

**BUKU TUGAS AKHIR  
CAPSTONE DESIGN**



**Mobilisasi Robot Pengantar Makanan  
Berbasis *Odometry* dan *QR Detection***

*Diajukan untuk memenuhi syarat kelulusan program sarjana di Prodi S1 Teknik Elektro*

**Oleh:**

**Rifqy Fachrizi / 1102201685**

**Gebby Novalisza / 1102201399**

**Muhammad Rafly / 1104202091**

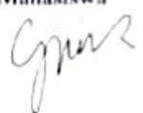
**PRODI S1 TEKNIK ELEKTRO  
FAKULTAS TEKNIK ELEKTRO  
UNIVERSITAS TELKOM  
BANDUNG**

2024

## Dokumentasi Produk Capstone Design

### Lembar Pengesahan Dokumen

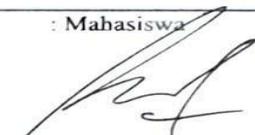
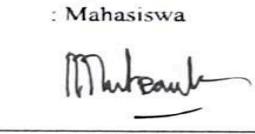
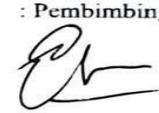
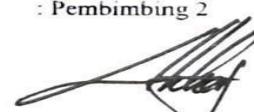
Judul Capstone Design	Mobilisasi Robot Pengantar Makanan Berbasis <i>Sensor Fusion</i> dan <i>Object Detection</i>
Jenis Dokumen	Usulan Gagasan dan Pemilihan Topik
Nomor Dokumen	FTE-CD-1
Nomor Revisi	1 (satu)
Tanggal Pengesahan	20 Oktober 2023
Fakultas	Fakultas Teknik Elektro
Program Studi	S1 Teknik Elektro
Jumlah Halaman	10

Data Pemeriksaan dan Persetujuan				
Dilulus Oleh	Nama NIM	Rifqy Fachrizi 1102201685	Jabatan Tanda Tangan	Mahasiswa 
	Nama NIM	Gebby Novalisza 1102201399	Jabatan Tanda Tangan	Mahasiswa 
	Nama NIM	Muhammad Rafly 1104202091	Jabatan Tanda Tangan	Mahasiswa 
Diperiksa Oleh	Nama Tanggal	Erwin Susanto, S.T., M.T., Ph.D.	Jabatan Tanda Tangan	Pembimbing 1 
	Nama Tanggal	Iriham Mulkan Rodiana, S.T., M.T.	Jabatan Tanda Tangan	Pembimbing 2 
Disetujui Oleh	Nama Tanggal	:	Jabatan Tanda Tangan	Pengaji 1
	Nama Tanggal	:	Jabatan Tanda Tangan	Pengaji 2

## Dokumentasi Produk Capstone Design

### Lembar Pengesahan Dokumen

Judul Capstone Design	:	Mobilisasi Robot Pengantar Makanan Berbasis Sensor Fusion dan Object Detection
Jenis Dokumen	:	Desain Konsep Solusi
Nomor Dokumen	:	FTE-CD-2
Nomor Revisi	:	1 (satu)
Tanggal Pengesahan	:	3 November 2023
Fakultas	:	Fakultas Teknik Elektro
Program Studi	:	SI Teknik Elektro
Jumlah Halaman	:	6 (enam)

Data Pemeriksaan dan Persetujuan				
Ditulis Oleh	Nama : Rifqy Fachrizi NIM : 1102201685	Jabatan : Mahasiswa Tanda Tangan : 		
	Nama : Gebby Novalisza NIM : 1102201399	Jabatan : Mahasiswa Tanda Tangan : 		
	Nama : Muhammad Rafly NIM : 1104202091	Jabatan : Mahasiswa Tanda Tangan : 		
Diperiksa Oleh	Nama : Erwin Susanto, S.T., M.T., Ph.D. Tanggal : 3 November 2023	Jabatan : Pembimbing 1 Tanda Tangan : 		
	Nama : Irham Mulkan Rodiana, S.T., M.T. Tanggal : 3 November 2023	Jabatan : Pembimbing 2 Tanda Tangan : 		
Disetujui Oleh	Nama : Tanggal :	Jabatan : Penguji 1 Tanda Tangan :		
	Nama : Tanggal :	Jabatan : Penguji 2 Tanda Tangan :		

# **LEMBAR PERNYATAAN ORISINALITAS**

Kami, yang bertanda tangan di bawah ini

No.	NIM	Nama	Email
1.	1102201685	Rifqy Fachrizi	kikifachrizi@student.telkomuniversity.ac.id
2.	1102201399	Gebby Novalisza	gebbynovalisza@student.telkomuniversity.ac.id
3.	1104202091	Muhammad Rafly	muhammadraflytelkom@student.telkomuniversity.ac.id

menyatakan bahwa Buku Tugas Akhir Capstone Design ini merupakan karya orisinal kelompok Capstone Design kami bersama dengan judul:

## **MOBILISASI ROBOT PENGANTAR MAKANAN BERBASIS SENSOR FUSION DAN OBJECT DETECTION**

*MOBILIZATION OF FOOD DELIVERY ROBOT BASED ON SENSORS FUSION AND  
OBJECT DETECTION*

Atas pernyataan ini, kami siap menanggung resiko/sanksi yang dijatuhkan kepada kami apabila kemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidak aslian karya ini.

Bandung, 18 Desember 2023

---

Rifqy Fachrizi                                 Gebby Novalisza                                 Muhammad Rafly  
1102201399                                     1102201399                                     1104202091

## TIMELINE REVISI DOKUMEN

Dokumen	Versi, Tanggal	Revisi	Perbaikan yang dilakukan	Halaman Revisi
CD 1	1, 6 Oktober 2023	Penambahan 1 kebutuhan	Ditambahkan satu kebutuhan pada bagian kebutuhan yang harus terpenuhi.	10
	2, 18 Oktober 2023	Kaidah penulisan	Kalimat utama di setiap paragraf tidak perlu ditebal ( <b>bold</b> ).	1,2
		Latar belakang difokuskan pada beberapa permasalahan utama saja	Latar belakang masalah hanya fokus pada segi <i>cost</i> dan <i>quality</i> saja. Beberapa paragraf dihapus dan data numerik dipindahkan ke informasi pendukung.	1,2
		Perbaikan informasi pendukung	Beberapa referensi robot komersil yang dinilai serupa dihapus dari informasi pendukung.	3,4
	3, 19 Oktober 2023	Penambahan informasi pendukung	Penambahan data gaji waiter restoran di Indonesia sebagai perbandingan terhadap harga robot pengantar makanan.	4
	4, 20 Oktober 2023	Perbaikan <i>Constraint</i>	Aspek kesehatan yang diganti menjadi aspek estetika dan desain karena dinilai kurang berkaitan dengan desain <i>trays</i> beserta penutupnya.	10
	7, 15 November 2023	Perbaikan latar belakang	Penambahan satu paragraf khusus yang membahas kekurangan aspek teknis dari robot pengantar makanan yang sudah ada.	1,2
		Perbaikan kebutuhan	Mengganti kata <i>sensor fusion</i> ke penggabungan sensor.	12
CD 2	5, 31 Oktober 2023	Pemetaan kebutuhan dan spesifikasi	Spesifikasi kebutuhan 1 dan 2 yang awalnya digabungkan menjadi dipisahkan.	11
		Penambahan verifikasi	Penambahan tabel verifikasi spesifikasi 2.	14
	6, 3 November 2023	Penambahan rincian spesifikasi	Pada spesifikasi 1, ditambahkan alasan yang memperkuat penentuan besarnya toleransi kecepatan.	11-12
		Perbaikan spesifikasi	Perbaikan spesifikasi 2 dan pergantian besarnya kuantitas pada spesifikasi 5.	11-12,14
	8, 19 November 2023	Perbaikan spesifikasi	Perbaikan tingkat akurasi pada spesifikasi 3	13,15
CD3	9, 1 Desember 2023	Perubahan Diagram Blok Level 1 dan 2	Diagram blok level 1 harusnya berisi perangkat keras/hardware yang digunakan sementara diagram blok level 2 menjelaskan proses kerjanya dengan lebih rinci	19-24

10, 3 Desember 2023	Desain Sistem	Perbaikan letak kamera pada sistem	39
11, 5 Desember 2023	Pemilihan Komponen	Perbaikan pembobotan kriteria tiap komponen yang seharusnya berkisar di angka 1,0, dan -1 saja.	27-38
	Cara Penggunaan Sistem	Tahapan penggunaan sistem dibuat lebih rinci	39-40
12, 12 Desember 2023	Jadwal Pengerjaan	Penambahan penjadwalan kerja dan pembagian tugas anggota tim serta gantt chart	41-42

## KATA PENGANTAR

Puji syukur ke hadirat Tuhan yang Maha Esa yang telah memberikan rahmat dan hidayah-Nya sehingga tim penulis mampu menyelesaikan Capstone Design yang berjudul “Mobilisasi Robot Pengantar Makanan Berbasis *Sensor Fusion* dan *Object Detection*” untuk memenuhi syarat kelulusan jenjang sarjana program studi S1 Teknik Elektro Universitas Telkom.

Tim penulis juga ingin menyampaikan rasa terima kasih kepada Bapak Erwin Susanto, S.T, M.T, Ph.D dan Bapak Irham Mulkan Rodiana, S.T, M.T selaku dosen pembimbing 1 dan 2 yang telah membimbing tim penyusun selama penyusunan dan penggerjaan Capstone Design ini.

Akhir kata, tim penulis menyadari bahwa Capstone Design ini masih jauh dari kata sempurna. Oleh karena itu, saya mengharapkan kritik dan saran yang membangun dari berbagai pihak sebagai bahan masukan untuk memperbaiki dan menyempurnakan Capstone Design ini. Besar harapan penulis bahwa Capstone Design ini mampu berdampak dan berkontribusi bagi bidang keilmuan robotika di masa mendatang.

Bandung, 18 Desember 2023

Tim Penulis

# DAFTAR ISI

LEMBAR PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
DAFTAR SINGKATAN	xiii
ABSTRAK	xiv
ABSTRACT	xv
BAB 1 ANALISIS KEBUTUHAN	1
1.1 Latar Belakang Masalah	1
1.2 Informasi Pendukung	2
1.3 <i>Constraint</i>	8
1.4 Kebutuhan yang Harus Dipenuhi	9
1.5 Tujuan	9
BAB 2 SPESIFIKASI DAN VERIFIKASI	10
2.1 Spesifikasi Produk	11
2.1.1 Pergerakan Menuju Titik Tujuan yang Telah Ditentukan di Dalam Ruangan pada Bidang Datar	11
2.1.2 Pergerakan ke Segala Arah/ <i>Omni-Directional</i>	11
2.1.3 Pengidentifikasi Meja Pelanggan Menggunakan Kode QR.	12
2.1.4 Desain <i>Tray</i> beserta Penutup di Sekelilingnya dan Salah Satu <i>Tray</i> Memiliki Fitur untuk Menyajikan Makanan kepada Pelanggan	13
2.2 Verifikasi	13
2.2.1 Spesifikasi 1	14
2.2.2 Spesifikasi 2	14
2.2.3 Spesifikasi 3	15
2.2.4 Spesifikasi 4	15
2.2.5 Spesifikasi 5	16
BAB 3 DESAIN RANCANGAN SOLUSI	17
3.1 Konsep Solusi	17
3.1.1 Diagram Fungsi	17
3.1.2 Karakteristik Solusi	18

3.2 Rencana Desain dari Konsep Solusi Sistem	18
<b>3.2.1</b> Diagram Blok Level 0	18
<b>3.2.2</b> Diagram Blok Level 1	19
<b>3.2.3</b> Diagram Blok Level 2	20
3.3 Pemilihan Komponen	28
3.4 Desain Sistem Terpilih dan Cara Penggunaannya	41
<b>3.4.1</b> Desain Sistem	41
<b>3.4.2</b> Cara Penggunaan Sistem	43
3.5 Jadwal Penggerjaan	44
<b>BAB 4 IMPLEMENTASI SOLUSI</b>	47
4.1 Implementasi Sistem	47
<b>4.1.1</b> Unit Kontrol	48
<b>4.1.2</b> Sensor Odometri dan <i>Obstacles Avoidances</i>	54
<b>4.1.3</b> Aktuator <i>Base Robot</i>	60
<b>4.1.4</b> Pendektsian QR	68
<b>4.1.5</b> Aktuator Penutup dan <i>Tray</i>	71
4.2 Hasil Akhir Integrasi Sistem	73
<b>BAB 5 PENGUJIAN SISTEM</b>	74
5.1 Pengujian Sistem (secara keseluruhan sesuai spesifikasi CD2)	74
<b>5.1.1</b> Pengujian Spesifikasi 1	74
<b>5.1.2</b> Pengujian Spesifikasi 2	75
<b>5.1.3</b> Pengujian Spesifikasi 3	77
<b>5.1.4</b> Pengujian Spesifikasi 4	78
<b>5.1.5</b> Pengujian Spesifikasi 5	81
5.2 Kesimpulan dan Saran	82
<b>5.2.1</b> Kesimpulan	82
<b>5.2.2</b> Saran	84
<b>DAFTAR PUSTAKA</b>	85
LAMPIRAN CD-1	89
LAMPIRAN CD-2	93
LAMPIRAN CD-3	95
LAMPIRAN CD-4	96
LAMPIRAN CD-5	97

## DAFTAR GAMBAR

<b>Gambar 1.2.1 The Matradee Food Delivery Robot by RichTech Robotics [5]</b>	<b>2</b>
<b>Gambar 1.2.2 PuduBot by Pudu Robotics [6]</b>	<b>3</b>
<b>Gambar 1.2.3 SwiftBot by Pudu Robotics [7]</b>	<b>3</b>
<b>Gambar 4 Overall Function</b>	<b>17</b>
<b>Gambar 5 Function Tree</b>	<b>17</b>
<b>Gambar 6 Diagram Blok Level 0</b>	<b>18</b>
<b>Gambar 7 Diagram Blok Level 1</b>	<b>19</b>
<b>Gambar 8 Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi</b>	<b>20</b>
<b>Gambar 9 Diagram Blok Level 2 pada Sensor External Rotary Encoder, IMU, dan jarak</b>	<b>21</b>
<b>Gambar 10 Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi Bagian 2</b>	
	<b>22</b>
<b>Gambar 11 Diagram Blok Level 2 pada Mikrokontroler Base</b>	<b>22</b>
<b>Gambar 12 Diagram Blok Level 2 pada Motor Driver dan Motor DC</b>	<b>23</b>
<b>Gambar 13 Diagram Blok Level 2 pada Kamera dan Mini-PC</b>	<b>23</b>
<b>Gambar 14 Diagram Blok Level 2 pada Mikrokontroler Mekanisme Tray, Motor Driver, Motor DC untuk Tray, dan Linear Aktuator</b>	<b>24</b>
<b>Gambar 15 Flowchart Sistem</b>	<b>27</b>
<b>Gambar 16 Desain Sistem Robot Pengantar Makanan Berbasis Odometri dan QR Detection</b>	<b>42</b>
<b>Gambar 17 Desain Sistem Robot Pengantar Makanan Berbasis Sensor Fusion dan Object Detection secara keseluruhan</b>	<b>42</b>
<b>Gambar 18 Jadwal Pengerjaan dan Pembagian Tugas Anggota Tim</b>	<b>45</b>
<b>Gambar 19 Gantt Chart</b>	<b>46</b>
<b>Gambar 20 Diagram Blok Level 0</b>	<b>47</b>
<b>Gambar 21 Diagram Blok Level 1 Beserta Sub-Sistemnya</b>	<b>47</b>
<b>Gambar 22 Diagram Arsitektur Sistem</b>	<b>48</b>
<b>Gambar 23 Tampilan Aplikasi sebagai User Interface</b>	<b>49</b>
<b>Gambar 24 Block Programming pada Perancangan Aplikasi User Interface</b>	<b>50</b>
<b>Gambar 25 Source Code Penerimaan Char dari Aplikasi ke ESP32 melalui Bluetooth</b>	<b>51</b>
<b>Gambar 26 Sensor Rotary Encoder</b>	<b>55</b>

<b>Gambar 27 Bagian-bagian Optical Rotary Encoder</b>	55
<b>Gambar 28 Roda Omnidirectional 60 mm</b>	56
<b>Gambar 29 Konfigurasi Roda Odometri (Kiri) dan Sudut Vektor Putaran Roda (Kanan)</b>	56
<b>Gambar 30 Sensor IMU MPU6050</b>	56
<b>Gambar 31 Pitch, Roll, dan Yaw dalam Ruang Gerak 3 Dimensi</b>	57
<b>Gambar 32 Source Code Fungsi Odometri</b>	58
<b>Gambar 33 Sensor Ultrasonik HCSR-04 (Kiri) dan Prinsip Kerjanya (Kanan)</b>	58
<b>Gambar 34 STM32F411CEU6-WeAct Blackpill V2.0 (Kiri) dan Motor DC PG45 (Kanan)</b>	60
<b>Gambar 35 Diagram Skematik Sub-Sistem Aktuator Base Robot</b>	60
<b>Gambar 36 Motor Driver BTS7960</b>	61
<b>Gambar 37 Fungsi</b>	62
<b>Gambar 38 Source Code Perhitungan PID Kecepatan</b>	66
<b>Gambar 39 Logitech Webcam C310</b>	69
<b>Gambar 40 Source Code Pendekripsi QR</b>	70
<b>Gambar 41 Hasil Integrasi Sistem</b>	73
<b>Gambar 42 Trajectory Pergerakan Robot Pengantar Makanan Berbasis Odometry dan QR Detection di Lokasi Pengujian Telyu Caffee</b>	74
<b>Gambar 43 Posisi Awal Pendekripsi Kode QR dan Kode QR Belum Terdeteksi</b>	77
<b>Gambar 44 Posisi Akhir Pendekripsi Kode QR dan Kode QR Berhasil Terdeteksi</b>	78
<b>Gambar 45 Simulasi FEA Sebelum Robot Diberi Beban</b>	79
<b>Gambar 46 Simulasi FEA Setelah Robot Diberi Beban Pada Tray Slider</b>	79
<b>Gambar 47 Simulasi FEA Setelah Robot Diberi Beban Pada Tray Statik</b>	80
<b>Gambar 48 Simulasi FEA Ketika Robot Diberi Beban Pada Kedua Tray</b>	80
<b>Gambar 49 Kondisi Tray Slider Sebelum (Kiri) dan Setelah (Kanan) Bergerak Menyajikan Makanan</b>	81
<b>Gambar 50 Pengukuran Perpindahan Tray Slider Setelah Bergerak Selama 24 Sekon</b>	82

## DAFTAR TABEL

<b>Tabel 1.1.2.1 Harga Robot Pengantar Makanan di Marketplace</b>	3
<b>Tabel 1.1.2.2 Daftar Penelitian Terkait yang Sudah Dilakukan Sebelumnya</b>	4
<b>Tabel 2.1 Pemetaan Kebutuhan dan Kaitannya Terhadap Spesifikasi</b>	10
<b>Tabel 2.2 Verifikasi Spesifikasi 1</b>	14
<b>Tabel 2.3 Spesifikasi 2</b>	14
<b>Tabel 2.4 Verifikasi Spesifikasi 3</b>	15
<b>Tabel 2.5 Verifikasi Spesifikasi 4</b>	15
<b>Tabel 2.6 Verifikasi Spesifikasi 5</b>	16
<b>Tabel 3.1 Rincian Diagram Blok Level 0</b>	18
<b>Tabel 3.2 Rincian Diagram Blok Level 1</b>	20
<b>Tabel 3.3 Rincian Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi</b>	21
<b>Tabel 3.4 Rincian Diagram Blok Level 2 pada Sensor External Rotary Encoder, IMU, dan jarak</b>	21
<b>Tabel 3.5 Rincian Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi Bagian 2</b>	22
<b>Tabel 3.6 Rincian Diagram Blok Level 2 pada Mikrokontroler Base</b>	22
<b>Tabel 3.7 Rincian Diagram Blok Level 2 pada Motor Driver dan Motor DC</b>	23
<b>Tabel 3.8 Rincian Diagram Blok Level 2 pada Kamera dan Mini-PC</b>	23
<b>Tabel 3.9 Rincian Diagram Blok Level 2 pada Mikrokontroler Mekanisme Tray, Motor Driver, Motor DC untuk Tray, dan Linear Aktuator</b>	24
<b>Tabel 3.10 Pemilihan Komponen Mikrokontroler Sensor dan Aplikasi</b>	28
<b>Tabel 3.11 Pemilihan Komponen Mikrokontroler Base</b>	29
<b>Tabel 3.12 Pemilihan Komponen Mikrokontroler Mekanisme Tray</b>	30
<b>Tabel 3.13 Pemilihan Konponen Mini PC</b>	31
<b>Tabel 3.14 Pemilihan Komponen Sensor External Rotary Encoder</b>	32
<b>Tabel 3.15 Pemilihan Komponen Sensor IMU</b>	33
<b>Tabel 3.16 Pemilihan Komponen Sensor Jarak</b>	33
<b>Tabel 3.17 Pemilihan Komponen Kamera</b>	34
<b>Tabel 3.18 Pemilihan Komponen Motor DC untuk Base</b>	35
<b>Tabel 3.19 Pemilihan Komponen Motor Driver</b>	35
<b>Tabel 3.20 Pemilihan Komponen Baterai</b>	36

<b>Tabel 3.21 Pemilihan Komponen Linear Aktuator</b>	37
<b>Tabel 3.22 Pemilihan Komponen Motor DC untuk Penggerak Penutup Tray</b>	37
<b>Tabel 3.23 Pemilihan Komponen Roda Omniwheel</b>	38
<b>Tabel 3.24 Pemilihan Komponen Kerangka Alumunium</b>	38
<b>Tabel 3.25 Pemilihan Komponen Material Penutup</b>	39
<b>Tabel 3.26 Pemilihan Komponen Material Tray</b>	40
<b>Tabel 3.27 Rangkuman Hasil Pemilihan Komponen</b>	40
<b>Tabel 4.1 Pengiriman dan Penerimaan Data Char dari Aplikasi ke ESP32</b>	52
<b>Tabel 4.2 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Depan</b>	59
<b>Tabel 4.3 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Belakang</b>	59
<b>Tabel 4.4 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kanan</b>	59
<b>Tabel 4.5 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kiri</b>	59
<b>Tabel 4.6 Nilai PWM dan Feedback Kontroler PID dengan Setpoint 0,5 PPR</b>	67
<b>Tabel 4.7 Nilai PWM dan Feedback Kontroler PID dengan Setpoint 1 PPR</b>	67
<b>Tabel 4.8 Nilai PWM dan Feedback Kontroler PID dengan Setpoint 1,5 PPR</b>	68
<b>Tabel 4.9 Jarak Terjauh Pendeksi QR</b>	71
<b>Tabel 4.10 Perpindahan Penutup yang Digerakkan oleh Motor DC PG28</b>	72
<b>Tabel 5.7 Perpindahan dan Kecepatan Linear Tray Slider</b>	72
<b>Tabel 5.1 Waktu Tempuh dan Kecepatan Linear Robot</b>	75
<b>Tabel 5.2 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Depan</b>	76
<b>Tabel 5.3 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Belakang</b>	76
<b>Tabel 5.4 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kanan</b>	76
<b>Tabel 5.5 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kiri</b>	77
<b>Tabel 5.6 Jarak Terjauh Pendeksi QR</b>	78
<b>Tabel 5.7 Perpindahan dan Kecepatan Linear Tray Slider</b>	82

## DAFTAR SINGKATAN

Singkatan	Arti
ADC	<i>Analog to Digital Converter</i>
AHRS	<i>Altitude and Heading Reference System</i>
CPU	<i>Central Processing Unit</i>
DC	<i>Direct Current</i>
FEA	<i>Finite Element Analysis</i>
FPS	<i>Frame per Second</i>
F&B	<i>Food &amp; Beverages</i>
GPIO	<i>General Purpose Input/Output</i>
GPS	<i>Global Positioning System</i>
GPU	<i>Graphics Processing Unit</i>
I2C	<i>Inter-Integrated Circuit</i>
IMU	<i>Inertial Measurement Unit</i>
I/O	<i>Input/Output</i>
IR	<i>Infra-Red</i>
LIDAR	<i>Light Detection and Ranging</i>
MP	<i>Mega Pixel</i>
NUC	<i>Next Unit of Computing</i>
PWM	<i>Pulse Width Modulation</i>
RFID	<i>Radio Frequency Identification</i>
RGBD	<i>Red, Green, Blue Depth</i>
ROS	<i>Robot Operating System</i>
SLAM	<i>Simultaneous Localization and Mapping</i>
UART	<i>Universal Asynchronous Receiver/Transmitter</i>
USD	<i>The U.S. Dollar</i>
VDC	<i>Voltage Direct Current</i>
YOLO	<i>You Only Look Once</i>

## **ABSTRAK**

Saat ini, layanan pengantaran makanan kepada pelanggan di kafe, restoran, atau kantin masih bergantung pada tenaga kerja manusia. Akan tetapi, sistem konvensional ini dianggap tidak efisien dari segi biaya dan kualitas. Oleh karena itu, penggunaan robot pengantar makanan dianggap sebagai alternatif yang baik. Meskipun demikian secara teknis, robot pengantar makanan yang dikembangkan melalui penelitian dan komersil masih belum sempurna karena dikendalikan secara manual, mobilitas yang tidak fleksibel, ketidakakuratan dalam menentukan lokasi tujuan serta desain tray yang bersifat statis. Harga jual robot pengantar makanan juga sangat tinggi.

Oleh karena itu, solusi yang diusulkan adalah pengembangan robot pengantar makanan yang mampu bergerak ke segala arah menuju tujuan yang telah ditentukan, mampu mendeteksi objek dalam ruang lingkup tertentu, serta dilengkapi desain penutup dan salah satu *tray* yang dapat bergerak untuk menyajikan makanan kepada pelanggan. Dengan demikian, harapan yang hendak dicapai adalah robot mampu mengantarkan makanan dengan akurat serta menunjang aspek estetika dan keamanan makanan hingga tiba di tangan pelanggan.

Kata kunci : Mobilisasi, *Object Detection*, Robot Pengantar Makanan, *Sensor Fusion*, *Tray*

## **ABSTRACT**

Currently, food delivery services in cafes, restaurants, or cafeterias still depend on human labor. However, this conventional system is considered inefficient in terms of cost and quality. Therefore, the utilization of food delivery robots is seen as a practical alternative. Nonetheless, technically, food delivery robots which developed through research and commercial nowadays are not yet perfected due to manual control, rigid mobility, inaccuracies in determining destination locations, and stationary tray design. Additionally, the food delivery robot is characterized by a significantly high price.

Hence, the proposed solution involves developing a food delivery robot that capable of moving in all directions towards predetermined destinations using sensor fusion and object detection within a specific range, and equipped with protection case along with a movable tray to serve food onto the customer's table. The ultimate goal is the robot can deliver food accurately while enhancing aesthetic and food safety aspects until it reaches the customer's hands.

Keywords: Food Delivery Robot, Mobilization, Object Detection, Sensor Fusion, Tray

# BAB 1

## ANALISIS KEBUTUHAN

### 1.1 Latar Belakang Masalah

Pada abad ke-21, industri F&B menjadi sektor usaha yang berkembang pesat secara global karena mampu mengadaptasi perkembangan teknologi, termasuk bidang robotika. Salah satunya dalam layanan pengantar makanan kepada pelanggan di kafe, restoran, atau kantin. Layanan pengantar makanan kepada pelanggan menjadi suatu aspek penting yang perlu diperhatikan karena berdampak langsung pada kepuasan dan loyalitas pelanggan terhadap kantin atau restoran tersebut. Sebelumnya, layanan pengantar makanan mengandalkan tenaga kerja manusia. Namun, sistem konvensional ini dinilai tidak efektif dari segi biaya (*cost*) dan kualitas (*quality*).

Apabila dilakukan perbandingan, gaji tenaga kerja manusia jauh lebih besar dibandingkan biaya penggunaan robot. Untuk mempergunakan robot, pengguna hanya perlu mengeluarkan dana yang cukup besar di awal dan dana yang jauh lebih kecil secara berkala untuk biaya pemeliharaan. Di sisi lain, gaji tenaga kerja manusia malah berpotensi mengalami kenaikan dari tahun ke tahun [1]. Hal ini tidak sebanding dengan kualitas pelayanan tenaga kerja manusia yang cenderung tidak stabil akibat keterbatasan energi. Tenaga kerja manusia juga berpotensi jatuh sakit dan merasa jemu [2]. Apabila tenaga kerja manusia telah kehabisan tenaga, kemungkinan buruk yang dapat terjadi adalah timbulnya *human error service* seperti kesalahan dalam pengantaran makanan yang dipesan kepada pelanggan. Jika hal ini terjadi, maka kualitas pelayanan restoran atau kantin tersebut akan menurun di mata pelanggan. Berbeda dengan robot yang mampu konsisten dalam melakukan pekerjaan repetitif selama daya terisi penuh dan dipelihara secara berkala [3].

Oleh karena itulah, robot pengantar makanan menjadi alternatif yang dipilih untuk menggantikan tenaga kerja manusia. Namun dari sisi teknis, robot pengantar makanan yang dikembangkan lewat penelitian belum sempurna karena masih dikendalikan secara manual. Selain itu, mobilitas robot cenderung tidak fleksibel karena hanya dapat bergerak dalam lintasan garis yang telah ditentukan saja. Beberapa robot juga tidak dilengkapi dengan fitur yang mampu mengidentifikasi dan membedakan meja pelanggan. Adapun desain *tray* yang diusung untuk mengangkut makanan masih bersifat statis dan belum mampu menyajikan makanan kepada pelanggan.

Berdasarkan permasalahan tersebut, diperlukan robot pengantar makanan yang memiliki fitur serupa dengan beberapa pengembangan/inovasi. Sama halnya seperti robot pengantar makanan pada umumnya, robot ini mampu bermobilisasi menuju ke titik tujuan yang telah ditentukan oleh pengguna dengan memanfaatkan informasi dari sensor pergerakan sehingga mampu memperkirakan posisi robot dari waktu ke waktu. Untuk mempermudah mobilisasi, robot juga dirancang mampu melakukan gerakan holonomik (*omni-directional mobile robot*) karena lebih unggul dari segi biaya yang rendah, kesederhanaan, dan keandalan dalam navigasi di berbagai medan lingkungan [4]. Selain itu, robot ini juga dilengkapi dengan *QR detection* untuk mengidentifikasi dan membedakan meja pelanggan satu dengan yang lainnya. Tak hanya terkait dengan mobilisasi, desain mekanik robot juga diperhatikan agar tetap mendukung faktor estetika dan robot mampu menyajikan pesanan di atas meja pelanggan.

## 1.2 Informasi Pendukung

Menurut penelusuran yang telah dilakukan, berikut ini beberapa robot pengantar makanan yang beredar di *marketplace*. Pertama, the Matradee adalah robot pelayanan yang diproduksi oleh Richtech Robotics. Robot ini memiliki *4 adjustable trays* yang mampu mengangkut beban hingga 22 lbs per tray-nya.



**Gambar 1.2.1 The Matradee Food Delivery Robot by Richtech Robotics [5]**

Tidak hanya Richtech Robotics yang merancang produk robot pengantar makanan dengan desain trays, Pudu Robotics juga mengeluarkan produk serupa yang dinamai PuduBot dan SwiftBot. Apabila dibandingkan desain PuduBot tampak serupa dengan The Matradee. Namun kelebihannya adalah dapat dilengkapi dengan *protection case* untuk melindungi makanan yang tengah dibawa dari paparan debu atau partikel kecil lain yang dapat terkontaminasi dan mengurangi kualitas makanan. Akan tetapi, salah satu kekurangan dari

*protection case* ini adalah masih memerlukan intervensi manual dari pelanggan karena tidak dapat terbuka secara otomatis [6].



**Gambar 1.2.2 PuduBot by Pudu Robotics [6]**

Salah satu produk Pudu Robotics yang nampak berbeda didesain dari dua robot pengantar makanan yang sebelumnya adalah SwiftBot. Robot tipe ini memiliki *automatic electric door* yang akan terbuka apabila sudah tiba di meja dan makanan hendak disajikan kepada pelanggan [7].



**Gambar 1.2.3 SwiftBot by Pudu Robotics [7]**

Meskipun begitu, kekurangan beberapa produk robot pengantar makanan komersil ini adalah harganya yang cukup tinggi. Harga dari beberapa tipe robot pengantar makanan tersebut disajikan pada tabel 1.2.1 berikut ini.

