

**APLIKASI ALARM PENCEGAH PENCURIAN SMARTPHONE
BERBASIS ANDROID**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika**



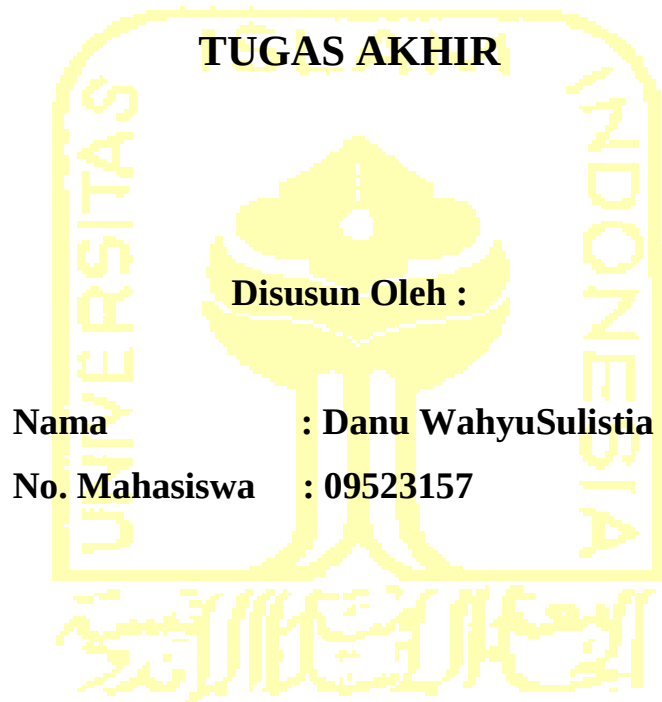
DISUSUN OLEH :

**Nama : Danu WahyuSulistia
No. Mahasiswa : 09523157**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2014

HALAMAN PENGESAHAN PEMBIMBING
APLIKASI ALARM PENCEGAH PENCURIAN
SMARTPHONE BERBASIS ANDROID



Yogyakarta, 17 Februari 2015

Telah disetujui dan disahkan oleh
Pembimbing

Ahmad Luthfi,S.Kom, M.Kom

LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertanda tangan di bawah ini,

Nama : Danu WahyuSulistia

No. Mahasiswa : 09523157

Menyatakan bahwa Tugas Akhir ini adalah hasil karya saya. Apabila di kemudian hari terbukti bahwa karya ini adalah bukan hasil karya saya sendiri, maka saya akan siap menanggung resiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 17 Februari 2015

Danu WahyuSulistia

LEMBAR PENGESAHAN PENGUJI
APLIKASI ALARM PENCEGAH PENCURIAN
SMARTPHONE BERBASIS ANDROID

Disusun oleh:

Nama : Danu WahyuSulistia

No.Mahasiswa : 09 523 157

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia.
Yogyakarta, Februari 2015

Tim Penguji

Ahmad Luthfi, S.Kom, M.Kom

Ketua

TaufiqHidayat, ST., MCS

Anggota I

Sheila Nurul huda, S.kom., M.Cs

Anggota II

Mengetahui,
Ketua Program Studi Teknik Informatika
Universitas Islam Indonesia

Hendrik, ST., M.Eng

HALAMAN PERSEMBAHAN

*Kupersembahkan untuk papa dan mama..Suhari dan sukinem..
Pa.. Ma..Terimakasih untuk pengorbanan, kasih sayang, suppor dan,
perlindungan yang tidak ada habisnya..*

*Kupersembahkan untuk wanita yang setia ingin selalu bersamaku..
Pendamping hidupku..*

Dan kupersembahkan untuk masa depanku..

HALAMAN MOTTO

“LA HAULA WALA QUWWATA ILLA BILLAH”

Tiadadayadanupayamelainkandenganpertolongan Allah

“MAN JADDA WA JADA”

Yang bersungguh – sungguhpastimendapathasil

KATA PENGANTAR



Assalamu'alaikum Wr. Wb.

Alhamdulillah, segala puja dan puji syukur kehadiran Allah SWT pemilik alam semesta yang memberikan nikmat kehidupan yang tak terhingga. Shalawat beriring salam kepada rasulullah Muhammad SAW yang telah membantu membimbing kita pada cahaya-Nya.

Tugas akhir dan laporan ini dibuat untuk memenuhi syarat akhir menjadi Sarjana Komputer pada jurusan Teknik Informatika Universitas Islam Indonesia. Tidak terasa selesai sudah perjalanan kuliah strata satu seiring selesainya tugas akhir ini. Ucapan terima kasih penulis ucapkan kepada:

1. Allah SWT, yang telah memberikan rahmat kehidupan dan kesehatan sehingga penulis bisa menyelesaikan tugas akhir ini.
2. Kedua orang tua penulis (Suhari dan Sukinem) yang selalu membimbing dan mensupport dalam segala hal kebaikan.
3. Bapak Hendrik ST., M.Eng, Selaku Ketua Program Studi Teknik Informatika Universitas Islam Indonesia
4. Bapak Ahmad Luthfi, S.Kom, M.Kom., bapak dosen pembimbing yang sabar membimbing, memberiarahan, inspirasi, nasihat dan motivasi yang sangat bermanfaat bagi penulis.
5. Buat yang terkasih Istri Putri Arba, S.Psi, yang selalu member motivasi dan semangat dalam penyelesaian tugas akhir ini.
6. Para dosen penguji.

7. Sahabat – sahabat yang membantu dalam penyusunan tugas akhir ini.
8. Sahabat – sahabat seperjuanganku di Yogyakarta.
9. Teman – teman mahasiswa Infinity Teknik Informatika 2009.

Penulis menyadari bahwa laporan ini jauh dari sempurna. Maka dari itu penulis memohon maaf apabila ada hal – hal yang membuat pembaca jadi tidak berkenan membacanya. Semoga tugas akhir dan laporan penulis dapat bermanfaat dan menjadi suatu karya yang bias member dampak positif untuk pembaca.

Yogyakarta, 17Februari 2015

Danu WahyuSulistia

SARI

Angka penjualan *smartphone* yang meningkat setiap tahun membuktikan bahwa *smartphone* telah menjadi kebutuhan hidup masyarakat Indonesia. *Smartphone* merupakan kebutuhan yang digunakan untuk berkomunikasi yang disebabkan oleh jarak baik itu tempat, daerah, kota bahkan Negara. Selain sebagai alat komunikasi, *smartphone* juga digunakan sebagai sarana hiburan dan sarana pembelajaran.

Namun dalam kehidupan, terkadang kita sering dihadapkan dengan kondisi – kondisi yang tidak nyaman yang mungkin disebabkan oleh lingkungan sekitar terutama masalah keamanan. *Smartphone* adalah alat yang sangat penting bagi kita dan merupakan barang yang mudah hilang jika kita tidak menjaganya. Oleh karena itu pembuatan Aplikasi *Alarm* Pencegah Pencurian *Smartphone* bertujuan untuk membantu para pemilik *smartphone* menjaga *smartphone* nya dari kehilangan oleh pihak – pihak lain atau dari kemalingan.

Metode pengembangan pada tugas akhir ini menggunakan UML (Unified Modeling Language). UML merupakan alat yang digunakan sebagai pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. Perancangan Aplikasi *Alarm* Pencegahan Pencurian *Smartphone* ini menggunakan Sistem Operasi berbasis android yang memanfaatkan fitur *accelerometer* sebagai alat pendeteksi gerakan dan penggunaan *password* sebagai keamanan *smartphone* serta alarm sebagai pemberi informasi atau peringatan kepada pengguna.

Kesimpulan yang dicapai adalah terciptanya Aplikasi *Alarm* Pencegah Pencurian *Smartphone* Berbasis Android yang bisa membantu mencegah hilangnya *smartphone* dari kemalingan.

Kata kunci : *Smartphone, Alarm, Accelerometer, Password.*

TAKARIR

<i>Android</i>	Sistem operasi perangkat mobile
<i>Smartphone</i>	Telepon genggam
<i>Accelerometer</i>	Alat untuk mengukur percepatan, mendeteksi dan mengukur getaran, serta mengukur percepatan akibat gravitasi
<i>Password</i>	Kata sandi
<i>Alarm</i>	Bunyi peringatan
<i>IDE</i>	Aplikasi untuk membuat serta mengembangkan perangkat lunak
<i>Open Source</i>	Pengembangan terbuka
<i>Interface</i>	Antar muka

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN PEMBIMBING.....	ii
HALAMAN PERNYATAAN KEASLIAN.....	iii
LEMBAR PENGESAHAN PENGUJI.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO.....	vi
KATA PENGANTAR.....	vii
SARI	ix
TAKARIR.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
 BAB I PENDAHULUAN.....	 1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	3
1.7 Metodologi Penelitian	3
1.8 Sistematika Penulisan	4

BAB II LANDASAN TEORI	6
2.2 Accelerometer.....	6
2.3 Android	9
2.4 UML (unified Modeling Language).....	11
2.4.1 Use Case Diagram.....	12
2.4.2 Activity Diagram.....	14
2.5 Eclipse IDE	15
2.6 Telaah Aplikasi Sejenis.....	16
2.6.1 Anti Theft Alarm.....	16
2.6.2 Prey Anti Theft.....	17
2.6.3 Camera Block Anti Spy Malware.....	17
2.7 Hasil Telaah Aplikasi.....	18
 BAB III METODOLOGI	19
3.1 Analisis Kebutuhan	19
3.1.1 Analisis Kebutuhan Masukan	20
3.1.2 Analisis Kebutuhan Proses.....	20
3.1.3 Analisis Kebutuhan Keluaran	21
3.1.4 Analisis Kebutuhan Antarmuka	21
3.2 Perancangan Perangkat Lunak.....	21
3.2.1 Metode Perancangan.....	21
3.3 Perancangan AntarMuka.....	22
3.3.1 Perancangan Menu Home	22
3.3.2 Perancangan Menu Pengaturan	23
3.4 Unified Modeling Language (UML).....	24
3.5.1 Use Case Diagram.....	24
3.5.2 Activity Diagram.....	25
3.5 Rencana Pengujian	29
 BAB IV IMPLEMENTASI DAN ANALISIS	30
4.1 Perangkat Lunak yang Digunakan	30

