BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis dan Percobaan Sistem

Analisis sistem bertujuan untuk mengidentifikasi permasalahan-permasalahan yang terdapat pada sistem serta menentukan kebutuhan-kebutuhan dari sistem yang akan dibangun. Analisis tersebut meliputi analisis masalah, analisis arsitektur sistem, spesifikasi kebutuhan non fungsional, analisis kebutuhan non fungsional, analisis data, dan analisis kebutuhan fungsional.

3.1.1 Analisis Masalah

Pembangunan *early warning security system* mengunakan arduiono berbasis android pada mobil merupakan suatu mekanisme yang dirancang sedemikian rupa yang digunakan untuk mengamankan kendaraan dari pihak yang tidak diinginkan dengan memberikan peringatan dini kepada pemilik kendaraan. Adapun berdasarkan penelitian yang dilakukan terdapat permasalahan yang terjadi sehingga dibuatnya sistem ini karena masih ada celah keamanan pada sistem keamanan mobil.

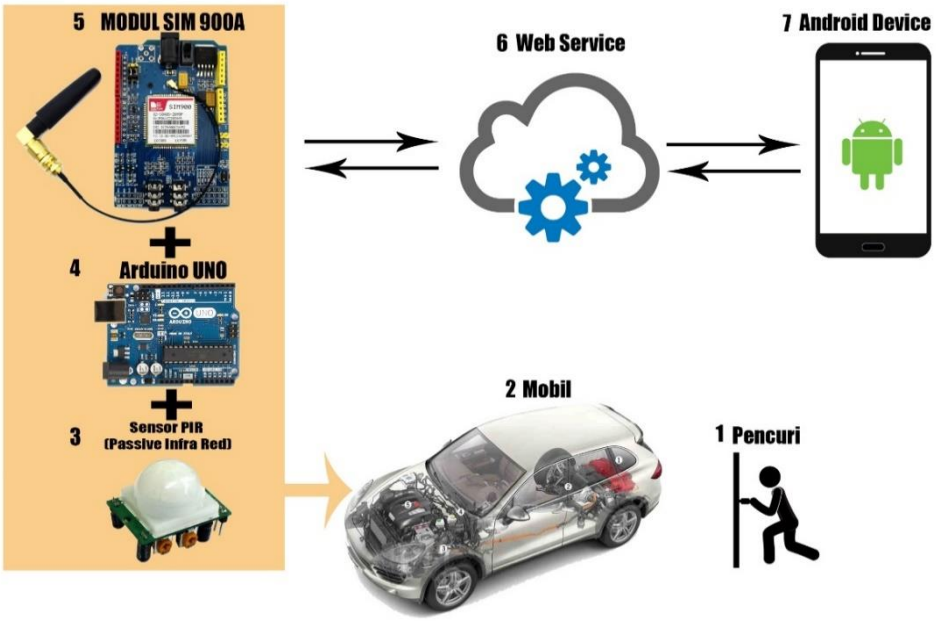
3.1.2 Analisis Sistem Yang Ditawarkan

Untuk mengatasi masalah tersebut maka perlu dibuat suatu sistem yang dapat mendeteksi pergerakan di dalam mobil, dan dapat memberikan informasi melalui perangkat mobile. Pada aplikasi mobile pengendara akan menerima peringatan berupa notifikasi jika terjadi pergerakan di dalam mobil.

3.1.3 Analisis Arsitektur Sistem

Analisis arsitektur sistem adalah tahapan untuk mendapatkan gambaran umum sistem yang akan dibangun. Baik itu gambaran sistem back end maupun

gambaran sistem front end pengguna atau user. Gambar arsitektur sistem dapat dilihat pada Gambar 3.1.



Gambar 3.1 Arsitektur Sistem

Berikut penjelasan dari gambaran arsitektur sistem pembangunan *early warning security system* mengunakan arduiono berbasis android pada mobil :

1. Pencuri berusaha masuk kedalam kendaraan tanpa diketahui oleh pemilik kendaraan .
2. Ketika pencuri berhasil masuk maka *system* akan mendeteksi gerakan dalam kendaraan.
3. Gerakan terdeteksi oleh sensor PIR lalu diteruskan ke Mikrokontroler Arduino.
4. Sinyal pergerakan dari sensor PIR oleh Mikrokontroler Arduino diteruskaan ke SIM900.
5. Sinyal pergerakan oleh SIM900 dikirim ke *webservice.*
6. *Webservice* meneruskan informasi yang dikirim oleh SIM900 *ke Android Device* dengan menggunakan *push notification.*
7. Informasi tentang adanya pergerekan dalam kendaraan ditampil di *Android Device*

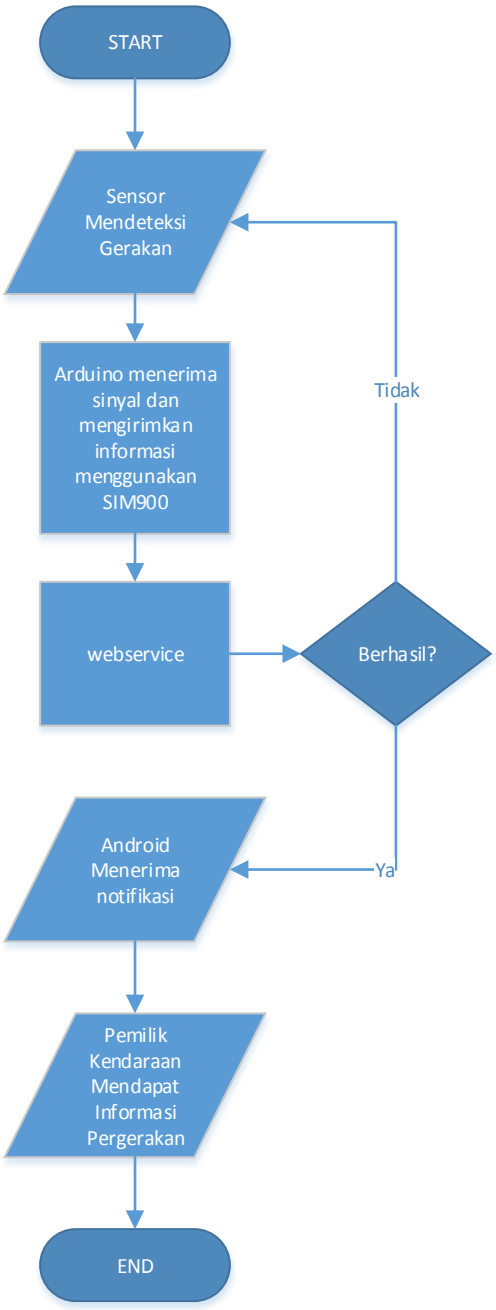
Sistem ini bermula dari sisi pengguna dan akan aktif jika pengguna menyalakan sistem ini dan berstatus aktif maupun sebaliknya melalui aplikasi Android.

Proses ketika pengguna menyalakan sistem ini dan statusnya berubah yang awalnya tidak aktif menjadi aktif yaitu dimana perangkat Android akan mengakses token dari database melalui koneksi Web Service, kemudian *Web Service* akan mereturn kepada perangkat Android berupa bahasa JSON yang akan diterima oleh perangkat Android.

SIM900 pada sistem ini adalah sebagai penghubung antara Arduino UNO dengan jaringan internet, sehingga Arduino UNO bisa terhubung dengan *Web Service* melalui jaringan internet. Pada saat perangkat android mengirimkan perintah mengaktifkan maupun menonaktifkan sistem dengan mengakses token ke *Web Service* maka perintah itu akan diteruskan dari *Web Service* ke perangkat Arduino melalui perangkat SIM900 yang sudah terhubung ke Internet.

Arduino UNO pada sistem ini adalah sebagai alat pemrosessan antara perangkat input dan output, pada sistem ini Arduino UNO akan merespon masukan dari SIM900 dan Arduino UNO akan memproses perintah tersebut untuk mengaktifkan maupun menonaktifkan pada sistem ini.

Setelah sistem diaktifkan dan statusnya adalah aktif, maka pada tahap ini semua komponen pada sistem ini akan berfokus kepada perangkat Sensor PIR (*Passive Infra Red*) yang berfungsi sebagai pendeteksi gerak manusia dengan merespon energi dari pancaran sinar inframerah pasif yang dimiliki oleh setiap benda yang terdeteksi olehnya. Sensor PIR (*Passive Infra Red*) pada sistem ini adalah sebagai trigger atau pemicu dari semua rangkaian pada *Early Watning Security System,* ketika Sensor PIR mendeteksi gerak manusia maka Sensor PIR akan mengirimkan sinyal ke mikrokontroler Arduino UNO. Sebaliknya jika sistem ini nonaktifkan maka fungsi Sensor gerak ini akan dimatikan.



Gambar 3.2 Proses Pendeteksian Gerakan

3.1.4 Analisis Data

Berikut ini merupakan data pada databse mysql *Web Service*, struktur tabel ini akan menjelaskan tentang detail atribut. struktur tabel dapat dilihat pada tabel dibawah ini.

Tabel 3.1 Struktur Tabel

No	Tabel	Atribut
1	tb_user	id : int(11) PK username : varchar(100) password : varchar(8)
2	thistory	id : int(10) PK waktu : datetime status : varchar(255) img_path : varchar(500)
3	pirsensorstatus	id : int(11) PK status1: int(5) status2 : int(5) status3 : int(5))

3.1.5 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak terbagi kedalam dua kebutuhan yaitu kebutuhan non fungsional dan fungsional. Kebutuhan fungsional dapat dilihat pada tabel 3.1.

Tabel 3.2 Spesifikasi Kebutuhan Perangkat Mobile

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-F-001	Subsistem *Mobile* menyediakan fasilitas melihat data History pergerakan yang terdeteksi oleh sensor.
SKPL-F-002	Subsistem *Mobile* menyediakan fasilitas bagi pengguna berupa notifikasi ketika perangkat sensor mendeteksi gerakan.
SKPL-F-003	Subsistem *Mobile* menyediakan fasilitas bagi pengguna berupa berupa tombol on dan off agar system dapat dikontrol.

Kebutuhan non fungsional pada perangkat lunak yang dibangun dapat dilihat pada tabel 3.2.

