



UNIVERSITAS INDONESIA

**RANCANG BANGUN *HUMAN INTERACTION* BERBASIS
ANDROID PADA ROBOT PENERIMA TAMU**

SKRIPSI

IMAM ASKOLANI

0806 455 282

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2012**



UNIVERSITAS INDONESIA

**RANCANG BANGUN *HUMAN INTERACTION* BERBASIS ANDROID
PADA ROBOT PENERIMA TAMU**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

IMAM ASKOLANI

0806 455 282

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2012**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Imam Askolani
NPM : 0806 455 282
Tanda Tangan : 
Tanggal : 25 Juni 2012



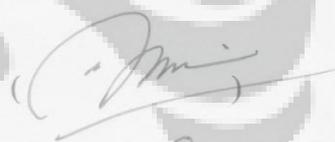
HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Imam Askolani
NPM : 0806 455 282
Program Studi : Teknik Elektro
Judul Skripsi : Rancang Bangun *Human Interaction* Berbasis
Android pada Robot Penerima Tamu

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing :
Dr. Abdul Muis, ST., M.Eng



Penguji 1 :
Ir. Wahidin Wahab M.Sc, Ph.D



Penguji 2 :
Prof. Dr.Eng. Drs. Benyamin Kusumoputro M.Eng.



Ditetapkan di : Depok

Tanggal : 25 Juni 2012

KATA PENGANTAR

Alhamdulillah, Puji dan syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat dan karunia sehingga proses penyusunan skripsi ini berjalan dengan baik dan dapat diselesaikan dengan baik. Skripsi ini diajukan untuk memenuhi salah satu persyaratan memperoleh gelar sarjana teknik dan sebagai bentuk kontribusi untuk kemajuan teknologi bangsa.

Penulis menyadari bahwa bantuan dan bimbingan serta dukungan dari berbagai pihak sangat berpengaruh terselesainya tulisan ini. Oleh karena itu, penulis mengucapkan banyak terima kasih kepada:

1. Untuk Mamak, Ayah, Yudha, Vioni, Novi, Oman, Isat, Nanang, Nine, Mamaci, So Eng, Ci Lin, Wa Nga, Mamaso, Waso, Bi Erna, Bi Nita dan saudara-saudara lainnya yang selalu mendukung demi kesuksesan dan kemandirian hidup penulis;
2. Dr. Abdul Muis, S.T., M.Eng selaku dosen pembimbing yang telah memberikan bimbingan, arahan, fasilitas, dan kepercayaan dalam melakukan penelitian;
3. Teman-teman bimbingan dan Keluarga Besar TRUI, atas masukan teknis, akomodasi peralatan, semangat kebersamaan, dan sebagai tempat bernaung penulis dalam mencari ilmu serta berbagi canda-tawa yang meredakan kepenatan.

Akhir kata penulis berharap dapat membalas kebaikan segala pihak yang telah membantu, dan berdoa semoga Allah membalas dengan kebaikan yang lebih banyak. Semoga laporan ini dapat bermanfaat bagi pengembangan ilmu pengetahuan dan teknologi bangsa.

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI
UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Imam Askolani

NPM : 0806 455 282

Program Studi : Teknik Elektro

Departemen : Teknik Elektro

Fakultas : Teknik

Jenis karya : Skripsi

demikian demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul:

**“RANCANG BANGUN HUMAN INTERACTION BERBASIS ANDROID PADA
ROBOT PENERIMA TAMU”**

berserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di: Depok

Pada tanggal : 25 Juni 2012

Yang menyatakan,



(Imam Askolani)

ABSTRAK

Nama : Imam askolani
Program Studi : Teknik Elektro
Judul : Rancang Bangun *Human Interaction* Berbasis Android
pada Robot Penerima Tamu

Robot penerima tamu merupakan salah satu jenis *service robot*. Dengan menggunakan protokol pengiriman data melalui Bluetooth serial, data dari tiap sensor yang berasal dari sistem stereo vision, *mobile robot* dan juga pergerakan lengan akan diolah oleh sistem android sebagai masukan untuk berinteraksi dengan manusia. Untuk meningkatkan kecerdasan robot agar lebih menghampiri kemampuan manusia dimanfaatkan sistem *text to speech* dan pengenalan suara dengan *voice recognition* pada android. Evaluasi kinerja sistem diuji dengan pemberian kombinasi kondisi sistem yang dikomparasi dengan database SQLite.

Kata Kunci: Robot penerima tamu, Android, *voice recognition*, *text to speech*, database SQLite

ABSTRACT

Name : Imam Askolani
Study Program : Electrical Engineering
Title : Human Interaction Development for Android-Based
Receptionist Robot

Receptionist robot is one of the service robots type. By using the protocol via Bluetooth serial data transmission, Android system will compute each data from the stereo vision system, and also the movement of mobile robot base and arm movement. To improve the intelligence of the robot to approach human capabilities, text to speech system and voice recognition on android is applied. Performance evaluation of the system was tested by comparing combination of system conditions given with SQLite database.

Keyword: receptionist robot, Android, voice recognition, text to speech, database SQLite

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan Penelitian.....	2
1.3 Pembatasan Masalah.....	2
1.4 Metodologi Penelitian.....	3
1.5 Sistematika Penulisan	3
BAB 2 SISTEM PENDUKUNG	5
2.1 Sistem Android.....	5
2.1.1 Perangkat Keras.....	7
2.1.2 Perangkat Lunak.....	15
2.2 Sistem Luar	24
2.2.1 Sistem Lengan	24
2.2.2 Kamera.....	25
2.2.3 Sensor SRF08.....	27
2.2.4 <i>Mobile robot</i>	28
2.3 Komunikasi	28
2.4 Eclipse.....	34

BAB 3 PERANCANGAN HUMAN INTERACTION.....	37
3.1 Perancangan Signaling Sistem Luar	38
3.1.1 Protokol Pengiriman Data	38
3.1.2 Sistem Pencitraan	40
3.1.3 Sistem Kecepatan	40
3.1.4 Sensor Jarak.....	41
3.1.5 Sistem Lengan	41
3.2 Perancangan Interaksi Manusia ke Robot	42
3.3 Perancangan Interaksi Robot ke Manusia	43
BAB 4 PENGUJIAN DAN ANALISIS.....	45
4.1 Hasil Pengujian dan Analisis Interaksi Manusia ke Robot	45
4.2 Hasil Pengujian Pengenalan Suara Berdasarkan Jarak dan Pelafalan	47
BAB 5 PENUTUP	49
5.1 Kesimpulan.....	49
DAFTAR REFERENSI	50

DAFTAR GAMBAR

Gambar 1.1 Service Robot	1
Gambar 2.1 Galaxy Spica.....	5
Gambar 2.2 Subsistem Android.....	6
Gambar 2.3 Aksis Accelerometer.....	8
Gambar 2.4 Contoh Aplikasi Pemanfaatan Digital Compass	10
Gambar 2.5 Audio System	11
Gambar 2.6 Prinsip Kerja Microphone	12
Gambar 2.7 Skema Speaker	13
Gambar 2.8 Diagram Alir Voice Recognition.....	18
Gambar 2.9 Perbedaan SQLite dengan Tipe Database RDBMS.....	20
Gambar 2.10 Tampilan Muka SQLite <i>Database</i> Browser.....	22
Gambar 2.11 Lengan Robot	25
Gambar 2.12 Diagram Alir Sistem Stereo Vision	26
Gambar 2.13 Implementasi Sistem Stereo Vision.....	26
Gambar 2.14 Prinsip Kerja SRF08	27
Gambar 2.15 Mobile Robot.....	28
Gambar 2.16 Bluetooth Serial	29
Gambar 2.17 Blok Diagram Bluetooth Serial	29
Gambar 2.18 <i>Voltage Divider</i> untuk Mendapatkan Tegangan 3.3 V	31
Gambar 2.19 Skematik Sistem Minimum AMS1117	31
Gambar 2.20 Layout PCB Bluetooth Serial	32
Gambar 2.21 PCB Bluetooth Serial.....	32
Gambar 3.1 Diagram Alir Robot Penerima Tamu.....	37

DAFTAR TABEL

Tabel 2.1 Spesifikasi Galaxy Spica	7
Tabel 2.2 Konfigurasi Pin Bluetooth Serial	30
Tabel 3.1 Paket Pengiriman Data	38
Tabel 3.2 Identitas Data	39
Tabel 3.3 Data Pencitraan	40
Tabel 3.4 Data Kecepatan	40
Tabel 3.5 Data Jarak	41
Tabel 3.6 Data Status Lengan.....	41
Tabel 3.7 Tabel Skenario <i>Human Interaction to Robot</i>	43
Tabel 3.8 Tabel Skenario <i>Robot Interaction</i>	44
Tabel 4.1 Tabel Hasil Pengujian Interaksi Manusia ke Robot.....	45
Tabel 4.2 Tabel Hasil Pengujian interaksi Robot ke Manusia	46
Tabel 4.3 Tabel Kebenaran Berdasarkan Jarak (cm).....	47
Tabel 4.4 Tabel Kebenaran Pengaruh Variasi Sumber Suara	48

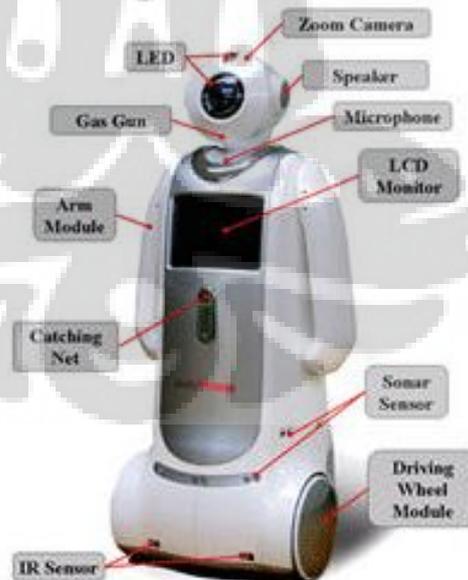
BAB 1

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi makin banyak ilmuwan yang meneliti dalam konsentrasi ilmu robotika. Bayangan masa depan dimana robot akan berinteraksi dengan manusia seperti makhluk hidup sesungguhnya dan membantu manusia dalam banyak hal menjadi tujuan utama pada *roboticist* atau ahli robot. Salah satu robot yang mulai banyak dikembangkan adalah *service robot*, yaitu robot yang dapat menggantikan pekerjaan manusia seperti menjadi bartender, membersihkan rumah atau menjadi penerima tamu.

Service robot memiliki tingkat kompleksitas yang tinggi. Para insinyurnya selalu berusaha agar bisa menghampiri bentuk kemampuan manusia; ciptaan Tuhan yang paling sempurna. Kompleksitas tersebut dapat diurai menjadi berbagai macam bidang, misal: tentang struktur dan mekanisme sistem mekanik, kehandalan dan kapabilitas pengendali, sensor, dan aktuator, kecerdasan buatan, teknologi mobilitas, teknologi komunikasi dan kerjasama, teknik pentautan sub-sistem, dan sebagainya.



Gambar 1.1 Service Robot

Salah satu bentuk *service robot* yang merupakan sebuah topik hangat saat ini adalah humanoid robot. Belakangan ini, kemajuan teknologi disekitar telah memungkinkan robot untuk menyerupai bentuk dan tingkah laku manusia. Kita dijanjikan oleh film, buku fiksi, dan televisi bahwa humanoid robot akan memasak untuk kita, membersihkan rumah, menjadi sahabat, bahkan jatuh cinta dengan manusia.

Banyak sisi dari humanoid robot yang bisa dikembangkan, mulai dari sistem gerak, komunikasi, tingkah laku, hingga sistem interaksi ke manusia dan lingkungan sekitarnya. Karena banyaknya subsistem tersebut mengharuskan adanya penelitian yang bersifat paralel antara satu aspek dengan yang lain, yang nanti akan diintegrasikan hasil akhirnya ke satu robot utuh.

Khususnya dalam sistem interaksi ke manusia atau human interaction terdapat banyak aspek lagi yang bisa dikembangkan, mulai dari cara gerak, bagaimana robot dapat mengerti kemauan manusia dan sebaliknya manusia mengerti akan aktifitas yang robot lakukan.

Android merupakan salah satu teknologi terbaru yang dapat dikembangkan untuk keperluan dalam ilmu robotika karena memiliki sensor yang lengkap dan fitur sistem pengenalan suara dengan *voice recognition* serta pembuatan suara dengan *text to speech* layaknya komputer.

1.2 Tujuan Penelitian

Adapun tujuan utama dari penelitian ini adalah untuk memberikan kontribusi nyata dalam ilmu robotika. Khususnya dalam bidang interaksi antara manusia dengan robot. Dengan menggunakan fitur *voice recognition* dan *text to speech* pada Android serta dukungan dari sensor-sensor yang terpasang pada robot diharapkan kemampuan interaksi robot penerima tamu dapat menyamai manusia.

1.3 Pembatasan Masalah

Masalah pokok dari penelitian ini adalah perancangan algoritma dan protokol untuk interaksi antara manusia dengan robot penerima tamu berbasis Android dengan menggunakan *Voice Recognition* dan *Text to Speech* dengan

berdasarkan masukan dari sensor-sensor yang terpasang dan dipakai di robot penerima tamu.

1.4 Metodologi Penelitian

Metode penelitian yang digunakan dalam penyusunan skripsi ini meliputi:

- a. Studi literatur mengenai sistem yang sudah pernah dikembangkan di internet
- b. Pendekatan diskusi dengan pembimbing skripsi
- c. Perakitan perangkat keras dan perancangan perangkat lunak
- d. Pengujian sistem terintegrasi

1.5 Sistematika Penulisan

Laporan ini akan dibagi menjadi lima bab utama dengan pembagian penjelasan ringkas sebagai berikut:

1. Pendahuluan

Pada bab ini akan dijelaskan tentang latar belakang permasalahan yang muncul dan alasan pentingnya penyelesaian permasalahan tersebut, tujuan penelitian, pembatasan permasalahan yang dibahas dalam laporan ini, metode penelitian, serta susunan penulisan skripsi.