4.2 Perangkat Keras yang Digunakan	30
4.3 Implementasi.....	31
4.3.1 Halaman Menu Utama	31
4.3.2 Halaman Menu Pengaturan	32
4.3.3 Menu Pilih Password.....	34
4.3.4 Menu Password Input Angka	35
4.3.5 Menu Password Pattern.....	37
4.3.6 Menu Password Suara	48
4.4 Pengujian Aplikasi.....	39
4.5 Hasil Pengujian	41
 BAB V PENUTUP	 43
5.1 Kesimpulan.....	43
5.2 Saran.....	43
 DAFTAR PUSTAKA	 44

DAFTAR TABEL

Tabel 2.1 Notasi Komponen Use Case Diagram	13
Tabel 2.2 Notasi Komponen Activity Diagram	14
Tabel 4.1 Perangkat Uji Coba	40
Tabel 4.2 Hasil Pengujian	42

DAFTAR GAMBAR

Gambar 2.1 Accelerometer	6
Gambar 2.2 Contoh Use Case Diagram	12
Gambar 2.4 Anti Theft Alarm	16
Gambar 2.5 Prey Anti Theft	17
Gambar 2.6 Camera Block Anti Spy Malware	17
Gambar 3.1 Perancangan Menu Home	22
Gambar 3.2 Perancangan Menu Pengaturan	23
Gambar 3.3 Perancangan Input Number, Pattern dan Suara	24
Gambar 3.5 Use Case Diagram	25
Gambar 3.6 Activity Diagram Aktif Sensor	26
Gambar 3.7 Activity Diagram Pengaturan	28
Gambar 4.1 Halaman Menu Utama	31
Gambar 4.2 Source Code Halaman Menu Home	32
Gambar 4.3 Halaman Menu Pengaturan	33
Gambar 4.4 Source Code Menu Halaman Pengaturan	33
Gambar 4.5 Menu Pilih Password	34
Gambar 4.6 Source Code Halaman Pilih Password	35
Gambar 4.7 Menu Password Input Angka	36

Gambar 4.8 Source code Menu Password Input Angka	36
Gambar 4.9 Menu Password Pattern.....	37
Gambar 4.10 Source Code Menu Password Pattern	38
Gambar 4.11 Menu Password Suara	38
Gambar 4.12 Source Code Menu Password Suara.....	39

BAB I

PENDAHULUAN

1.1 Latar Belakang

Smartphone merupakan salah satu alat komunikasi yang memegang peranan penting dalam kehidupan sehari-hari. Teknologi yang berkembang dari hari ke hari membuat *smartphone* menjadi pilihan. Banyak aplikasi yang dapat digunakan sebagai sarana hiburan maupun sebagai alat bantu kerja bahkan sudah ada aplikasi yang dikembangkan untuk melacak jika *smartphone* tersebut hilang.

Namun, alangkah baiknya kita dapat mencegah *smartphone* yang kita miliki dari kehilangan. *Smartphone* yang beredar di pasaran saat ini memiliki teknologi yang canggih sehingga banyak fitur yang dapat digunakan sebagai parameter dalam membuat aplikasi pencegah kehilangan *smartphone* diakibatkan oleh tindakan orang lain. Salah satunya adalah fitur *accelerometer*. *Accelerometer* adalah sebuah *transducer* 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 *smartphone*. Dari kedua hal tersebut, maka dapat dikembangkan suatu aplikasi pengaman *smartphone* secara fisik dengan nama "Aplikasi *Alarm* Pencegah Pencurian *Smartphone* berbasis Android" dengan menggunakan fitur *accelerometer*.

Cara kerja aplikasi ini adalah mendeteksi pergerakan yang terjadi pada *smartphone*. Aplikasi ini dilengkapi juga dengan fitur penonaktifan *alarm* dengan *password* berupa kombinasi angka, *Pattern* berupa kombinasi kunci gerakan atau *password* dengan suara berbasis *google speech recognizer*. Sehingga *smartphone* dapat mengenali identitas pemiliknya dan menghindari kesalahan identifikasi pemilik seperti halnya remot pada *alarm* kendaraan.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas dapat dirumuskan masalah yang akan diselesaikan, yaitubagaimana merancang suatu Aplikasi *Alarm* Pencegah Pencurian*Smartphone* berbasis Android dengan menggunakan fitur *acceleromete*.

1.3 Batasan Masalah

Batasan untuk aplikasi yang akan dibangun adalah :

1. Aplikasi ini hanya berjalan pada perangkat dengan sistem operasi Android.
2. Deteksi suara bebasis *googlespeech recognizer* tidak mencakup keunikan deteksi suara, hanya mendeteksi kata kunci yang diucapkan.
3. Aplikasi inidapat berjalan di latar belakang (secara background).
4. Pendeteksian gerak menggunakan fitur *accelometer* yang terdapat pada *smartphone* android.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah untuk membantu para pengguna *smartphone* dalam menjaga *smartphonenya* dari pencurian oleh orang-orang yang tidak bertanggung jawab.

1.5 Manfaat Penelitian

Adapun manfaat penelitian adalah sebagai berikut :

1. Bagi Pengguna/orang lain, Sebagai salah satu aplikasi untuk pencegahan hilangnya *smartphone*.
2. Bagi Dunia Ilmu Pengetahuan, aplikasi ini diharapkan menjadi percontohan bagi aplikasi-aplikasi berikut nya.

1.6 Metodologi Penelitian

Metode penelitian yang digunakan penulis pada pembuatan skripsi “Aplikasi Alarm Pencegah Pencurian Smartphone berbasis Android” adalah sebagai berikut :

1. Studi Pustaka

Metode pustaka merupakan metode yang mengacu pada buku-buku pedoman yang berguna sebagai referensi baik itu pada perpustakaan, maupun pada literatur lainnya. Penulis mengambil beberapa informasi dari beberapa buku yang terkait dengan pembuatan aplikasi ini.

2. Analisis

Tahapan untuk menganalisa semua dari kebutuhan proses, *input*, *output*, dan antar muka yang akan dibangun untuk aplikasi Android Alarm Pencegah Kemalingan Smartphone Berbasis Android.

3. Perancangan

Pada tahap ini penulis akan melakukan perancangan aplikasi. Perancangan aplikasi dilakukan berdasarkan hasil penelitian yang telah dilakukan. Perancangan yang dilakukan peneliti meliputi perancangan UML berupa *Use Case* dan *Activity* diagram, serta merancang *user interface* dari aplikasi tersebut.

4. Implementasi

Pada tahap ini penulis memasuki tahap pembuatan aplikasi, pembuatan aplikasi dilakukan sesuai dengan perancangan yang telah dilakukan sebelumnya.

5. Pengujian

Pada tahap ini, penulis melakukan pengujian dari aplikasi yang telah dibuat untuk mengetahui kekurangan kekurangan apa saja yang terdapat pada aplikasi tersebut.

1.7 Sistematika Penulisan

Pada penulisan skripsi ini, penulis menguraikan nya menjadi 5 BAB yang masing masing diuraikan sebagai berikut:

BAB I. PENDAHULUAN

Pada Bab ini penulis akan membahas tentang Latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penelitian, metode penelitian, dan sistematika penulisan.

BAB II. LANDASAN TEORI

Membahas dasar teori yang berfungsi sebagai sumber atau alat dalam memahami permasalahan dan berkaitan dengan konsep dasar aplikasi, serta perangkat lunak yang digunakan untuk pembuatan aplikasi.

BAB III METODOLOGI

Bab ini menjelaskan tentang analisis aplikasi *Alarm Pencegah Pencurian Smartphone* yang meliputi tentang analisis kebutuhan aplikasi, analisis kelayakan aplikasi, dan perancangan aplikasi.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Membahas analisis kebutuhan perangkat lunak, metode perancangan sistem ataupun aplikasi, metode perancangan antarmuka. Analisis kebutuhan memuat hasil analisis yang merupakan kebutuhan dalam pembuatan fungsi aplikasi serta antarmuka dari aplikasi yang dibangun.

BAB V PENUTUP

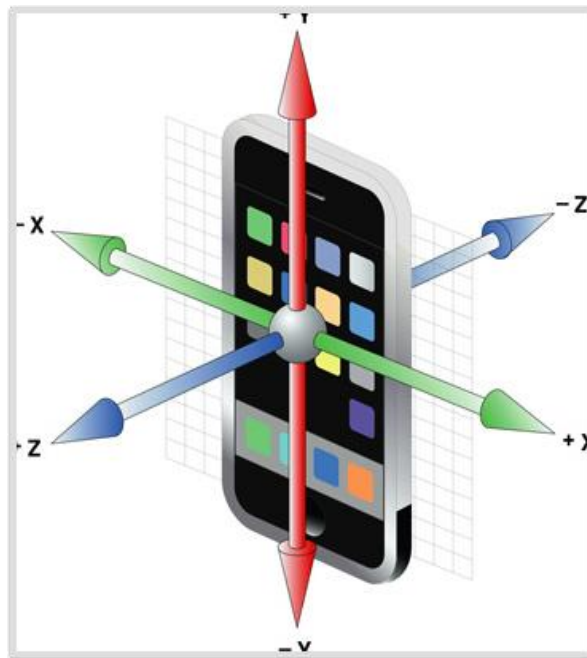
Bab ini merupakan bab terakhir dari serangkaian penyusunan penulisan skripsi yang berisi kesimpulan dan saran tentang aplikasi *Alarm Pencegah Pencurian Smartphone* berbasis Android.

BAB II

LANDASAN TEORI

2.1 Accelerometer

Accelerometer adalah alat yang digunakan untuk mengukur percepatan, mendeteksi dan mengukur getaran (vibrasi), dan mengukur percepatan akibat gravitasi. Sensor *accelerometer* mengukur percepatan dari 3 sumbu gerakan akibat gerakan benda yang melekat padanya (Wiryadinata,2009). Accelerometer dalam penggunaannya dapat dilihat pada gambar 2.1.



Gambar 2.1 Accelerometer

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.

Prinsip kerja dari accelerometer 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.

