BAB 3
ANALISIS DAN PERANCANGAN SISTEM

Analisis dan perancangan sistem memerlukan tahapan yang sistematis untuk mendapatkan aplikasi yang baik dan bersesuaian dengan kegunaan dan tujuannya. Tahap awal dari analisis adalah menganalisis kebutuhan-kebutuhan sistem mulai dari kebutuhan pengguna, kebutuhan non fungsional, dan kebutuhan fungsional. Sedangkan untuk tahap perancangan aplikasi yaitu perancangan sistem, dan perancangan antarmuka.

3.1 Analisis Sistem

Analisis sistem merupakan kegiatan penguraian suatu sistem informasiyang utuh dan nyata ke dalam bagian-bagian atau komponen-komponen komputer yang bertujuan untuk mengidentifikasi serta mengevaluasi masalah-masalah yangmuncul, hambatan-hambatan yang akan mungkin terjadi dan kebutuhan-kebutuhanyang diharapkan sehingga mengarah kepada solusi untuk perbaikanmaupun pengembangan ke arah yang lebih baik dan sesuai dengan kebutuhanserta perkembangan teknologi. Adapun tahapan-tahapan yang ada pada analisa sistem ini meliputi: analisis masalah, analisis metode, analisis kebutuhan sistem.

3.1.1 Analisis Masalah

Analisis masalah dilakukan dengan menguraikan masalah-masalah yang terjadi sehingga permasalahan ini dapat diketahui melalui hasil dari membaca buku dan referensi melalui *website*. Berikut ini adalah hasil analisis masalah yang didapat:

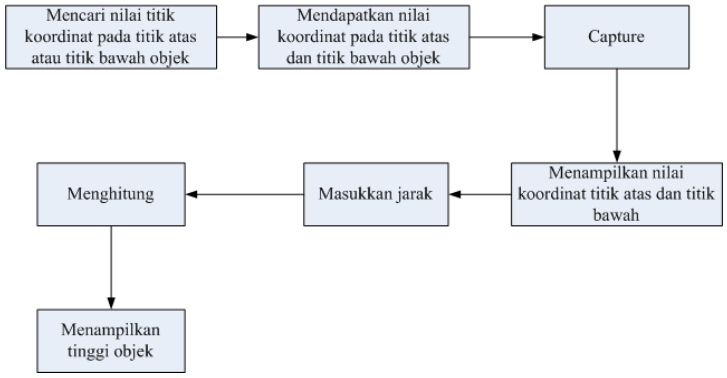
1. Keterbatasan alat ukur dan kondisi yang digunakan pada objek yang tinggi seperti menara dan gedung, sehingga menyulitkan pada saat pengukuran.
2. Pengukuran ketinggian hanya bisa dilakukan oleh ahli, sehingga tidak sembarang orang dapat melakukan pengukuran dengan tepat.

3.1.2 Analisis Metode

Analisis metode digunakan untuk mengetahui alur proses dari sebuah metode yang digunakan dapat diterapkan ke dalam aplikasi yang dibangun. Pembangunan aplikasi ini menggunakan metode Trigonometri dengan memanfaatkan *accelerometer* pada sistem operasi android untuk mempermudah dalam perhitungan tinggi objek.

3.1.2.1 Analisis Sistem yang akan dibangun

Sistem yang akan dibangun adalah sistem aplikasi pengukuran tinggi objek. Aplikasi ini dapat mengukur tinggian objek dari yang paling rendah sampai yang paling tinggi dengan memanfaatkan sensor *accelerometer* yang terdapat pada sistem operasi android untuk mencari nilai koordinat kemiringan ponsel, adapun gambaran umum tentang sistem yang akan dibanguna dapat dilihat pada gambar dibawah ini



Gambar 3.1 Analisi Sistem yang akan dibangun

Berikut penjelasan alur proses sistem yang akan dibangun:

