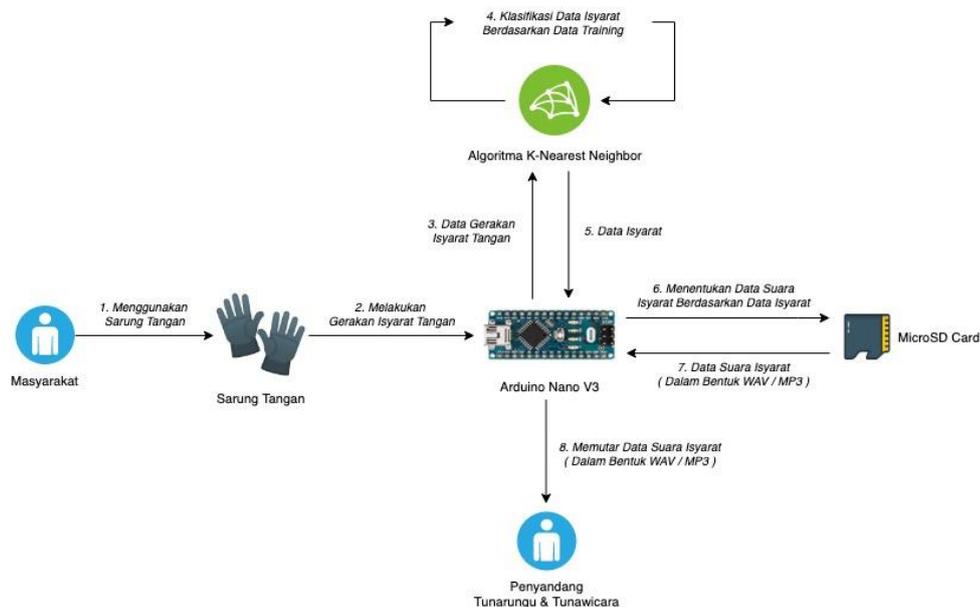
Gambar 3.1 Prosedur Yang Sedang Berjalan

Berikut adalah penjelasan terkait **Gambar 3.1** mengenai prosedur pembelajaran isyarat huruf dan angka yang sedang berjalan.

1. Masyarakat melakukan gerakan isyarat huruf dan angka.
2. Jika penyandang tunarungu dan tunawicara tidak memahami gerakan isyarat tangan yang disampaikan oleh masyarakat maka masyarakat akan mengulang kembali gerakan isyarat tangan sebelumnya.
3. Jika penyandang tunarungu dan tunawicara memahami gerakan isyarat tangan yang disampaikan oleh masyarakat, maka penyandang tunarungu dan tunawicara menerima pembelajaran isyarat huruf maupun angka.

3.1.3. Arsitektur Sistem

Analisis arsitektur sistem bertujuan untuk mengidentifikasi arsitektur yang akan dibangun dan menjelaskan gambaran sistem secara keseluruhan agar bisa terlihat secara jelas bagaimana sistem berjalan. Berikut merupakan arsitektur yang akan dibangun dalam penelitian ini yang dapat dilihat pada **Gambar 3.2**.



Gambar 3.2 Arsitektur Sistem

Berikut adalah penjelasan terkait **Gambar 3.2** mengenai arsitektur sistem:

1. Masyarakat menggunakan sarung tangan.
2. Masyarakat melakukan gerakan isyarat tangan terhadap perangkat Arduino Nano V3.
3. Arduino Nano V3 mengirim data gerakan isyarat tangan yang dilakukan oleh Masyarakat terhadap Algoritma *K-Nearest Neighbors*.
4. Algoritma *K-Nearest Neighbors* melakukan klasifikasi terhadap data gerakan isyarat berdasarkan *data training*.
5. Dari hasil proses klasifikasi Algoritma *K-Nearest Neighbors*, didapatkan hasil berupa data isyarat.
6. Arduino Nano V3 menentukan data suara isyarat pada MicroSD Card berdasarkan data isyarat yang sudah didapatkan dari proses klasifikasi Algoritma *K-Nearest Neighbors*.
7. Setelah data suara isyarat ditentukan, maka Arduino akan memanggil data suara tersebut dari MicroSD Card.
8. Arduino memutar data suara isyarat (dalam bentuk wav ataupun mp3) untuk diperdengarkan kepada penyandang tunarungu dan tunawicara.

3.1.4. Analisis Tekukan Jari Tangan

Analisis tekukan jari tangan yaitu analisis untuk mengetahui nilai dari setiap tekukan jari tangan yang dilakukan oleh pengguna yaitu masyarakat. Nilai dari tekukan jari tangan ini ditentukan berdasarkan nilai dari derajat kelengkungan *flex sensor* yang dapat dilihat pada Gambar 2.12, dimana *flex sensor* terbagi menjadi 5 lengkungan yaitu keadaan lengkungan 0° , 45° , 180° , 135° dan 180° . Gerakan dari tekukan jari tangan tersebut dapat dilihat pada Tabel 3.1.

Tabel 3.1 Derajat Tekukan Jari Tangan

No	Gambar	Keterangan
1		Tekukan Jari Tangan 0°
2		Tekukan Jari Tangan 45°
3		Tekukan Jari Tangan 90°
4		Tekukan Jari Tangan 135°
5		Tekukan Jari Tangan 180°

Berdasarkan dari nilai sudut tiap tekukan jari tangan tersebut, maka diperoleh data tekukan jari tangan dari masing-masing gerakan isyarat huruf dan angka berdasarkan Lampiran A, data tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2 Nilai Tekukan Jari Tangan Berdasarkan Isyarat Huruf Dan Angka

Gambar	Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Isyarat
	1	2	3	4	5	1	2	3	4	5	
	0	0	180	180	180	0	0	180	180	180	A
	180	0	180	180	180	180	0	0	0	180	B
	180	180	180	180	180	90	90	180	180	180	C
	180	0	180	180	180	90	90	180	180	180	D
	180	0	180	180	180	90	90	180	180	180	E
	180	0	180	180	180	180	0	0	180	180	F
	180	180	180	180	180	180	180	180	180	180	G
	180	0	180	180	180	180	0	0	180	180	H
	180	180	180	180	180	180	180	180	180	0	I
	180	180	180	180	180	180	180	180	180	0	J
	180	0	180	180	180	180	135	180	180	180	K
	180	180	180	180	180	0	0	180	180	180	L
	0	0	0	0	0	180	0	0	0	180	M

Gambar	Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Isyarat
	1	2	3	4	5	1	2	3	4	5	
	0	0	0	0	0	180	0	0	180	180	N
	180	180	180	180	180	90	90	0	0	0	O
	180	0	180	180	180	0	0	180	180	180	P
	90	90	0	0	0	180	90	180	180	180	Q
	180	180	180	180	180	0	45	90	45	45	R
	90	90	180	180	180	90	90	180	180	180	S
	180	0	180	180	180	180	0	180	180	180	T
	180	180	180	180	180	0	0	180	180	180	U
	180	180	180	180	180	180	0	0	180	180	V
	0	0	180	180	180	0	0	180	180	180	W
	180	0	180	180	180	180	0	180	180	180	X
	180	0	180	180	180	0	0	180	180	180	Y
	180	180	180	180	180	45	45	45	45	45	Z
	90	90	0	0	0	180	180	180	180	180	0

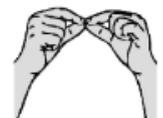
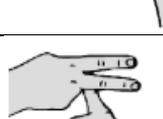
Gambar	Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Isyarat
	1	2	3	4	5	1	2	3	4	5	
 1	180	0	180	180	180	180	180	180	180	180	1
 2	180	0	0	180	180	180	180	180	180	180	2
 3	180	0	0	0	180	180	180	180	180	180	3
 4	180	0	0	0	0	180	180	180	180	180	4
 5	0	0	0	0	0	180	180	180	180	180	5
 6	0	0	0	0	0	0	180	180	180	180	6
 7	0	0	0	0	0	0	0	180	180	180	7
 8	0	0	0	0	0	0	0	0	180	180	8
 9	0	0	0	0	0	0	0	0	0	180	9
 10	0	0	0	0	0	0	0	0	0	0	10

3.1.5. Analisis Kemiringan Tangan

Analisis kemiringan tangan yaitu analisis untuk mengetahui nilai dari setiap gerakan kemiringan tangan yang dilakukan oleh pengguna yaitu masyarakat saat melakukan gerakan isyarat. Nilai kemiringan tangan tersebut diperoleh dari *accelerometer sensor* dimana nilai sensor tersebut memiliki 3 buah nilai diantaranya nilai kemiringan sumbu x, nilai kemiringan sumbu y dan nilai kemiringan sumbu z. Untuk mengetahui nilai pada *accelerometer sensor* perlu dilakukannya pembacaan nilai terlebih dahulu menggunakan mikrokontroler

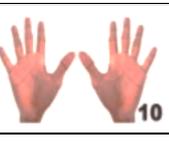
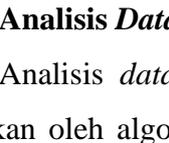
Arduino. Berikut data kemiringan tangan yang didapatkan dari hasil pembacaan nilai pada *accelerometer sensor* berdasarkan gerakan isyarat huruf dan angka, yang dapat dilihat pada Tabel 3.3.

Tabel 3.3 Nilai Kemiringan Tangan Berdasarkan Isyarat Huruf Dan Angka

Gambar	Nilai Kemiringan Tangan			Isyarat
	Sumbu X	Sumbu Y	Sumbu Z	
	0,61	0,10	-0,45	A
	0,09	1,51	0,63	B
	0,61	0,10	-0,45	C
	0,61	0,10	-0,45	D
	0,61	0,10	-0,45	E
	0,09	1,51	0,63	F
	0,09	1,51	0,63	G
	0,09	1,51	0,63	H

Gambar	Nilai Kemiringan Tangan			Isyarat
	Sumbu X	Sumbu Y	Sumbu Z	
	0,61	0,10	-0,45	I
	0,09	1,51	0,63	J
	0,68	0,37	-0,45	K
	0,61	0,10	-0,45	L
	0,96	-0,12	0,48	M
	0,96	-0,12	0,48	N
	0,61	0,10	-0,45	O
	0,61	0,10	-0,45	P
	0,84	0,33	-0,19	Q
	0,25	-0,27	0,83	R

Gambar	Nilai Kemiringan Tangan			Isyarat
	Sumbu X	Sumbu Y	Sumbu Z	
	0,49	-0,09	-0,23	S
	0,08	0,54	-0,44	T
	0,61	0,10	-0,45	U
	0,61	0,10	-0,45	V
	0,61	0,10	-0,45	W
	0,84	0,33	-0,19	X
	0,21	0,13	0,57	Y
	0,08	0,54	-0,44	Z
	0,61	0,10	-0,45	O
	0,61	0,10	-0,45	1

Gambar	Nilai Kemiringan Tangan			Isyarat
	Sumbu X	Sumbu Y	Sumbu Z	
 2	0,61	0,10	-0,45	2
 3	0,61	0,10	-0,45	3
 4	0,61	0,10	-0,45	4
 5	0,61	0,10	-0,45	5
 6	0,61	0,10	-0,45	6
 7	0,61	0,10	-0,45	7
 8	0,61	0,10	-0,45	8
 9	0,61	0,10	-0,45	9
 10	0,61	0,10	-0,45	10

3.1.6. Analisis Data Training

Analisis *data training* yaitu analisis untuk mengetahui data yang akan digunakan oleh algoritma klasifikasi dalam memprediksi suatu *class* dari setiap *instance*. setiap *instance* akan memiliki *attribute* dan *class*. *attribute* adalah keterangan yang dikandung dalam data itu sendiri, setiap data bisa memiliki lebih dari satu *attribute*. biasanya *attribute* menggunakan variabel diskrit. *attribute* yang

akan digunakan dalam *data training* ini yaitu *attribute* dari hasil analisis tekukan jari tangan dan analisis kemiringan tangan. *Data training* ini menggunakan delapan *attribute* yang didapatkan pada analisis tekukan jari tangan dan analisis posisi kemiringan tangan, berikut adalah *attribute* yang akan digunakan dalam penelitian ini dapat dilihat pada Tabel 3.4.

