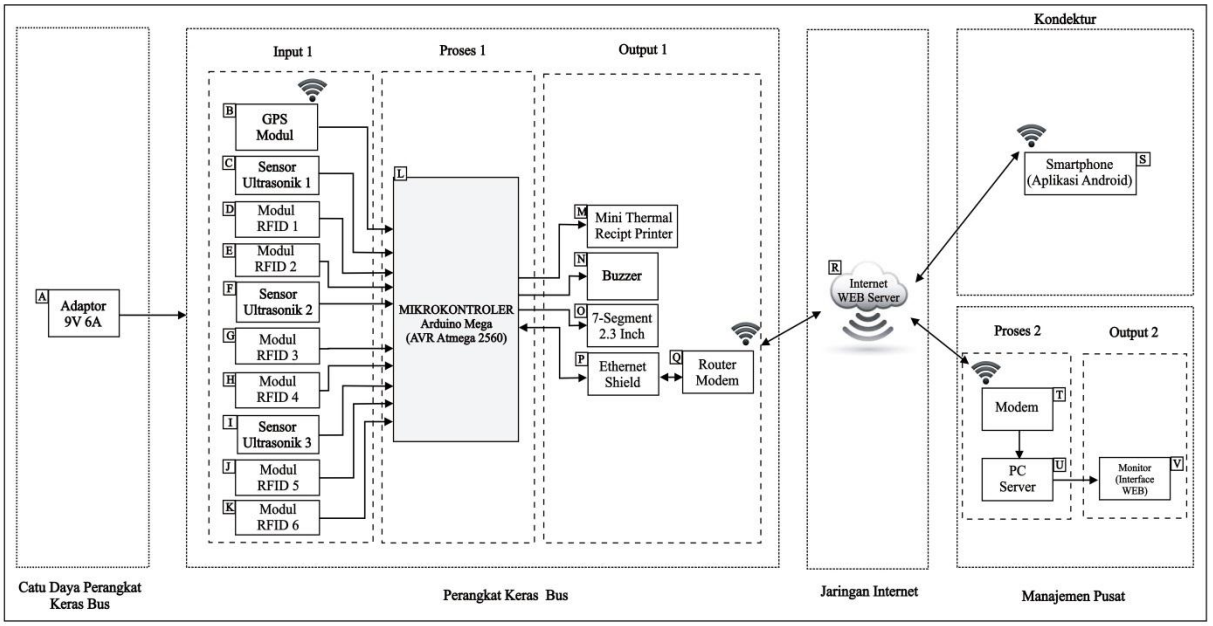
BAB III
PERANCANGAN SISTEM

Pada bab ini akan dijelaskan keseluruhan dari sistem yang akan dibuat, perancangan sistem terdiri dari perancangan perangkat keras dan perangkat lunak.

3.1 Perancangan Perangkat Keras

Perancangan perangkat keras yang dilakukan meliputi dua tahap. Yang pertama adalah perancangan diagram blok sistem secara umum. Yang kedua adalah perancangan perangkat keras secara keseluruhan, meliputi perancangan masukan berupa sensor ultrasonik, modul RFID dan GPS, serta keluaran berupa tampilan 7-Segment, bunyi peringatan buzzer, serta printer untuk pencetakan data penumpang. Diagram blok sistem secara umum, yang akan dibangun, ditunjukan pada gambar 3.1.



Gambar 3.1 Diagram Blok Sistem Secara Umum

a. Adaptor : Berfungsi sebagai pengkonversi *supply* tegangan ac ke tegangan dc.

b. **GPS :** Sebagai pemantau posisi Bus, sehingga dapat diketahui apabila bus berada pada posisi *"safezone"* (kondisi pengiriman jumlah penumpang tidak dilakukan), atau berada pada posisi *"outzone"* (kondisi pengiriman jumlah penumpang dilakukan).

c. **Sensor Ultrasonik 1 :** Berfungsi sebagai pendeteksi penumpang naik, pada pintu depan bus.

d. **Modul RFID 1 :** Berfungsi sebagai pembeda kondektur dengan penumpang, saat kondisi kondektur naik pada pintu luar depan bus.

e. **Modul RFID 2 :** Berfungsi sebagai pembeda kondektur dengan penumpang, saat kondisi kondektur turun, pada pintu dalam depan bus.

f. **Sensor Ultrasonik 2 :** Berfungsi sebagai pendeteksi apabila ada penumpang naik, melalui pintu supir bus.

g. **Modul RFID 3** : Berfungsi sebagai pembeda Sopir dengan penumpang saat kondisi sopir naik, pada pintu luar sopir bus.

h. **Modul RFID 4** : Berfungsi sebagai pembeda sopir dengan penumpang, saat kondisi sopir turun, pada pintu dalam sopir bus.

i. **Sensor Ultrasonik 3 :** Berfungsi sebagai pendeteksi penumpang turun, pada pintu belakang bus.

j. **Modul RFID 5** : Berfungsi sebagai pembeda kondektur dengan penumpang, saat kondisi kondektur naik, pada pintu luar belakang bus.

k. **Modul RFID 6** : Berfungsi sebagai pembeda kondektur dengan penumpang, saat kondisi kondektur turun, pada pintu dalam belakang bus.

l. **Mikrokontroler Arduino Mega2560**: Sebagai pemeroses (*controller*) yang mengatur sistem kerja dari alat yang ada didalam bus.

m. **Mini Thermal Printer** : Sebagai alat yang digunakan untuk mencetak *receipt* (struk) data penumpang yang diterima dari web.

n. **Buzzer** : Perangkat bunyi yang digunakan sebagai penanda perhatian bahwa kapasitas penumpang sudah melebihi kapasitas maksimum.

o. **7-Segment** : Sebagai penampil jumlah penumpang didalam bus.

p. **Ethernet Shield** : Sebagai modul penghubung antara *hardware (Mikrokontroler Arduino)* dengan jaringan internet.

q. ***Router* dan Modem1** : Sebagai perangkat yang digunakan untuk dapat mengakses jaringan internet pada Ethernet Shield w5100.

r. **Web**: Web digunakan sebagai media pertukaran data, dan sebagai pusat *database* penyimpanan informasi jumlah penumpang yang didapat dari bus, serta data penumpang yang didapat dari aplikasi Android kondektur.

s. **Perangkat Android** : Sebagai perangkat yang digunakan untuk menerima pemberitahuan(*Pop-Up*) dari web manajemen pusat armada bus, serta sebagai perangkat yang digunakan oleh kondektur untuk memasukan serta mengirim data penumpang bus pada web manajemen pusat armada bus.

t. **Modem2 :** Perangkat yang digunakan untuk mengakses jaringan internet pada PC *server***.**

u. **PC server :** Sebagai pengatur dan pemonitor web pada manajemen pusat armada bus.

v. **Monitor :** Sebagai antarmuka web yang akan digunakan pekerja pada Manajemen Pusat armada bus.

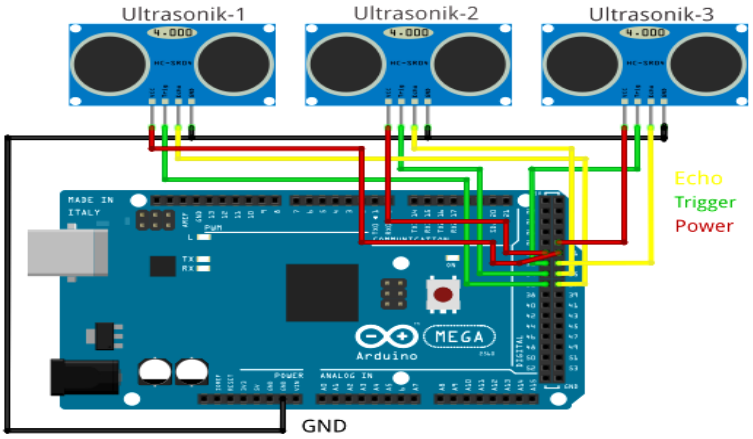
3.1.1 Perancangan Masukan

Masukan data pada Mikrokontroler Arduino Mega2560, berasal dari beberapa komponen masukan sebagai berikut:

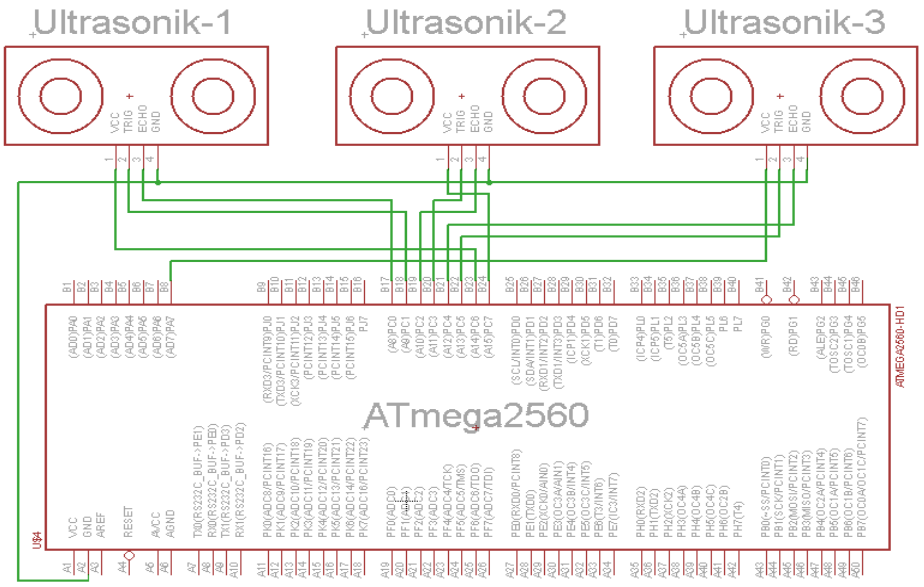
a. Sensor Ultrasonik

Sensor ini bekerja dengan memantulkan suara pada frequensi 40kHZ, dimana pantulan suara akan dirubah menjadi satuan jarak. Sensor ini mampu mendeteksi adanya penumpang apabila jarak pantulan dari sensor yang didapat kurang dari jarak maksimum pantulan yang telah ditentukan. Dalam proses perancangan alat ini ditentukan jarak maksimum pantulan sepanjang 70cm. Selain itu terdapat 3 buah sensor yang akan dipasang, yakni sensor ultrasonik-1 digunakan sebagai pendeteksi penumpang naik pada pintu depan bus, sensor ultrasonik-2 digunakan pada pintu masuk sopir bus, serta sensor

ultrasonik-3 digunakan sebagai pendeteksi penumpang turun pada pintu belakang bus. rangkaian sensor ultrasonik ini dapat dilihat pada gambar 3.2 dan gambar 3.3.



Gambar 3.2 Rangkaian Pada Sensor Ultrasonik



Gambar 3.3 Rangkaian Skematik Pada Sensor Ultrasonik

Keterangan dari gambar 3.2 dan gambar 3.3, konfigurasi rangkaian sensor ultrasonik dapat dilihat pada tabel 3.1.