**Tabel 1.1.2.1 Harga Robot Pengantar Makanan di Marketplace**

No.	Manufaktur	Nama Robot	Harga*
1	Richtech Robotics	The Matradee	20.000 USD [8] setara Rp307.400.000,00

2	Pudu Robotics	PuduBot	10.000 USD [9] setara Rp153.700.000,00
3		SwiftBot	28.500 USD [10] setara Rp438.045.000,00

\*Belum termasuk biaya *Good and Services Taxes (GST)* dan *Shipping Cost*. Berdasarkan konversi *USD ke IDR per tanggal 22 September 2023 pukul 7:59 UTC*

Adapun sebagai bahan perbandingan, gaji *waiter* di PT. Solaria Indonesia berada di kisaran angka Rp84.000.000,00 – Rp132.000.000,00 per tahun. Biaya ini belum termasuk biaya tunjangan lainnya yang diperoleh pegawai [11]. Dengan kualitas pelayanan yang cenderung konsisten, robot pengantar makanan menjadi alternatif yang baik dalam sektor ini. Bahkan saat ini di Indonesia, telah ada dua kafe dan restoran yang menggunakan robot pengantar makanan sebagai pelayan yakni Rasa Koffie, Jakarta [12] dan Reveuse Resto, Bandung [13]. Dengan mengusung konsep futuristik dengan penggunaan teknologi canggih, keberadaan robot pengantar makanan menjadi daya tarik tersendiri bagi pelanggan.

Selain melakukan penelusuran terhadap produk robot pengantar makanan yang telah diproduksi secara komersil oleh beberapa perusahaan robotika, juga terdapat penelitian-penelitian sebelumnya yang membahas mengenai robot pengantar makanan yang dirincikan pada tabel 1.2.2 berikut ini.

**Tabel 1.1.2.2 Daftar Penelitian Terkait yang Sudah Dilakukan Sebelumnya**

No	Judul Penelitian Sebelumnya	Fitur-fitur	Kekurangan	Referensi
1	IoT Waiter Bot: A Low Cost IoT Based Multi-Functional Robot for Restaurants	<ul style="list-style-type: none"> <li>- Sensor IR mendekripsi garis (<i>line following</i>) dan memungkinkan pergerakan secara otonom.</li> <li>- RFID <i>reader</i> mengenali pesanan tertentu milik pelanggan.</li> <li>- Sistem antarmuka yang menghubungkan pelanggan dan pengelola usaha.</li> </ul>	<ul style="list-style-type: none"> <li>- Belum disertai dengan desain <i>trays</i> untuk mengangkut makanan.</li> <li>- Performa robot dan mobilitasnya kurang fleksibel, tergantung jalur garis saja.</li> <li>- Belum mampu mendekripsi lingkungan sekitar dan keberadaan pelanggan.</li> </ul>	[14]
2	Fabrication of Low-Cost Food Delivery Robot	<ul style="list-style-type: none"> <li>- Sensor ultrasonik dan IR untuk <i>mapping</i> dan lokalisasi dengan konsep <i>line following</i> sehingga mampu bergerak secara otonom.</li> </ul>	<ul style="list-style-type: none"> <li>- Kurang akurat dalam mendekripsi destinasi (meja yang ditempati pelanggan terkait).</li> <li>- 3 <i>trays</i> pengangkut makanan bersifat statis.</li> </ul>	[15]

		<ul style="list-style-type: none"> <li>- Desain 3 <i>trays</i> untuk mengangkut makanan.</li> <li>- Modul bluetooth untuk komunikasi kontrol gerakan secara manual.</li> </ul>		
3	An Autonomous Mobile Robot for Medical Purpose	<ul style="list-style-type: none"> <li>- Penutup akrilik melindungi makanan pada <i>trays</i>.</li> <li>- Bagian atas robot (<i>trays</i>) bersifat modular artinya bisa diganti dengan sub-sistem lain.</li> <li>- 24 VDC <i>heater element</i> dan sensor DHT22 untuk mengendalikan suhu dari <i>trays</i> makanan.</li> <li>- Modul bluetooth untuk komunikasi kontrol gerakan secara manual.</li> </ul>	<ul style="list-style-type: none"> <li>- Penutup akrilik belum mampu terbuka-tertutup secara otomatis.</li> <li>- Tinggi robot belum menyesuaikan ketinggian aktual meja.</li> <li>- Pemilihan motor stepper untuk base segi empat dengan roda mekanum kurang tepat karena mengurangi torsi dan kecepatan robot.</li> <li>- Tidak membahas metode kontrol pergerakan menuju titik tertentu.</li> </ul>	[16]
4	A New Automated Food Delivery System Using Autonomous Track Guided Centre-Wheel Drive Robot	<ul style="list-style-type: none"> <li>- Fotodioda mendeksi garis yang tercetak di lantai (<i>line following</i>) sehingga memungkinkan pergerakan otonom.</li> <li>- <i>Tray</i> makanan dengan mekanisme <i>lifting</i> menggunakan katrol yang terhubung dengan motor dan <i>limit switch</i>.</li> </ul>	<ul style="list-style-type: none"> <li>- Mobilitas robot kurang fleksibel (masih tergantung pada jalur garis di lantai).</li> <li>- Jumlah makanan yang mampu diantarkan robot dalam satu waktu terbatas karena ukuran <i>tray</i> yang kecil.</li> </ul>	[17]
5	D-Bot: A Food Serving Robot During Pandemic Situation	<ul style="list-style-type: none"> <li>- <i>Meal compartment</i> dengan sistem konveyor vertikal yang tertutup oleh <i>chassis</i> dan dilengkapi dengan <i>slide door</i>.</li> <li>- Layar <i>touchscreen</i> dan RFID untuk sistem antarmuka pengguna.</li> <li>- ROS sebagai <i>middleware</i> antarmuka pengguna.</li> </ul>	<ul style="list-style-type: none"> <li>- Mekanisme konveyor vertikal cukup memakan waktu dalam menyajikan makanan ke pelanggan.</li> </ul>	[18]

		<ul style="list-style-type: none"> <li>- LIDAR untuk navigasi otonom dan <i>obstacle avoidance</i>.</li> <li>- Raspberry Pi 4 sebagai <i>main processor</i>.</li> </ul>		
6	Design and Implementation of a Robotic technique Based Waiter	<ul style="list-style-type: none"> <li>- <i>Rotary encoder</i> untuk odometri.</li> <li>- <i>Load cell unit</i> untuk mendeteksi keberadaan pesanan di atas <i>tray</i>.</li> <li>- Buzzer sebagai alarm notifikasi kepada pelanggan.</li> <li>- Aplikasi antarmuka untuk pengelola usaha dan pelanggan.</li> <li>- Modul bluetooth untuk komunikasi.</li> </ul>	<ul style="list-style-type: none"> <li>- Desain masih berupa prototipe.</li> <li>- Kendali pergerakan secara manual.</li> </ul>	[19]
7	Food Delivery Monitoring System with a Line Follower Robot	<ul style="list-style-type: none"> <li>- Sensor ultrasonik dan IR untuk <i>mapping</i> dan lokalisasi sehingga mampu bergerak secara otonom.</li> <li>- Aplikasi Blynx untuk memonitoring pesanan lewat ponsel.</li> </ul>	<ul style="list-style-type: none"> <li>- Mobilitas robot kurang fleksibel.</li> <li>- Desain masih berupa prototipe.</li> </ul>	[20]
8	Positioning and Navigating of Meal Delivery Robot Using Magnetic Sensor and RFID	<ul style="list-style-type: none"> <li>- Konsepnya sama seperti <i>line follower</i>, hanya saja jalur garis dibuat menggunakan <i>magnetic stripes</i> yang akan dideteksi oleh <i>magnetic sensor</i>.</li> <li>- RFID <i>reader</i> untuk mengidentifikasi lokasi meja pelanggan.</li> </ul>	<ul style="list-style-type: none"> <li>- Mobilitas robot kurang fleksibel.</li> <li>- Tidak ada sistem antarmuka.</li> </ul>	[21]
9	Development of a Robotic Waiter System	<ul style="list-style-type: none"> <li>- ROS untuk <i>path planning</i> dan navigasi otonom.</li> <li>- Intel NUC Mini PC untuk otak komputasi dan pemrosesan <i>framework ROS</i>.</li> <li>- Mbed sebagai mikrokontroler aktuator.</li> </ul>	<ul style="list-style-type: none"> <li>- Konsumsi daya yang lebih besar sehingga masa pakai cukup rendah.</li> <li>- Pemrograman yang lebih rumit untuk base segi empat dengan roda mekanum.</li> <li>- Robot sulit melakukan manuver dan</li> </ul>	[22]

		<ul style="list-style-type: none"> <li>- LIDAR dan IMU untuk stabilisasi dan posisi, navigasi, dan pengukuran kecepatan robot.</li> <li>- Encoder untuk perhitungan pergerakan, presisi.</li> </ul>	cenderung kurang stabil.	
10	Campus Canteen Takeaway Robot Based on Raspberry Pi	<ul style="list-style-type: none"> <li>- GPS untuk pemetaan dan navigasi.</li> <li>- Raspberry Pi 4 untuk komputasi.</li> <li>- LiDAR dan sensor ultrasonik untuk <i>obstacles avoidance</i>.</li> </ul>	<ul style="list-style-type: none"> <li>- Kontrol motor yang kurang akurat.</li> <li>- GPS berkemungkinan ketidakakuratan dalam penentuan lokasi.</li> </ul>	
11	Autonomous Service Robot	<ul style="list-style-type: none"> <li>- <i>Deep learning</i> untuk pendeketian objek dengan Arsitektur YOLO.</li> <li>- <i>Kinect sensor</i> untuk pemindaian 3D.</li> <li>- Raspberry Pi 3B+ untuk komputasi.</li> <li>- <i>Inverse Kinematic</i> untuk pengendalian lokasi yang akurat.</li> <li>- <i>ROS</i> untuk simulasi pemetaan lokasi robot.</li> </ul>	<ul style="list-style-type: none"> <li>- Pemilihan 2 motor DC beroda <i>bidirectional</i> membuat robot sulit bermanuver dan sulit untuk stabil.</li> </ul>	[23]
12	Development of a Robotic Waiter System for the Food and Beverage Industry	<ul style="list-style-type: none"> <li>- RFID untuk identifikasi lokasi meja.</li> <li>- ROS untuk simulasi pemetaan lokasi.</li> <li>- LIDAR untuk pemetaan dan navigasi robot.</li> <li>- Encoder untuk odometri posisi.</li> <li>- AHRS untuk orientasi robot.</li> </ul>	<ul style="list-style-type: none"> <li>- Pemilihan base dengan roda mekanum rawan bergetar</li> <li>- Gelombang IR membutuhkan waktu untuk mendapatkan data yg bagus sehingga memakan waktu lama dan sulit di-<i>tracking</i>.</li> <li>- Perlu memperhatikan posisi penempatan yang tepat untuk IR transmitter dan RFID.</li> </ul>	[24]
13	Non-Contact Service Robot Development in Fast-Food Restaurant	<ul style="list-style-type: none"> <li>- <i>LED display</i> untuk interaksi dengan pengguna via teks dan suara.</li> <li>- Metode <i>sensor fusion</i> (kamera RGBD, 2D laser, sensor</li> </ul>	<ul style="list-style-type: none"> <li>- Tidak ada produk fisik, hanya berupa simulasi dengan ROS – Gazebo.</li> <li>- <i>User interface</i> masih perlu dikembangkan supaya lebih mudah</li> </ul>	[25]

		ultrasonik) untuk mendapatkan informasi terkait <i>positioning</i> dengan menginterasikan <i>3D point cloud map</i> dan <i>2D occupancy grid map</i> (OGM).	dimengerti oleh khalayak umum.	
--	--	---	--------------------------------	--

Berdasarkan daftar penelitian terkait, kekurangan robot pengantar makanan yang telah dikembangkan sebelumnya adalah beberapa robot masih berupa prototipe (belum menggambarkan bentuk riil) atau pengembangan algoritma/simulasi saja. Adapun robot lainnya yang bukan berupa prototipe terkendala pada mobilisasinya yang kurang fleksibel sehingga robot tidak mampu bermanuver dengan baik. Desain mekanis yang masih terbatas juga menyebabkan jumlah makanan yang diangkut robot berjumlah sedikit sehingga diperlukan perkembangan dengan menerapkan desain berupa rak bersusun/tray. Robot juga belum memiliki mekanisme menyajikan makanan kepada pelanggan.

### 1.3 *Constraint*

#### 1.3.1 Aspek Ekonomi

Berdasarkan tabel 2.2.1, produk robot pengantar makanan yang diproduksi oleh Richtech Robotics, Bear Robotics, dan Pudu Robotics dipasarkan dengan harga mencapai ratusan juta rupiah. Hal ini belum termasuk biaya pajak, pengiriman, instalasi, serta pemeliharaan. Besarnya biaya yang perlu dikeluarkan tersebut masih tergolong tinggi bagi kafe, restoran, dan/atau kantin di Indonesia yang mayoritas berskala menengah ke bawah.

Oleh karena itulah, keberadaan produk robot pengantar makanan dengan harga yang lebih terjangkau akan sangat membantu sektor usaha ini. Diharapkan robot tersebut dikembangkan dengan dana yang lebih rendah, namun memiliki fitur yang tidak jauh berbeda dengan produk komersial yang telah beredar di *marketplace*.

#### 1.3.2 Aspek Manufakturabilitas (*manufacturability*)

Dari segi manufaktur, perancangan robot pengantar makanan diusahakan menggunakan bahan baku dan komponen penyusun yang cenderung mudah diperoleh serta banyak dijual di Indonesia dengan harga yang terjangkau. Proses produksi juga melibatkan peralatan yang relatif sederhana seperti *printer 3D*, mesin *laser cutting*,

gerinda, bor, dan lain sebagainya. Tak hanya itu, perancangan desain juga memperhatikan efektivitas kinerja dan nilai estetika robot tersebut.

### **1.3.3 Aspek Keberlanjutan (*sustainability*)**

Untuk menunjang aspek keberlanjutan, sangat penting bagi robot pengantar makanan agar dapat dipergunakan secara terus-menerus dengan masa pakai yang cukup lama selama dayanya terisi. Kemudian, robot dapat *di-charge* kembali jika telah kehabisan daya. Selain itu, robot juga tersusun atas bahan baku dan komponen dengan *durability* yang teruji. Hal ini juga didukung sistem penyusun yang sederhana sehingga memberikan kemudahan dalam penggunaan dan perbaikan/pemeliharaan secara berkala. Dengan demikian, robot pengantar makanan memungkinkan untuk dioperasikan dalam jangka panjang selama mengikuti prosedur penggunaan dengan baik dan benar. Namun tetap perlu diperhatikan bahwa robot ini hanya mampu dipergunakan di dalam ruangan dengan area jangkauan yang terbatas.

### **1.3.4 Aspek Estetika dan Desain**

Salah satu aspek yang perlu diperhatikan adalah aspek estetika dan desain sehingga robot pengantar makanan tetap menarik dipandang mata. Sehingga desain mekanis dari robot mampu meningkatkan daya tarik pelanggan. Desain ini juga menunjang keselamatan makanan agar tidak tercecer selama robot bergerak dan terjaga dari kontaminasi luar hingga sampai ke tangan pelanggan.

## **1.4 Kebutuhan yang Harus Dipenuhi**

Berdasarkan analisis yang telah dilakukan, robot pengantar makanan memerlukan sejumlah hal yang harus terpenuhi berdasarkan atas kebutuhan-kebutuhan pengguna yakni sebagai berikut:

1. Robot dapat bergerak menuju titik tujuan yang telah ditentukan di dalam ruangan pada bidang datar dan permukaan yang rata.
2. Robot mampu bergerak ke segala arah/*omni-directional*.
3. Robot mampu mengidentifikasi dan membedakan meja pelanggan dengan menggunakan kode QR.
4. Robot didesain dengan *trays* beserta penutup di sekelilingnya. Salah satu *tray* memiliki fitur untuk menyajikan makanan kepada pelanggan.

## **1.5 Tujuan**

Tujuan yang hendak dicapai dari penelitian *Capstone Design* ini adalah mampu mengantarkan makanan dengan tepat kepada pelanggan dengan kemampuan mobilisasi yang

mumpuni dan mendekripsi kode QR. Selain itu, desain mekanis robot ini juga bertujuan untuk menunjang estetika dan keselamatan makanan hingga sampai ke tangan pelanggan. Dengan demikian, pelayanan mengantar makanan pada kafe, restoran, dan/atau kantin akan lebih efektif dan efisien dari segi biaya (*cost*) dan kualitas (*quality*).

## **BAB 2**

### **SPESIFIKASI DAN VERIFIKASI**

Tabel 2.1 menunjukkan ringkasan kebutuhan yang harus terpenuhi oleh produk yang hendak dirancang.

**Tabel 2.1 Pemetaan Kebutuhan dan Kaitannya Terhadap Spesifikasi**

No	Kebutuhan	Spesifikasi
1	Robot dapat bergerak menuju titik tujuan yang telah ditentukan di dalam ruangan pada bidang datar dan permukaan yang rata.	Spesifikasi 1: Robot bergerak menuju ke titik koordinat tertentu dengan toleransi ketepatan 10 cm dan kecepatan linear maksimal sebesar $0.3 \frac{m}{s}$ .
2	Robot mampu bergerak ke segala arah/ <i>omni-directional</i> .	Spesifikasi 2: Robot mampu bergerak ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot.
3	Robot mampu mengidentifikasi dan membedakan meja pelanggan dengan menggunakan kode QR.	Spesifikasi 3: Jarak minimal pendekatan kode QR sebesar 30 cm.
4	Robot didesain dengan <i>trays</i> beserta penutup di sekelilingnya. Salah satu <i>tray</i> memiliki fitur untuk menyajikan makanan kepada pelanggan.	Spesifikasi 4: Tiap <i>tray</i> mampu menahan beban dengan berat maksimal sebesar 29,43 N (massa setara 3 kg). Spesifikasi 5: Salah satu <i>tray</i> mampu melakukan pergerakan slide ke luar dengan kecepatan maksimal $1 \frac{cm}{s}$ .

## 2.1 Spesifikasi Produk

Berdasarkan kebutuhan pada tabel 2.1, maka uraian spesifikasinya adalah sebagai berikut:

### 2.1.1 Pergerakan Menuju Titik Tujuan yang Telah Ditentukan di Dalam Ruangan pada Bidang Datar

Sama halnya seperti mekanisme kerja robot pengantar makanan yang dijual di *marketplace*, robot akan menerima makanan dari titik penjemputan (*pick-up point*). Kemudian pengguna dapat menentukan lokasi meja yang ditempati sang pelanggan sebagai titik tujuan. Untuk mendapatkan data mengenai titik tujuan, odometri menjadi metode navigasi yang dipilih. Metode ini menggunakan sensor pergerakan untuk mengukur perubahan posisi relatif terhadap posisi awal. Sehingga robot menghitung gerakan yang dibutuhkan untuk sampai pada titik tujuan yang telah ditetapkan .

Selama menuju titik tujuan ini, kecepatan linear menjadi salah satu parameter yang perlu diperhitungkan. Sebab jika kecepatan robot terlalu tinggi, maka makanan yang diangkut kemungkinan tertumpah/tercerer terutama untuk hidangan berkuah dan minuman [26]. Pada umumnya, robot pengantar makanan memiliki kecepatan linear minimal sebesar  $0.5 \frac{m}{s}$  [5], [6], [7]. Namun karena robot ini memiliki mekanisme mobilisasi yang berbeda dan belum dilengkapi dengan *braking system* yang mumpuni, maka kecepatan linearnya diatur menjadi lebih rendah. Hal ini dilakukan agar pergerakan robot tetap stabil. Adapun beberapa variabel seperti gesekan roda dengan lantai, *roller* roda yang slip, dan lain sebagainya mengakibatkan pergerakan robot terhambat dan kurang sesuai dengan kecepatan yang telah ditentukan. Oleh karena itu selain batasan kecepatan, juga ditentukan besarnya toleransi ketepatan robot terhadap titik tujuan yang diinginkan. Toleransi ini merupakan rentang nilai yang masih diterima untuk suatu ukuran yang telah ditentukan. Pertimbangan nilai toleransi berdasarkan besarnya ruang kosong di bawah meja dan sekitar kaki-kaki penopang yang dianggap sebagai *safe zone* bagi robot. Hal ini berarti selama masih berada di kisaran nilai toleransi, robot dapat berhenti dengan aman tanpa menabrak atau membentur kaki-kaki penopang meja. Maka dari itu, robot diatur agar dapat bergerak menuju titik koordinat tertentu dengan toleransi ketepatan 10 cm dan kecepatan linear maksimal sebesar  $0.3 \frac{m}{s}$ .

### 2.1.2 Pergerakan ke Segala Arah/*Omni-Directional*

Fleksibilitas dalam mobilisasi menjadi salah satu aspek penting yang perlu diperhatikan pada setiap *mobile robot* terutama robot yang diaplikasikan pada bidang pelayanan jasa

seperti robot pengantar makanan. Robot sejenis ini membutuhkan sistem penggerak yang mampu bermanuver dengan lincah. Maka dari itu, sebagian besar robot *service* merupakan robot *omni-directional* yang mampu bergerak ke segala arah (holonomik) tanpa harus memutar orientasi robot. Hal ini memudahkan robot dalam menghindari halang rintang baik yang bersifat statis seperti meja maupun dinamis seperti manusia yang bergerak di dalam kafe, restoran, dan/atau kantin terkait [6], [7], [27]. Kelebihan lain dari robot *omni-directional* adalah biaya dan daya yang diperlukan relatif lebih kecil. Selain itu, robot ini juga lebih hemat dalam penggunaan ruang [4] sehingga memungkinkan untuk beroperasi pada area yang sempit dan terbatas seperti lorong antar susunan meja di kafe, restoran, dan/atau kantin. Berdasarkan pertimbangan tersebut, robot mampu bergerak ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot.

### 2.1.3 Pengidentifikasi Meja Pelanggan Menggunakan Kode QR.

Kode QR (Quick Response Code) adalah jenis kode matriks dua dimensi yang memungkinkan penyimpanan dan pemindaian informasi secara cepat dan efisien. Kode ini terdiri dari modul hitam dan putih yang diatur dalam pola grid persegi, di mana setiap modul mewakili bit data. Ketika kode QR dipindai menggunakan perangkat seperti smartphone atau scanner QR, perangkat lunak dalam aplikasi pemindai akan mengidentifikasi pola tersebut dan menerjemahkannya menjadi informasi yang dapat dibaca, seperti URL, teks, atau data kontak. Proses ini memanfaatkan elemen-elemen seperti finder pattern, alignment pattern, dan timing pattern untuk memastikan kode dapat dikenali dan dibaca dari berbagai sudut, serta memiliki kemampuan koreksi kesalahan untuk memungkinkan pemindaian yang tetap berhasil meskipun sebagian kode rusak atau tertutup.

Keunggulan kode QR terletak pada kapasitas data yang tinggi, memungkinkan penyimpanan informasi yang lebih banyak dibandingkan dengan kode batang tradisional. Kode QR juga dirancang untuk pemindaian cepat, menjadikannya ideal untuk digunakan dalam situasi yang memerlukan akses informasi instan, seperti pemasaran, iklan, dan pembayaran digital. Selain itu, fleksibilitas penggunaan kode QR sangat luas, mencakup bidang seperti manajemen inventaris, tiket dan reservasi elektronik, serta berbagai aplikasi lain yang memerlukan interaksi digital yang cepat dan efisien. Kombinasi dari kemampuan penyimpanan data yang besar, kecepatan pemindaian, dan ketahanan terhadap kesalahan membuat kode QR menjadi alat yang sangat berguna dan serbaguna dalam berbagai aspek kehidupan modern.

#### **2.1.4 Desain *Tray* beserta Penutup di Sekelilingnya dan Salah Satu *Tray* Memiliki Fitur untuk Menyajikan Makanan kepada Pelanggan**

Pada robot ini, makanan akan diletakkan di atas *trays* yang dilingkupi dengan penutup di sekitarnya. Penutup bagian depan akan terbuka secara otomatis dengan mekanisme *slide* ke bawah begitu robot sampai pada meja pelanggan. Dengan demikian, pelanggan dapat mengambil makanan yang tersaji di atas *trays*. Adapun *trays* tersebut tersusun secara vertikal sebanyak 3 tingkat. Untuk kapasitas beban, tiap *tray* mampu membawa 2 porsi makanan dan minuman. Berdasarkan observasi yang dilakukan, massa totalnya sebesar 2.09 kg (belum termasuk sendok, garpu, serta perlengkapan makanan lainnya). Oleh karena itulah, tiap *tray* pada robot ini didesain agar mampu menahan beban dengan berat maksimal sebesar 29.43 N (massa setara 3 kg).

Adapun salah satu cara untuk menguji ketahanan desain 3 dimensi adalah *Finite Element Analysis* (FEA). *Finite element analysis* (FEA) adalah proses simulasi perilaku suatu bagian atau rakitan dalam kondisi tertentu. Simulasi fenomena fisik dengan FEA akan memberikan gambaran optimalisasi komponen tanpa harus menguji prototipe fisik [30]. Dalam kasus ini, metode FEA yang digunakan adalah *static stress* dimana sistem diberikan beban (*load*) statis pada titik dan besaran tertentu. Keluaran dari simulasi ini berupa *safety factor*. *Safety factor* didefinisikan sebagai rasio tegangan akhir (*ultimate stress*) dari bahan komponen terhadap tegangan kerja (*working stress*) [31]. Sistem dikatakan memiliki ketahanan yang mumpuni apabila memiliki *safety factor* lebih dari 1 [32].

Bagian teratas *tray* juga dirancang sedemikian rupa sehingga ketinggiannya bersesuaian dengan ketinggian meja. Serupa dengan penutup bagian depan yang mempergunakan mekanisme *slide* agar dapat terbuka dan tertutup secara otomatis, *tray* ini juga dapat melakukan pergerakan yang sama namun arahnya berbeda yakni ke luar. Oleh karena itu, *tray* ini perlu diatur kecepatan linearinya sebesar  $10 \frac{cm}{s}$  agar makanan yang diangkut tidak akan tertumpah dan tercercer karena pengaruh momentum selama *tray* bergerak.

## **2.2 Verifikasi**

Beberapa spesifikasi yang telah dirancang sebelumnya perlu diverifikasi untuk memastikan bahwa spesifikasi tersebut teruji kebenarannya dan bersesuaian dengan kebutuhan yang hendak dipenuhi.

### 2.2.1 Spesifikasi 1

**Tabel 2.2 Verifikasi Spesifikasi 1**

<b>Spesifikasi</b>	Robot bergerak menuju ke titik koordinat tertentu dengan toleransi ketepatan 10 cm dan kecepatan linear maksimal sebesar $0,3 \frac{m}{s}$ .
<b>Metode Pengukuran</b>	Pengukuran jarak yang ditempuh robot yang bergerak selama rentang waktu tertentu dengan menggunakan pita ukur dan stopwatch.
<b>Prosedur Pengujian</b>	<ol style="list-style-type: none"><li>1. Tentukan titik koordinat yang diinginkan dan tandai titik tersebut.</li><li>2. Rancang algoritma yang memungkinkan robot bergerak otomatis dari titik awal ke titik koordinat tersebut.</li><li>3. Hitung waktu pergerakan robot menggunakan stopwatch.</li><li>4. Begitu robot selesai bergerak, tandai titik akhir.</li><li>5. Kemudian, ukur titik akhir robot terhadap titik koordinat menggunakan pita ukur.</li><li>6. Kecepatan linear diperoleh dengan mengkalkulasikan perubahan posisi per waktu tempuh sampai ke titik akhir.</li></ol>

### 2.2.2 Spesifikasi 2

**Tabel 2.3 Spesifikasi 2**

<b>Spesifikasi</b>	Robot mampu bergerak ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot.
<b>Metode Pengukuran</b>	Keberhasilan robot bergerak ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot.
<b>Prosedur Pengujian</b>	Pengujian akan dilakukan dengan memasukkan 4 titik posisi yang harus dituju oleh robot (depan, belakang, kiri, dan kanan). Robot harus mampu bergerak menuju titik tersebut ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot.

### 2.2.3 Spesifikasi 3

**Tabel 2.4 Verifikasi Spesifikasi 3**

<b>Spesifikasi</b>	Jarak terjauh pendekripsi kode QR sebesar 25 cm.
<b>Metode Pengukuran</b>	Pengukuran jarak terjauh pendekripsi kode QR oleh kamera.
<b>Prosedur Pengujian</b>	<ol style="list-style-type: none"> <li>Siapkan kode QR yang akan digunakan.</li> <li>Tempatkan posisi awal kode QR sejauh 1 meter dari kamera.</li> <li>Dekatkan kode QR hingga kode QR terdeteksi pada program. Hal ini ditandai dengan munculnya bounding box yang melingkupi kode QR.</li> <li>Ukur jarak terjauh di mana kamera mampu mendekripsi kode QR menggunakan pita ukur.</li> </ol>

### 2.2.4 Spesifikasi 4

**Tabel 2.5 Verifikasi Spesifikasi 4**

<b>Spesifikasi</b>	Tiap <i>tray</i> mampu menahan beban dengan berat maksimal sebesar 29,43 N (massa setara 3 kg).
<b>Metode Pengukuran</b>	Pengujian ketahanan kerangka robot terhadap beban pada <i>tray</i> menggunakan simulasi FEA ( <i>Finite Element Analysis</i> ) <i>software</i> Autodesk Fusion 360.
<b>Prosedur Pengujian</b>	<ol style="list-style-type: none"> <li>Siapkan desain 3D robot pada <i>software</i> Autodesk Fusion 360.</li> <li>Pilih simulasi dengan metode <i>static stress</i>.</li> <li>Tentukan material penyusun robot pada bagian <i>materials</i>.</li> <li>Berikan <i>loads</i> dalam bentuk <i>force</i> sebesar 29,43 N (setara massa 3 kg).</li> <li>Tampilkan hasil simulasi berupa <i>safety factor</i> beserta diagram warnanya.</li> <li>Sistem dikatakan memenuhi syarat apabila <i>safety factor</i> berada di atas angka 1.</li> </ol>



## 2.2.5 Spesifikasi 5

**Tabel 2.6 Verifikasi Spesifikasi 5**

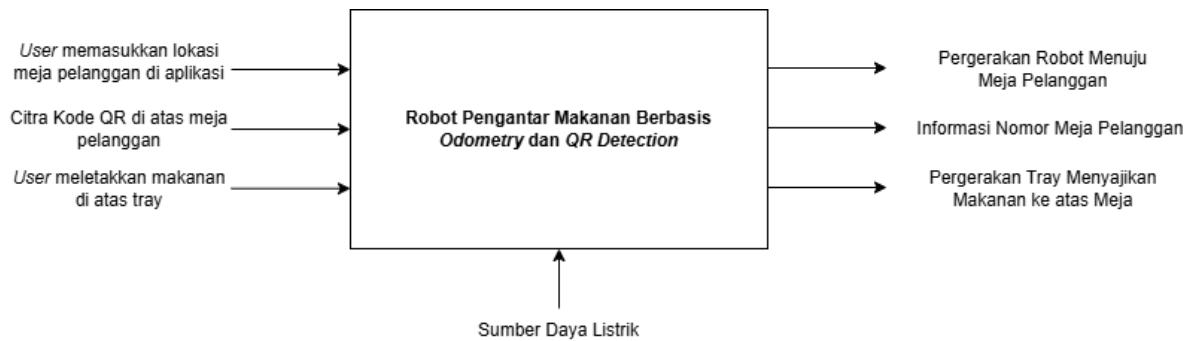
<b>Spesifikasi</b>	Salah satu <i>tray</i> mampu melakukan pergerakan slide ke luar dengan kecepatan linear maksimal $1 \frac{cm}{s}$ .
<b>Metode Pengukuran</b>	Pengukuran perpindahan <i>tray</i> selama bergerak rentang waktu tertentu dengan menggunakan stopwatch.
<b>Prosedur Pengujian</b>	<ol style="list-style-type: none"><li>1. Atur perpindahan <i>tray</i> hingga ke titik akhir.</li><li>2. Hitung waktu yang dibutuhkan <i>tray</i> untuk berpindah dari titik awal ke titik akhir dengan menggunakan stopwatch.</li><li>3. Besar kecepatan linear diperoleh dengan membagi perpindahan <i>tray</i> per waktu tempuh.</li></ol>

## BAB 3

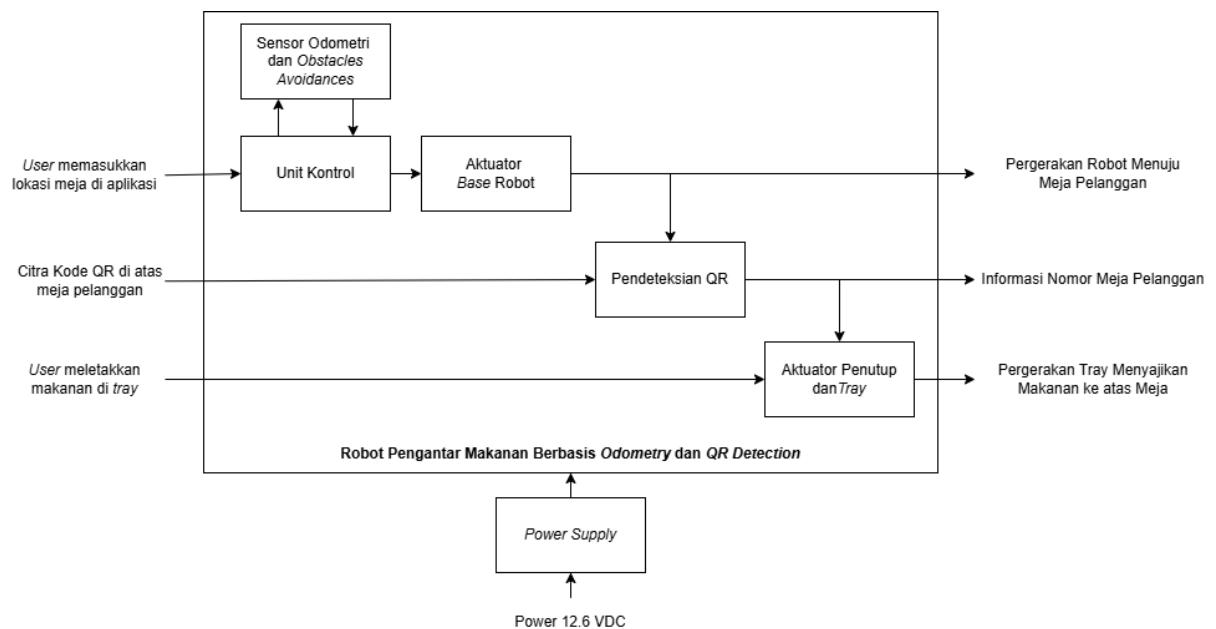
### DESAIN RANCANGAN SOLUSI

#### 3.1 Konsep Solusi

##### 3.1.1 Diagram Fungsi



**Gambar 4 Overall Function**



**Gambar 5 Function Tree**

Berdasarkan diagram *overall function* yang ditunjukkan pada gambar 3.1, sistem akan memiliki masukan berupa lokasi meja pelanggan, citra kode QR yang berada di atas meja pelanggan serta makanan yang diletakkan di atas *tray*. Dalam rincian *function tree* pada gambar 3.2, *user* akan memasukkan lokasi meja pelanggan pada aplikasi. Selanjutnya, unit kontrol akan memproses masukan tersebut sehingga aktuator *base* robot mulai bergerak. Selain itu, unit kontrol juga mengoperasikan sensor-sensor yang berperan dalam odometri dan

*obstacles avoidances*. Hasil proses ini akan dikirimkan kembali ke unit kontrol sehingga robot dapat mengetahui posisi relatif terhadap posisi awal.