Tabel 3.4 Attribute Data Instance

<i>Attribute</i>	Keterangan
X1	Nilai tekukan ibu jari pada tangan kiri
X2	Nilai tekukan jari telunjuk pada tangan kiri
X3	Nilai tekukan jari tengah pada tangan kiri
X4	Nilai tekukan jari manis pada tangan kiri
X5	Nilai tekukan jari kelingking pada tangan kiri
X6	Nilai tekukan ibu jari pada tangan kanan
X7	Nilai tekukan jari telunjuk pada tangan kanan
X8	Nilai tekukan jari tengah pada tangan kanan
X9	Nilai tekukan jari manis pada tangan kanan
X10	Nilai tekukan jari kelingking pada tangan kanan
X11	Nilai posisi tangan pada sumbu x
X12	Nilai posisi tangan pada sumbu y
X13	Nilai Posisi tangan pada sumbu z

Setiap melakukan gerakan isyarat secara berulang akan menghasilkan kemungkinan nilai yang berbeda-beda akan ada saatnya nilai dari gerakan isyarat tersebut berbeda dengan nilai yang telah didapatkan pada analisis sebelumnya yaitu analisis tekukan jari tangan, oleh karena itu perlunya pemberian nilai toleransi terhadap nilai data training tepatnya pada nilai tekukan jari tangan, pemberian nilai toleransi ini dilakukan dengan cara melakukan duplikasi *data training* dari masing-masing gerakan isyarat sebanyak 3 kali, lalu data tersebut diberi nilai tambah maupun kurang sebanyak 5 derajat. Berikut ini adalah *data training* dari masing-masing gerakan isyarat yang diperoleh dapat dilihat pada Tabel 3.5.

Tabel 3.5 Data Training Gerakan Isyarat Huruf Dan Angka

<i>Instances</i>													<i>Classes</i>
Nilai Tekukan Jari Pada Tangan Kiri				Nilai Tekukan Jari Pada Tangan Kanan					Posisi Kemiringan Tangan				
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	
0	0	180	180	180	0	0	180	180	180	0,61	0,10	-0,45	A
5	5	175	175	175	5	5	175	175	175	0,61	0,10	-0,45	A
10	10	170	170	170	10	10	170	170	170	0,61	0,10	-0,45	A
180	0	180	180	180	180	0	0	0	180	0,09	1,51	0,63	B
175	5	175	175	175	175	5	5	5	175	0,09	1,51	0,63	B
170	10	170	170	170	170	10	10	10	170	0,09	1,51	0,63	B
180	180	180	180	180	90	90	180	180	180	0,61	0,10	-0,45	C

<i>Instances</i>													<i>Classes</i>
Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Posisi Kemiringan Tangan			
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	
175	175	175	175	175	85	85	175	175	175	0,61	0,10	-0,45	C
170	170	170	170	170	95	95	170	170	170	0,61	0,10	-0,45	C
180	0	180	180	180	90	90	180	180	180	0,61	0,10	-0,45	D
175	5	175	175	175	85	85	175	175	175	0,61	0,10	-0,45	D
170	10	170	170	170	95	95	170	170	170	0,61	0,10	-0,45	D
180	0	180	180	180	90	90	180	180	180	0,61	0,10	-0,45	E
175	5	175	175	175	85	85	175	175	175	0,61	0,10	-0,45	E
170	10	170	170	170	95	95	170	170	170	0,61	0,10	-0,45	E
180	0	180	180	180	180	0	0	180	180	0,09	1,51	0,63	F
175	0	175	175	175	175	5	5	175	175	0,09	1,51	0,63	F
170	0	170	170	170	170	10	10	170	170	0,09	1,51	0,63	F
180	180	180	180	180	180	180	180	180	180	0,09	1,51	0,63	G
175	175	175	175	175	175	175	175	175	175	0,09	1,51	0,63	G
170	170	170	170	170	170	170	170	170	170	0,09	1,51	0,63	G
180	0	180	180	180	180	0	0	180	180	0,09	1,51	0,63	H
175	5	175	175	175	175	5	5	175	175	0,09	1,51	0,63	H
170	10	170	170	170	170	10	10	170	170	0,09	1,51	0,63	H
180	180	180	180	180	180	180	180	180	0	0,61	0,10	-0,45	I
175	175	175	175	175	175	175	175	175	5	0,61	0,10	-0,45	I
170	170	170	170	170	170	170	170	170	10	0,61	0,10	-0,45	I
180	180	180	180	180	180	180	180	180	0	0,09	1,51	0,63	J
175	175	175	175	175	175	175	175	175	5	0,09	1,51	0,63	J
170	170	170	170	170	170	170	170	170	10	0,09	1,51	0,63	J
180	0	180	180	180	180	135	180	180	180	0,68	0,37	-0,45	K
175	0	175	175	175	175	140	175	175	175	0,68	0,37	-0,45	K
170	0	170	170	170	170	130	170	170	170	0,68	0,37	-0,45	K
180	180	180	180	180	0	0	180	180	180	0,61	0,10	-0,45	L
175	175	175	175	175	5	5	175	175	175	0,61	0,10	-0,45	L
170	170	170	170	170	10	10	170	170	170	0,61	0,10	-0,45	L
0	0	0	0	0	180	0	0	0	180	0,96	-0,12	0,48	M
5	5	5	5	5	175	5	5	5	175	0,96	-0,12	0,48	M
10	10	10	10	10	170	10	10	10	170	0,96	-0,12	0,48	M
0	0	0	0	0	180	0	0	180	180	0,96	-0,12	0,48	N
5	5	5	5	5	175	5	5	175	175	0,96	-0,12	0,48	N
10	10	10	10	10	170	10	10	170	170	0,96	-0,12	0,48	N
180	180	180	180	180	90	90	0	0	0	0,61	0,10	-0,45	O
175	175	175	175	175	95	95	5	5	5	0,61	0,10	-0,45	O
170	170	170	170	170	85	85	10	10	10	0,61	0,10	-0,45	O
180	0	180	180	180	0	0	180	180	180	0,61	0,10	-0,45	P
175	5	175	175	175	5	5	175	175	175	0,61	0,10	-0,45	P
170	10	170	170	170	10	10	170	170	170	0,61	0,10	-0,45	P
90	90	0	0	0	180	90	180	180	180	0,84	0,33	-0,19	Q
95	95	5	5	5	175	95	175	175	175	0,84	0,33	-0,19	Q
85	85	10	10	10	170	85	170	170	170	0,84	0,33	-0,19	Q
180	180	180	180	180	0	45	90	45	45	0,25	-0,27	0,83	R
175	175	175	175	175	0	50	95	50	50	0,25	-0,27	0,83	R
170	170	170	170	170	0	40	85	40	40	0,25	-0,27	0,83	R
90	90	180	180	180	90	90	180	180	180	0,49	-0,09	-0,23	S
95	95	175	175	175	95	95	175	175	175	0,49	-0,09	-0,23	S

<i>Instances</i>													<i>Classes</i>
Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Posisi Kemiringan Tangan			
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	
85	85	170	170	170	85	85	170	170	170	0,49	-0,09	-0,23	S
180	0	180	180	180	180	0	180	180	180	0,08	0,54	-0,44	T
175	5	175	175	175	175	5	175	175	175	0,08	0,54	-0,44	T
170	10	170	170	170	170	10	170	170	170	0,08	0,54	-0,44	T
180	180	180	180	180	0	0	180	180	180	0,61	0,10	-0,45	U
175	175	175	175	175	5	5	175	175	175	0,61	0,10	-0,45	U
170	170	170	170	170	10	10	170	170	170	0,61	0,10	-0,45	U
180	180	180	180	180	180	0	0	180	180	0,61	0,10	-0,45	V
175	175	175	175	175	175	5	5	175	175	0,61	0,10	-0,45	V
170	170	170	170	170	170	10	10	170	170	0,61	0,10	-0,45	V
0	0	180	180	180	0	0	180	180	180	0,61	0,10	-0,45	W
5	5	175	175	175	5	5	175	175	175	0,61	0,10	-0,45	W
10	10	170	170	170	10	10	170	170	170	0,61	0,10	-0,45	W
180	0	180	180	180	180	0	180	180	180	0,84	0,33	-0,19	X
175	5	175	175	175	180	5	175	175	175	0,84	0,33	-0,19	X
170	10	170	170	170	180	10	170	170	170	0,84	0,33	-0,19	X
180	0	180	180	180	0	0	180	180	180	0,21	0,13	0,57	Y
175	5	175	175	175	5	5	175	175	175	0,21	0,13	0,57	Y
170	10	170	170	170	10	10	170	170	170	0,21	0,13	0,57	Y
180	0	180	180	180	45	45	45	45	45	0,08	0,54	-0,44	Z
175	5	175	175	175	50	50	50	50	50	0,08	0,54	-0,44	Z
170	10	170	170	170	40	40	40	40	40	0,08	0,54	-0,44	Z
180	0	180	180	180	180	0	180	180	180	0,61	0,10	-0,45	1
175	5	175	175	175	175	5	175	175	175	0,61	0,10	-0,45	1
170	10	170	170	170	170	10	170	170	170	0,61	0,10	-0,45	1
180	0	0	180	180	180	0	180	180	180	0,61	0,10	-0,45	2
175	5	5	175	175	175	5	175	175	175	0,61	0,10	-0,45	2
170	10	10	170	170	170	10	170	170	170	0,61	0,10	-0,45	2
180	0	0	0	180	180	0	180	180	180	0,61	0,10	-0,45	3
175	5	5	5	175	175	5	175	175	175	0,61	0,10	-0,45	3
170	10	10	10	170	170	10	170	170	170	0,61	0,10	-0,45	3
180	0	0	0	0	180	0	180	180	180	0,61	0,10	-0,45	4
175	5	5	5	5	175	5	175	175	175	0,61	0,10	-0,45	4
170	10	10	10	10	170	10	170	170	170	0,61	0,10	-0,45	4
0	0	0	0	0	180	0	180	180	180	0,61	0,10	-0,45	5
5	5	5	5	5	175	5	175	175	175	0,61	0,10	-0,45	5
10	10	10	10	10	170	10	170	170	170	0,61	0,10	-0,45	5
0	0	0	0	0	0	180	180	180	180	0,61	0,10	-0,45	6
5	5	5	5	5	5	175	175	175	175	0,61	0,10	-0,45	6
10	10	10	10	10	10	170	170	170	170	0,61	0,10	-0,45	6
0	0	0	0	0	0	0	180	180	180	0,61	0,10	-0,45	7
5	5	5	5	5	5	5	175	175	175	0,61	0,10	-0,45	7
10	10	10	10	10	10	10	170	170	170	0,61	0,10	-0,45	7
0	0	0	0	0	0	0	0	180	180	0,61	0,10	-0,45	8
5	5	5	5	5	5	5	5	175	175	0,61	0,10	-0,45	8
10	10	10	10	10	10	10	10	170	170	0,61	0,10	-0,45	8
0	0	0	0	0	0	0	0	0	180	0,61	0,10	-0,45	9
5	5	5	5	5	5	5	5	5	175	0,61	0,10	-0,45	9
10	10	10	10	10	10	10	10	10	170	0,61	0,10	-0,45	9

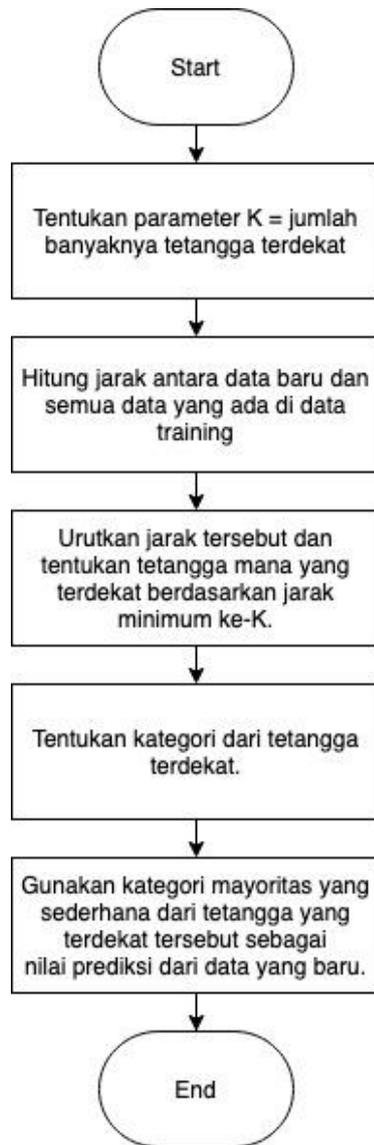
<i>Instances</i>													<i>Classes</i>
Nilai Tekukan Jari Pada Tangan Kiri					Nilai Tekukan Jari Pada Tangan Kanan					Posisi Kemiringan Tangan			
X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	
0	0	0	0	0	0	0	0	0	0	0,61	0,10	-0,45	10
5	5	5	5	5	5	5	5	5	5	0,61	0,10	-0,45	10
10	10	10	10	10	10	10	10	10	10	0,61	0,10	-0,45	10

3.1.7. Analisis Penyimpanan *Data Training*

Analisis penyimpanan *data training* adalah analisis untuk mengetahui bagaimana *data training* yang sudah didapatkan pada analisis sebelumnya yaitu analisis *data training* tersebut disimpan. Dalam penelitian ini data training disimpan di dalam *array*, dimana terdapat dua buah *array*, yaitu *array instances* dan *array classes*. *Array instances* digunakan untuk menyimpan data *attribute* gerakan tangan yang terdiri dari nilai tekukan jari tangan dan nilai posisi kemiringan tangan, sedangkan *array classes* digunakan untuk menyimpan data hasil klasifikasi dari *array instance*. Data *array* tersebut nantinya akan disimpan dalam sebuah *file* dan di *compile* bersamaan dengan *syntax program*.