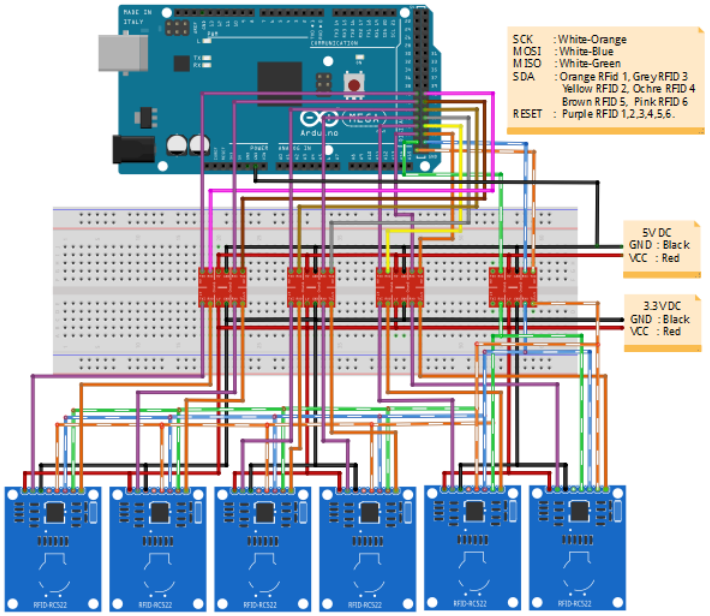
Tabel 3.1 Konfigurasi Pin I/O Arduino Pada Rangkaian Sensor Ultrasonik

Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
29	Digital	PA7 (No Pin 71)	*Output*	Tegangan pada sensor ultrasonik-3
30	Digital	PC7 (No Pin 60)	*Output*	Tegangan pada sensor

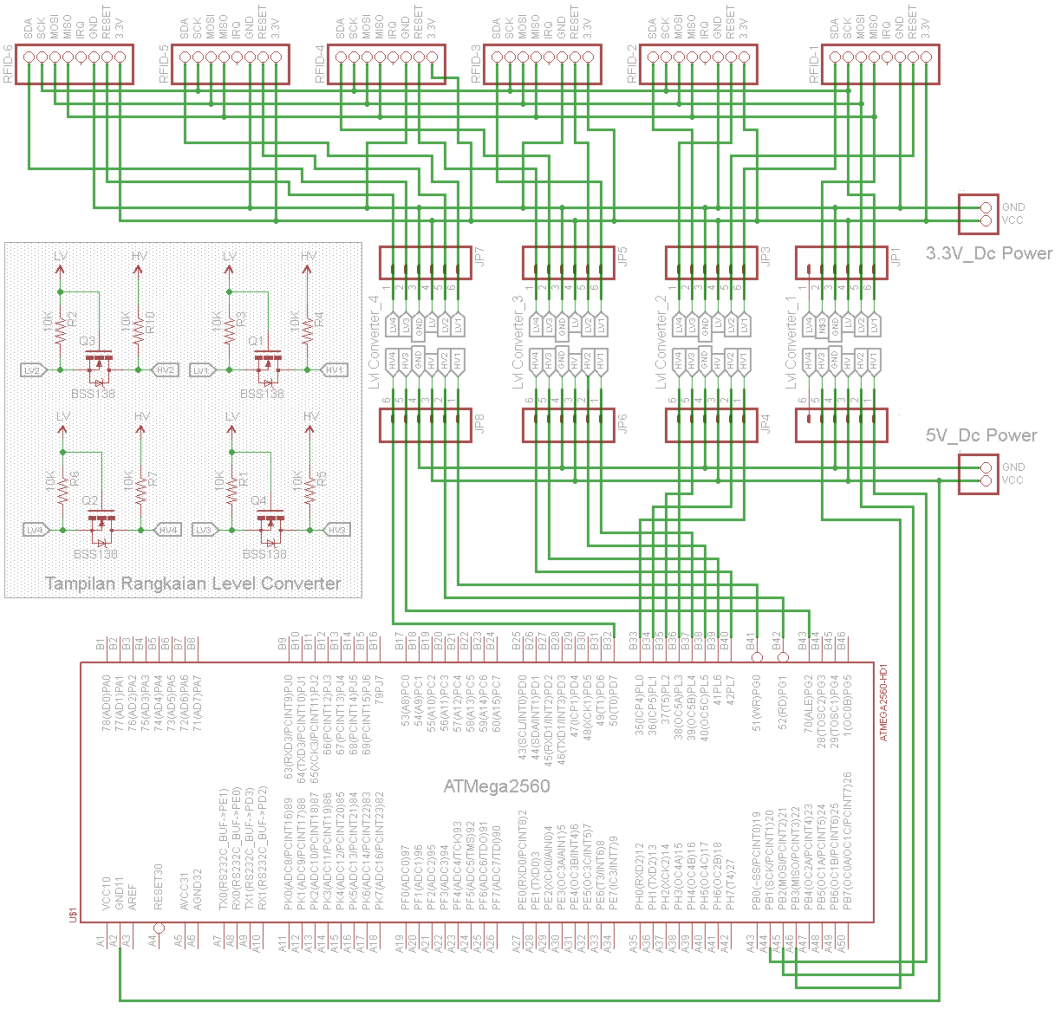
				ultrasonik-2
31	Digital	PC6 (No Pin 59)	*Output*	Tegangan pada sensor ultrasonik-1
32	Digital	PC5 (No Pin 58)	*Output*	*Trigger* pada sensor ultrasonik-3
33	Digital	PC4 (No Pin 57)	*Input*	Penerima (*Echo*) dari sensor ultrasonik-3
34	Digital	PC3 (No Pin 56)	*Output*	*Trigger* pada sensor ultrasonik-2
35	Digital	PC2 (No Pin 55)	*Input*	Penerima (*Echo*) dari sensor ultrasonik-2
36	Digital	PC1 (No Pin 54)	*Output*	*Trigger* pada sensor ultrasonik-1
37	Digital	PC0 (No Pin 53)	*Input*	Penerima (*Echo*) dari sensor ultrasonik-1
GND	*Power*	*Ground*	Tegangan	Menghubungkan sensor ultrasonik-1,2,3 dengan *ground*

b. RFID

Modul RFID MFRC522 merupakan *short* RFID *range*, yang mampu mendeteksi *tag* dengan kode id yang telah ditentukan, dimana *tag* berikut akan diberikan kepada kondektur dan sopir. Pada perancangan alat, RFID digunakan sebagai kondisi untuk mematikan sensor ultrasonik sementara waktu, hal ini dimaksudkan agar kondektur atau sopir tidak terbaca oleh sensor ultrasonik, dengan demikian kondisi kondektur dan sopir ketika naik dan turun dari bus tidak akan terhitung, sehingga tidak akan mempengaruhi jumlah dari penumpang didalam bus. Dalam proses perancangan alat ini digunakan 6 buah RFID, dimana 2 buah RFID akan ditempatkan pada masing-masing pintu, yaitu pintu depan, pintu belakang, dan pintu sopir. rangkaian modul RFID ini dapat dilihat pada gambar 3.4 dan gambar 3.5.



Gambar 3.4 Rangkaian Pada Modul RFID



Gambar 3.5 Skematik Pada Modul RFID

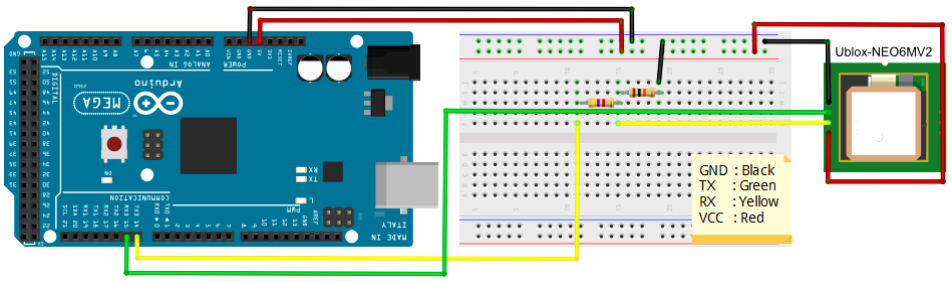
Keterangan dari gambar 3.4 dan gambar 3.5 konfigurasi rangkaian modul RFID dapat dilihat pada tabel 3.2.

Tabel 3.2 Konfigurasi Pin I/O Arduino Dengan Modul RFID

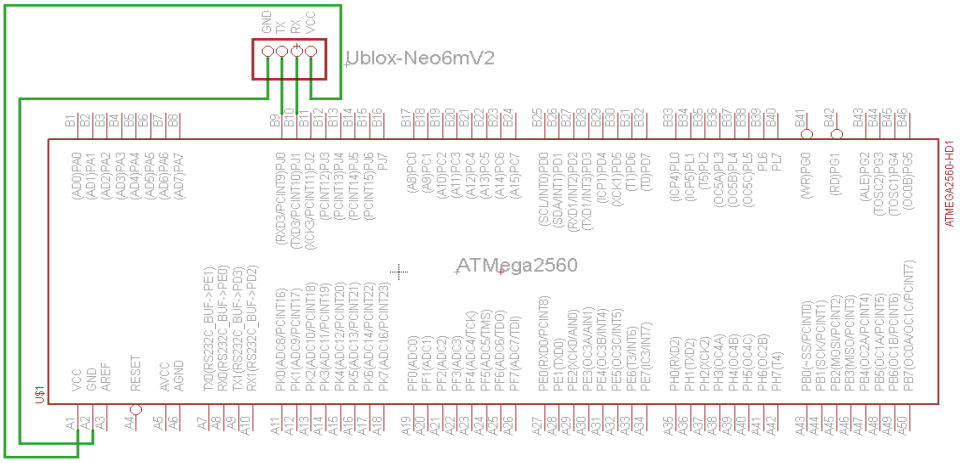
Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
52	Digital	PB1 (No Pin 20)	*Input*	Masukan data SCK dari RFID 1,2,3,4,5,6
51	Digital	PB2 (No Pin 21)	*Input*	Masukan data MOSI dari RFID 1,2,3,4,5,6
50	Digital	PB3 (No Pin 22)	*Input*	Masukan data MISO dari RFID 1,2,3,4,5,6
49	Digital	PL0 (No Pin 35)	*Input*	Masukan data SDA dari RFID 1
48	Digital	PL1 (No Pin 36)	*Input*	Masukan data Reset dari RFID 1
47	Digital	PL2 (No Pin 37)	*Input*	Masukan data SDA dari RFID 2
46	Digital	PL3 (No Pin 38)	*Input*	Masukan data Reset dari RFID 2
45	Digital	PL4 (No Pin 39)	*Input*	Masukan data SDA dari RFID 3
44	Digital	PL5 (No Pin 40)	*Input*	Masukan data Reset dari RFID 3
43	Digital	PL6 (No Pin 41)	*Input*	Masukan data SDA dari RFID 4
42	Digital	PL7 (No Pin 42)	*Input*	Masukan data Reset dari RFID 4
41	Digital	PG0 (No Pin 51)	*Input*	Masukan data SDA dari RFID 5
40	Digital	PG1 (No Pin 52)	*Input*	Masukan data Reset dari RFID 5
39	Digital	PG2 (No Pin 70)	*Input*	Masukan data SDA dari RFID 6
38	Digital	PD7 (No Pin 50)	*Input*	Masukan data Reset dari RFID 6
Vcc	*Power*	Vcc 3.3 V DC	Tegangan	Menghubungkan komponen dengan tegangan 3.3V DC
GND	*Power*	*Ground*	Tegangan	Menghubungkan Komponen dengan ground

c. GPS Ublox Neo6 mV2

Modul GPS Neo 6Mv2 U-Blox berfungsi sebagai penerima GPS (*Global Positioning System Receiver*) yang dapat mendeteksi lokasi dengan menangkap dan memroses sinyal dari satelit navigasi. Dalam proses perancangan alat ini, lokasi koordinat yang telah didapatkan digunakan sebagai penentu kondisi pengiriman jumlah penumpang didalam bus. rangkaian modul GPS Ublox Neo 6Mv2 ini dapat dilihat pada gambar 3.6 dan gambar 3.7.