1. Pengguna mengarahkan titik fokus untuk mencari titik koordinat kemiringan posel dengan memanfaatkan sensor *accelerometer* yang terdapat pada sistem operasi android.
2. Sistem mencari titik koordinat kemiringan ponsel
3. Setelah mendapatkan nilai koordinat kemiringan pengguna dapat mengambil dengan mengklik capture yang terdapat pada aplikasi
4. Sistem akan menampilkan nilai koordinat kemiringan

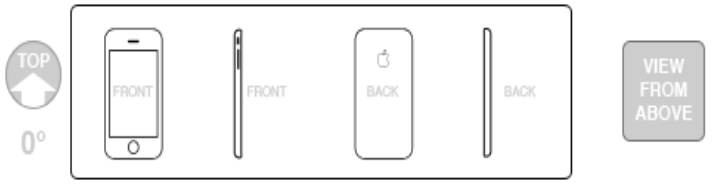
5. Pengguna dapat memasukkan jarak posel dengan jaral objek yang akan dicari tingginya
6. Pengguna dapat pengklik perhitungan untuk mencari tinggi objekSistem memproses perhitungan dan menampilkan hasil nilai tinggi objek

3.1.2.2 Analisis *Accelerometer*

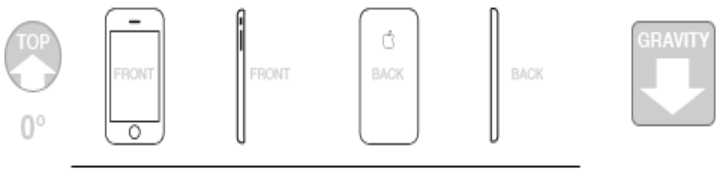
Accelerometer modern tidak lain adalah MEMS (*micro electro mechanical system*) berskala kecil. *Accelerometer* adalah suatu alat untuk mengukur percepatan, mendeteksi dan mengukur getaran, atau untuk mengukur percepatan garvitasi(inklinasi).

Pengukuran bisa secara analog maupun digital. *Accelerometer* dapat digunakan untuk mengukur percepatan baik statis maupun dinamis. *Accelerometer* akan mengalami percepatan dalam kisara dari -1g sampai +1g (9.8m/s), dan hinggan kemiringan 180°.

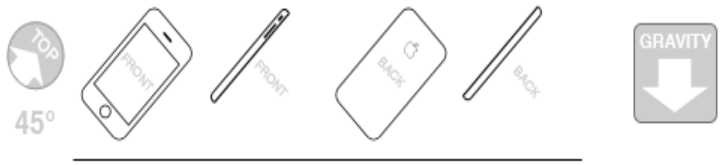
Pada *smartphone accelerometer* merupakan sensor yang bisa membaca pergerakan sehingga dapat mengubah tampilan layar dari posisi *landscape* ke *portrait* atau sebaliknya dengan cukup memiringkan badan ponsel secara otomatis, berikut nilai-nilai kemiringan sensor *accelerometer x,y,z.*



X = 0.00 Y = 0.00 Z = -1.00	X = 1.00 Y = 0.00 Z = 0.00	X = 0.00 Y = 0.00 Z = 1.00	X = -1.00 Y = 0.00 Z = 0.00



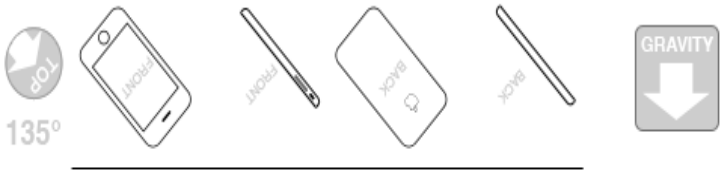
X = 0.00 Y = -1.00 Z = 0.00	X = 0.00 Y = -1.00 Z = 0.00	X = 0.00 Y = -1.00 Z = 0.00	X = 0.00 Y = -1.00 Z = 0.00