3.1.8. Analisis Klasifikasi Gerakan Isyarat

Analisis Klasifikasi Gerakan Isyarat adalah analisis untuk mengklasifikasikan data baru (*data test*) yaitu data dari gerakan isyarat tangan yang dilakukan oleh penyandang tunarungu dan tunawicara dengan *data training* yang sudah ditentukan pada analisis *data training*. Dalam pengklasifikasian gerakan isyarat ini menggunakan Algoritma *K-Nearest Neighbors* (KNN). Langkah-langkah pengklasifikasian gerakan isyarat menggunakan Algoritma *K-Nearest Neighbors* (KNN) dapat dilihat pada **Gambar 3.3**.



Gambar 3.3 Flowchart Algoritma *K-Nearest Neighbors*

Adapun penjelasan dari langkah-langkah Algoritma *K-Nearest Neighbors*

(KNN) sebagai berikut :

1. Tentukan parameter K = jumlah banyaknya tetangga terdekat
2. Hitung jarak antara data baru dan semua data yang ada di data training.
3. Urutkan jarak tersebut dan tentukan tetangga mana yang terdekat berdasarkan jarak minimum ke- K .
4. Tentukan kategori dari tetangga terdekat.
5. Gunakan kategori mayoritas yang sederhana dari tetangga yang terdekat tersebut sebagai nilai prediksi dari data yang baru.

Berikut adalah Algoritma K-Nearest Neighbors (KNN) yang dinyatakan dalam notasi pseudo-code sebagai berikut :

```

Procedure ClassifyKNN(data : float, classesNum[] : int, d[] : float, k : int, rows : int, columns : int)
{ Klasifikasi Gerakan Isyarat Huruf dan Angka
  Masukan : data {data training}, classesNum, d[] {data test}, k, rows, columns
  Keluaran : best position classes ( Nilai Yang Sering Muncul Terdekat Dengan Data Training)
}

Deklarasi
temp2[rows] : float { data distance sorted for extracting the first k}
firstK[k] : float {first k near to data}
classes[rows] : int { temporal copy of classesNum used for sort temp2}
kClasses[k] : int {first k near to data (the output classes)}

Algoritma
{ Langkah 1 }
k ← 3

{ Langkah 2 }
{ calculate the euclidean distance of d from the database }
i ← 0
j ← 0;
pt2[columns] : float

for ← i < rows do
  for ← j < columns do
    pt2[j] = instances[i * columns + j];
  end for
  temp[i] ← euclideanDistance(pt1, pt2, columns)
end for

{ Langkah 3 }
{ sort distances Ascending }
for ← i <= rows do
  f = 0;
  for (j = 0; j < (rows - 1); j++) {
    if (data[j + 1] < data[j]) {
      {descending order >, ascending order}
      temp[0] ← data[j]
      temp[1] ← classes[j]

      data[j] ← data[j + 1]
      classes[j] ← classes[j + 1]
      data[j + 1] ← temp[0]
      classes[j + 1] ← temp[1]
      f ← 1
    }
  }
end for
end for

{ Langkah 4 }
{ calculate the k near }
best ← temp[0]
bestPosition ← 0

```

```

for cont < k do
  if temp[cont] > best then
    bestPosition ← cont
    best ← temp[cont]
  end if
end for

{ Langkah 5 }
return classes[bestPosition]

```

3.1.9. Analisis Penentuan Data Suara Isyarat

Analisis penentuan data suara isyarat adalah analisis untuk mengetahui bagaimana menentukan data suara isyarat berdasarkan gerakan isyarat tangan yang dilakukan oleh pengguna yaitu masyarakat. Setiap gerakan isyarat tangan memiliki data suara isyarat tersendiri, data isyarat tersebut berupa *file* WAV ataupun MP3 yang disimpan dalam sebuah MicroSD Card.

Dalam penelitian ini data suara isyarat ditentukan berdasarkan hasil klasifikasi yang didapatkan pada analisis sebelumnya yaitu analisis klasifikasi gerakan isyarat, dari analisis klasifikasi gerakan isyarat tersebut didapatkan data isyarat berupa *string*, jika hasil klasifikasi gerakan isyarat berhasil dan di temukan data yang sesuai maka *file* suara isyarat tersebut akan diputar. Berikut adalah data suara berdasarkan hasil klasifikasi gerakan isyarat huruf dan angka yang dapat dilihat pada Tabel 3.6.

Tabel 3.6 Data Suara Isyarat Huruf Dan Angka

No	Hasil Klasifikasi	Nama File	Keterangan
1	"A"	001-HURUF_A.wav	<i>Output</i> yang dihasilkan adalah suara huruf "A"
2	"B"	002-HURUF_B.wav	<i>Output</i> yang dihasilkan adalah suara huruf "B"
3	"C"	003-HURUF_C.wav	<i>Output</i> yang dihasilkan adalah suara huruf "C"
4	"D"	004-HURUF_D.wav	<i>Output</i> yang dihasilkan adalah suara huruf "D"
5	"E"	005-HURUF_E.wav	<i>Output</i> yang dihasilkan adalah suara huruf "E"
6	"F"	006-HURUF_F.wav	<i>Output</i> yang dihasilkan adalah suara huruf "F"
7	"G"	007-HURUF_G.wav	<i>Output</i> yang dihasilkan adalah suara huruf "G"
8	"H"	008-HURUF_H.wav	<i>Output</i> yang dihasilkan adalah suara huruf "H"

No	Hasil Klasifikasi	Nama File	Keterangan
9	“I”	009-HURUF_I.wav	<i>Output</i> yang dihasilkan adalah suara huruf “I”
10	“J”	010-HURUF_J.wav	<i>Output</i> yang dihasilkan adalah suara huruf “J”
11	“K”	011-HURUF_K.wav	<i>Output</i> yang dihasilkan adalah suara huruf “K”
12	“L”	012-HURUF_L.wav	<i>Output</i> yang dihasilkan adalah suara huruf “L”
13	“M”	013-HURUF_M.wav	<i>Output</i> yang dihasilkan adalah suara huruf “M”
14	“N”	014-HURUF_N.wav	<i>Output</i> yang dihasilkan adalah suara huruf “N”
15	“O”	015-HURUF_O.wav	<i>Output</i> yang dihasilkan adalah suara huruf “O”
16	“P”	016-HURUF_P.wav	<i>Output</i> yang dihasilkan adalah suara huruf “P”
17	“Q”	017-HURUF_Q.wav	<i>Output</i> yang dihasilkan adalah suara huruf “Q”
18	“R”	018-HURUF_R.wav	<i>Output</i> yang dihasilkan adalah suara huruf “R”
19	“S”	019-HURUF_S.wav	<i>Output</i> yang dihasilkan adalah suara huruf “S”
20	“T”	020-HURUF_T.wav	<i>Output</i> yang dihasilkan adalah suara huruf “T”
21	“U”	021-HURUF_U.wav	<i>Output</i> yang dihasilkan adalah suara huruf “U”
22	“V”	022-HURUF_V.wav	<i>Output</i> yang dihasilkan adalah suara huruf “V”
23	“W”	023-HURUF_W.wav	<i>Output</i> yang dihasilkan adalah suara huruf “W”
24	“X”	024-HURUF_X.wav	<i>Output</i> yang dihasilkan adalah suara huruf “X”
25	“Y”	025-HURUF_Y.wav	<i>Output</i> yang dihasilkan adalah suara huruf “Y”
26	“Z”	026-HURUF_Z.wav	<i>Output</i> yang dihasilkan adalah suara huruf “Z”
27	“1”	027-ANGKA_01.mp3	<i>Output</i> yang dihasilkan adalah suara angka “1”
28	“2”	028-ANGKA_02.mp3	<i>Output</i> yang dihasilkan adalah suara angka “2”
29	“3”	029-ANGKA_03.mp3	<i>Output</i> yang dihasilkan adalah suara angka “3”
30	“4”	030-ANGKA_04.mp3	<i>Output</i> yang dihasilkan adalah suara angka “4”
31	“5”	031-ANGKA_05.mp3	<i>Output</i> yang dihasilkan adalah suara angka “5”
32	“6”	032-ANGKA_06.mp3	<i>Output</i> yang dihasilkan adalah suara angka “6”
33	“7”	033-ANGKA_07.mp3	<i>Output</i> yang dihasilkan adalah suara angka “7”
34	“8”	034-ANGKA_08.mp3	<i>Output</i> yang dihasilkan adalah suara angka “8”

No	Hasil Klasifikasi	Nama File	Keterangan
35	“9”	035-ANGKA_09.mp3	<i>Output</i> yang dihasilkan adalah suara angka “9”
36	“10”	036-ANGKA_10.mp3	<i>Output</i> yang dihasilkan adalah suara angka “10”
37	“0”	037-ANGKA_00.mp3	<i>Output</i> yang dihasilkan adalah suara angka “0”

3.1.10. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non-fungsional dilakukan untuk menghasilkan rincian tentang hal-hal yang dilakukan sistem ketika diimplementasikan. Adapun beberapa bagian yang termasuk kedalam analisis kebutuhan Non Fungsional yaitu analisis kebutuhan perangkat keras, analisis kebutuhan perangkat lunak, analisis jaringan, analisis pengguna, dan analisis pengkodean.

3.1.10.1. Analisis Kebutuhan Perangkat Keras

Pada bagian analisis kebutuhan perangkat keras akan diuraikan kebutuhan-kebutuhan non fungsional yang berhubungan dengan perangkat keras. Berikut ini adalah perangkat keras beserta spesifikasi yang diusulkan berdasarkan kebutuhan untuk menjalankan alat penerjemah huruf dan angka, dapat dilihat pada Tabel 3.7.

Tabel 3.7 Analisis Kebutuhan Perangkat Keras

No	Jenis Perangkat Keras	Spesifikasi
1	Mikrokontroler	Arduino Nano V3
2	<i>Flex Sensor</i>	Sparkfun Flex Sensor 2.2”
3	<i>Accelerometer Sensor</i>	ADXL335 Accelerometer (3-Axis)
4	<i>Bluetooth Module</i>	HC-05 Bluetooth
5	<i>Battery Charging Module</i>	TP5100U (Support 2 Cell Battery)
7	<i>MP3 Player Module</i>	YX5300 UART Control Serial MP3
6	<i>Battery</i>	Lithium 18650 (2 Cell)
8	MicroSD Card	SanDisk MicroSD Card Kapasitas 1 GB
9	Speaker	Speaker 0.5W 8 Ω

3.1.10.2. Analisis Kebutuhan Perangkat Lunak

Pada bagian analisis kebutuhan perangkat lunak ini akan diuraikan kebutuhan-kebutuhan non fungsional yang berhubungan dengan spesifikasi perangkat lunak. Berikut ini adalah spesifikasi perangkat lunak yang dibutuhkan untuk menjalankan alat penerjemah huruf dan angka, dapat dilihat pada Tabel 3.8.