Berikut beberapa contoh penggunaan *accelerometer*:

1. Transportasi: 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.
2. Bidang Medis: *Sport Watch*, berupa jam tangan olahraga yang juga dapat menghitung berapa banyak langkah yang telah kita lakukan, menggunakan *accelerometer* untuk menghitung kecepatan dan jarak dari si pelari yang menggunakannya.
3. *Science and Engineering*: *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.

4. Peralatan Elektronik: *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 *smartphone* yang menggunakan *accelerometer* untuk mengubah lagu yang dimainkan (Track Switching). Camera recorder menggunakan *accelerometer* untuk menstabilkan gambar (image stabilization). Camera digital menggunakan *accelerometer* untuk menu pilihan anti blur ketika mengambil gambar.

2.2 Android

Android *Operating system* merupakan sistem operasi yang berbasis linux yang yang diperuntukan untuk telepon pintar (Smartphone) maupun tablet. Android merupakan system operasi yang bersifat *opensource* / terbuka untuk para developer agar dapat menciptakan aplikasi untuk berbagai *device* android. Google Inc. membeli system android yang merupakan pendatang baru didunia ponsel. Untuk membentuk android, dibentuklah *Open Handset Alliance*, konsorium dari berbagai perusahaan *hardware* (perangkat keras), *software* (perangkat lunak), dan telekomunikasi, termasuk hTc ,Intel Motorola, Qualcomm, dan N-Vidia (Safaat, 2012) .

Adapun versi - versi Android adalah sebagai berikut :

a. Android versi 1.1

Pada 9 maret 2009 android merilis Android versi 1.1 .android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, kam alarm, voice search (pencarian suara), pengiriman pesan dengan gmail, dan pemberitahuan email.

b. Android versi 1.5 (Cupcake)

Pada pertengahan tahun 2009, google kembali merilis *smartphone* dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 yang dikenal dengan kode Cupcake. Terdapat banyak pembaharuan baru pada versi 1.5 seperti kemampuan merekam dan menonton video dengan modus kamera, upload video ke Youtube, dan gambar ke google Picasa langsung dari *smartphone*, dukungan *Bluetooth A2DP*, kemampuan terhubung ke *headsetBluetooth*, animasi layar, dan *keyboard* pada layar yang dapat disesuaikan dengan *system*.

c. Android versi 1.6 (Donut)

Android versi ini diliris pada bulan September 2009 dengan menampilkan proses pencarian yang lebih baik disbanding sebelumnya, penggunaan baterai indicator, dan *control applet* VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus, kamera, *camcorder*, dan galeri yang diintegrasikan, jaringan CDMA / EVDO,

802.1x, penggunaan Virtual private network (VPN), gesture, dan *text to speech engine*, dan penggunaan resolusi VWGA.

d. Android versi 2.1 (Éclair)

Android dengan nama kode Éclair dirilis pada 3 Desember 2009 dengan terdapat beberapa perubahan seperti pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan *user interface* dengan *browser* baru dan dukungan HTML 5, daftar kontak yang baru, dukungan *flash* untuk kamera 3.2 MP, digital zoom, dan *Bluetooth* 2.1. Dengan semakin berkembangnya dan semakin bertambahnya jumlah *handset* Android, semakin banyak pula pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada *system* operasi android. Aplikasi terkenal yang diubah ke *system* operasi android adalah Shazam, Backgrounds, dan WheaterBug. *System* operasi android dalam situs internet juga dianggap penting untuk menciptakan aplikasi android asli, contohnya oleh MySpace dan Facebook.

e. Android versi 2.2 (froyo)

Pada tanggal 20 Mei 2010, Android versi 2.2 diluncurkan. Perubahan perubahan umum terhadap versi sebelumnya antara lain dukungan Adobe Flash 10.1 kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 *Javascript engine* yang dipakai Google Chrome yang mampu mempercepat kemampuan rendering pada *browser*, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portable, dan kemampuan *autoupdate* dalam aplikasi Android Market.

f. Android versi 2.3 (Gingerbread)

Pada 6 desember 2010, android versi 2.3 (Gingerbread) diluncurkan. Perubahan perubahan umum yang terdapat pada versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi *copy paste*, *User Interface* yang didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

g. Android versi 3.0 (Honeycomb)

Android versi ini dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada honeycomb juga berbeda karena dirancang untuk *tabletphone*. *Honeycomb* juga mendukung multi prosesor dan juga akselerasi *hardware* untuk grafis.

h. Android versi 4.0 (ICS: Ice Cream Sandwich)

Android ini diumumkan pada tanggal 19 oktober 2011, membawa fitur *honeycomb* untuk *smartphone* dan menambah fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan control, terpadu kontak jaringan social, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan NFC. Ponsel pertama yang menggunakan *system* operasi ini adalah Samsung Galaxy Nexus.

i. Android versi 4.1 (Jelly Bean)

Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input *keyboard*, desain baru fitur pencarian, UI yang baru dan pencarian melalui *Voice Search* yang lebih cepat. Tak ketinggalan *Google Now* yang juga menjadi bagian yang diperbaharui. *Google now* memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu lintas, ataupun hasil pertandingan olahraga. *System* operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7.

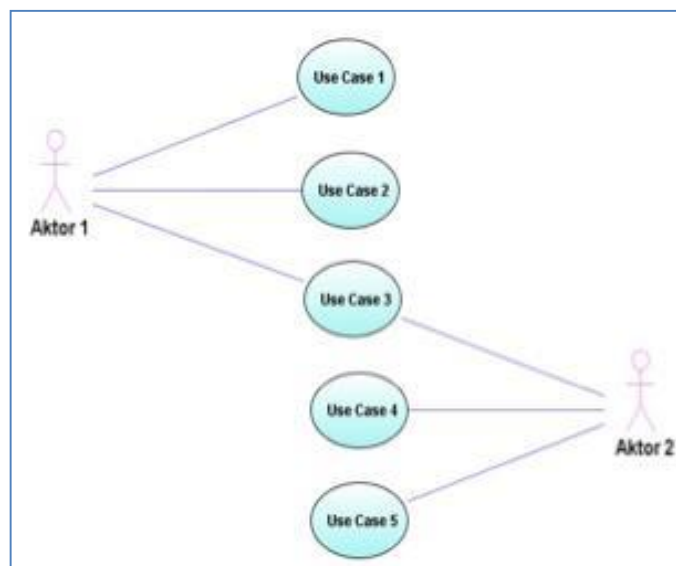
2.4 UML (Unified Modeling Language)

Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan dan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan *system* OOP dan sekelompok tool untuk mendukung pengembangan *system* tersebut. UML mulai diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan

model, teknologi, dan standart OOP sejak tahun 1980-an. Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP. UML merupakan dasar bagi *tool* desain berorientasi sejak dari IBM(Kroenke, 2003).

2.4.1 Use Case Diagram

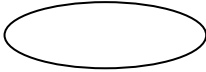
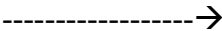
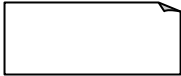
Use Case Diagram digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang akan berinteraksi dengan sistem aplikasi. Lihat gambar 2.2 untuk contoh *use case diagram*.



Gambar 2.2 Contoh Use Case Diagram

Dalam *use case* terdapat elemen-elemen notasi yang berbeda untuk setiap kegunaan yang berbeda. Penjelasan elemen notasi *use case* dapat dilihat pada tabel 2.1.

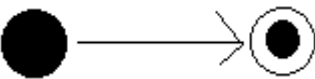
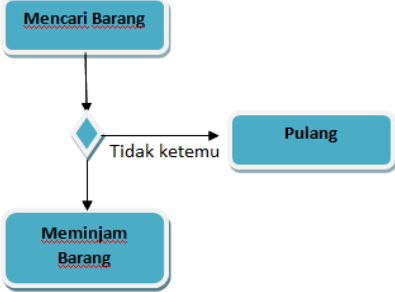
Table 2.1 Notasi Komponen Diagram *Use Case Diagram*

Notasi	Nama Notasi	Fungsi
	Package	Untuk mengelompokan elemen elemen dari aplikasi yang sedang dibangun.
	Actor	Menggambarkan apa yang harus diperbuat oleh aplikasi perangkat lunak.
	Usecase	Menjelaskan apa yang dilakukan oleh actor dan aplikasi
	Depedency	Memberikan penjelasan bahwa perubahan yang terjadi pada salah satu elemen member pengaruh pada elemen lain.
	Association	Menggambarkan relasi antara actor dan usecase
	Generalization	Menggambarkan relasi lanjut antar usecase
	Note/Comment	Memberi keterangan dan komentar tambahan dari suatu elemen.
	Anchor to note	Melekatkan catatan pada usecase atau actor tertentu dalam diagram.

2.4.2 Activity Diagram

Activity diagram menggambarkan berbagai aliran aktifitas dalam sistem yang sedang dirancang dari use case dalam bentuk grafik. Dalam *activity diagram* terdapat elemen – elemen notasi yang berbeda untuk setiap kegunaan yang berbeda. Penjelasan elemen notasi *activity diagram* dapat dilihat pada tabel 2.2.

Tabel 2.2 Notasi Komponen diagram *Activity Diagram*.

Notasi	Nama Notasi	Fungsi
	Action	Menggambarkan 1 arah dalam alur aplikasi
	Start Node / Initial	Menggambarkan dimana aliran berawal
	Activity Final Node	Menggambarkan akhir dari banyak aliran pada diagram aktivitas
	Flow Final Node	Menunjukkan akhir dari aliran tunggal dalam diagram aktivitas.
	Control Flow	Menunjukkan aliran kendali dari satu aksi ke aksi yang lain.
	Decision Node	Untuk menggabungkan alur. Sebuah keputusan akan memiliki kondisi yang harus ada untuk menentukan alur yang akan diambil

2.5 Eclipse IDE

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di banyak *platform*. Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya. Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, pengetesan perangkat lunak, pengembangan web, dan lain sebagainya. Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *opensource*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in* (Hermawan, 2012).