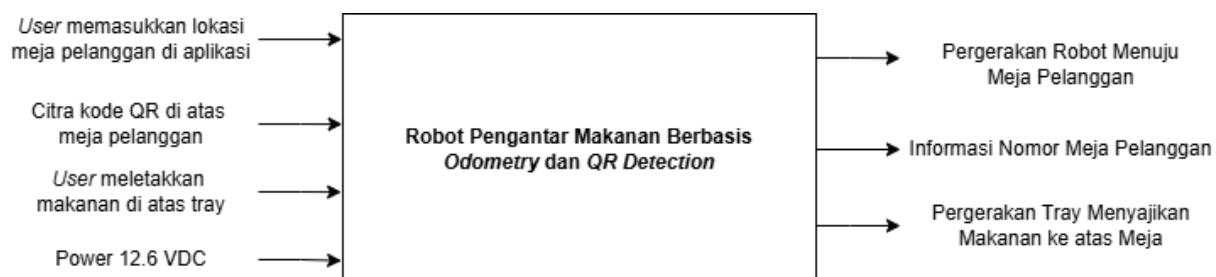
Setelah sampai di titik tujuan, unit kontrol akan mengirimkan perintah ke aktuator penggerak penutup dan *tray* untuk membuka penutup tray. Begitu terbuka, tray akan bergerak menyajikan makanan di atas meja pelanggan.

### 3.1.2 Karakteristik Solusi

- **Fitur Utama:**
  - Robot memiliki kemampuan untuk bergerak ke segala arah/*omni-directional* untuk menuju lokasi meja pelanggan yang telah ditentukan.
  - Robot mampu mendeteksi kode QR yang berada di atas meja pelanggan.
  - Salah satu *tray* pada robot mampu bergerak untuk menyajikan makanan ke atas meja pelanggan begitu sampai di lokasi yang dikehendaki.
- **Fitur Dasar:**
  - Metode odometri digunakan untuk memperkirakan posisi robot relatif terhadap posisi awal selama bergerak menuju meja pelanggan.
- **Fitur Tambahan:**
  - Robot memiliki kemampuan untuk menghindari objek yang menghalangi dengan mencari jalur yang kosong lalu kembali lagi ke jalur awal.

## 3.2 Rencana Desain dari Konsep Solusi Sistem

### 3.2.1 Diagram Blok Level 0



Gambar 6 Diagram Blok Level 0

Tabel 3.1 Rincian Diagram Blok Level 0

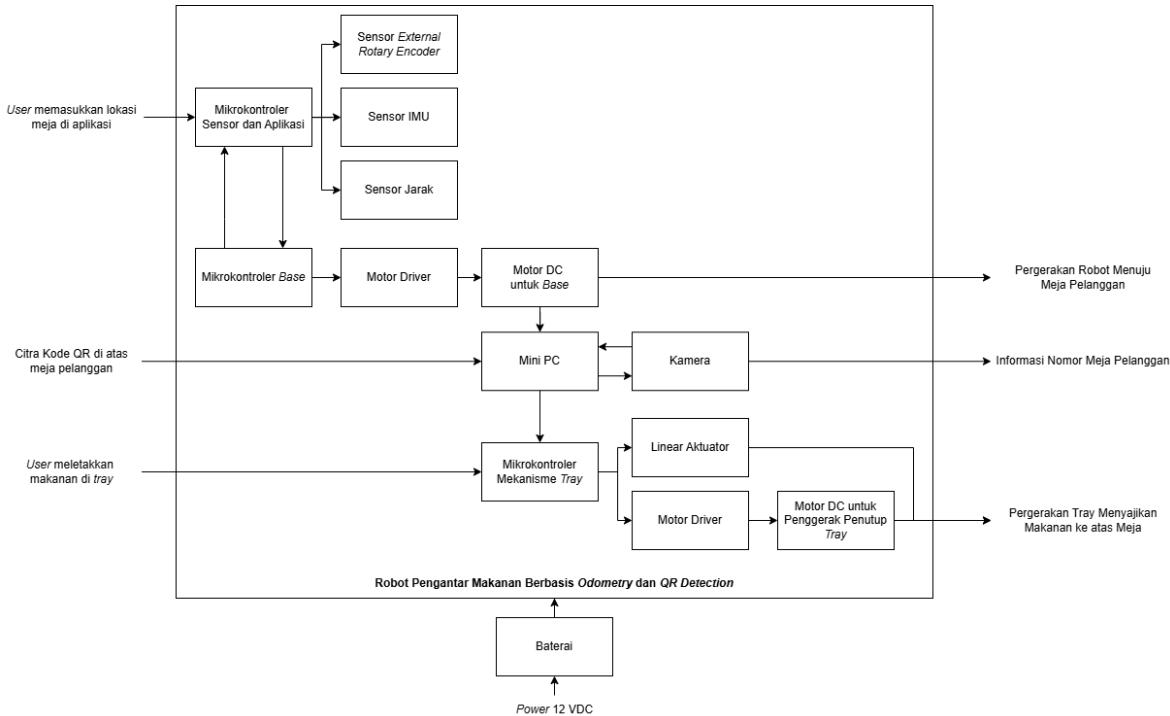
<b>Modul</b>	Robot Pengantar Makanan Berbasis <i>Odometry</i> dan <i>QR Detection</i>
<b>Input</b>	<ul style="list-style-type: none"> <li>- User memasukkan lokasi meja pelanggan di aplikasi</li> <li>- Citra kode QR di atas meja pelanggan</li> </ul>

	<ul style="list-style-type: none"> <li>- <i>User</i> meletakkan makanan di atas <i>tray</i></li> <li>- <i>Power</i> 12.6 VDC</li> </ul>
<b><i>Output</i></b>	<ul style="list-style-type: none"> <li>- Pergerakan robot menuju meja pelanggan</li> <li>- Informasi nomor meja pelanggan</li> <li>- Pergerakan <i>tray</i> menyajikan makanan ke atas meja</li> </ul>
<b><i>Functionality</i></b>	Mengantarkan makanan menuju meja pelanggan yang telah ditentukan <i>user</i> , kemudian menyajikan makanan yang telah diangkut di atas <i>tray</i> ke atas meja pelanggan.

Sesuai diagram blok yang disajikan pada gambar 3.3, sistem utama yang akan dirancang adalah modul robot pengantar makanan berbasis *odometry* dan *QR detection*. Sistem ini memiliki 4 masukan yakni lokasi meja pelanggan, makanan, citra kode QR di atas meja pelanggan, serta *power* 12.6 VDC.

Masukan pertama yang akan diberikan adalah *power* 12 VDC yang akan menyala sistem pada robot. Kemudian pengguna akan menaruh makanan yang hendak diantarkan ke atas *tray* dan memilih lokasi meja pelanggan pada aplikasi yang telah disediakan. Setelah tiga masukan ini diberikan, keluaran pertama yang diperoleh adalah pergerakan robot menuju meja pelanggan. Begitu sampai di titik tersebut, robot akan menangkap citra kode QR yang diletakkan di atas meja pelanggan sebagai masukan terakhir untuk mendapatkan informasi mengenai nomor meja tersebut. Apabila sesuai, aktuator penutup dan *tray* akan beroperasi sehingga menghasilkan pergerakan *tray* menyajikan makanan di atas meja.

### 3.2.2 Diagram Blok Level 1



**Gambar 7 Diagram Blok Level 1**

**Tabel 3.2 Rincian Diagram Blok Level 1**

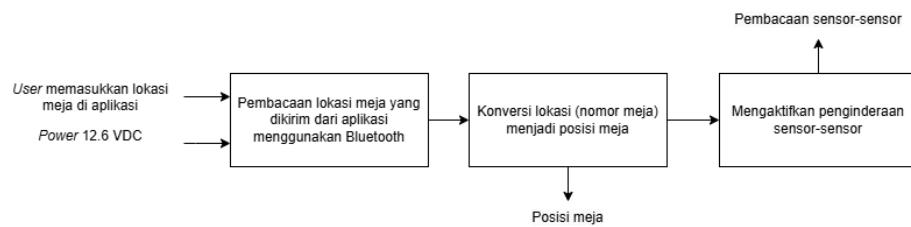
Modul	Robot Pengantar Makanan Berbasis <i>Odometry</i> dan <i>QR Detection</i> .
<i>Input</i>	<ul style="list-style-type: none"> <li>- <i>User</i> memasukkan lokasi meja pelanggan di aplikasi</li> <li>- Citra kode QR di atas meja pelanggan</li> <li>- <i>User</i> meletakkan makanan di atas <i>tray</i></li> <li>- <i>Power</i> 12.6 VDC</li> </ul>
<i>Output</i>	<ul style="list-style-type: none"> <li>- Pergerakan robot menuju meja pelanggan</li> <li>- Informasi nomor meja pelanggan</li> <li>- Pergerakan <i>tray</i> menyajikan makanan ke atas meja</li> </ul>
<i>Functionality</i>	Mengantarkan makanan menuju meja pelanggan yang telah ditentukan <i>user</i> , kemudian menyajikan makanan yang telah diangkut di atas <i>tray</i> ke atas meja pelanggan.

Diagram blok level 1 di atas menjabarkan proses kerja robot pengantar makanan berbasis *odometry* dan *QR detection* dengan lebih spesifik. *User* memberi masukan berupa lokasi pelanggan di aplikasi dan makanan yang diletakkan pada *tray*. Informasi mengenai lokasi meja diproses oleh mikrokontroler sensor dan aplikasi. Mikrokontroler ini akan mengirimkan perintah kepada mikrokontroler *base* untuk menggerakkan aktuator *base* yakni motor DC. Selain itu, mikrokontroler ini juga akan menjalankan sensor *rotary*, jarak, dan

IMU untuk melakukan komputasi untuk *odometry* dan *obstacles avoidances*. Begitu sampai di lokasi tujuan, mikrokontroler mekanisme *tray* akan mulai beroperasi. Sehingga motor DC akan bergerak membuka penutup *tray* sebelum linear aktuator mendorong *tray* keluar. Dengan begitu, makanan dapat tersaji kepada pelanggan.

### 3.2.3 Diagram Blok Level 2

#### 3.2.3.1 Level 2.1

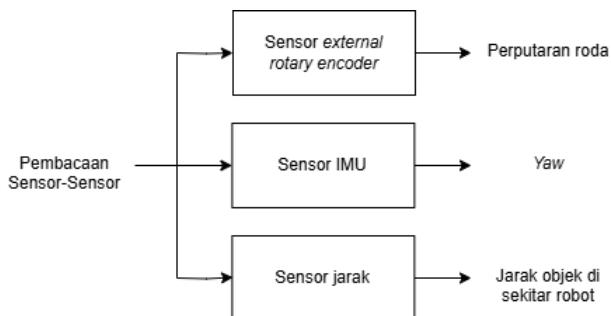


**Gambar 8 Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi**

**Tabel 3.3 Rincian Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi**

<b>Modul</b>	Mikrokontroler sensor dan aplikasi.
<b>Input</b>	User memasukkan lokasi meja di aplikasi pada robot dan <i>power</i> 12.6 VDC.
<b>Output</b>	Pembacaan sensor-sensor dan posisi meja.
<b>Functionality</b>	Menerima lokasi (nomor) meja pelanggan dan mengonversinya menjadi posisi, serta mengaktifkan penginderaan sensor-sensor.

Begitu *user* memasukkan lokasi meja pelanggan berupa nomor meja di aplikasi dan sistem diberi *power* 12.6 VDC, mikrokontroler sensor dan aplikasi akan mulai menerima lokasi meja yang dikirim menggunakan Bluetooth. Lokasi meja kemudian akan diterjemahkan menjadi posisi meja. Selain itu, mikrokontroler ini juga akan mengaktifkan pembacaan sensor-sensor.



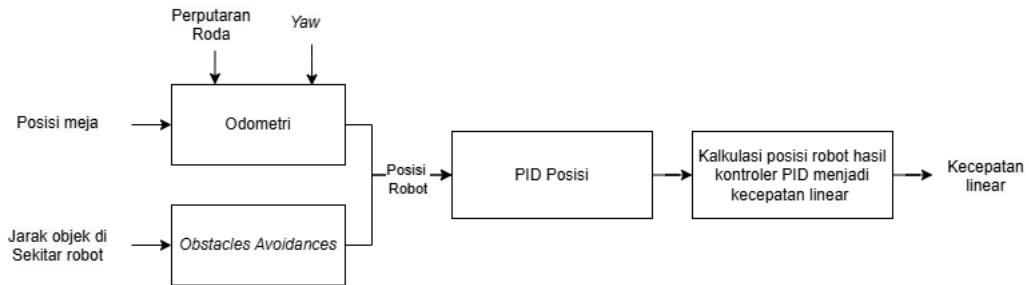
**Gambar 9 Diagram Blok Level 2 pada Sensor *External Rotary Encoder*, IMU, dan jarak**

**Tabel 3.4 Rincian Diagram Blok Level 2 pada Sensor *External Rotary Encoder*, IMU, dan jarak**

<b>Modul</b>	Sensor <i>external rotary encoder</i> , IMU, dan jarak.
<b>Input</b>	Pembacaan sensor-sensor.
<b>Output</b>	Perputaran roda, yaw, jarak objek di sekitar robot.
<b>Functionality</b>	Memperoleh data-data untuk komputasi odometri dan <i>obstacles avoidance</i> .

Perintah pembacaan sensor akan dikirimkan ke sensor *external rotary encoder*, IMU, dan jarak untuk melakukan penginderaan. Sensor *external rotary encoder* digunakan untuk menghitung banyaknya putaran yang ditempuh roda, sementara sensor IMU (*inertial measurement unit*) digunakan untuk mendeteksi perubahan *yaw*. Dalam konteks dinamika pergerakan robot, *yaw* merujuk pada rotasi atau gerakan berputar robot di sekitar sumbu vertikalnya. Sensor jarak sendiri berguna untuk mengetahui jarak antara robot dengan objek yang berada di sekitar. Ketiga data yang diperoleh akan dikomputasi pada mikrokontroler sensor dan aplikasi.

### 3.2.3.2 Level 2.2

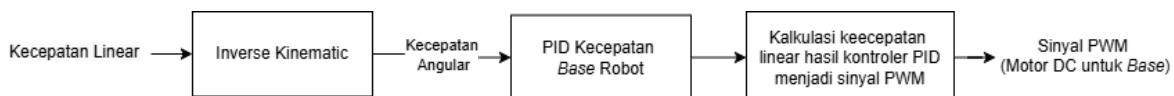


**Gambar 10 Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi Bagian 2**

**Tabel 3.5 Rincian Diagram Blok Level 2 pada Mikrokontroler Sensor dan Aplikasi Bagian 2**

<b>Modul</b>	Mikrokontroler Sensor dan Aplikasi Bagian 2.
<b>Input</b>	Posisi meja, perputaran roda, yaw, dan jarak objek di sekitar robot.
<b>Output</b>	Kecepatan linear.
<b>Functionality</b>	Melakukan komputasi odometri dan <i>obstacles avoidances</i> serta PID posisi untuk kontrol posisi robot.

Pada level 2.2, ketiga data yang diterima dari pembacaan sensor-sensor serta posisi meja dikomputasi di mikrokontroler sensor dan aplikasi. Data posisi meja, perputaran roda, dan *yaw* digunakan dalam metode odometri untuk memperkirakan posisi robot relatif terhadap posisi awal serta posisi meja pelanggan yang dituju. Sementara itu, jarak objek di sekitar robot digunakan untuk *obstacles avoidances* sehingga selama pergerakan, robot tidak akan menabrak objek yang berada di sekitarnya. Selanjutnya dilakukan komputasi PID posisi untuk mengendalikan posisi robot. Hasil kontroler PID posisi kemudian dikalkulasi sehingga menghasilkan keluaran kecepatan linear.



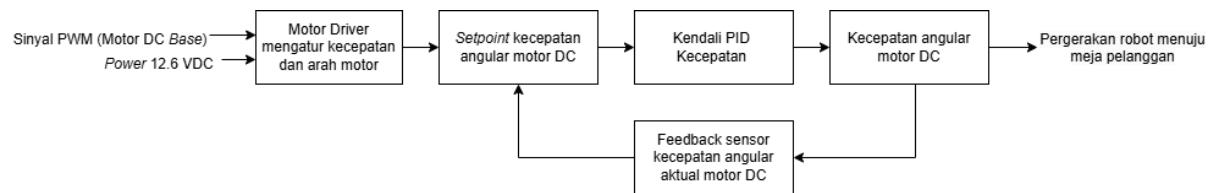
**Gambar 11 Diagram Blok Level 2 pada Mikrokontroler Base**

**Tabel 3.6 Rincian Diagram Blok Level 2 pada Mikrokontroler Base**

<b>Modul</b>	Mikrokontroler Base.
<b>Input</b>	Kecepatan linear.
<b>Output</b>	Sinyal PWM (Motor DC untuk Base).
<b>Functionality</b>	Melakukan komputasi <i>inverse kinematic</i> dan PID kecepatan motor DC untuk base serta mengirim sinyal PWM ke motor driver.

Setelah memperoleh nilai kecepatan linear dari mikrokontroler sensor dan aplikasi, mikrokontroler *base* akan memproses perhitungan PID kecepatan untuk mengontrol kecepatan motor DC *base*. Kecepatan linear hasil kontroler PID kecepatan kemudian

dikalkulasi menjadi sinyal PWM (*pulse width modulation*). Lalu, mikrokontroler ini akan mengirimkan sinyal tersebut kepada motor driver.



**Gambar 12 Diagram Blok Level 2 pada Motor Driver dan Motor DC**

**Tabel 3.7 Rincian Diagram Blok Level 2 pada Motor Driver dan Motor DC**

<b>Modul</b>	Motor driver dan motor DC
<b>Input</b>	Sinyal PWM
<b>Output</b>	Pergerakan robot menuju meja pelanggan.
<b>Functionality</b>	Mengendalikan arah dan kecepatan pergerakan robot agar stabil hingga sampai ke titik tujuan yang dikehendaki (meja pelanggan).

Sinyal PWM dikirimkan mikrokontroler ke motor driver yang terhubung dengan motor DC sebagai aktuator robot. Motor driver berfungsi untuk mengatur kecepatan dan arah motor berdasarkan besarnya sinyal PWM yang diberikan. Selanjutnya agar kendali kecepatan robot tetap stabil, digunakan kendali PID (*proportional, integral, derivative*) kecepatan sesuai *setpoint* kecepatan angular tiap motor. Kendali ini akan menghasilkan kecepatan angular. Kecepatan angular aktual tiap motor akan kembali dideteksi oleh sensor encoder internal sebagai bentuk umpan balik untuk komputasi kendali PID selanjutnya.

### 3.2.3.3 Level 2.3

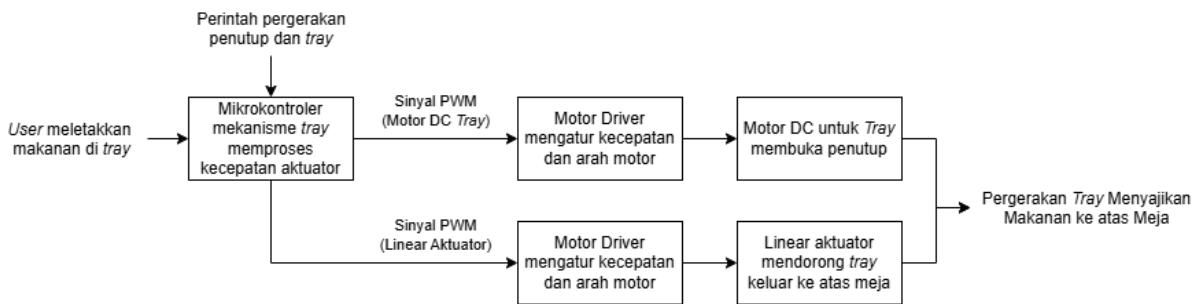


**Gambar 13 Diagram Blok Level 2 pada Kamera dan Mini-PC**

**Tabel 3.8 Rincian Diagram Blok Level 2 pada Kamera dan Mini-PC**

<b>Modul</b>	Kamera dan Mini-PC.
<b>Input</b>	Citra QR di atas meja pelanggan.
<b>Output</b>	Informasi nomor meja pelanggan.
<b>Functionality</b>	Mengidentifikasi apakah meja yang telah dituju sesuai dengan nomor meja pelanggan yang dimasukkan pengguna ke aplikasi.

Setelah robot sampai di titik tujuan, sistem pendekripsi QR mulai beroperasi. Kamera diaktifkan untuk menangkap citra QR. Selanjutnya hasil tangkapannya akan diolah di Mini-PC sehingga akan diperoleh informasi nomor meja pelanggan.



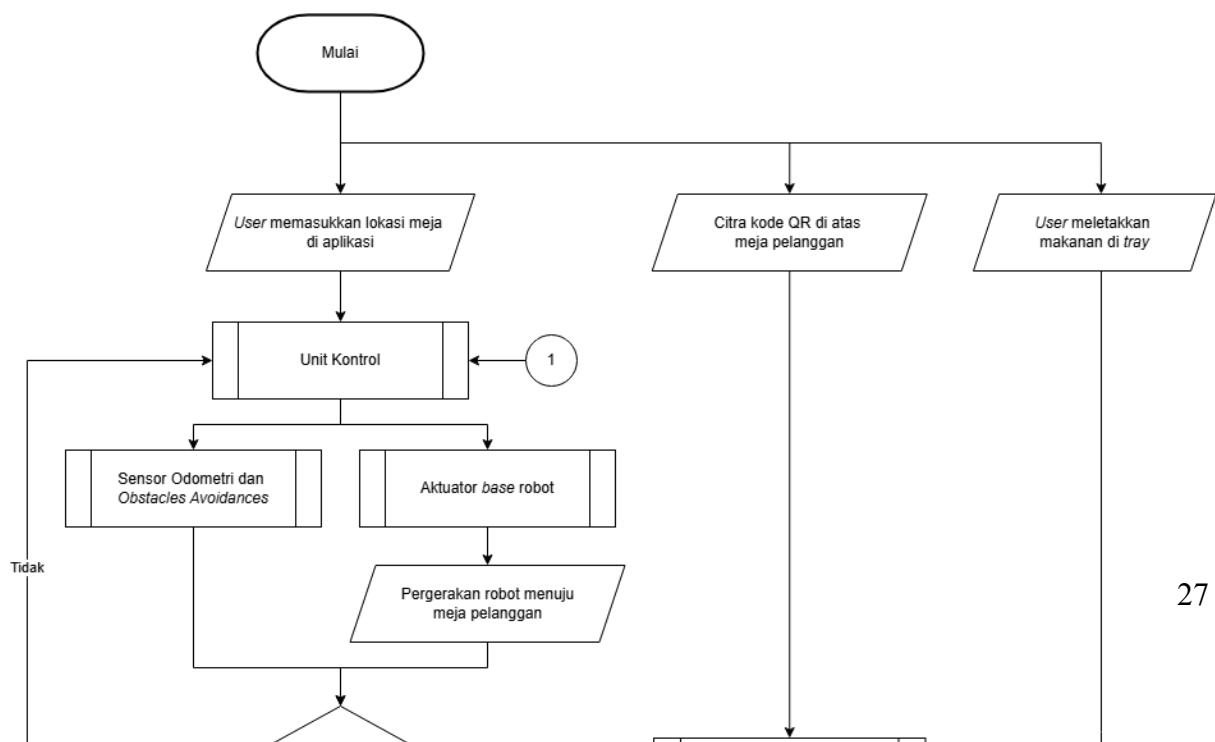
**Gambar 14 Diagram Blok Level 2 pada Mikrokontroler Mekanisme *Tray*, Motor Driver, Motor DC untuk *Tray*, dan Linear Aktuator**

**Tabel 3.9 Rincian Diagram Blok Level 2 pada Mikrokontroler Mekanisme *Tray*, Motor Driver, Motor DC untuk *Tray*, dan Linear Aktuator**

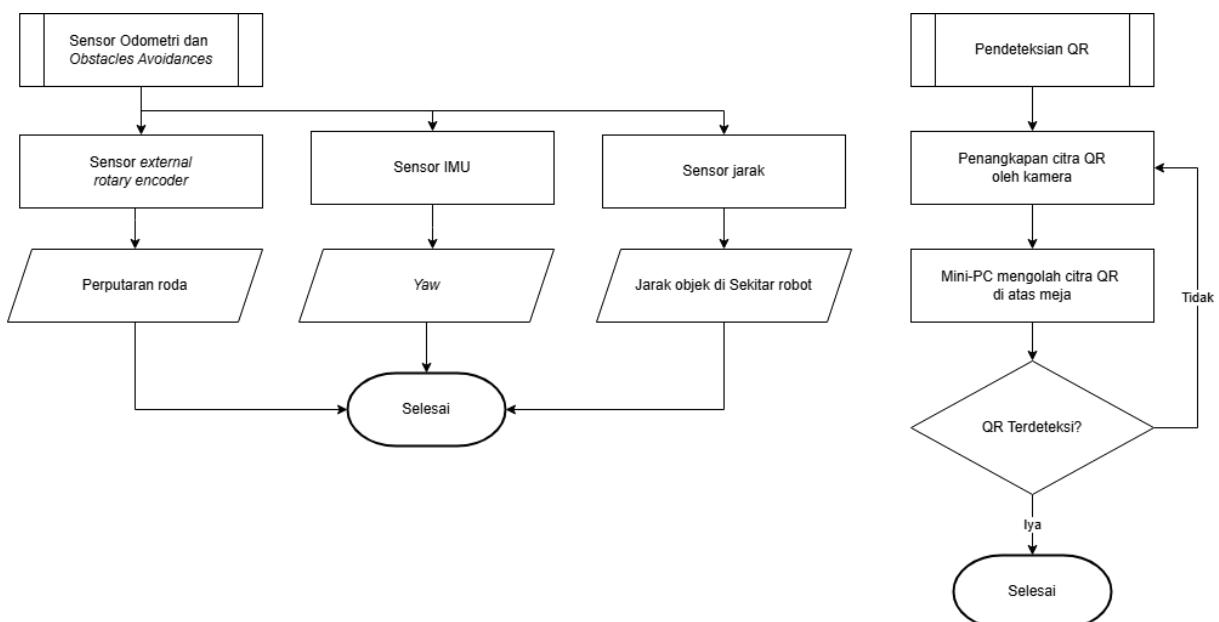
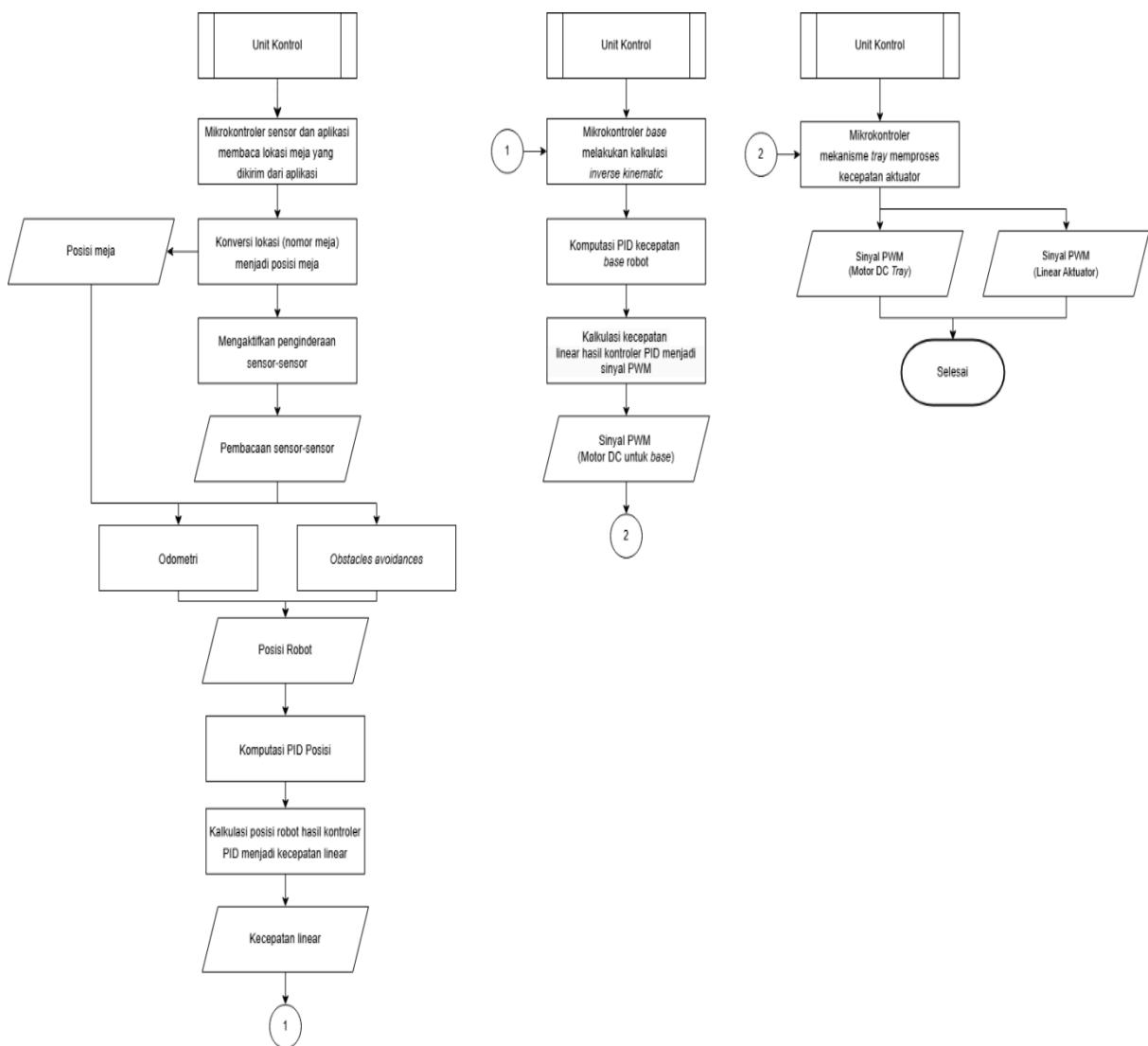
<b>Modul</b>	Mikrokontroler Mekanisme <i>Tray</i> , Motor Driver, Motor DC untuk <i>Tray</i> , dan Linear Aktuator.
<b>Input</b>	User meletakkan makanan di <i>tray</i> dan perintah pergerakan penutup dan <i>tray</i> .
<b>Output</b>	Pergerakan <i>tray</i> menyajikan makanan di atas meja.
<b>Functionality</b>	Mengangut dan menyajikan makanan ke atas meja pelanggan