Tabel 3.8 Analisis Kebutuhan Perangkat Lunak

No	Jenis Perangkat Lunak	Spesifikasi
1	Arduino Bootloader	ATmega328P Bootloader

3.1.10.3. Analisis Pengguna

Analisis pengguna dimaksudkan untuk mengetahui berbagai *user* yang akan terlibat dalam penggunaan alat yang akan dibangun, sehingga diharapkan *user* yang menggunakan alat dapat memenuhi beberapa. Adapun pengguna yang akan terlibat dalam penggunaan alat dapat dilihat pada Tabel 3.9.

Tabel 3.9 Analisis Pengguna

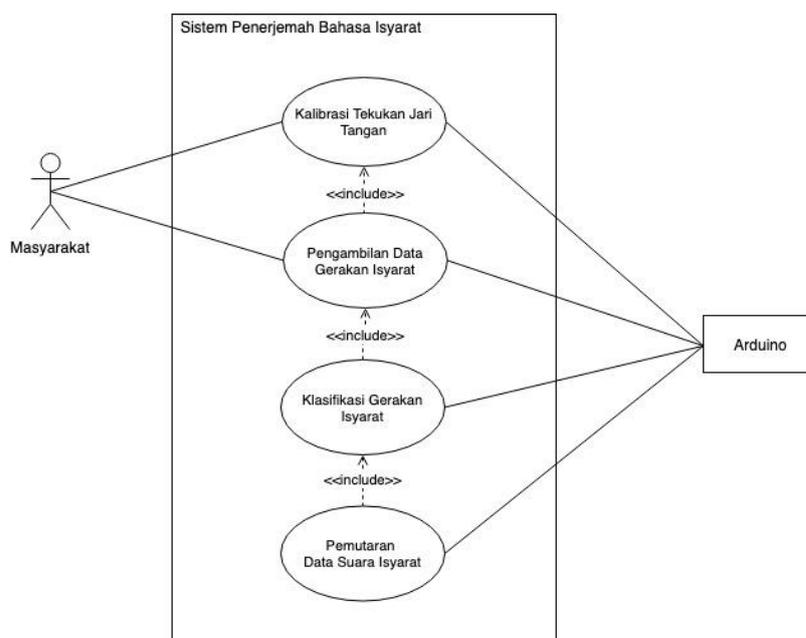
No	Pengguna	Tingkat Keterampilan
1	Masyarakat	Memahami isyarat huruf dan angka BISINDO.

3.1.11. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional merupakan gambaran atau pengaturan dari beberapa elemen terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Alat bantu untuk menggambarkan sistem ini dengan menggunakan *Unified Modelling Language* (UML), dimana tahapan-tahapannya terdiri dari *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*.

3.1.11.1. Use Case Diagram

Use case diagram digunakan untuk mendeskripsikan fungsi dari sebuah sistem sehingga dapat menggambarkan kebutuhan pengguna. Berikut *use case diagram* dapat dilihat pada **Gambar 3.4**.



Gambar 3.4 Use Case Diagram Sistem Alat Penerjemah Huruf Dan Angka

3.1.11.2. Definisi Aktor

Definisi aktor merupakan penjelasan aktor yang terdapat pada *use case diagram*. Berikut definisi aktor dapat dilihat pada Tabel 3.10.

Tabel 3.10 Definisi Aktor

No	Aktor	Deskripsi
1	Masyarakat	Merupakan aktor pada sistem yang akan dibangun dan mempunyai peran sebagai pengguna alat yang dapat melakukan kalibrasi tekukan jari tangan dan pengambilan data gerakan isyarat.
2	Arduino	Merupakan aktor pada sistem yang akan dibangun dan mempunyai peran dalam melakukan kalibrasi tekukan jari tangan, pengambilan data gerakan isyarat, klasifikasi gerakan isyarat dan pemutaran data suara isyarat.

3.1.11.3. Definisi Use Case

Definisi *use case* merupakan penjelasan *use case* yang terdapat pada *use case diagram*. Berikut definisi *use case* dapat dilihat pada Tabel 3.11.

Tabel 3.11 Definisi Use Case

No	Use Case	Deskripsi
1	Kalibrasi Tekukan Jari Tangan	Proses yang dilakukan untuk menentukan nilai <i>minimal</i> dan <i>maximal</i> tekukan jari tangan.
2	Pengambilan Data Gerakan Isyarat	Proses yang dilakukan untuk mengambil nilai dari gerakan isyarat tangan.
3	Klasifikasi Gerakan Isyarat	Proses yang dilakukan untuk menentukan data isyarat tangan.
4	Pemutaran Data Suara Isyarat	Proses yang dilakukan untuk memutar suara isyarat tangan.

3.1.11.4. Skenario Use Case

Skenario *use case* merupakan gambaran dari alur penggunaan sistem. Setiap skenario digambarkan dari sudut pandang aktor atau alat yang berinteraksi langsung dengan sistem. Adapun skenario *use case* pada sistem yang akan dibangun sebagai berikut :

1. Use Case Scenario Kalibrasi Tekukan Jari Tangan

Tabel 3.12 Skenario Kalibrasi Tekukan Jari Tangan

<i>Use Case Name</i>	Kalibrasi Tekukan Jari Tangan	
<i>Goal Context</i>	Mendapatkan range nilai <i>minimal</i> dan nilai <i>maximal</i> dari tekukan jari tangan	
<i>Preconditions</i>	Alat penerjemah huruf dan angka berupa sarung tangan sudah dalam kondisi digunakan dan dinyalakan	
<i>Successful End Condition</i>	Alat akan menyalakan LED dengan warna hijau	
<i>Failed End Condition</i>	Alat akan menyalakan LED dengan warna biru	
<i>Primary Actor</i>	Masyarakat, Arduino	
<i>Trigger</i>	-	
<i>Main Flow</i>	<i>Step</i>	<i>Action</i>

	1	Alat penerjemah huruf dan angka berupa sarung tangan sudah dalam kondisi digunakan dan diyalakan
	2	Masyarakat melakukan gerakan mengepal tangan
	3	Sistem membaca nilai tekukan jari dari <i>Flex Sensor</i>
	4	Sistem menentukan nilai <i>minal</i> dan nilai <i>maximal Flex Sensor</i>
	5	Sistem menyalakan LED dengan warna hijau sebagai tanda kalibrasi selesai

2. Use Case Scenario Pengambilan Data Gerakan Isyarat

Tabel 3.13 Skenario Pengambilan Data Gerakan Isyarat

<i>Use Case Name</i>	Pengambilan Data Gerakan Isyarat	
<i>Goal Context</i>	Mendapatkan nilai tekukan jari dari <i>Flex Sensor</i> dan nilai kemiringan tangan dari <i>Accelerometer Sensor</i>	
<i>Preconditions</i>	Alat penerjemah huruf dan angka berupa sarung tangan sudah melalui proses kalibrasi tekukan jari tangan.	
<i>Succesful End Condition</i>	-	
<i>Failed End Condition</i>	-	
<i>Primary Actor</i>	Masyarakat, Arduino	
<i>Trigger</i>	-	
<i>Main Flow</i>	<i>Step</i>	<i>Action</i>
	1	Masyarakat melakukan gerakan isyarat huruf maupun angka
	2	Sistem membaca nilai tekukan jari tangan melalui <i>Flex Sensor</i>
	3	Sistem membaca nilai kemiringan tangan melalui <i>Accelerometer Sensor</i>
	4	Sistem menyimpan nilai tekukan jari tangan dan kemiringan tangan

3. Use Case Scenario Klasifikasi Gerakan Isyarat

Tabel 3.14 Skenario Klasifikasi Gerakan Isyarat

<i>Use Case Name</i>	Klasifikasi Gerakan Isyarat	
<i>Goal Context</i>	Mendapatkan nilai isyarat berdasarkan hasil klasifikasi gerakan isyarat huruf maupun angka yang dilakukan	
<i>Preconditions</i>	Masyarakat telah melakukan gerakan isyarat huruf maupun angka.	
<i>Succesful End Condition</i>	-	
<i>Failed End Condition</i>	-	
<i>Primary Actor</i>	Arduino	
<i>Trigger</i>	-	
<i>Main Flow</i>	<i>Step</i>	<i>Action</i>
	1	Sistem menentukan nilai K (jumlah tetangga terdekat)
	2	Sistem menghitung jarak <i>data test</i> dengan <i>data training</i>
	3	Sistem mengurutkan data berdasarkan data yang mempunyai jarak <i>Euclidian</i> terkecil
	4	Sistem menentukan hasil <i>data test</i> berdasarkan label mayoritas pada K

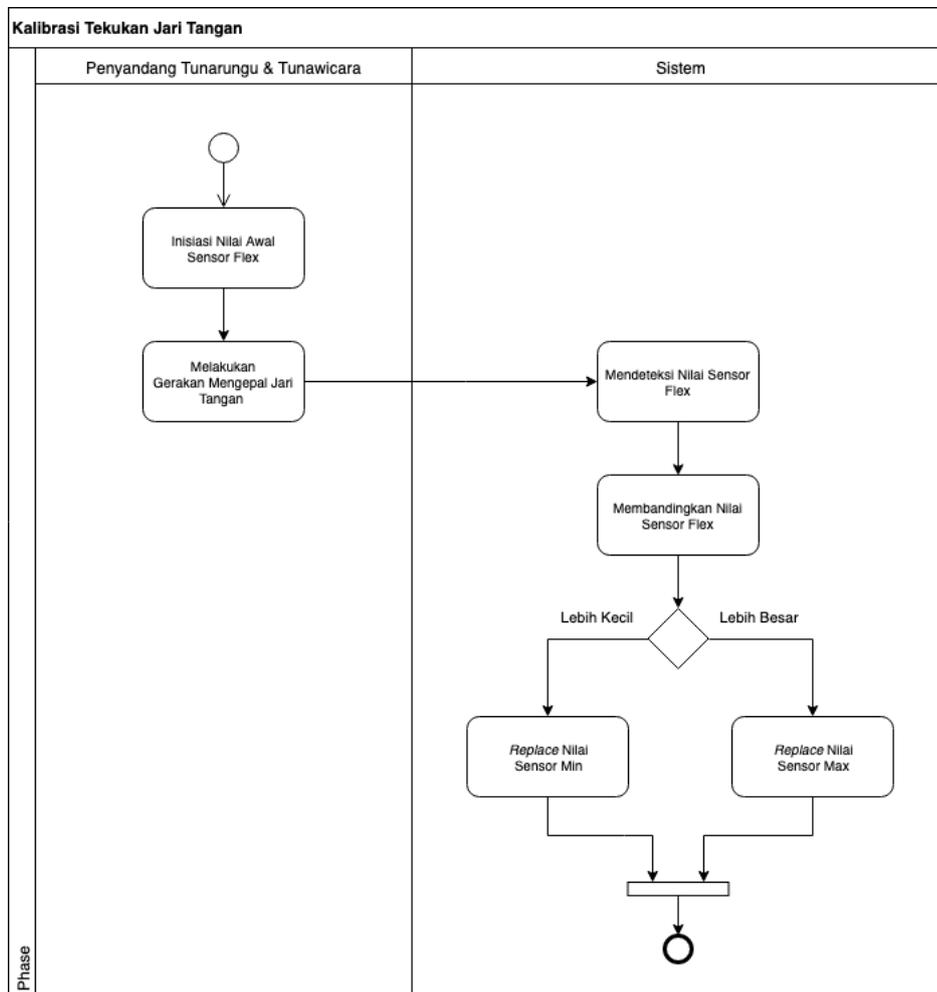
4. *Use Case Scenario* Pemutaran Data Suara Isyarat**Tabel 3.15 Skenario Pemutaran Data Suara Isyarat**