Banyak orang mengenal Eclipse sebagai IDE (*Integrated Development Environment*) untuk bahasa Java, tapi kenyataannya Eclipse lebih dari sekedar IDE untuk Java. Komunitas Eclipse memiliki lebih dari 60 proyek *opensource*. Proyek – proyek ini secara konsep terbagi menjadi 7 kategori :

1. *Enterprise Development*
2. *Embedded and Device Development*
3. *Rich Client Platform*
4. *Rich Internet Application*
5. *Application Frameworks*
6. *Application Lifecycle Management (ALM)*
7. *Service Oriented Architecture (SOA)*

Secara umum Eclipse digunakan untuk membangun *software* inovatif berstandar industri, dan alat bantu beserta *framework*nya membantu pekerjaan menjadi lebih mudah.

.

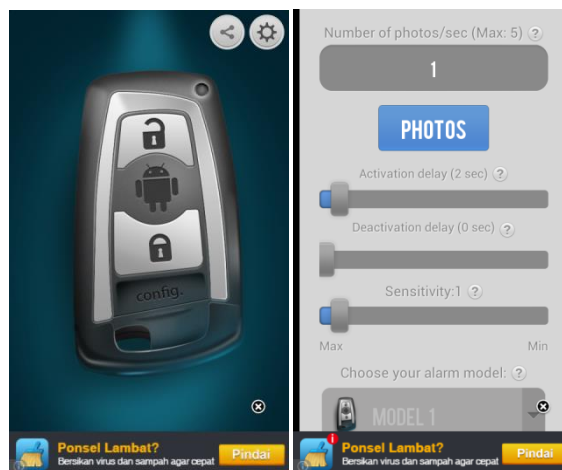
2.6 Telaah Aplikasi Sejenis

Berikut akan ditampilkan aplikasi sejenis berdasarkan kategori fungsinya sebagai aplikasi alarm pencegah anti maling.

2.6.1 Anti Theft Alarm

Anti theft alarm adalah aplikasi alarm untuk mencegah hilangnya *smartphone* atau anti maling dengan peringatan alarm, dimana aplikasi ini bekerja ketika terjadinya gerakan atau perpindahan posisi pada *smartphone*, kelebihan aplikasi ini adalah berjalannya kamera saat aplikasi berbunyi.

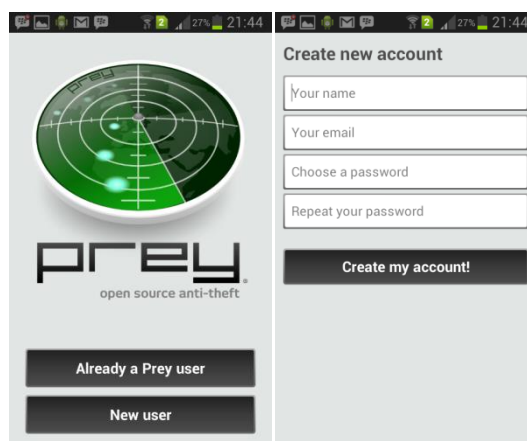
Adapun fitur-fitur yang ada pada aplikasi ini yaitu tombol pengaktifkan alarm, tombol mematikan alarm, serta pengaturan nada, sensitivitas, active delay dan pengaturan password.



Gambar 2.4Anti Theft Alarm

2.6.2 Prey Anti Theft

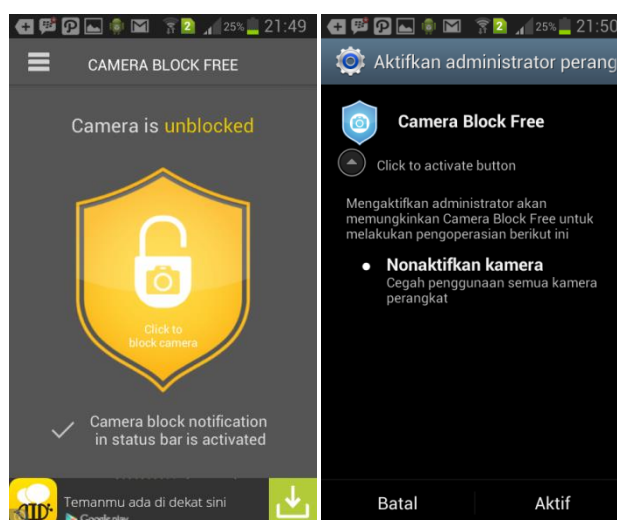
Prey anti theft adalah aplikasi tracking *smartphone* dimana saat *smartphone* hilang atau dicuri kita dapat melacak posisi *smartphone* milik kita dan tentunya harus terkoneksi dengan internet.



Gambar 2.5Prey Anti Theft

2.6.3 Camera Block Anti Spy Malware

Camera block anti spy malware adalah aplikasi pengamanan camera pada *smartphone* agar tidak dapat di akses sembarangan oleh pihak lain. Namun dalam penggunaan nya setelah mengaktifkan aplikasi ini untuk membuka kembali kamera agar dapat digunakan harus dihapus, karena dalam aplikasi ini tidak terdapat menu untuk membuka camera yang sudah diblock atau dimatikan.



Gambar 2.6Camera Block Anti Spy Malware

2.7 Hasil Telaah Aplikasi

Dari hasil telaah aplikasi sejenis diatas penulis mengambil beberapa kesimpulan untuk menggunakan beberapa fitur yang sudah ada pada aplikasi Anti Theft Alarm yaitu :

1. Menggunakan fitur pemilihan nada.
2. Menggunakan fitur aktif delay.
3. Menggunakan fitur sensitifitas.
4. Menggunakan fitur aktif sensor dan nonaktif sensor.
5. Serta fitur pengaturan password.

BAB III

METODOLOGI

3.1 Analisis Kebutuhan

Analisis merupakan identifikasi terhadap hal-hal yang berkaitan dengan detail atau struktur pembangunan perangkat lunak. Tahap analisis merupakan tahap yang penting, karena hal tersebut bertujuan untuk mengidentifikasi, mengevaluasi permasalahan, serta mengetahui kebutuhan yang diperlukan sehingga dapat diusulkan perbaikannya.

Metode analisis ini diperlukan berkaitan dengan perancangan suatu sistem ataupun aplikasi yang akan dibuat dan berhubungan dengan hasil yang ingin dicapai oleh sistem ataupun aplikasi itu sendiri. Metode analisis digunakan setelah mengetahui tujuan dan persoalan apa yang ingin dipecahkan.

Secara garis besar gambaran kebutuhan sistem ataupun aplikasi yang akan dibuat yaitu suatu perangkat lunak yang dapat membantu pencegahan hilangnya *smartphone* berbasis android, yang nantinya aplikasi ini bertujuan untuk membantu pengguna dalam menjaga kehilangan *smartphone* dari orang-orang lingkungan sekitar yang mungkin berpotensi dalam melakukan pencurian *smartphone*, yakni *alarm* sebagai alat keamanan yang akan memberi informasi kepada pemilik bahwa *smartphone* yang dimiliki dalam kondisi aman atau tidak. Ada beberapa fitur yang membantu dalam keamanan *smartphone* dalam aplikasi ini yaitu dilengkapi dengan *password* yang terdiri dari 3 pilihan yakni *password* berupa input number, pattern dan suara. Ada beberapa analisis yang dibutuhkan untuk menyelesaikan masalah tersebut. Analisis - analisis tersebut adalah analisis kebutuhan masukan, analisis kebutuhan proses, analisis kebutuhan keluaran, dan analisis kebutuhan antarmuka.

3.1.1 Analisis Kebutuhan Masukan

Kebutuhan Masukan dalam aplikasi ini berdasarkan pada kebutuhan pengguna dalam mengamankan *smartphonenya* dari kemungkinan-kemungkinan hilangnya *smartphone* dari tangan-tangan yang tidak bertanggung jawab.

Adapun kebutuhan masukkan aplikasi ini adalah:

1. Kunci *alarm* dan buka *alarm* serta pengaturan.
2. Pengaturan yang berguna untuk mengatur penyesuaian sensitifitas, pemilihan nada, pemilihan tipe *password* dan *active delay*.
3. Menu *input password* berupa masukan angka.
4. Menu *input password pattern*.
5. Menu *input password password* suara .

3.1.2 Analisis Kebutuhan Proses

Setelah menentukan analisis kebutuhan masukan, langkah selanjutnya yaitu melakukan analisa kebutuhan proses, kebutuhan proses dalam Aplikasi *Alarm Pencegahan Pencurian Smartphone Berbasis Android* ini adalah:

1. Proses menyimpan nilai sensitifitas.
2. Proses menghidupkan sensor *accelerometer*.
3. Proses mematikan sensor *accelerometer*.
4. Proses menyimpan nilai pada pengaturan.

3.1.3 Analisis Kebutuhan Keluaran

Kebutuhan keluaran yang diperlukan dalam pengembangan ini adalah :

1. Menampilkan menu buka *password* saat aplikasi berbunyi.
2. Menyuarakan nada *alarm* saat aplikasi berjalan saat terjadi gerakan.
3. Menampilkan pemberitahuan *alarm active* dan *non active*.

3.1.4 Analisis Kebutuhan Antarmuka

Kebutuhan antarmuka (*interface*) dalam aplikasi ini dibuat secara mudah dan interaktif. Ini bertujuan untuk memudahkan pengguna berinteraksi dengan aplikasi.

3.2 Perancangan Perangkat Lunak

Aplikasi *Alarm Pencegahan Pencurian Smartphone* Berbasis Android ini merupakan aplikasi yang akan dibangun dengan menggunakan bahasa pemrograman Java yang sudah terdapat di *Eclipse*. Pemilihan bahasa pemrograman Java sebagai bahasa yang akan digunakan dalam pembangunan Aplikasi *Alarm Pencegahan Hilangnya Smartphone* Berbasis Android ini dikarenakan bahasa pemrograman Java merupakan bahasa yang bisa digunakan di Android. Selain itu, bahasa perograman bersifat berorientasi objek, sehingga dapat memudahkan dalam pembangunan aplikasi ini. Untuk pendukung *interface*, aplikasi ini akan menggunakan XML (*Extensible Markup Language*) yang merupakan bahasa *markup* kelanjutan dari HTML (*Hypertext Markup Language*). Di dalam Eclipse, XML sudah bisa langsung digabungkan dengan Java dalam satu *Project*. Pengujian aplikasi ini akan menggunakan emulator Android dan juga menggunakan beberapa *smartphone*.