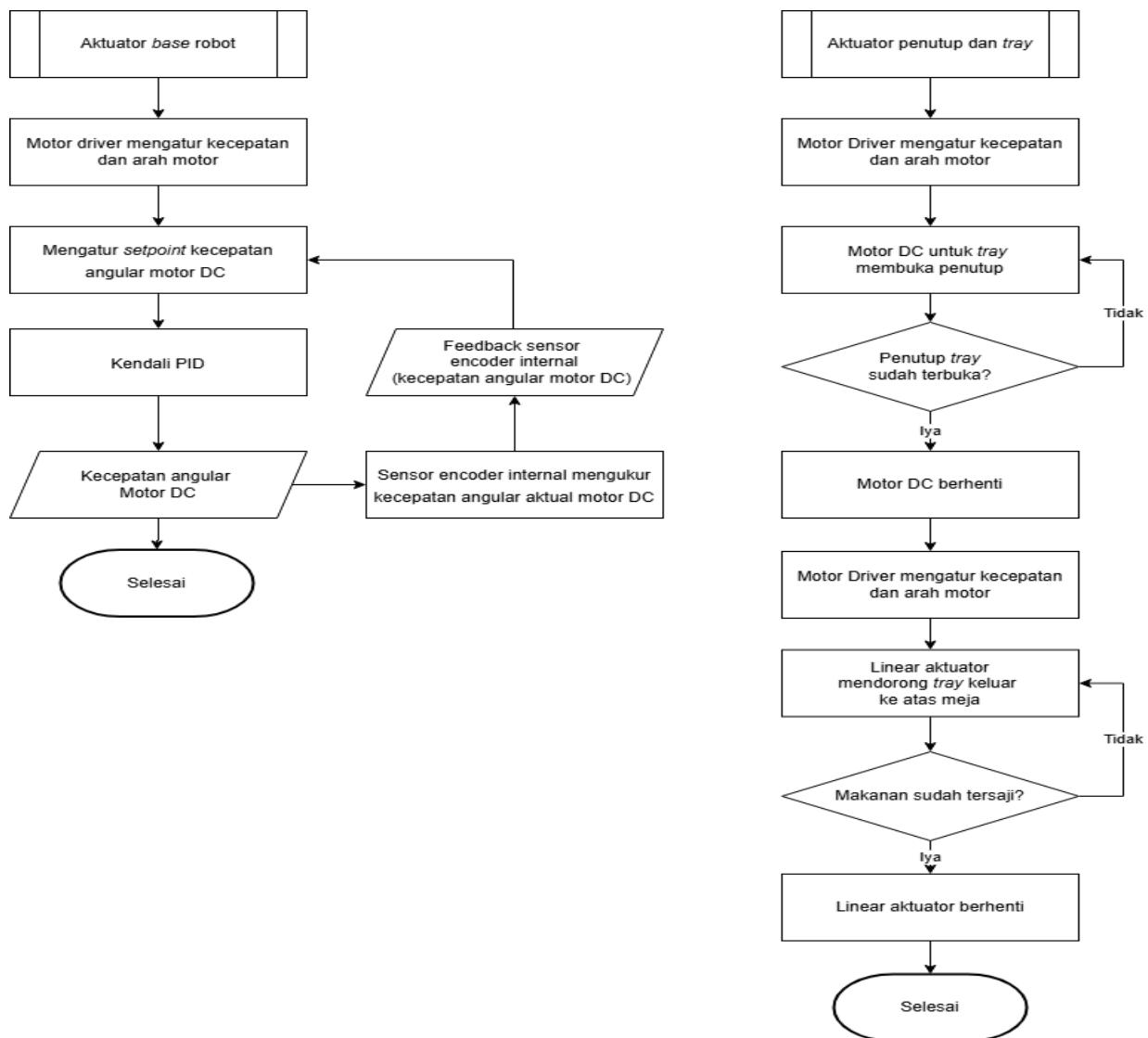
Di level terakhir saat robot telah sampai di titik tujuan, mikrokontroler mekanisme *tray* akan mengirimkan sinyal PWM kepada motor driver yang mengontrol motor DC untuk membuka penutup *tray* sehingga *tray* berada dalam kondisi terbuka. Sesaat setelah itu, sinyal PWM dari mikrokontroler mekanisme *tray* dikirim ke motor driver yang mengatur arah dan kecepatan linear aktuator. Linear aktuator akan mendorong *tray* ke luar hingga makanan yang diletakkan di atasnya akan tersaji ke atas meja pelanggan.

### 3.2.4. Flowchart









Gambar 15 Flowchart Sistem

Berdasarkan diagram alir atau *flowchart* sistem pada Gambar 3.12, *user* akan memasukkan lokasi meja di aplikasi kemudian meletakkan makanan yang hendak diantar di atas *tray*. Lokasi meja kemudian akan dikirim ke unit kontrol menggunakan komunikasi Bluetooth. Selanjutnya Lokasi meja akan dikonversi menjadi posisi meja. Lalu, unit kontrol akan mengaktifkan sensor pada sub-sistem sensor odometri dan *obstacles avoidances* untuk melakukan penginderaan. Hasil pembacaan sensor-sensor pada sub-sistem ini akan diolah kembali oleh unit kontrol untuk odometri dan *obstacles avoidances* sehingga mendapatkan posisi robot relatif terhadap posisi awal. Untuk memastikan agar posisi robot stabil, dilakukan proses komputasi PID posisi. Keluaran kontroler PID posisi lalu diubah menjadi kecepatan linear.

Selanjutnya unit kontrol memberikan perintah kepada sub-sistem aktuator *base* robot untuk menggerakan *base* robot. Pergerakan inilah yang membawa robot menuju ke meja pelanggan. Begitu sampai di meja pelanggan, sub-sistem pendektsian QR akan beroperasi. Pada sub-sistem pendektsian QR, kamera akan menangkap citra kode QR yang diletakkan di atas meja pelanggan. Setelah itu, mini-PC mengolah hasil tangkapan dan memperoleh informasi nomor meja pelanggan. Apabila nomor meja pelanggan bersesuaian dengan nomor meja yang dipilih *user*, unit kontrol akan mengirim perintah kepada sub-sistem aktuator penutup dan *tray*. Hal inilah yang memicu motor DC bekerja membuka penutup tray sebelum akhirnya linear aktuator mendorong *tray* ke luar sampai makanan tersaji di atas meja pelanggan.

Di sisi lain, apabila nomor meja pelanggan tidak sesuai, maka unit kontrol akan kembali mengendalikan robot untuk bergerak dan melakukan penginderaan kembali hingga menuju titik lainnya.

### 3.3 Pemilihan Komponen

Berdasarkan diagram blok level 1 pada gambar 3.4, berikut pemilihan komponen berdasarkan unsur-unsur pembentuknya.

#### 3.3.1 Mikrokontroler Sensor dan Aplikasi

Dalam sistem ini, robot membutuhkan mikrokontroler yang memungkinkan adanya komunikasi menggunakan protokol Bluetooth untuk konektivitas ke aplikasi. Tak hanya komunikasi Bluetooth, jenis komunikasi lain yang didukung mikrokontroler juga perlu diperhatikan sebab akan digunakan untuk konektivitas dengan dua mikrokontroler lain (mikrokontroler *base* dan mekanisme *tray*). Selain itu, mikrokontroler ini diharapkan memiliki jumlah GPIO dan clock speed yang tinggi sehingga mampu melakukan komputasi yang cepat terhadap data yang diterima dari sensor-sensor. *Operating voltage* dan harga komponen juga perlu disesuaikan.

Maka, beberapa kriteria pemilihan mikrokontroler untuk sensor dan aplikasi ini adalah operating voltage, jumlah GPIO, protokol Bluetooth, jenis komunikasi lain yang didukung, clock speed, dan harga.

**Tabel 3.10 Pemilihan Komponen Mikrokontroler Sensor dan Aplikasi**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		ESP32 Dev-Kit V4	Arduino Nano 33 BLE	STMicroelectronics BlueNRG-2
<i>Operating Voltage</i>	10	3.3V	3.3V	1.7V – 3.6V
		0.1	0.1	0.1

Jumlah GPIO	20	36 0.2	22 0	15 -0.2
Protokol	25	Bluetooth 4.2 dan BLE	Bluetooth 5.0 dan BLE	Bluetooth 5.0
		0.25	0.25	0.25
Jenis Komunikasi	10	Wi-Fi, SPI, UART, I2C	SPI, UART, I2C	SPI, UART, I2C
		0.1	0.1	0.1
Clock Speed	20	160 MHz 0.2	64 MHz 0	32 MHz -0.2
		Rp67.500,00 0.15	Rp640.000,00 -0.15	Rp250.000,00 -0.15
Total	100	1	0.5	-0.1

Komponen terpilih adalah **ESP32 Dev-Kit V4** karena unggul dalam jumlah GPIO, memiliki *clock speed* yang tinggi, namun dengan harga yang lebih terjangkau.

### 3.3.2 Mikrokontroler Base

Mikrokontroler base berfungsi untuk mengendalikan keseluruhan sistem pada base robot sehingga dibutuhkan mikrokontroler dengan jenis core yang memungkinkan proses komputasi yang cepat. Hal ini ditandai dengan frekuensi core yang besar. Selain itu, hal lain yang perlu diperhatikan adalah jumlah pinout, hardware interrupt, komunikasi (UART dan I2C) dan PWM (pulse width modulation). Semua spesifikasi ini hendaknya diiringi dengan harga komponen yang terjangkau.

Oleh karena itu, sejumlah kriteria pemilihan mikrokontroler base adalah core, core frequency, pinout, operating voltage, hardware interrupt, jumlah pin komunikasi (UART dan I2C) dan PWM, serta harga.

**Tabel 3.11 Pemilihan Komponen Mikrokontroler Base**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		NUCLEO-STM32 F401RE	STM32 F411CEU6 -WeAct Blackpill V2.0	NUCLEO-STM32 F446RE
Core	10	ARM® 32-bit Cortex®-M4 CPU with FPU	ARM® 32-bit Cortex®-M4 CPU with FPU	ARM® 32-bit Cortex®-M4 CPU with FPU
		0.1	0.1	0.1
Core Frequency	15	84 MHz	100 MHz	180 MHz
		0	0.15	0.15
Pinout	15	50	40	50
		0.15	0	0.15
Operating voltage	10	7V – 15V	3.52V – 5.25V	7V – 12V
		0.1	0.1	0.1

<i>Hardware Interrupt</i>	10	7	GPIO with external interrupt capability	7
		0.1	0	0.1
Jumlah pin UART	5	4	3	2
		0.05	0	-0.05
Jumlah pin I2C	5	3	3	4
		0.05	0.05	0.05
Jumlah pin PWM	10	47	25	47
		0.1	0	0.1
Harga	20	Rp350.000,00	Rp106.500,00	RP789.500,00
		-0.2	0.2	-0.2
Total	100	0.45	0.6	0.55

Komponen yang terpilih adalah **STM32 F411CEU6-WeAct Blackpill V2.0** karena jumlah pinout serta pin komunikasi (UART dan I2C) serta pin PWM yang cukup banyak dibanding mikrokontroler sejenis namun dengan harga yang lebih terjangkau.

### 3.3.3 Mikrokontroler Mekanisme *Tray*

Sama halnya seperti spesifikasi mikrokontroler *base* dan sensor dan aplikasi, mikrokontroler mekanisme *tray* juga memerlukan jumlah GPIO, jumlah pin PWM, dan *clock speed* yang tinggi karena akan mengendalikan pergerakan motor stepper dan linear aktuator. Jumlah pin UART juga menjadi pertimbangan karena mikrokontroller ini akan terhubung dengan mikrokontroler lain menggunakan komunikasi serial. Namun tetap berada dalam kisaran *operating voltage* yang mumpuni bagi sistem. Semua kriteria ini hendaknya didukung dengan harga yang terjangkau.

Maka pemilihan mikrokontroler mekanisme *tray* dilakukan berdasarkan kriteria *operating voltage*, *clock speed*, jumlah GPIO, jumlah pin PWM, jumlah pin UART, dan harga.

**Tabel 3.12 Pemilihan Komponen Mikrokontroler Mekanisme *Tray***

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Arduino Uno R3	Arduino Nano V3	Arduino Mega 2560 R3
<i>Operating Voltage</i>	15	5V	5V	5V
		0.15	0.15	0.15
<i>Clock Speed</i>	20	16 MHz	16 MHz	16 MHz
		0.2	0.2	0.2
Jumlah GPIO	15	14 Digital, 6 Analog	14 Digital, 8 Analog	54 Digital, 16 Analog
		0.15	0.15	0.15
Jumlah pin PWM	15	6	6	15
		0.15	0.15	0.15

Jumlah pin UART	15	1 0	1 0	6 0.15
Harga	20	Rp63.000,00 0.2	Rp70.000,00 0	Rp180.000,00 -0.2
Total	100	0.85	0.65	0.6

Komponen terpilih adalah **Arduino Uno R3**. Meskipun ketiga pilihan mikrokontroler tersebut memiliki spesifikasi yang hamper sama, harga yang lebih terjangkau menjadi keunggulan Arduino Uno R3 dibandingkan yang lainnya.

### 3.3.4 Mini PC

Mini PC digunakan untuk komputasi sistem pendekripsi kode QR yang akan menjadi penanda antar satu meja dan yang lainnya. Komponen ini juga akan mengolah data lanMaka dibutuhkan Mini PC dengan tipe CPU (*Central Processing Unit*) dan GPU (*Graphics Processing Unit*) yang cepat serta penyimpanan yang mencukupi. Di sisi lain, banyaknya I/O, besarnya *operating voltage*, koneksiitas serta harga juga patut menjadi kriteria tambahan yang dipertimbangkan.

**Tabel 3.13 Pemilihan Konponen Mini PC**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Jetson Nano Recomputer J1010 Seed Studio	Raspberry Pi 4 Model B	Raspberry Pi 5 Model B
Tipe CPU	10	NVIDIA Maxwell™ architecture with 128 NVIDIA CUDA® cores 0.5 TFLOPs (FP16)	Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8GHz	Broadcom BCM2712 2.4GHz quad-core 64-bit Arm Cortex-A76 CPU, with cryptography extensions, 512KB per-core L2 caches and a 2MB shared L3 cache
		0.1	0.1	0.1
Tipe GPU	10	Quad-core ARM® Cortex®-A57 MPCore processor	VideoCore VI	VideoCore VII GPU, supporting OpenGL ES 3.1, Vulkan 1.2
		0.1	0.1	0.3
I/O	20	1 SDIO, 2 SPI, 2 I2C, 2 I2S dan GPIO	Raspberry Pi standard 40 pin GPIO header	Raspberry Pi 40-pin GPIO header
		0.2	0	0
GPU Memory	15	4 GB 64-bit LPDDR4	-	-

		1600MHz - 25.6 GB/s		
		0.15	-0.15	-0.15
Penyimpanan	15	16 GB eMMC 5.1 Flash	8 GB LPDDR4-3200 SDRAM	LPDDR4X-4267 SDRAM (4GB and 8GB SKUs available at launch)
		0.15	0	0
<i>Operating Voltage</i>	5	5V DC via USB-C connector (minimum 3A*)	5V DC via USB-C connector (minimum 3A*)	5V/5A DC power via USB-C, with Power Delivery support
		0.05	0.05	0.05
Konektivitas	5	Wi-Fi requires external chip dan - 10/100/1000 BASE-T Ethernet	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE Gigabit Ethernet	2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE Gigabit Ethernet
		0.05	0	0
Harga	20	Rp4.450.000,00	Rp3.150.000,00	Rp2.150.000,00
		0	0.2	0.2
Total	100	0.8	0.3	0.5

Komponen terpilih adalah **Jetson Nano Recomputer J1010 Seeed Studio** karena didukung tipe GPU dan CPU dengan pemrosesan yang cepat dan kuat, serta GPU memori dan kapasitas penyimpanan yang lebih besar.

### 3.3.5 Sensor *External Rotary Encoder*

Sensor *external rotary encoder* berfungsi untuk mengukur perputaran roda selama robot bergerak. Oleh karena itulah, pemilihan komponen ini didasarkan pada beberapa kriteria yakni resolusi, *pinout*, *operating voltage*, serta harga.

**Tabel 3.14 Pemilihan Komponen Sensor External Rotary Encoder**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		External Rotary Encoder 600 ppr	DIY Built-in PG45	DIY Optical Rotary Encoder from Built-in PG45
Resolusi	40	600 ppr	7 ppr	200 ppr
		0.4	-0.4	0
<i>Pinout</i>	10	Phase A, Phase B, VCC, GND	Phase A, Phase B, VCC, GND	Phase A, phase B, VCC, GND
		0.1	0.1	0.1
<i>Operating voltage</i>	20	5V – 24V	3.3V – 5V	3.3V – 5V
		0.2	0.2	0.2

Harga	30	Rp232.500,00	Rp150.000,00	Rp350.000,00
		0	0.3	-0.3
Total	100	0.7	0.2	0

Komponen yang terpilih adalah **External Rotary Encoder 600 ppr** karena resolusi pembacaan yang tinggi. Resolusi pembacaan yang tinggi akan berdampak pada ketelitian posisi robot.

### 3.3.6 Sensor IMU

Sensor IMU berguna untuk mengetahui percepatan gerak robot selama mengantar makanan menuju lokasi tujuan. Sehingga prioritas utama dalam pemilihan sensor IMU adalah axis dan resolusi ADC sebab kedua kriteria ini akan mempengaruhi hasil pembacaan sensor. Selain itu, kriteria lain yang menjadi pertimbangan adalah *operating voltage*, *interface* yang didukung, serta harga.

**Tabel 3.15 Pemilihan Komponen Sensor IMU**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		BNO055	MPU6050	MPU9250
<i>Operating Voltage</i>	10	2.4V – 6V	2.375V – 3.46V	2.4V – 3.6V
		-0.1	0.1	0
<i>Interface</i>	10	I2C, UART	I2C serial interface only	I2C port
		0.1	0	0
<i>Axis</i>	25	9-axis	6-axis	9-axis
		0.25	0.25	0.25
Resolusi ADC	30	14-bit accelerometer, a triaxial 16-bit gyroscope	16-bit ADC	16-bit ADC
		0.3	0.3	0.3
Harga	25	Rp868.000,00	Rp29.500,00	Rp100.000,00
		-0.25	0.25	0
Total	100	0.3	0.9	0.55

Komponen yang terpilih adalah **MPU6050** sebab memiliki resolusi ADC dan *axis* yang mumpuni, namun membutuhkan tegangan yang lebih rendah dan harga yang terjangkau.

### 3.3.7 Sensor jarak

Sensor jarak berguna untuk mendeteksi jarak robot dengan objek yang berada di sekitarnya sehingga robot dapat menghindari objek tersebut selama bergerak menuju titik tujuan (*obstacles avoidance*). Oleh karena itu, sensor jarak dipilih berdasarkan kriteria *input voltage*, jarak pendektsian, dan harga.

**Tabel 3.16 Pemilihan Komponen Sensor Jarak**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Sensor Ultrasonik HCSR-04	Sensor Ultrasonik SRF-04	Ultrasonic US100
<i>Operating Voltage</i>	20	5V	5V	5V
		0.2	0.2	0.2
Jarak Deteksi	40	2 cm – 450 cm	3 cm – 600 cm	2 cm – 450 cm
		0.4	0.4	0.4
Harga	40	Rp9.000,00	Rp479.050,00	Rp50.000,00
		0.4	-0.4	0
Total	100	1	0.2	0.6

Komponen yang terpilih adalah **Sensor Ultrasonik HCSR-04** karena memiliki jarak deteksi yang mencukupi dengan harga yang lebih terjangkau dibandingkan sensor sejenis.

### 3.3.8 Kamera

Kamera berfungsi untuk menangkap citra objek yang berada di kafe, restoran, dan/atau kantin yang menjadi masukan bagi sistem pendekripsi objek sehingga diperlukan kamera dengan resolusi maksimum yang tinggi dan besar MP yang tinggi sehingga kualitas citra yang ditangkap jernih dan tajam. Kriteria lainnya yang perlu diperhatikan adalah jenis fokus, bidang pandang, dan harga.

**Tabel 3.17 Pemilihan Komponen Kamera**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Logitech C310 Webcam	Logitech C920 HD Pro Webcam	Razer Kiyo X
Resolusi Maksimum	25	720p/30fps	720p/30fps	1080p/30fps, 720p/60fps, 480p/30fps, 360p/30fps
		0.25	0.25	0.25
Besar MP	25	0.9	3	2.1
		0	0.25	0.25
Jenis Fokus	10	<i>Fixed focus</i>	<i>Auto focus</i>	<i>Auto focus</i>
		0	0.1	0.1
Bidang Pandang	10	55 derajat	78 derajat	82 derajat
		0.1	0.1	0.1
Harga	30	Rp270.000,00	Rp999.000,00	Rp649.000,00
		0.3	-0.3	-0.3
Total	100	0.65	0.4	0.4

Komponen terpilih adalah **Logitech C310 Webcam** karena resolusi maksimum dan besar MP yang mencukupi dengan harga yang lebih terjangkau.

### 3.3.9 Motor DC untuk Base

Motor DC digunakan sebagai penggerak *base* robot. Sehingga komponen ini harus memiliki besar torsi dan kecepatan sesuai kebutuhan yang hendak dipenuhi. Dimensi panjang dan diameter serta massa juga diperhatikan untuk menghemat penggunaan ruang dan meringankan beban robot. Kriteria lain yang perlu diperhatikan adalah besar *operating voltage*, daya yang digunakan, serta harga.

**Tabel 3.18 Pemilihan Komponen Motor DC untuk Base**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		PG45 19.2K 24V 7 ppr encoder	PG45 13K 24V 200 ppr encoder	PG45 10K 24V 200 ppr encoder
<i>Operating Voltage</i>	10	24 VDC	24 VDC	24 VDC
		0.1	0.1	0.1
Daya	10	60 W	60 W	60 W
		0.1	0.1	0.1
Torsi Maksimum	20	12kgf.cm – 25 kgf.cm	10 kgf.cm – 20 kgf.cm	23.86 kgf.cm
		0.2	0.2	0.2
Kecepatan Tanpa Beban ( <i>No Load Speed</i> )	20	468.75 RPM	650 RPM	2400 RPM
		0.2	0	-0.2
Dimensi Panjang	10	139.8 mm	139.8 mm	139.8 mm
		0.1	0.1	0.1
Diameter	10	45 mm	45 mm	45 mm
		0.1	0.1	0.1
Massa	10	850 gram	850 gram	850 gram
		0.1	0.1	0.1
Harga	10	Rp900.000,00	Rp1.200.000,00	Rp1.000.000,00
		0.1	-0.1	-0.1
Total	100	1	0.6	0.4

Komponen terpilih adalah **PG45 19.2K 24V 7 ppr Encoder** karena memiliki spesifikasi torsi dan kecepatan yang mencukupi dengan harga yang lebih terjangkau.

### 3.3.10 Motor Driver

Fungsi motor driver adalah mengendalikan kecepatan dan arah motor sehingga pemilihan komponen ini didasarkan pada beberapa kriteria yakni besarnya *operating voltage*, *continuous current*, *peak current*, *pinout* yang digunakan, dan harga.

**Tabel 3.19 Pemilihan Komponen Motor Driver**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		BTS7960	Cytron MDS30C	Driver Motor 50A 48V by MRI
<i>Operating Voltage</i>	15	6V – 27V	5V – 25V	48 V
		0.15	0.15	-0.15

<i>Continuous Current</i>	25	35A	30A	50A
		0.35	0	0.25
<i>Peak Current</i>	20	43A	80A	50A
		0	0.2	0.2
<i>Pinout</i>	10	PWM, EN_RIGHT, EN_LEFT, VCC, GND	PWM, EN, VCC, GND	PWM, DIR1, DIR2
		0.1	0	0
Harga	30	Rp65.000,00	Rp550.000,00	Rp600.000,00
		0.3	-0.3	-0.3
Total	100	0.8	0.25	0

Komponen terpilih adalah **BTS7960** karena disesuaikan dengan kebutuhan motor DC yang digunakan yakni Motor DC PG45 19.2K 24V 7 ppr encoder. Harga komponen ini juga lebih murah dibandingkan Cytron dan Motor Driver by MRI.

### 3.3.11 Baterai

Baterai dipilih berdasarkan kriteria kapasitas, tegangan, massa, dimensi, serta harga.

**Tabel 3.20 Pemilihan Komponen Baterai**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		SMT-Power Aki Elektronik 12V 7Ah	Lithium Lifepo4 24V 12Ah	ONBO 6s
Kapasitas	25	7000 mAh	12000 mAh	2200 mAh
		0.25	0.25	-0.25
Tegangan	20	12 VDC	24 VDC	22.2 VDC
		0	0.2	0.2
Massa	20	3000 gram	2000 gram	350 gram
		-0.2	-0.2	0.2
Dimensi	20	64 mm x 94 mm x 151 mm	68 mm x 112 mm x 130 mm	112mm x 35mm x 49mm
		0.2	-0.2	0
Harga	15	Rp205.000,00	Rp1.460.000,00	Rp659.000,00
		0.15	-0.15	-0.15
Total	100	0.4	0.1	0

Komponen terpilih adalah **SMT-Power Aki Elektronik 12V 7Ah** karena bersesuaian dengan suplai daya yang dibutuhkan untuk menghasilkan kecepatan aktuator yang maksimal. Kapasitas baterai yang cukup besar memungkinkan robot mampu beroperasi dalam jangka waktu yang lebih panjang. Baterai yang dipilih juga memiliki dimensi dan massa yang lebih kecil sehingga tidak memakan begitu banyak ruang dan mengurangi beban pada robot.

### 3.3.12 Linear Aktuator

Linear aktuator berguna untuk mendorong *tray* ke atas meja pelanggan sehingga kriteria pemilihan linear aktuator adalah *stroke*, *drive voltage*, kecepatan, gaya dorong maksimum, dan harga.

**Tabel 3.21 Pemilihan Komponen Linear Aktuator**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Electric Linear Actuator 100 mm Stroke	Electric Linear Actuator 150 mm Stroke	Electric Linear Actuator 300 mm Stroke
<i>Stroke</i>	40	100 mm	150 mm	300 mm
		-0.4	0	0.4
<i>Drive Voltage</i>	10	12 V	12 V	12 V
		0.1	0.1	0.1
Kecepatan	25	10 mm/s	6 mm/s	5 mm/s
		0.25	0	0
Gaya Dorong maksimum	10	1500 N	1500 N	6000 N
		0.1	0.1	0.1
Harga	15	Rp550.000,00	Rp590.000,00	Rp795.000,00
		0.15	0	-0.15
Total	100	0.2	0.2	0.45

Komponen terpilih adalah **Electric Linear Actuator 300 mm Stroke** karena memiliki jangkauan *stroke* yang lebih tinggi.

### 3.3.13 Motor DC untuk Penggerak Penutup *Tray*

Sama halnya seperti penggerak *base*, motor DC juga diperlukan sebagai aktuator untuk penutup *tray*. Namun, motor DC yang digunakan untuk penggerak penutup *tray* hendaknya berukuran lebih kecil dengan besar kecepatan dan torsi yang mumpuni. Maka, beberapa kriteria yang menjadi pertimbangan dalam pemilihan komponen motor DC untuk penggerak penutup *tray* adalah *operating voltage*, kecepatan, torsi, dimensi, dan harga.

**Tabel 3.22 Pemilihan Komponen Motor DC untuk Penggerak Penutup *Tray***

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Motor DC PG28 7 ppr	Motor DC GA25-370	Motor DC 27GB-520
<i>Operating Voltage</i>	10	6V – 12V	6V – 12V	6V – 12V
		0.1	0.1	0.1
Torsi Maksimum	20	0.4 Nm	0.35 Nm	0.45 Nm
		0	-0.2	0.2
Kecepatan Tanpa Beban ( <i>No Load Speed</i> )	20	4500 RPM	4000 RPM	4200 RPM
		0.2	-0.2	0

Dimensi (Diameter)	10	28 mm	25 mm	37 mm
		0	0.1	-0.1
Harga	10	Rp194.850,00	Rp157.500,00	Rp236.250,00
		0	0.1	-0.1
Total	100	0.3	-0.1	0.1

Komponen terpilih adalah **Motor DC PG28** 7 ppr karena memiliki kecpatan tanpa beban (*no load speed*) yang paling tinggi serta torsi yang mumpuni untuk menggerakan penutup *tray*.

### 3.3.14 Roda *Omniwheel*

Roda omniwheel adalah salah satu tipe roda yang memungkinkan robot bermobilisasi dan bermanuver ke segala arah. Dalam memilih roda *omniwheel* yang tepat, sejumlah kriteria yang perlu diperhatikan antara lain diameter, *load capacity*, massa, jumlah roller, dan harga.

**Tabel 3.23 Pemilihan Komponen Roda *Omniwheel***

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Roda <i>Omniwheel</i> 127 mm	Roda <i>Omniwheel</i> 152 mm	Roda <i>Omniwheel</i> 100 mm
Diameter	30	127 mm	152 mm	100 mm
		0.3	0.3	-0.3
<i>Load Capacity</i>	15	30 kg/set	40 kg/set	30 kg/set
		0.15	0.15	0.15
Massa	20	480 gram	630 gram	335 gram
		0.2	-0.2	0.2
Jumlah Roller	15	22	28	18
		0.15	0.15	-0.15
Harga	20	Rp1.496.358,00/set	Rp1.528.027,00/set	Rp1.430.000,00/set
		0	-0.2	0.2
Total	100	0.8	0.2	0.1

Komponen terpilih adalah **Roda *Omniwheel* 127 mm**. Hal ini dikarenakan bersesuaian dengan dimensi robot yang besar. Selain itu, roda ini juga memiliki load capacity yang mumpuni sehingga mampu menahan massa robot. Jumlah *roller* yang mencukupi juga memungkinkan pergerakan robot menjadi lebih halus.

### 3.3.15 Kerangka alumunium

Robot akan tersusun atas kerangka alumunium. Material ini ringan namun kokoh. Alumunium juga bersifat tidak korosif sehingga tahan lama. Ada beberapa jenis alumunium yang umumnya digunakan sebagai kerangka yakni alumunium V-slot, hollow, dan hollow brown. Kriteria pemilihannya adalah dimensi, ukuran celah/ketebalan, massa, dan harga.