<i>Use Case Name</i>	Pemutaran Data Suara Isyarat	
<i>Goal Context</i>	Memutar data suara isyarat (dalam bentuk wav maupun mp3) sesuai hasil klasifikasi gerakan isyarat	
<i>Preconditions</i>	Sudah mendapatkan data isyarat dari proses klasifikasi gerakan isyarat	
<i>Successful End Condition</i>	Memutar data suara isyarat berdasarkan hasil klasifikasi yang sesuai dengan gerakan yang dilakukan	
<i>Failed End Condition</i>	Memutar data suara isyarat berdasarkan hasil klasifikasi yang tidak sesuai dengan gerakan yang dilakukan	
<i>Primary Actor</i>	Arduino	
<i>Trigger</i>	-	
<i>Main Flow</i>	<i>Step</i>	<i>Action</i>
	1	Sistem mengambil data isyarat dari hasil klasifikasi gerakan isyarat
	2	Sistem melakukan cek hasil data suara isyarat
	3	Sistem memutar data suara isyarat berdasarkan hasil klasifikasi yang sesuai dengan gerakan yang dilakukan
<i>Extension</i>	<i>Step</i>	<i>Branching Action</i>
	3.1	Sistem memutar data suara isyarat berdasarkan hasil klasifikasi yang tidak sesuai dengan gerakan yang dilakukan

3.1.11.5. Activity Diagram

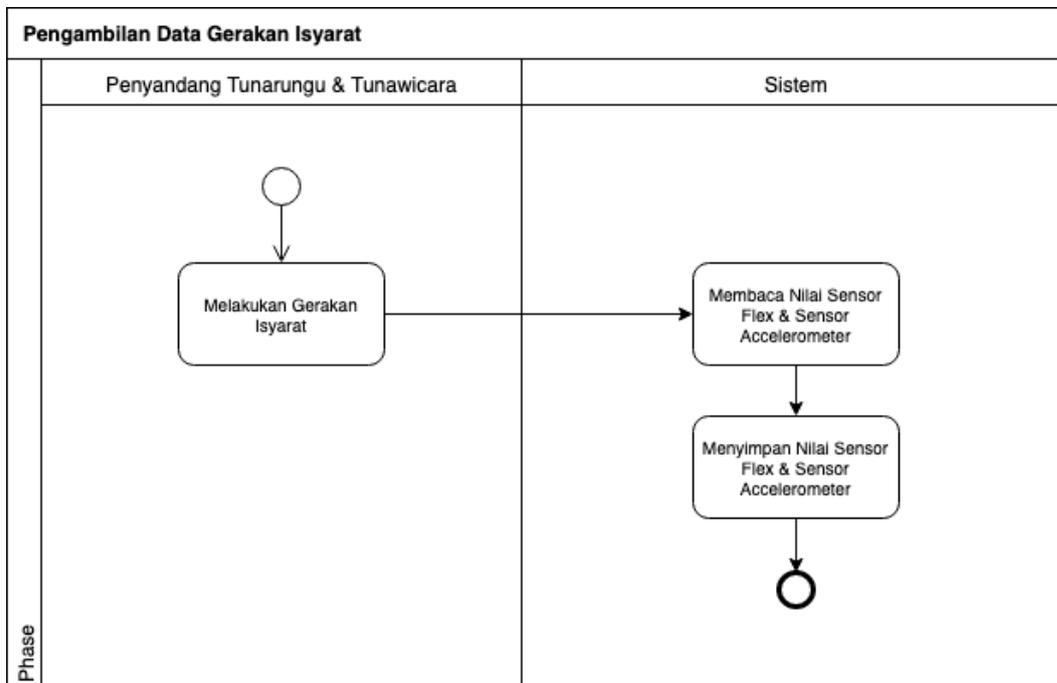
Activity Diagram merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program. *Activity Diagram* memodelkan *event-event* yang terjadi didalam suatu *use case* dan digunakan untuk pemodelan aspek dinamis dari sistem. *Activity diagram* akan digambarkan berdasarkan pengguna.

1. Activity Diagram Kalibrasi Tekukan Jari Tangan



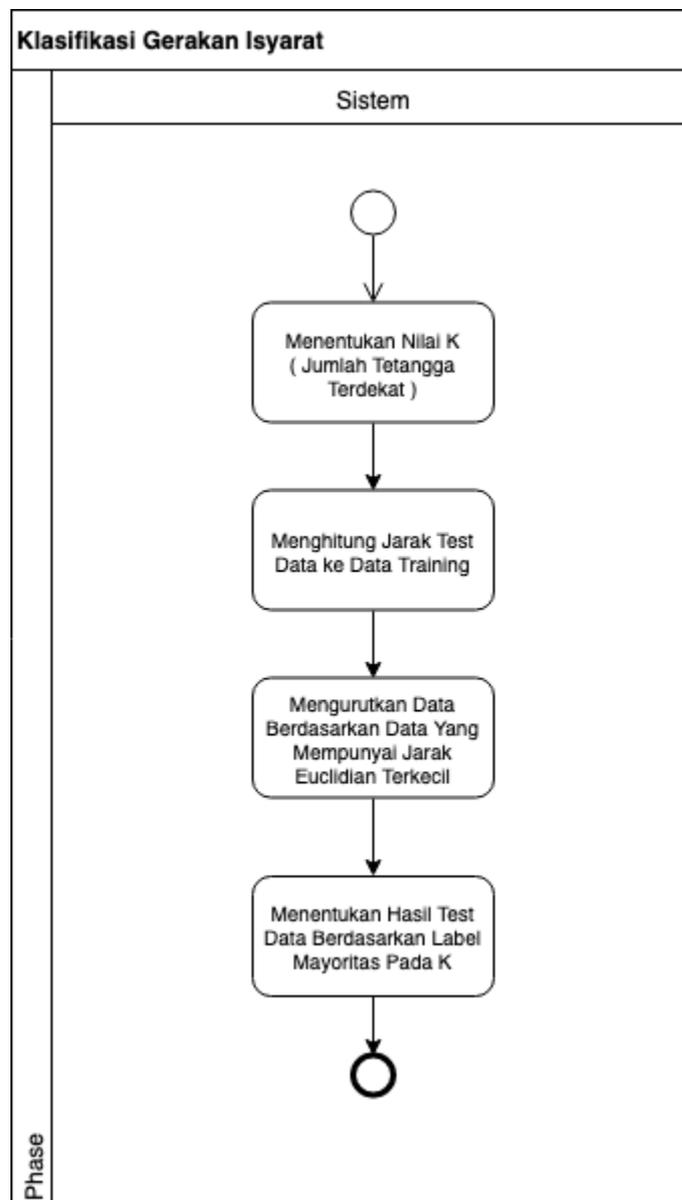
Gambar 3.5 Activity Diagram Kalibrasi Tekukan Jari Tangan

2. Activity Diagram Pengambilan Data Gerakan Isyarat



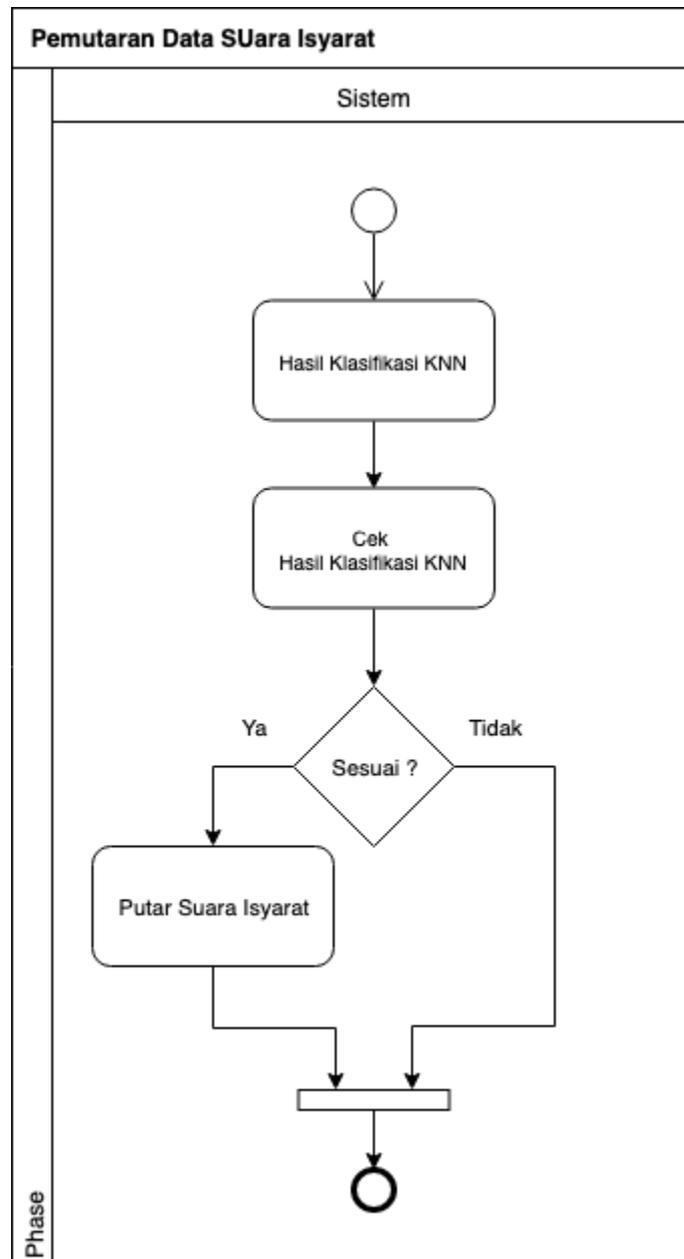
Gambar 3.6 Activity Diagram Pengambilan Data Gerakan Isyarat

3. *Activity Diagram* Klasifikasi Gerakan Isyarat



Gambar 3.7 *Activity Diagram* Klasifikasi Gerakan Isyarat

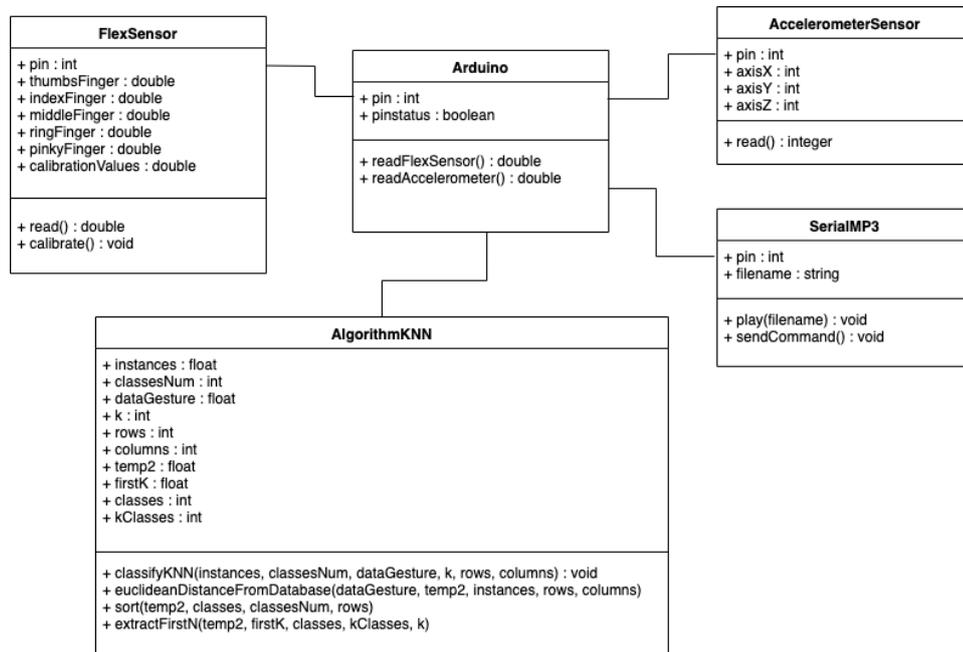
4. Activity Diagram Pemutaran Data Suara Isyarat



Gambar 3.8 Activity Diagram Pemutaran Data Suara Isyarat

3.1.11.6. Class Diagram

Class Diagram digunakan untuk menggambarkan *class-class* yang terlibat dalam analisis sebuah sistem yang akan dibangun. Dengan *class diagram* struktur dan deskripsi *class* serta hubungan antar *class* akan terlihat dengan jelas beserta *method-method* setiap *class*-nya. Adapun *class diagram* dari sistem yang akan dibangun dapat dilihat pada **Gambar 3.9**.