Gambar 3.6 Rangkaian Pada GPS Ublox Neo 6mV2



Gambar 3.7 Skematik Pada GPS Ublox Neo 6mV2

Keterangan dari gambar 3.6 dan gambar 3.7, konfigurasi rangkaian GPS dapat dilihat pada tabel 3.3.

Tabel 3.3 Konfigurasi Pin I/O Arduino Pada Rangkaian GPS Ublox Neo

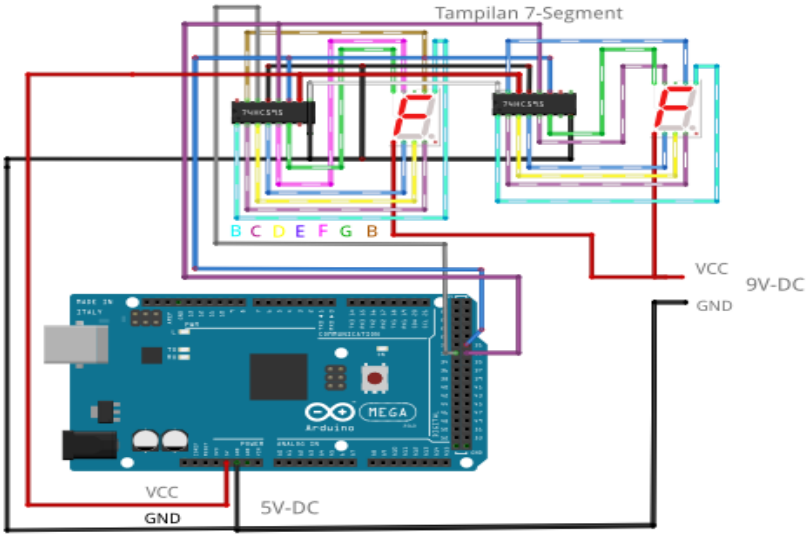
Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
14	Digital	PJ1 (No Pin 64)	*Output*	Masukan data pada Rx GPS
15	Digital	PJ0 (No Pin 63)	*Input*	Keluaran data tx Dari GPS
Vcc	*Power*	Vcc 3.3 V DC	Tegangan	Menghubungkan komponen dengan tegangan 3.3 V DC
GND	*Power*	*Ground*	Tegangan	Menghubungkan Komponen dengan *ground*

3.1.2 Perancangan Keluaran

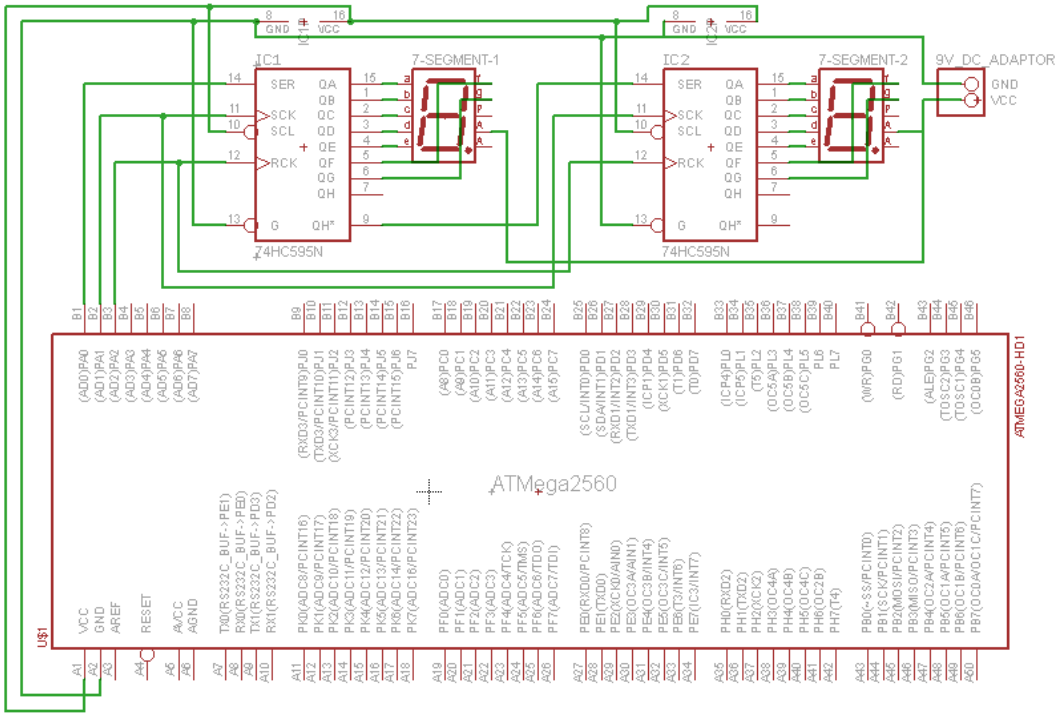
Keluaran dari Mikrokontroler Arduino Mega2560, terdiri dari beberapa komponen sebagai berikut:

a. 7-Segment display dengan IC SN74HC595

7-Segment 2.3 Inch, digunakan sebagai tampilan dari jumlah penumpang bus. Dimana jumlah penumpang yang ada didalam bus akan sesuai dengan nilai yang ditampilkan pada 7-Segment. Sehingga memudahkan kondektur dalam proses pendataan jumlah penumpang. 7-Segment ini bekerja pada tegangan *input* Dc = 6.8-8V dengan Dc *Forward Current per segment* = 30mA. Selain itu penyalaan 7-Segment dilakukan dengan cara *Shift Register*, dengan menggunakan 1-buah IC SN74HC595 pada setiap 7-Segment. Rangkaian 7-Segment ini dapat dilihat pada gambar 3.8 dan gambar 3.9.



Gambar 3.8 Rangkaian Pada 7-segment 2.3 Inch



Gambar 3.9 Skematik Pada 7-segment 2.3 Inch

Keterangan dari gambar 3.8 dan gambar 3.9 konfigurasi rangkaian tampilan *7-Segment* dapat dilihat pada tabel 3.4.

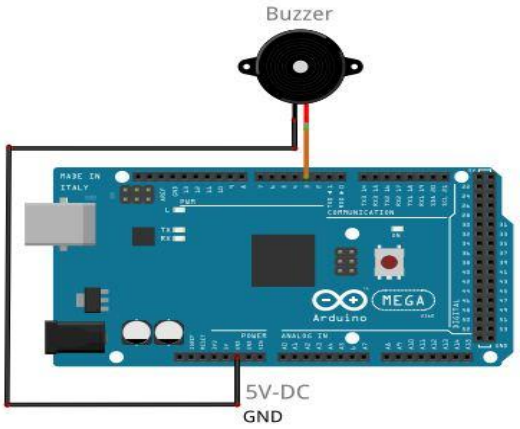
Tabel 3.4 Konfigurasi pin I/O Arduino Pada Rangkaian 7-Segment

Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
22	Digital	PA0 (No Pin 78)	*Output*	Data Shift register
23	Digital	PA1 (No Pin 77)	*Output*	*Clock* Shift Register
24	Digital	PA2 (No Pin 76)	*Output*	*Latch* Shift Register
Vcc	*Power*	Vcc 5 V DC	Tegangan	Menghubungkan komponen dengan tegangan 5 V DC
GND	*Power*	*Ground*	Tegangan	Menghubungkan Komponen dengan *ground*

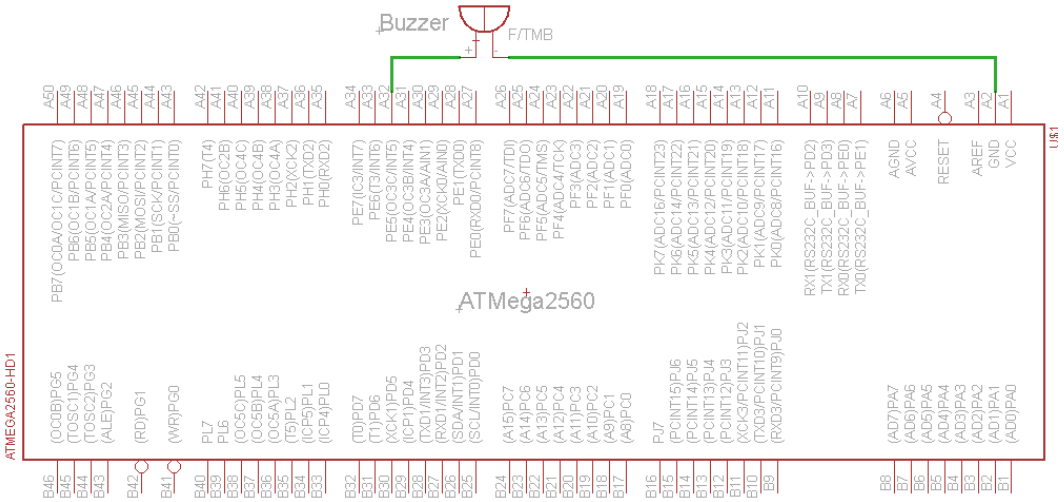
b. Buzzer

Buzzer digunakan sebagai pemberi peringatan dalam Bus. Buzzer akan menyala ketika jumlah penumpang yang berada didalam bus melebihi kapasitas maksimal dari jumlah penumpang yang telah ditetapkan. Gambar

3.10 dan gambar 3.11 Menjelaskan konfigurasi buzzer pada pin Arduino Mega2560.



Gambar 3.10 Konfigurasi Rangkaian Buzzer



Gambar 3.11 Skematik Rangkaian Buzzer

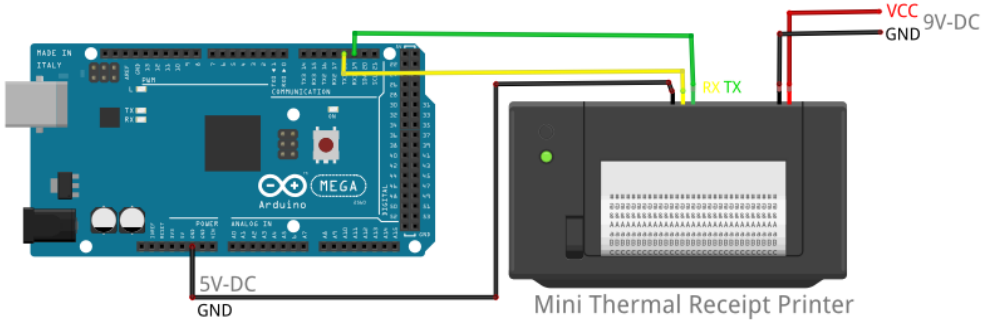
Keterangan dari gambar 3.10 dan gambar 3.11 konfigurasi rangkaian buzzer dapat dilihat pada tabel 3.5.