2. Sistem Pendukung

Pada bab ini akan dipaparkan mengenai sistem-sistem apa saja yang mendukung hingga bekerjanya robot penerima tamu. Dan juga dijelaskan tentang komponen-komponen pendukung yang membuat penelitian ini dapat dilaksanakan.

3. Perancangan Human Interaction

Pada bab ini akan dijelaskan mengenai perancangan dari sistem robot di bagian Android yang mengintegrasikan segala masukan dan keluaran dari dan ke Android sebagai bentuk *human interaction*.

4. Pengujian dan Analisis

Pada bab ini dijelaskan tentang objek eksperimen, piranti eksperimen, konvensi yang dilakukan dalam eksperimen, eksekusi eksperimen, dan pemaparan hasil eksperimen beserta analisis terhadap hasil eksperimen.

5. Penutup

Bab terakhir ini berisi perihal-perihal yang diperoleh selama penelitian, terutama yang berhubungan dengan proses penyelesaian masalah yang ada dan riset-riset yang dapat dilakukan kedepannya.



BAB 2

SISTEM PENDUKUNG

Setiap sistem mempunyai perangkat dasar yang harus dimiliki agar dapat berfungsi dan menghasilkan keluaran sesuai dengan yang diinginkan. Pada sistem robotika terdiri atas tiga subsistem utama yaitu subsistem elektrik, subsistem mekanik dan subsistem artificial intelligent (AI) atau perangkat lunak.

Pada bagian ini akan dijelaskan mengenai gambaran umum dari sistem pendukung robotika yang akan digunakan serta teori dasar untuk pengembangan perancangan robot penerima tamu.

2.1 Sistem Android

Robot penerima tamu adalah robot yang berfungsi untuk menerima tamu layaknya manusia. Untuk dapat melakukan pekerjaan robot harus memiliki “indera” yang menyerupai milik manusia dan dapat mengolah hasilnya untuk nantinya mengeluarkan respon yang semestinya dari robot. Untuk itu diperlukan perangkat yang dapat difungsikan sebagai sarana masukan dan keluaran robot yang efisien dan multifungsi.



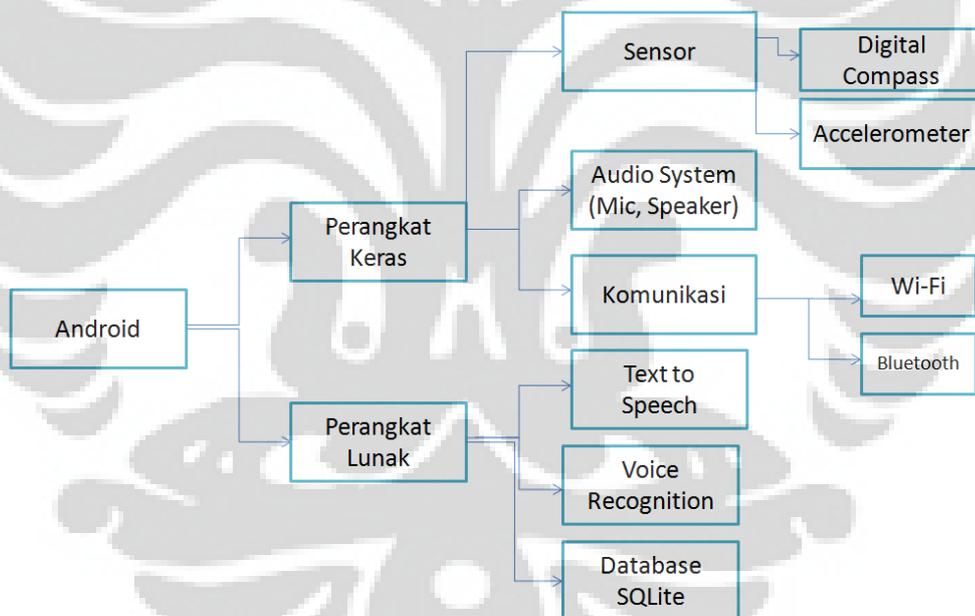
Gambar 2.1 Galaxy Spica

Salah satu perangkat yang dapat difungsikan untuk dapat melakukan pekerjaan itu adalah ponsel Android. Ponsel Android yaitu jenis ponsel dengan sistem operasi *open source* yang dikembangkan oleh Google, dikenal dengan nama Android. Awalnya sistem ini dikembangkan oleh Android Inc., lalu tahun

2005 dibeli oleh Google. Perangkat ponsel Android, umumnya dilengkapi dengan layar sentuh, sensor *G-Sensor*, *E-compass sensor / Proximity*, *Ambient Light sensor* dan *camera* serta GPS.

Android merupakan produk *open source* dimana semua pengguna bebas membuat dan memodifikasi software dengan memanfaatkan fitur yang ada. Menggunakan bahasa pemrograman berorientasi objek yang dalam hal ini adalah Java serta *software compiler* seperti Eclipse.

Pada skripsi kali ini digunakan ponsel keluaran Samsung dengan tipe Galaxy Spica. Fitur sensor yang sudah mencukupi yaitu *GPS/Wifi*, *accelerometer* dan *digital compass*. Dengan sistem operasi yang telah ditingkatkan menjadi 2.2 (Froyo) lebih memudahkan dalam pembuatan aplikasi yang diinginkan karena memiliki fitur *software* yang lebih lengkap. Ponsel ini juga telah dilengkapi dengan Bluetooth yang nantinya digunakan sebagai media pengiriman data.



Gambar 2.2 Subsistem Android

Pada diagram di atas dapat terlihat bahwa Android terdiri atas sistem perangkat keras dan perangkat lunak. Adapun perangkat keras yang ada adalah sensor seperti *Digital Compass* dan *accelerometer*, serta audio system seperti microphone dan speaker. Untuk perangkat lunak yang melengkapi Android seperti

text to speech, voice recognition, database tipe SQLite. Seiring perkembangan teknologi akan ada banyak lagi fitur-fitur yang makin melengkapi Android.

Berikut ini adalah spesifikasi dari Galaxy Spica yang digunakan dalam pengujian ini:

Tabel 2.1 Spesifikasi Galaxy Spica

Dimensi	115x57x13.2 mm
Berat	124 g
Baterai	Talk time hingga 11 jam 30 menit, Stand-by hingga 650 jam
Sistem Operasi	Android OS, v2.2.2 (setelah dimodifikasi)
Memori	180 MB (internal) + microSD hingga 32 GB (eksternal)
Prosesor	800 MHz
Konektifitas	Bluetooth v2.1 A2DP, USB, GPRS kelas 12 (48 kbps), EDGE kelas 12, 3G (HSDPA 3.6 MBps), WLAN (Wi-Fi 802.11 b/g)
Ukuran Layar	320x480 piksel, 3.2 inches
Warna Layar	TFT capacitive touchscreen, 16 juta warna
Frekuensi Kerja / Band	GSM 850 / 1800 / 1900 (HSDPA 900 / 2100)
Browser	HTML
Warna	Hitam, Putih
Fitur	GPS + A-GPS support, Digital compass, Speakerphone, Google search, Google Maps, Gmail

Pada skripsi kali ini akan lebih dikhususkan pada penggunaan *Voice Recognition* dan *Text to Speech* yang ada pada Android serta penggunaan Bluetooth untuk pengiriman data serial. Dengan ketiga hal tersebut diharapkan robot penerima tamu mempunyai *interface* dan sistem navigasi yang sesuai keinginan.

2.1.1 Perangkat Keras

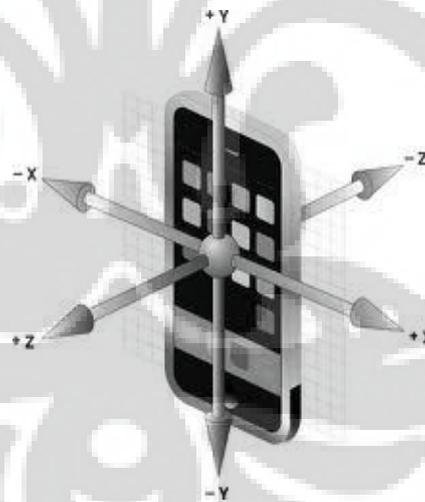
Perangkat keras dalam Android adalah perangkat-perangkat yang terdapat dalam Android yang membuatnya mempunyai fitur-fitur yang layak dipakai pada robot penerima tamu dalam keperluan *human interaction*. Adapun perangkat-perangkat keras penting yang dapat diperdagangkan dalam

pembuatan robot adalah *accelerometer*, *digital compass*, *audio system* (*microphone & speaker*), *WiFi*, dan *Bluetooth*.

2.1.1.1 *Accelerometer*

Accelerometer adalah sebuah *transduser* yang berfungsi untuk mengukur percepatan, mendeteksi dan mengukur getaran, ataupun untuk mengukur percepatan akibat gravitasi bumi. *Accelerometer* juga dapat digunakan untuk mengukur getaran yang terjadi pada kendaraan, bangunan, mesin, dan juga bisa digunakan untuk mengukur getaran yang terjadi di dalam bumi, getaran mesin, jarak yang dinamis, dan kecepatan dengan ataupun tanpa pengaruh gravitasi bumi.

Percepatan merupakan suatu keadaan berubahnya kecepatan terhadap waktu. Bertambahnya suatu kecepatan dalam suatu rentang waktu disebut juga percepatan (*acceleration*). Jika kecepatan semakin berkurang daripada kecepatan sebelumnya, disebut *deceleration*. Percepatan juga bergantung pada arah/orientasi karena merupakan penurunan kecepatan yang merupakan besaran vektor. Berubahnya arah pergerakan suatu benda akan menimbulkan percepatan pula.



Gambar 2.3 Aksis Accelerometer

Prinsip kerja dari *transduser* ini berdasarkan hukum fisika bahwa apabila suatu konduktor digerakkan melalui suatu medan magnet, atau jika suatu medan magnet digerakkan melalui suatu konduktor, maka akan timbul suatu tegangan induksi pada konduktor tersebut. *Accelerometer* yang diletakan di permukaan bumi dapat mendeteksi percepatan 1g (ukuran gravitasi bumi) pada titik

vertikalnya, untuk percepatan yang dikarenakan oleh pergerakan horizontal maka *accelerometer* akan mengukur percepatannya secara langsung ketika bergerak secara horizontal. Hal ini sesuai dengan tipe dan jenis sensor *Accelerometer* yang digunakan karena setiap jenis sensor berbeda-beda sesuai dengan spesifikasi yang dikeluarkan oleh perusahaan pembuatnya. Saat ini hampir semua sensor/*transducer accelerometer* sudah dalam bentuk digital (bukan dengan sistem mekanik) sehingga cara kerjanya hanya berdasarkan temperatur yang diolah secara digital dalam satu *chip*.

Salah satu penggunaan *accelerometer* yang sangat umum yaitu dalam sistem *airbag* yang terdapat pada kendaraan, khususnya mobil. *Accelerometer* ini digunakan untuk mendeteksi penurunan percepatan yang sangat besar yang biasanya terjadi ketika terjadinya tabrakan antar kendaraan. *Accelerometer* banyak digunakan untuk menghitung percepatan dan penurunan percepatan dari sebuah kendaraan. *Accelerometer* membantu untuk mengevaluasi performansi dari mesin dan sistem percepatan dan juga *breaking system* (sistem penurunan percepatan). Kecepatan yang biasa ditampilkan pada kendaraan anda umumnya didapatkan dari penggunaan *accelerometer*. Selain itu juga biasa digunakan untuk menghitung vibrasi pada kendaraan, mesin, bangunan, dan sistem keamanan pada kendaraan (*safety installation*). *Accelerometer* juga dapat mengkalkulasi percepatan yang diakibatkan oleh gravitasi bumi. *Accelerometer* yang menghitung gravitasi secara spesifik digunakan pada gravimetry, disebut sebagai gravimeter. *Notebook* atau laptop juga dilengkapi dengan *accelerometer* untuk mengevaluasi guncangan yang dirasakan oleh laptop tersebut. *Accelerometer* pada laptop biasanya digunakan pada sistem Sudden Motion Sensor, yang biasa digunakan untuk mendeteksi jatuhnya laptop. Jika kondisi pada saat jatuh terdeteksi, *hard disk drive* yang ada akan diproteksi sehingga tidak terjadi *data loss*. Sekarang ini juga terdapat *notebook* yang menggunakan *accelerometer* untuk secara otomatis mengubah arah layar (menjadi miring ataupun terbalik) sesuai dengan arah monitor tersebut ditegakkan (*portrait* atau *landscape*). Terdapat juga sejumlah *handphone* yang menggunakan *accelerometer* untuk mengubah lagu yang dimainkan (*Track Switching*). *Camera recorder* menggunakan *accelerometer*

untuk menstabilkan gambar (*image stabilization*). Kamera digital menggunakan *accelerometer* untuk menu pilihan anti blur ketika mengambil gambar.