3.2.1 Metode Perancangan

Metode perancangan terbagi menjadi dua bagian yaitu perancangan sistem ataupun aplikasi dan perancangan antar muka. Dalam perancangan sistem ataupun aplikasi metode yang digunakan adalah metode-metode perancangan berorientasi objek dengan menggunakan UML yaitu *use case diagram*, dan *activity diagram*.

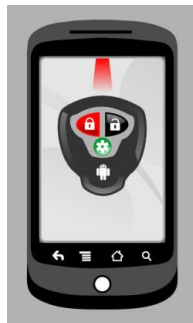
Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan dan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok tool untuk mendukung pengembangan sistem tersebut. Aplikasi ini menggunakan *use case diagram*, dan *activity diagram*.

3.3 Perancangan Antarmuka

Perancangan antarmuka dimaksudkan untuk menjelaskan bagaimana tampilan aplikasi ini dibangun nantinya. Dalam aplikasi akan ada 2 menu utama yang setiap menu memberikan fungsi sesuai fiturnya masing – masing. Adapun 2 menu tersebut adalah Menu Home dan Menu Pengaturan

3.3.1 Perancangan Menu Home

Menu Home aplikasi dapat dilihat pada gambar 3.1.



Gambar 3.1 Perancangan Menu Home

Adapun penjelasan gambar tersebut adalah gambar gembok terkunci berlatar belakang merah untuk mengaktifkan alarm, dan gambar gembok terbuka yang berlatar belakang hitam untuk membuka kunci atau mematikan alarm. Sedangkan gambar bintang berlatar belakang hijau untuk masuk ke menu pengaturan.

3.3.2 Perancangan Menu Pengaturan

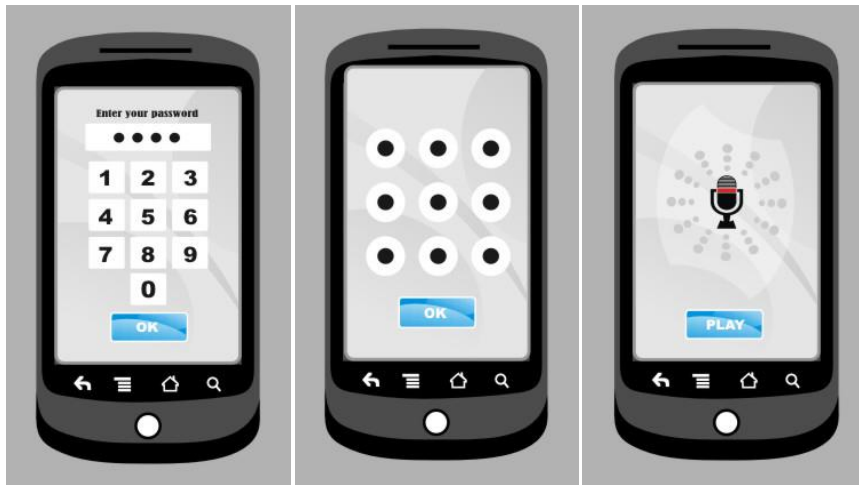
Menu pengaturan akan menampilkan beberapa opsi yang berguna untuk mengatur kerja aplikasi ini diantaranya pemilihan *password*, pemilihan nada, pengaturan sensitivitas, dan *active delay*. Lihat gambar 3.2.



Gambar 3.2 Perancangan Menu Pengaturan

Adapun penjelasan gambar diatas adalah:

1. tombol opsi untuk pengaturan *password*, tombol nada untuk pengaturan nada, tombol sensitifitas untuk pengaturan sensitivitas dan tombol *active delay* untuk pengaturan *active delay*.
2. Jika opsi password ditekan maka pengguna akan dihadapkan pada antar muka menu *password*.
3. Pada menu *Password* akan ditampilkan pilihan *password* yang akan dijadikan sebagai pembuka alarm yaitu terdiri dari *password* berupa *input number*, *pattern* dan *google speech recognition*(suara). Lihat gambar 3.3.



Gambar 3.3 Perancangan input number, pattern dan suara

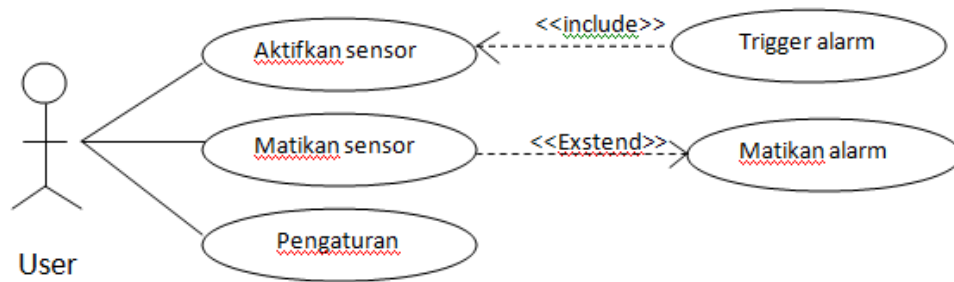
Gambar 3.3 diatas menampilkan perancangan *input number* (kiri), *pattern lock*(tengah), dan suara (kanan).

3.4 Unified Modelling Language(UML)

Unified Modeling Language (UML) adalah sebuah bahasa yang digunakan untuk menentukan, visualisasi, membangun, dan mendokumentasikan sebuah aplikasi. *Diagram* UML yang digunakan pada aplikasi ini adalah *Use CaseDiagram* dan *Activity Diagram*.

3.4.1 Use Case Diagram

Use casediagram merupakan gambaran skenario dari interaksi antara user dengan sistem. Sebuah diagram *use case* menggambarkan hubungan antara *actor*/ pengguna dan kegiatan yang dapat dilakukan terhadap aplikasi. Adapun *use casediagram* dari aplikasi yang akan di bangun biasa dilihat pada Gambar 3.5



Gambar 3.5 Use Case Diagram

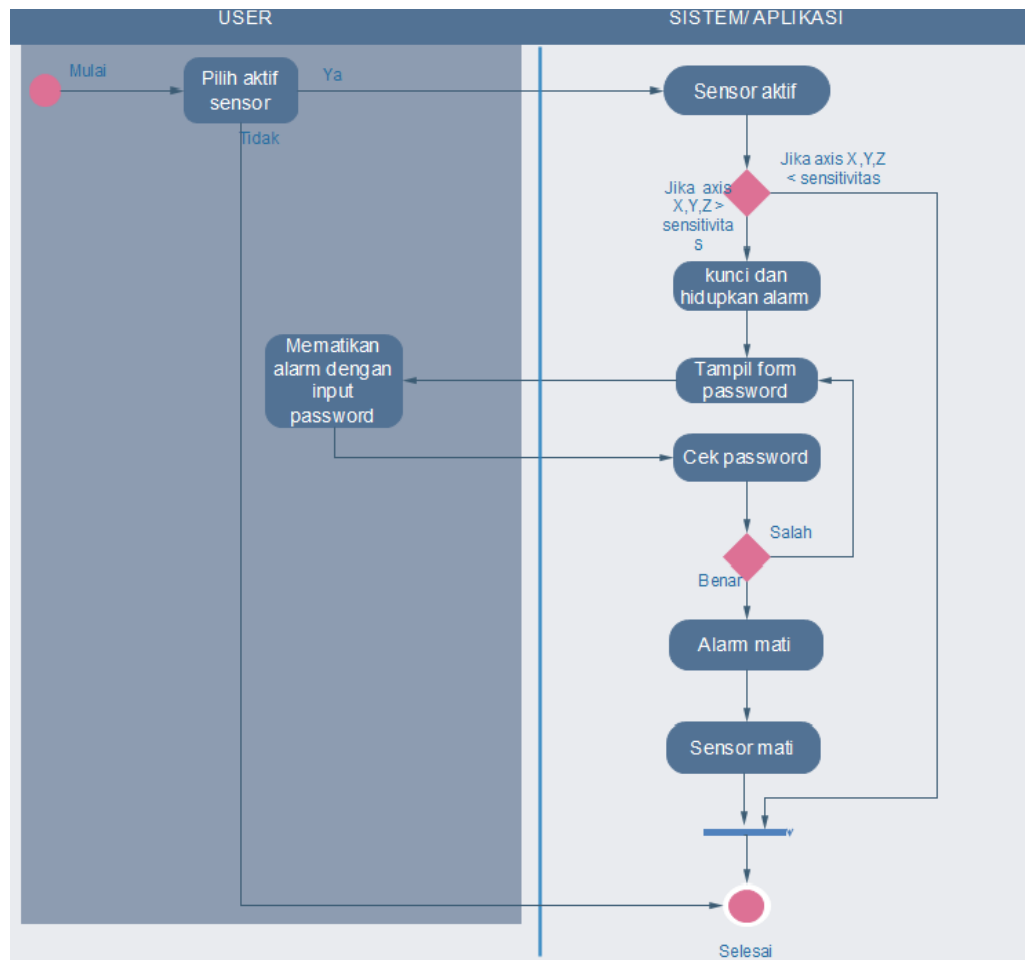
Gambar 3.5 menjelaskan *use case* aplikasi ini dimana pengguna dapat mengakses menghidupkan sensor, mematikan sensor dan mengatur pengaturan.

3.4.2 Activity Diagram

Activity diagram merupakan gambaran proses-proses yang terjadi di dalam menu aplikasi dari aktivitas memulai sampai aktivitas berhenti. Aplikasi ini terdapat 2 *activity diagram* yaitu *activity diagram* aktif sensor dan *activity diagram* pengaturan.

a. Activity Diagram Aktif Sensor

Activity diagram aktif sensor dapat dilihat gambar 3.6



Gambar 3.6 Activity Diagram Aktif Sensor

Gambar 3.6 menjelaskan alur dari aplikasi yang dijalankan Pengguna. Sensor *accelerometer* ini bekerja saat *smartphone* berpindah atau bergerak entah itu jatuh atau berpindah tangan dengan syarat sensor telah diaktifkan, sensor akan mendeteksi pergerakan melalui 3 sumbu axisX, axisY, axisZ jika sumbu axisX, axisY dan axisZ lebih dari sensitif, maka suara alarm akan berbunyi dan layar terkunci.