Tabel 3.5 Konfigurasi Pin I/O Arduino Pada Rangkaian Buzzer

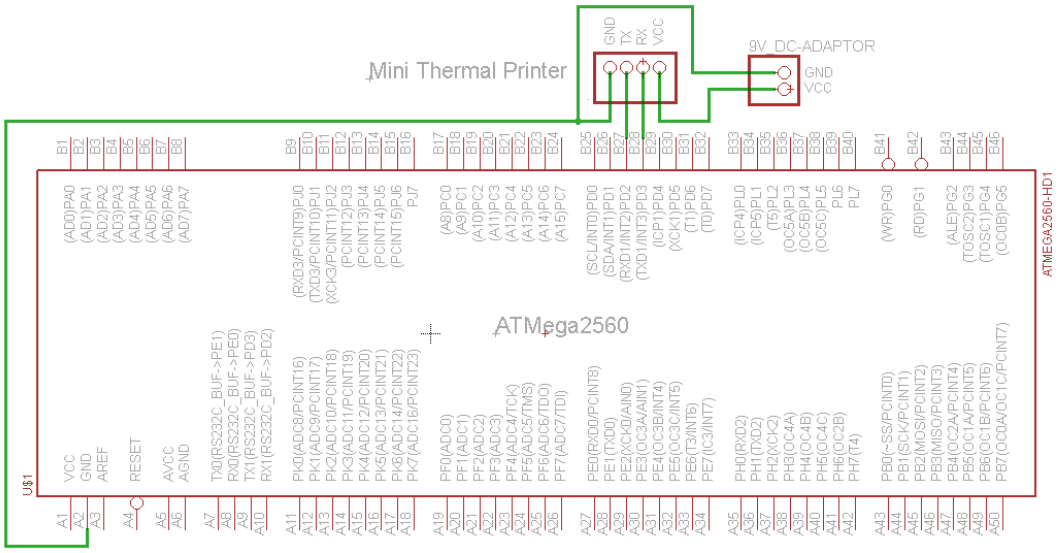
Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
3	Digital	PE5 (No Pin 7)	*Output*	Memberikan tegangan pada buzzer
GND	*Power*	*Ground* 5V DC	Tegangan	Menghubungkan Komponen dengan *ground*

c. Mini Thermal Recipt Printer

Mini Thermal Recipt Printer digunakan sebagai komponen pencetak data penumpang yang telah dimasukan oleh kondektur melalui aplikasi Android. Selain itu pencetakan tampilan kertas dapat disesuaikan dengan mengedit *library* printer yang dapat diunduh pada forum Arduino. Printer ini memiliki sistem kerja dengan cara memanaskan kertas khusus yang dinamakan dengan kertas *Thermal* (*Thermal paper*) atau *thermochromic paper* melalui jarum (*Thermal Print Head*) yang dipanaskan dan didekatkan diatas kertas *thermal* ini sehingga permukaan kertas ini akan menghitam yang selanjutnya terbentuklah suatu obyek seperti tulisan atau gambar. Bentuk konfigurasi rangkaian printer ini dapat dilihat pada gambar 3.12 dan gambar 3.13.



Gambar 3.12 Konfigurasi Rangkaian Mini Thermal Printer



Gambar 3.13 Skematik Rangkaian Mini Thermal Printer

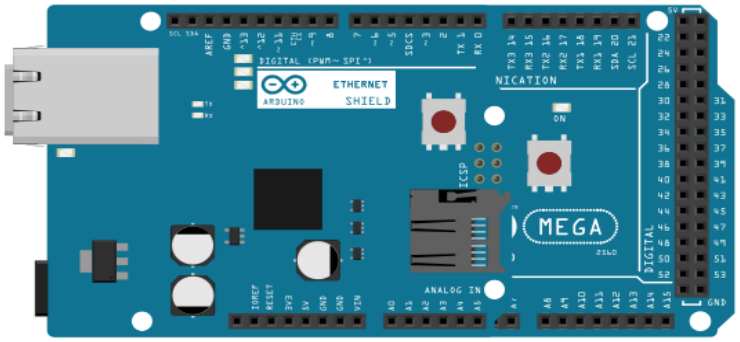
Keterangan dari gambar 3.12 dan gambar 3.13 konfigurasi rangkaian *Mini Thermal Printer* dapat dilihat pada tabel 3.6 berikut.

Tabel 3.6 Konfigurasi Pin I/O Arduino Pada Rangkaian Mini Thermal Printer

Arduino Pin	Port	Pin ATmega2560	Tipe	Keterangan
19	Digital	PD2 (No Pin 45)	*Input*	Menerima data dari printer
18	Digital	PD3 (No Pin 46)	*Output*	Memberikan data pada printer
GND	*Power*	*Ground* 5V DC	Tegangan	Menghubungkan Komponen dengan *ground*

d. Ethernet Shield w5100

Ethernet Shield w5100 digunakan sebagai penghubung Arduino Mega dengan jaringan internet, dimana Ethernet Shield w5100 ini dapat langsung dipasangkan dengan cara memasangkan Ethernet Shield w5100 diatas Arduino Mega2560 dengan menyesuaikan pin dari Arduino Mega2560 tersebut. Gambar 3.14 menjelaskan posisi penempatan ethernet shiled pada Arduino Mega2560.



Gambar 3.14 Penempatan Posisi Ethernet Shield Pada Arduino Mega2560

e. Modem dan Router

Pada perancangan sistem ini modem dan router berfungsi sebagai penyedia layanan jaringan internet. Router yang digunakan adalah router TP-Link MR3020 dengan ukuran yang cukup kecil router ini memiliki konektor RJ45. Dimana konektor ini dapat dihubungkan langsung melalui kabel LAN pada Ethernet Shield w5100, sehingga Arduino Mega2560 akan memiliki

akses pada jaringan internet. Gambar 3.15 menunjukan bentuk fisik router Tp-Link MR3020 dengan modem yang terhubung.



Gambar 3.15 Modem & Router

3.1.3 Perancangan Pemakaian Daya

Perancangan pemakaian daya dilakukan dengan menghitung penggunaan daya dari tiap komponen yang digunakan sebagai berikut:

1. Penghitungan Estimasi Pemakaian Daya Komponen Dari Arduino

Penghitungan estimasi pemakaian daya dari Arduino Mega2560, dilakukan dengan mendapatkan nilai *operating range* tegangan dan arus yang dibutuhkan berdasarkan *datasheet* dari tiap komponen. Kemudian dibandingkan dengan nilai tegangan dan arus yang dapat diberikan oleh tiap *input/output* pin dari Arduino Mega2560. Data hasil penghitungan estimasi pemakaian daya dari Arduino Mega2560 dapat dilihat pada tabel 3.7.

Tabel 3.7 Penghitungan Estimasi Pemakaian Daya Komponen dari Arduino Mega2560

No	Komponen	Vin *Operating Range* Komponen	Vout Arduino Pada Komponen	Keterangan	I *Operating Range* Komponen	Iout Arduino Pada Komponen	Keterangan
1	Sensor Ultrasonik 1	5V	5V	Sesuai	15mA	40mA	Sesuai
2	Sensor Ultrasonik 2	5V	5V	Sesuai	15mA	40mA	Sesuai
3	Sensor Ultrasonik 3	5V	5V	Sesuai	15mA	40mA	Sesuai

4	IC SN74HC595N Shift Register 1	2-6V	5V	Sesuai Range	6mA	40mA	Sesuai
5	IC SN74HC595N Shift Register 2	2-6V	5V	Sesuai Range	6mA	40mA	Sesuai
6	Buzzer	4-7V	5V	Sesuai Range	<30mA	40mA	Sesuai
7	Gps	2.7-3.6V	3.3V	Sesuai Range	45mA	50mA	Sesuai
				Jumlah Arus	<132mA	290mA	Terpenuhi

Dari tabel 3.7, dapat diketahui bahwa nilai tegangan dan arus yang dapat diberikan oleh tiap *input/output* pin dari arduino Mega2560, dapat memenuhi nilai operating range tegangan dan arus yang dibutuhkan oleh tiap komponen.

2. Penghitungan Estimasi Pemakaian Daya Komponen Dari Modul Breadboard Power Supply MB102

Penghitungan pemakaian daya dari modul PSU, dilakukan dengan mendapatkan nilai *operating range* tegangan dan arus yang dibutuhkan berdasarkan *datasheet* dari tiap komponen. Kemudian dibandingkan dengan nilai tegangan dan arus yang dapat diberikan modul Breadboard Power Supply MB102. Data hasil penghitungan estimasi pemakaian daya dari modul PSU dapat dilihat pada tabel 3.8.

Tabel 3.8 Penghitungan Estimasi Pemakaian Daya Komponen dari Modul Breadboard Power Supply MB102

No	Komponen	Vin *Operating Range*	Vin Pada Komponen	Keterangan	I *Operating Range*	I Pada Komponen	Keterangan
1	RFID 1	2.5-3.6V	2.92V	Sesuai Range	13-26mA	16.4mA	Sesuai
2	RFID 2	2.5-3.6V	2.93V	Sesuai Range	13-26mA	16.3mA	Sesuai
3	RFID 3	2.5-3.6V	2.93V	Sesuai Range	13-26mA	19.1mA	Sesuai
4	RFID 4	2.5-3.6V	2.91V	Sesuai Range	13-26mA	19mA	Sesuai
5	RFID 5	2.5-3.6V	2.93V	Sesuai Range	13-26mA	19.2mA	Sesuai
6	RFID 6	2.5-3.6V	2.94V	Sesuai Range	13-26mA	19.1mA	Sesuai
				Jumlah Arus	78-156mA	109.1mA	Terpenuhi

Dari tabel 3.8, dapat diketahui bahwa nilai tegangan dan arus yang dapat diberikan modul *breadboard power supply*, dapat memenuhi nilai operating range tegangan dan arus yang dibutuhkan oleh tiap komponen RFID.

3.2 Perancangan Perangkat Lunak

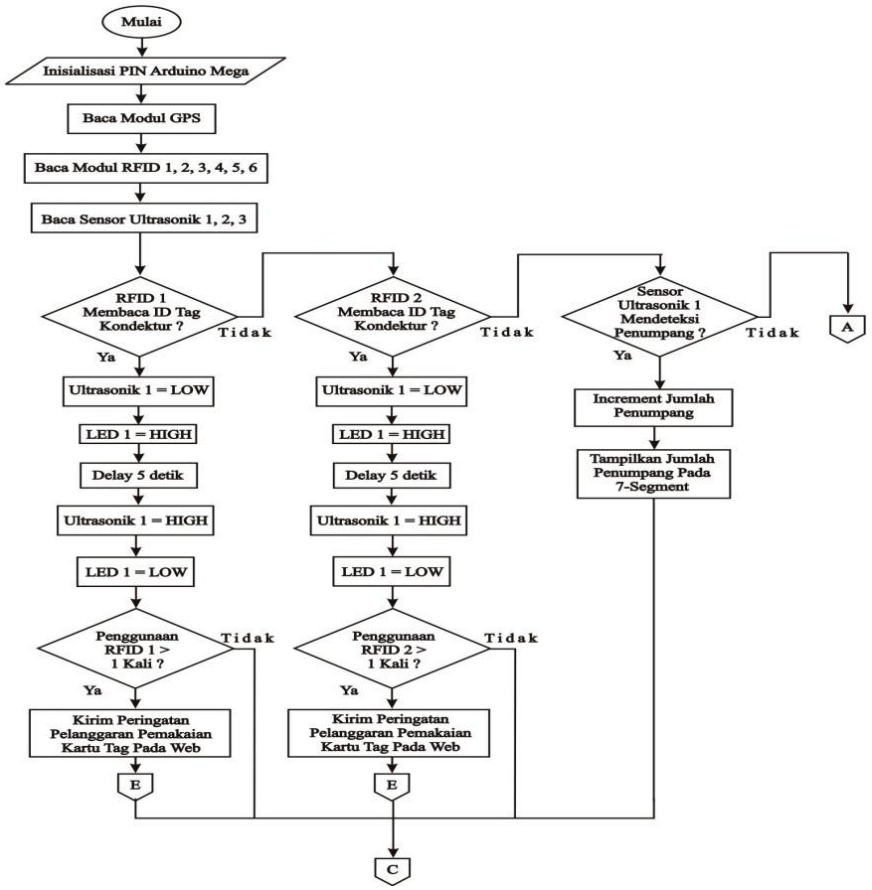
Perancangan perangkat lunak dari sistem yang akan dibuat terdiri dari tiga bagian utama, yaitu perancangan perangkat lunak pada Arduino Mega2560, Perancangan perangkat lunak pada aplikasi Android, serta perancangan perangkat lunak pada Web sebagai pusat penyimpanan data operasional armada bus.