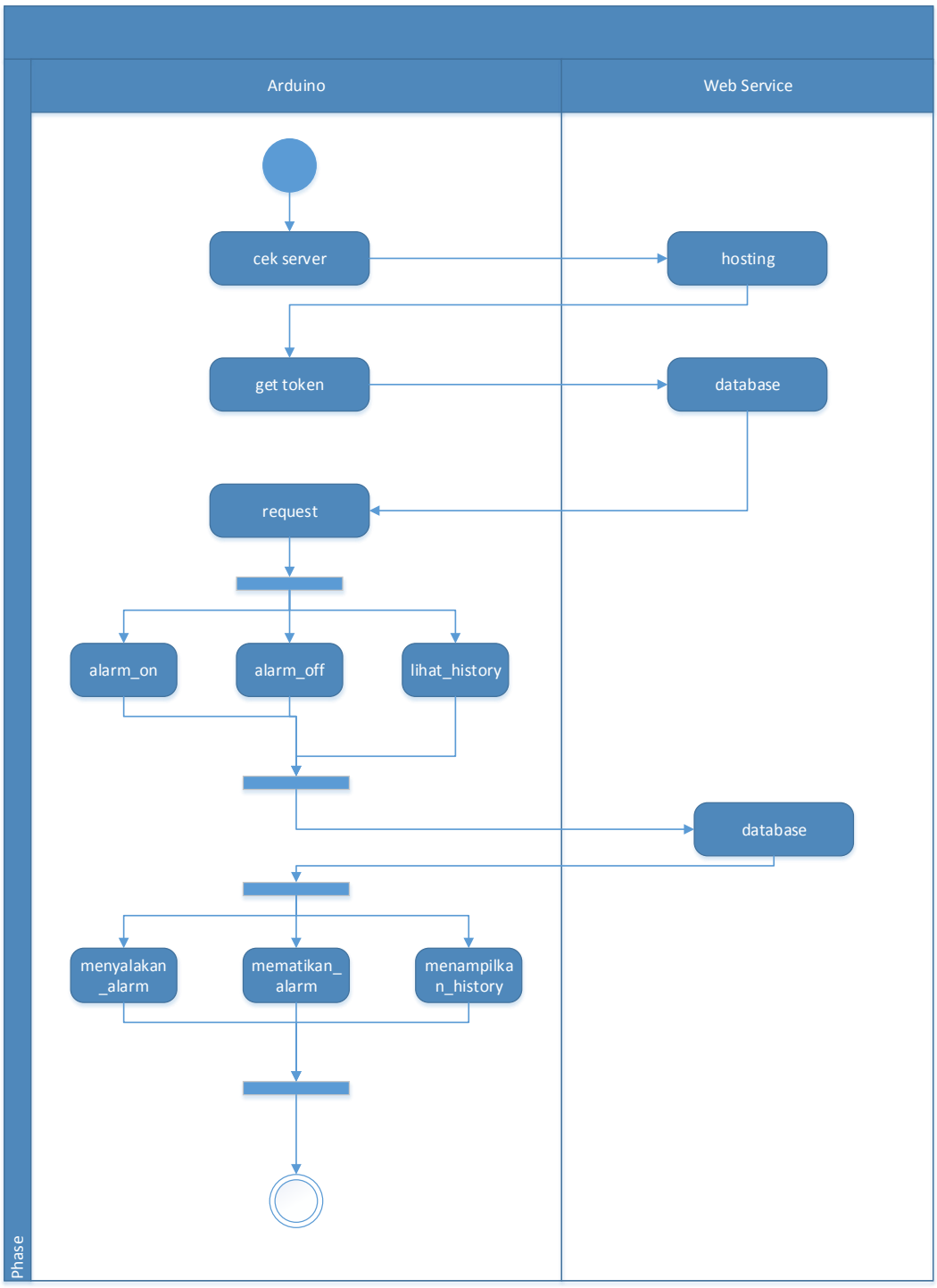
Tabel 3.3 Spesifikasi Kebutuhan Non Fungsional

Kode SKPL	Spesifikasi Kebutuhan Perangkat Lunak
SKPL-NF-001	Sistem yang dibangun menggunakan perangkat *Mobile.*
SKPL-NF-002	Sistem yang dibangun minimal menggunakan sistem operasi android versi 4.0 ICS (Ice Cream Sandwich).
SKPL-NF-003	Sistem yang dibangun membutuhkan *Hardware* Arduino UNO, SIM900, Sensor PIR (*Passive Infra Red*).
SKPL-NF-004	Sistem dibangun dengan spesifikasi *hardware* yang memenuhi standar minimum kebutuhan

3.1.6 Analisis Kerja Sistem Arduino UNO

Arduino berjalan dengan program yang ditanam didalam chip mikrokontroler, dalam penelitian ini arduino terkoneksi pada internet yang dihubungkan melalui SIM900 dan terkoneksi pada server yang didalam nya terdapat web *service.*

Adapun sistem kerja arduino dapat dilihat pada gambar dibawah ini.



Gambar 3.3Alur Kerja Sistem Arduino UNO

3.1.7 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang akan mempengaruhi perilaku pada sistem yang akan dibangun, diantaranya kebutuhan pengguna, analisi kebutuhan perangkat keras dan perangkat lunak sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.1.7.1 Analisis Kebutuhan Pengguna

Analisis kebutuhan pengguna dimaksudkan untuk mengetahui siapa pengguna yang terlibat dalam menjalankan sistem. Hal ini untuk mengetahui karakteristik pengguna dalam mengoprasikan sistem yang akan dibangun.

Tabel 3.4 Karakteristik Pengguna Perangkat Mobil

Pengguna	Hak Akses	Kemampuan yang sudah ada	Kemampuan yang diperlukan
Pengendara Mobil (Pengguna Smartphone)	Mengelola Pengaturan *system*, mengelola notifikasi	Mengerti penggunaan smartphone berbasis Android	Mengerti penggunaan smartphone berbasis Android

Berdasarkan Tabel 3.3 bahwa pengguna smartphone yaitu kendaraan telah memenuhi kriteria yang diperlukan agar sistem ini bisa digunakan dengan baik.

3.1.7.2 Analisis Kebutuhan Perangkat Keras

Perangkat keras yang akan digunakan berdasarkan kebutuhan minimal yang haru terpenuhi yaitu :

1) Perangkat keras *Mobile*

Spesifikasi minimum pada sebuah perangkat smartphone dengan sistem operasi Android. Kebutuhan perangkat keras untuk *smartphone* ditunjukan pada Tabel 3.5.

Tabel 3.5 Kebutuhan Minimum Perangkat Keras Mobile

Perangkat Keras	Kebutuhan Minimal
Processor	800 Mhz
RAM	512 MB
Memory Internal	1 GB
Konektifitas	Koneksi Internet

2) Perangkat keras Arduino UNO

Sistem ini membutuhkan mikrokontroler Arduino UNO sebagai alat pemrosessan antara perangkat input dan output. Spesifikasi perangkat keras Arduino UNO ditunjukan pada tabel 3.6.

Tabel 3.6 Spesifikasi Perangkat Keras Arduino UNO

Perangkat	Spesifikasi
Mikrokontroler	ATmega 328
Tegangan Pengoperasian	5 V
Tegangan Input yang disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah pin I/O Digital	14 pin digital (6 diantaranya menyediakan keluaran PWM)
Jumlah pin input Analog	6 pin
Arud DC tiap pin I/O	40mA
Arus DC untuk pin 3.3 V	50mA
Memori Flash	32 KB (ATmega 328) sekitar 0,5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
Clock Speed	16 MHz

3) Perangkat keras SIM900

Sistem ini membutuhkan SIM900 yang berfungsi sebagai penghubung antara arduino board dengan jaringan internet. Adapun spesifikaksi SIM900 yaitu sebagai berikut :

1. *Quad-Band* 850/900/1800/1900MHz
2. *GPRS multi-slot class* 10/8
3. *GPRS mobile station class* B
4. *Class 4* (2W@850/900MHz)
5. *Control via commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)*
6. *Short message service* (Pesan Singkat)
7. *Free serial port selection*
8. Tombol *power on/off* dan *reset* didukung oleh *Arduino interface*

4) Perangkat keras Sensor PIR (Passive Infra Red)

Sistem ini membutuhkan Sensor PIR (*Passive Infra Red*) yang berfungsi sebagai alat pendeteksi gerak (manusia). Adapun spesifikasi Sensor PIR (*Passive Infra Red*) ditunjukan pada tabel 3.7

Tabel 3.7 Spesifikasi Perangkat Keras Sensor PIR

Perangkat	Spesifikasi
Radius	120 derajat
Jarak deteksi	5 – 7 meter
Dimensi	3.2 cm x 2.4 cm x 2.3 cm
Rated voltage	3V – 20V
Time delay	5 – 200 detik (dapat diatur)
Work temperature	-20 ‘C - + 60 ‘C
Calibrating Time	20-60 detik

3.1.7.3 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan software. Kebutuhan minimal spesifikasi

perangkat lunak pada perangkat *smartphone* yang akan menggunakan perangkat lunak ini dapat dilihat pada Tabel 3.8.

Tabel 3.8 Spesifikasi Minimum Sofware

Jenis Sistem Operasi Mobile	Versi
Android	Sistem Operasi Android 4.0 Ice Cream Sandwich

Sedangkan dari sisi *developer*, perangkat lunak yang digunakan untuk membangun sistem ini adalah sebagai berikut:

1. Subsistem *Mobile*
   a. Sistem operasi Microsoft Windows 10 Professional.
   b. JDK (*Java Development Kit*) versi 7.
   c. Android Studio versi 1.5.1.

3.1.8 Analisis Kebutuhan Fungsional

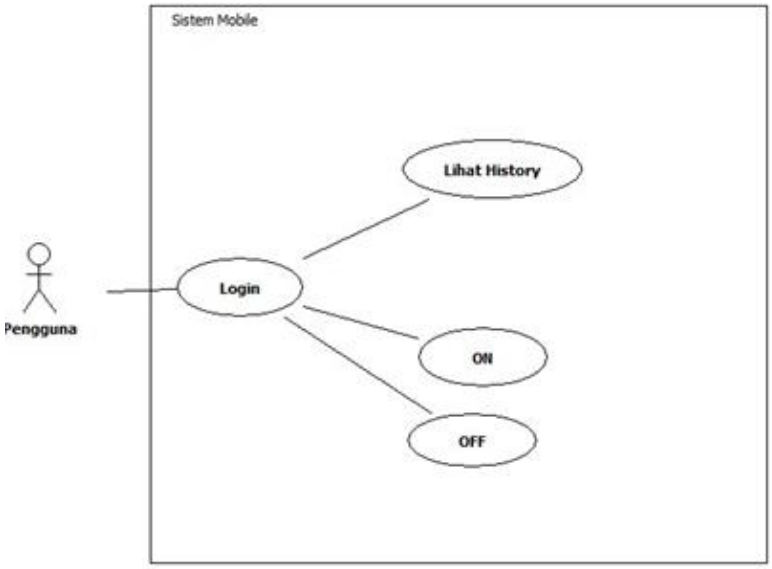
Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan. Analisis kebutuhan fungsional ini meliputi analisis kebutuhan sistem, analisis kebutuhan data, spesifikasi sistem, dan pemodelan sistem.

Pemodelan sistem dilakukan menggunakan UML (Unified Modelling Language) dimana tahap-tahap pemodelan dalam analisis tersebut antara lain use case diagram, identifikasi use case, scenario use case, activity diagram, sequence diagram dan class diagram.

3.1.8.1 Use Case Diagram

Use case diagram merupakan bagian dari fungsionalitas yang dimiliki sistem yang akan menggambarkan bagaimana seseorang atau aktor akan

menggunakan dan memanfaatkan sistem ini. Diagram ini juga mendeskripsikan apa yang akan dilakukan oleh sistem. *Use case Diagram early warning security system* mengunakan arduiono berbasis android pada mobil dapat dilihat pada Gambar 3.4.