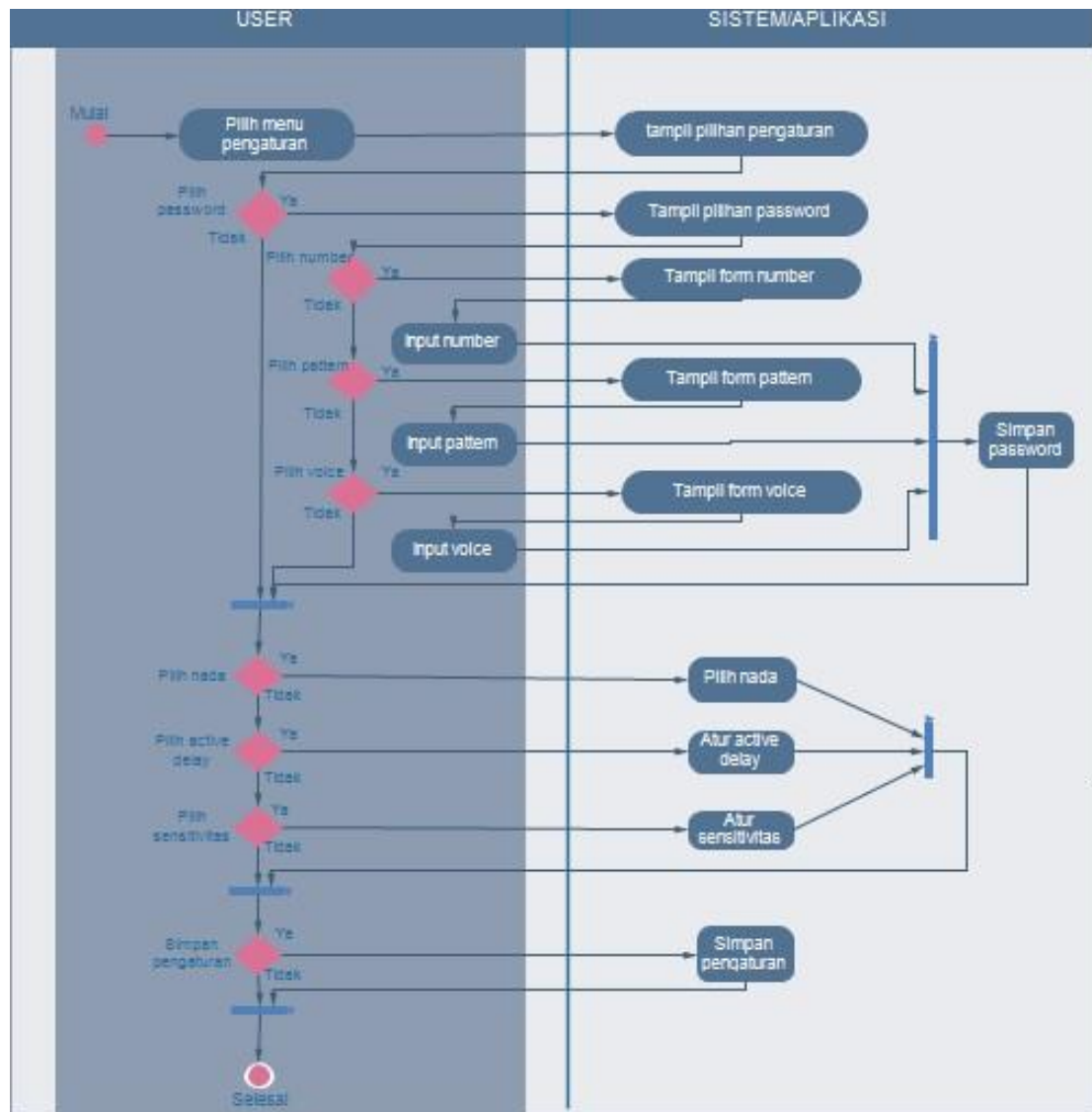
Penjelasan diagram aktif sensor adalah sebagai berikut:

1. Mulai lalu akan tampil pilihan.
2. Apakah *Aktifkan* atau *Nonaktifkan* sensor *alarm*? Pilih (Kunci) maka sensor *Accelerometer* aktif .

3. Jika nilai sensor bertambah karena pergerakan, $axisX > sensitif$ atau $axisY > sensitif$ atau $axisZ > sensitif$, suara alarm berbunyi dan kunci layar dengan *password*, untuk membuka dan menonaktifkan dengan memasukkan *password*.
4. Jika Input password salah maka akan kembali di minta untuk menginput password.
5. Ketika input password benar maka alarm akan berhenti dan stop sensor.
6. Selesai.

b. Activity Diagram Pengaturan

Activity Diagram pengaturan dapat dilihat Gambar 3.7



Gambar 3.7 Activity Diagram Pengaturan

Gambar 3.6 diatas terlihat alur proses menu pengaturan, penjelasan proses menu pengaturan sebagai berikut :

1. Mulai
2. Tampil menu pengaturan
3. Jika memilih *password* akan ada pilihan 3 tipe *password number*, *pattern* dan *google speech* (suara).
4. Memilih nada
5. Memilih *active delay*

6. Memilih sensitivitas
7. Jika *password*, nada, *active delay* dan sensitivitas telah dipilih lalu akan disimpan dan selesai kembali keluar.

3.5 Rencana Pengujian

Rencana dari pengujian aplikasi ini dilakukan menggunakan berbagai merk dan berbagai versi android, dengan maksud untuk mengetahui apakah sistem ataupun aplikasi yang telah dibuat dapat berjalan sesuai dengan rancangan pembuatan aplikasi. Pengujian akan dilakukan pada *smartphone* yang bersistem operasikan android *Jelly Be* dan dengan versi yang berbeda dan dengan *smartphone* yang berbeda merk.

Pengujian yang akan dilakukan pada aplikasi ini adalah:

1. Pengujian beralih halaman.
2. Pengujian aktif sensor.
3. Pengujian tombol *disable volume*.
4. Pengujian tombol *back*.

BAB IV

IMPLEMENTASI DAN ANALISIS

4.1 Perangkat Lunak yang Digunakan

Perangkat lunak yang digunakan dalam pengembangan aplikasi *Alarm Pencegah PencurianSmartphone* berbasis Android dengan menggunakan Fitur *accelerometer* ini adalah :

- a. Sistem operasi : Windows 8
- b. Eclipse ADT dan Android SDK (Wahana,2013).
- c. Corel draw
- d. Emulator android

4.2 Perangkat Keras yang Digunakan

Perangkat keras yang digunakan dalam pengembangan aplikasi *Alarm Pencegahan PencurianSmartphone* ini adalah :

- a. Processor Intel Core TM i3-3227U CPU @1.90GHz
- b. RAM (*Random Acces Memory*) 4 GB atau 4096 MB
- c. Tipe Memory DDR3
- d. Tipe Grafis HD graphic
- e. Hard Disk 500 GB
- f. Monitor
- g. Keyboard
- h. Mouse

Adapun perangkat keras smartphone yang sering digunakan dalam uji coba aplikasi ini adalah Samsung Galaxy Core Duos dengan spesifikasi :

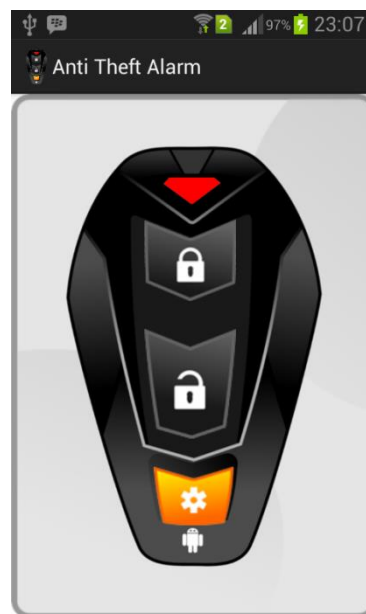
- a. Processor :Dual-core 1.2 GHz Cortex-A5
- b. Memori RAM :1 GB RAM
- c. Penyimpanan Internal : 16 GB
- d. Layar :4.3 inches

4.3 Implementasi

Tahap implementasi adalah tahap pemakaian sistem ataupun aplikasi. Dari implementasi akan diketahui apakah aplikasi berjalan baik atau tidak dan apakah output yang dihasilkan sesuai dengan perancangan yang telah dibuat sebelumnya. Pada sub bab implementasi ini, akan ditampilkan *screenshot* aplikasi dan sedikit penjelasannya. Adapun yang akan dijelaskan pada bagian ini adalah *menu home* atau halaman utama aplikasi, menu pengaturan, menu pemilihan *password* dan tipe *password* serta menu tampilan pada saat pembukaan kunci atau mematikan alarm.

4.3.1 Halaman Menu Home

Halaman menu home menampilkan menu yang terdapat pada aplikasi ini, berikut merupakan tampilan menu home dari aplikasi *alarm* pencegah pencurian *smartphone*.



Gambar 4.1 Halaman Menu Home

Adapun penjelasan dari setiap *button* pada gambar di atas ialah :

1. *Image button* yang bergambar gembok terkunci untuk menghidupkan sensor atau untuk mengaktifkan *alarm*.
2. *Image button* yang bergambar gembok terbuka untuk mematikan sensor atau untuk menonaktifkan *alarm*.

3. *Image button* yang bergambar bintang untuk masuk ke menu pengaturan aplikasi.