3.2.1 Perancangan Perangkat Lunak Mikrokontroler ATmega2560

Perancangan perangkat lunak pada Mikrokontroler Arduino Mega2560 meliputi diagram alir kondisi naik turunnya Penumpang, Kondektur dan Sopir, pada pintu depan, pintu sopir dan pintu belakang bus. Serta diagram alir kondisi GPS sebagai penentu pengiriman dan penerimaan data penumpang.

a. Kondisi Pintu Depan Bus

Kondisi perancangan perangkat lunak pada pintu depan bagian bus akan dijelaskan melalui diagram alir pada gambar 3.16.



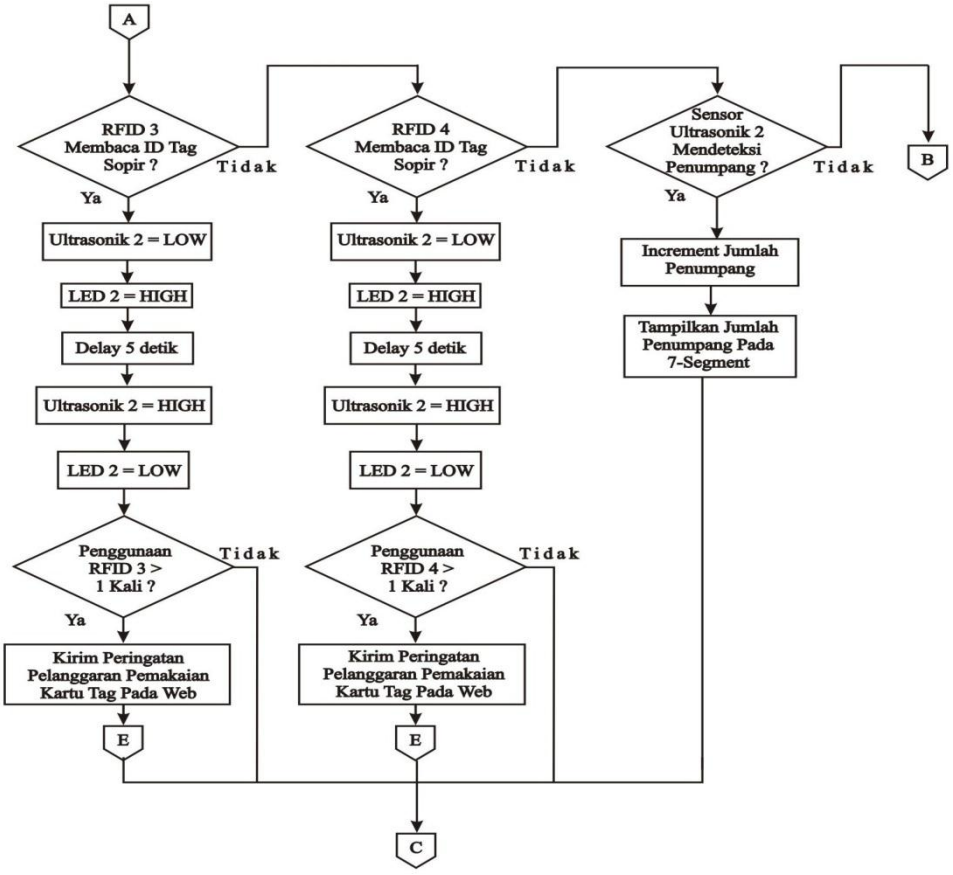
Gambar 3.16 Diagram Alir Perancangan Perangkat Lunak Pada Pintu depan Bus

Pertama-tama program melakukan inisialisasi pin Arduino Mega2560. Kemudian modul GPS, RFID dan sensor ultrasonik mulai bekerja. selanjutnya proses pembacaan *ID tag* dilakukan oleh masing-masing RFID *reader*. Kondisi pertama, apabila RFID-1 (naik kedalam Bus) membaca *ID tag* kondektur, maka LED-1 pada RFID-1 akan menyala, dan mematikan sensor ultrasonik-1 pada pintu depan bus untuk sementara waktu, selain itu pemakaian RFID-1 akan di cek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi kedua apabila RFID-2 (turun dari bus) membaca *ID tag* kondektur maka LED-1 pada RFID-2 akan menyala, dan mematikan sensor ultrasonik-1 pada pintu depan bus untuk sementara waktu, selain itu pemakaian RFID-2 akan di cek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi ketiga, apabila tidak ada

pembacaan kartu terlebih dahulu dan ada kondisi penumpang naik maka Sensor ultrasonik-1 akan mendeteksinya, kemudian akan menambahkan jumlah penumpang dan menampilkan jumlah penumpang terbaru pada 7-Segment *Display*.

b. Kondisi Pintu Sopir Bus

Kondisi perancangan perangkat lunak pada pintu sopir bus akan dijelaskan melalui diagram alir pada gambar 3.17.



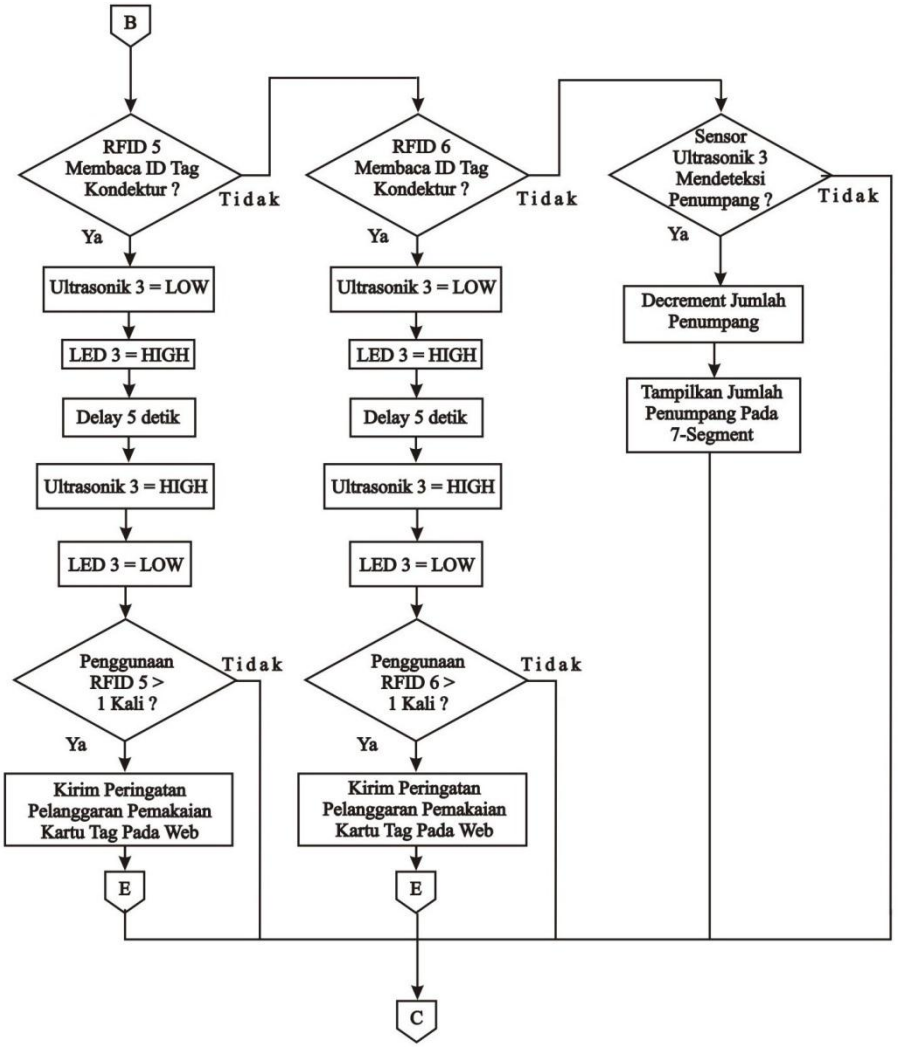
Gambar 3.17 Diagram Alir Perancangan Perangkat Lunak Pada Pintu Sopir Bus

Pada pintu sopir bus, kondisi pertama, apabila RFID-3 (naik kedalam Bus) membaca *ID tag* sopir, maka LED-2 pada RFID-3 akan menyala, dan mematikan sensor ultrasonik-2 pada pintu sopir bus untuk sementara waktu, selain itu pemakaian RFID-3 akan dicek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi kedua apabila RFID-4 (turun dari bus) membaca *ID tag* sopir maka LED-2 pada RFID-4 akan menyala,

dan mematikan sensor ultrasonik-2 pada pintu sopir bus untuk sementara waktu, selain itu pemakaian RFID-4 akan di cek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi ketiga, apabila tidak ada pembacaan kartu terlebih dahulu dan ada kondisi penumpang naik maka Sensor ultrasonik-2 akan mendeteksinya, kemudian akan menambah jumlah penumpang dan menampilkan jumlah penumpang terbaru pada 7-Segment *Display*.

c. Kondisi Pintu Belakang Bus

Kondisi perancangan perangkat lunak pada pintu belakang bagian bus akan dijelaskan melalui diagram alir pada gambar 3.18.

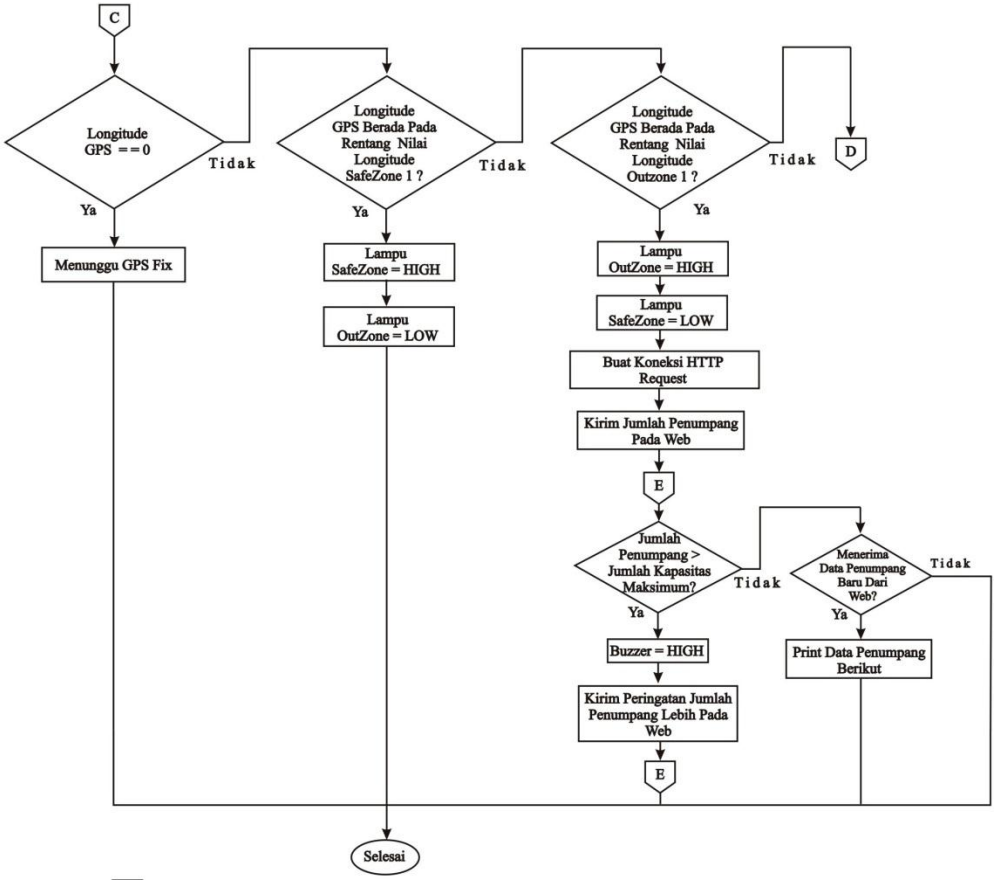


Gambar 3.18 Diagram Alir Kondisi Pintu Belakang Bus

Pada pintu belakang bus, Kondisi pertama, apabila RFID-5 (naik kedalam Bus) membaca *ID tag* kondektur, maka LED-3 pada RFID-5 akan menyala, dan mematikan sensor ultrasonik-3 pada pintu belakang bus untuk sementara waktu, selain itu pemakaian RFID-5 akan di cek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi kedua apabila RFID-6 (turun dari bus) membaca *ID tag* kondektur maka LED-3 pada RFID-6 akan menyala, dan mematikan sensor ultrasonik-3 pada pintu belakang bus untuk sementara waktu, selain itu pemakaian RFID-6 akan di cek apakah pemakaiannya sudah lebih dari satu kali (secara berturut-turut) apabila iya maka kirim laporan pelanggaran penggunaan kartu tag pada web manajemen pusat armada bus. Kondisi ketiga, apabila tidak ada pembacaan kartu terlebih dahulu dan ada kondisi penumpang turun maka sensor ultrasonik-3 akan mendeteksinya, kemudian akan mengurangi jumlah penumpang dan menampilkan jumlah penumpang terbaru pada 7-Segment *Display*.