Gambar 3.4 Use Case Diagram Mobile

1. Deskripsi Aktor

Aktor adalah abstraksi dari orang dan sistem yang mengaktifkan fungsi dari target sistem. Berdasarkan *Use Case Diagram* pada Gambar 3.3, deskripsi aktor dapat dilihat pada Tabel 3.9.

Tabel 3.9 Deskripsi Aktor

No.	Aktor	Deskripsi
1.	Pengguna	Merupakan aktor yang dapat berperan sebagai orang yang memakai aplikasi mobile.

2. Deskripsi Use Case

Deskripsi *Use Case* didalam sistem dapat dilihat pada Tabel 3.10

Tabel 3.10 Deskripsi Use Case

No.	*Use Case*	Deskripsi
1.	Lihat *History*	Fungsionalitas ini digunakan oleh pengguna untuk melihat data history dari sistem yang mendeteksi pergerakan sebelumnya.

No.	*Use Case*	Deskripsi
3.	*ON*	Fungsionalitas ini digunakan untuk mengaktifkan status Alarm menjadi aktif.
4.	*OFF*	Fungsionalitas ini digunakan untuk mengaktifkan status Alarm menjadi non aktif.

3.1.8.2 Use Case Skenario

Skenario proses-proses yang terdapat didalam use case dapat dilihat pada tabel-tabel berikut :

Tabel 3.11 Skenario Use Case Lihat History

Use Case Name	Lihat *History*	
Goal In Context	Menampilkan data History deteksi pergerakan	
Precoditions	*Pengguna telah membuka halaman utama*	
Succesful End Condition	Berhasil menampilkan data History	
Failed End Condition	Gagal menampilkan data history	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna menyentuh tombol Lihat History	
Main Flow	*Step*	*Action*
	1	Pengguna menyentuh tombol Lihat History
	2	Sistem mengambil data History dari web service
	3	Sistem menampilkan data History
Extensions	*Step*	*Branching Action*
	2.1	Sistem menampilkan pesan bahwa data history tidak bisa ditampilkan

Tabel 3.12 Skenario Use Case ON

Use Case Name	ON
Goal In Context	Mengaktifkan status Alarm
Precoditions	*Pengguna telah membuka Pengaturan Alarm*

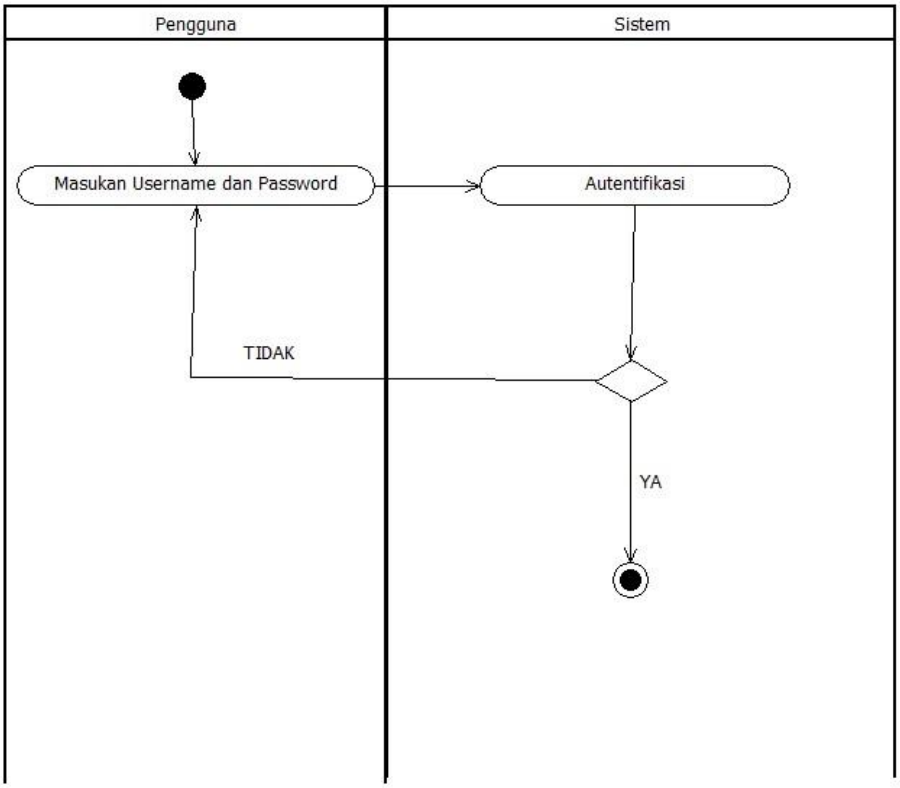
Succesful End Condition	Berhasil mengaktifkan status Alarm	
Failed End Condition	Gagal mengaktifkan status Alarm	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna menyentuh tombol ON	
Main Flow	*Step*	*Action*
	1	Pengguna menyentuh tombol ON
	2	Sistem mengirim data ke *Web Service*
	3	Sistem mengaktifkan status Alarm
Extensions	*Step*	*Branching Action*
	2.1	Sistem gagal mengaktifkan status Alarm

Tabel 3.13 Skenario Use Case OFF

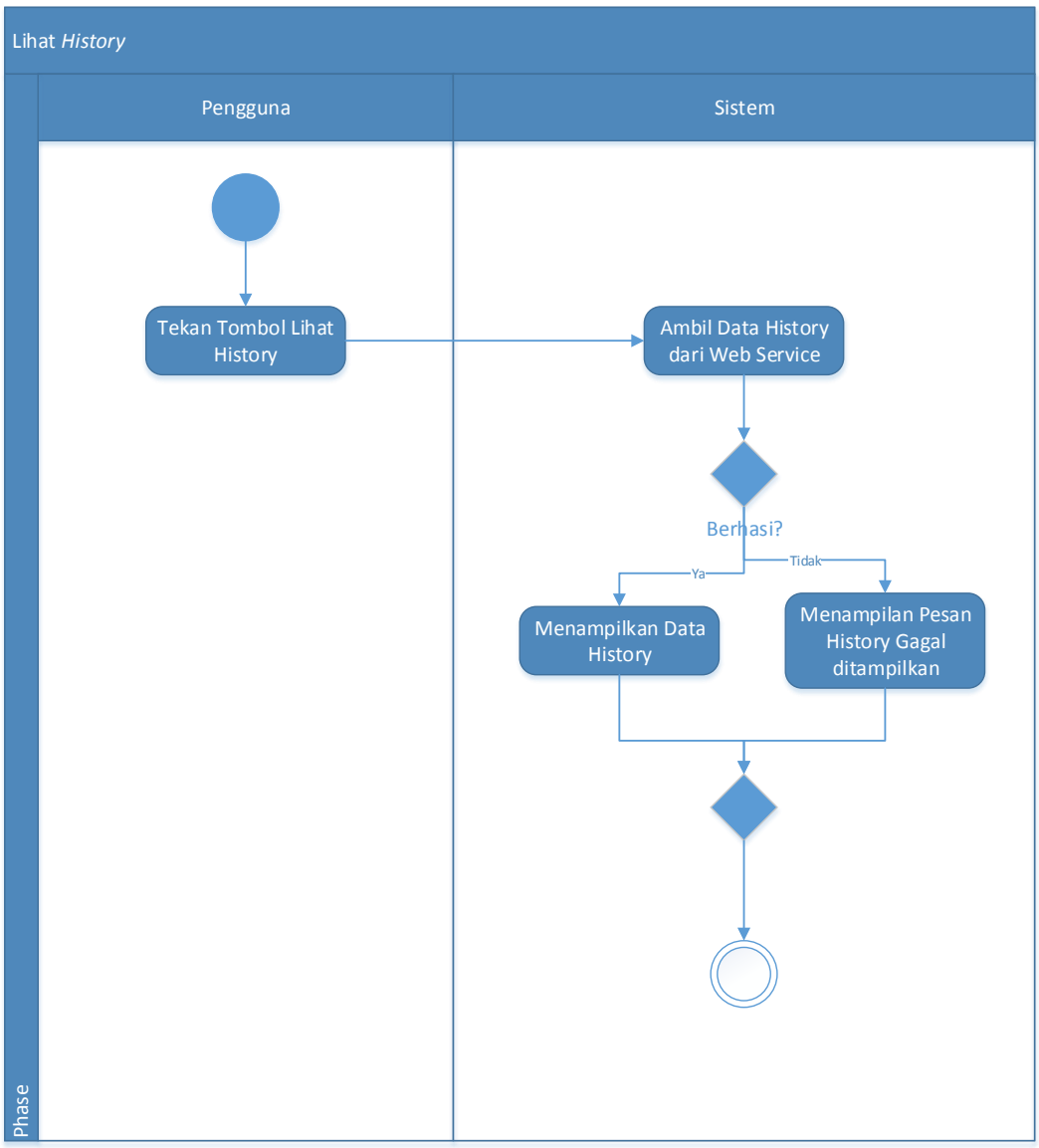
Use Case Name	OFF	
Goal In Context	Menonaktifkan status Alarm	
Precoditions	*Pengguna telah membuka Pengaturan Alarm*	
Succesful End Condition	Berhasil Menongaktifkan status Alarm	
Failed End Condition	Gagal Menongaktifkan status Alarm	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna menyentuh tombol OFF	
Main Flow	*Step*	*Action*
	1	Pengguna menyentuh tombol OFF
	2	Sistem mengirim data ke *Web Service*
	3	Sistem menonaktifkan status Alarm
Extensions	*Step*	*Branching Action*
	2.1	Sistem gagal menongaktifkan status Alarm

3.1.8.3 Activity Diagram

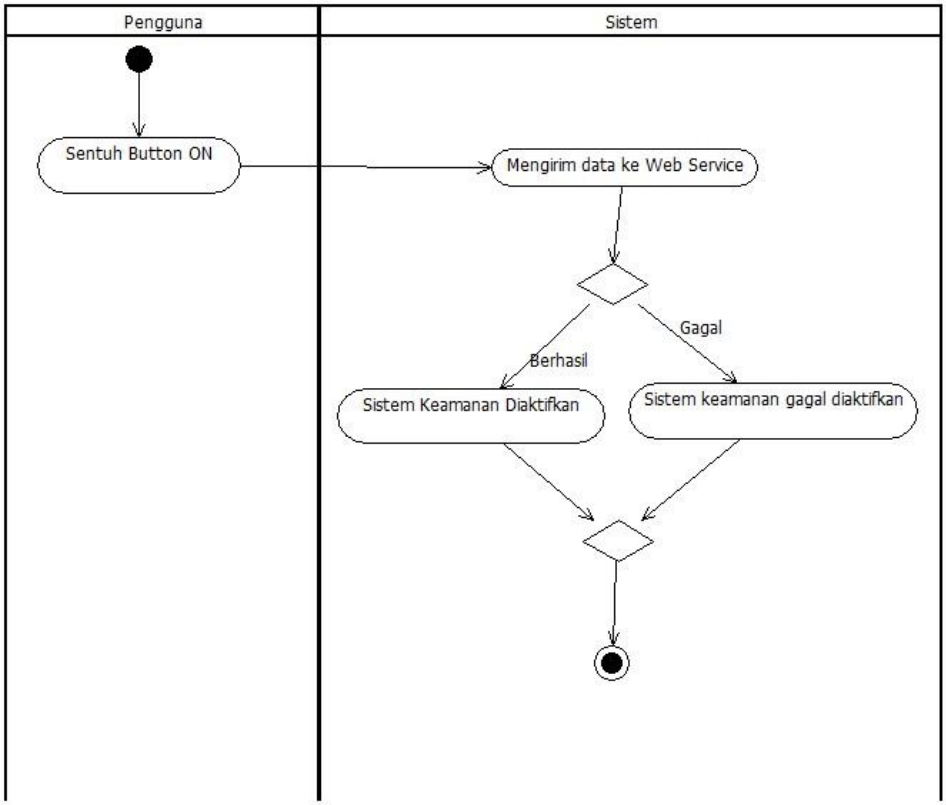
Activity diagram merupakan bagian dari penggambaran sistem secara fungsional menjelaskan proses-proses logika atau fungsi yang terimplementasi oleh kode program. *Activity diagram* memodelkan *event-event* yang terjadi didalam suatu *use case* dan digunakan untuk pemodelan aspek dinamis dari sistem. *Activity diagram* dari pembangunan *early warning security system* mengunakan arduiono berbasis android pada mobil ini dibuat berdasarkan beberapa *use case* yang dapat dilihat pada gambar sebagai berikut:



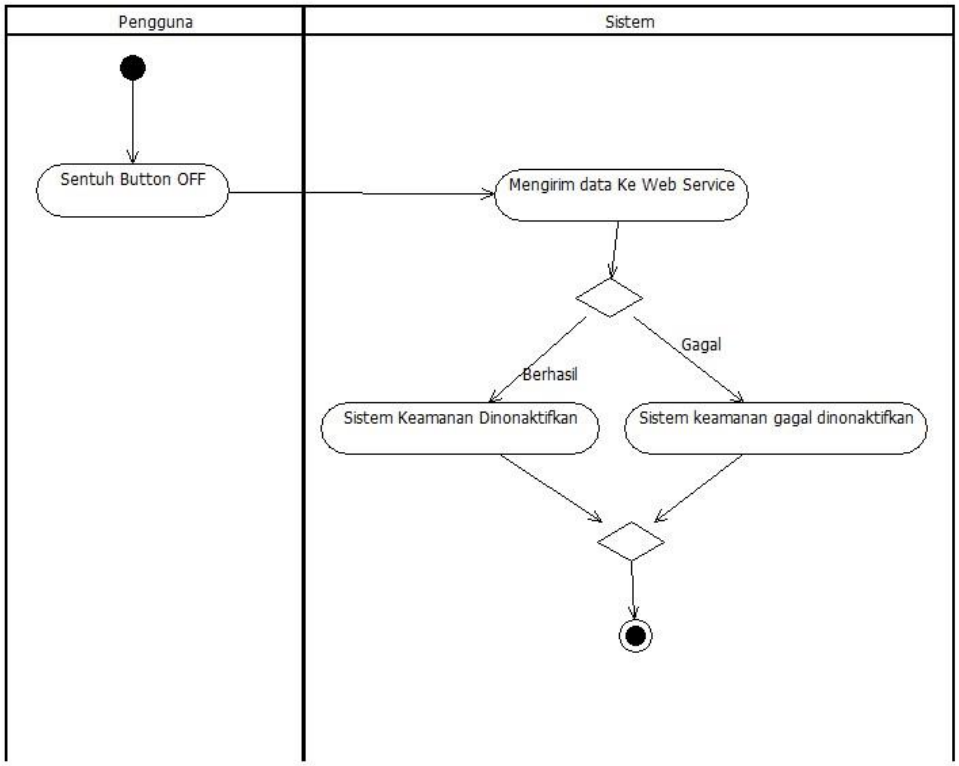
Gambar 3.5 *Activity Diagram* Login



Gambar 3.6 Activiy Diagram Lihat History



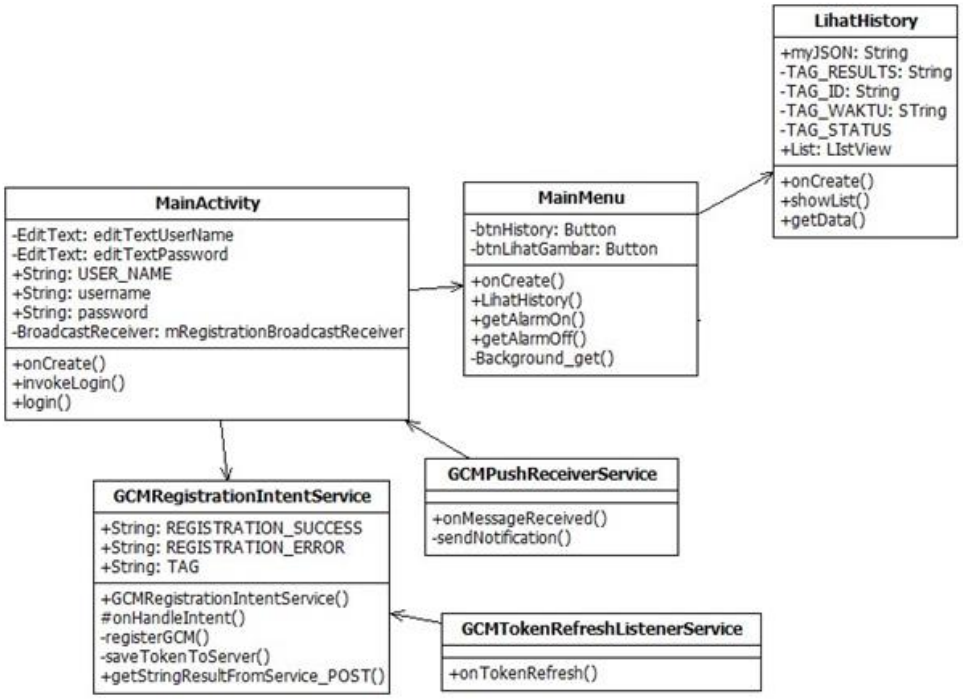
Gambar 3.7 Activity Diagram PIR Sensor ON



Gambar 3.8 Activity Diagram PIR Sensor OFF

3.1.8.4 Class Diagram

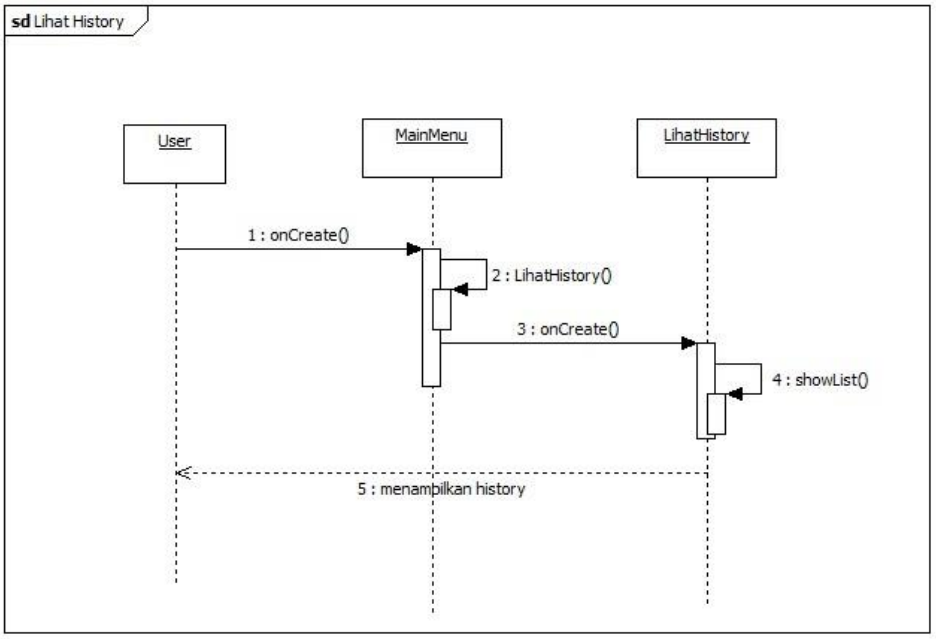
Class Diagram adalah sebuah spesifikasi dari fungionalitas yang menghasilkan objek dan merupakan inti dari pengembangan aplikasi ini. *Class diagram* dari *early warning security system* menggunakan arduino berbasis android pada *mobil*. Untuk lebih jelasnya dapat dilihat pada dibawah ini.



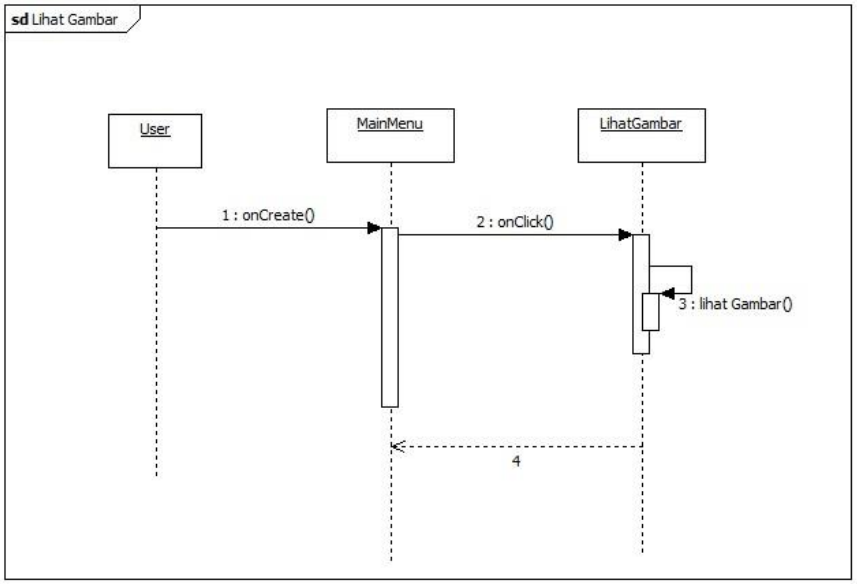
Gambar 3.9 Class Diagram

3.1.8.5 Sequence Diagram

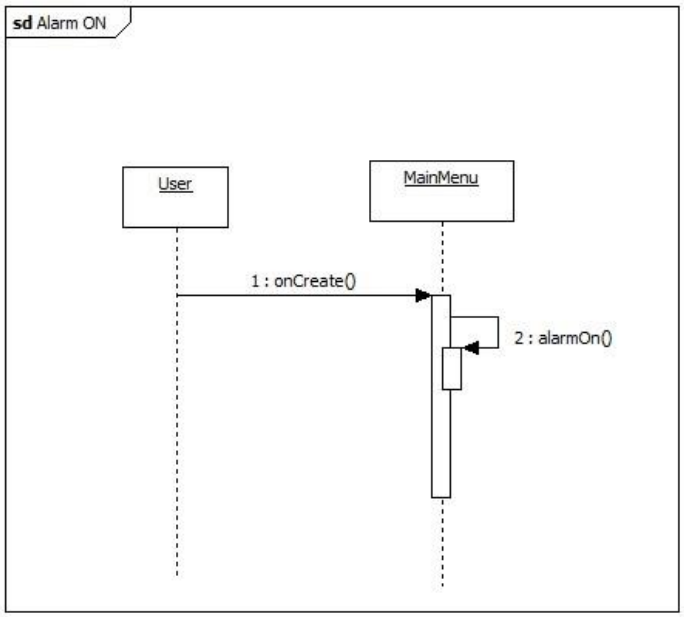
Sequance diagram digunakan untuk menggambarkan arus pekerjaan, pesan yang disampaikan, dan bagaimana elemen-elemen di dalamnya bekerja sama dari waktu ke waktu untuk mencapai suatu hasil. *Sequence diagram* dari sistem yang akan dibangun dapat dilihat pada gambar-gambar berikut.