Untuk dapat menggunakan accelerometer pada Android, dalam penulisan program perlu ditambahkan kelas-kelas khusus yaitu:

```
import Android.hardware.Sensor;
import Android.hardware.SensorEvent;
import Android.hardware.SensorEventListener;
import Android.hardware.SensorManager;
```

Setelah itu akan muncul method khusus untuk setiap perubahan data accelerometer, baik perubahan akurasi maupun perubahan nilai.

```
public void onAccuracyChanged(Sensor sensor, int accuracy) {
}

public void onSensorChanged(SensorEvent event) {
}
```

Lakukan penulisan

```
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
accelerometer = sensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
registerListener(sensorListener, Sensor.TYPE_ACCELEROMETER,
SensorManager.SENSOR_DELAY_NORMAL);
```

2.1.1.2 Digital Compass

Digital Compass sesuai namanya adalah suatu perangkat digital yang dapat menunjukkan posisi arah mata angin. Secara umum prinsip kerja yang dimiliki oleh *Digital Compass* hampir sama dengan Kompas analog biasa dimana ia bekerja dengan cara mendeteksi besaran nilai magnetik yang terdapat di sekitarnya. Yang menjadi perbedaan ialah cara mereka mendeteksi nilai magnetik tersebut dimana pada *Digital Compass* digunakan komponen-komponen digital untuk menentukannya.



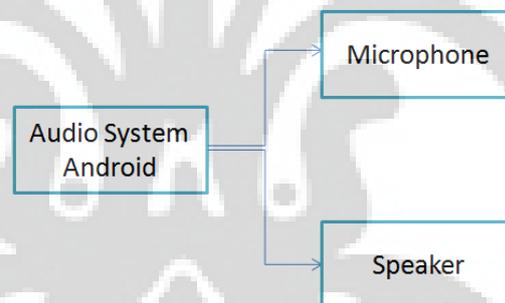
Gambar 2.4 Contoh Aplikasi Pemanfaatan Digital Compass

Ada beberapa cara untuk menghitung nilai besaran magnetik tersebut, namun perangkat yang paling umum digunakan adalah penggunaan Magnetometer. Perangkat ini memiliki kemampuan untuk mendeteksi secara elektronikal perbedaan medan magnet yang dimiliki oleh bumi dari gangguan yang disebabkan oleh elemen eksternal seperti bahan fero-magnetik dan medan magnet yang dibuat oleh perangkat elektronik mobile.

Selain penggunaan Magnetometer tersebut, umum juga digunakan Fibre Optic Gyrocompass dalam *Digital compass* terutama di bidang pelayaran. Berbeda dengan Kompas pada umumnya, Fibre Optic Gyrocompass tidak memiliki komponen bergerak seperti kompas pada umumnya. Ia menggunakan beberapa Sensor Fibre Optic Gyroscope dan Komputer untuk mendeteksi utara.

2.1.1.3 Audio System

Secara umum audio system yang mendukung Android terbagi atas dua hal yaitu microphone dan speaker.



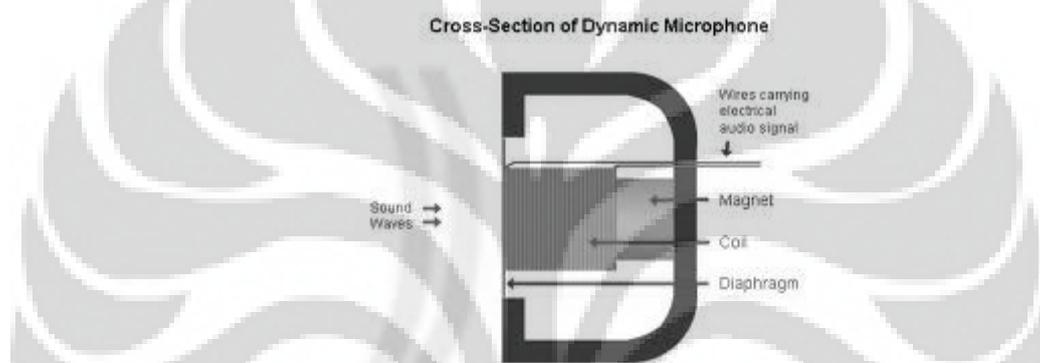
Gambar 2.5 Audio System

a. Microphone

Mic atau Microfon adalah suatu alat masukan (alat input) berupa suara. Ia memiliki prinsip kerja dengan cara untuk mengubah daya akustik menjadi energi listrik yang memiliki karakteristik gelombang dasarnya serupa. Secara sederhana, mikrofon mengkonversi gelombang suara menjadi tegangan listrik. Selanjutnya dari tegangan listrik ini ia dapat diolah oleh perangkat-perangkat tertentu dan kemudian ia dapat diubah lagi menjadi suara dengan bantuan speaker.

Ketika kita berbicara, gelombang suara yang dihasilkan oleh suara kita akan membawa energi ke arah microphone. Didalam microphone ada sebuah lapisan diafragma (biasanya terbuat dari plastik tipis) yang akan bergerak maju mundur ketika gelombang suara menabraknya. Pada lapisan diafragma ini tersambung pula kumparan yang juga akan bergerak maju mundur.

Di dalam mic juga terdapat sebuah magnet permanen dimana ia akan menghasilkan medan magnet yang menembus kumparan. Saat kumparan tersebut maju mundur melalui medan magnet tersebut, arus listrik kemudian akan mengalir. Arus listrik tersebut yang kemudian memiliki karakteristik yang sama dengan inputan awalnya.



Gambar 2.6 Prinsip Kerja Microphone

Didalam perkembangannya, Microphone memiliki fungsi yang amat penting dimana ia menjadi komponen utama dalam proses masukan. Perangkat-perangkat telekomunikasi merupakan salah satu jenis peralatan elektronika yang memiliki kebutuhan yang amat tinggi pada microphone. Dalam perangkat-perangkat ini, microphone menjadi inputan suara.

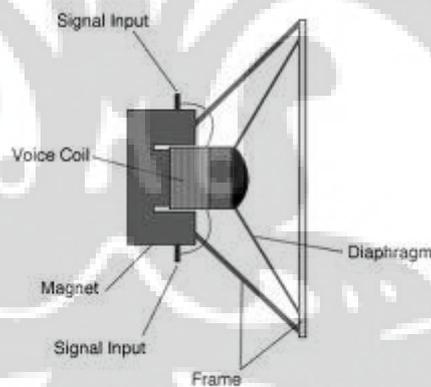
b. Speaker

Speaker atau penguat suara adalah sebuah perangkat transduser yang memiliki kemampuan untuk mengubah sinyal listrik menjadi frekuensi audio (suara) melalui komponen-komponen yang terpasang di dalamnya seperti lilitan, magnet, dan membran. Getaran membran inilah yang nantinya akan menghasilkan suara.

Cara kerja speaker pada prinsipnya merupakan kebalikan dari cara kerja mikrofon. Jika pada mikrofon ia merupakan perangkat input yang merubah suara menjadi sinyal elektrik, Speaker yang merupakan sebuah perangkat output bekerja dengan cara merubah sinyal elektrik menjadi suara dengan menggunakan getaran membrane.

Didalam sebuah speaker terdapat sebuah drivers yang dapat memproduksi gelombang suara dengan cara menggetarkan cone fleksibel (diafragma) secara cepat. Getaran yang dihasilkan tersebut merupakan hasil induksi dari magnet yang mengalir melalui lilitan. Cone yang ada tersebut pada umumnya terbuat dari kertas yang terhubung pada ujung suspension (surround).

Elektromagnet diposisikan pada bidang magnet yang konstan yang terbuat dari magnet permanen. Kedua magnet tersebut, yaitu elektromagnet dan magnet permanen kemudian akan berinteraksi satu sama lain seperti dua magnet yang berhubungan pada umumnya. Kutub positif pada elektromagnet tertarik oleh kutub negatif pada magnet permanen dan kutub negatif pada elektromagnet ditolak oleh kutub negatif magnet permanen.



Gambar 2.7 Skema Speaker

Pada saat orientasi kutub electromagnet saling tertukar maka akan tertukar pula arah dan gaya tarik-menariknya. Ketika ini terjadi, arus bolak-balik secara konstan membalikkan dorongan magnet antara voice coil (lilitan) dan magnet permanen. Secara keseluruhan, proses inilah yang mendorong coil kembali dan begitu seterusnya dengan cepat. Sewaktu coil bergerak, ia mendorong dan menarik speaker cone. Hal tersebut menggetarkan udara di depan speaker dan membentuk gelombang suara.

2.1.1.4 WiFi

Wi-Fi, adalah singkatan dari wireless fidelity, merupakan pengembangan dari istilah Hi-Fi, sebuah teknologi jaringan nirkabel yang digunakan di seluruh dunia. WiFi merupakan kependekan dari Wireless Fidelity, memiliki pengertian yaitu sekumpulan standar yang digunakan untuk Jaringan Lokal Nirkabel (Wireless Local Area Networks - WLAN) yang didasari pada spesifikasi IEEE 802.11. Standar terbaru dari spesifikasi 802.11a atau b, seperti 802.16 g, saat ini sedang dalam penyusunan, spesifikasi terbaru tersebut menawarkan banyak peningkatan mulai dari luas cakupan yang lebih jauh hingga kecepatan transfernya.

Didalam pengembangannya, Wi-Fi (Wireless Fidelity) memungkinkan untuk melakukan koneksi tanpa kabel seperti handphone dengan mempergunakan teknologi radio sehingga pemakainya dapat mentransfer data dengan cepat dan aman. Secara umum, kemampuan dari Wi-Fi ialah ia memungkinkan untuk pembuatan jaringan tanpa kabel antar divais. Hal ini memberikan banyak efek positif dimana ia dapat menjadi jaringan pertukaran data, internet, dan lain-lain.

WiFi bekerja menggunakan gelombang radio seperti pada telepon selular, televisi dan radio. Cara kerja WiFi bahkan memiliki banyak persamaan dengan komunikasi radio dua arah dimana pada setiap perangkat yang akan saling mengirim dan menerima data, data-data tersebut nantinya akan diterjemahkan ke dalam sinyal radio dan di transmisikan menggunakan antenna. Kemudian sinyal tersebut akan di dekode oleh perangkat penerima dan begitu seterusnya.

2.1.1.5 Bluetooth

Bluetooth adalah sebuah teknologi komunikasi wireless (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz unlicensed ISM (Industrial, Scientific and Medical) dengan menggunakan sebuah frequency hopping tranceiver yang mampu menyediakan layanan komunikasi data dan suara secara real-time antara host-host bluetooth dengan jarak jangkauan layanan yang terbatas.

Bluetooth sendiri dapat berupa card yang bentuk dan fungsinya hampir sama dengan card yang digunakan untuk wireless local area network (WLAN)

dimana menggunakan frekuensi radio standar IEEE 802.11, hanya saja pada bluetooth mempunyai jangkauan jarak layanan yang lebih pendek dan kemampuan transfer data yang lebih rendah.

da dasarnya bluetooth diciptakan bukan hanya menggantikan atau menghilangkan penggunaan kabel didalam melakukan pertukaran informasi, tetapi juga mampu menawarkan fitur yang baik untuk teknologi mobile wireless dengan biaya yang relatif rendah, konsumsi daya yang rendah, interoperability yang menjanjikan, mudah dalam pengoperasian dan mampu menyediakan layanan yang bermacam-macam.

2.1.2 Perangkat Lunak

Setelah mengetahui perangkat keras yang tersedia di dalam Android, kebutuhan perangkat lunak sangat tinggi agar dapat memanipulasi semua perangkat keras yang ada dan memperdayagunakannya sesuai kebutuhan. Adapun sistem yang terdapat di Android dan akan dimanfaatkan dalam penelitian ini adalah *text to speech*, *voice recognition* dan database menggunakan SQLite.

2.1.2.1 *Text to Speech*

Text to Speech merupakan sebuah sistem yang mengkonversi suatu teks, baik kata maupun suatu kalimat utuh menjadi suara. Hasil keluaran dari *Text to Speech* adalah suara yang mirip suara manusia, dapat diatur tempo dan aksent bahasanya.

Text to Speech sendiri menggunakan *speech synthesis* dalam pengolahan outputnya. *Speech synthesis* adalah sistem buatan manusia yang dapat menghasilkan suara yang hampir sama dengan suara manusia, Sistem ini banyak diimplementasikan di hardware maupun software.

Speech synthesis merupakan hasil kecerdasan buatan dari pembicaraan manusia. Komputer atau perangkat yang digunakan untuk tujuan ini disebut dengan *speech synthesizer*. Sebuah sistem *Text to Speech* merubah bahasa normal menjadi pembicaraan.

Sebuah sistem *Text to Speech* terdiri atas dua bagian utama yaitu *front-end* dan *back-end*. Pada bagian *front-end* terdapat dua hal yang akan terjadi.

Pertama mengubah teks mentah yang berisi kombinasi simbol, angka dan singkatan menjadi setara dengan tulisan per kata, Proses ini sering disebut dengan normalisasi teks, pra-pengolahan atau *tokenization*. Setelah proses *front-end* yang pertama lalu diteruskan dengan tahap menetapkan transkripsi fonetik untuk setiap kata dan membagi serta menandai ke prosodic unit seperti frase dan kalimat. Proses transkripsi fonetik untuk menetapkan kata-kata ini disebut teks-ke-fonem atau grammar-ke-fonem konversi. Fonetis transkripsi dan informasi ilmu persajakan bersama-sama membentuk representasi simbolik yang *linguistic output* dengan *front-end*. *Back-end* sering disebut sebagai synthesizer, karena mengubah representasi *linguistic* simbolik menjadi suara.

Untuk dapat menggunakan sistem ini dalam sebuah program Android cukup menambahkan dua kelas berikut:

```
import Android.speech.tts.TextToSpeech;
import Android.speech.tts.TextToSpeech.OnInitListener;
```

Dan pada saat onCreate dilakukan inisialisasi

```
tts = new TextToSpeech(this, this);
```

Setelah itu melakukan pengecekan apakah terdapat *resource* yang berisi data *text to speech* di Android,

```
Intent checkIntent = new Intent();
checkIntent.setAction(TextToSpeech.Engine.ACTION_CHECK_TTS_DATA);
startActivityForResult(checkIntent, MY_DATA_CHECK_CODE);
```

Setelah itu kita dapat langsung memakai fungsi ini dengan menggunakan *speak() method*, contoh:

```
tts.speak(suara, TextToSpeech.QUEUE_FLUSH, null);
```

2.1.2.2 Voice Recognition

Untuk dapat berkomunikasi antara robot dengan manusia diperlukan suatu tool yang mampu menangkap dan mengolah informasi dari manusia ke robot maupun sebaliknya. Banyak sekali perangkat yang dapat dijadikan sebagai media masukan informasi untuk robot, misalnya sensor jarak untuk mengetahui jarak robot dengan objek dan kamera untuk identifikasi pencitraan. Selain perangkat keras juga ada teknik yang dapat diupayakan untuk mengenali lingkungan sekitar, salah satunya adalah *voice recognition* atau pengenalan suara, dengan memanfaatkan *hardware* berupa *microphone* sebagai media masukan.