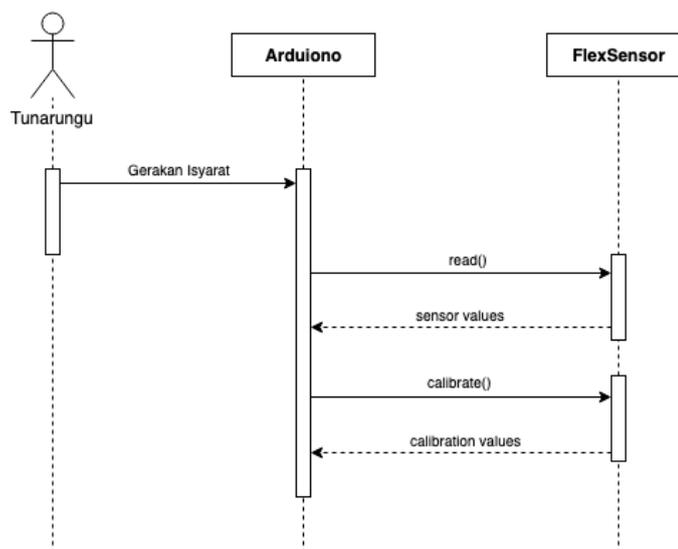


Gambar 3.9 Class Diagram

3.1.11.7. Sequence Diagram

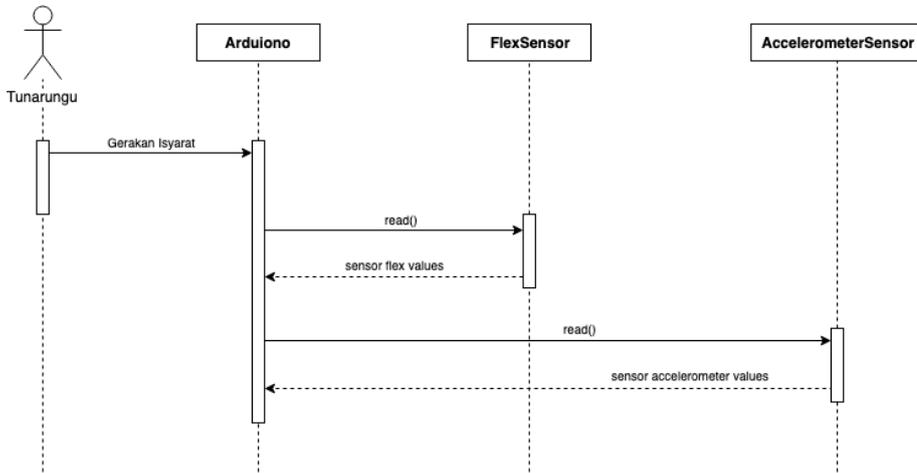
Sequence Diagram menggambarkan interaksi antar masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi.

1. *Sequence Diagram* Kalibrasi Tekukan Jari Tangan



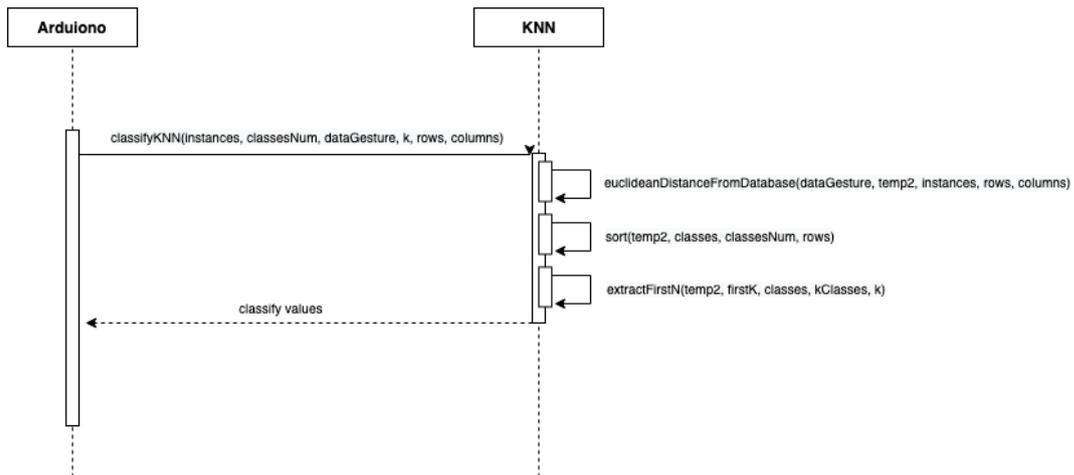
Gambar 3.10 Sequence Diagram Kalibrasi Tekukan Jari Tangan

2. *Sequence Diagram* Pengambilan Data Gerakan Isyarat



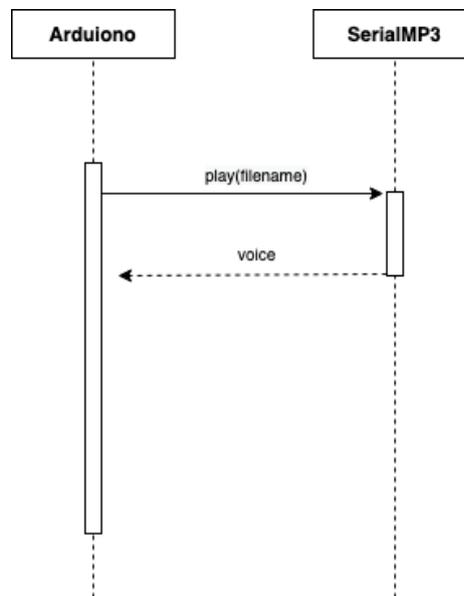
Gambar 3.11 *Sequence Diagram* Pengambilan Data Gerakan Isyarat

3. *Sequence Diagram* Klasifikasi Gerakan Isyarat



Gambar 3.12 *Sequence Diagram* Klasifikasi Gerakan Isyarat

4. *Sequence Diagram* Pemutaran Data Suara Isyarat



Gambar 3.13 *Sequence Diagram* Pemutaran Data Suara Isyarat

3.2. Perancangan Sistem

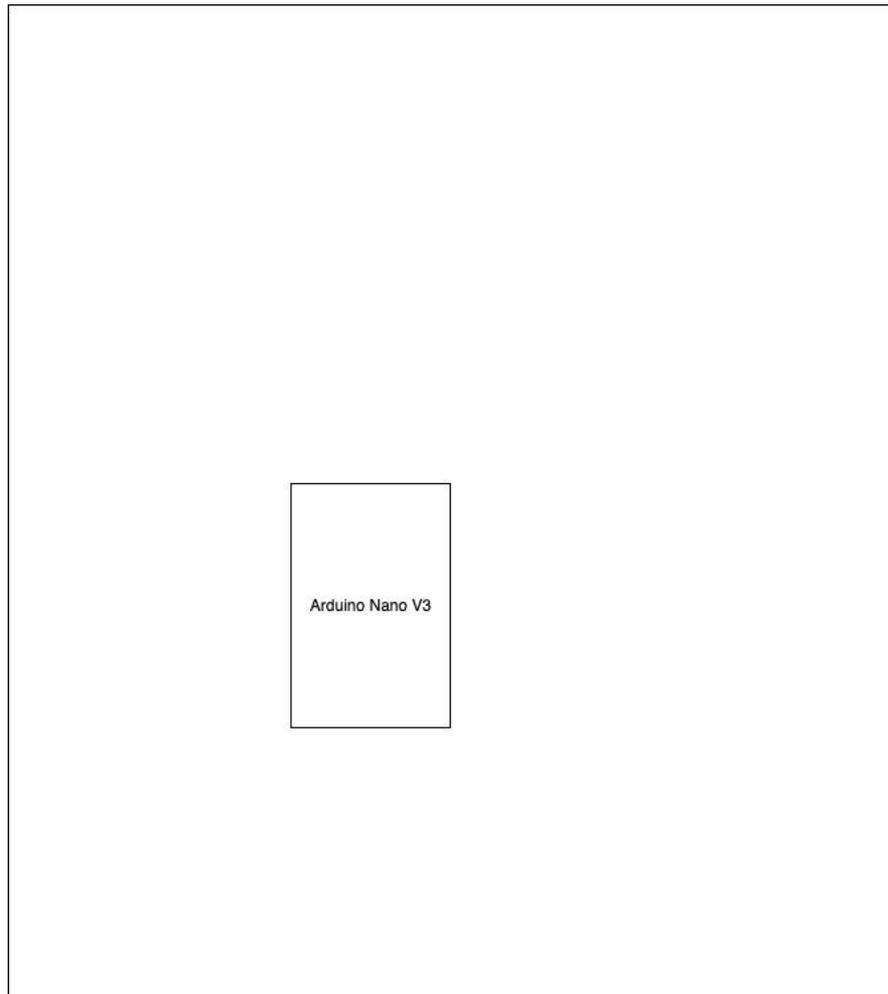
Perancangan sistem merupakan penggambaran, perencanaan, dan pembuatan dari beberapa elemen sistem yang terpisah ke dalam suatu kesatuan model yang utuh berdasarkan analisis sistem yang telah dilakukan sebelumnya. Perancangan akan dimulai setelah tahap analisis terhadap sistem selesai dilakukan.

3.2.1. Perancangan Tata Letak Perangkat

Pada tahapan perancangan tata letak perangkat, akan dilakukan pemetaan tata letak terhadap komponen perangkat keras, sehingga memudahkan dalam proses implementasi letak komponen perangkat pada sarung tangan, berikut tahapan-tahapan tata letak perangkat yang akan dibangun.

1. Tata Letak Perangkat Arduino Nano V3

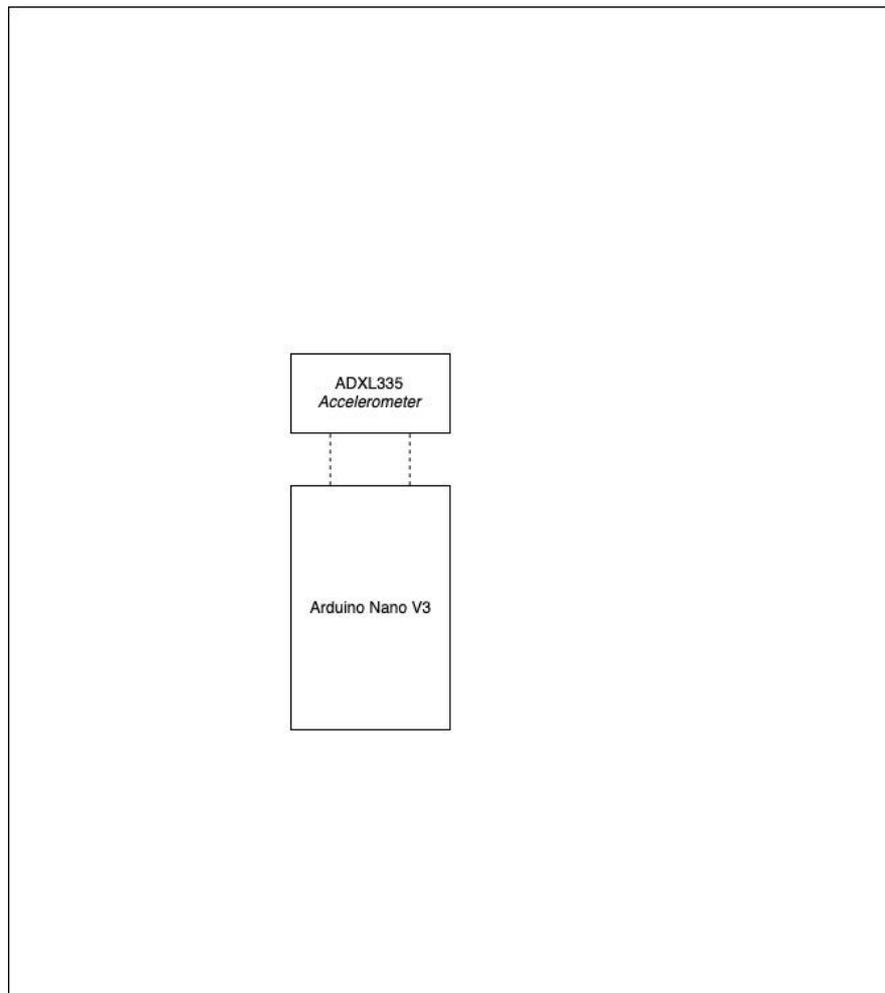
Pada tahapan awal ini yaitu tahapan perancangan tata letak terhadap Arduino Nano V3. Arduino Nano V3 ini digunakan sebagai pengelola sistem yang akan membaca dan memproses beberapa sensor yang dihubungkan dengan mikrokontroler Arduino tersebut. Tata letak perangkat Arduino Nano V3 dapat dilihat pada **Gambar 3.14**.