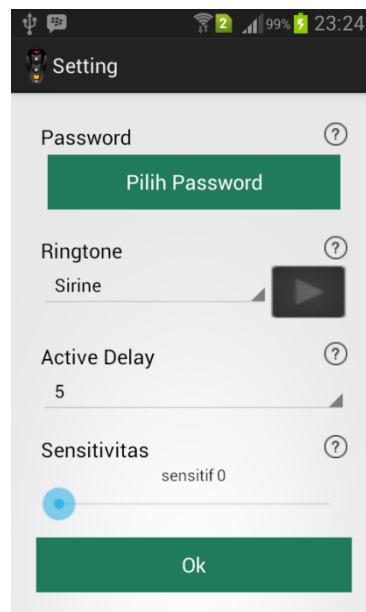
Source code gambar diatas dapat dilihat pada gambar 4.2

```
btnStart.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View arg0) {
        RbHelper.pesan(getApplicationContext(), "start sensor");
        aktif();
        final Handler handler = new Handler();
        Timer t = new Timer();
        t.schedule(new TimerTask() {
            public void run() {
                handler.post(new Runnable() {
                    public void run() {
                        startService(new Intent(MainActivity.this, AccelerometerService.class));
                        notificationMain.createInfoNotification("Start service");
                        ivInfo.setImageResource(R.drawable.red_button);
                        RbHelper.pesan(getApplicationContext(), "aktif sensor");
                        btnStart.setEnabled(false);
                        btnStop.setEnabled(true);
                        btnStart.setCompoundDrawablesWithIntrinsicBounds(imgLock, null, null, null);
                        btnStop.setCompoundDrawablesWithIntrinsicBounds(imgUnlock, null, null, null);
                        saveTombol(true);
                    }
                });
            }
        }, getDelayValue());
        finish();
    }
});
```

Gambar 4.2 Source Code Halaman Menu Home

4.3.2 Halaman Menu Pengaturan

Pada menu pengaturan pengguna dapat mengatur *password* dengan 3 pilihan *password*, pengguna juga dapat mengatur nada, active delay (mengatur lama aktif nya aplikasi setelah di aktifkan) dan pengaturan sensitivitas sensor.



Gambar 4.3Halaman Menu Pengaturan

Setelah pengguna mengatur fitur- fitur yang terdapat pada aplikasi ini yang dapat di lihat pada gambar pengaturan diatas untuk menyimpan nilai yang telah di atur maka cukup menekan tombol Ok.

Source code gambar diatas dapat dilihat pada Gambar 4.4

```

btnSimpan = (Button) findViewById(R.id.buttonSimpan);
btnSimpan.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View arg0) {
        RbHelper.pesan(getApplicationContext(), "simpan");

        idNada = (int) spnNada.getSelectedItemId();
        saveNada(idNada);

        idDelay = (int) spnDelay.getSelectedItemId();
        saveDelay(idDelay);

        RbHelper.pesan(getBaseContext(), "val : " + valueSensitif);
        saveSensitif(valueSensitif);

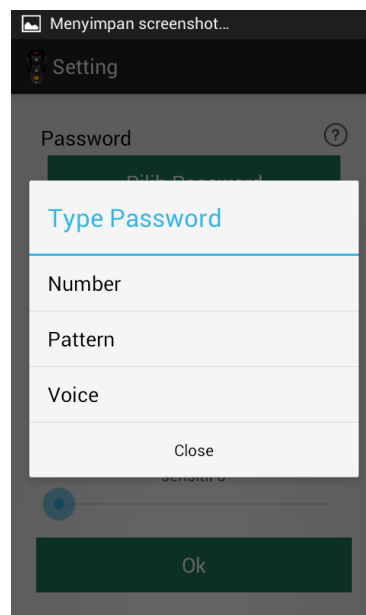
        if (mStartPlaying == true) {
            player.stop();
            mStartPlaying = false;
            player.setLooping(false);
        }
        finish();
    }
});

```

Gambar 4.4Source Code Menu Halaman Pengaturan

4.3.3 Menu Pilih Password

Pada saat *user* ingin mengatur password akan terdapat 3 pilihan tipe *password* yang dapat digunakan. Seperti yang dapat dilihat pada gambar berikut.



Gambar 4.5 Menu Pilih Password

Seperti terlihat pada gambar di atas. Terdapat 3 pilihan *password* yang dapat digunakan yaitu *password* berupa input angka, *password* berupa *pattern* dan *password* berupa *google speech recognition* (suara).

Source code gambar diatas dapat dilihat pada Gambar 4.6

```

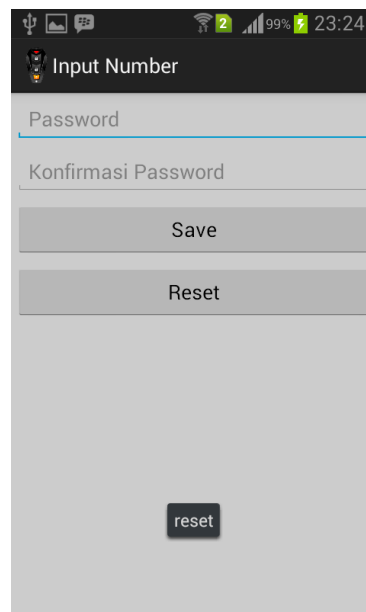
/
public void showDialogPassword() {
    final String[] info = new String[] { "Number", "Pattern", "Voice" };
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setCancelable(false);
    builder.setNeutralButton("Close",
        new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                dialog.dismiss();
            }
        });
    builder.setTitle("Type Password");
    builder.setItems(info, new DialogInterface.OnClickListener() {
        public void onClick(DialogInterface dialog, int id) {
            switch (id) {
                case 0:
                    RbHelper.pesan(getApplicationContext(), "pilih angka " + id);
                    intent = new Intent(getApplicationContext(), PassAngkaInput.class);
                    startActivity(intent);
                    savePassword(id);
                    saveCekPassword(true);
                    break;
                case 1:
                    RbHelper.pesan(getApplicationContext(), "pilih pola " + id);
                    Intent intent = new Intent(
                        LockPatternActivity.ACTION_CREATE_PATTERN, null,
                        PengaturanActivity.this, LockPatternActivity.class);
                    startActivityForResult(intent, REQ_CREATE_PATTERN);
                    savePassword(id);
                    saveCekPassword(true);
                    break;
                case 2:
                    RbHelper.pesan(getApplicationContext(), "pilih suara " + id);
                    intent = new Intent(getApplicationContext(), VoicePassSet.class);

```

Gambar 4.6 Source code Halaman Pilih Password

4.3.4 Menu Password Input Angka

Ketika *user* memilih *password* tipe input angka *user* diminta memasukkan beberapa digit angka untuk dijadikan *password* yang dapat dilihat seperti gambar berikut.



Gambar 4.7 Menu Password Input Angka

Pada gambar diatas terlihat ada 2 kolom yang terletak paling atas. kolom pertama yang paling atas tersebut digunakan untuk pengisian *password* angka sedangkan kolom kedua digunakan untuk pengulangan *password* yang telah di masukan (confirm password). Setelah menginputan angka pilih tombol save untuk menyimpan atau tekan tombol *reset* untuk mengulang masukan *password*.

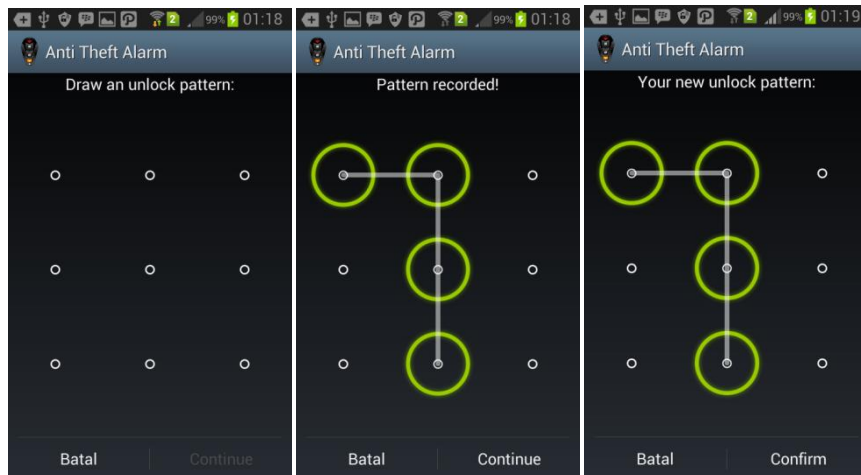
Source code gambar diatas dapat dilihat pada Gambar 4.8

```
case R.id.btnOk:
    getInput = tampil.getText().toString();
    String passw = "" + DbPreference.readInteger(this, DbPreference.ANGKA, 0);
    if (getInput.equals(passw)) {
        stopService(new Intent(getApplicationContext(), AccelerometrService.class));
        DbPreference.writeString(this, DbPreference.varSensor, "nonaktif");
        String varSens = DbPreference.readString(this, DbPreference.varSensor, "");
        Toast.makeText(getApplicationContext(), varSens, Toast.LENGTH_LONG).show();
        finish();
    } else {
        clear();
        tampil.setHint("Password Salah");
        tampil.setHintTextColor(Color.RED);
    }
    RbHelper.pesan(getApplicationContext(), getInput);
    break;
}
```

Gambar 4.8 Source code Menu Password Input Angka

4.3.5 Menu Password Pattern

Ketika user memilih *password* tipe *pattern* user diminta memasukkan *password* dengan menyentuh layar *smartphone* dengan menggeserkan jari sesuai pola yang tersedia. Seperti pada gambar berikut.



Gambar 4.9 Menu Password Pattern

Pada gambar diatas terlihat bentuk tampilan *password pattern*. Berikut penjelasan gambar diatas.

1. Pada gambar yang paling kiri merupakan tampilan pattern yang belum dibuat pola untuk dijadikan *password*.
2. Pada gambar yang ditengah merupakan tampilan *pattern* yang sudah di jadikan suatu pola untuk dijadikan *password*.
3. Pada gambar yang paling kanan merupakan tampilan *pattern* untuk mengkonfirmasi ulang dari pola yang sudah dibuat sebelumnya.
4. Lalu klik *confirm* untuk menggunakan *password pattern* dengan pola yang sudah dibuat.