Voice recognition atau *speech recognition* adalah suatu pengembangan teknik dan sistem yang memungkinkan komputer atau perangkat pintar lainnya untuk menerima masukan berupa kata yang diucapkan. Teknologi ini memungkinkan suatu perangkat untuk mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat atau suatu *server*. Kata-kata yang diucapkan diubah bentuknya menjadi sinyal digital dengan cara mengubah gelombang suara menjadi sekumpulan kode tertentu yang kemudian digunakan untuk mengidentifikasi kata-kata tersebut. Hasil dari identifikasi kata yang diucapkan dapat ditampilkan dalam bentuk tulisan atau dapat dibaca oleh perangkat teknologi sebagai sebuah perintah untuk melakukan suatu pekerjaan, misalnya penekanan tombol pada telepon genggam yang dilakukan secara otomatis dengan komando suara.

Voice Recognition memanfaatkan sampel kata yang telah diucapkan dari pengguna. Sampel kata akan didigitalisasi, disimpan dalam komputer atau suatu perangkat khusus atau *server*, dan kemudian digunakan sebagai basis data dalam mencocokkan kata yang diucapkan selanjutnya. Sebagian besar *Voice Recognition* sifatnya masih tergantung kepada pengeras suara. Alat ini hanya dapat mengenal kata yang diucapkan dari satu atau dua orang saja dan hanya bisa mengenal kata-kata terpisah, yaitu kata-kata yang dalam penyampaiannya terdapat jeda antar kata. Hanya sebagian kecil dari peralatan yang menggunakan teknologi ini yang sifatnya tidak tergantung pada pengeras suara. Alat ini sudah dapat mengenal kata yang diucapkan oleh banyak orang dan juga dapat mengenal kata-kata kontinu, atau kata-kata yang dalam penyampainnya tidak terdapat jeda antar kata.

Untuk mengubah percakapan menjadi teks *on-screen* atau perintah tertentu, komputer melakukan beberapa langkah yang kompleks. Ketika berbicara, Anda mengeluarkan getaran di udara. Kemudian, *analog-to-digital converter* (ADC) yang ada di *soundcard* menerjemahkan gelombang analog ini menjadi data digital yang dapat dimengerti oleh komputer.

Untuk melakukan hal tersebut, sistem *Speech Recognition* melakukan *sampling* atau *digitizing* suara dengan cara mengambil ukuran yang paling pas dari gelombang. Sistem menyaring suara yang telah didigitalkan tersebut dan

membuang gangguan (*noise*), dan kadang-kadang memisahkannya ke dalam pita frekuensi yang berbeda. Frekuensi adalah panjang gelombang suara, yang terdengar oleh telinga manusia sebagai tinggi nada (*pitch*) yang berbeda.

Sistem ini juga menormalkan suara, atau mengaturnya ke dalam tingkat volume yang tetap, terkadang juga mendatarkan suara. Manusia tidak berbicara dalam kecepatan yang sama sehingga suara harus diatur dengan kecepatan yang sama dengan sampel-sampel template suara yang tersimpan dalam komputer atau *server*.



Gambar 2.8 Diagram Alir Voice Recognition

Langkah selanjutnya adalah memecah sinyal menjadi bagian-bagian kecil, dengan durasi seperseratus detik, atau bahkan seperseribu pada kasus bunyi-bunyi konsonan atau mati. Konsonan memberhentikan produksi suara dengan menghalangi aliran gelombang pada bidang vokal, seperti “p” atau “t”.

Program di komputer kemudian mencocokkan bagian-bagian kecil ini dengan fonem yang dikenal dalam bahasa tertentu. Fonem adalah elemen terkecil dalam sebuah bahasa, merepresentasikan suara yang kita hasilkan, dan merangkainya ke dalam bentuk ujaran yang memiliki makna.

Tahap berikutnya kelihatan sederhana, tapi pada dasarnya merupakan proses yang paling susah diselesaikan, sekaligus merupakan inti dari sebagian besar penelitian di bidang *Speech Recognition*. Komputer memeriksa fonem-fonem dalam konteks (hubungan) dengan fonem-fonem lain yang menyertainya.

Komputer menjalankan alur (*plot*) melalui sebuah model statistika yang kompleks, dan membandingkannya dengan koleksi kata, frase, dan kalimat yang telah dikenal. Program *Speech Recognition* selanjutnya menentukan apa yang mungkin dikatakan oleh pengguna, dan juga mengetikkannya sebagai teks atau mengeluarkannya sebagai perintah pada komputer.

Untuk dapat memanfaatkan sistem ini pada program diperlukan beberapa langkah fundamental dalam baris pemrograman, yaitu impor kelas-kelas yang berhubungan dengan voice recognition, yaitu:

```
import android.speech.RecognizerIntent;
```

Setelah itu melakukan pengecekan apakah pelayanan *voice recognition* dapat dilakukan di Android yang akan dicoba:

```
PackageManager pm = getPackageManager();
List<ResolveInfo> activities = pm.queryIntentActivities
new Intent (RecognizerIntent.ACTION_RECOGNIZE_SPEECH), 0);
```

Setelah dilakukan pengecekan lalu membuat sebuah method untuk pengenalan suara, yaitu:

```
private void startVoiceRecognitionActivity()
{
    Intent intent = new
    Intent (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);

    intent.putExtra (RecognizerIntent.EXTRA_LANGUAGE_MODEL, RecognizerIntent.LA
    NGUAGE_MODEL_FREE_FORM);
    startActivityForResult (intent, REQUEST_CODE);
}
}
```

Untuk pengambilan data hasil pengenalan dapat dilakukan dengan membuat method `onActivityResult`:

```
protected void onActivityResult (int requestCode, int resultCode, Intent
data)
{
    database.open ();
    Data = database.getAll ();
    if (requestCode == REQUEST_CODE && resultCode == RESULT_OK)
    {
        ArrayList<String> matches =
        data.getStringArrayListExtra (
            RecognizerIntent.EXTRA_RESULTS);
    }
}
```

2.1.2.3 Database SQLite

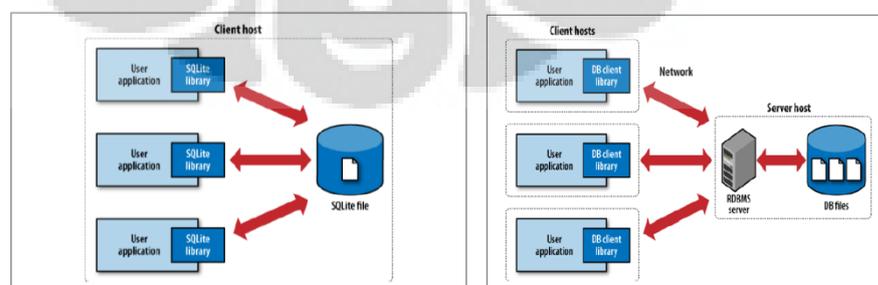
SQLite adalah *Relational Database Management Server* (RDBMS), salah satu basis data mandiri yang tersedia lintas sistem operasi dan dipergunakan di banyak arsitektur dan platform hardware, dari mulai telepon genggam hingga *desktop*. Kata “Lite” bukan berarti aplikasi RDBMS ini memiliki kemampuan yang sedikit / minim, tetapi mengacu pada keringanan/kemudahan dalam setup (instalasi), administrasi dan penggunaan. Kelebihannya dari jenis *database* ini adalah dari penggunaan SQLite adalah kemampuan portabilitasnya yang sangat

minim penyertaan *library*, cukup menyertakan API dari SQLite sendiri dan *wrapper* untuk masing-masing bahasa pemrograman.

RDBMS (*Relational Database Management Server*) merupakan sebuah program komputer (seperangkat program komputer) yang di desain untuk mengatur/memanajemen sebuah basis data sebagai sekumpulan data yang disimpan secara terstruktur, dan melakukan operasi-operasi atas data atas permintaan penggunaanya. Contoh penggunaan DBMS ada banyak sekali dan dalam berbagai bidang kerja, misalnya akuntansi, manajemen sumber daya manusia, dan lain sebagainya.

SQLite menggunakan Query Language yang mirip dengan RDBMS yang lain, sehingga pengguna yang terbiasa menggunakan MySQL, Ms. SQL, dan lain sebagainya tidak perlu bersusah payah untuk beradaptasi. Meskipun, pada beberapa bagian ada sedikit perbedaan dan ada juga beberapa perintah yang tidak disupport oleh SQLite.

Tidak seperti pada paradigma *client-server* umumnya, Inti SQLite bukanlah sebuah sistem yang mandiri yang berkomunikasi dengan sebuah program, melainkan sebagai bagian integral dari sebuah program secara keseluruhan. Sehingga protokol komunikasi utama yang digunakan adalah melalui pemanggilan API secara langsung melalui bahasa pemrograman. Mekanisme seperti ini tentunya membawa keuntungan karena dapat mereduksi *overhead*, *latency times*, dan secara keseluruhan lebih sederhana. Seluruh elemen basisdata (definisi data, tabel, indeks, dan data) disimpan sebagai sebuah file. Kesederhanaan dari sisi desain tersebut bisa diraih dengan cara mengunci keseluruhan file basis data pada saat sebuah transaksi dimulai.



Gambar 2.9 Perbedaan SQLite dengan Tipe Database RDBMS

Pustaka SQLite mengimplementasikan hampir seluruh elemen-elemen standar yang berlaku pada SQL-92, termasuk transaksi yang bersifat *atomic*, konsistensi basisdata, isolasi, dan durabilitas (dalam bahasa Inggris lebih sering disebut ACID), *trigger*, dan kueri-kueri yang kompleks. Tidak ada pengecekan tipe sehingga data bisa dientrikan dalam bentuk string untuk sebuah kolom bertipe integer. Beberapa kalangan melihat hal ini sebagai sebuah inovasi yang menambah nilai guna dari sebuah basisdata, utamanya ketika digunakan dalam bahasa pemrograman berbasis script (PHP, Perl), sementara kalangan lain melihat hal tersebut sebagai sebuah kekurangan.

Beberapa proses ataupun thread dapat berjalan secara bersamaan dan mengakses basisdata yang sama tanpa mengalami masalah. Hal ini disebabkan karena akses baca data dilakukan secara paralel. Sementara itu akses tulis data hanya bisa dilakukan jika tidak ada proses tulis lain yang sedang dilakukan; jika tidak, proses tulis tersebut akan gagal dan mengembalikan kode kesalahan (atau bisa juga secara otomatis akan mencobanya kembali sampai sejumlah nilai waktu yang ditentukan habis). Hanya saja ketika sebuah tabel temporer dibuat, mekanisme penguncian pada proses multithread akan menyebabkan masalah. Update yang terkini (versi 3.3.4) dikatakan telah memperbaiki masalah ini.

Sebuah program yang mandiri dinamakan *sqlite* disediakan dan bisa digunakan untuk mengeksekusi kueri dan manajemen *file-file* basisdata SQLite. Program tersebut juga merupakan contoh implementasi penulisan aplikasi yang menggunakan pustaka SQLite.

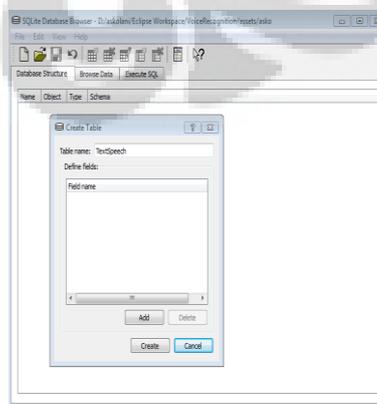
Adapun fitur-fitur yang ada pada SQLite adalah:

- a. *Serverless*, SQLite tidak memerlukan proses pada *server* atau sistem untuk menjalankannya, melainkan hanya sebuah file yang diakses oleh *library* SQLite.
- b. *Cross Platform*, semua instan *database* berada dalam sebuah file yang cross-platform, tidak memerlukan administrasi.
- c. *Zero Configuration*, Tidak ada *server* berarti tidak perlu setup, membuat sebuah *database* instan adalah semudah anda membuat file biasa.

- d. *Small Runtime Footprint*, untuk membangun *database* SQLite hanya membutuhkan kurang dari satu *megabyte library* (kode program) dan hanya membutuhkan beberapa *megabyte memory*, bahkan dengan beberapa *adjustment* baik ukuran *library* maupun *memory* dapat diperkecil.
- e. *Self-Contained*, sebuah *library* mengandung keseluruhan dari sistem *database*, yang langsung terintegrasi pada sebuah aplikasi program
- f. *Ransactional*, SQLite transaction memperbolehkan aksi penyimpanan melalui beberapa proses *thread*.
- g. Full Featured, SQLite mensupport hampir sebagai besar standar SQL92 (SQL2)
- h. Highly Reliable, Tim pengembang SQLite mengembangkan melalui kode program yang sangat serius serta telah melewati proses *testing*.

Eclipse tidak mendukung pembuatan *database* SQLite di dalam fitur-fiturnya. Untuk itu diperlukan software pembantu yang dapat membuat sebuah *database* kosong yang nantinya *database* tersebut dapat digunakan oleh Android untuk dimodifikasi atau diakses oleh suatu program Android tertentu.