Gambar 3.14 Perancangan Tata Letak Perangkat Arduino Nano V3

2. Tata Letak Perangkat *Accelerometer Sensor ADXL335*

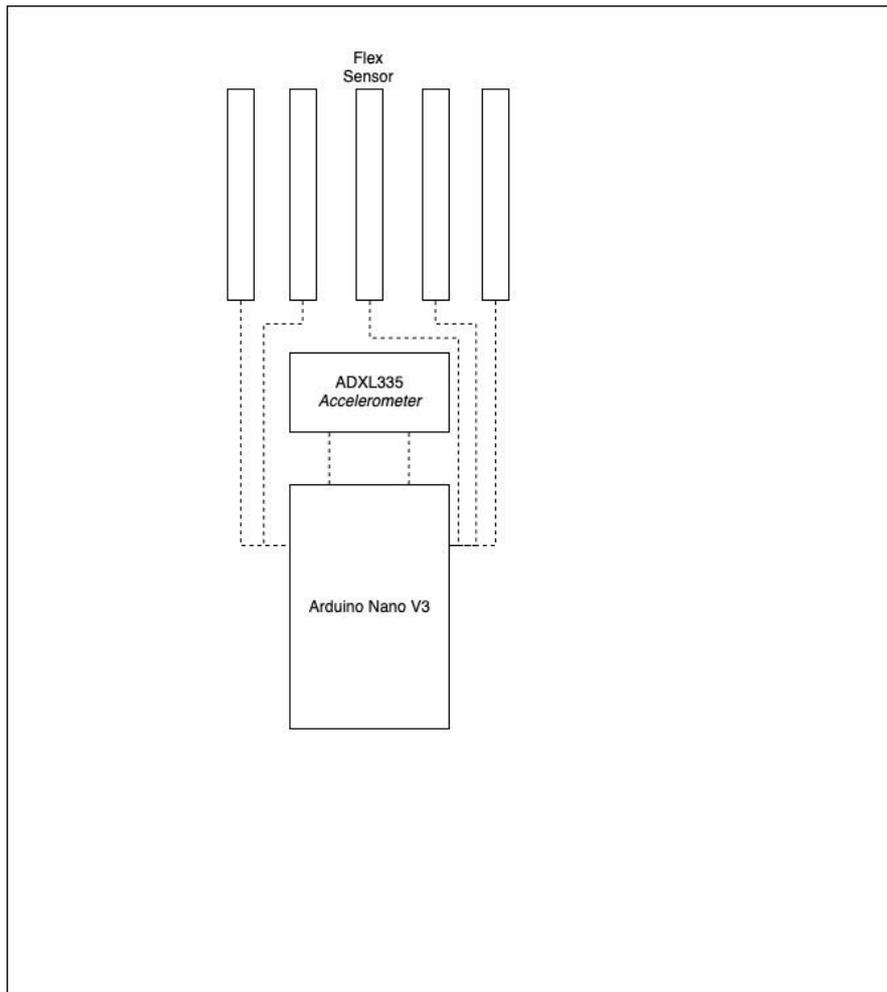
Pada tahapan ke dua ini yaitu tahapan perancangan tata letak terhadap *Accelerometer Sensor ADXL335*. *Accelerometer Sensor ADXL335* dihubungkan dengan Arduino dan berfungsi untuk mendeteksi kemiringan tangan. Tata letak perangkat *Accelerometer Sensor ADXL335* dapat dilihat pada **Gambar 3.15**.



Gambar 3.15 Tata Letak Perangkat *Accelerometer Sensor ADXL335*

3. Tata Letak Perangkat *Flex Sensor*

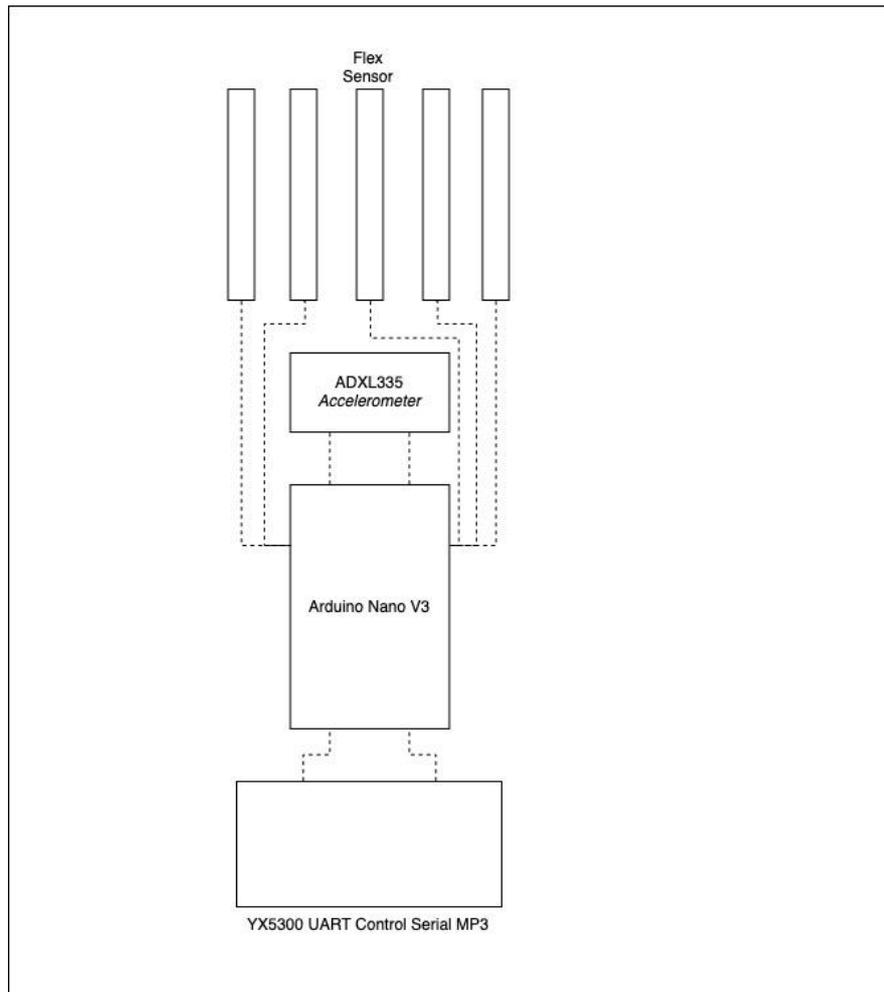
Pada tahapan ke tiga ini yaitu tahapan perancangan tata letak terhadap *Flex Sensor*. *Flex Sensor* dihubungkan dengan Arduino dan berfungsi untuk mendeteksi tekukan jari tangan. Tata letak perangkat *Flex Sensor* dapat dilihat pada **Gambar 3.16**.



Gambar 3.16 Tata Letak Perangkat *Flex Sensor*

4. Tata Letak Perangkat *Serial* MP3 YX5300

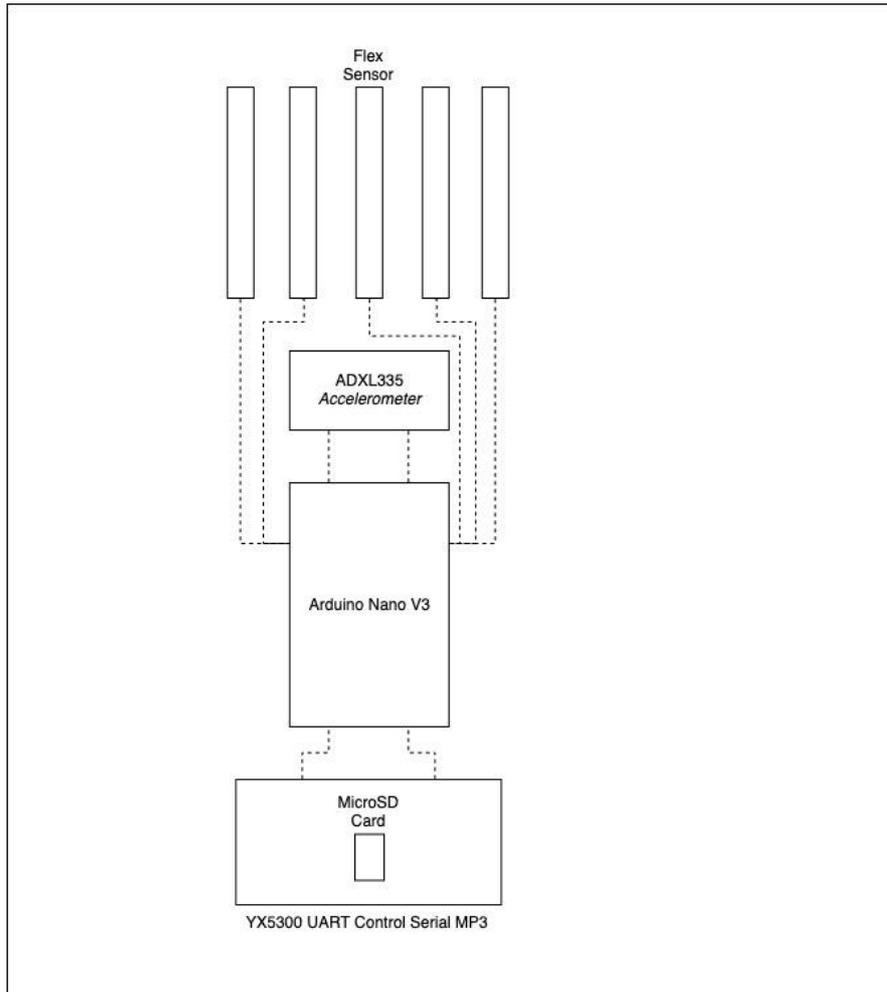
Pada tahapan ke empat ini yaitu tahapan perancangan tata letak terhadap *Serial* MP3 YX5300. *Serial* MP3 YX5300 dihubungkan dengan Arduino dan berfungsi untuk memutar data suara isyarat. Tata letak perangkat *Serial* MP3 YX5300 dapat dilihat pada **Gambar 3.17**.