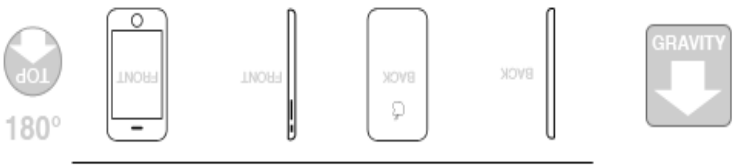
X = 0.75 Y = -0.75 Z = 0.00	X = 0.00 Y = 0.75 Z = 0.75	X = -0.75 Y = -0.75 Z = 0.00	X = 0.00 Y = -0.75 Z = -0.75



X = 1.00 Y = 0.00 Z = 0.00	X = 0.00 Y = 0.00 Z = 1.00	X = -1.00 Y = 0.00 Z = 0.00	X = 0..00 Y = 0.00 Z = -1.00

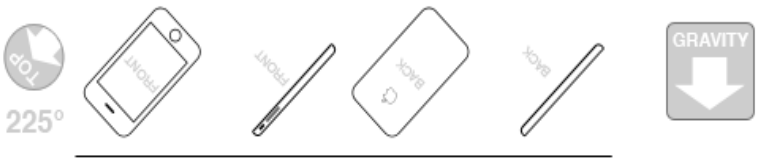


X = 0.75 Y = 0.75 Z = 0.00	X = 0.00 Y = 0.75 Z = 0.75	X = -0.75 Y = 0.75 Z = 0.00	X = 0.00 Y = 0.75 Z = -0.75

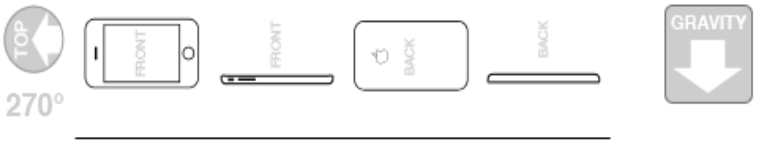


X = 0.00 Y = 1.00	X = 0.00 Y = 1.00	X = 0.00 Y = 1.00	X = 0.00 Y = 1.00

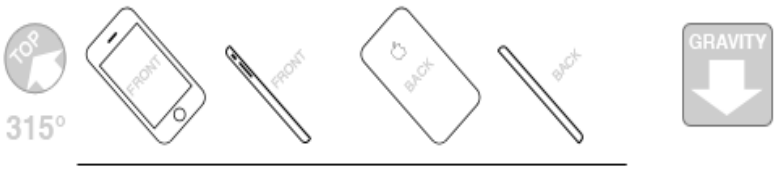
Z = 0.00	Z = 0.00	Z = 0.00	Z = 0.00



X = -0.75 Y = 0.75 Z = 0.00	X = 0.00 Y = 0.75 Z = -0.75	X = 0.75 Y = 0.75 Z = 0.00	X = 0.00 Y = 0.75 Z = 0.75



X = 0.0 Y = 0.0 Z = -1.0	X = 0.0 Y = 0.0 Z = -1.0	X = 0.0 Y = 0.0 Z = 1.0	X = 0.0 Y = 0.0 Z = 1.0

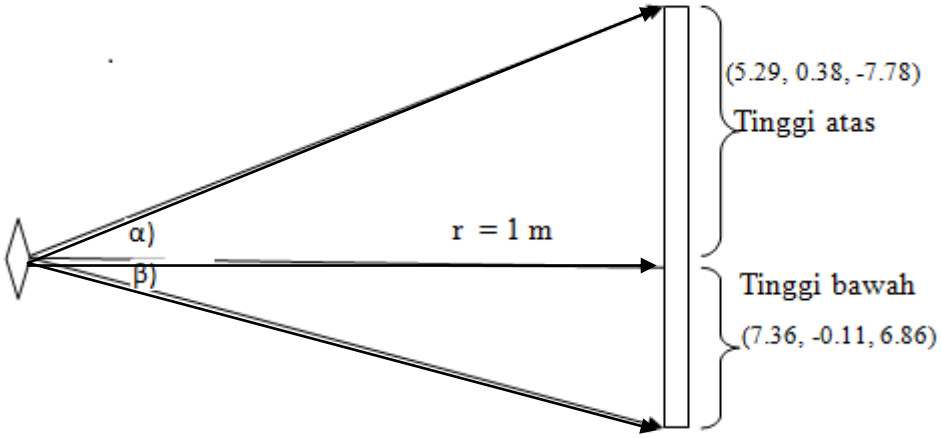


X = -0.75 Y = -0.75 Z = 0.00	X = 0.00 Y = -0.75 Z = -0.75	X = 0.75 Y = -0.75 Z = 0.00	X = 0.00 Y = -0.75 Z = 0.75

3.1.2.3 Analisis Metode Trigonometri

Objek yangdijadikan ilustrasi adalah tinggi pintu 2.4m yang diukur dengan alat meteran, dan akan dibandingkan dengan metode trigonometri yang terdapat pada aplikasi pengukuran tinggi objek dengan memanfaatkan *accelerometer* pada sistem operasi android yang akan dibangun

Analisi metode trigonometri pada aplikasi pengukuran tinggi objek dengan memanfaatkan *accelerometer* pada sistem operasi android, untuk mendapatkan tinggi objek pengguna harus mencari koordinat (x,y,z) kemiringan ponsel dengan memanfaatkan sensor *accelerometer* yang terdapat pada ponsel dan pengguna memasukkan jarak ponsel dengan objek r = 1 m



Setelah mendapatkan nilai koordinat kemiringan titik atas dan titik bawah maka dapat dicari alpha dan beta tinggi objek terlebih dahulu dengan rumus sebagai berikut

1. Mencari Nilai Alpha