**Tabel 3.24 Pemilihan Komponen Kerangka Alumunium**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Alumunium V-Slot	Alumunium Hollow	Alumunium Hollow Brown
Dimensi	25	2 cm x 2 cm	2 cm x 2 cm	2 cm x 2 cm
		0.25	0.25	0.25
Ukuran celah/ketebalan	35	6 mm	2 mm	1.2 mm
		0.35	0	-0.35
Massa	15	2.7 kg/600 cm	2.13 kg/600 cm	1.47 kg/600 cm
		0.15	0.15	0.15
Harga	25	Rp330.000,00 /600 cm	Rp240.000,00 /600 cm	Rp210.000,00 /600 cm
		0	0.25	0.25
Total	100	0.75	0.65	0.3

Komponen terpilih adalah **Alumunium V-Slot**. Meskipun ketiga memiliki dimensi yang sama, akan tetapi ukuran celah/ketebalan alumunium v-slot lebih tebal sehingga dinilai lebih kokoh dibandingkan jenis alumunium lainnya.

### 3.3.16 Material Penutup

Penutup *tray* merupakan salah satu bagian yang penting yang berperan dalam melindungi makanan dari kontaminasi luar selama robot mengantarkannya ke meja pelanggan. Sehingga material penyusun penutup tray harus diperhatikan. Adapun kriteria pemilihan penutup tray antara lain kekuatan, kemudahan pemrosesan, massa, estetika, daya tahan, dan harga.

**Tabel 3.25 Pemilihan Komponen Material Penutup**

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Infraboard	Akrilik	PVC Board
Kekuatan	15	Cukup kuat, mampu menahan benturan ringan	Sangat kuat, tahan pecah	Kuat, tahan terhadap benturan dan tekanan
		0.15	0.15	0.15
Kemudahan Pemrosesan	20	Mudah dipotong dan dipasang, dapat dibentuk	Sulit dipotong dan dibentuk, membutuhkan alat khusus	Mudah dipotong dan dibentuk, dapat dilas
		0.20	-0.20	0.20
Massa	15	Ringan	Berat	Sedang
		0.15	-0.15	0
Estetika	15	Finishing yang baik, tersedia dalam berbagai warna	Transparan dan berkilau, estetika tinggi	Permukaan halus, variasi warna terbatas

		0.15	0.15	0
Daya Tahan	20	Tahap terhadap air dan kelembapan, tidak tahan lama	Tahan terhadap cuaca dan sinar UV, awet	Tahan air dan cuaca, awet
		0	0.20	0
Harga	15	Rp20.000,00 /3 mm x 85 cm x 60 cm	Rp30.000,00/ 4 mm x 10 cm x 10 cm	Rp56.000,00 /5 mm x 60 cm x 80 cm
		0.15	-0.15	0
Total	100	0.85	0	0.35

Komponen terpilih adalah **Infraboard** karena cukup kuat, mudah diproses, massanya lebih ringan dengan estetika dan daya tahan yang mumpuni.

### 3.3.17 Material *Tray*

Selain kerangka robot dan penutup, jenis material *tray* penting untuk diperhatikan agar penutup dan *tray* yang dirancang ringan namun tetap kuat untuk menahan beban makanan yang diletakkan di atasnya serta memiliki tampilan yang menarik.

**Tabel 3.26 Pemilihan Komponen Material *Tray***

Kriteria	Bobot (%)	Skor Penilaian (1, 0, -1)		
		Akrilik	Plat Alumunium	Papan Kayu
Tampilan dan Tekstur	25	Berwarna/tidak berwarna dan halus	Solid dan tekstur menyesuaikan kebutuhan	Serat kayu dan kasar/halus
		0.25	0.25	0
Kekuatan dan Ketahanan	25	Tahan terhadap tekanan dan goresan	Tahan terhadap benturan	Kurang kuat terhadap benturan keras
		0.25	0.25	0
Massa dan Densitas	20	Lebih ringan dengan densitas lebih rendah	Lebih berat namun kuat	Lebih berat dengan densitas bervariasi sesuai ukuran
		0.2	0	0
Ketebalan	10	5 mm	5 mm	5 mm
		0.1	0.1	0.1
Harga	20	Rp75.000,00 per 29cm x 34cm	Rp260.000 per 38cm x 43cm	Rp58.000,00 per 20cm x 400 cm
		0	-0.2	0.2
Total	100	0.8	0.4	0.3

Komponen terpilih adalah **Akrilik** karena akrilik lebih unggul dibanding material lainnya dalam hal ketahanan terhadap tekanan dan goresan, namun dengan massa yang lebih ringan.

Sehingga berdasarkan *decision matrix* pemilihan setiap komponen, tabel 3.27 menunjukkan rangkuman hasil komponen terpilih.

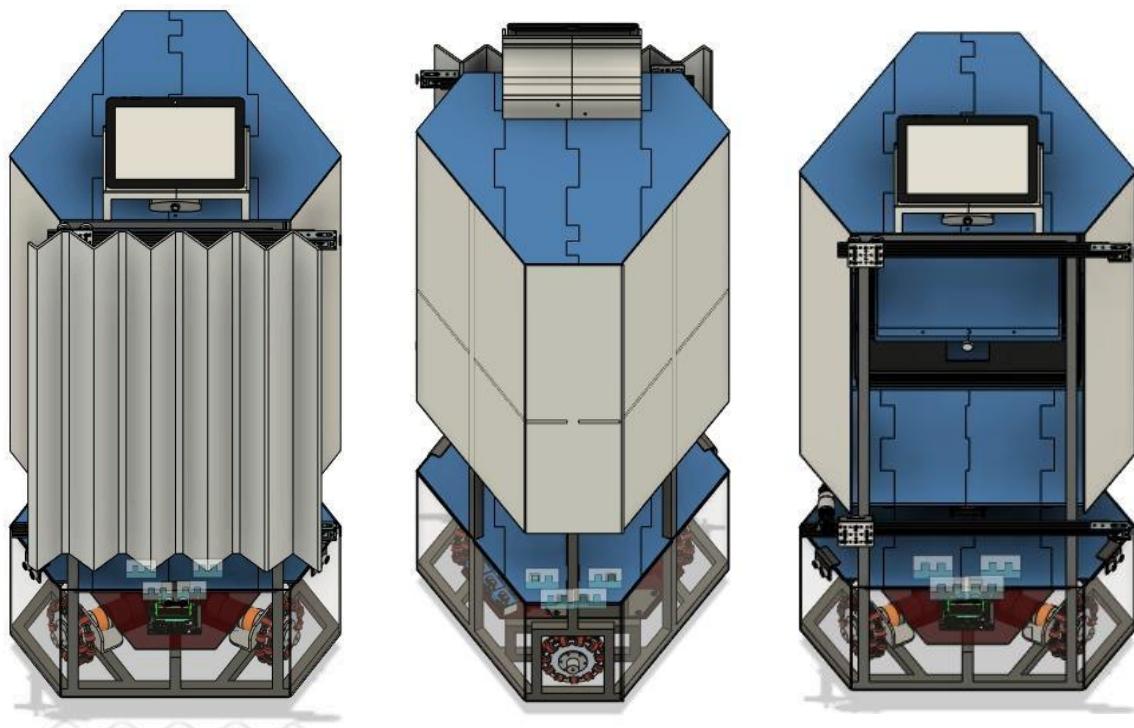
**Tabel 3.27 Rangkuman Hasil Pemilihan Komponen**

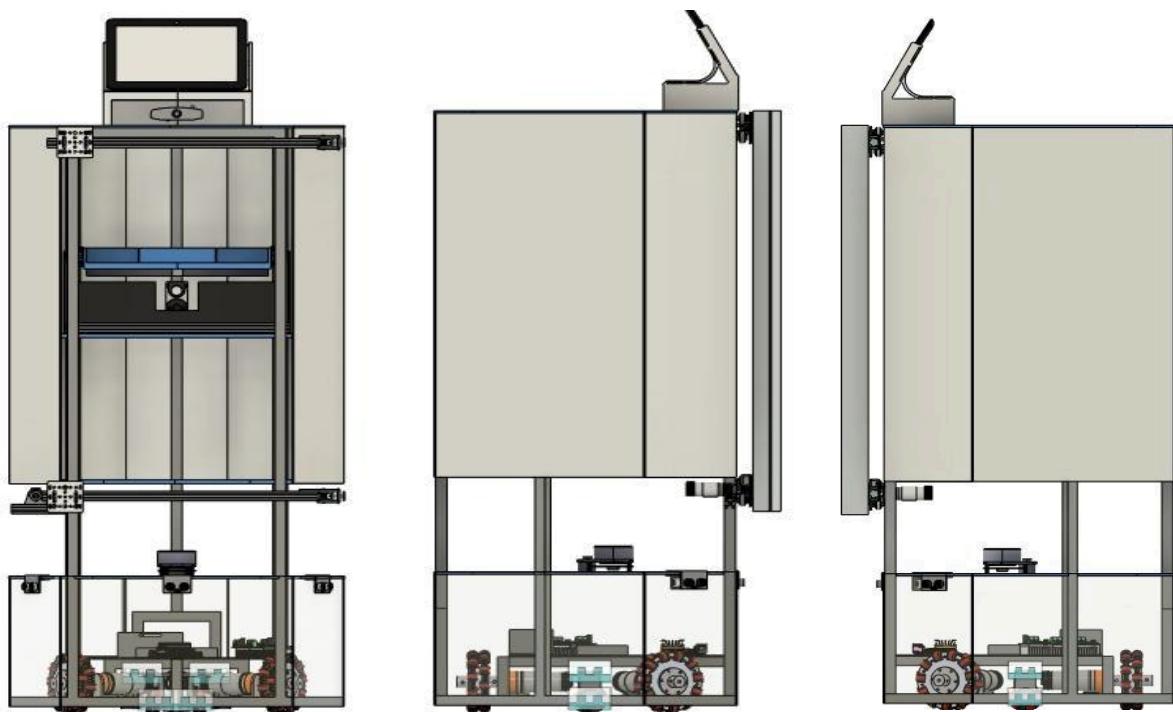
No	Unsur pembentuk sistem	Jenis komponen	Komponen terpilih
1	Unit Kontrol	Mikrokontroler Sensor dan Aplikasi	ESP32 Dev-Kit V4
		Mikrokontroler <i>Base</i>	STM32 F411CEU6 -WeAct Blackpill V2.0
		Mikrokontroler Mekanisme <i>Tray</i>	Arduino Uno R3
2	Sensor Odometri dan <i>Obstacles Avoidances</i>	Sensor <i>External Rotary Encoder</i>	External Rotary Encoder 600 ppr
		Sensor IMU	MPU6050

		Sensor Jarak	Sensor Ultrasonik HCSR-04
3	Pendeteksian QR	Mini PC	Jetson Nano Recomputer J1010 Seeed Studio
		Kamera	Logitech C310 Webcam
4	Aktuator Base Robot	Motor DC untuk <i>Base</i>	PG45 19.2K 24V 7 ppr encoder
		Motor Driver	BTS7960
		Baterai	SMT-Power Aki Elektronik 12V 7Ah
5	Aktuator Penutup dan <i>Tray</i>	Linear Aktuator	Electric Linear Actuator 300 mm Stroke
		Motor DC untuk Penggerak Penutup <i>Tray</i>	Motor DC PG28 7 ppr
		Roda <i>Omniwheel</i>	Roda <i>Omniwheel</i> 127 mm
		Kerangka Alumunium	Alumunium V-Slot
		Material Penutup	Infraboard
		Material <i>Tray</i>	Akrilik

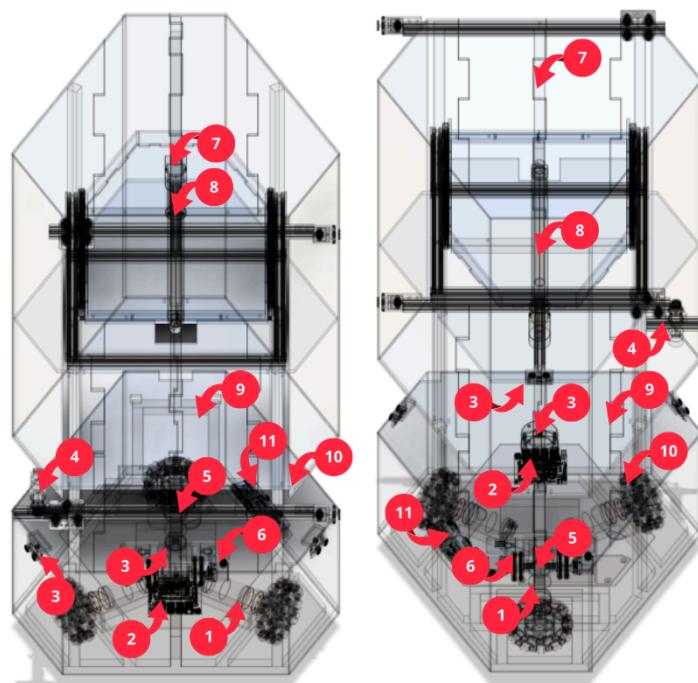
### 3.4 Desain Sistem Terpilih dan Cara Penggunaannya

#### 3.4.1 Desain Sistem





**Gambar 16 Desain Sistem Robot Pengantar Makanan Berbasis Odometri dan QR Detection**



## DESKRIPSI

1. PG-45 sebagai aktuator robot.
2. Jetson Nano & Arduino Uno sebagai MCU *QR computation*.
3. Ultrasonik HC SR-04 & Lidar sebagai fungsi *Obstacle Avoidance*.
4. PG-28 sebagai aktuator buka tutup pintu tray.
5. Baterai Aki 12 Volt & Ah sebagai daya robot.
6. Sensor *External Rotary* untuk mengukur perputaran roda.
7. Kamera dan tablet untuk mendeteksi kode QR di meja dan *user interface*.
8. Linear aktuator sebagai aktuator *tray slider*.
9. *Tray Static* untuk menyimpan makanan dan minuman.
10. Penutup Box Elektronika.
11. MPU6050 untuk mengukur percepatan pada robot.

**Gambar 17 Desain Sistem Robot Pengantar Makanan Berbasis *Sensor Fusion* dan *Object Detection* secara keseluruhan**

### **3.4.2 Cara Penggunaan Sistem**

Cara penggunaan sistem terpilih adalah sebagai berikut:

1. Pengguna meletakkan makanan yang hendak diantar ke dalam *tray slider* (8). Pengguna juga dapat menggunakan *tray* bawah (9).
2. Kemudian pengguna memilih lokasi meja pada aplikasi/*user interface* yang ditampilkan pada tablet (7) untuk memulai pergerakan robot.
3. Motor DC PG28 (4) bergerak menarik penutup ke kanan sehingga *tray* tertutup dan makanan tersimpan di dalamnya.
4. Ketiga motor DC PG45 pada *base* akan mulai bergerak.
5. Begitu robot melakukan pergerakan menuju meja pelanggan, sensor *external rotary encoder* (6), HC-SR04 (3), serta MPU6050 (11) mulai melakukan penginderaan sehingga robot mampu memperkirakan posisi dari waktu ke waktu relatif terhadap titik awal.
6. Apabila dalam pergerakannya terdapat objek yang menghalangi, robot akan menghindari dan mencari jalur terdekat untuk dilewati. Lalu kembali ke jalur semula.
7. Saat robot sampai di lokasi meja, kamera (7) akan aktif untuk mendeteksi QR yang berada di atas meja pelanggan.
8. Jetson Nano (2) mengolah hasil tangkapan QR untuk mendapatkan nomor meja pelanggan.
9. Apabila nomor meja hasil deteksi QR bersesuaian dengan lokasi meja yang dimasukkan pada aplikasi, motor DC PG28 (4) akan mendorong penutup ke kiri sehingga *tray* berada dalam posisi terbuka.
10. Linear aktuator (8) bergerak mendorong *tray slider* ke atas meja pelanggan.
11. Makanan pun tersaji ke pelanggan.

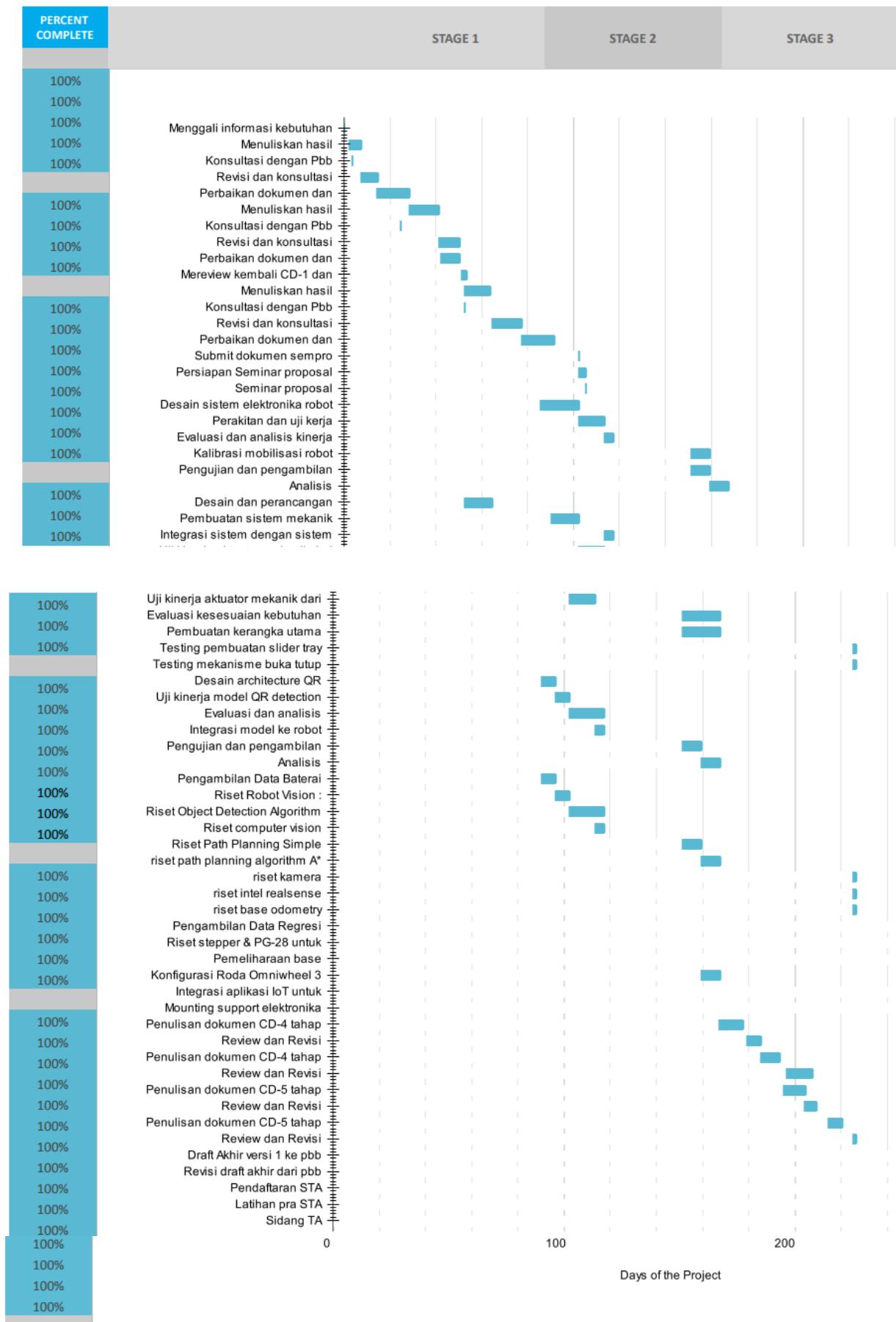
### 3.5 Jadwal Pengerjaan

Berikut ini adalah tabel jadwal pengerjaan dan pembagian tugas anggota tim serta gantt chart dalam pengerjaan robot pengantar makanan berbasis *sensor fusion* dan *object detection*.

Task Name	Start Date	End Date	Start On Day*	Duration* (Work Days)	Team Member
<b>Dokumen CD-1</b>					
Menggali informasi kebutuhan CD-1	19-Sep-2023	23-Sep-2023	0	1/4	ALL
Menuliskan hasil	21-Sep-2023	26-Sep-2023	2	6	gebby
Konsultasi dengan Pbb	22-Sep-2023	22-Sep-2023	3	1	ALL
Revisi dan konsultasi	26-Sep-2023	3-Oct-2023	7	8	ALL
Perbaikan dokumen dan pengesahan	3-Oct-2023	17-Oct-2023	14	15	ALL
<b>Dokumen CD-2</b>					
Menuliskan hasil	17-Oct-2023	30-Oct-2023	28	14	rafly
Konsultasi dengan Pbb	13-Oct-2023	13-Oct-2023	24	1	ALL
Revisi dan konsultasi	30-Oct-2023	8-Nov-2023	41	10	rafly,gebby
Perbaikan dokumen dan pengesahan	31-Oct-2023	8-Nov-2023	42	9	ALL
<b>Dokumen CD-3</b>					
Mereview kembali CD-1 dan CD2	9-Nov-2023	11-Nov-2023	51	3	ALL
Menuliskan hasil	10-Nov-2023	21-Nov-2023	52	12	kiki,rafly
Konsultasi dengan Pbb	10-Nov-2023	10-Nov-2023	52	1	ALL
Revisi dan konsultasi	22-Nov-2023	5-Dec-2023	64	14	kiki,gebby
Perbaikan dokumen dan pengesahan	5-Dec-2023	19-Dec-2023	77	15	ALL
Submit dokumen sempro	30-Dec-2023	30-Dec-2023	102	1	gebby
Persiapan Seminar proposal	30-Dec-2023	2-Jan-2024	102	4	ALL
Seminar proposal	2-Jan-2024	2-Jan-2024	105	1	ALL
<b>Sistem Mobilisasi Robot Delivery Food</b>					
Desain sistem elektronika robot	13-Dec-2023	30-Dec-2023	85	18	kiki
Perakitan dan uji kerja	30-Dec-2023	10-Jan-2024	102	12	kiki
Evaluasi dan analisis kinerja	10-Jan-2024	14-Jan-2024	113	5	kiki

Kalibrasi mobilisasi robot	17-Feb-2024	25-Feb-2024	151	9	kiki
Pengujian dan pengambilan data	17-Feb-2024	25-Feb-2024	151	9	kiki
Analisis	25-Feb-2024	4-Mar-2024	159	9	ALL
<b>Sistem Mekanik Robot Delivery Food</b>					
Desain dan perancangan sistem mekanik	10-Nov-2023	22-Nov-2023	52	13	rafly
Pembuatan sistem mekanik	18-Dec-2023	30-Dec-2023	90	13	rafly
Integrasi sistem dengan sistem mekanik	10-Jan-2024	14-Jan-2024	113	5	rafly
Uji kinerja aktuator mekanik dari case sensor	30-Dec-2023	10-Jan-2024	102	12	rafly
Evaluasi kesesuaian kebutuhan	17-Feb-2024	4-Mar-2024	151	17	ALL
Pembuatan kerangka utama	17-Mar-2024	4-Apr-2024	151	17	rafly
Testing pembuatan slider tray	1-May-2024	30-May-2024	225	2	ALL
Testing mekanisme buka tutup	1-May-2024	30-May-2024	225	2	ALL
<b>Sistem QR Detection Robot Delivery Food</b>					
Desain architecture QR detection	18-Dec-2023	24-Dec-2023	90	7	gebby, kiki
Uji kinerja model QR detection	24-Dec-2023	30-Dec-2023	96	7	gebby, kiki
Evaluasi dan analisis	30-Dec-2023	14-Jan-2024	102	16	gebby,rafly
Integrasi model ke robot	10-Jan-2024	14-Jan-2024	113	5	kiki
Pengujian dan pengambilan data	17-Feb-2024	25-Feb-2024	151	9	ALL
Analisis	25-Feb-2024	4-Mar-2024	159	9	ALL
<b>Riset Power Sistem Robot + Kendali Robot</b>					
Pengambilan Data Baterai tahap 1,2 dan 3	18-Dec-2023	24-Dec-2023	90	7	kiki, rafly
Riset Robot Vision :	24-Dec-2023	30-Dec-2023	96	7	kiki
Riset Object Detection Algorithm	30-Dec-2023	14-Jan-2024	102	16	kiki
Riset computer vision	10-Jan-2024	14-Jan-2024	113	5	kiki
Riset Path Planning Simple Algorithm	17-Feb-2024	25-Feb-2024	151	9	kiki
riset path planning algorithm A*	25-Feb-2024	4-Mar-2024	159	9	kiki
riset kamera	1-May-2024	30-May-2024	225	2	gebby
riset intel realsense	1-May-2024	30-May-2024	225	2	kiki
riset base odometry	1-May-2024	30-May-2024	225	2	kiki, rafly
Pengambilan Data Regresi Linear	10-Jun-2024	18-Jun-2024	265	6	kiki,rafly
Riset stepper & PG-28 untuk mekanisme buka tutup	3-Jun-2024	10-Jun-2024	258	10	rafly, gebby
Pemeliharaan base	3-Jun-2024	20-Jul-2024	268	20	kiki, rafly
Konfigurasi Roda Omniwheel 3 roda PG-45	25-Feb-2024	4-Mar-2024	159	9	rafly
Integrasi aplikasi IoT untuk tombol Interface Pelanggan	10-Jun-2024	18-Jun-2024	265	6	rafly
Mounting support elektronika dan mekanik	3-Jun-2024	20-Jul-2024	268	20	kiki
<b>Dokumen CD-4</b>					
Penulisan dokumen CD-4 tahap 1	4-Mar-2024	14-Mar-2024	167	11	kiki,gebby
Review dan Revisi	16-Mar-2024	22-Mar-2024	179	7	kiki,gebby
Penulisan dokumen CD-4 tahap 2	22-Mar-2024	30-Mar-2024	185	9	ALL
Review dan Revisi	2-Apr-2024	13-Apr-2024	196	12	ALL
<b>Dokumen CD-5</b>					
Penulisan dokumen CD-5 tahap 1	1-Apr-2024	10-Apr-2024	195	10	rafly,gebby
Review dan Revisi	10-Apr-2024	20-Apr-2024	204	6	rafly,gebby
Penulisan dokumen CD-5 tahap 2	20-Apr-2024	30-Apr-2024	214	7	kiki,gebby
Review dan Revisi	1-May-2024	30-May-2024	225	2	ALL
<b>Sidang Tugas Akhir</b>					
Draft Akhir versi 1 ke pbb	3-Jun-2024	10-Jun-2024	258	10	ALL
Revisi draft akhir dari pbb	10-Jun-2024	18-Jun-2024	265	6	ALL
Pendaftaran STA	18-Jun-2024	20-Jun-2024	273	7	gebby,rafly
Latihan pra STA	19-Jul-2024	28-Jun-2024	275	6	ALL
Sidang TA	29-Jul-2024	9-Aug-2024	285	6	ALL

**Gambar 18 Jadwal Pengerjaan dan Pembagian Tugas Anggota Tim**



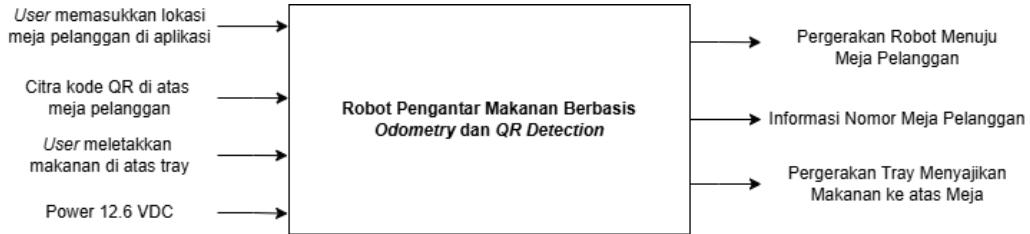


## BAB 4

# IMPLEMENTASI SOLUSI

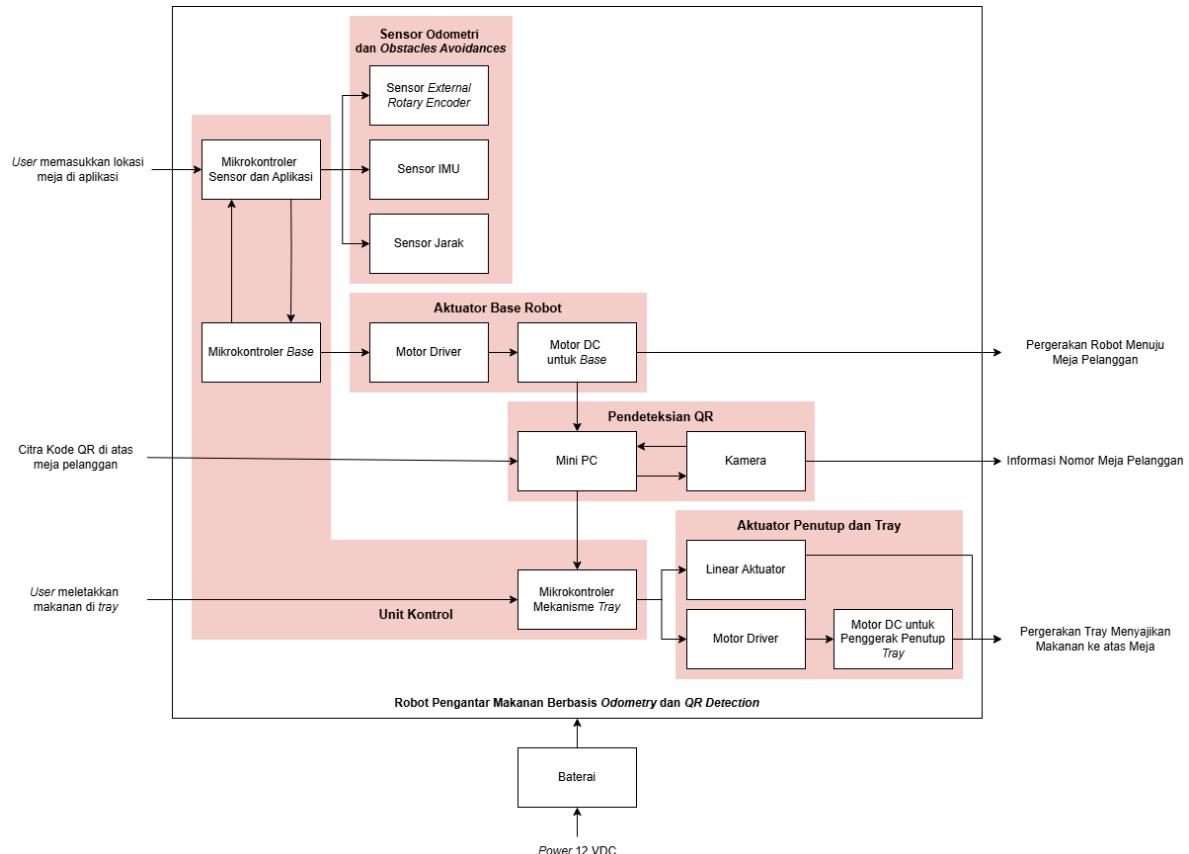
### 4.1 Implementasi Sistem

Bagian ini berisi penjabaran mengenai cara kerja, implementasi oleh pengguna, pengujian/kalibrasi dan hasilnya dari setiap sub-sistem yang dirancang.



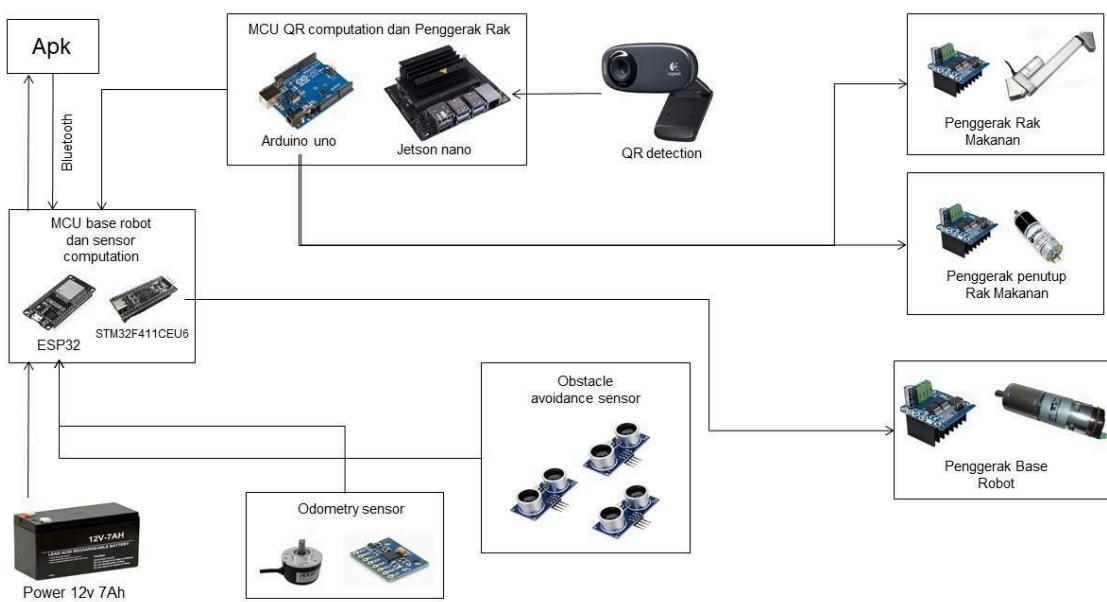
**Gambar 20 Diagram Blok Level 0**

Diagram blok level 0 pada gambar 4.1 menjelaskan masukan dan keluaran sistem. Sistem memiliki masukan yang berasal dari pengguna menggunakan tombol pada robot. Masukan ini akan diproses pada mikrokontroler utama untuk menyalakan sejumlah sub-sistem lain.