d. Kondisi GPS Untuk Pengiriman Jumlah Penumpang dan Penerimaan Data Penumpang Bagian-1

Kondisi pengiriman jumlah penumpang dan penerimaan data penumpang ditentukan oleh posisi bus berada. Posisi bus yang didapatkan dari modul GPS digunakan untuk menyeleksi kondisi proses pengiriman jumlah penumpang dan penerimaan data penumpang. Gambar 3.19 menunjukan diagram alir dari kondisi tersebut.



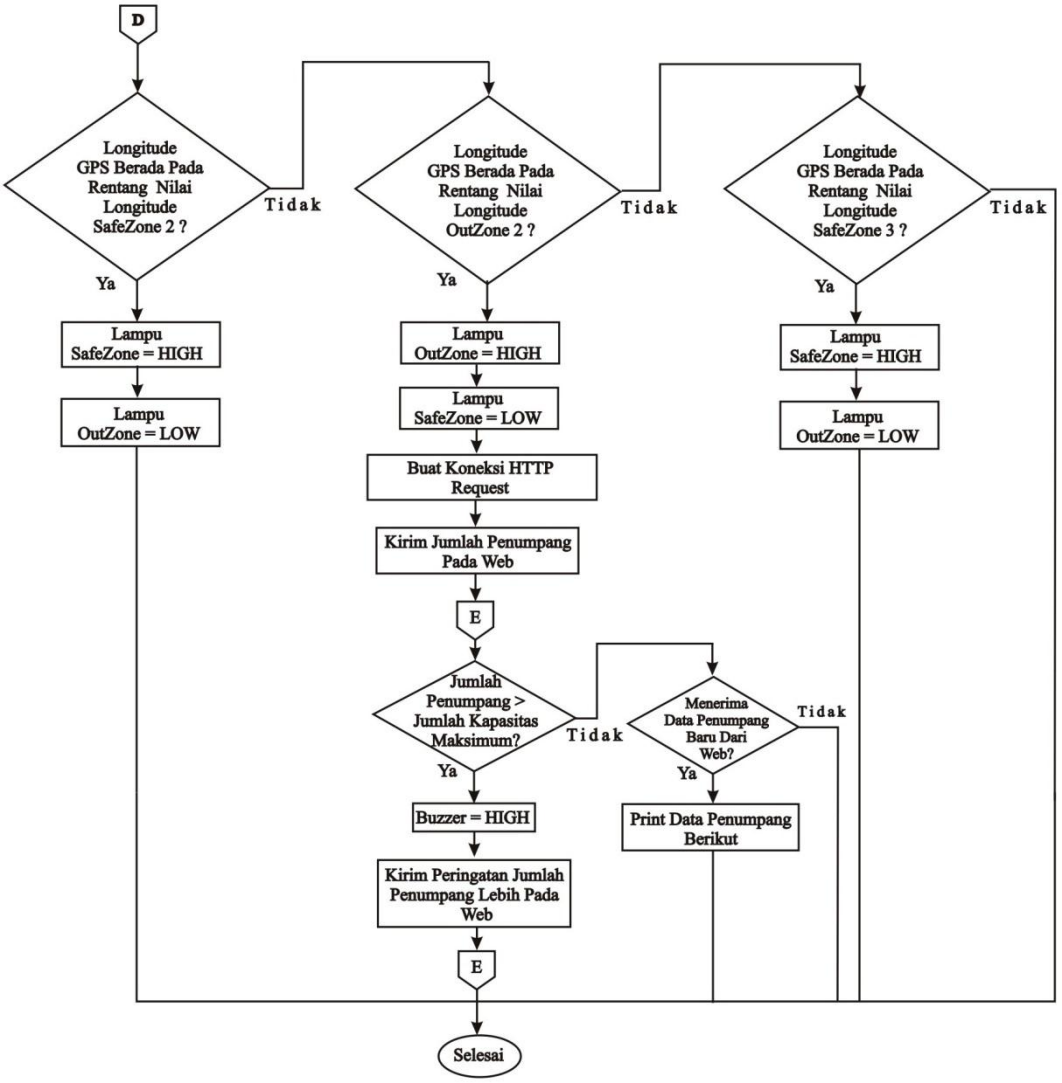
Gambar 3.19 Diagram Alir Kondisi Penerimaan Jumlah Penumpang Pada Web Bagian(1)

Kondisi pertama, menunggu GPS untuk *fix* mendapatkan data *longitude*. Apabila data *longitude* sudah diterima maka memulai untuk melakukan seleksi kondisi pengiriman jumlah penumpang berdasarkan posisi bus berada. Kondisi kedua apabila posisi *longitude* bus berada pada rentang posisi *Safezone-1* (koordinat Kota Bandung, *longitude* = 107.59550-107.57063) maka lampu *Safezone* akan menyala dan koneksi pada web manajemen pusat armada bus akan dihentikan. Kondisi ketiga, apabila posisi *longitude* bus berada pada rentang posisi *Outzone-1* (koordinat jalan dari Bandung menuju Cianjur) maka lampu *Outzone* akan menyala dan sistem akan melakukan koneksi pada web manajemen pusat armada bus, untuk melakukan pengiriman jumlah penumpang. Kemudian pengecekan jumlah penumpang akan dilakukan, apabila jumlah penumpang melebihi kapasitas maksimum, maka nyalakan *buzzer* kemudian peringatan status kelebihan jumlah penumpang akan dikirim pada web manajemen pusat armada bus. Sedangkan apabila kondisi penumpang tidak melebihi jumlah kapasitas maksimum, maka

kondisi selanjutnya adalah melakukan pengecekan apakah ada data penumpang baru diterima?, apabila data penumpang baru diterima, maka lakukan pencetakan data penumpang tersebut.

e. Kondisi GPS Untuk Pengiriman Jumlah Penumpang dan Penerimaan Data Penumpang Bagian-2

Kondisi pengiriman jumlah penumpang dan Penerimaan Data Penumpang Bagian-2 dapat dilihat pada Gambar 3.20.



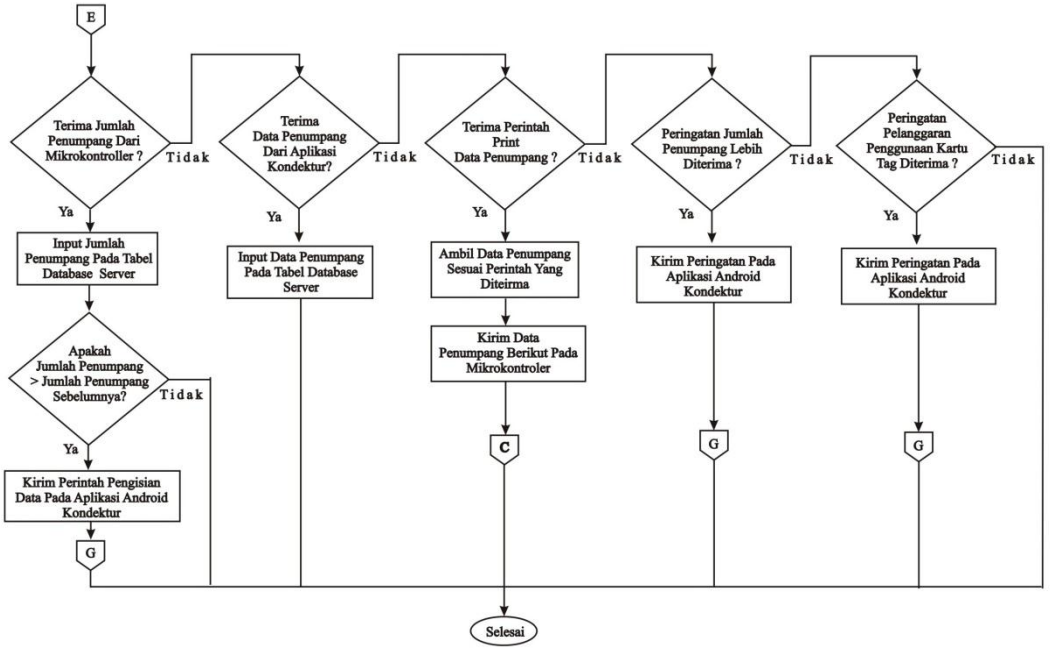
Gambar 3.20 Diagram Alir Kondisi Penerimaan Jumlah Penumpang Pada Web Bagian(2)

Kondisi keempat, apabila posisi *longitude* bus berada pada rentang posisi *Safezone-2* (koordinat Kota Cianjur, *longitude* = 107.29309-107.09236) maka

lampu *Safezone* akan menyala dan koneksi pada web manajemen pusat armada bus akan dihentikan. Kondisi kelima, apabila posisi *longitude* bus berada pada rentang posisi *Outzone-2* (koordinat jalan dari Cianjur menuju Sukabumi) maka lampu *Outzone* akan menyala dan sistem akan melakukan koneksi pada web manajemen pusat armada bus, untuk melakukan pengiriman jumlah penumpang. Kemudian pengecekan jumlah penumpang dilakukan, apabila jumlah penumpang melebihi kapasitas maksimum, maka nyalakan buzzer kemudian peringatan status kelebihan jumlah penumpang akan dikirim pada web manajemen pusat armada bus. Sedangkan apabila kondisi penumpang tidak melebihi jumlah kapasitas maksimum, maka kondisi selanjutnya adalah melakukan pengecekan apakah ada data penumpang baru yang diterima, apabila data penumpang baru diterima, maka lakukan pencetakan data penumpang tersebut. Kondisi keenam, apabila posisi *longitude* bus berada pada rentang posisi *Safezone-3* (koordinat Kota Sukabumi, *longitude* = 106.97354-106.91600) maka lampu *Safezone* akan menyala dan koneksi pada web manajemen pusat armada bus akan dihentikan.