$$\alpha = \text{acos}\left(\frac{Vtop_x}{\sqrt{Vtop_x^2 + Vtop_y^2 + Vtop_z^2}}\right)$$

$$\alpha = \text{acos}\left(\frac{5.29}{\sqrt{5.29^2 + 0.38^2 + (-7.78^2)}}\right)$$

$$\alpha = \text{acos}(0.561)$$

$$\alpha = 55.818$$

2. Mencari Nilai Beta

$$\beta = \text{acos}\left(\frac{Vbottom_x}{\sqrt{Vbottom_x^2 + Vbottom_y^2 + Vbottom_z^2}}\right)$$

$$\beta = \text{acos}\left(\frac{7.36}{\sqrt{7.36^2 + (-0.11^2) + 6.86^2}}\right)$$

$$\beta = \text{acos}(0.731)$$

$$\beta = 42.989$$

Setelah itu dapat dimasukkan jarak objek dengan ponsel secara manual dengan jarak r = 1

Setelah mendapatkan nilai alpha dan beta makan akan dicari nilai tinggi atas dan nilai tinggi bawah dengan menggunakan rumus sebagai berikut,

3. Tinggi Atas

   Tinggi atas = tan (α) * r

   Tinggi atas = tan (55.818)*1 = 1.4724

4. Tinggi Bawah

   Tinggi bawah = tan (β) * r

   Tinggi bawah = tan (42.989) * 1 = 0.9321

Setelah mendapatkan tinggi atas dan tinggi bawah maka dapat dijemlahkan untuk mengetahui tinggi objek tersebut dengan rumur sebagai berikut

Tinggi objek = tinggi atas + tinggi bawah

Tinggi objek = 1.4724 + 0.9321 = 2.4045 (m)

3.1.3 Analisis Kebutuhan Sistem

Kebutuhan sistem yang diperlukan dan yang digunakan dalam pembuatan sistem ini terdiri dari beberapa bagian, yaitu: kebutuhan fungsional dan kebutuhan non fungsional.

Tabel 3.1 Kebutuhan Fungsional

Nomor	Keterangan
SKPL-F-01	Sistem dapat menghasilkan kemiringan ponsel dengan memanfaatkan sensor *accelerometer*
SKPL-F-02	Sistem dapat menghasilkan tinggi objek dari hasil kemiringan ponsel

Tabel 3.2 Kebutuhan Non-Fungsional

Nomor	Keterangan
SKPL-NF-01	Sistem dibangun dalam bentuk aplikasi *mobile*
SKPL-NF-02	Sistem dapat dijalankan pada ponsel android minimal versi 2.1 (*froyo*)
SKPL-NF-03	Pada ponselharus memiliki kamera belakang
SKPL-NF-04	Ponselharus memiliki sensor *accelerometer* untuk mencari hasil kemiringan ponsel