**Gambar 21 Diagram Blok Level 1 Beserta Sub-Sistemnya**

Berdasarkan diagram blok level 1 pada gambar 4.2, terdapat lima sub-sistem yang menyusun robot ini yaitu unit kontrol, sensor odometri dan *obstacles avoidances*, aktuator *base* robot, pendekstian QR, serta aktuator penutup dan *tray*. Selain diagram blok level 1, sistem ini juga dapat disajikan dalam bentuk diagram arsitektur. Diagram arsitektur menunjukkan hubungan antara komponen. Setiap komponen terhubung berdasarkan urutan pekerjaan yang dilakukan. Berikut ini diagram blok robot pengantar makanan berbasis odometri dan *QR Detection*.



**Gambar 22 Diagram Arsitektur Sistem**

Adapun dalam implementasi solusi, beberapa sub-sistem yang akan diuji sebelum diintegrasikan secara keseluruhan adalah sebagai berikut:

1. Unit Kontrol
2. Sensor Odometri dan *Obstacles Avoidances*
3. Aktuator *Base* Robot
4. Pendekstian QR
5. Aktuator Penutup dan *Tray*

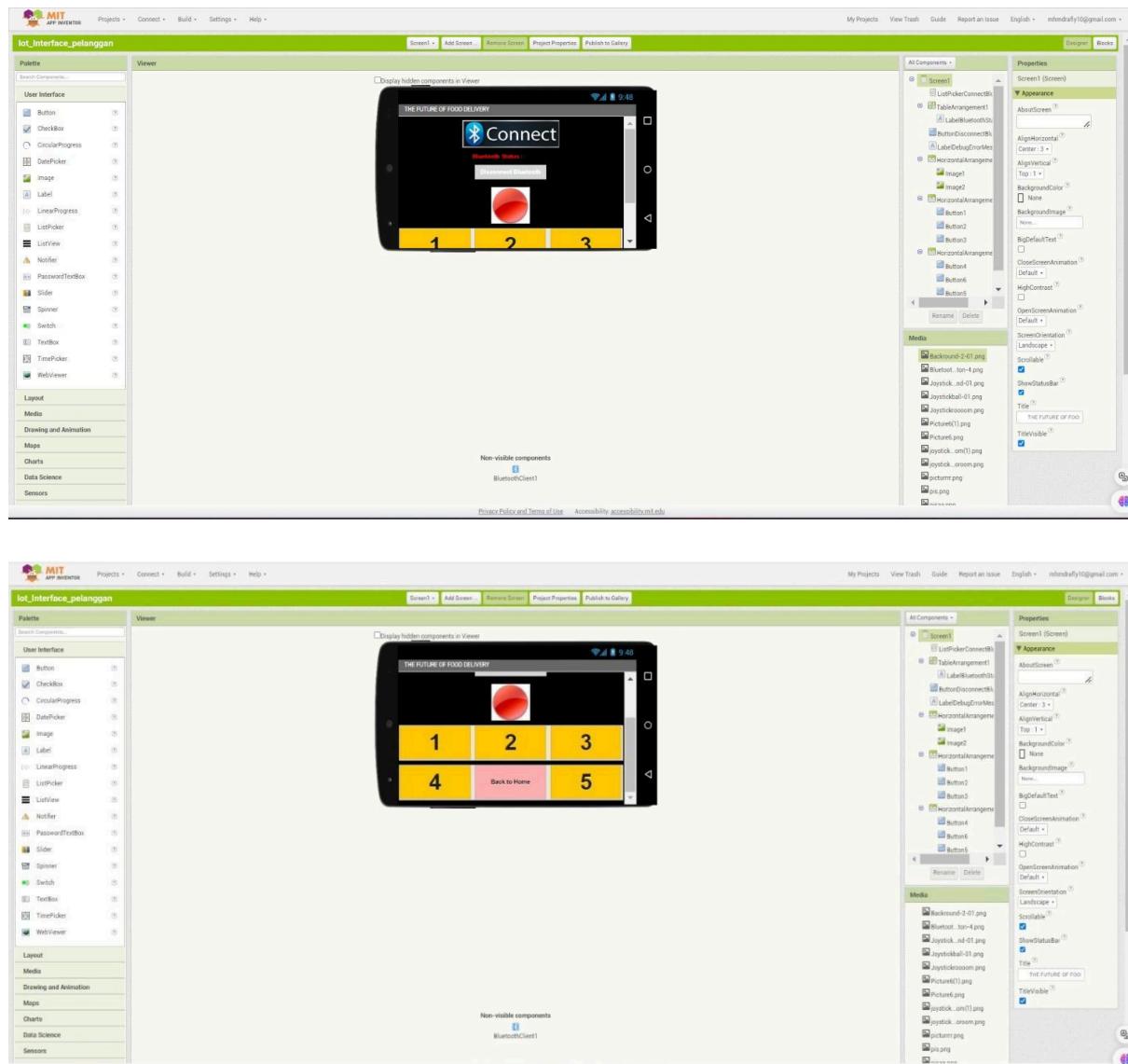
#### 4.1.1 Unit Kontrol

Dalam sistem ini, setidaknya terdapat 3 mikrokontroler yang digunakan yakni ESP32 Dev-Kit V4 untuk sensor dan aplikasi, STM32 F411CEU6-WeAct Blackpill V2.0 untuk *base* robot, serta Arduino Uno R3 untuk mekanisme *tray*. Untuk menguji fungsionalitas unit

kontrol ini, beberapa pengujian harus terhubung dengan sub-sistem lainnya seperti odometri dan *obstacles avoidances* serta PID posisi dengan sub-sistem sensor odometri dan *obstacles avoidances*, PID kecepatan dengan sub-sistem aktuator *base* robot. Oleh karena itu, pengujian yang dilakukan pada sub-sistem ini hanya komunikasi Bluetooth antara aplikasi dengan ESP32 Devkit v4 saja.

#### 4.1.1.1 Cara Kerja Sub Sistem

Berikut ini adalah tampilan aplikasi yang menjadi *user interface* robot pengantar makanan berbasis *odometry* dan *QR Detection*. Aplikasi dirancang dengan menggunakan MIT App Inventor.

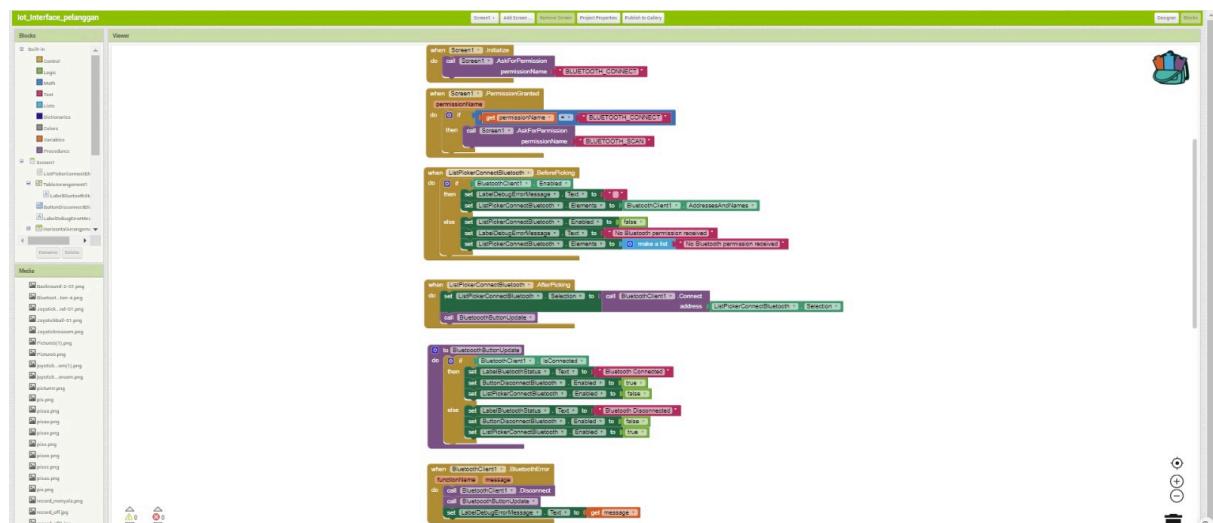


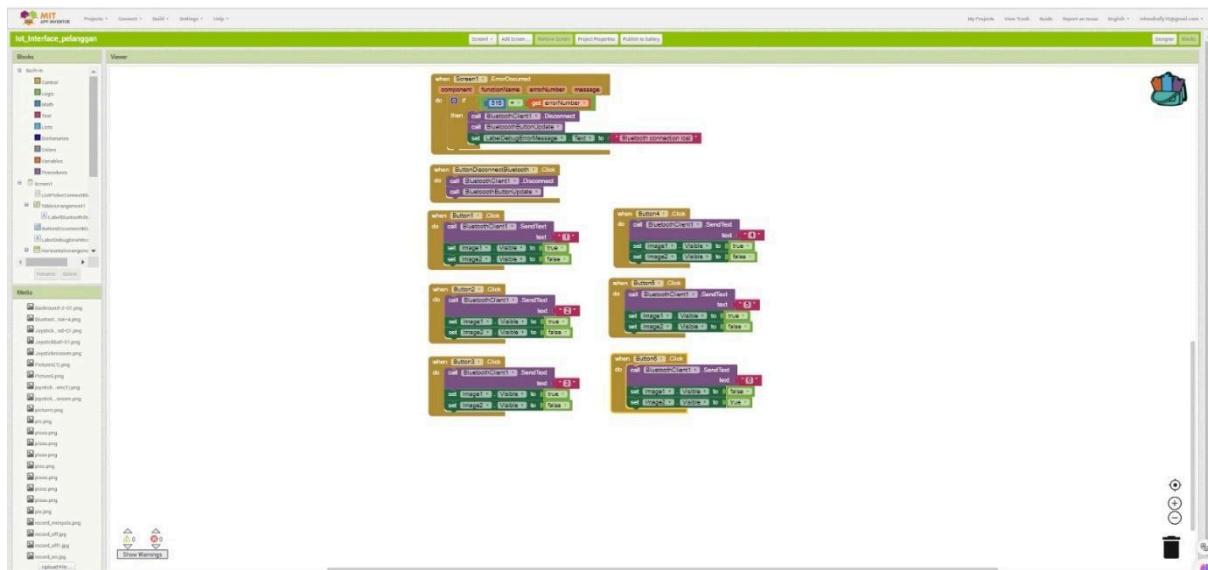
Gambar 23 Tampilan Aplikasi sebagai *User Interface*

Seperi yang dtunjukkan pada gambar 4.4, pada bagian teratas aplikasi terdapat tombol koneksi Bluetooth. Ketika tombol ini ditekan, aplikasi akan memindai Bluetooth yang tersedia kemudian pengguna dapat memilih jaringan Bluetooth dari mikrokontroler ESP32 DevKit V4. Di bawahnya, terdapat lampu indikator yang akan menyala merah apabila berhasil Bluetooth berhasil tersambung. Ketika tombol lokasi meja ditekan oleh pengguna, aplikasi akan mengirimkan variabel dengan tipe data *char* yang berbeda-beda untuk masing-masing tombol. Variabel ini mewakili titik koordinat meja pelanggan tertentu.

#### 4.1.1.2 Implementasi

Proses pembuatan aplikasi pada MIT App Inventor menggunakan bahasa pemrograman *block programming* yang apabila telah berhasil tersambung dengan koneksi Bluetooth ESP32, pengguna dapat menekan tombol angka pada aplikasi. Sebagaimana yang tertera gambar 4.5, tombol 1 akan mengirim char "1", tombol 2 akan mengirim char "2", dan seterusnya.





**Gambar 24 Block Programming pada Perancangan Aplikasi *User Interface***

Gambar 4.6 menunjukkan cuplikan *source code* yang berfungsi untuk menerima kiriman char dari aplikasi pada ESP32 dengan menggunakan Bluetooth.

```
#include <MPU6050_tockn.h>
#include <Wire.h>
#include "BluetoothSerial.h"

#if! defined(CONFIG_BT_ENABLED) || defined(CONFIG_BLUEDROID_ENABLED)
//#error Bluetooth is not enabled! Please run make menuconfig to and
enable it
#endif

BluetoothSerial SerialBT;

char data_;
int status_jalan;

void setup() {
    Serial.begin(115200);
    Serial2.begin(9600);
    SerialBT.begin("ESP32test");
}

void loop() {
    if(millis() - prev_t >= 10){
        prev_t = millis();
        controller();
    }
}

void controller() {
    if (SerialBT.available()) {
        data_ = SerialBT.read();
        Serial.println(data_);
        if(data_ == '1'){
            serial.println("data");movement = 1;status_jalan = 1;//path pola 1
go to table
        else if(data_ == '2'){movement = 2;status_jalan = 1;}//path pola 2
    }
}
```

Pada source code tersebut, ESP32 akan menerima *char* yang disimpan pada variabel *data\_*. Kemudian nilai *char* masuk ke dalam percabangan. Variabel "movement" mewakili nomor meja pelanggan. Sementara variabel "status jalan" merupakan inisiasi untuk tiap tahap yang akan dilakukan robot selama pergerakannya. Nilai dari variabel tersebut akan ditambah terus menerus hingga tahap akhir selesai (*increment*).

#### 4.1.1.3 Pengujian/Kalibrasi

Berikut ini merupakan langkah-langkah pengujian/kalibrasi kerja sub-sistem unit kontrol ESP32.

1. Siapkan aplikasi yang telah dirancang menggunakan MIT App Inventor. Buka Aplikasi hingga muncul laman pertama seperti yang ditunjukkan pada gambar 4.4.
  2. Colokkan kabel micro USB untuk meng-upload *program* pada ESP32. Lalu, upload program hingga status dinyatakan berhasil.
  3. Pada aplikasi, klik tombol *connect* dengan logo Bluetooth untuk memindai koneksi Bluetooth yang tersedia.
  4. Sambungkan aplikasi dengan koneksi Bluetooth dari ESP32. Apabila sudah terhubung, buka Serial Monitor pada Arduino IDE.
  5. Tekan satu persatu tombol pada aplikasi dan lihat data yang diterima oleh ESP32 melalui Serial Monitor.
- Hasil Pengujian

Tabel 4.1 menunjukkan hasil pengiriman dan penerimaan data *char* dari Aplikasi ke mikrokontroler ESP32 yang terbaca melalui Serial Monitor Arduino IDE.

**Tabel 4.1 Pengiriman dan Penerimaan Data *Char* dari Aplikasi ke ESP32**

No.	Data <i>char</i> yang dikirim Aplikasi	Data <i>char</i> yang diterima ESP32
1	1	1
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	1
9	1	1
10	1	1
11	2	2
12	2	2

13	2	2
14	2	2
15	2	2
16	2	2
17	2	2
18	2	2
19	2	2
20	2	2
21	3	3
22	3	3
23	3	3
24	3	3
25	3	3
26	3	3
27	3	3
28	3	3
29	3	3
30	3	3
31	4	4
32	4	4
33	4	4
34	4	4
35	4	4
36	4	4
37	4	4
38	4	4
39	4	4
40	4	4
41	5	5
42	5	5
43	5	5
44	5	5
45	5	5
46	5	5
47	5	5
48	5	5
49	5	5
50	5	5
51	6	6
52	6	6
53	6	6

54	6	6
55	6	6
56	6	6
57	6	6
58	6	6
59	6	6
60	6	6

- Analisis Pengujian

Berdasarkan hasil pengujian pada tabel 4.1, data *char* yang dikirim Aplikasi dan diterima oleh ESP32 bersesuaian. Hal ini menunjukkan bahwa sub-sistem unit kontrol bekerja dengan baik.

#### 4.1.2 Sensor Odometri dan *Obstacles Avoidances*

##### 4.1.2.1 Cara Kerja Sub Sistem

Dalam sub-sistem ini, ada 2 jenis sensor yang dipergunakan yakni *external rotary encoder* dan *Inertial Measurement Unit* (IMU). Keduanya digunakan untuk odometri. Odometri adalah penggunaan data yang diperoleh dari sensor pergerakan sehingga dapat memperkirakan perubahan posisi (perpindahan) robot dari waktu ke waktu. Dalam prosesnya, sistem odometri akan mengkalkulasi banyaknya *pulse* yang terbaca dari *external rotary encoder* sebelum dilakukan perhitungan secara matematis sehingga posisi robot dapat diperkirakan. Sensor IMU juga digunakan dalam odometri untuk mengetahui orientasi arah robot selama bergerak. Adapun data dari kedua sensor akan dikomputasi kembali di mikrokontroler ESP32.

Selain odometri, sub-sistem ini juga dirancang untuk algoritma *obstacles avoidances* agar robot mampu bergerak tanpa menabrak objek atau hambatan di sekitarnya. Dalam algoritma ini, sensor ultrasonik digunakan untuk mengukur jarak ke objek tersebut dengan memancarkan gelombang ultrasonik dan menerima kembali pantulannya. Sehingga dengan memanfaatkan sub-sistem ini, robot dapat bergerak menuju ke titik tujuan yang diinginkan pada bidang datar kartesian.

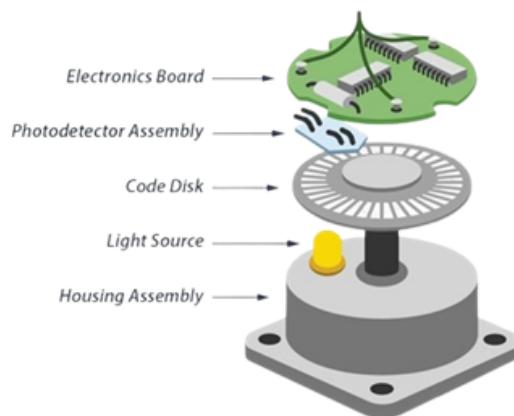
##### 4.1.2.2 Implementasi

Dalam sub-sistem ini, sensor rotary encoder dan MPU6050 digunakan untuk komputasi odometri. Rotary encoder adalah sensor yang akan mendeteksi banyaknya putaran roda dengan data yang diambil berupa *pulse*.



**Gambar 26 Sensor Rotary Encoder**

Adapun sensor rotary encoder yang digunakan adalah External Rotary Encoder 600 ppr. Sensor ini termasuk dalam jenis *optical encoder* yang memanfaatkan cahaya yang bersinar melalui disk berkode untuk melacak pergerakan poros. Sebagaimana ditunjukkan pada gambar 4.8, sinyal listrik akan dihasilkan setiap kali cahaya yang berasal dari *lightning source* sampai ke *photodetector assembly* yang terhubung dengan papan elektronika. Dengan begitu, *optical encoder* akan membaca 1 *pulse* pergerakan. Banyaknya *pulse* yang terdeteksi oleh *optical encoder* bergantung pada kecepatan putaran disk berkode yang terhubung dengan suatu poros. Pada poros ini akan disambungkan dengan roda. Sehingga jika roda tersebut berputar akibat pergerakan robot, maka *optical encoder* akan mulai menghasilkan pembacaan *pulse*.



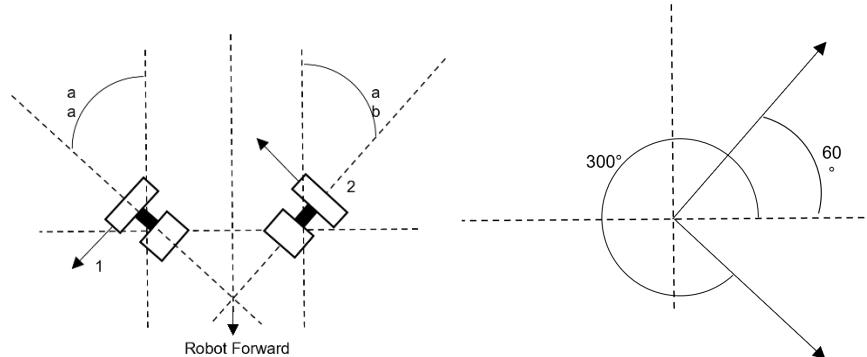
**Gambar 27 Bagian-bagian Optical Rotary Encoder**

Selain itu, External Rotary Encoder 600 ppr ini termasuk dalam *incremental encoder* di mana pulse yang dihasilkan mewakili perubahan posisi relatif dan tidak menyimpan informasi mengenai posisi absolut. Kemudian, sensor rotary encoder ini akan terhubung dengan roda *omniwheel* 60 mm.



**Gambar 28 Roda Omniwheel 60 mm**

Selama roda tersebut berputar ketika robot bergerak, sensor rotary encoder akan terus menerus menghasilkan pembacaan *pulse*. Selain itu, roda omniwheel 60 mm dipasang pada robot dengan konfigurasi khusus sehingga mampu membaca pergerakan pada sumbu X dan Y robot.

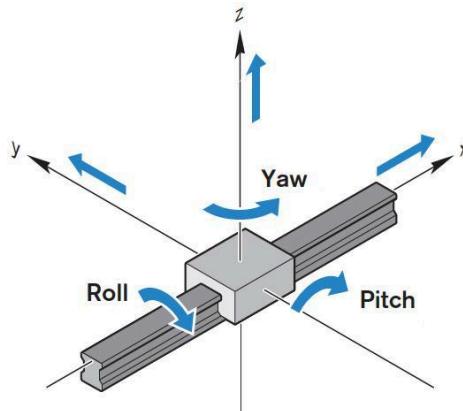


**Gambar SEQ Gambar \\* ARABIC 29 Konfigurasi Roda Odometri (Kiri) dan Sudut Vektor Putaran Roda (Kanan)**



**Gambar 30 Sensor IMU MPU6050**

Sementara itu, MPU6050 menggunakan *gyroscope* untuk mendeteksi perubahan sudut atau rotasi robot pada quaternion. Hal ini memungkinkan robot untuk membaca nilai perubahan sudut *pitch*, *roll*, dan *yaw*.



**Gambar 31 Pitch, Roll, dan Yaw dalam Ruang Gerak 3 Dimensi**

Dalam konteks pergerakan robot, *pitch*, *roll*, dan *yaw* digunakan untuk menggambarkan orientasi atau rotasi sebuah objek dalam ruang tiga dimensi sehingga dapat mendeskripsikan bagaimana suatu objek berputar di sekitar sumbu utamanya. Dalam sub-sistem ini, nilai perubahan sudut yang dibutuhkan hanya *yaw* saja. *Yaw* (belok) mengacu pada rotasi di sekitar sumbu vertikal (sumbu Z) yang membentang dari atas hingga bawah robot. Ketika melakukan pergerakan *yaw* (berbelok), objek atau robot tersebut mampu berputar ke kiri atau kanan, sama halnya seperti gerakan kemudi kapal atau berbelok pada kendaraan darat.

MPU6050 memiliki library tersendiri untuk mengkonversi nilai *yaw* menjadi *angle* dalam satuan derajat (*degree*). Data *angle* tersebut akan diubah ke radian dengan mengalikan nilai *angle* dan konstanta 0.01745329251. Lalu data pembacaan *pulse* dari sensor rotary encoder dan *angle* MPU6050 akan dimasukkan ke formula perhitungan odometri berdasarkan konfigurasi roda dan arah vector yang memengaruhi sudut trigonometri sebagaimana ditunjukkan pada persamaan 4.1 dan 4.2.

$$Dx = P2 \cos(ab + (\text{YawAngleMPU6050} \times 0.01745329251)) + P1 \cos(aa + (\text{YawAngleMPU6050} \times 0.01745329251)) \quad (4.1)$$

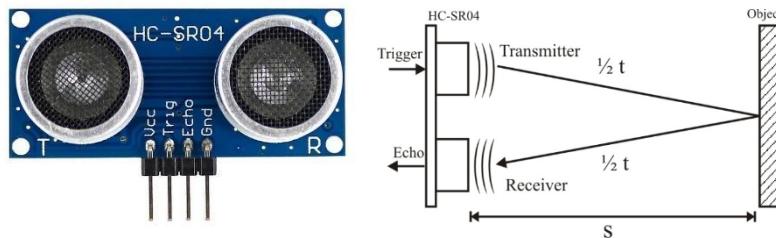
$$Dy = P2 \sin(ab + (\text{YawAngleMPU6050} \times 0.01745329251)) + P1 \cos(aa + (\text{YawAngleMPU6050} \times 0.01745329251)) \quad (4.2)$$

Di mana *Dx* dan *Dy* adalah hasil pembacaan posisi sumbu X dan Y. Sementara itu, *P1* dan *P2* adalah pembacaan *pulse* sensor rotary encoder 1 dan 2 serta *aa* dan *ab* adalah konversi sudut sensor *rotary encoder* 1 dan 2 ke sumbu Y. Nilai 0.01745329251 adalah konstanta yang menyatakan perubahan sudut dari satuan derajat ke radian.

Gambar 4.13 menunjukkan cuplikan *source code* fungsi odometri yang dikomputasi pada mikrokontroler ESP32.

```
void odometry() {
    thetta = mpu6050.getAngleZ();
    dX = (encoder_value_2 * cos(ab + (thetta*0.01745329251)) +
encoder_value_1 * cos(aa + (thetta*0.01745329251)));
    dY = (encoder_value_2 * sin(ab + (thetta*0.01745329251)) +
encoder_value_1 * sin(aa + (thetta*0.01745329251)));
```

Selain metode odometri, robot juga menerapkan algoritma *obstacles avoidances* menggunakan sensor ultrasonik HCSR-04. Pada sensor ini, bagian *trigger* akan memancarkan gelombang ultrasonik. Gelombang tersebut akan memantul apabila terkena suatu objek. Pantulan gelombang akan diterima kembali oleh bagian *Echo*. Jarak antara sensor ultrasonik dan objek tersebut dapat diketahui dengan mengkalkulasinya waktu tempuh dengan kecepatan rambat gelombang ultrasonik di udara.



**Gambar 33 Sensor Ultrasonik HCSR-04 (Kiri) dan Prinsip Kerjanya (Kanan)**

Berikut ini adalah cuplikan *source code obstacles avoidances* dengan menggunakan sensor ultrasonik HCSR-04.

```
float read_ultra(int pin_trig, int pin_echo){
    digitalWrite(pin_trig, LOW);
    delayMicroseconds(2);
    digitalWrite(pin_trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(pin_trig, LOW);

    duration = pulseIn(pin_echo, HIGH);
    distance = (duration*.0343)/2;
    return distance;
}
```

#### 4.1.2.3 Pengujian/Kalibrasi

Berikut ini merupakan langkah-langkah pengujian/kalibrasi kinerja sub-sistem sensor odometri dan *obstacles avoidances*.

1. Dalam sub-sistem ini digunakan 3 sensor ultrasonik HCSR-04 sehingga hubungkan pin *trigger* dan *echo* sensor 1 ke pin 13 dan 12, pin *trigger* dan *echo* sensor 2 ke pin 18 dan 5 serta pin *trigger* dan *echo* sensor 3 dengan pin 27 dan 14 mikrokontroler ESP32.
2. Untuk sensor *rotary encoder* 1, hubungkan *channel A* dan *B* masing-masing dengan pin 32 dan 33. Sementara itu *channel A* dan *B* sensor *rotary encoder* 2 dihubungkan dengan pin 26 dan 25 mikrokontroler ESP32.

3. Sementara itu, MPU6050 dihubungkan dengan pin SDA dan SCL pada mikrokontroler ESP32.
4. Apabila ketiga sensor sudah terhubung dengan mikrokontroler, *upload* program untuk menjalankan robot ke titik tertentu dan buka *serial monitor*.
5. Sambungkan baterai aki pada robot dan nyalakan switch pada robot.
6. Pada serial monitor, masukkan nilai tertentu yang mewakili titik yang hendak dituju robot dalam satuan cm hingga robot bergerak. Dalam pengujian ini, titik yang dituju sejauh 100 cm ke arah depan, belakang, kiri, serta kanan.
7. Amati data yang ditampilkan serial monitor.

- Hasil Pengujian

Tabel 4.2 hingga 4.5 menunjukkan nilai target dan perpindahan aktual robot hasil kalkulasi metode odometri ke arah depan, belakang, kiri, dan kanan.

**Tabel 4.2 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Depan**

No	Home - Depan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	98	97,02	0,98
2	97	97,7	0,7
3	98	98,24	0,24
4	104	98,78	5,22
5	105	98,46	6,54

No	Home - Depan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	97	98,03	1,03
7	97	98,54	1,54
8	97	98,66	1,66
9	102	97,72	4,28
10	101	98,25	2,75

**Tabel 4.3 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Belakang**

No	<i>Home - Belakang</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	-94	-92,33	1,67
2	-89	-91,29	2,29
3	-85	-90,39	5,39
4	-86	-91,38	5,38
5	-84	-91,97	7,97

No	<i>Home - Belakang</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	-81	-91,58	10,58
7	-91	-91,56	0,56
8	-88	-92,04	4,04
9	-86	-91,01	5,01
10	-85	-91,56	6,56

**Tabel 4.4 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kanan**

No	<i>Home - Kanan</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	102	91,62	10,38
2	100	90,2	9,8
3	102	90,48	11,52
4	102	91,76	10,24
5	97	90,85	6,15

No	<i>Home - Kanan</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	98	91,36	6,64
7	97	91,37	5,63
8	97	91,4	5,6
9	95	90,39	4,61
10	95	90,71	4,29

**Tabel 4.5 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kiri**

No	<i>Home - Kiri</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	-100	-90,11	9,89
2	-103	-90,64	12,36
3	-102	-91,64	10,36
4	-104	-90,82	13,18

5	-103	-91,45	11,55
No	<i>Home - Kiri</i>		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	-97	-90,15	6,85
7	-95	-90,04	4,96
8	-97	-90,47	6,53
9	-95	-90,24	4,76

10	-96	-91,17	4,83
----	-----	--------	------

- Analisis Pengujian

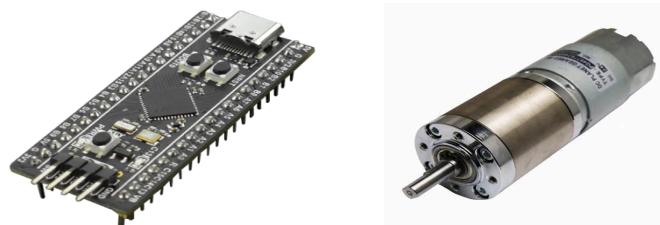
Berdasarkan hasil pengujian tersebut dapat dilihat bahwa metode odometri dan *obstacles avoidance* yang diterapkan pada robot berhasil membawa robot ke titik tujuan yang dikehendaki namun tetap terjadi galat yang kecil.

#### 4.1.3 Aktuator *Base* Robot

##### 4.1.3.1 Cara Kerja Sub Sistem

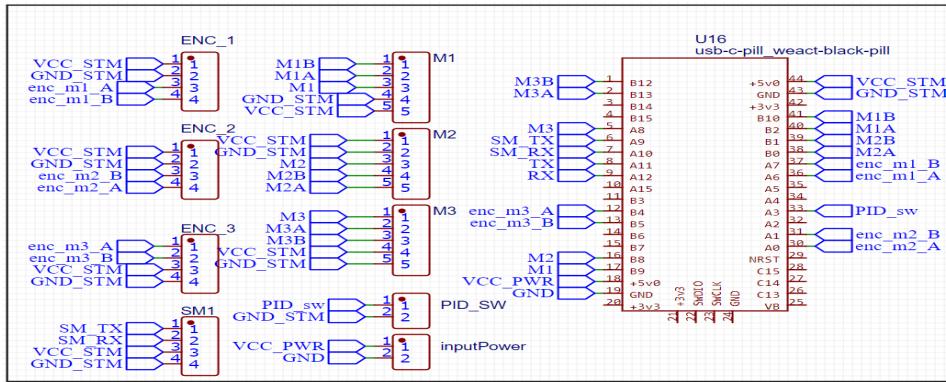
Dalam sub sistem ini, sinyal PWM akan menjadi masukan bagi motor driver sehingga motor driver dapat mengeluarkan tegangan. Kemudian motor DC akan bergerak. Perputaran motor DC juga ikut memutar encoder internal yang terhubung dengan *shaft* bagian belakang motor DC. Encoder internal akan menghasilkan pembacaan *pulse* yang menjadi umpan balik untuk kendali motor DC.