Source code gambar diatas dapat dilihat pada Gambar4.10

```

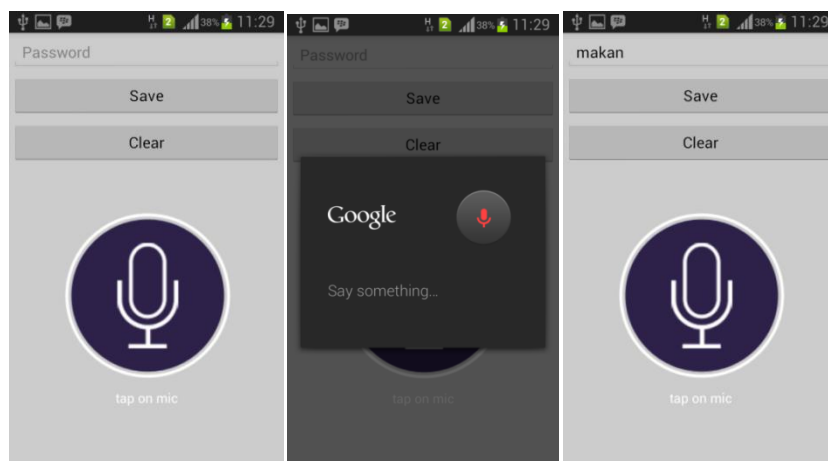
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    switch (requestCode) {
        case REQ_CREATE_PATTERN:
            if (resultCode == RESULT_OK) {
                savedPattern = data.getCharArrayListExtra(LockPatternActivity.EXTRA_PATTERN);
                SecurityPrefs.setPattern(LockPattern.this, savedPattern);
                Log.i("create", String.valueOf(savedPattern));
            }
            break;
        case REQ_ENTER_PATTERN:
            switch (resultCode) {
                case RESULT_OK:
                    Log.i("enter", "RESULT_OK");
                    Toast.makeText(LockPattern.this, "Ok! Cocok", Toast.LENGTH_SHORT).show();
                    stopService(new Intent(getApplicationContext(), AccelerometerService.class));
                    finish();
                    break;
                case RESULT_CANCELED:
                    Log.i("enter", "RESULT_CANCELED");
                    break;
                case LockPatternActivity.RESULT_FAILED:
                    Log.i("enter", "RESULT_FAILED");
                    break;
                case LockPatternActivity.RESULT_FORGOT_PATTERN:
                    Log.i("enter", "RESULT_FORGOT_PATTERN");
                    break;
            }
            break;
        default:
            break;
    }
}

```

Gambar 4.10Source Code Menu Password Pattern

4.3.6 Menu Password Suara

Ketika user memilih *password* suara, user diminta mengatakan suatu kata untuk dijadikan *password*. Dapat dilihat seperti gambar berikut.



Gambar 4.11Menu Password Suara

Penjelasan gambar diatas adalah sebagai berikut :

1. Untuk memulai penggunaan *password* suara sentuh gambar yang berbentuk *mic* yang berlatar belakang biru seperti gambar diatas.
2. Kemudian akan ditampilkan kotak suara google dimana kita dapat mengatakan suatu kata untuk dijadikan *password*.
3. Setelah kata diucapkan, kata akan muncul dikolom paling atas seperti pada gambar, yang pada contoh gambar diatas user menggunakan kata “makan” sebagai kata kunci untuk pembukaan *password*.
4. Kemudian pilih *save* untuk menyimpan kata yang telah diucapkan untuk dijadikan *password*.
5. Atau untuk mengganti *password* pilih *clear*.

Source code gambar diatas dapat dilihat pada gambar 4.12

```
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    switch (requestCode) {
        case REQ_CODE_SPEECH_INPUT: {
            if (resultCode == RESULT_OK && null != data) {
                ArrayList<String> result = data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                txtSpeechInput.setText(result.get(0));
                String hasil = result.get(0);
                String passw = ""+ DbPreference.readString(this, DbPreference.SUARA, "");
                if (hasil.equals(passw)) {
                    stopService(new Intent(getApplicationContext(), AccelerometrService.class));
                    finish();
                } else {
                    txtSpeechInput.setText("Silahkan Coba Lagi..!!!");
                }
            }
            break;
        }
    }
}
```

Gambar 4.12 Source Code Menu Password Suara

4.4 Pengujian Aplikasi

Aplikasi ini diujikan pada beberapa perangkat *smartphone* yang berbeda spesifikasi serta versi android. Adapun perangkat perangkat yang digunakan dalam pengujian aplikasi ini adalah Samsung Galaxy Core, Evercross A7S, Sony Xperia M2, dan Asus Zenphone 5. Spesifikasi singkatnya dapat dilihat pada tabel 4.1.

Tabel 4.1 Perangkat Uji Coba

Perangkat	Versi Android	Layar dan Resolusi	Sensor Accelerometer
Samsung Galaxy Core Duos	jelly bean 4.1.2	4.3 inches, 480 x 800 pixels	ya
Evercross A7S	jelly bean 4.2.2	4.5 inch, 960 x 540 pixels	ya
Sony Eperia M2	jelly bean 4.3	4,8 inci, 960 x 540 pixels	ya
Asus Zenphone 5	kitkat 4.4.2	5.0 inches, 720 x 1280 pixels	ya

Pengujian aplikasi memiliki maksud agar dapat diketahui apakah aplikasi ini dapat berjalan sebagai mana mestinya pada *smartphone* yang berbeda versi android.

Adapun pengujian yang dilakukan yaitu :

1. Pengujian beralih halaman.

Pengujian ini untuk mengetahui apakah .java yang menangani *layout .xml* bisa berjalan pada sistem operasi berbeda dan juga mengetahui apakah halaman bisa dipindah dari halaman satu ke yang lain tanpa kendala.

2. Pengujian halaman Home.

Pengujian ini untuk mengetahui apakah halaman *home* berjalan sebagai mana mestinya.

3. Pengujian menu Pengaturan.

Pengujian ini untuk mengetahui apakah menu pengaturan yang berisi pemilihan *password*, pemilihan nada, pengaturan *active delay* dan sensitivitas berjalan sebagaimana mestinya.

4. Pengujian menu Password.

Pengujian ini untuk mengetahui apakah pilihan *password* yang di sajikan yaitu *input number*, *pattern* dan *password* suara dapat berjalan sebagaimana mestinya.

5. Pengujian Aktif Sensor

Pengujian ini untuk mengetahui apakah sensor *accelerometer* berjalan dengan baik atau tidak.

6. Pengujian Tombol Disable Volume saat Alarm berbunyi

Pengujian ini untuk mengetahui apakah *volume* suara benar – benar tidak dapat di kecilkan saat *alarm* berbunyi.

7. Pengujian Tombol Back

Pengujian ini untuk mengetahui apakah tombol *Back* benar – benar tidak dapat di gumakan saat *alarm* berbunyi.

4.5 Hasil Pengujian

Dari Pengujian yang dilakukan pada beberapa perangkat android, maka didapatkan hasil bahwa Aplikasi *Alarm* Pencegah Pencurian *Smartphone* Berbasis Android ini bisa berjalan lancar tanpa kendala pada semua pengujian poin 1, 2,3,4,5,6 dan 7.

Dari apa yang di terangkan di atas maka penulis membuat tabel untuk mempermudah menjelaskan hasil pengujian yang dapat dilihat pada tabel 4.2.

Tabel 4.2 HasilPengujian

No	Samsung	Evercross	Sony Eperia	Asus Zenphone
1	☒	☒	☒	☒
2	☒	☒	☒	☒
3	☒	☒	☒	☒
4	☒	☒	☒	☒
5	☒	☒	☒	☒
6	☒	☒	☒	☒
7	☒	☒	☒	☒

Dari tabel pengujian tersebut maka didapatkan penjelasan hasilnya sebagai berikut :

1. Pada percobaan 1 semua perangkat tidak mengalami kendala apapun dalam berpindah halaman.
2. Pada percobaan 2 semua perangkat tidak mengalami kendala apapun saat mengakses menu Home.
3. Pada percobaan 3 semua perangkat tidak mengalami kendala apapun saat melakukan uji fitur menu Pengaturan.
4. Pada percobaan 4 semua perangkat tidak mengalami kendala apapun saat melakukan uji fitur menu Pilihan *Password*.
5. Pada percobaan 5 semua perangkat tidak mengalami kendala apapun saat melakukan uji fitur Aktif Sensor.
6. Pada percobaan 6 semua perangkat tidak mengalami kendala apapun saat melakukan uji *Disable Volume* saat *Alarm* Berbunyi.
7. Pada percobaan 7 semua perangkat tidak mengalami kendala apapun saat melakukan uji Tombol *Back* saat *Alarm* Berbunyi.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melalui beberapa tahap dalam pembuatan Aplikasi *Alarm* Pencegah Pencurian *Smartphone* Berbasis Android ini, maka dapat disimpulkan bahwa:

1. Aplikasi ini membantu pengguna dalam menjaga *smartphonenya* dari pencurian atau kehilangan oleh pihak – pihak yang tidak bertanggung jawab.
2. *Password* suara tidak mencakup keunikan deteksi suara, *password* suara hanya mendeteksi kata yang diucapkan.
3. Dalam kondisi apapun saat aplikasi telah dihidupkan ketika terjadi gerakan sensor akan bekerja dan *alarm* akan berbunyi.

5.2 Saran

Berdasarkan pengujian yang dilakukan pada aplikasi *Alarm* Pencegah Pencurian *Smartphone* ini maka ditemukan kekurangan dan juga ditemukan potensi didalamnya. Oleh karena hal – hal tersebut maka berikut ini saran serta masukan yang penulis berikan dengan harapan aplikasi ini dapat dikembangkan lebih lanjut dan menjadi lebih baik lagi.

Adapun saran dan masukan tersebut adalah :

1. Menambahkan fitur untuk *disable* tombol *home* karena pada saat penulis membuat aplikasi ini belum ditemukan cara untuk menonaktifkan tombol *home*.
2. Masih ada celah untuk mengakses tombol power pada saat *alarm* berbunyi.
3. Saat aplikasi berbunyi atau aktif ketika *smartphone* dimatikan maka aplikasi juga akan berhenti.

DAFTAR PUSTAKA

Hermawan, Ade. 2012. 30 Software yang Sedang Berkembang. From : <http://www.adehermawan.com/2012/10/30-software-yang-sedang-berkembang.html>, diakses Agustus 2014.

Komputer, Wahana. 2013. Android : Programming with Eclipse. Yogyakarta : Penerbit Andi.

Kroenke, David M. 2005. *Database Processing Jilid 1 edisi 9*. Jakarta : Penerbit Erlangga.

Safaat, Nasrudin, 2012. Android : Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Bandung : Informatika Bandung.

Wiryadinata, Romi. 2009. Prinsip Kerja Accelerometer. From : <http://wiryadinata.web.id/?p=22>, diakses Juli 2014.