3.1.3.1 Analisis Kebutuhan Non Fungsional

a. Spesifikasi Perangkat Lunak

Kebutuhan perangkat lunak ini terdiri dari spesifikasi perangkat lunak yang digunakan dalam membangun dan mengimplementasikan aplikasi yang dibangun dalam penelitian. Spesifikasi kebutuhan perangkat lunak yang digunakan dalam membangun aplikasi pengukuran tinggi objek dapat dilihat pada tabel 3.3.

Tabel 3.3 Spesifikasi Kebutuhan Perangkat lunak

No	Perangkat Lunak
1	JDK v 1.6 Android SDK
2	ADT (*Android Development Tools*)
3	AVD (*Android Virrtual Device*)
4	Sistem operasi android v 2.2 keatas

b. Spesifikasi Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini yaitu komputer dan *smartphone.* Komputer digunakan dalam merancang dan membangun aplikasi pengukuran tinggi objek. Sedangkan *smartphone* digunakan sebagai alat untuk implementasi aplikasi pengukuran tinggi objek yang telah dibangun pada penelitian ini.

Adapun kebutuhan spesifikasi minimum komputer dan *smartphone* yang harus digunakan dalam penelitian ini dapat dilihat pada tabel 3.4 dan tabel 3.5

Tabel 3.4 Spesifikasi Perangkat Keras Komputer

No	Spesifikasi
1	Intel(R) Core(TM) i5 CPU
2	RAM 2 GB
3	Harddisk 500 GB
4	Mouse dan Keyboard

Tabel 3.5 Spesifikasi Perangkat Keras *Smartphone*

No	Spesifikasi
1	Memori Internal 1 GB
2	Memori Eksternal Up to 32GB
3	Layar 5 inchi

c. Spesifikasi Pengguna (*User*)

Spesifikasi pengguna adalah kriteria yang harus dimiliki oleh pengguna agar tidak kesulitan dalam menjalankan aplikasi pengukuran tinggi objek ini. Spesifikasi pengguna adalah sebagai berikut:

Tabel 3.6 Spesifikasi Penggunan (U*ser*)

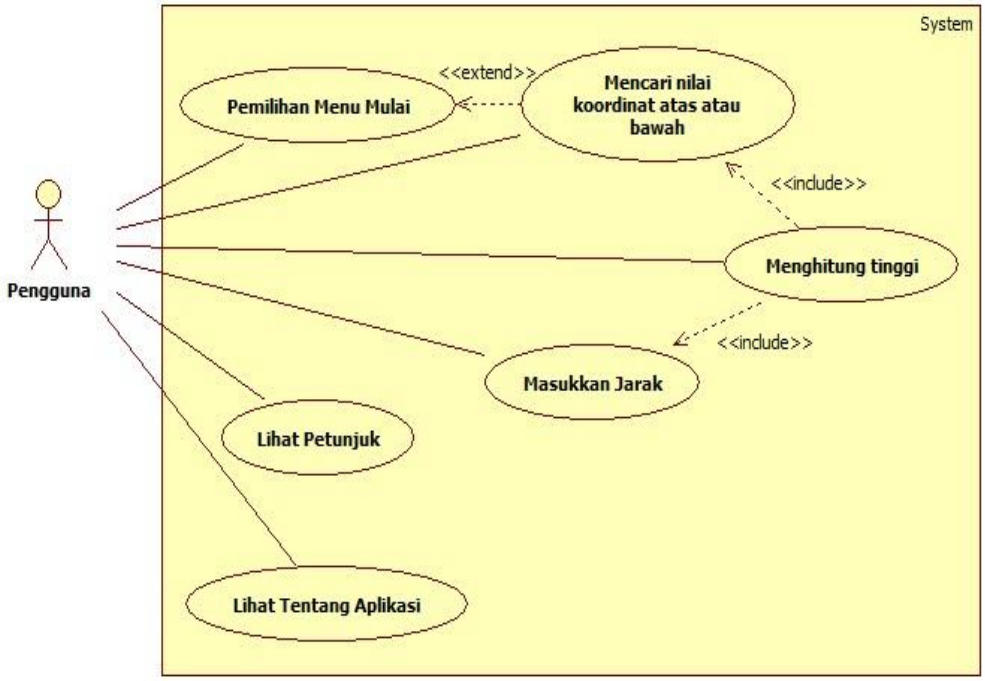
Syarat	Pengguna
Mengerti Operasional *Smartphone*	√
Dapat mengoperasikan Aplikasi pada *smartphone*	√
Minimal Memilik *smartphone Android 2.1(froyo)*	√