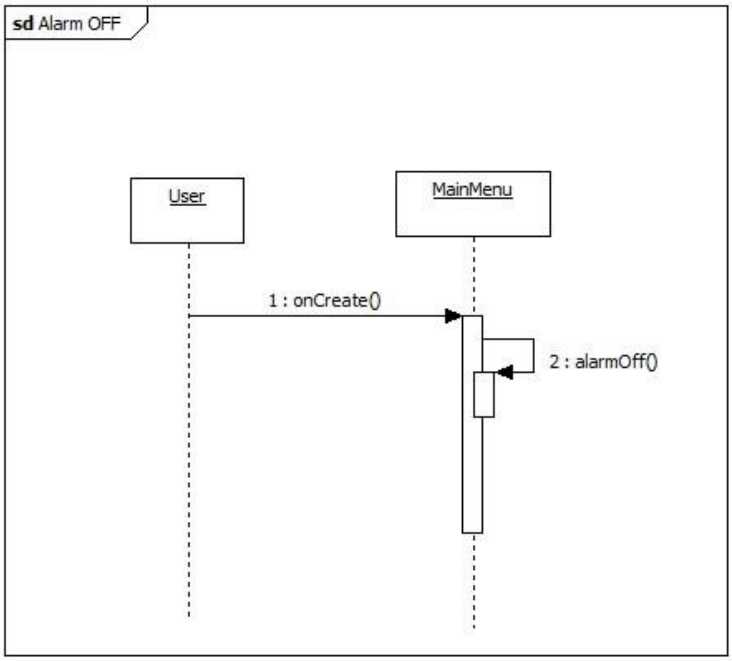
Gambar 3.10 Sequance Diagram Lihat History



Gambar 3.11 Sequance Diagram Lihat Gambar



Gambar 3.12 Sequance Diagram ON



Gambar 3.13 Sequance Diagram OFF

3.1.9 Analisis Komunikasi Perangkat

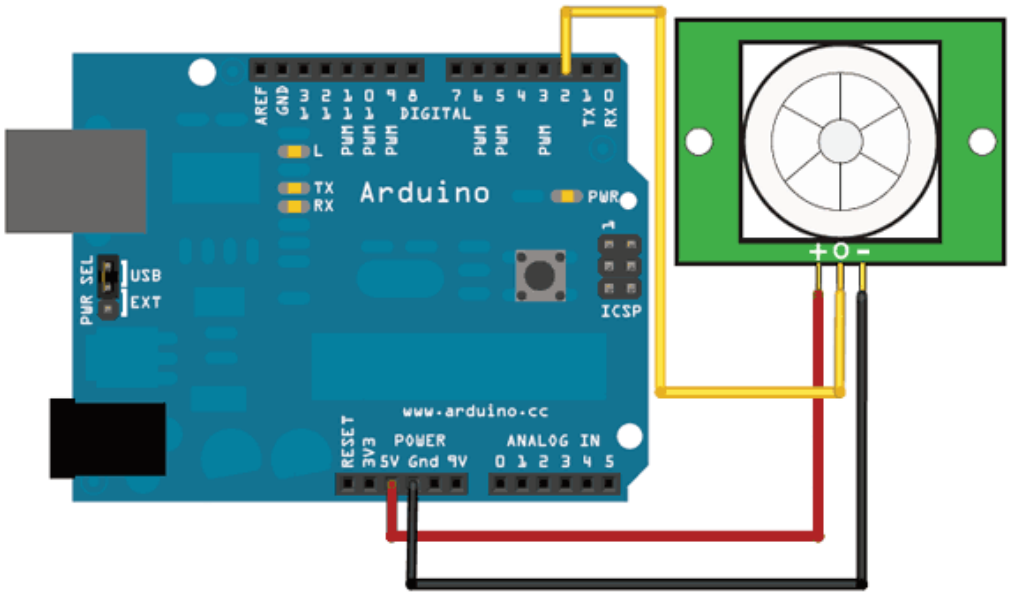
Percobaaan yang sudah dilakukan sebelum melakukan implementasi adalah melakukan percobaan pendeteksian gerakan menggunakan modul sensor PIR dan SIM900, percobaan komunikasi antara sensor PIR dengan Arduino dan SIM900 dengan Web Service.

3.1.9.1 Analisis Pendeteksian Gerakan Menggunakan Sensor PIR

Pada percobaan ini, digunakan 1 buah perangkat Arduino UNO dan satu buah modul sensor PIR. Modul sensor PIR dihubungkan Arduino UNO dengan menggunakan kabel *jumper*. Modul sensor PIR akan medeteksi gerakan berdasarkan pancaran *infra red* yang dihasilkan oleh manusia yang melewati bagian pendeteksi pada modul.

Untuk menjalankan fungsi pendeteksian gerakan pada modul sensor PIR, digunakan *library* PIR Sensor pada program yang dibuat. Program dalam percobaan ini didapatkan dari program *example* yang disediakan oleh *library* PIR

Sensor. Arduino UNO akan menjalankan fungsi pendeteksian gerakan pada modul Sensor PIR. Setiap kali pendeteksian gerakan dilakukan, Arduino UNO akan menampilkan data waktu gerakan ke layar *serial monitor* miliknya. Gambar 3.8 menunjukkan skema penggunakan sensor PIR pada Arduino UNO.



Gambar 3.14 Skema Penggunaan Sensor PIR dengan Arduino UNO

Berikut adalah scenario bagaimana percobaan dilakukan:

1. Arduino menjalankan fungsi modul sensor PIR untuk melakukan pendeteksian gerakan
2. Arduino menampilkan data waktu pergerakan pada *serial monitor*.
3. Arduino akan menampilkan *delay* selama 30 detik sebelum melakukan poin 1. Hal ini dikarenakan modul Sensor PIR membutuhkan waktu 30 detik untuk melakukan kalibrasi agar pedeteksian dapat berjalan dengan optimal

sketch_feb25a

```
//VARS
//the time we give the sensor to calibrate (10-60 secs according to the datasheet)
int calibrationTime = 30;

//the time when the sensor outputs a low impulse
long unsigned int lowIn;

//the amount of milliseconds the sensor has to be low
//before we assume all motion has stopped
long unsigned int pause = 5000;

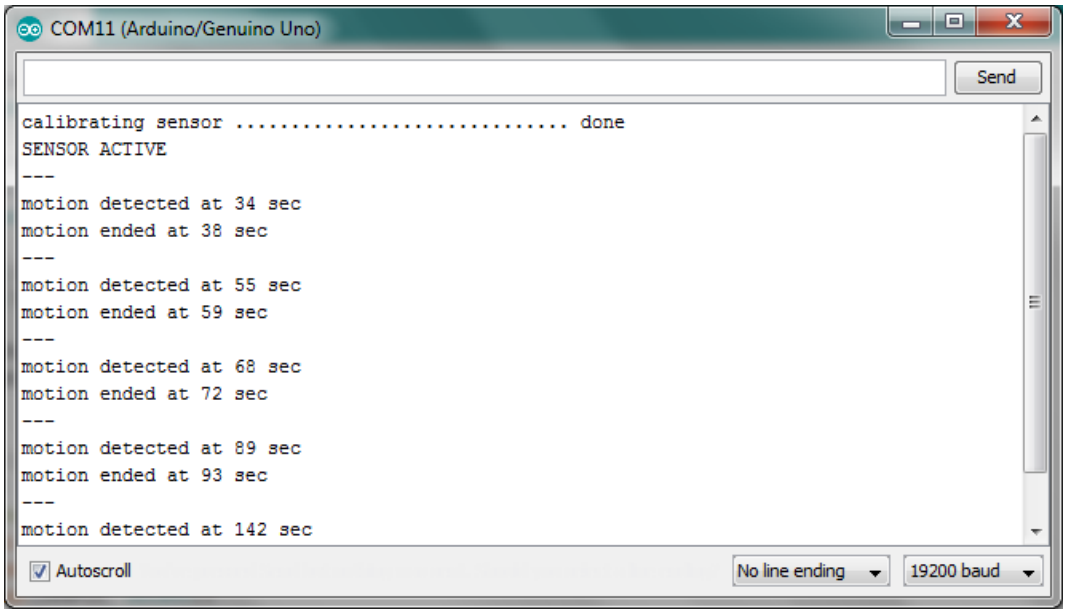
boolean lockLow = true;
boolean takeLowTime;

int pirPin = 3;    //the digital pin connected to the PIR sensor's output
int ledPin = 13;


////////////////////////////
//SETUP
void setup(){
  Serial.begin(19200);
  pinMode(pirPin, INPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(pirPin, LOW);

  //give the sensor some time to calibrate
  Serial.print("calibrating sensor ");
```

Gambar 3.15 Kode Program PIR

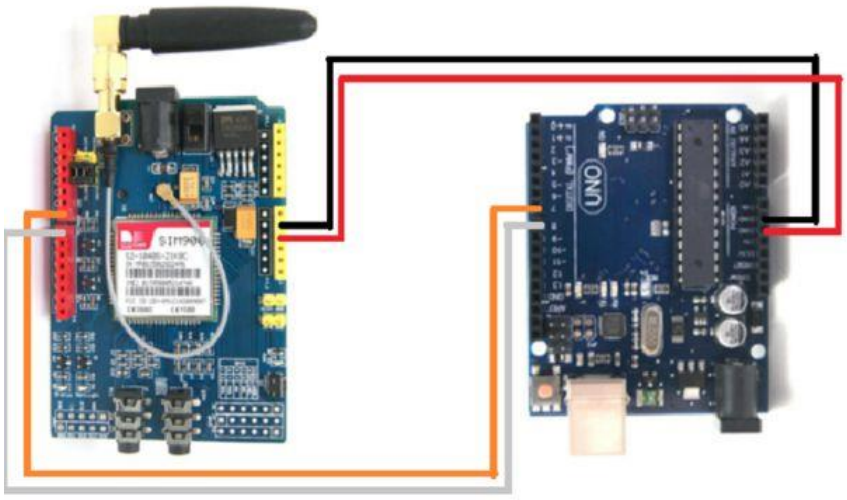


Gambar 3.16 Hasil Pendeteksian Gerakan menggunakan Modul Sensor PIR

Gambar 3.10 merupakan hasil percobaan pendeteksian suhu menggunakan modul Sensor PIR pada Arduino UNO. Ditunjukkan bahwa pada *serial monitor* Arduino ditampilkan waktu pergerakan yang dideteksi menggunakan modul Sensor PIR. Hal tersebut menunjukkan bahwa percobaan pendeteksian gerakan menggunakan modul Sensor PIR pada Arduino telah berhasil dilakukan.