##### 4.1.3.2 Implementasi



**Gambar 34 STM32F411CEU6-WeAct Blackpill V2.0 (Kiri) dan Motor DC PG45 (Kanan)**

Dalam implementasinya, mikrokontroler STM32F411CEU6-WeAct Blackpill V2.0 yang menjadi mikrokontroler untuk sub-sistem aktuator *base* robot terhubung dengan *input channel* motor driver BTS7960. Lalu keluaran dari motor driver akan terhubung dengan motor DC PG45. Encoder internal yang terpasang pada motor DC PG45 juga terhubung ke mikrokontroler. Adapun gambar 4.16 menunjukkan desain skematik untuk sub-sistem aktuator *base* robot.



**Gambar 35 Diagram Skematik Sub-Sistem Aktuator Base Robot**

Hasil pembacaan *pulse* oleh encoder internal dapat merepresentasikan besarnya kecepatan putaran motor DC. Akan tetapi dalam sistem aktual, hasil pembacaan ini tidak selalu bernilai stabil sehingga diperlukan suatu metode kendali. Metode kendali bertujuan untuk mengendalikan suatu parameter agar nilainya stabil pada titik tertentu yang diinginkan. Salah satu metode kendali yang paling umum digunakan adalah metode kendali PID (Proportional Integral Derivative). Adapun metode kendali PID dapat dinyatakan dalam bentuk persamaan matematis berikut.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de}{dt} \quad (4.3)$$

Dengan  $K_p$ ,  $K_i$ , dan  $K_d$  berturut-turut parameter proporsional, integral dan derivatif. Tiap-tiap parameter PID memiliki karakteristik dan pengaruh yang berbeda terhadap respon sistem.

- Parameter Proporsional (P): Memperbaiki respon transien khususnya *rise time* dan *settling time* menjadi lebih cepat, menambah atau mengurangi kestabilan, dan mengurangi *error steady-state*.
- Parameter Integral (I): Mengurangi *error steady-state*, respon lebih lambat dibandingkan parameter proporsional. Parameter integral juga dapat menimbulkan ketidakstabilan karena menambah orda sistem.
- Parameter Derivative (D): Memberikan efek redaman pada osilasi sehingga bisa memperbesar pemberian nilai  $K_p$ . Parameter ini hanya berubah saat perubahan *error*. Selain itu parameter ini tidak dapat digunakan tanpa adanya parameter lainnya karena dengan adanya *error* statis, parameter ini tidak akan dapat bereaksi. Hal pertama yang perlu dilakukan menentukan setpoint terlebih dahulu. Dalam pengujian ini besarnya

setpoint (*pulse*) adalah 0. Hal ini berarti metode kendali PID akan berusaha untuk menahan robot untuk tetap diam.

Adapun komputasi PID akan menghasilkan nilai PWM yang dikirimkan ke motor driver BTS7960 yang mengatur kecepatan dan arah perputaran motor.



**Gambar 36 Motor Driver BTS7960**

Adapun gambar 4.16 menunjukkan cuplikan *source code* kalkulasi PID kecepatan pada sub-sistem aktuator *base* robot.

```
#include "mbed.h"
#include "RotaryEncoder.h"

DigitalIn pid_sw(PA_3);

DigitalOut m2A(PB_0);
DigitalOut m2B(PB_1);
PwmOut m2(PB_8);

DigitalOut m1A(PB_2);
DigitalOut m1B(PB_10);
PwmOut m1(PB_9);

DigitalOut m3A(PB_13);
DigitalOut m3B(PB_12);
PwmOut m3(PA_8);

RotaryEncoder enc_m1(PA_6,PA_7,2,134);
RotaryEncoder enc_m2(PA_0,PA_1,2,134);
RotaryEncoder enc_m3(PB_4,PB_5,2,134);

Serial esp(PA_11, PA_12);

const int numChars = 100;
char receivedChars[numChars]; // an array to store the received data
char *vx;
char *dx;
float dX = 0;
float dY = 0;
float Exr = 0;
float Eyr = 0;
float Eth = 0;
float Vy = 0;
float Vx = 0;
float tetta = 0;
float kar=0;

bool newData = false;
```

```
while (esp.readable() > 0 && newData == false){
    rc = esp.getc();
    // printf("%c\n",rc);

    if (rc != endMarker){
        receivedChars[ndx] = rc;
        ndx++;
        if (ndx >= numChars) {
            ndx = numChars;
        }
    }
    else {
        receivedChars[ndx] = '\0'; // terminate the string
        ndx = 0;
        newData = true;
    }
}

void olahDataOdom() {
    dx = strtok(receivedChars, ";");
    Exr = atof(dx);

    dx = strtok(NULL, ";");
    Eyr = atof(dx);

    dx = strtok(NULL, ";");
    Eth = atof(dx);

    // dx = strtok(NULL, ";");
    // kar = atof(dx);

}

extern serial_t      stdio_uart;
extern int           stdio_uart_initiated;

Ticker pid;
float dt = 0.01;
float e[3];

float eI[3];
float eD[3];
float laste[3];
```

```

float lastPid[3];
float pid_var[3];

// ideal
// float kp = 0.235;
// float ki = 0.010;
// float kd = 0.0005;

// tambah beban & gesekan
float kp = 0.338;
float ki = 0.0161;
float kd = 0.0005;

void pid_start(){
    pid_var[0] = enc_m1.getPulse();
    pid_var[1] = enc_m2.getPulse();
    pid_var[2] = enc_m3.getPulse();

    if(pid_sw.read() == 0){

        w[2] = Exr*0.5 + Eyr*-0.86602540378 + Eth;
        w[1] = Exr*0.5 + Eyr*0.86602540378 + Eth;
        w[0] = Exr*-1 + Eyr*0 + Eth;

        for(int i = 0;i < 3;i++){
            e[i] = w[i] - pid_var[i];
            eI[i] += e[i];
            eD[i] = e[i] - laste[i];

            hP[i] = e[i]*kp;
            hI[i] = eI[i]*ki*dt;
            hD[i] = eD[i]*kd/dt;

            if(hI[i] >=5){setI[i] = 1;}
            else if(hI[i] < -5){setI[i] = 1;}
            else{
                setI[i] = hI[i];
            }

            pidPwm[i] = hP[i]+setI[i]+hD[i];
            laste[i] = e[i];

            if(pidPwm[i] >= 1){pwmLebih[i] = 1;}
            else if(pidPwm[i] < -1){pwmLebih[i] = -1;}
        }
    }
}

```

```

//motor direksi start
if(pidPwm[0] < 0){m1A = 1; m1B = 0;}
else{m1A = 0; m1B = 1;}
if(pidPwm[1] < 0){m2A = 1; m2B = 0;}
else{m2A = 0; m2B = 1;}
if(pidPwm[2] < 0){m3A = 1; m3B = 0;}
else{m3A = 0; m3B = 1;}
//motor direksi end

m1 = fabs(pwmLebih[0]);
m1.period(0.01);
m2 = fabs(pwmLebih[1]);
m2.period(0.01);
m3 = fabs(pwmLebih[2]);
m3.period(0.01);

}else{
    for(int i = 0;i<3;i++){
        hP[i] = 0;
        hI[i] = 0;
        setI[i] = 0;
        hD[i] = 0;
        pidPwm[i] = 0;
        pwmLebih[i] = 0;
        laste[i] = 0;
        e[i] = 0;
        eI[i] = 0;
        eD[i] = 0;
    }
    m1 = 0;m1A =0;m1B =0;
    m2 = 0;m2A =0;m2B =0;
    m3 = 0;m3A =0;m3B =0;
}
}

float targetOdom = 16000;
float err_odom;
Timer rec;
char *data_;
float data__ = 0;

int main()
{
    esp.baud(9600);
    serial init(&stdio uart, PA 9, PA 10); // other than Serial2
}

```

```

while (true) {
    recvWithEndMarker();
    if(newData == true){
        olahDataOdom();
    }
    newData = false;
}

```

Adapun alur kerja dari *source code* tersebut adalah STM32F411CEU6-WeAct Blackpill V2.0 terhubung dengan ESP32 secara serial. Data hasil odometri dan obstacles avoidances dari ESP32 akan dikirim ke STM32F411CEU6-WeAct Blackpill V2.0 untuk menggerakkan motor PG36 pada base robot dalam bentuk string yang dibatasi dengan delimiter *semicolon* (;). Oleh karena itulah, pada fungsi OlahDataOdom() dilakukan proses *parsing* atau pencacahan datas string berdasarkan delimiternya menjadi 4 data berbeda yang akan masuk dalam fungsi pid\_start().

#### 4.1.3.3 Pengujian/Kalibrasi

Berikut ini merupakan langkah-langkah pengujian/kalibrasi kinerja PID kecepatan pada sub-sistem aktuator *base* robot.

1. Pada sub-sistem ini digunakan 3 motor DC PG45 sehingga masing-masing pin M+ dan Mi pada motor DC harus terhubung dengan pin M+ dan M- pada tiap motor driver BTS7960.
2. Selanjutnya pin R\_EN dan L\_EN motor driver BTS7960 diseri dan dihubungkan dengan pin PB9 untuk motor 1, PB8 untuk motor 2, dan PA8 untuk motor 3.
3. Pada motor motor driver BTS7960, pin VCC disambungkan ke sumber tegangan 12V dari baterai aki dan pin GND ke GND baterai aki. Sementara itu, pin R\_PWM dan L\_PWM juga terhubung pin digital pada STM32F411CEU6-WeAct Blackpill V2.0. Dalam hal ini, hubungkan pin R\_PWM dan L\_PWM motor 1 dengan pin PB2 dan PB10, pin R\_PWM dan L\_PWM motor 2 dengan pin PB0 dan PB1, dan pin R\_PWM dan L\_PWM motor 3 dengan pin PB13 dan PB12 sesuai *source code* yang telah dirancang.
4. Untuk internal *rotary encoder* bawaan motor DC PG45 memiliki 2 pin yakni channel A dan channel B. Oleh karena itu, hubungkan *channel A* dan *B* motor 1 pada pin PA6 dan PA7, *channel A* dan *B* motor 2 pada pin PA0 dan PA1, serta *channel A* dan *B* motor 3 pada pin PB4 dan PB5. Hubungkan juga pin 3.3V ke pin 3.3V pada

STM32F411CEU6-WeAct Blackpill V2.0, begitulah dengan pin GND ke GND mikrokontroler tersebut.

5. Hubungkan *switch* untuk menjalankan base pada pin PA3.
6. Apabila semua sistem elektronika base sudah terhubung, *upload* program ke STM32F411CEU6-WeAct Blackpill V2.0. Jika telah berhasil, nyalakan serial monitor untuk melihat *feedback* kontroler PID kecepatan.
7. Nyalakan robot dalam kondisi *floating* dan berikan nilai setpoint tertentu pada robot.
8. Amati nilai *feedback* kontroler PID kecepatan pada serial monitor.

- Hasil Pengujian

Dalam proses pengujian, ada 3 setpoint yang digunakan yakni 0,5, 1, dan 1,5 ppr (*pulse per rotation*). Tabel 4.6 sampai 4.8 menunjukkan nilai PWM keluaran kontroler PID dan pembacaan feedbacknya dalam ppr.

**Tabel 4.6 Nilai PWM dan *Feedback* Kontroler PID dengan Setpoint 0,5 PPR**

No.	PWM			<i>Feedback</i>		
	M1	M2	M3	M1	M2	M3
1	0,21	-0,21	-0,21	0	1	1
2	0,22	0,22	0,22	0	0	0
3	0,17	0,17	0,22	0	0	0
4	0,65	-0,16	-0,21	-1	1	1
5	0,17	0,22	0,17	0	0	0
6	0,22	-0,21	-0,21	0	1	1
7	0,22	0,17	0,17	0	0	0
8	-0,21	0,22	-0,16	1	0	1
9	-0,21	0,22	-0,21	1	0	1
10	0,22	0,17	-0,21	0	0	1

**Tabel 4.7 Nilai PWM dan *Feedback* Kontroler PID dengan Setpoint 1 PPR**

No.	PWM			<i>Feedback</i>		
	M1	M2	M3	M1	M2	M3
1	-0,049	-0,049	-0,049	1	1	1
2	-0,387	-0,049	0,043	2	1	0
3	0,001	0,043	-0,048	1	0	1

4	0,440	-0,04	0,002	0	1	1
5	0,828	0,002	0,003	-1	1	1
6	0,003	0,441	-0,04	1	0	1
7	-0,433	0,004	-0,044	2	1	1
8	0,004	-0,433	-0,431	1	2	2
9	0,005	0,005	-0,429	1	1	2
10	0,832	0,394	0,397	-1	0	0

**Tabel 4.8 Nilai PWM dan *Feedback* Kontroler PID dengan Setpoint 1,5 PPR**

No.	PWM			<i>Feedback</i>		
	M1	M2	M3	M1	M2	M3
1	0,617	-0,211	-0,206	0	2	2
2	0,618	0,177	0,182	0	1	1
3	-0,206	0,616	0,234	2	0	1
4	0,182	-0,208	0,185	1	2	1
5	0,183	0,230	0,187	1	1	1
6	0,233	0,180	0,237	1	1	1
7	-0,253	0,231	0,188	2	1	1
8	-0,202	0,182	0,627	2	1	0
9	-0,151	0232	0,190	2	1	1
10	-0,250	0,233	0,191	2	1	1

- Analisis Pengujian

Berdasarkan hasil pengujian tersebut, dapat terlihat bahwa *feedback* PID kecepatan bernilai cukup stabil terhadap *setpoint* yang telah ditentukan yakni 0,5, 1, dan 1,5.

#### 4.1.4 Pendekstrian QR

##### 4.1.4.1 Cara Kerja Sub Sistem

Ketika robot telah sampai ke titik tujuan, kamera akan menangkap citra dari kode QR yang ditampilkan oleh pengguna. Proses komputasi pendekstrian QR terjadi pada mikrokomputer Jetson Nano Recomputer J1010 Seeed Studio.

#### 4.1.4.2 Implementasi

Pada sub-sistem ini, kamera yang digunakan adalah Logitech Webcam C310 karena memiliki resolusi yang mumpuni untuk pendekripsi kode QR.