3.1.3.2 Analisi Kebutuahan 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 sesuai kebutuhan. Analisis kebutuhan fungsional dimodelkan dengan menggunakan UML (*Unfied Modelling Language*). Dimana tahap-tahap perancangan yang dilakukan dalam membangun perangkat lunak pencocokan string antara lain *Use Case Diagram, Activity Diagram, Class Diagram, Sequence Diagram.*

3.1.3.2.1 Use Case Diagram

Pemodelan yang akan digunakan dalam menggambarkan kebutuhan fungsional pada aplikasi yang dibangun yaitu dengan pemodelan Use Case Diagram. Use case diagram pada aplikasi pengukuran tinggi objek adalah sebagi berikut:



Gambar 3.2 Use Case Diagram

1. Identifikasi Aktor

Identifikasi aktor adalah Aktor yang berperan dalam menjalankan sistem, aktor yang berperan dapat dilihat pada tabel 3.7

Tabel 3.7 Idetifikasi Aktor

No	Aktor	Deskripsi
A-01	Pengguna	Merupakan aktor yang berperan dalam menggunakan aplikasi pengukuran tinggi objek untuk mendapatkan berapa tinggi objek yang ingin diketahui

2. Identifikasi Use Case

Identifikasi *Use Case* adalah *Use Case* yang bekerja pada sistem aplikasi dan *Use Case* yang bekerja pada sistem adalah sebagai berikut:

Tabel 3.8 Identifikasi Use Case

No	Nama Use Case	Deskripsi
UC-01	Pilih menu mulai	Fungsionalitas untuk menjalankan aplikasi pengukuran tinggi objek
UC-02	Lihat petunjuk	Fungsionalitas untuk melihat petunjuk cara menggunakan aplikasi pengukuran tinggi objek
UC-03	Lihat tentang aplikasi	Fungsionalitas untuk melihat tentang aplikasi pengukuran tingii objek
UC-04	Mencari nilai koordinat atas atau bawah	Fungsionalitas untuk mendapatkan nilai kemiringan *accelerometer* pada titik atas atau titik bawah
UC-05	Masukkan jarak	Pengguna memasukkan jarak secara manual
UC-06	Menghitung tinggi	Funsionalitas untuk mendapatkan nilai tinggi objek yang dituju

3. Skenario Use Case

Skenario Use Case adalah alur cerita atau proses-proses yang terjadi pada sistem antara aktor dengan Use Case. Skenario Use Case ditujukkan pada table-tabel sebagai berikut:

Tabel 3.9 Skenario *Use Case* Mulai

Identifikasi	
Nomor	UC-01
Nama	Pilih menu mulai
Tujuan	Untuk menjalankan aplikasi pengukuran tinggi objek
Aktor	Pengguna
Skenarioa Utama	
Kondisi Awal	Menu utama ditampilkan

Aksi Aktor	Reaksi Sistem
1. Klik menu mulai	
	2. Menampilkan aplikasi pengukuran tinggi objek
Kondisi Akhir	Pengukuran dimulai