Software pendukung untuk pembuatan *database* SQLite kosong yang penulis gunakan adalah SQLite *Database* Browser. SQLite *Database* Browser merupakan aplikasi open source yang dapat memvisualisasikan struktur *database* dan datanya. Tampilan GUI yang user friendly membuat pengguna tidak harus mengerti tentang bahasa SQL. Selain itu mudah dalam memodifikasi data yang sangat membantu programmer. Software ini tersedia di platform Windows, Linux dan Mac OS X. Berikut adalah tampilan muka dari SQLite *Database* Browser:



Gambar 2.10 Tampilan Muka SQLite *Database* Browser

Pembuatan *database* dilakukan dengan membuat file SQLite baru lalu di dalam file tersebut diciptakan tabel dengan nama yang nantinya diakses menggunakan method yang ada di Eclipse untuk Android.

Berikut adalah langkah-langkah pembuatan database SQLite pada Eclipse:

- Pembuatan Kelas DataHelper

Sebuah kelas yang berfungsi untuk mengatur pembuatan database dan *update*-nya.

- Pembuatan Kelas DataWrite

- Deklarasi variabel untuk field data/kolom, contoh:

```
public static final String KEY_ROWID = "_id";
public static final String KEY_TRIGGER = "trigger";
public static final String KEY_OUTPUT = "output";
public static final String KEY_PROTOKOL = "protokol";
public static final String KEY_FEEDBACK = "feedback";
```

- Deklarasi nama database, tabel dan versi, contoh:

```
private static final String DATABASE_NAME = "textspeechmain";
private static final String DATABASE_TABLE = "textspeech1";
private static final int DATABASE_VERSION = 1;
```

- Membuat adapter database

```
public DatabaseWrite(Context ctx)
{
    this.context = ctx;
}
```

- Inserting data baru untuk database, contoh:

```
public Cursor getById(long rowId) throws SQLException
{
    Cursor mCursor =
        db.query(true, DATABASE_TABLE, new String[] {
            KEY_ROWID,
            KEY_INPUT,
            KEY_OUTPUT,
            KEY_KAMERA,
            KEY_KECEPATAN,
            KEY_JARAK,
            KEY_LENGAN},
            KEY_ROWID + "=" + rowId,
            null,
            null,
            null,
            null,
            null);
    if (mCursor != null) {
        mCursor.moveToFirst();
    }
    return mCursor;
}
```

- Membuat method untuk membuka dan menutup database

```
public DatabaseWrite open() throws SQLException
{
    DBHelper = new DatabaseHelper(context);
    db = DBHelper.getWritableDatabase();
    return this;
}

public void close()
{
    DBHelper.close();
}
```

- Menghapus Data

```
public boolean deleteById(long rowId)
{
    return db.delete(DATABASE_TABLE, KEY_ROWID +
        "=" + rowId, null) > 0;
}
```

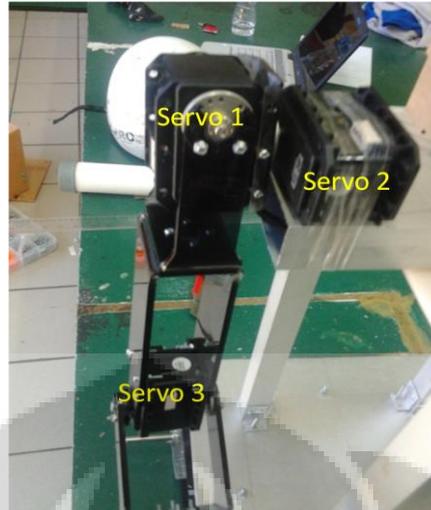
2.2 Sistem Luar

Sistem luar adalah sistem-sistem yang menjadi penunjang sehingga robot dapat menjalankan fungsinya dengan baik dari luar. Dari luar ini maksudnya adalah luar Android, adapun robot ini mempunyai beberapa sistem di setiap komponennya, yaitu sistem pengolahan citra menggunakan kamera, sistem pengolahan jarak menggunakan SRF08, sistem pergerakan tangan robot serta sistem pergerakan *mobile robot*.

2.2.1 Sistem Lengan

Lengan robot penerima tamu terdiri atas tiga servo Dynamixel dan tiga motor DC untuk tiap jari. Lengan robot didesain layaknya seperti lengan manusia. Pengendalian pergerakan lengan dan jari robot menggunakan strategi *compliance control*.

Strategi *compliance control* pada dasarnya dibagi menjadi dua group, yaitu *active compliance* dan *passive compliance*. *Passive compliance* menggunakan divais mekanik yang fleksibel yang ditempatkan pada ujung robot. Berhubungan dengan tugas yang di susun, divais ini mengatur posisi obyek yang dimanipulasi ketika sebuah gaya interaksi diterapkan. Sedangkan *active compliance* adalah suatu metode yang didasarkan pada divais aktif, seperti *Force/Torque (F/T) sensors* untuk mengukur interaksi dengan robot. Metode *compliance control* yang banyak digunakan untuk sekarang ini adalah *active compliance*.

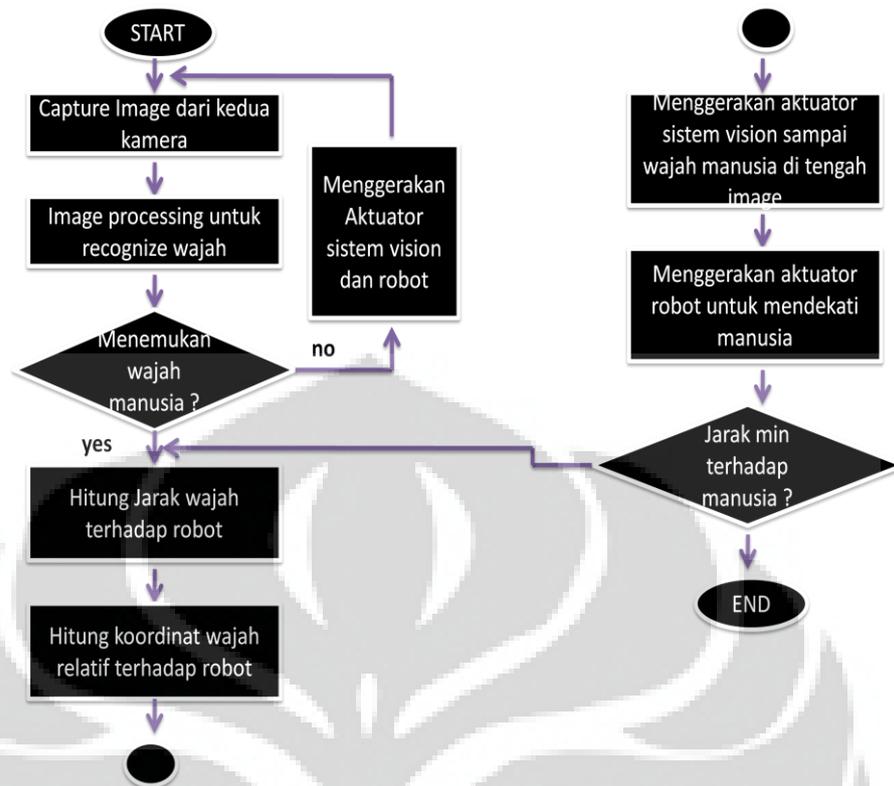


Gambar 2.11 Lengan Robot

Compliance control merupakan kombinasi antara *motion control* dan *force control*. Dengan menerapkan *Resolved Motion Rate Control (RMRC)*, tangan robot dapat bergerak sesuai instruksi yang di perintahkan dan setiap waktu, posisi, kecepatan, serta gaya aktual pada setiap joint dapat terekam melalui sensor. Dengan menggunakan *Neuton-Euler equation* tentang dinamika maka bisa didapatkan persamaan gaya (*force*) ditiap titik gerakan. Kinematik mentransformasikan koordinat kartesius menjadi koordinat sudut sedangkan inverse kinematik sebaliknya. Dengan menerapkan keduanya maka dengan lebih mudah *motion control* dapat diterapkan.

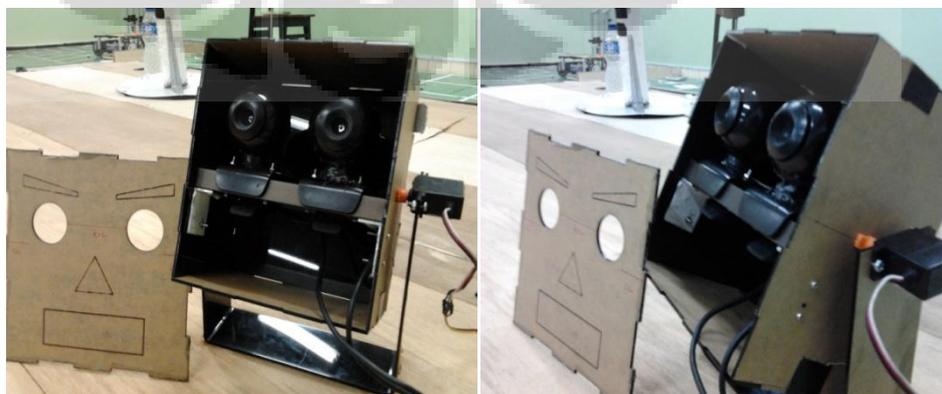
2.2.2 Kamera

Kamera yang digunakan robot ini berjumlah dua buah yaitu Logitech E1000 yang mempunyai resolusi sebesar 1.3 MP. Pengolahan hasil capturing image yang dilakukan oleh kamera diproses oleh BeagleBoard. BeagleBoard mengolah data dari dua kamera tersebut untuk nantinya menghasilkan suatu kordinat suatu objek, berikut diagram alir dari sistem stereo vision yang dibangun:



Gambar 2.12 Diagram Alir Sistem Stereo Vision

Sistem dimulai dengan melakukan *capture* melalui webcam kiri dan kanan. Proses utama yang dilakukan adalah menjalankan fungsi *face recognition* sampai sistem kepala RPT menemukan wajah manusia. Jika belum menemukan, maka aktuator sistem vision (dengan batasan arah putar tertentu) yang berada di kepala RPT akan bergerak memutar secara horizontal dan vertikal untuk memperluas pandangan dari kamera. Jika pergerakan dari aktuator juga tidak menemukan wajah manusia, maka aktuator robot (motor base) akan mulai bergerak untuk lebih memperluas pandangan kamera.



Gambar 2.13 Implementasi Sistem Stereo Vision

Setelah menemukan wajah manusia, maka BeagleBoard akan terus melakukan image processing berupa face tracking dan memberi perintah kepada slave untuk menjalankan actuator robot guna untuk mendekati ke arah manusia berada dalam waktu bersamaan. Prosedur untuk menggerakkan actuator baik pada sistem vision maupun pada robot secara keseluruhan berdasarkan data informasi dari jarak manusia terhadap robot dan perbedaan koordinat yang didapat dari image processing dengan koordinat pusat kamera.

2.2.3 Sensor SRF08

Sensor jenis ini dikategorikan dalam sensor jarak. Dalam pendeteksian jaraknya sensor ini menggunakan gelombang ultrasonik yang dihasilkan oleh sensor ini. Sensor jarak ini memiliki dua buah bagian, yaitu *transmitter* pada sisi kirim dan *receiver* pada sisi terima. Prinsip kerjanya adalah gelombang ultrasonik akan dipancarkan oleh *transmitter* selama periode waktu tertentu yang kemudian gelombang ultrasonik tersebut akan diterima oleh *receiver*.

Data keluaran dari sensor ini berupa logika 0 atau 1 yang kemudian berdasarkan *datasheet* SRF-08 dapat dihitung besarnya jarak benda terhadap sensor ini. Perhitungan jaraknya adalah dengan mengalikan kecepatan gelombang ultrasonik dengan waktu tempuh dan kemudian dibagi 2 akibat adanya waktu gelombang ultrasonik dipantulkan ke *receiver* yang besarnya sama dengan waktu gelombang ultrasonik menyentuh benda.



Gambar 2.14 Prinsip Kerja SRF08

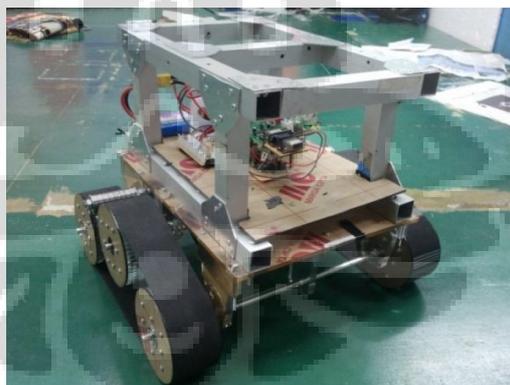
Sensor ini memiliki performansi yang cukup baik, yaitu mampu mengukur jarak sepanjang 3 cm sampai 6 m dan sensor ini sudah cukup banyak digunakan dalam berbagai aplikasi robotika. Komunikasi sensor ini dengan mikrokontroller

dilakukan melalui I2C (*Inter Integrated Circuit*), yaitu suatu standar protokol sistem komunikasi data serial yang dikembangkan oleh Philips dengan mengirimkan data secara serial melalui sebuah jalur data dua buah arah I2C. Karena menggunakan jalur data I2C, maka hanya memerlukan dua buah pin saja untuk berkomunikasi, yaitu pin untuk *data* (SDA) dan pin untuk *signal clock* (SCL). Sistem komunikasi ini cukup terkenal karena penggunaannya yang mudah dan mampu menghemat penggunaan pin-pin pada mikrokontroler.

Apabila dari hasil image processing didapat jarak manusia relative terhadap robot telah sampai pada jarak minimum, maka robot akan berhenti dan akan melakukan gerakan selanjutnya sebagaimana layaknya penerima tamu yaitu bersalaman, *say hello*, dan lain sebagainya. Setelah proses diatas selesai, maka robot kembali ke tempat semula untuk melakukan gerakan selanjutnya.