3.2.2 Perancangan Perangkat Lunak Web

Perancangan perangkat lunak pada web manajemen pusat armada bus dilakukan untuk mengetahui bagaimana kondisi penerimaan jumlah penumpang, penerimaan data penumpang serta pengiriman data penumpang pada web manajemen pusat armada bus. Diagram alir Perancangan perangkat lunak pada web manajemen pusat armada bus ini dapat dilihat pada diagram alir Gambar 3.21.

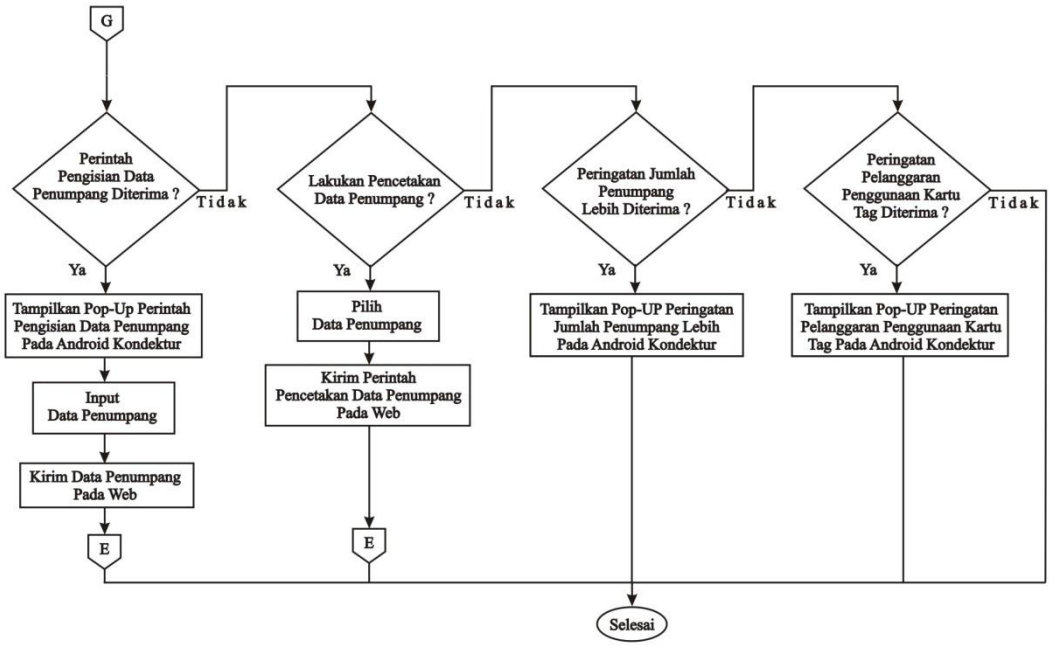


Gambar 3.21 Diagram alir penerimaan jumlah penumpang dan pengiriman data penumpang pada web

Kondisi pertama, ketika jumlah penumpang diterima dari mikrokontroler, maka jumlah penumpang tersebut akan disimpan pada tabel *database* jumlah penumpang. Kemudian apabila nilai jumlah penumpang terbaru yang diterima, lebih besar dibandingkan nilai jumlah penumpang sebelumnya, maka perintah pengisian data penumpang akan dikirim kepada Android kondektur. Kondisi kedua, ketika data penumpang dari Android kondektur diterima, maka data penumpang tersebut akan disimpan pada tabel *database* data penumpang. Kondisi ketiga, ketika perintah print/cetak data penumpang diterima dari Android kondektur, maka ambil data penumpang dari tabel *database* data penumpang, kemudian kirimkan data tersebut pada mikrokontroler. Kondisi keempat, apabila peringatan jumlah penumpang melebihi batas kapasitas maksimum diterima, maka kirim peringatan bahwa jumlah penumpang melebihi batas maksimum pada Android kondektur. Kondisi kelima, apabila peringatan pelanggaran penggunaan kartu *tag* diterima, maka kirim peringatan pelanggaran penggunaan kartu *tag* pada Android kondektur.

3.2.3 Perancangan Perangkat Lunak Aplikasi Android Kondektur

Kondisi diagram alir dari aplikasi Android kondektur dapat dilihat pada Gambar 3.22.



Gambar 3.22 Diagram Alir Pada Aplikasi Android Kondektur

Kondisi pertama, ketika perintah pengisian data penumpang diterima, maka *Pop-up* perintah pengisian data akan muncul, kemudian kondektur akan mengisi data penumpang, setelah data penumpang terisi maka kirim data tersebut pada web manajemen pusat armada bus. Kondisi kedua, apabila kondektur ingin melakukan pencetakan data penumpang, maka pilih data penumpang atau pilih *range* data penumpang dan kirim perintah tersebut pada web manajemen pusat armada bus. Kondisi ketiga, apabila peringatan jumlah penumpang lebih diterima maka tampilkan *Pop-up* Jumlah penumpang melebihi kapasitas maksimum pada Android kondektur. Kondisi keempat, apabila peringatan pelanggaran penggunaan kartu *tag* diterima, maka tampilkan *pop-up* peringatan penggunaan kartu *tag* pada Android kondektur.

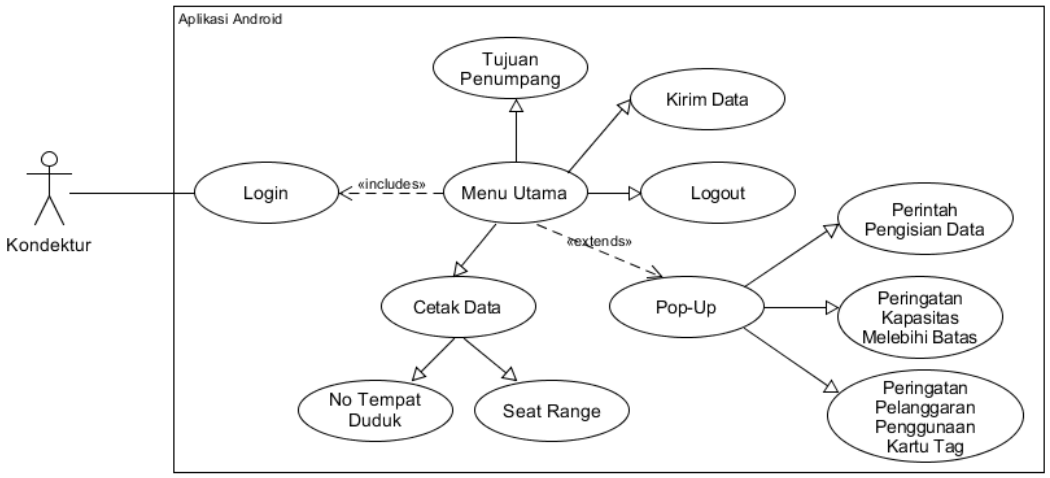
3.3 Perancangan Pemodelan Sistem Aplikasi Android

Pada analisis fungsional ini menggunakan pemodelan UML (Unifiled Modeling Language). Adapun diagram-diagram yang dirancang pada sistem

aplikasi ini diantaranya: Use Case diagram Aplikasi Android, serta Entity Relationship Diagram dan Data Flow Diagram Web.

3.3.1 Use Case Diagram Aplikasi Android Kondektur

Use Case Diagram digunakan untuk mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. Gambar 3.23 menggambarkan Diagram *Use Case* dari program aplikasi Android kondektur.



Gambar 3.23 Use Case Diagram Aplikasi Android Kondektur

Penjelasan *Use Case Diagram* pada Gambar 3.23 akan ditunjukan pada Tabel 3.9 berikut .

Tabel 3.9 Penjelasan Use Case Diagram Aplikasi Android Kondektur

No	Use Case	Deskripsi
1	Login	Proses pengisian *user name* dan *Password* untuk memperoleh hak akses aplikasi untuk membuka menu utama.
2	Menu Utana	Tampilan utama aplikasi.
3	Tujuan Penumpang	Tampilan menu pengisian data penumpang, berupa tujuan penumpang
4	Kirim Data	Menu untuk proses pengiriman data penumpang.
5	Cetak Data	Menu untuk proses pencetakan data penumpang. Dapat sesuai nomor tempat duduk atau sesuai *range* nomor tempat duduk.
7	Logout	Proses untuk keluar dari menu utama aplikasi.

Berikut merupakan skenario yang dilakukan user dalam menjalankan aplikasi serta reaksi sistem terhadap aksi *actor*/user.

- Nama *Use Case*: *Login*

Scenario:

Tabel 3.10 Skenario Login

Aksi Aktor	Reaksi Sistem
Skenario Normal.	
1. *User* membuka aplikasi.	
	1. Aplikasi berjalan tampilkan menu *login.*
2. *Input Username* dan *Password.*	
	2. Aplikasi memverifikasi *Username* dan *Password* terdaftar. 3. Tampilkan Menu Utama.

- Nama *Use Case*: *Menu Utama*

Scenario:

Tabel 3.11 Skenario Menu Utama

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* menunggu tampilan menu utama.	
	2. Aplikasi menampilkan menu utama.

- Nama *Use Case*: *Tujuan Penumpang*

Scenario:

Tabel 3.12 Skenario Tujuan Penumpang

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih *button* tujuan penumpang.	

	1. Aplikasi menampilkan tujuan yang bisa dipilih.
2. *User* memilih tujuan penumpang.	
	2. Aplikasi memverifikasi tujuan kemudian memilih dan menampilkan harga dari tujuan tersebut.

- Nama *Use Case*: Kirim Data

Scenario:

Tabel 3.13 Skenario Kirim Data

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih pengiriman data	
	1. Aplikasi mengirimkan data penumpang pada web

- *Use Case*: Cetak Data

Scenario:

Tabel 3.14 Skenario Cetak Data

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih cetak data	
	1. Aplikasi menampilkan *opsi* pencetakan data sesuai nomor atau *range* tempat duduk.

- Nama *Use Case*: No Tempat Duduk

Scenario:

Tabel 3.15 Skenario No Tempat Duduk

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih nomor tempat duduk.	
	1. Mengirimkan perintah pencetakan sesuai nomor tempat duduk yang dipilih.

- Nama *Use Case*: Seat Range

Scenario:

Tabel 3.16 Skenario Seat Range

Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih pencetakan data sesuai *seat range*	
	1. Aplikasi menampilkan opsi pencetakan sesua range nomor tempat duduk berikut : (1-15), (16-30), (31-45).
2. *User* memilih range yang diinginkan	
	2. Aplikasi melakukan perintah pencetakan data sesuai range nomor tempat duduk kepada web.

- Nama *Use Case*: *Logout*

Scenario:

Tabel 3.17 Skenario Use Case Logout

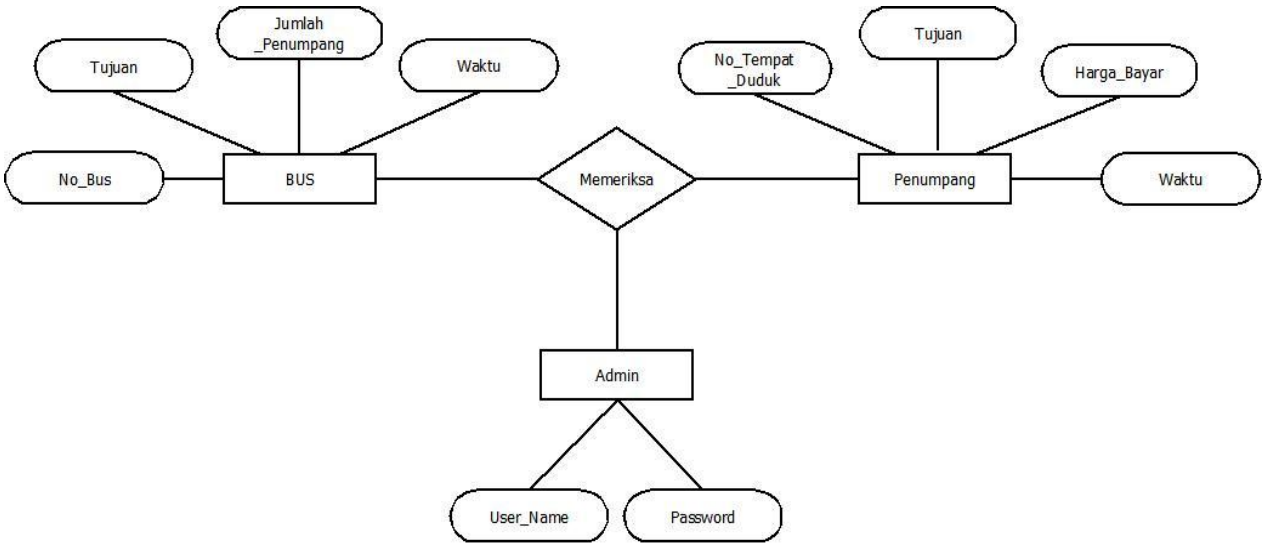
Aksi Aktor	Reaksi Sistem
Skenario Normal	
1. *User* memilih *logout*	
	1. Aplikasi akan keluar dari menu utama.