3.1.9.2 Analisis Komunikasi Antara SIM900 dengan Web Service

Pada percobaan ini, digunakan satu buah Arduino Uno dan satu buah modul SIM900. Arduino uno dihubungkan dengan modul SIM900. Gambar 3.10 menunjukkan skema hubungan Arduino UNO dan modul SIM900.



Gambar 3.17 Skema Hubungan Arduino UNO dan Modul SIM900

Sumber Gambar : http://cdn.instructables.com

Berikut adalah scenario bagaimana percobaan dilakukan:

1. Modul SIM900 mengkonfigurasi Port Software Serial yang terdapat pada PIN 7 dan 8 yang sudah terhubung.

2. Setelah Modul Sim900 dihidupkan maka akan delay selama 30 detik agar Modul Sim900 terhubung ke jaringan GSM.
3. Modul SIM 900 akan menjalankan *method SubmitHttpRequest* untuk terhubung dengan webservice.

KoneksiGPRS

```
#include <SoftwareSerial.h>

SoftwareSerial SIM900(7, 8); //Konfigurasi Port Software Serial

void setup() {

  SIM900.begin(19200);
  Serial.begin(19200);
  Serial.println("power up");
  delay(30000);

}

void loop()
{

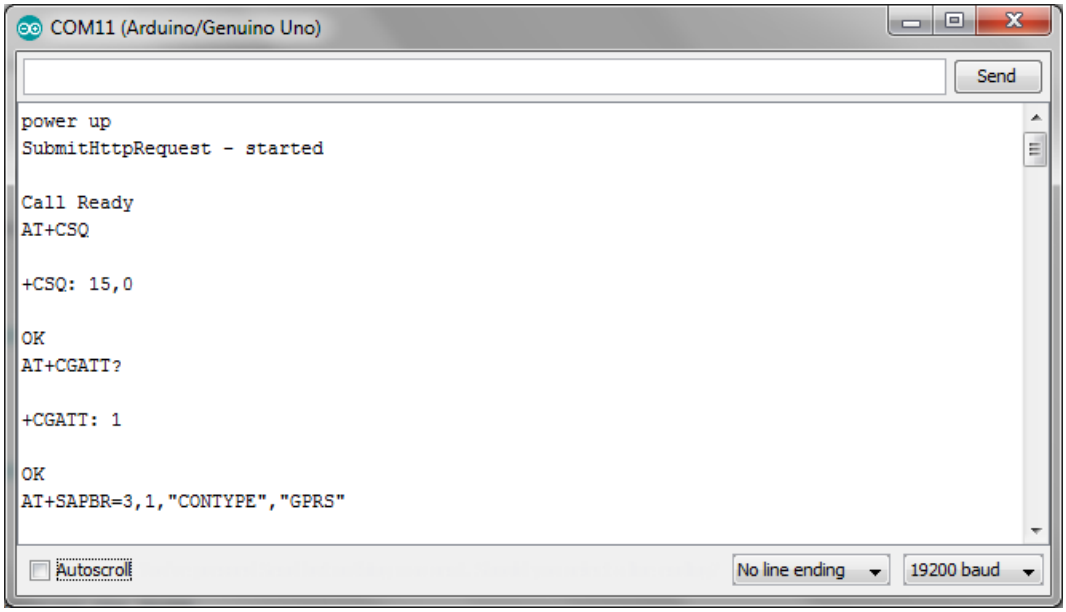
  Serial.println("SubmitHttpRequest - started" );
  SubmitHttpRequest();
  Serial.println("SubmitHttpRequest - finished" );
  delay(10000);

}

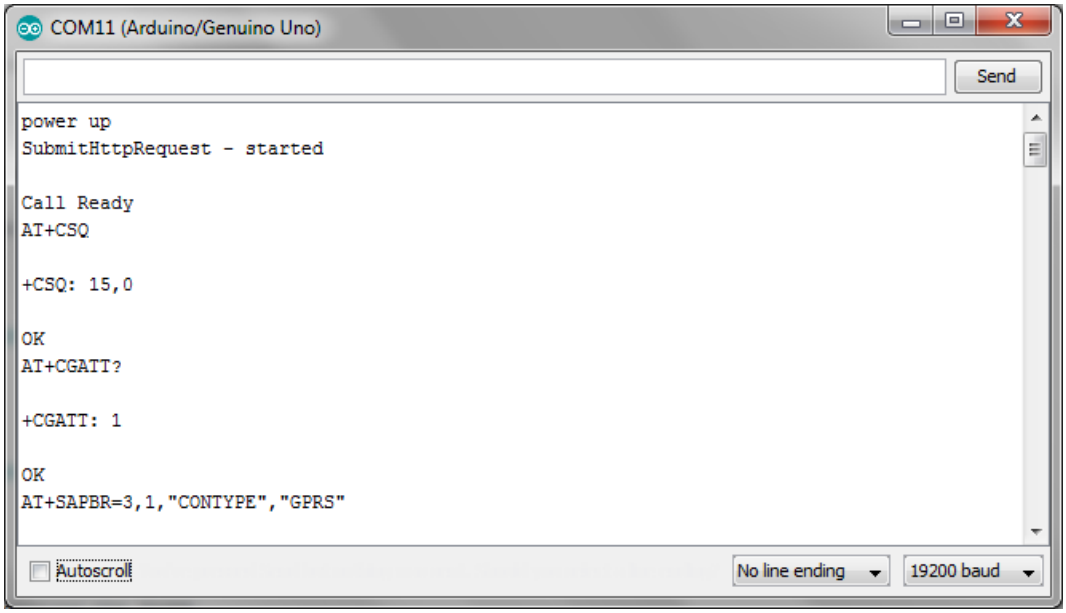
void SubmitHttpRequest()
{

  SIM900.println("AT+CSQ"); // Cek Kualitas Sinyal
```

Gambar 3.18 Kode Program Koneksi GPRS



Gambar 3.19 Percobaan Koneksi Modul SIM900 dengan Web-Service



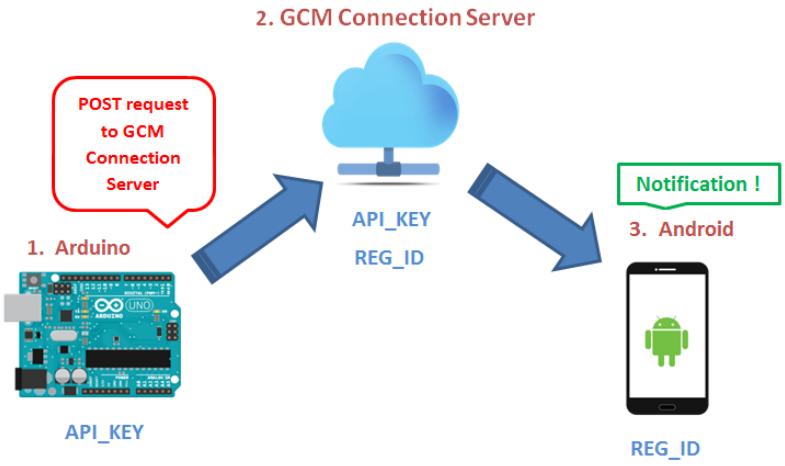
Gambar 3.20 Hasil Percobaan Koneksi Modul SIM900 dengan Web-Service

Gambar 3.13 dan Gambar 3.14 merupakan hasil percobaan koneksi menggunakan modul SIM900 pada Arduino UNO. Ditunjukkan bahwa pada *serial monitor* Arduino ditampilkan proses koneksi ke Web Service menggunakan modul SIM900.

Hal tersebut menunjukkan bahwa percobaan komunikasi dengan webservice menggunakan modul SIM900 pada Arduino telah berhasil dilakukan.

3.1.9.3 Google Cloud Messaging

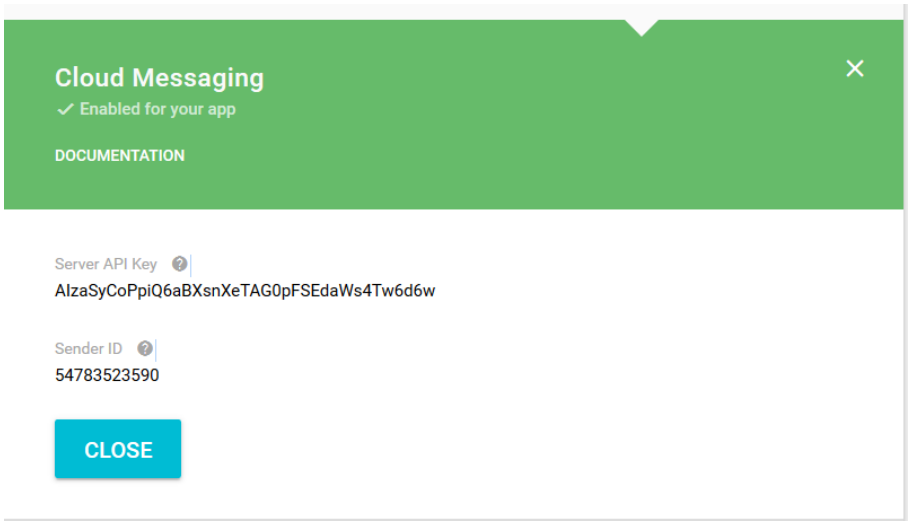
Pada tahap ini akan menjelaskan tentang gambaran proses pengiriman dan penerimaan pesan berupa notifikasi antara perangkat *Mikrokontroler*, *Google Cloud Messaging*, dan *Android*.



Gambar 3.21 Proses Notifikasi

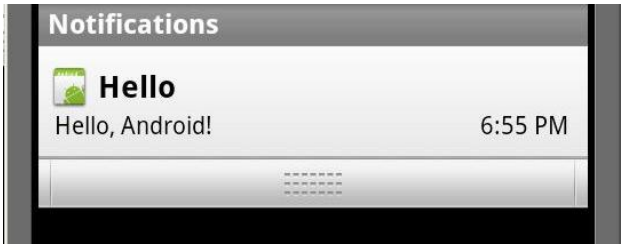
Proses Notifikasi sebagai berikut :

1. *Arduino* sebagai triger dari proses pengiriman notifikasi melalui *Google Cloud Messaging* yaitu dengan mengirimkan mengirimkan token yang sebelumnya telah didaftarkan ke server GCM.



Gambar 3.22 API Key

2. *Google Cloud Messaging* menerima request dari *Arduino* lalu meresponndan kemudian mengirim pesan atau notification ke *Android* melalui REG_ID yang telah terdaftar.
3. *Android* menerima *Notification* yang dikirim oleh server GCM.