2.2.4 Mobile robot

Pergerakan robot dilakukan oleh dua motor yang telah diberi algoritma tertentu dalam pengendaliannya. Merupakan pengendalian loop tertutup dengan memanfaatkan encoder sebagai masukan umpan balik pergerakan tiap motor. Kedua motor ini diatur oleh satu master yaitu ATmega2560 yang nantinya juga sebagai tempat berkomunikasi dengan Android.



Gambar 2.15 Mobile Robot

2.3 Komunikasi

Komunikasi antara mikrokontroler dengan Android dilakukan melalui media Bluetooth. Untuk dapat melakukan penyambungan Bluetooth antara mikrokontroler dengan Android perlu melewati dan melaksanakan protokol yang

sudah ditentukan. Protokol disini adalah protocol dari sisi *server* atau master dan dari sisi client atau slave.

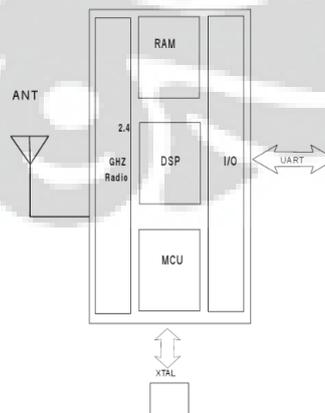
Untuk sisi mikrokontroller kita setting menjadi tipe slave, karena memudahkan dalam pemrograman di mikrokontroller, Bluetooth serial yang terpasang akan menjadi device serial biasa yang mana datanya dapat kita ambil kapan saja.

Jika Android sudah terintegrasi dengan Bluetooth, berbeda dengan mikrokontroller ATMega2560, karena diperlukan sebuah device tambahan agar mikrokontroller dapat menangkap data yang dikirimkan oleh Android atau mengirimkan data ke Android. Bluetooth adapter merupakan solusi dari masalah ini.

Bluetooth serial merupakan salah satu Bluetooth adapter yang dapat dipasang pada mikrokontroller. Bluetooth Serial merupakan perangkat elektrik yang dapat menjadi Bluetooth adapter dan menghasilkan keluaran berupa data serial. Bekerja dengan



Gambar 2.16 Bluetooth Serial



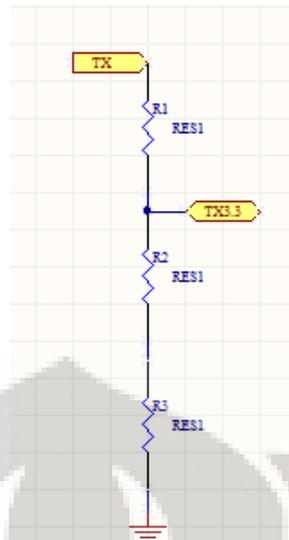
Gambar 2.17 Blok Diagram Bluetooth Serial

Tabel 2.2 Konfigurasi Pin Bluetooth Serial

No. Pin	Nama	Tipe	Fungsi
1	UART-TX	CMOS Output	UART Data Output
2	UART-RX	CMOS Input	UART Data Input
3	UART-CTS	CMOS Input	UART Clear to Send Active low
4	UART-RTS	CMOS Output	UART Request to Send Active Low
11	RESETB	CMOS Input	Reset ketika kondisi low
12	3.3V	POWER	+3.3V Supply
13-14, 21-22	GND	Ground	
5-10, 15-20, 23-34	NC	-	-

Adapun berdasarkan hasil perancangan dan pengujian yang telah dilakukan sebelumnya didapatkan sebuah sistem minimum Bluetooth serial yang layak untuk dipakai dalam rancang bangun *human interaction* robot penerima tamu berbasis Android ini.

Berdasarkan tabel spesifikasi Bluetooth serial yang telah dijelaskan bahwa Bluetooth serial mempunyai tegangan kerja pada 3.3 V, sedangkan mikrokontroler bekerja pada tegangan 4.5 V hingga 5.5 V, untuk itu perlu dilakukan regulasi nilai tegangan antara mikrokontroler dengan Bluetooth serial agar kedua belah pihak dapat bekerja dengan baik sehingga tidak ada kesalahan dalam pengiriman data. Karena nilai 3.3 V masih dianggap berlogika high atau 1 oleh mikrokontroler jadi tidak ada masalah untuk keluaran TX dari Bluetooth serial ke RX mikrokontroler. Sedangkan daerah tegangan kerja Bluetooth serial hanya 3.3 V dan apabila lebih dari itu perangkat ini tidak akan berfungsi dengan baik bahkan terancam rusak. Untuk menangani masalah itu penulis memanfaatkan sistem *voltage divider* menggunakan tiga buah resistor bernilai sama yang dihubungkan secara seri antara TX mikrokontroler ke ground, dan pin RX Bluetooth serial hanya menerima 2/3 dari total keluaran TX mikrokontroler dengan cara menyambungkan RX Bluetooth serial di node setelah satu dari tiga resistor yang dipasangkan secara seri ke ground tersebut. Untuk lebih jelasnya dapat dilihat pada gambar 2.18.



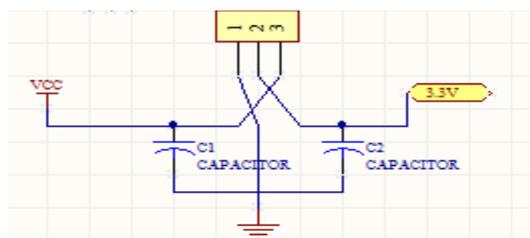
Gambar 2.18 *Voltage Divider* untuk Mendapatkan Tegangan 3.3 V

Karena keluaran dari mikrokontroler selalu konstan yaitu 4.8 – 5.1 V maka penulis yakin bahwa hasil *voltage divider* akan tidak jauh dari 3.3 V dan masih dalam toleransi.

$$\begin{aligned} V_{\text{output}} &= \frac{2}{3} \cdot V_{\text{input}} \\ &= \frac{2}{3} \cdot (4.8 \sim 5.1) \text{ V} \\ &= 3.2 \sim 3.4 \text{ V} \end{aligned}$$

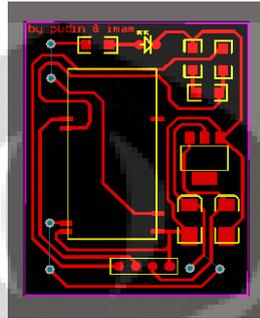
Untuk power supply Bluetooth serial penulis memanfaatkan regulator AMS1117, sebuah IC yang meregulasi tegangan dari 5 V menjadi keluaran 3.3 V. IC ini merupakan IC tipe SMD (*Surface Mounting Device*) yang berukuran sangat kecil dan mudah dalam pemasangan.

Regulator AMS1117 juga memiliki sistem minimum sendiri yang membuat IC ini dapat mengeluarkan nilai tegangan yang konstan dan *ripple* yang rendah. Pemasangan dua buah kapasitor polar bernilai 100 μ F akan sangat membantu dalam memotong ripple yang ada.



Gambar 2.19 Skematik Sistem Minimum AMS1117

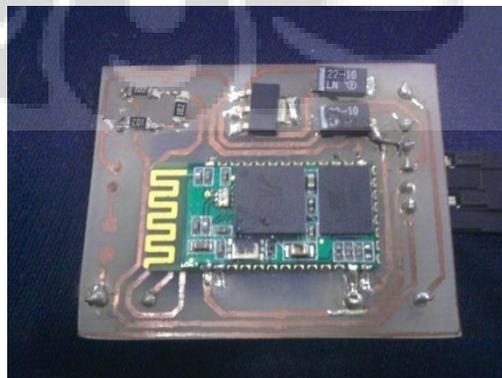
Kedua hal yang telah dijelaskan di atas dijadikan dalam satu PCB yang semua komponennya berjenis SMD lalu akan dikeluarkan 4 pin utama yang akan terhubung ke mikrokontroller yaitu VCC, Ground, RX, dan TX. Agar lebih memperoleh gambaran berikut layout PCB yang sudah jadi:



Gambar 2.20 Layout PCB Bluetooth Serial

Sisi Android diatur menjadi master atau yang membuka socket Bluetooth antar Bluetooth. Android memiliki Bluetooth API yang memperkenankan pembuatan aplikasi pengiriman data menggunakan media Bluetooth. Socket sendiri adalah jalur yang dibentuk oleh master untuk menghubungkan master dengan slave melalui komunikasi Bluetooth.

Perancangan komunikasi ini dilakukan dengan menggunakan software pembantu yaitu Eclipse. Di sini penulis menggunakan Eclipse Galileo yang sudah terintegrasi dengan Android SDK. Penulis memanfaatkan IDE ini karena open source ditambah dengan proses debugging untuk tiap melakukan pemrograman sangat user friendly.



Gambar 2.21 PCB Bluetooth Serial

Ada beberapa langkah penting yang harus dilakukan agar dapat membuat sebuah aplikasi yang menggunakan fitur Bluetooth di Android, yaitu:

- Deklarasi Permissions

Pada AndroidManifest.xml menulis dua deklarasi Bluetooth permissions, yaitu BLUETOOTH dan BLUETOOTH.ADMIN. BLUETOOTH permission adalah deklarasi yang mengizinkan Android untuk menerima permintaan penyambungan, memberikan permintaan penyambungan serta transfer data. Sedangkan BLUETOOTH.ADMIN permission adalah deklarasi khusus yang mengizinkan untuk memanipulasi discoverable Bluetooth dan aktivasi Bluetooth (hidup dan mati).

```
<uses-permission Android:name="Android.permission.BLUETOOTH" />
<uses-permission Android:name="Android.permission.BLUETOOTH_ADMIN" />
```

- Impor Kelas

Pada tahap awal yaitu importing basic class untuk komunikasi Bluetooth yang semua itu ada di Android.Bluetooth package.

```
import java.io.InputStream;
import java.io.OutputStream;
```

Merupakan kelas dasar untuk segala *input* dan *output* streams, yang berarti melakukan pembacaan data dalam bentuk byte untuk semua keluaran dan masukan data pada Android.

```
import java.util.UUID;
```

Kelas ini merepresentasikan 128-bit UUID (Universally Unique Identifier). Digunakan untuk mengidentifikasi UUID setiap device yang akan melakukan hubungan dengan Android.

```
import Android.Bluetooth.BluetoothDevice;
import Android.Bluetooth.BluetoothSocket;
import Android.os.Handler;
```

Kelas BluetoothDevice dan BluetoothSocket digunakan untuk pengaturan koneksi antar Bluetooth. Dan kelas Handler mengizinkan pengiriman pesan atau informasi atau objek yang dapat dipakai oleh thread lain.

- Menampilkan Bluetooth Device yang Sudah terpasang

```
Set<BluetoothDevice> pairedDevices =
mBluetoothAdapter.getBondedDevices();
for (BluetoothDevice device : pairedDevices) {
mArrayAdapter.add(device.getName()+"\n"+device.getAddress());
```

- Method untuk Mengirimkan dan Menerima Data

```
public void write(byte[] bytes) {
    try {
        outputStream.write(bytes);
    } catch (IOException e) { }
}

public void read(byte[] buffer) {
    try {
        inputStream.read(buffer);
    } catch (IOException e) {
    }
}
```

2.4 Eclipse

Selain dalam bentuk fisik, ada pula aspek pendukung sehingga penelitian ini dapat berlangsung, yaitu software pendukung. Disini akan dijelaskan secara umum tentang software yang mendukung pembuatan program dan proses debugging program.

Eclipse merupakan IDE (Integrated Development Environment) multilanguage, karena selain dapat menggunakan bahasa Java, Eclipse juga mendukung bahasa seperti C++, Cobol, Fortran. Dapat dikatakan bahwa Eclipse merupakan sebuah single environment untuk pengembangan aplikasi yang berjalan di banyak platform karena target dari Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X. Selain itu Eclipse adalah aplikasi multirole karena dapat digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, tes perangkat lunak, pengembangan web dan lain sebagainya.

Integrated Development Environment (IDE) adalah sebuah aplikasi yang menjadikan sebuah workspace atau ruang kerja untuk programmer agar dapat membuat aplikasi dengan mudah dan cepat. Dengan kemampuan drag and drop dan kelebihan lainnya membuat IDE menjadi pilihan yang banyak dipakai.

Eclipse merupakan salah satu pilihan IDE yang banyak dipakai dan menjadi favorit karena gratis dan open source, setiap orang dapat mengakses kode pemrograman pada aplikasi atau perangkat lunak ini. Kelebihan lain dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan plug-in.

Pada awal pengembangannya oleh IBM Eclipse diciptakan untuk menggantikan perangkat lunak IBM Visual Age Java 4.0. Pada awal

peluncurannya pada tanggal 5 November 2001 program ini mendapat investasi hingga US \$ 40 untuk pengembangannya. Semenjak itu Eclipse Foundation mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya.

Sejak versi 3.0, Eclipse pada dasarnya merupakan sebuah *kernel*, yang mengangkat *plug-in*. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari *plug-in* yang sudah diinstal. Ini merupakan basis dari Eclipse yang dinamakan *Rich Client Platform* (RCP). Berikut ini adalah komponen yang membentuk RCP:

- *Core platform*
- OSGi
- SWT (*Standard Widget Toolkit*)
- JFace

Secara standar Eclipse selalu dilengkapi dengan JDT (*Java Development Tools*), *plug-in* yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (*Plug-in Development Environment*) untuk mengembangkan *plug-in* baru. Eclipse beserta *plug-in*-nya diimplementasikan dalam bahasa pemrograman Java.