**Gambar 39 Logitech Webcam C310**

Kamera terhubung dengan mikrokomputer Jetson Nano Recomputer J1010 Seeed Studio yang akan memproses komputasi pendekripsi objek. QR dan pendekripsi objek menggunakan bahasa pemrograman Python dan library OpenCV yang bersifat *open-source* serta dirancang untuk pengolahan citra dan *computer vision*. Gambar 4.21 menunjukkan *source code* sub-sistem pendekripsi QR.

```
import cv2
import time
import serial

arduino = serial.Serial(port='/dev/ttyACM0', baudrate=115200, timeout=.1)

def write_read(x):
    arduino.write(bytes(x, "utf-8"))
    time.sleep(0.05)
    data = arduino.readline()
    return data

camera_id = 0
delay = 1
window_name = 'OpenCV QR Code'
data_sent_0 = False
data_sent_1 = False
data_sent_2 = False

qcd = cv2.QRCodeDetector()
cap = cv2.VideoCapture(camera_id)

while True:
    ret, frame = cap.read()

    if ret:
        ret_qr, decoded_info, points, _ = qcd.detectAndDecodeMulti(frame)
        if ret_qr:
            print("QR Code Detected") # Debug output
            for s, p in zip(decoded_info, points):
                print(f"Decoded info: {s}") # Debug output
                if s == '0' and not data_sent_0:
                    print("Sending 9") # Debug output
                    write_read('9') # Send '9' to Arduino
```

```

if ret:
    ret_qr, decoded_info, points, _ = qcd.detectAndDecodeMulti(frame)
    if ret_qr:
        print("QR Code Detected") # Debug output
        for s, p in zip(decoded_info, points):
            print(f"Decoded info: {s}") # Debug output
            if s == '0' and not data_sent_0:
                print("Sending 9") # Debug output
                write_read('9') # Send '9' to Arduino
                data_sent_0 = True
                color = (255, 255, 0)
            elif s == '1' and not data_sent_1:
                print("Sending 0") # Debug output
                write_read('0') # Send '0' to Arduino
                data_sent_1 = True
                color = (0, 255, 0)
            elif s == '2' and not data_sent_2:
                print("Sending 1") # Debug output
                write_read('1') # Send '1' to Arduino
                data_sent_2 = True
                color = (0, 255, 0)
            else:
                color = (0, 0, 255)
        frame = cv2.polylines(frame, [p.astype(int)], True,
color, 8)
    else:
        print("No QR Code detected or QR Code unreadable") # Debug
output
    cv2.imshow(window_name, frame)

    if cv2.waitKey(delay) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

#### 4.1.4.3 Pengujian/Kalibrasi

Dalam sub-sistem ini, pengujian dilakukan terhadap jarak terjauh kamera dari kode QR. Ini adalah jarak maksimal dimana kode QR masih dapat terdeteksi oleh sub-sistem yang ditandai dengan adanya *bounding box* yang melingkupi kode QR tersebut. Adapun berikut ini langkah-langkah pengujian kinerja sub-sistem pendekripsi kode QR.

1. Siapkan kode QR yang akan digunakan.
2. Tempatkan posisi awal kode QR sejauh 1 meter dari kamera.

3. Dekatkan kode QR hingga kode QR terdeteksi pada program. Hal ini ditandai dengan munculnya bounding box yang melingkupi kode QR.
  4. Ukur jarak terjauh di mana kamera mampu mendeteksi kode QR menggunakan pita ukur.
- Hasil Pengujian

Tabel 5.6 menunjukkan seberapa jauh jarak maksimal pendekripsi agar kode QR mampu tetap terdeteksi. Hal ini ditandai dengan munculnya *bounding box* yang melingkupi kode QR pada *frame* kamera yang terbuka di layar PC.

**Tabel 4.9 Jarak Terjauh Pendekripsi QR**

No.	Percobaan Ke-	Jarak Terjauh (cm)	No.	Percobaan Ke-	Jarak Terjauh (cm)
1	1	26,0	6	6	26,0
2	2	25,8	7	7	27,0
3	3	25,7	8	8	27,5
4	4	26,1	9	9	27,4
5	5	25,9	10	10	27,7

- Analisis Pengujian

Berdasarkan data tabel 4.9 dapat terlihat bahwa pendekripsi kode QR mampu terdeteksi hingga jarak terjauh 27,7 cm selama 10 kali percobaan.

#### 4.1.5 Aktuator Penutup dan *Tray*

##### 4.1.5.1 Cara Kerja Sub Sistem

Apabila kode QR sudah berhasil terdeteksi, sub-sistem penutup aktuator penutup dan tray akan bekerja. Mikrokontroler Arduino Uno akan mengirimkan perintah berupa sinyal PWM kepada motor driver yang terhubung linear aktuator dan motor DC sehingga penutup dapat terbuka dan tray dapat bergerak maju.

##### 4.1.5.2 Implementasi

Dalam sub-sistem ini, linear aktuator yang digunakan adalah linear aktuator 300 mm stroke dan motor DC yang digunakan adalah motor DC PG28.

##### 4.1.5.3 Pengujian/Kalibrasi

Dalam sub-sistem ini, dilakukan 2 pengujian yaitu perpindahan penutup dan juga kecepatan linear *tray slider*. Berikut ini langkah pengujian untuk penutup *tray*.

1. Upload program perpindahan *penutup* dan *tray slider*. Kemudian penutup akan mulai bergerak.
2. Ukur perpindahan penutup.

3. Selanjutnya pada program yang sama, atur perpindahan *tray* hingga ke titik akhir lalu *upload* kembali.
  4. Hitung waktu yang dibutuhkan *tray* untuk berpindah dari titik awal ke titik akhir dengan menggunakan stopwatch.
  5. Besar kecepatan linear diperoleh dengan membagi perpindahan *tray* per waktu tempuh.
- Hasil Pengujian

Tabel 4.10 menunjukkan data perpindahan penutup sementara tabel 4.11 menunjukkan data kecepatan linear *tray slider* selama 10 kali percobaan.

**Tabel 4.10 Perpindahan Penutup yang Digerakkan oleh Motor DC PG28**

No.	Percobaan Ke-	Perpindahan Penutup (cm)
1	1	40,0
2	2	39,6
3	3	40,0
4	4	41,1
5	5	41,1

No.	Percobaan Ke-	Perpindahan Penutup (cm)
6	6	41,0
7	7	40,0
8	8	41,0
9	9	39,8
10	10	40,2

**Tabel 4.11 Perpindahan dan Kecepatan Linear *Tray Slider***

No.	Percobaan Ke-	Perpindahan <i>Tray</i> (cm)	Kecepatan Linear <i>Tray</i> ( $\frac{cm}{s}$ )
1	1	23,0	0,958333
2	2	22,8	0,95
3	3	22,8	0,95
4	4	22,9	0,954167
5	5	23,0	0,958333

No.	Percobaan Ke-	Perpindahan <i>Tray</i> (cm)	Kecepatan Linear <i>Tray</i> ( $\frac{cm}{s}$ )
6	6	22,9	0,954167
7	7	23,0	0,958333
8	8	22,8	0,95
9	9	22,9	0,954167
10	10	23,0	0,958333

- Analisis Pengujian

Berdasarkan pengujian tersebut didapatkan bahwa nilai rata-rata perpindahan penutup adalah 40,38 serta kecepatan linear tray berada di bawah 1 cm/s.

## 4.2 Hasil Akhir Integrasi Sistem

Gambar 4.22 menunjukkan hasil akhir integrasi dari kelima sub-sistem yang telah dirancang.



**Gambar 41 Hasil Integrasi Sistem**

## BAB 5

# PENGUJIAN SISTEM

### 5.1 Pengujian Sistem (secara keseluruhan sesuai spesifikasi CD2)

#### 5.1.1 Pengujian Spesifikasi 1

##### 5.1.1.1 Langkah Pengujian

Spesifikasi pertama yang harus terpenuhi sesuai analisis kebutuhan adalah robot bergerak menuju ke titik koordinat tertentu dengan toleransi ketepatan 10 cm dan kecepatan linear maksimal sebesar  $0,3 \frac{m}{s}$ . Oleh karena itu, berikut ini merupakan langkah-langkah pengujian yang dilakukan untuk memverifikasi spesifikasi 1.

1. Tentukan titik koordinat yang diinginkan dan tandai titik tersebut.
2. Rancang algoritma yang memungkinkan robot bergerak otomatis dari titik awal ke titik koordinat tersebut.
3. Hitung waktu pergerakan robot menggunakan stopwatch.
4. Begitu robot selesai bergerak, tandai titik akhir.
5. Kemudian, ukur titik akhir robot terhadap titik koordinat menggunakan pita ukur. Kecepatan linear diperoleh dengan mengkalkulasikan perubahan posisi terhadap waktu tempuh sampai ke titik akhir.

##### 5.1.1.2 Hasil Pengujian

Pengujian berlokasi di Telyu Caffee yang merupakan salah satu *coffee* di kawasan kampus Telkom University. Gambar 5.1 menunjukkan *trajectory* pergerakan robot selama pengujian (meja barista hingga ke meja pelanggan).



**Gambar 42 Trajectory Pergerakan Robot Pengantar Makanan Berbasis *Odometry* dan *QR Detection* di Lokasi Pengujian Telyu Caffee**

Berdasarkan hasil pengukuran yang telah dilakukan, perpindahan robot selama bergerak dalam *trajectory* tersebut sejauh 9,51. Tabel 5.1 menunjukkan waktu tempuh robot menuju meja pelanggan serta kecepatan linear robot yang dihitung berdasarkan persamaan 5.1.

$$\vec{v} = \frac{\vec{\Delta x}}{\Delta t} \quad (5.1)$$

Dengan:

$\vec{v}$  = Kecepatan linear ( $\frac{m}{s}$ )

$\vec{\Delta x}$  = Perubahan posisi (m)

$\Delta t$  = Perubahan waktu (s)

**Tabel 5.1 Waktu Tempuh dan Kecepatan Linear Robot**

No.	Percobaan Ke-	Waktu Tempuh (s)	Kecepatan Linear ( $\frac{m}{s}$ )
1	1	33,60	0,283036
2	2	34,64	0,274538
3	3	40,73	0,233489
4	4	50,92	0,186764
5	5	38,40	0,247656

No.	Percobaan Ke-	Waktu Tempuh (s)	Kecepatan Linear ( $\frac{m}{s}$ )
6	6	35,45	0,268265
7	7	37,47	0,253803
8	8	36,20	0,262707
9	9	35,45	0,268265
10	10	41,80	0,227512

### 3. Analisis Pengujian

Berdasarkan tabel 5.1 didapatkan bahwa nilai kecepatan linear ketika robot bergerak menuju ke titik koordinat tertentu tidak melebihi kecepatan linear maksimal sebesar  $0,3 \frac{m}{s}$ .

#### 5.1.2 Pengujian Spesifikasi 2

##### 5.1.2.1 Langkah Pengujian

Pengujian spesifikasi 2 dilakukan dengan memasukkan sejumlah beberapa titik posisi yang harus dituju oleh robot. Dalam pengujian ini, ada 4 titik yang digunakan yakni 100 cm ke depan, 100 cm ke belakang, 100 cm ke kanan, serta 100 cm ke kiri. Apabila berhasil, hal ini menngindikasikan bahwa robot mampu bergerak ke segala arah (maju, mundur, kiri, dan kanan) tanpa mengubah orientasi robot sama sekali.

##### 5.1.2.2 Hasil Pengujian

Tabel 5.2 sampai tabel 5.5 berturut-turut menunjukkan data target perpindahan dan perpindahan aktual robot beserta errornya ketika bergerak ke arah depan, belakang, kiri, dan kanan.

**Tabel 5.2 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Depan**

No	Home - Depan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	98	97,02	0,98
2	97	97,7	0,7
3	98	98,24	0,24
4	104	98,78	5,22
5	105	98,46	6,54

No	Home - Depan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	97	98,03	1,03
7	97	98,54	1,54
8	97	98,66	1,66
9	102	97,72	4,28
10	101	98,25	2,75

**Tabel 5.3 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Belakang**

No	Home - Belakang		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	-94	-92,33	1,67
2	-89	-91,29	2,29
3	-85	-90,39	5,39
4	-86	-91,38	5,38
5	-84	-91,97	7,97

No	Home - Belakang		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	-81	-91,58	10,58
7	-91	-91,56	0,56
8	-88	-92,04	4,04
9	-86	-91,01	5,01
10	-85	-91,56	6,56

**Tabel 5.4 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kanan**

No	Home - Kanan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	102	91,62	10,38
2	100	90,2	9,8
3	102	90,48	11,52
4	102	91,76	10,24
5	97	90,85	6,15

No	Home - Kanan		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	98	91,36	6,64
7	97	91,37	5,63
8	97	91,4	5,6
9	95	90,39	4,61
10	95	90,71	4,29



**Tabel 5.5 Target dan Perpindahan Aktual Robot Ketika Bergerak ke Kiri**

No	Home - Kiri		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
1	-100	-90,11	9,89
2	-103	-90,64	12,36
3	-102	-91,64	10,36
4	-104	-90,82	13,18
5	-103	-91,45	11,55

No	Home - Kiri		
	Target Perpindahan (cm)	Perpindahan Aktual (cm)	Error Perpindahan (cm)
6	-97	-90,15	6,85
7	-95	-90,04	4,96
8	-97	-90,47	6,53
9	-95	-90,24	4,76
10	-96	-91,17	4,83

#### 5.1.2.3 Analisis Pengujian

Data pergerakan robot pada tabel 5.2 hingga 5.5 menunjukkan bahwa spesifikasi 2 yakni robot mampu bergerak ke segala arah dalam bidang 2 dimensi tanpa mengubah orientasi robot telah terpenuhi.

#### 5.1.3 Pengujian Spesifikasi 3

##### 5.1.3.1 Langkah Pengujian

Spesifikasi ketiga yang harus terpenuhi sesuai analisis kebutuhan adalah jarak terjauh pendeksiyan kode QR sebesar 30 cm. Oleh karena itu, berikut ini merupakan langkah-langkah pengujian yang dilakukan untuk memverifikasi spesifikasi 3.

5. Siapkan kode QR yang akan digunakan.
6. Tempatkan posisi awal kode QR sejauh 1 meter dari kamera.
7. Dekatkan kode QR hingga kode QR terdeteksi pada program. Hal ini ditandai dengan munculnya bounding box yang melingkupi kode QR.
8. Ukur jarak terjauh di mana kamera mampu mendeksiyan kode QR menggunakan pita ukur.
- Hasil Pengujian



**Gambar 43 Posisi Awal Pendeksiyan Kode QR dan Kode QR Belum Terdeteksi**



**Gambar 44 Posisi Akhir Pendektsian Kode QR dan Kode QR Berhasil Terdeteksi**

Tabel 5.6 menunjukkan seberapa jauh jarak maksimal pendektsian agar kode QR mampu tetap terdeteksi. Hal ini ditandai dengan munculnya *bounding box* yang melingkupi kode QR pada *frame* kamera yang terbuka di layar PC.

**Tabel 5.6 Jarak Terjauh Pendektsian QR**

No.	Percobaan Ke-	Jarak Terjauh (cm)
1	1	26,0
2	2	25,8
3	3	25,7
4	4	26,1
5	5	25,9

No.	Percobaan Ke-	Jarak Terjauh (cm)
6	6	26,0
7	7	27,0
8	8	27,5
9	9	27,4
10	10	27,7

- Analisis Pengujian

Berdasarkan data tabel 5.6 dapat terlihat bahwa pendektsian kode QR mampu terdeteksi hingga jarak terjauh 27,7 cm selama 10 kali percobaan. Hal ini membuktikan bahwa spesifikasi 3 yakni adalah jarak minimal pendektsian kode QR sebesar 25 cm terpenuhi.

#### 5.1.4 Pengujian Spesifikasi 4

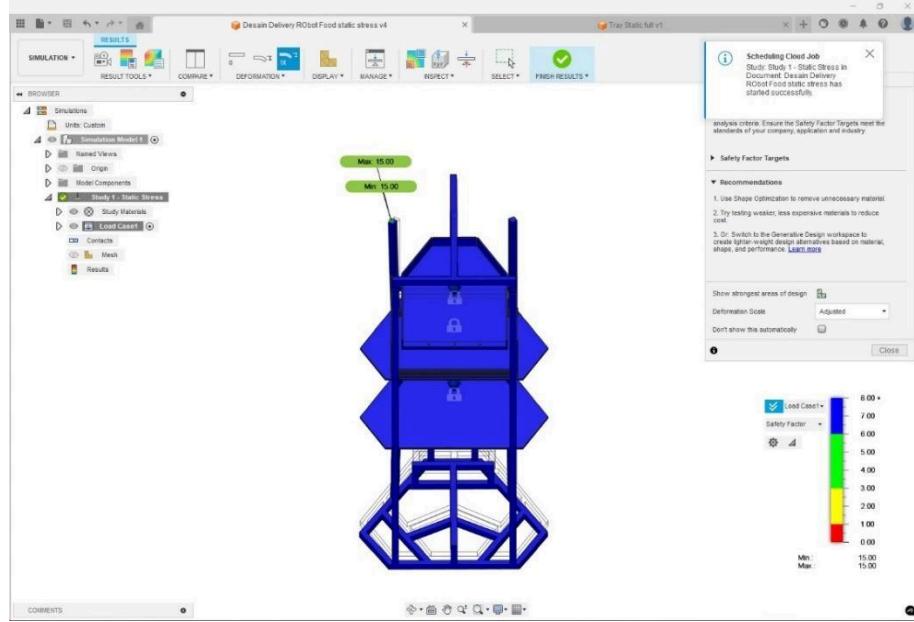
##### 5.1.4.1 Langkah Pengujian

Spesifikasi keempat yang harus terpenuhi sesuai analisis kebutuhan adalah tiap *tray* mampu menahan beban dengan berat maksimal sebesar 29.43 N (massa setara 3 kg). Pengujian ketahanan kerangka robot terhadap beban pada *tray* menggunakan simulasi FEA (*Finite Element Analysis*) software Autodesk Fusion 360. Oleh karena itu, berikut ini merupakan langkah-langkah pengujian yang dilakukan untuk memverifikasi spesifikasi 4.

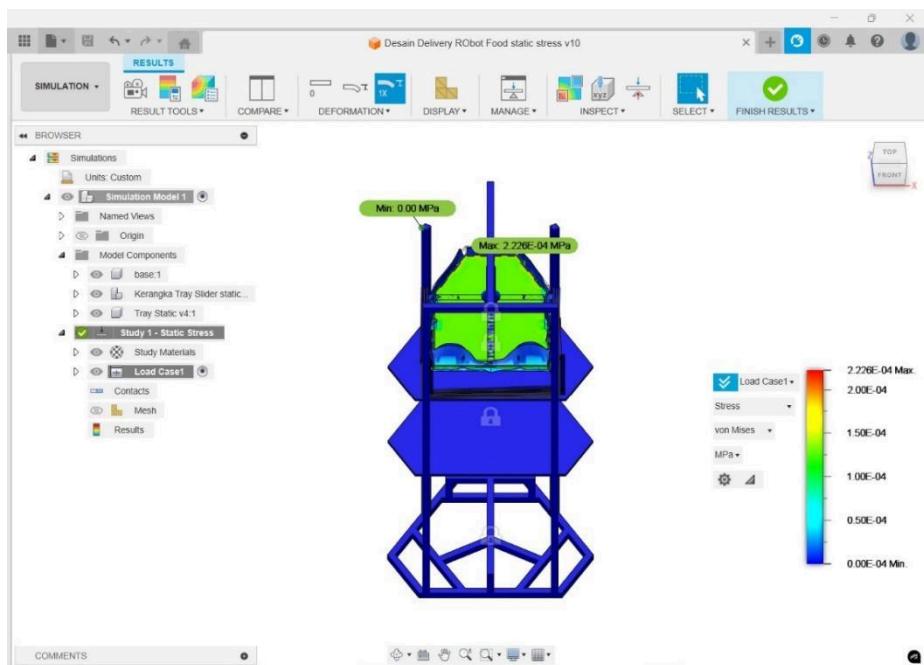
1. Siapkan desain 3D robot pada software Autodesk Fusion 360.
2. Pilih simulasi dengan metode *static stress*.
3. Tentukan material penyusun robot pada bagian *materials*.

4. Berikan *loads* dalam bentuk *force* sebesar 29,43 N (setara massa 3 kg).
  5. Tampilkan hasil simulasi berupa *safety factor* beserta diagram warnanya.
  6. Sistem dikatakan memenuhi syarat apabila *safety factor* berada di atas angka 1.
- Hasil Pengujian

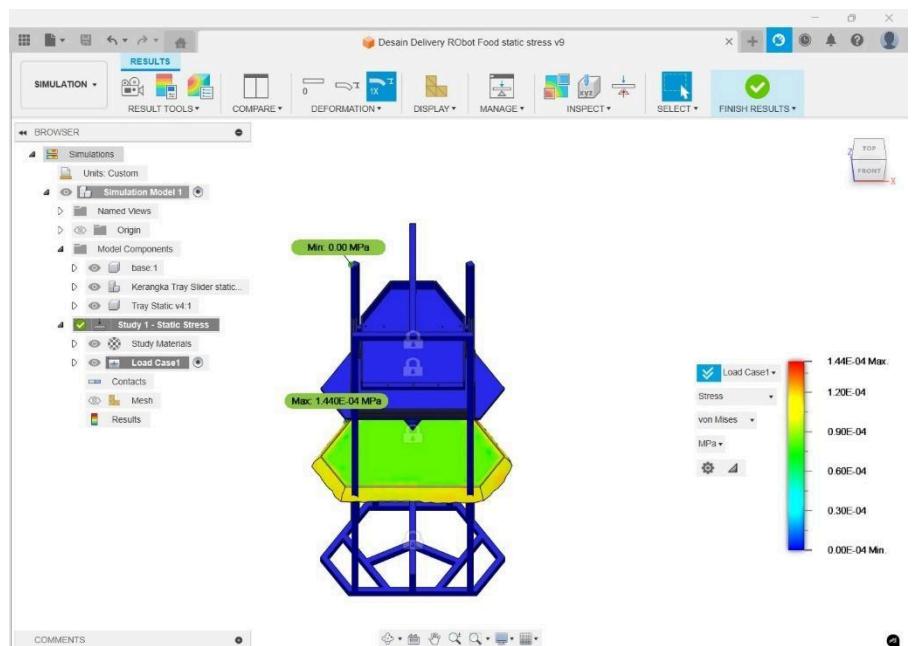
Setidaknya ada 3 kondisi pengujian ketahanan kerangka robot terhadap beban pada tray ini yaitu ketika diberi beban pada *tray slider*, diberi beban pada *tray statik*, serta diberi beban pada keduanya (*tray slider* maupun statik).



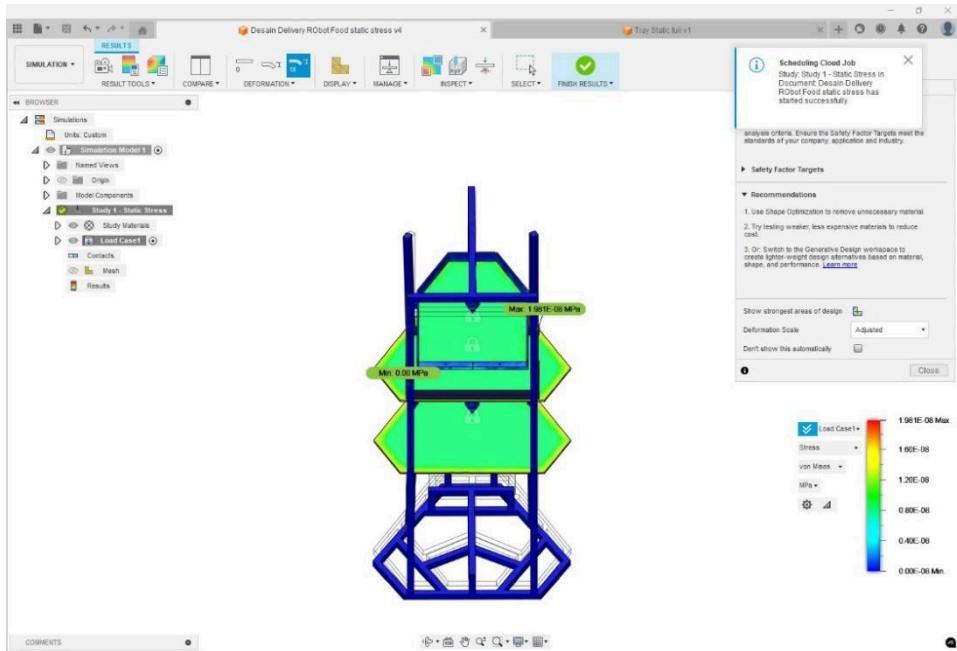
**Gambar 45 Simulasi FEA Sebelum Robot Diberi Beban**



**Gambar 46 Simulasi FEA Setelah Robot Diberi Beban Pada *Tray Slider***



**Gambar 47 Simulasi FEA Setelah Robot Diberi Beban Pada *Tray Statik***



**Gambar 48 Simulasi FEA Ketika Robot Diberi Beban Pada Kedua Tray**

- **Analisis Pengujian**

Berdasarkan hasil pengujian tersebut dalam dilihat bahwa nilai *safety factor* robot masih berada di ambang batas yang baik. Hal ini ditandai dengan indikator warna biru dan hijau pada robot ketika simulasi FEA berlangsung. Hal ini membuktikan bahwa spesifikasi 3 terpenuhi.

### 5.1.5 Pengujian Spesifikasi 5

#### 5.1.5.1 Langkah Pengujian

Spesifikasi terakhir yang harus terpenuhi adalah salah satu *tray* (dalam hal ini *tray slider*) mampu melakukan pergerakan slide ke luar dengan kecepatan linear maksimal  $0,1 \frac{cm}{s}$ . Berikut ini merupakan langkah-langkah pengujian yang dilakukan untuk memverifikasi spesifikasi 5.

6. Atur perpindahan *tray* hingga ke titik akhir.
7. Hitung waktu yang dibutuhkan *tray* untuk berpindah dari titik awal ke titik akhir dengan menggunakan stopwatch.
8. Besar kecepatan linear diperoleh dengan membagi perpindahan *tray* per waktu tempuh.

#### 5.1.5.2 Hasil Pengujian



**Gambar 49 Kondisi *Tray Slider* Sebelum (Kiri) dan Setelah (Kanan) Bergerak Menyajikan Makanan**



**Gambar 50 Pengukuran Perpindahan *Tray Slider* Setelah Bergerak Selama 24 Sekon**

Tabel 5.7 menunjukkan nilai perpindahan *tray slider* ketika bergerak selama 24 sekon serta besar kecepatannya. Nilai perpindahan *tray* juga dihitung dengan persamaan 5.1.

**Tabel 5.7 Perpindahan dan Kecepatan Linear *Tray Slider***

No.	Percobaan Ke-	Perpindahan <i>Tray</i> (cm)	Kecepatan Linear <i>Tray</i> ( $\frac{cm}{s}$ )
1	1	23,0	0,958333
2	2	22,8	0,95
3	3	22,8	0,95
4	4	22,9	0,954167
5	5	23,0	0,958333

No.	Percobaan Ke-	Perpindahan <i>Tray</i> (cm)	Kecepatan Linear <i>Tray</i> ( $\frac{cm}{s}$ )
6	6	22,9	0,954167
7	7	23,0	0,958333
8	8	22,8	0,95
9	9	22,9	0,954167
10	10	23,0	0,958333

#### 5.1.5.3 Analisis Pengujian

Berdasarkan data tabel 5.7 dapat terlihat bahwa kecepatan linear *tray slider* setelah bergerak selama 24 sekon tidak melebihi  $1 \frac{cm}{s}$  dalam 10 kali percobaan. Hal ini membuktikan bahwa spesifikasi 5 yakni salah satu *tray* mampu melakukan pergerakan slide ke luar dengan kecepatan maksimal  $1 \frac{cm}{s}$  terpenuhi.

## 5.2 Kesimpulan dan Saran

### 5.2.1 Kesimpulan

Kesimpulan dari analisis pengujian yang didapatkan untuk pengembangan robot pengantar makanan menunjukkan bahwa robot harus memenuhi kebutuhan yang telah dibuat. Pertama, robot dapat bergerak menuju titik tujuan yang telah ditentukan di dalam ruangan pada bidang datar dan permukaan yang rata, dengan meninjau tujuan ini berdasarkan hasil pengujian robot dapat berjalan dengan kecepatan linear rata-rata 0,245 yang tidak lebih dari 0,3 m/s , dengan begitu robot dapat memenuhi kebutuhan dari segi kecepatan pada kondisi lantai yang rata. Kedua , robot mampu bergerak ke segala arah/omni-directional, pengujian untuk kebutuhan ini dilakukan dengan cara memberi perintah pada robot untuk bergerak ke depan, belakang, kanan , dan kiri tanpa mengubah orientasi robot (robot tetap pada orientasi 0 derajat). dengan pergerakan yang dilakukan diambil data asli dan data dari sensor didapat rata - rata error pada perpindahan robot kurang lebih 10cm , hal ini membuktikan robot dapat bergerak ke segala arah tanpa harus mengubah orientasinya terlebih dahulu.

Ketiga, robot mampu mengidentifikasi dan membedakan meja pelanggan dengan menggunakan kode QR. diperlukan identifikasi meja untuk melakukan pemindaian pesanan sehingga robot dapat mengirim pesanan ke pelanggan dengan tepat atau tidak salah meja dalam mengantarkan pesanan. pada pengujian kali ini dilakukan peninjauan jarak optimal

robot dapat mendeteksi kode QR yang ada di meja, dan didapatkan jarak optimalnya 25cm, dengan demikian kebutuhan ini terpenuhi. Selanjutnya kebutuhan yang terakhir , robot didesain dengan trays beserta penutup di sekelilingnya. Salah satu tray memiliki fitur untuk menyajikan makanan kepada pelanggan. pada hasil integrasi hal ini telah teruji dimana robot ditutup rapat untuk menutup makanan dan terdapat tray yang dapat bergerak untuk mengeluarkan makanan, gunanya untuk menambah daya tarik pada robot.

### 5.2.2 Saran

Memperbaiki sistem navigasi robot akan sangat membantu dalam development robot, misalnya bisa ditingkatkan ke yang lebih fleksibel dan advance seperti menggunakan hector slam. Lalu mungkin dapat memanfaatkan kamera yang lebih optimal, ketimbang scan QR saja mungkin bisa mendeteksi keberadaan meja dan pelanggan dan mengkonversikan nilai koordinat ke koordinat robot sehingga dapat diintegrasikan pada pergerakan robot. Selain itu mungkin dapat memperbaiki kembali di beberapa mekanisme, seperti laci yang keluar masuk yang bisa di ganti dengan aktuator yang lebih simpel. Lalu yang terakhir bagian estetika robot mungkin dapat diperindah dengan sedikit memoles penutup dengan beberapa stiker contohnya.

## DAFTAR PUSTAKA

- [1] C. Spencer, "Restaurant Hires \$1000-a-Month Robot Waiter, Leading to Tip Surge," *Changing America*, Changing America. [Online]. Available: <https://thehill.com/changing-america/resilience/smart-cities/577452-restaurant-hires-1000-a-month-robot-waiter-and-tips/>
- [2] S. Stone, "Robots in Restaurants? What Restauranteurs Need to Know About This Growing Trend," *The Bottom Line*. [Online]. Available: <https://squareup.com/us/en/the-bottom-line/operating-your-business/robots-in-restaurants/>
- [3] C. Hu, "What Robots Can and Can't Do For A Restaurant," Popular Science. [Online]. Available: <https://www.popsci.com/technology/robots-in-restaurants/>
- [4] H. Taheri and C. X. Zhao, "Omnidirectional mobile robots, mechanisms and navigation approaches," *Mech Mach Theory*, vol. 153, p. 103958, 2020, doi: 10.1016/j.mechmachtheory.2020.103958.
- [5] Richtech Robotics, "The Matradee : Provide the Ultimate Experience with Our Serving Robot." [Online]. Available: [https://www.richtechrobotics.com/matradee#section\\_matradee-applications](https://www.richtechrobotics.com/matradee#section_matradee-applications)
- [6] Pudu Robotics, "PuduBot : An Intelligent Delivery Robot." [Online]. Available: <https://www.pudurobotics.com/products/pudubot>
- [7] Pudu Robotics, "SwiftBot : All-Powerful Delivery Robot." [Online]. Available: <https://www.pudurobotics.com/products/swiftbot>
- [8] K. Rogers, "Restaurants Prep for Long-Term Labor Crunch by Turning to Robots to Work the Fryer, Shuttle Food to Tables," CNBC News. [Online]. Available: <https://www.cnbc.com/2021/10/22/restaurants-looking-for-labor-and-speed-turn-to-robots-.html>
- [9] RobotLab, "PuduBot Robot for Assisted Living." [Online]. Available: <https://www.robotlab.com/assisted-living-robots/store/pudubot>
- [10] National Kitchen Equipment, "SWFBOT Pudu Versatile SwiftBot Delivery Robot." [Online]. Available: <https://www.nationalkitchenequipment.com.au/swfbot-pudu-versatile-swiftbot-delivery-robot/>

- [11] M. Y. Yusuf, “Daftar Gaji Karyawan Solaria dari Pramusaji hingga Manajer Toko,” IDX Channel. [Online]. Available: <https://www.idxchannel.com/milenomic/daftar-gaji-karyawan-solaria-dari-pramusaji-hingga-manajer-toko>
- [12] Y. Julia, “Rasa Koffie, Cafe Kekinian dengan Robot Pelayan yang Mengagumkan di Jakarta,” yoursay.id. [Online]. Available: <https://yoursay.suara.com/ulasan/2023/09/25/201500/rasa-koffie-cafe-kekinian-dengan-robot-pelayan-yang-mengagumkan-di-jakarta>
- [13] A. Aurellia, “Cafe Instagramable dan Canggih di Bandung, Makanan Diantarkan Robot,” Detik Jabar. [Online]. Available: <https://www.detik.com/jabar/kuliner/d-6049498/cafe-instagramable-dan-canggih-di-bandung-makanan-diantarkan-robot>
- [14] T. M. N. U. Akhund, M. A. B. Siddik, M. R. Hossain, M. M. Rahman, N. T. Newaz, and M. Saifuzzaman, “IoT Waiter Bot: A Low Cost IoT based Multi Functioned Robot for Restaurants,” *ICRITO 2020 - IEEE 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)*, pp. 1174–1178, 2020, doi: 10.1109/ICRITO48877.2020.9197920.
- [15] P. N. Reddy, P. S. Roy, M. Sanjay, A. Naveen, S. Jayabalan, and M. Sakhineti, “Fabrication of Low Cost Food Delivery Robot,” *7th International Conference on Communication and Electronics Systems, ICCES 2022 - Proceedings*, no. Icces, pp. 85–89, 2022, doi: 10.1109/ICCES54183.2022.9835764.
- [16] C. H. A. H. B. Baskoro, H. M. Saputra, M. Mirdanies, V. Susanti, M. F. Radzi, and R. I. A. Aziz, “An Autonomous Mobile Robot Platform for Medical Purpose,” *Proceeding - 2020 International Conference on Sustainable Energy Engineering and Application: Sustainable Energy and Transportation: Towards All-Renewable Future, ICSEEA 2020*, pp. 41–44, 2020, doi: 10.1109/ICSEEA50711.2020.9306161.
- [17] Y. Tan, B. Lew, K. Tan, K. Goh, K. Lee, and Z. Khot, “A New Automated Food Delivery System Using Autonomous Track Guided Centre-Wheel Drive Robot,” *Proceeding of the 2010 IEEE Conference on Sustainable Utilization and Development in Engineering and Technology University Tunku Abdul Rahman*, no. November, pp. 32–36, 2010, doi: <https://doi.org/10.1109/STUDENT.2010.5686986>.

- [18] W. H. Guan *et al.*, “D-Bot: A Food Serving Robot During Pandemic Situation,” *Proceedings of 2021 IEEE International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things, RAAICON 2021*, pp. 22–25, 2021, doi: 10.1109/RAAICON54709.2021.9929991.
- [19] M. Kamruzzaman and M. Tareq, “Design and implementation of a robotic technique based waiter,” *3rd International Conference on Electrical Information and Communication Technology, EICT 2017*, vol. 2018-Janua, no. December, pp. 1–5, 2018, doi: 10.1109/EICT.2017.8275250.
- [20] M. Z. Zulkifly and M. J. Homam, “Food Delivery Monitoring System with a Line Follower Robot,” *Evolution in Electrical and Electronic Engineering*, vol. 2, no. 2, pp. 757–765, 2021, [Online]. Available: <http://publisher.uthm.edu.my/periodicals/index.php/eeee>
- [21] Y. A. Lu and G. S. Huang, “Positioning and navigation of meal delivery robot using magnetic sensors and RFID,” *Proceedings - 2014 International Symposium on Computer, Consumer and Control, IS3C 2014*, pp. 808–811, 2014, doi: 10.1109/IS3C.2014.214.
- [22] A. Cheong, M. W. S. Lau, E. Foo, J. Hedley, and J. W. Bo, “Development of a Robotic Waiter System,” *IFAC-PapersOnLine*, vol. 49, no. 21, pp. 681–686, 2016, doi: 10.1016/j.ifacol.2016.10.679.
- [23] A. Javeed, V. G. Prakash, and S. P. Kulkarni, “Autonomous service robot,” *ACM International Conference Proceeding Series*, no. July, pp. 2–7, 2019, doi: 10.1145/3352593.3352626.
- [24] A. CHEONG, E. FOO, H. GAN, J. CHEN, and M. LAU, “Development of a Robotics Waiter System for the food and beverage industry,” no. October 2015, pp. 21–25, 2015, doi: 10.15224/978-1-63248-066-8-57.
- [25] C. S. Chen, C. J. Lin, and C. C. Lai, “Non-Contact Service Robot Development in Fast-Food Restaurants,” *IEEE Access*, vol. 10, pp. 31466–31479, 2022, doi: 10.1109/ACCESS.2022.3155661.
- [26] T. Tanksley, “Food Delivery Robots: The Comprehensive Guide,” Richtech Robotics. [Online]. Available: <https://www.richtechrobotics.com/blog/food-delivery-robots>
- [27] Pudu Robotics, “BellaBot : Premium Delivery Robot.” [Online]. Available: <https://www.pudurobotics.com/products/bellabot>

- [28] “Robotics Vision Processing: Object Detection and Tracking,” *Embedded Computing Design*, 2019. [Online]. Available: <https://embeddedcomputing.com/application/industrial/robotics-vision-processing-object-detection-and-tracking>
- [29] H. S. G. Supreeth and C. M. Patil, “Moving object detection and tracking using deep learning neural network and correlation filter,” *Proceedings of the International Conference on Inventive Communication and Computational Technologies, ICICCT 2018*, no. Icicct, pp. 1775–1780, 2018, doi: 10.1109/ICICCT.2018.8473354.
- [30] “What is Finite Element Analysis (FEA)?,” TWI. [Online]. Available: <https://www.twi-global.com/technical-knowledge/faqs/finite-element-analysis>
- [31] A. K. Dey, “Factor of Safety: Definition, Equation, Examples, Calculator (With PDF),” What is Piping. [Online]. Available: <https://whatispiping.com/factor-of-safety/>
- [32] Sandy Suryady and Eko Aprianto Nugroho, “Simulasi Faktor Keamanan dan Pembebanan Statik Rangka Pada Turbin Angin Savonius,” *Jurnal Ilmiah Multidisiplin*, vol. 1, no. 2, pp. 42–48, 2022, doi: 10.56127/jukim.v1i2.94.

## LAMPIRAN CD-1

### Lampiran 1 Interpretasi Kebutuhan

**Costumer:** Erwin Susanto, S.T., M.T., Ph.D.  
dan Irham Mulkan Rosdiana,  
S.T., M.T.

**Address:** Jl. Telekomunikasi no. 1, Terusan  
Buah Batu – Bojongsoang,  
Telkom University, Sukapura,  
Kec. Dayeuhkolot, Kabupaten  
Bandung, Jawa Barat 40257

**Telephone:** Pak Erwin: +62821-2151-4878  
Pak Irham: +62813-3000-3095

**Email:**  
[erwinelektro@telkomuniversity.ac.id](mailto:erwinelektro@telkomuniversity.ac.id);  
[irhammulkan@telkomuniversity.ac.id](mailto:irhammulkan@telkomuniversity.ac.id)

**Interviewer(s):** Rifqy Fachrizi, Gebby  
Novalisza, Muhammad  
Rafly

**Date:** Jumat, 22 September 2023

**Currently Uses:** Deliver food to the  
Costumer

<i>Question/Prompt</i>	<i>User Statement</i>	<i>Interpreted Need</i>
Keunggulan produk yang sudah ada	Mampu bergerak secara menuju titik tujuan yang ditentukan oleh pengguna di dalam ruangan pada bidang datar	Dibutuhkan robot yang mampu bergerak menuju titik tujuan yang telah ditentukan oleh pengguna di dalam ruangan pada bidang datar
Keunggulan produk yang sudah ada Kekurangan produk yang sudah ada	Mampu bergerak ke segala arah/ <i>omni-directional</i>	Dibutuhkan robot yang mampu bergerak ke segala arah/ <i>omni-directional</i>
	Pergerakan memanfaatkan <i>sensor fusion</i> untuk memperoleh data yang akurasi tentang lokasi target yang diinginkan	Dibutuhkan robot yang mampu bergerak dengan memanfaatkan <i>sensor fusion</i>
	Mampu mendeteksi objek yang berada di lingkungan sekitar	Dibutuhkan robot yang mampu mendeteksi objek di sekitarnya
	Produk dipasarkan dengan harga yang cukup tinggi	Dibutuhkan robot yang dikembangkan dengan harga yang jauh lebih terjangkau
Kekurangan produk yang sudah ada Rekomendasi Penyempurnaan	Proses produksi melibatkan komponen dan peralatan yang sulit ditemukan di Indonesia	Dibutuhkan robot dengan komponen dan peralatan yang banyak tersedia di Indonesia.

	Kebanyakan masih berupa prototipe dengan ukuran yang belum sesuai dengan kondisi aktual	Dibutuhkan robot dengan ukuran yang telah disesuaikan dengan kondisi aktual.
	<i>Tray</i> pada produk yang sudah ada jarang disertai dengan penutup	Dibutuhkan robot yang didesain dengan <i>trays</i> beserta penutup di sekelilingnya. Salah satu <i>tray</i> memiliki fitur untuk menyajikan makanan kepada pelanggan.
Rekomendasi Penyempurnaan	<i>Tray</i> pada produk bersifat statis (tidak dapat digerakkan)	Dibutuhkan robot yang didesain dengan <i>trays</i> beserta penutup di sekelilingnya. Salah satu <i>tray</i> memiliki fitur untuk menyajikan makanan kepada pelanggan.

## Lampiran 2 Pengelompokan Kebutuhan dan Ranking Kebutuhan

### Skala Kebutuhan:

\*\*\* Fitur Utama

\*\* Fitur yang Bagus

\* Fitur Pendukung

\*\*\* **Robot mampu bergerak menuju titik tujuan yang telah ditentukan oleh pengguna dengan *sensor fusion***

\*\*\* **Robot mampu bergerak di dalam ruangan pada bidang datar**

\*\* Robot dikembangkan dengan harga yang jauh lebih terjangkau

\*\* Robot dikembangkan dengan komponen dan peralatan yang banyak tersedia di Indonesia

\*\*\* **Robot mampu bergerak ke segala arah/*omni-directional***

\*\*\* **Robot mampu mendeteksi objek di lingkungan sekitarnya**

\*\*\* **Robot didesain dengan *tray* beserta penutup di sekelilingnya**

\*\*\* **Salah satu *tray* dapat digerakkan untuk menyajikan makanan ke pelanggan**

\*\* Robot didesain ukuran yang telah disesuaikan dengan kondisi aktual

### Lampiran 3 Penyusunan prioritas kebutuhan

Parameter skala kepentingan fitur pada robot pengantar makanan:

6. Fitur yang diinginkan, tetapi akan mempertimbangkan alat dengan fitur ini
7. Fitur tidak penting, tetapi tidak masalah jika diterapkan pada alat
8. Akan baik memiliki fitur ini tetapi tidak perlu diterapkan
9. Fitur sangat diperlukan tetapi akan dipertimbangkan jika tidak diterapkan
10. Fitur sangat diperlukan dan akan diterapkan pada alat

Tabel berikut ini merupakan skala kepentingan/prioritas fitur pada robot pengantar makanan:

Skala	Fitur	Keunikan*
5	Robot mampu bergerak menuju titik tujuan yang telah ditentukan di dalam ruangan pada bidang datar dengan <i>sensor fusion</i> .	<input type="checkbox"/>
5	Robot mampu bergerak ke segala arah/ <i>omni-directional</i> .	<input type="checkbox"/>
5	Robot mampu mendeteksi objek di lingkungan sekitarnya.	<input type="checkbox"/>
5	Robot didesain dengan <i>tray</i> beserta penutup di sekelilingnya. Salah satu <i>tray</i> dapat digerakkan untuk menyajikan makanan ke pelanggan.	<input checked="" type="checkbox"/>

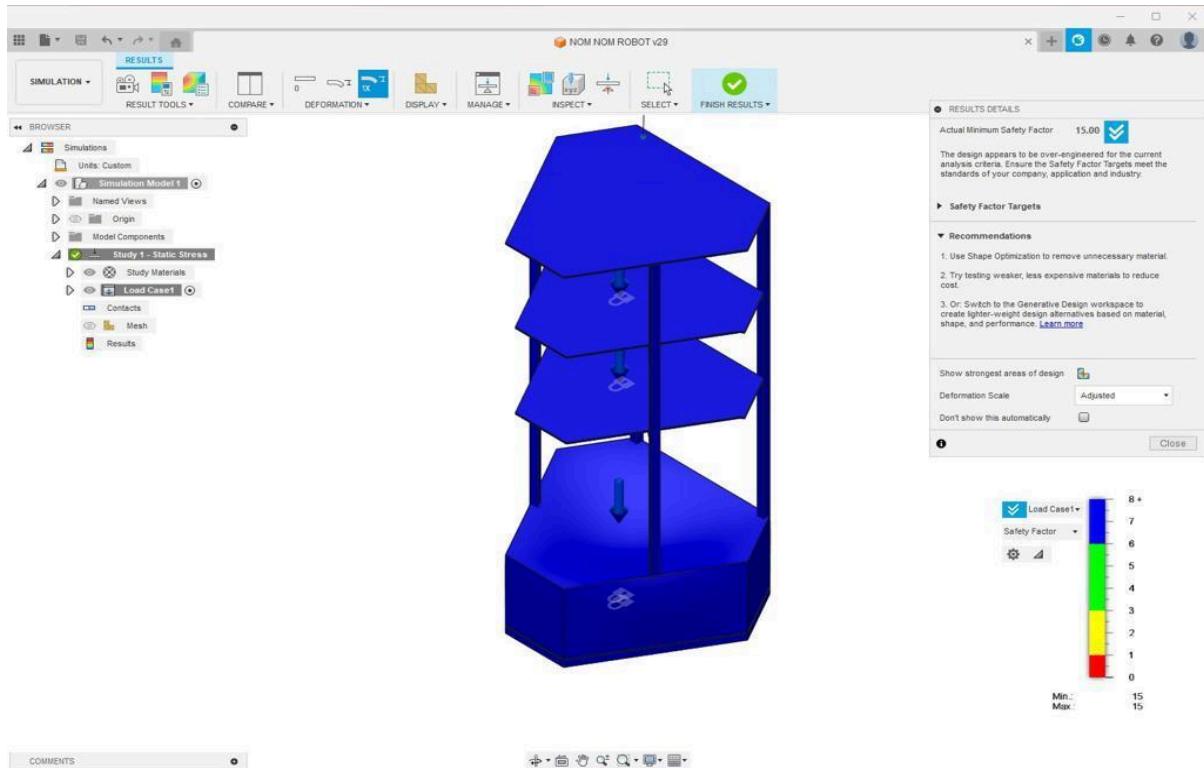
\* Fitur yang dianggap unik diberikan tanda cross (☒)

### Lampiran 4 Mission Statement

Mobilisasi Robot Pengantar Makanan Berbasis <i>Sensor Fusion</i> dan <i>Object Detection</i>	
Deskripsi Produk	Robot yang mampu mengantarkan makanan kepada pelanggan
Keuntungan	<ul style="list-style-type: none"> <li>- Memiliki harga yang lebih terjangkau jika dibandingkan produk robot pengantar makanan yang komersial di pasar.</li> <li>- Bahan baku dan komponen yang mudah diperoleh serta proses manufaktur melibatkan peralatan yang sederhana.</li> <li>- Dapat digunakan secara terus-menerus dengan masa pakai yang cukup lama. Mudah dalam penggunaan serta perbaikan/pemeliharaan</li> <li>- Mampu menjaga kualitas makanan agar tetap higienis sampai ke pelanggan</li> </ul>
Target Utama	Semua pemilik usaha F&B ( <i>Food &amp; Beverages</i> )
Target Sekunder	Kafe, restoran, dan/atau kantin yang memiliki layanan <i>dine-in</i>

Asumsi	<ul style="list-style-type: none"> <li>- Robot mampu bergerak menuju titik tujuan yang telah ditentukan di dalam ruangan pada bidang datar dengan <i>sensor fusion</i>.</li> <li>- Robot mampu bergerak ke segala arah/<i>omni-directional</i>.</li> <li>- Robot mampu mendeteksi objek yang berada di lingkungan sekitar</li> <li>- Robot didesain dengan <i>trays</i> beserta penutup di sekelilingnya. Salah satu <i>tray</i> memiliki fitur untuk menyajikan makanan kepada pelanggan.</li> </ul>
<i>Stakeholder</i>	<ul style="list-style-type: none"> <li>- Pengguna</li> <li>- Laboratorium Electronics and Intelligence Robotic Research Group (EIRRG)</li> </ul>

## LAMPIRAN CD-2





## **LAMPIRAN CD-3**

## **LAMPIRAN CD-4**

## **LAMPIRAN CD-5**