Gambar 3.23 Contoh Notifikasi

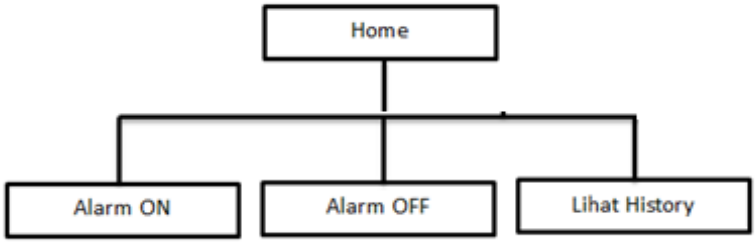
3.1.10 Perancangan Sistem

Di dalam perancangan sistem ini, hal yang akan dibahas adalah perancangan struktur menu, perancangan antarmuka, dan jaringan semantik untuk masing-masing subsistem.

3.1.10.1 Perancangan Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi sehingga aplikasi yang dibangun mudah dipahami dan mudah digunakan. Struktur menu yang terdapat pada pembangunan *early warning security system*

menggunakan Arduino berbasis Android pada *mobil* dapat dilihat pada Gambar 3.15



Gambar 3.24 Perancangan Struktur Menu

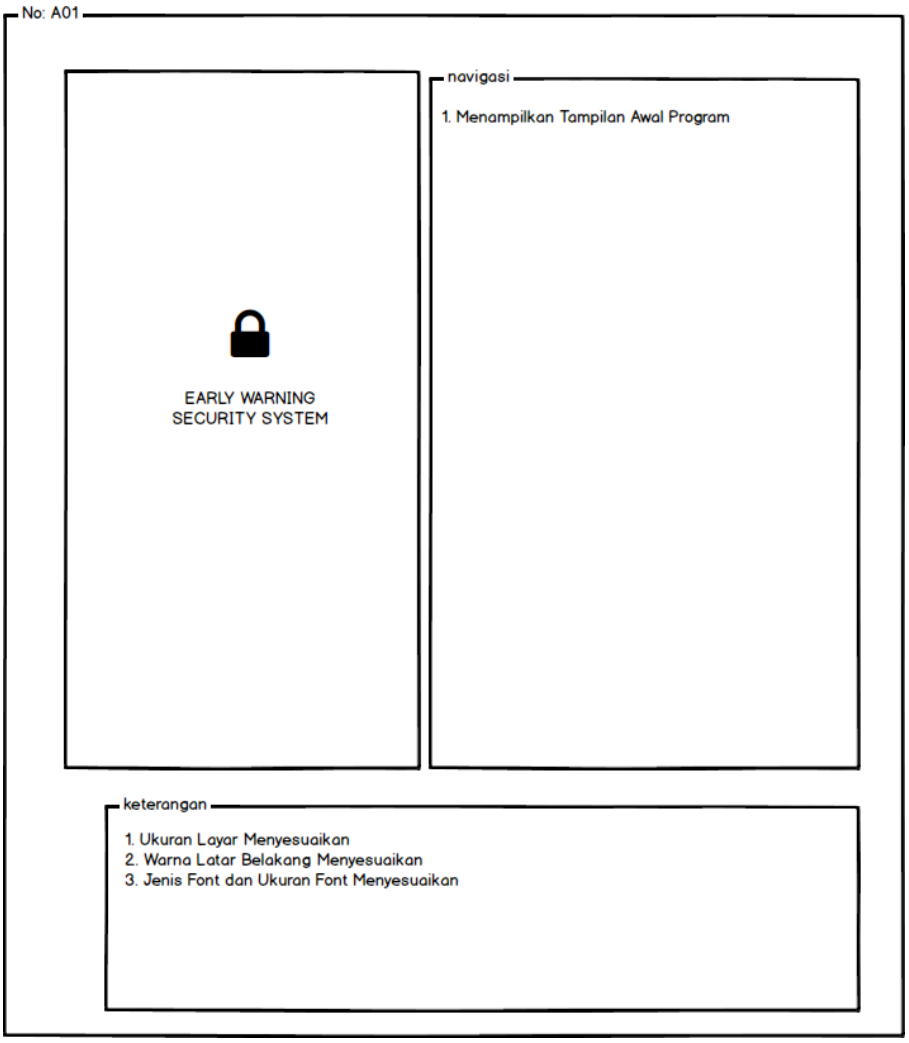
3.1.10.2 Perancangan Antarmuka

Sistem yang akan dibangun diharapkan menyediakan interface (antarmuka) yang mudah dipahami dan digunakan oleh user. Berikut ini adalah perancangan antarmuka dari kedua sub sistem yang akan dibuat.

Berikut adalah Perancangan Antarmuka :

1) Perancangan Antarmuka *Opening*

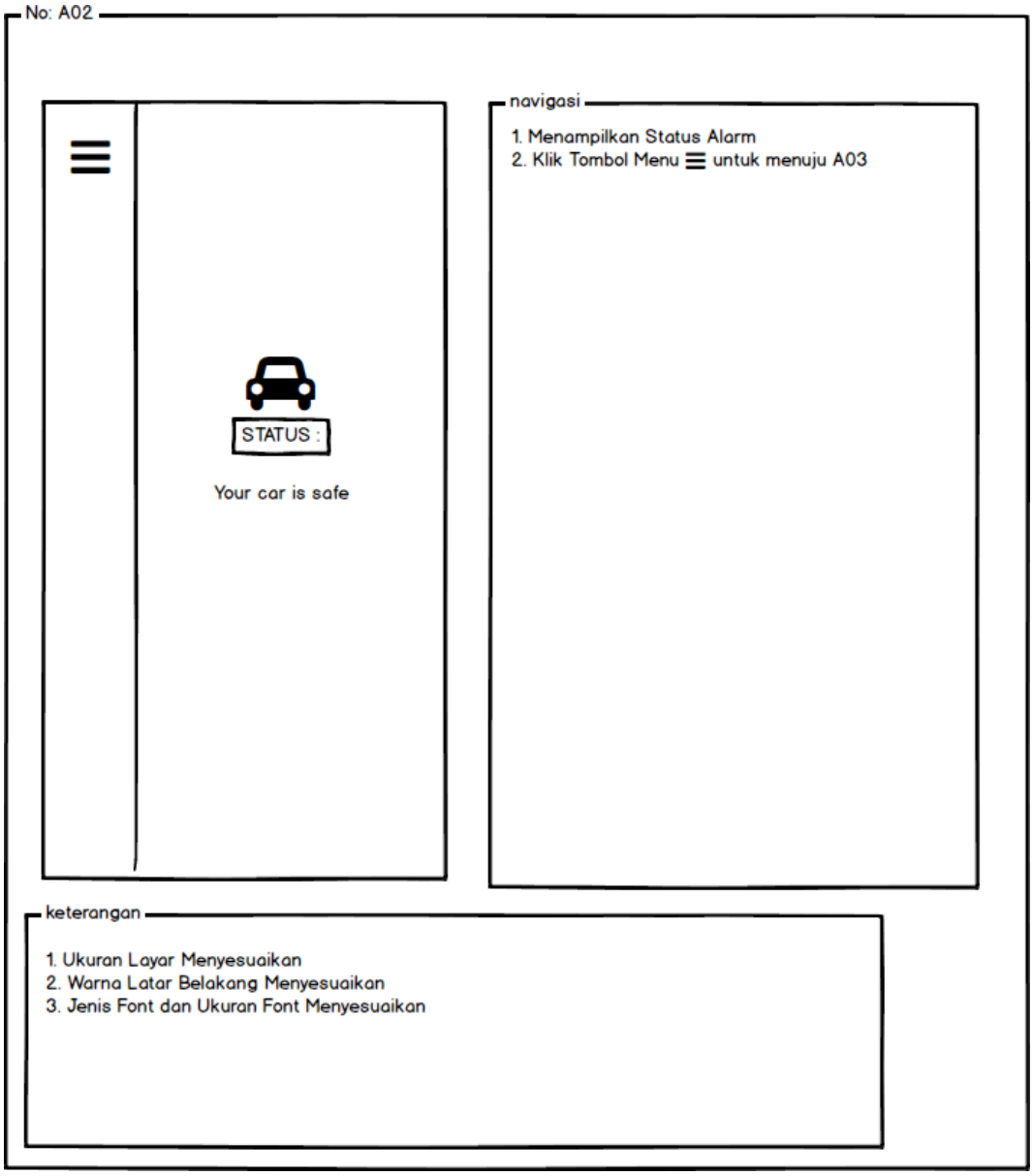
Perancangan antarmuka opening ketika aplikasi pertama kali dibuka.



Gambar 3.25 Perancangan Antarmuka Opening

2. Perancangan Antarmuka Halaman Utama

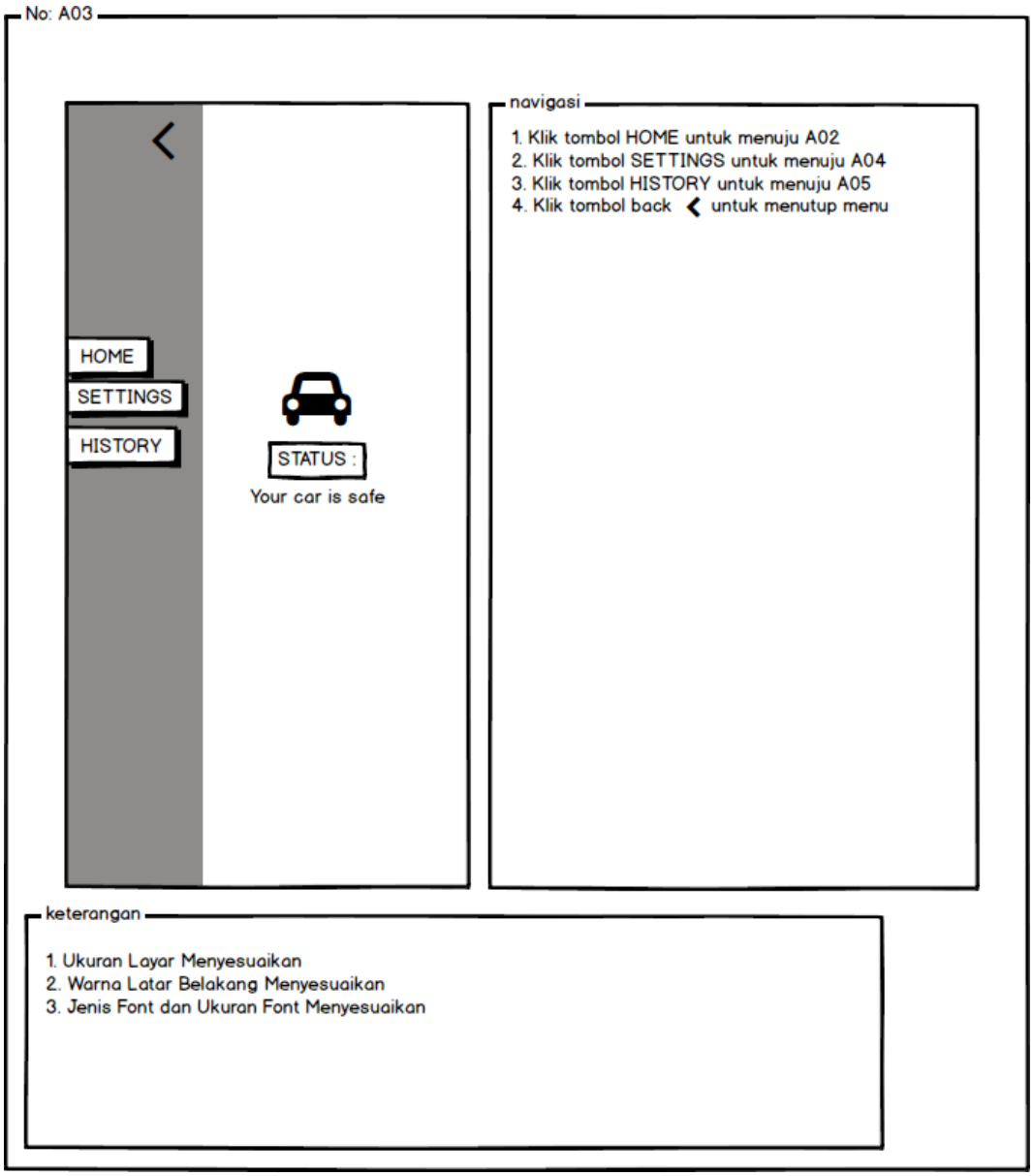
Perancangan atarmuka halaman utama ketika pertama membuka aplikasi