Tabel 3.10 Skenario *Use Case* Petunjuk

Identifikasi	
Nomor	UC-02
Nama	Lihat petunjuk
Tujuan	Melihat cara menggunakan aplikasi pengukuran tinggi objek
Aktor	Pengguna
Skenarioa Utama	
Kondisi Awal	Menu utama ditampilkan
Aksi Aktor	Reaksi Sistem
1. Klik petunjuk	
	2. menampilkan petunjuk cara menggunakan aplikasi
Kondisi Akhir	Menampilkan cara menggunakan aplikasi

Tabel 3.11 Skenario *Use Case* Tentang

Identifikasi	
Nomor	UC-03
Nama	Lihat tentang aplikasi
Tujuan	Melihat informasi tentang aplikasi
Aktor	Pengguna
Skenarioa Utama	
Kondisi Awal	Menu utama ditampilkan
Aksi Aktor	Reaksi Sistem
1. Klik tentang	
	2. menampilkant tentang aplikasi
Kondisi Akhir	Menampilkan informasi tentang aplikasi pengyukuran tinggi objek

Tabel 3.12 Skenario *Use Case* mencari nilai koordinat atas atau bawah

Identifikasi	
Nomor	UC-04
Nama	Mencari nilai koordinat atas atau bawah
Tujuan	Untuk mendapatkan nilai kemiringan *accelerometer* padatitik atas objek atau titik bawah objek

Aktor	Pengguna	
Skenarioa Utama		
Kondisi Awal	Pengambilan nilai kemiringan	
Aksi Aktor		Reaksi Sistem
1. Pengguna mengarahkan titik fokus yang terdapat pada aplikasi untuk mencari nilai kemiringan sensor *accelerometer* pada titik atas atau titik bawa objek		
		2. Sistem mencari nilai koordinat kemiringan sensor *accelerometer* pada titik atas atau titik bawah
		3. Sistem mendapatkan nilai koordinat kemiringan sensor *accelerometer* titik atas dan titik bawah
4. Pengguna mengambil titik kemiringan jika sudah tepat pada titik atas atau titik bawah objek		
		5. Sistem menampilkan nilai koordinat sensor *accelerometer* yang telah diambil oleh pengguna
		6. Sistem menyimpan hasil nilai koordinat kemiringan sensor *accelerometer*
Kondisi Akhir	Menampilkan nilai koordinat kemiringan titik atas dan titik bawah objek	

Tabel 3.13 Skenario *Use Case* masukkan jarak

Identifikasi	
Nomor	UC-05
Nama	Masukkan jarak
Tujuan	Untuk mendapatkan nilai tinggi objek
Aktor	Pengguna
Skenarioa Utama	
Kondisi Awal	Masukkan jarak objek pada ponsel
Aksi Aktor	Reaksi Sistem
1. pengguna memasukkan jarak ponsel ke objek	

yang akan dicari tingginya	
	2. Sistem menyimpan nilai jarak yang dimasukkan oleh pengguna
Skenario Alternatif	
Aksi Aktor	Reaksi Sistem
	1. Menampilkan pesan "Jarak tidak boleh kosong"
2. Masukkan jarak	
Kondisi Akhir	Jarak yang dimasukkan sudah tersimpan

Tabel 3.14Skenario *Use Case* menghitung tinggi

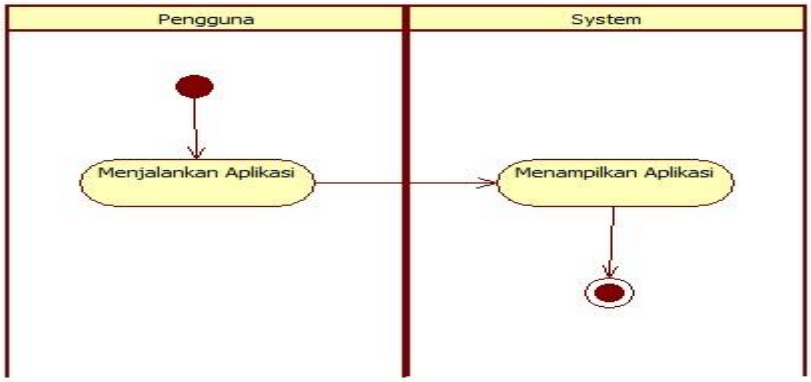
Identifikasi	
Nomor	UC-06
Nama	Menghitung tinggi
Tujuan	Untuk mendapatkan nilai tinggi objek
Aktor	Pengguna
Skenarioa Utama	
Kondisi Awal	Meproses perhitungan
Aksi Aktor	Reaksi Sistem
1. Pengguna mengklik tombol hitung untuk memprose perhitungan	
	2. Sistem mengambil hasil nilai kemiringan titik atas objek
	3. Sistem mengambil hasil nilai kemiringan titik bawah objek
	4. Sistem mengambil nilai jarak yang telah dimasukkan oleh pengguna
	5. Sistem memproses perhitungan tinggi objek
	6. Mendapatkan nilai tinggi objek
Kondisi Akhir	Menampilkan nilai tinggi objek yang telah diproses oleh sistem