Gambar 3.17 Tata Letak Perangkat *Serial* MP3 YX5300

5. Tata Letak Perangkat MicroSD Card

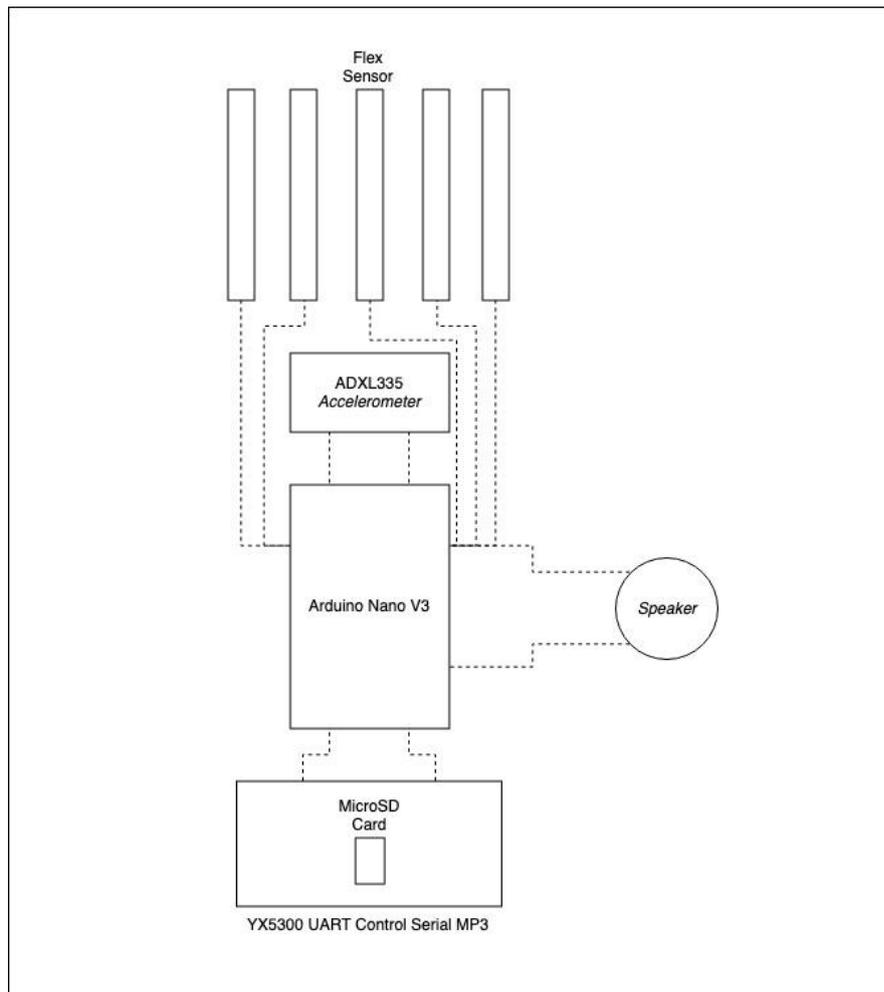
Pada tahapan ke lima ini yaitu tahapan perancangan tata letak terhadap MicroSD Card. MicroSD Card dihubungkan dengan Serial MP3 YX5300 dan digunakan untuk menyimpan data suara isyarat berupa file WAV maupun MP3. Tata letak perangkat MicroSD Card dapat dilihat pada **Gambar 3.18**.



Gambar 3.18 Tata Letak Perangkat MicroSD Card

6. Tata Letak Perangkat *Speaker*

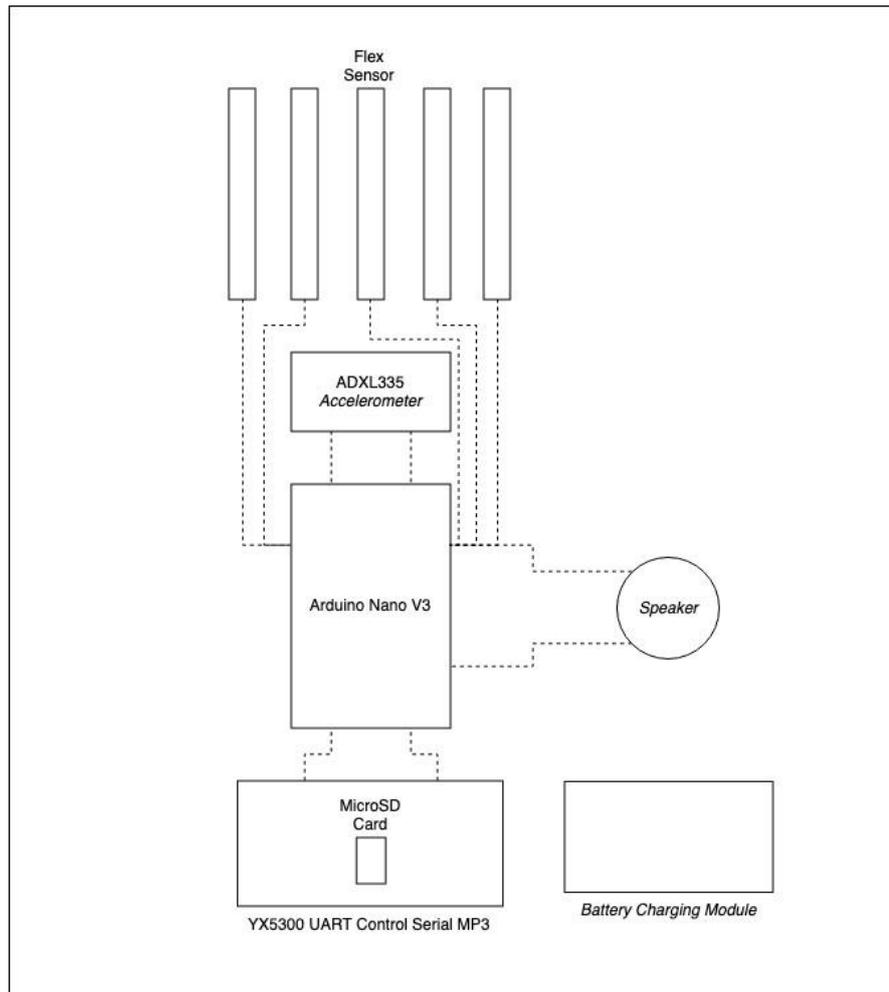
Pada tahapan ke enam ini yaitu tahapan perancangan tata letak terhadap *Speaker*. *Speaker* dihubungkan dengan Arduino dan berfungsi untuk memberikan *output* suara. Tata letak perangkat *Speaker* dapat dilihat pada **Gambar 3.19**.



Gambar 3.19 Tata Letak Perangkat *Speaker*

7. Tata Letak Perangkat *Battery Charging* TP5100

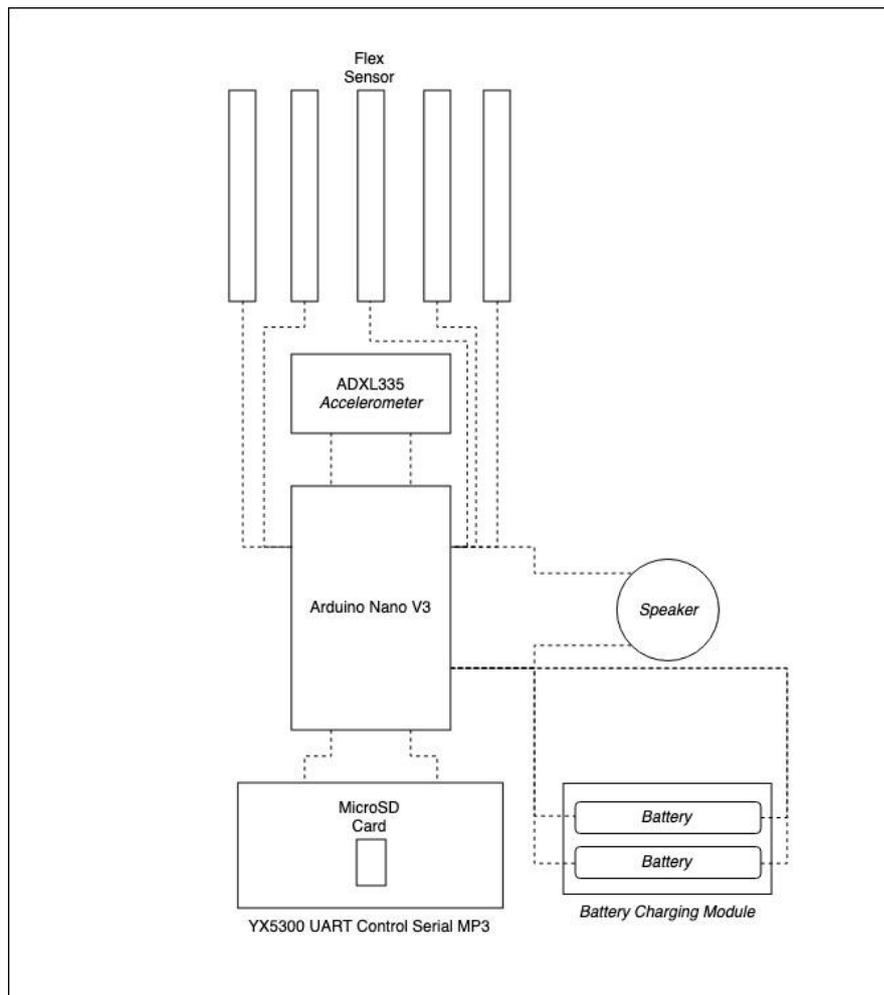
Pada tahapan ke tujuh ini yaitu tahapan perancangan tata letak terhadap *Battery Charging* TP5100. *Battery Charging* TP5100 ini berfungsi untuk mengisi daya baterai. Tata letak perangkat *Battery Charging* TP5100 dapat dilihat pada **Gambar 3.20**.



Gambar 3.20 Tata Letak Perangkat *Battery Charging* TP5100

8. Tata Letak Perangkat *Battery Lithium 18650*

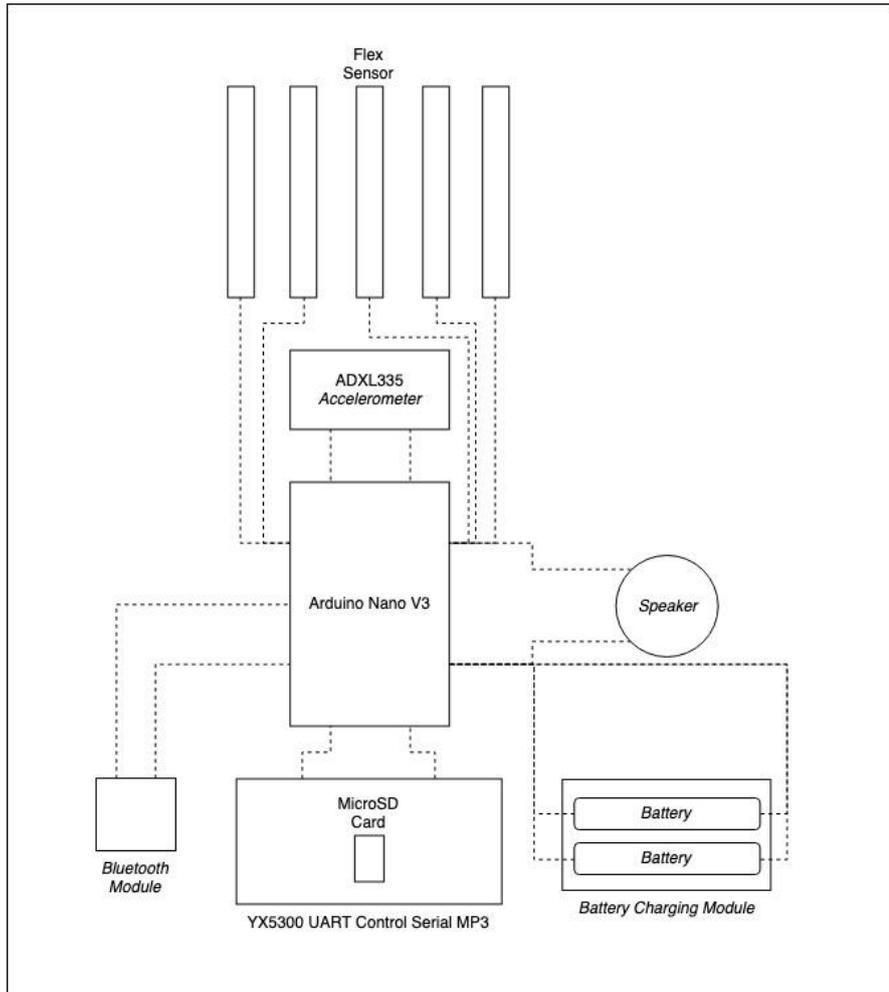
Pada tahapan ke delapan ini yaitu tahapan perancangan tata letak terhadap *Battery Lithium 18650*. *Battery Lithium 18650* dihubungkan dengan Arduino dan *Battery Charging TP5100* yang berfungsi untuk memberikan daya (*power*) terhadap Arduino. Tata letak perangkat *Battery Lithium 18650* dapat dilihat pada **Gambar 3.21**.



Gambar 3.21 Tata Letak Perangkat *Battery Lithium 18650*

9. Tata Letak Perangkat *Bluetooth* HC-05

Pada tahapan terakhir ini yaitu tahapan perancangan tata letak terhadap *Bluetooth* HC-05. *Bluetooth* HC-05 dihubungkan dengan Arduino dan berfungsi sebagai penghubung antar mikrokontroler arduino. Tata letak perangkat *Bluetooth* HC-05 dapat dilihat pada **Gambar 3.22**.



Gambar 3.22 Tata Letak Perangkat *Bluetooth* HC-05