Gambar 3.26 Perancangan Antarmuka Halaman Utama

3. Perancangan Antarmuka Menu

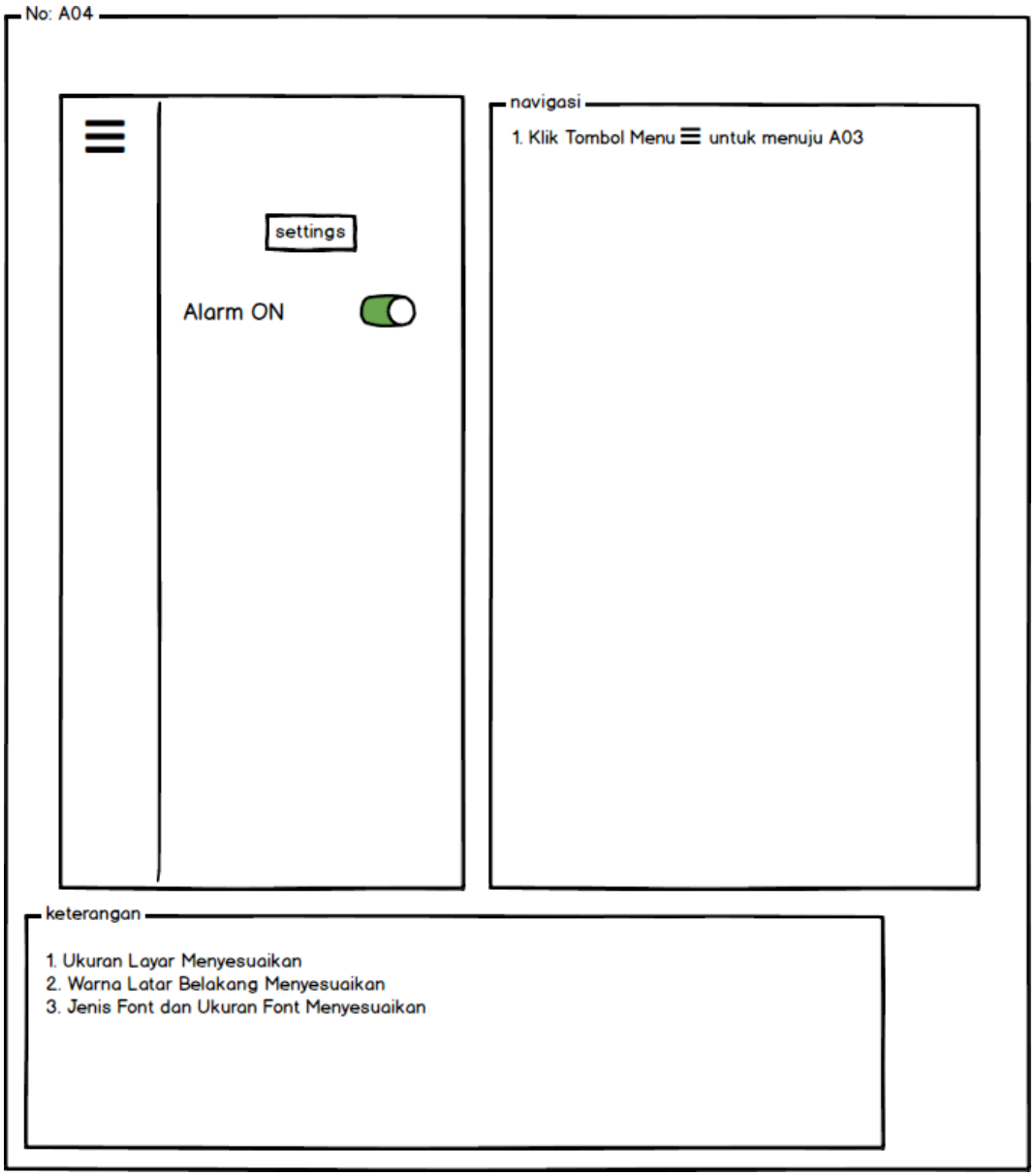
Perancangan atarmuka menu untuk memilih menu yang tersedia



Gambar 3.27 Perancangan Antarmuka Menu

4. Perancangan Antarmuka Settings

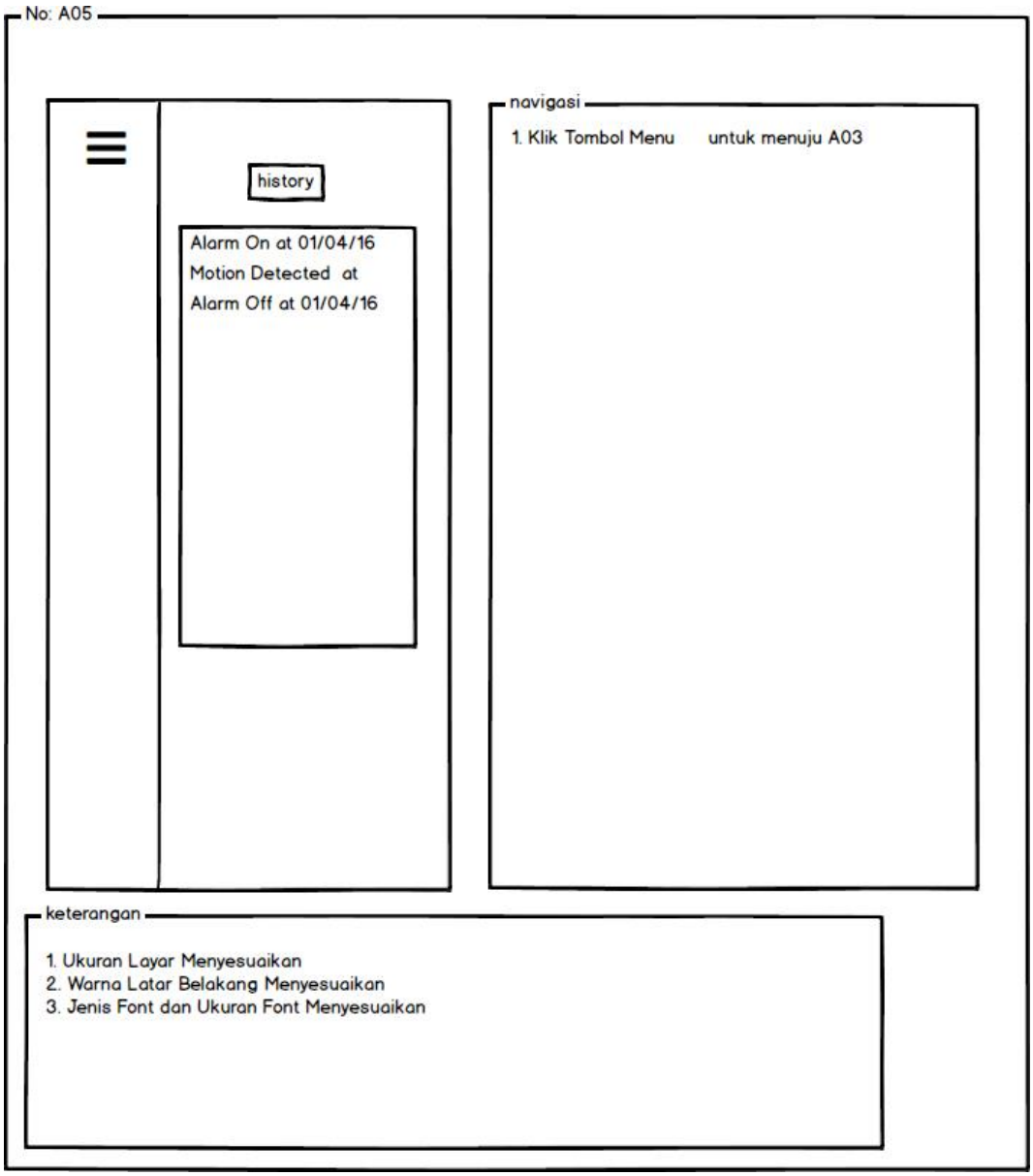
Perancangan atarmuka Settings untuk konfigurasi alarm



Gambar 3.28 Perancangan Antarmuka Konfigurasi Alarm

5. Perancangan Antarmuka History

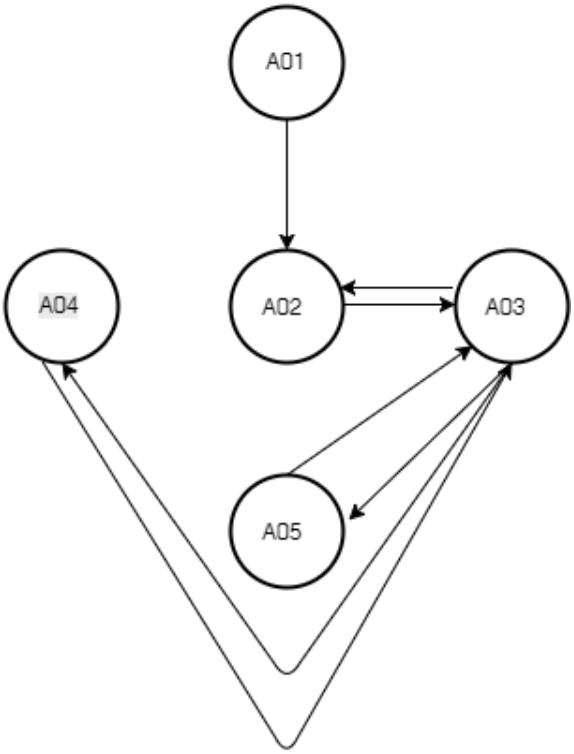
Perancangan atarmuka History untuk melihat catatan waktu perubahan konfigurasi dan peringatan ketika terjadi pergerakan.



Gambar 3.29 Perancangan Antarmuka History

3.1.11 Jaringan Semantik

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukkan hubungan antar objek (lingkaran) dan garis yang menggambarkan formasi antar objek tersebut.



Gambar 3.30 Jaringan Semantik