3.4 Perancangan Pemodelan Sistem Web

Pada perancangan pemodilan sistem web ini digunakan diagram pemodelan *Entity Relationship Diagram* (ERD) dan *Data flow diagram* (DFD).

3.4.1 Entity Relationship Diagram Web

Entity Relationship diagram (ERD) dari web digunakan untuk memodelkan hubungan serta kebutuhan setiap data pada web yang akan dibuat. *Entity Relationship Diagram* web ini dapat dilihat pada Gambar 3.24.



Gambar 3.24 Entity Relationship Diagram Web

Dalam sistem web terdapat 3 entitas utama yang saling berhubungan. Yaitu Admin yang berfungsi sebagi pemeriksa data, Bus berfungsi sebagai tempat penyimpanan data operasional bus, dan Penumpang yang berfungsi sebagai tempat penyimpanan data penumpang. Penjelasan akan dijelaskan pada Tabel 3.18 berikut.

Tabel 3.18 Penjelasan Entity Relationship Diagram Web

No	Entitas	Atribut
1	Admin	Username dan Password
2	Bus	- Tujuan - Jumlah_Penumpang - Nomor_Bus - Waktu
3	Penumpang	- Tujuan - Harga_Bayar - Nomor_Tempat_Duduk - Waktu

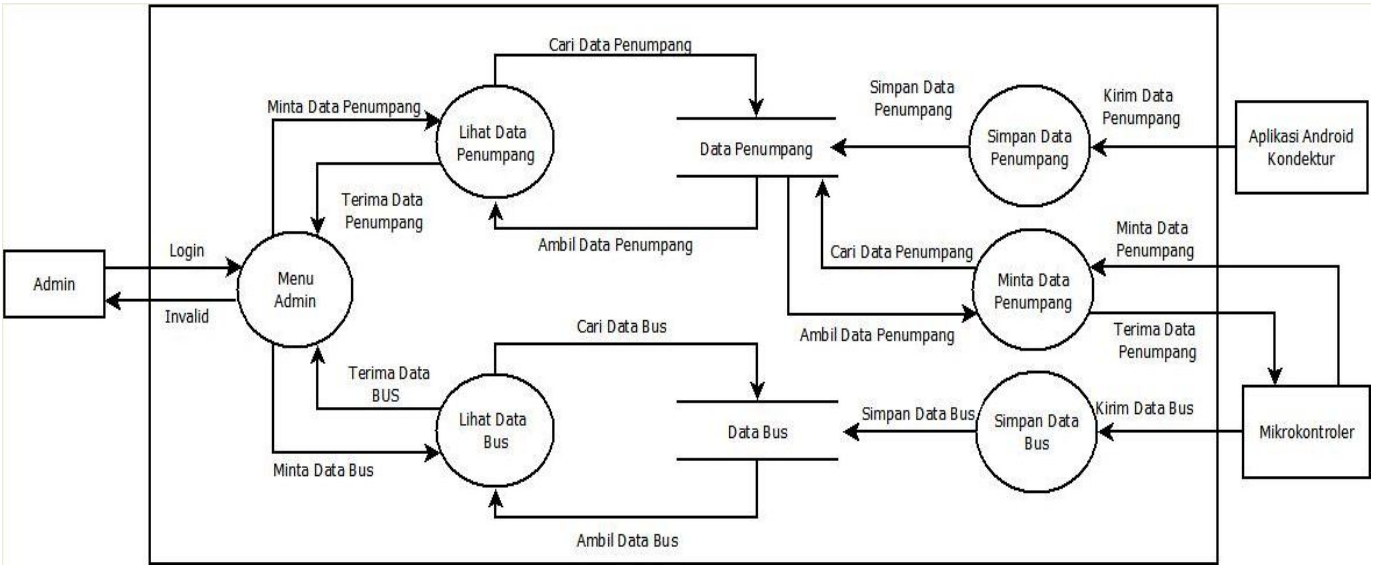
Hubungan relasi dari entitas Admin dengan Bus, Admin dapat memeriksa data bus yang berupa tujuan, jumlah penumpang, nomor Bus dan waktu penerimaan data. Demikian pula relasi Admin dengan Penumpang, dimana Admin dapat memeriksa data penumpang yang berupa tujuan, harga bayar, nomor tempat duduk dan waktu penerimaan data. Penjelasan relasi antar Entitas akan dijelaskan pada Tabel 3.19 berikut.

Tabel 3.19 Relasi Entitas

No	Entitas A	Relasi	Entitas B
1	Admin	Memeriksa	Bus
2	Admin	Memeriksa	Penumpang

3.4.2 Data Flow Diagram Web

Data Flow Diagram (DFD) dari web digunakan untuk memodelkan alur data pada web yang akan dibuat. Data Flow Diagram web ini dapat dilihat pada Gambar 3.25.



Gambar 3.25 Data Flow Diagram Web

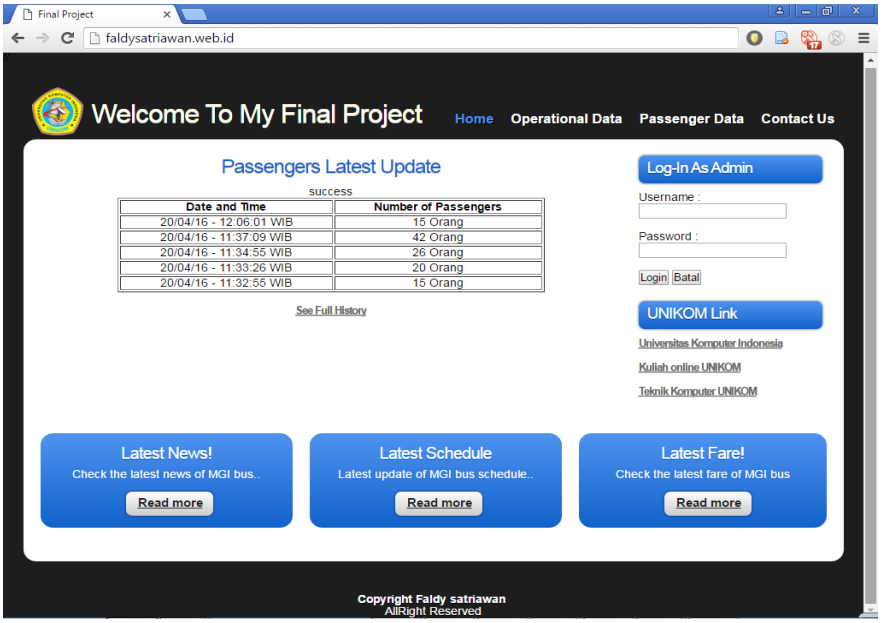
Penjelasan DFD :

1. **Admin :** Admin merupakan pengguna utama dari sistem monitoring pada web yang berjalan. Dimana admin dapat melihat data penumpang maupun data jumlah penumpang dari bus.
   - Proses Lihat Data Penumpang **:** ketika admin meminta data penumpang maka sistem akan melakukan pencarian data yang diinginkan pada *database* data penumpang yang memuat informasi (tujuan, harga, nomor tempat duduk penumpang), kemudian ketika data ditemukan maka sistem akan mengambil dan menampilkan data tersebut pada admin.
   - Proses Lihat Data Bus : ketika admin meminta data bus maka sistem akan melakukan pencarian data yang diinginkan pada *database* data bus yang memuat informasi (jumlah penumpang didalam bus), kemudian ketika data ditemukan maka sistem akan mengambil dan menampilkan data tersebut pada admin.

2. **Aplikasi Android Kondektur :** berfugsi sebagai aplikasi pengisian data penumpang oleh kondektur, data penumpang tersebut berupa data tujuan, harga dan nomor tempat duduk. Ketika kondektur melakukan pengiriman data pada web, maka sistem akan menyimpan data yang diterima pada *database* data penumpang.
3. **Mikrokontroler :** Mikrokontroler berperan sebagai pengirim dan penerima data dari web.
    - Sebagai pengirim : Mikrokontroler akan mengirimkan jumlah penumpang didalam bus pada web, data yang diterima oleh web kemudian akan disimpan pada *database* data bus..
    - Sebagai penerima : Mikrokontroler akan meminta data pada web ketika proses pencetakan data dilakukan, kemudian server akan melakukan pengecekan apakah data penumpang ada pada *database* data penumpang, apabila ada data maka kirim data penumpang tersebut pada mikrokontroler untuk dilakukan pencetakan data penumpang.

3.5 Perancangan Tampilan Web

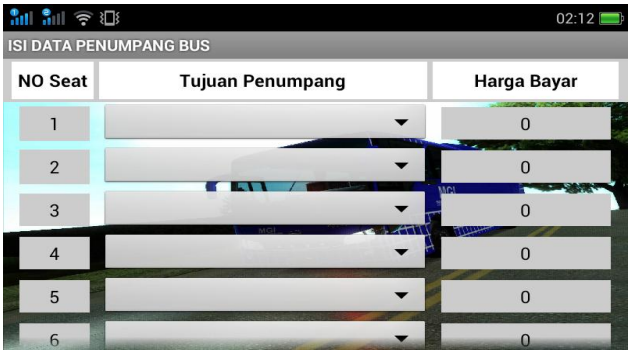
Perancangan web akan dibangung dengan menggunakan bahasa pemrograman php. Dimana *file* php ini akan ditempatkan pada *web hosting* yang berada pada *domain* penulis yaitu faldysatriawan.web.id. Pada tampilan monitoring *website*, parameter-parameter yang akan ditampilkan adalah waktu dan jumlah dari penumpang. Kedua parameter ini akan terus *update* selama selang waktu 1 detik. Gambar 3.26 menunjukan rancangan tampilan dari *website* yang akan dibangun.



Gambar 3.26 Perancangan Tampilan Website

3.6 Perancangan Aplikasi Android

Perancangan aplikasi Android akan dibangung dengan menggunakan program App Inventor. Dimana aplikasi Android ini berfungsi sebagai perangkat lunak pengisian data penumpang yang akan digunakan oleh kondektur. Aplikasi Android ini memiliki beberapa *form* pengisian data, seperti kode bus, nomor tempat duduk penumpang, tujuan penumpang, biaya tujuan penumpang serta total pendapatan dari armada bus itu sendiri. Tampilan dari aplikasi Android ini dapat dilihat pada gambar 3.27.



Gambar 3.27 Perancangan Tampilan Aplikasi Android