Konsep Eclipse adalah IDE yang terbuka (*open*), mudah diperluas (*extensible*) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, Eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal *plug-in* yang dibutuhkan. Apabila ingin mengembangkan program C/C++ terdapat *plug-in* CDT (*C/C++ Development Tools*). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, *plug-in* UML2 tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat *plug-in* sesuai dengan keinginannya.

Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama *Simultaneous Release*. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan dari sistem ini adalah untuk menyediakan distribusi Eclipse dengan fitur-fitur dan versi yang terstandarisasi.

Hal ini juga dimaksudkan untuk mempermudah deployment dan maintenance untuk sistem enterprise, serta untuk kenyamanan. Peluncuran simultan dijadwalkan pada bulan Juni setiap tahunnya.



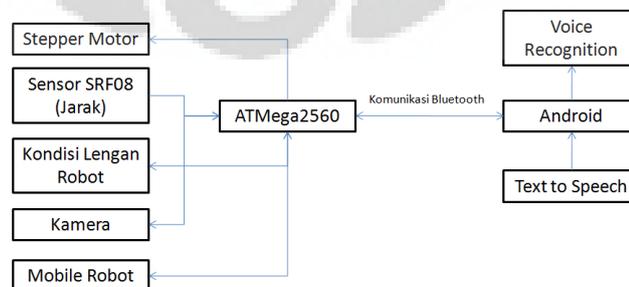
BAB 3

PERANCANGAN HUMAN INTERACTION

Sebagai injinir harus mempunyai perencanaan yang matang dalam segala hal, terutama masalah teknis. Dalam hal pembuatan robot penerima tamu ini diperlukan adanya pembuatan perancangan terlebih dahulu sebelum nantinya akan diimplementasikan ke robot. Pada bab kali ini akan dijabarkan mengenai perancangan *human interaction* yang dibutuhkan oleh robot penerima tamu untuk dapat berfungsi sebagaimana mestinya dari sisi Android.

Perancangan terdiri atas tiga hal yaitu perancangan *signaling* sistem luar, perancangan interaksi manusia ke robot dan perancangan interaksi robot ke manusia. Pada perancangan *signaling* sistem luar akan dirancang suatu sistem yang dapat menangani pengolahan data dari sisi Android terhadap masukan data serial kiriman mikrokontroller mengenai data pengolahan kamera, kecepatan robot, jarak robot dengan objek dan kondisi aktifitas lengan. Pada perancangan interaksi manusia ke robot akan dirancang suatu sistem yang dapat menangani masukan dari manusia, dalam hal ini lebih dikhususkan dari sisi *voice recognition* sebagai jalur masukan.

Masukan data sensor yang terpasang di robot penerima tamu dan hasil pengolahan *voice recognition* dijadikan sebagai bahan untuk diolah oleh Android dan mengeluarkan output berupa suara yang menggunakan sistem *text to speech* dan juga perintah-perintah menggunakan protokol untuk pergerakan komponen-komponen robot seperti tangan, sistem gerak, putaran badan menggunakan motor stepper dan sejenisnya.



Gambar 3.1 Diagram Alir Robot Penerima Tamu

3.1 Perancangan Signaling Sistem Luar

Pada perancangan signaling sistem luar akan dirancang dan dibuat suatu sistem yang dapat menangani segala jenis masukan data serial hasil kiriman dari mikrokontroller mengenai data pengolahan kamera, kondisi kecepatan robot, jarak robot dengan objek dan kondisi aktifitas lengan.

Program Android yang akan dibuat memanfaatkan data keluaran dari sensor-sensor yang dipakai pada robot sebagai pemicu. Pemanfaatan ini dimaksudkan agar antara aktifitas yang dilakukan oleh Android sinkron dengan kondisi keseluruhan pada robot. Sensor-sensor yang terpasang di robot penerima tamu antara lain adalah dua buah kamera, empat sensor jarak SRF08, *mobile robot* menggunakan encoder. Selain itu juga ada aktifitas dari robot yang harus sinkron dengan Android yaitu pergerakan *mobile robot* dan pergerakan tangan. Semua data sensor dan aktifitas robot tersebut akan diterima oleh mikrokontroller yang nantinya akan dikirimkan ke Android melalui jalur komunikasi Bluetooth.

Oleh karena itu perlu dibuat sebuah protokol pengiriman data antara mikrokontroller sebagai tempat pemberhentian pertama dari semua data sensor dan aktifitas robot sebelum dikirim ke Android. Dengan adanya protokol ini diharapkan dari sisi mikrokontroller maupun Android dapat mengetahui data sensor mana yang sedang dikirimkan atau data mana yang ingin diperoleh, dan adanya penyesuaian antara pergerakan robot dengan aktifitas Android.

3.1.1 Protokol Pengiriman Data

Adapun protocol pengiriman ini menggunakan tipe data dalam bentuk byte. Panjang data sebesar 4 byte, dimana masing-masing byte merupakan header, identitas, data dan checksum.

Tabel 3.1 Paket Pengiriman Data

HDR	IDE	DATA	CKS
-----	-----	------	-----

a. HDR

HDR merupakan header dari paket data. Pada saat mikrokontroller mulai mengirimkan data, nilai ini akan secara otomatis dibangkitkan oleh mikrokontroller. Begitu pula saat mikrokontroller menerima data maka akan

secara otomatis HDR akan diekstraksi atau dipisahkan dari paket pengiriman data. HDR mengindikasikan dimulainya pengiriman data. Nilai header sebesar 0xFF.

b. IDE

IDE merupakan identitas pengiriman data. IDE akan merepresentasikan data sensor mana yang akan dikirimkan oleh mikrokontroler ke Android maupun dari Android ke mikrokontroler. Berikut tabel identitas pengiriman data:

Tabel 3.2 Identitas Data

Nilai	Identitas	Fungsi
0xA0	Kamera	Mengetahui data dari kamera
0xB0	Kecepatan	Mengetahui besar kecepatan pergerakan robot
0xC0	Jarak	Mengetahui jarak robot terhadap objek terdekat
0xD0	Lengan	Mengetahui kondisi lengan robot
0xE0	Suara	Mengetahui data hasil Voice recognition
0xF0	Sensor	Mengetahui data accelerometer milik Android

c. DATA

Merupakan nilai data yang dikirim atau diterima. Besarnya adalah 1 byte untuk setiap pengiriman data.

d. CKS

CKS adalah checksum dari paket data sebagai indikator validitas data. Pada saat pengiriman nilai ini akan dibangkitkan secara otomatis bergantung jumlah data yang akan dikirim.

3.1.2 Sistem Pencitraan

Pada sistem pencitraan menggunakan dua buah kamera webcam dengan pengolahan gambar menggunakan BeagleBoard. Adapun keluaran yang diharapkan untuk dikirimkan ke android melalui mikrontroller dari pengolahan citra menggunakan BeagleBoard adalah tingkat keramaian yang ada di depan robot. Datanya hanya untuk empat kondisi ramai, sepi, mengenali dan tidak mengenali.

Tabel 3.3 Data Pencitraan

Paket Data	Kondisi
0xFF-0xA0-0x00-0xA0	Kamera mendeteksi keramaian
0xFF-0xA0-0x01-0xA1	Kamera tidak mendeteksi keramaian
0xFF-0xA0-0x02-0xA2	Kamera mengenali objek manusia
0xFF-0xA0-0x03-0xA3	Kamera tidak mengenali objek manusia

3.1.3 Sistem Kecepatan

Pada sistem kecepatan untuk *mobile robot* dapat diketahui dengan menggunakan encoder yang sudah terpasang di masing-masing *shaft* pada kedua sisi roda. Dengan menggunakan suatu algoritma tertentu akan didapatkan nilai kecepatan dari *mobile robot* pada saat bergerak. Hasil perhitungan algoritma tersebut digunakan untuk menentukan respon apa yang harus diberikan oleh android untuk interaksi ke manusia.

Pengiriman data kecepatan dari *mobile robot* dilakukan melalui komunikasi bluetooth serial. Data yang dikirim berupa data byte yang nantinya akan diekstrak oleh Android sebagai informasi mengenai data kecepatan robot saat pembacaan. Data yang akan dikirimkan dari mikrokontroler ke Android berbentuk byte dengan panjang 1 byte. Merupakan data kecepatan pergerakan robot.

Tabel 3.4 Data Kecepatan

Paket Data	Kondisi
0xFF-0xB0-0x00-0xB0	Kecepatan 0%
0xFF-0xB0-0x01-0xB1	Kecepatan 60%
0xFF-0xB0-0x02-0xB2	Kecepatan 100%

3.1.4 Sensor Jarak

Sensor jarak yang digunakan adalah SRF08 yang mempunyai daerah kerja antara 3 centimeter hingga 6 meter. Pengiriman data ke mikrokontroller menggunakan jalur komunikasi I2C. Jalur komunikasi I2C membuat SRF08 dapat dipasang dengan jumlah hingga 18 buah ke dalam satu mikrokontroller.

Data yang diterima oleh mikrokontroller dari SRF08 lalu akan dikonversi dan dilakukan perhitungan hingga menghasilkan suatu informasi jarak dengan satuan tertentu. Informasi atau data jarak tersebut akan dikirim ke Android menggunakan jalur komunikasi Bluetooth. Sebelumnya data tersebut dikonversi terlebih dahulu ke dalam bentuk byte dan dikirimkan dengan menggunakan header tertentu sebagai bentuk identifikasi sebagai data sensor jarak.

Data yang telah diterima oleh Android akan dikenali dengan header yang menyertakannya. Data tersebut lalu diekstrak dan digunakan untuk membuat suatu komparasi dengan database yang sudah dibuat yang nantinya menjadi *trigger* eksekusi perintah selanjutnya.

Tabel 3.5 Data Jarak

Paket Data	Kondisi
0xFF-0xC0-0x00-0xC0	Dekat
0xFF-0x C0-0x01-0xC1	Normal
0xFF-0x C0-0x02-0xC2	Jauh

3.1.5 Sistem Lengan

Tabel 3.6 Data Status Lengan

Paket Data	Kondisi	
	Lengan Kanan	Lengan Kiri
0xFF-0xD0-0x00-0xD0	180°	0°
0xFF-0xD0-0x01-0xD1	0°	180°
0xFF-0xD0-0x02-0xD2	180°	180°
0xFF-0xD0-0x03-0xD3	90°	0°
0xFF-0xD0-0x04-0xD4	0°	90°
0xFF-0xD0-0x05-0xD5	90°	90°
0xFF-0xD0-0x06-0xD6	0°	0°

3.2 Perancangan Interaksi Manusia ke Robot

Perancangan interaksi manusia ke robot adalah perancangan yang dilakukan agar manusia dapat mempengaruhi cara kerja robot. Diharapkan dari masukan yang manusia berikan ke robot lalu robot akan langsung merespon sebagaimana mestinya manusia dengan menggunakan sistem yang ada di robot tersebut. Perancangan ini lebih banyak melibatkan sistem *Voice Recognition* dan penggunaan sensor jarak serta kamera sebagai masukannya.

Voice Recognition menjadi sistem utama dalam interaksi manusia ke robot. Dengan adanya *Voice Recognition* maka dapat dengan mudah manusia memberikan perintah atau peringatan atau sekedar sapaan yang nantinya akan direspon oleh robot memanfaatkan *database Voice Recognition* milik Google yang dapat diakses secara *realtime* dan gratis.

Tabel database skenario *human interaction* terbagi atas empat informasi yaitu ID, Input, Output dan Status.

- ID adalah kolom identitas dari masing-masing baris data, ID bertipe integer primary key. Setiap baris mempunyai ID yang unik dan tidak sama dengan barisan data yang lain.
- Input adalah kolom untuk voice recognition, karena aktifitas voice recognition menghasilkan keluaran berupa teks, maka kolom Input merupakan kumpulan data-data yang bertipe string.
- Output merupakan kolom untuk aktifitas text to speech, karena aktifitas ini membutuhkan data bertipe text maka kolom Output merupakan kumpulan data-data yang bertipe string.
- Status adalah kolom data-data yang akan dikirimkan ke mikrokontroller dan dapat dipakai juga konfigurasi data-data sensor yang dikirimkan oleh mikrokontroller ke Android yang nantinya akan direspon dengan keluaran berupa gerakan robot dan suara.

Khusus untuk Status terdiri atas 4 data berbeda dalam data selebar empat byte. Byte pertama berisi data kamera, kedua merupakan data kecepatan, ketiga merupakan data jarak dan yang terakhir merupakan data lengan. Data status ini akan dibandingkan dengan data sensor hasil kiriman dari mikrokontroller yang sudah diekstraksi.

Tabel 3.7 Tabel Skenario *Human Interaction to Robot*

ID	Input	Output	Status
1	How are you	Fine, thank you, and you?	02-00-00-06
2	Nice to meet you	Nice to meet you too	02-00-00-03
3	Goodbye	Bye	02-00-00-00
4	Good night	Good night	02-00-00-06
5	Good afternoon	Good afternoon	02-00-00-06
6	Good evening	Good evening	02-00-00-06
7	Good morning	Good morning	02-00-00-06
8	See you later	See you	02-00-00-03
9	It was a pleasure seeing you	Thanks	02-00-00-06
10	What is your name	My name is Ascobot	02-00-00-06
11	My name is		02-00-00-03
12	Hello	Hello	02-00-00-05

3.3 Perancangan Interaksi Robot ke Manusia

Tabel database skenario robot interaction terbagi atas empat informasi yaitu ID, Input, Output dan Status.

- ID adalah kolom identitas dari masing-masing baris data, ID bertipe integer primary key. Setiap baris mempunyai ID yang unik dan tidak sama dengan barisan data yang lain.
- Input adalah kolom untuk voice recognition, karena aktifitas voice recognition menghasilkan keluaran berupa teks, maka kolom Input merupakan kumpulan data-data yang bertipe string. Untuk robot interaction ini tidak menggunakan data-data pada kolom dua. Jadi kolom Input pada tabel yang ditampilkan di selanjutnya akan lebih digunakan untuk menerangkan kondisi dari robot dan lingkungan yang digunakan sebagai *trigger* respon.
- Output merupakan kolom untuk aktifitas text to speech, karena aktifitas ini membutuhkan data bertipe text maka kolom Output merupakan kumpulan data-data yang bertipe string.

- Status adalah kolom data-data yang akan dikirimkan ke mikrokontroler dan dapat dipakai juga konfigurasi data-data sensor yang dikirimkan oleh mikrokontroler ke Android yang nantinya akan direspon dengan keluaran berupa gerakan robot dan suara.

Khusus untuk Status terdiri atas 4 data berbeda dalam data selebar empat byte. Byte pertama berisi data kamera, kedua merupakan data kecepatan, ketiga merupakan data jarak dan yang terakhir merupakan data lengan. Data status ini akan dibandingkan dengan data sensor hasil kiriman dari mikrokontroler yang sudah diekstraksi.

Tabel 3.8 Tabel Skenario *Robot Interaction*

ID	Keterangan	Output	Status
14	Robot sedang bergerak ke suatu objek dan mendeteksi adanya halangan	Stay away	00-03-01-06
15	Robot starting	Initialization	09-09-09-09
16	Robot mendeteksi kerumunan orang pada saat diam di tempat	It's so crowded	00-00-01-06

BAB 4

PENGUJIAN DAN ANALISIS

Setelah melakukan perancangan integrasi Android terhadap robot penerima tamu dengan memanfaatkan sensor yang terpasang di robot sebagai masukannya dan suara dan gerakan robot sebagai keluarannya maka didapatkan suatu hasil data pengujian. Tujuan dari pengujian adalah untuk menentukan parameter status yang tepat yang akan diberikan ke mikrokontroler yang nantinya digunakan untuk menggerakkan semua subsistem, dan juga penentuan respon yang tepat sesuai status dari semua subsistem yang teintegrasi di robot.

4.1 Hasil Pengujian dan Analisis Interaksi Manusia ke Robot

Hasil pengujian didapat dari menjalankan sistem hasil perancangan yang telah dibuat lalu memberikan perlakuan-perlakuan khusus lalu mengamati respon sistem yang telah dibuat. Pada pengujian dana analisis interaksi robot ke manusia telah dilakukan dicoba semua skenario yang telah dirancang di bab 3, dan berikut adalah tabel kebenarannya berrdasarkan ID masing-masing data:

Tabel 4.1 Tabel Hasil Pengujian Interaksi Manusia ke Robot

ID	Respon Suara	Kondisi Subsistem Setelah Diberikan Masukan Suara			
		Kamera	Kecepatan	Jarak	Lengan
1	Benar	Deteksi Wajah	Nol	Dekat	Default
2	Benar	Deteksi Wajah	Nol	Dekat	Lengan kanan terangkat 90° ke depan, menggenggam
3	Benar	Deteksi Wajah	Nol	Dekat	Lengan kanan terangkat 180° ke atas
4	Benar	Deteksi Wajah	Nol	Dekat	Default
5	Benar	Deteksi Wajah	Nol	Dekat	Default
6	Benar	Deteksi Wajah	Nol	Dekat	Default
7	Benar	Deteksi Wajah	Nol	Dekat	Default
8	Benar	Deteksi Wajah	Nol	Dekat	Lengan kanan terangkat 90° ke depan, menggenggam
9	Benar	Deteksi Wajah	Nol	Dekat	Default
10	Benar	Deteksi Wajah	Nol	Dekat	Default
11	Benar	Deteksi Wajah	Nol	Dekat	Default
12	Benar	Deteksi Wajah	Nol	Dekat	Lengan kiri dan kanan terangkat 90° ke depan, menggenggam

Terlihat pada tabel bahwa respon dalam bentuk suara dan semua subsistem dari robot penerima tamu selalu benar, sesuai dengan skenario yang telah dibuat. Untuk bagian respon dalam bentuk suara Android berhasil memberikan keluaran. Dan untuk respon lain lebih difokuskan ke pergerakan lengan, kondisi *default* pada status lengan adalah kondisi dimana kedua lengan sedang tidak melakukan gerakan atau berada di titik 0°.

Dengan tingkat keberhasilan yang 100% atau sesuai dengan yang diinginkan menunjukkan sistem sudah stabil. Sistem Android telah cukup handal untuk melakukan proses logika dan dapat menghasilkan output yang sesuai dengan keinginan.

Satu hal yang paling penting dalam pembacaan masukan suara adalah jarak dan *pronouncation* dari yang mengucapkan. Karena robot tidak mengenali kalimat yang diucapkan apabila terlalu jauh atau pelafalan yang buruk dari sumber.

Untuk hasil pengujian pada skenario interaksi robot ke manusia dilakukan dengan cara mengatur agar robot mencapai kondisi sesuai data status dari masing-masing sensor yang telah dibuat agar robot dapat memberikan respon berdasarkan kondisi itu dan diamati apakah robot mengeluarkan respon atau tidak.

Setelah melakukan beberapa pengujian dari sistem yang telah dibuat tercipatalah tabel kebenaran dari hasil respon robot, yaitu:

Tabel 4.2 Tabel Hasil Pengujian interaksi Robot ke Manusia

ID	Kondisi Subsistem				Respon
	Kamera	Kecepatan	Jarak	Lengan	
14	Mendeteksi keramaian	100%	Normal	Diam	Benar
15	-	0%	Normal	Diam	Benar
16	Mendeteksi keramaian	0%	Normal	Diam	Benar

Khusus untuk ID 15 merupakan ID starting, mikrokontroller akan secara otomatis mengirimkan data status apabila pada saat starting robot. Adapun statusnya adalah 09-09-09-09. Dengan tingkat keberhasilan yang 100% atau sesuai dengan yang diinginkan menunjukkan sistem sudah stabil. Sistem Android

telah cukup handal untuk melakukan proses logika dan dapat menghasilkan output yang sesuai dengan keinginan.

4.2 Hasil Pengujian Pengenalan Suara Berdasarkan Jarak dan Pelafalan

Dengan menggunakan hanya dari *microphone* yang ada di Android diduga keberhasilan pengenalan suara akan sangat terpengaruh. Untuk itu dilakukan pengujian berdasarkan variasi jarak dengan tingkat kebisingan normal. Pengujian dilakukan sepuluh kali untuk tiap jarak dengan variasi kata satu hingga 3 kata.

Tabel 4.3 Tabel Kebenaran Berdasarkan Jarak (cm)

Jumlah Kata	Persentase Keberhasilan					
	40	50	60	70	80	90
1	80%	80%	60%	40%	10%	10%
2	80%	70%	30%	20%	20%	0%
3	100%	70%	20%	30%	0%	0%
Rata-rata	87%	73%	37%	30%	10%	3.3%

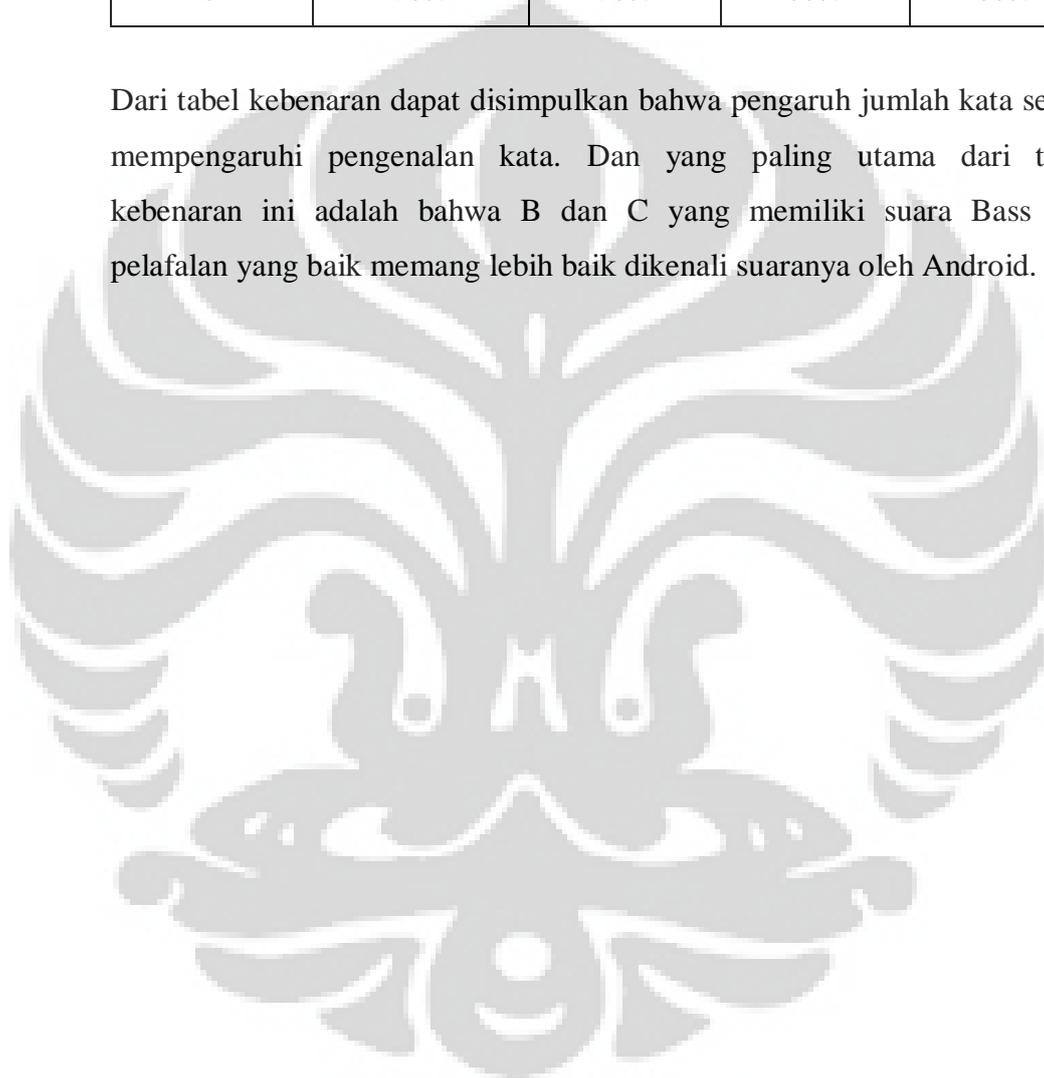
Dari tabel kebenaran diketahui bahwa data terbaik dimiliki oleh jarak antara 40 hingga 50 cm. Hal ini dikarenakan makin jauh dari sumber maka *noise* akan lebih banyak yang bercampur yang membuat suara utama menjadi tidak fokus lagi, selain itu pula suara sumber makin jauh akan makin lemah diterima. Hal ini sangat bergantung sekali pada *microphone*. Dan untuk robot penerima tamu komunikasi dengan manusia melalui media suara disarankan dengan jarak di bawah 50 cm.

Setiap manusia memiliki frekuensi suara yang berbeda-beda, selain itu, pelafalan bahasa Inggris yang berbeda-beda pula. Dua hal tersebut diujikan untuk mengetahui pengaruh sumber suara dan pelafalan terhadap hasil pengenalan suara oleh Android. Adapun pengujian dilakukan dengan mengambil sampel ke empat orang berbeda. Pengujian dilakukan sepuluh kali untuk tiap jarak dengan variasi kata satu hingga tiga kata. Berikut adalah tabel kebenaran yang didapat:

Tabel 4.4 Tabel Kebenaran Pengaruh Variasi Sumber Suara

Jumlah Kata	Persentase Keberhasilan			
	A	B	C	D
1	70%	90%	100%	80%
2	60%	90%	80%	70%
3	70%	70%	60%	60%

Dari tabel kebenaran dapat disimpulkan bahwa pengaruh jumlah kata selalu mempengaruhi pengenalan kata. Dan yang paling utama dari tabel kebenaran ini adalah bahwa B dan C yang memiliki suara Bass dan pelafalan yang baik memang lebih baik dikenali suaranya oleh Android.



BAB 5 PENUTUP

5.1 Kesimpulan

Dari keseluruhan pembahasan dalam skripsi ini dapat disimpulkan beberapa hal, yaitu:

- Komunikasi Bluetooth serial antara mikrokontroller dengan Android berhasil dengan baik dan *robust*. Hal ini dibuktikan dengan tidak adanya kesalahan dalam pengiriman dan penerimaan data selama dilakukan pengujian.
- Jarak dan pelafalan sangat mempengaruhi hasil pengenalan suara atau *voice recognition*. Hal ini sesuai dengan hasil pengujian pada tabel kebenaran 4.3 dan 4.4.
- Sistem yang diciptakan berhasil diuji kehandalannya dan dapat diaplikasikan pada robot penerima tamu.
- Penggunaan database SQLite memudahkan untuk pembuatan *library* baru.

DAFTAR REFERENSI

- [1] Allen, Grant, and Owens Mike. *The Definitive Guide to SQLite*. New York: Apress, 2010.
- [2] *Android Developers*. <http://developer.android.com>.
- [3] Dimarzio, Jerome (J.F). *Android A Programmer's Guide*. New York: McGraw-Hill, 2008.
- [4] Harneldo, Diko. *Rancang Bangun Sistem Bilateral Teleoperation Multi DOF dengan Serial Servo*. Bachelor Thesis, Depok: Universitas Indonesia, 2010.
- [5] Meier, Reto. *Professional Android Application Development*. Indianapolis: Wiley Publishing, Inc, 2009.
- [6] Wijaya, Tjundra, and Dani Setianto. "Pemanfaatan Bluetooth dan Sensor Accelerometer pada Ponsel Berbasis Android untuk Pengontrolan Gerak Mobile Robot." 2012: 1-9.
- [7] Wiryadinata, Romi. <http://wiryadinata.web.id>.