3.1.3.2.2 Activity Diagram

Activity diagram adalah teknik untuk mendeskripsikan logika procedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram mempunyai peran seperti halnya *flowchart*, akan tetapi perbedannya

dengan *flowchart* adalah activity diagram bisa mendukung perilaku parallel sedangkan *flowchart* tidak bisa.

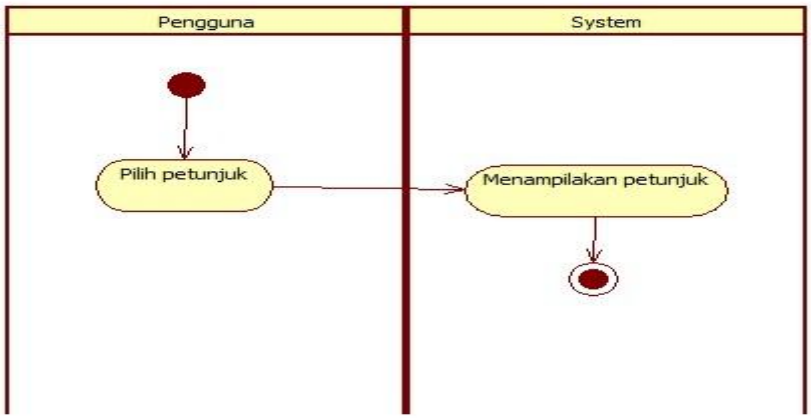
1. *Activity* DiagramPilih Menu Mulai



Gambar 3.3 *activity* diagram pilih menu mulai

Gambar diatas merupakan *activity diagram* yang memperlihakan aktivitas dan interaksi antara pengguna dan sistem untuk menjalankan aplikasi pengukuran tinggi objek

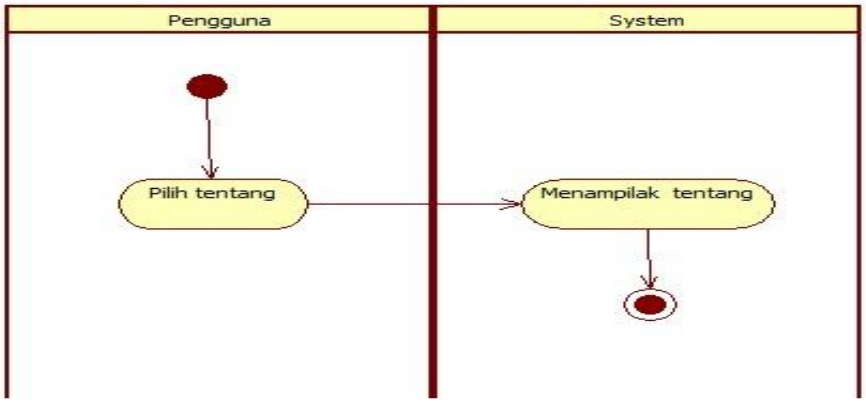
2. *Activity* Diagram Lihat Petunjuk



Gambar 3.4 *activity* diagram lihat petunjuk

Gambar diatas merupakan *activity diagram* yang memperlihakan aktivitas dan interaksi antara pengguna dan sistem untuk petunjuk cara menggunaka aplikasi pengukuran tinggi objek

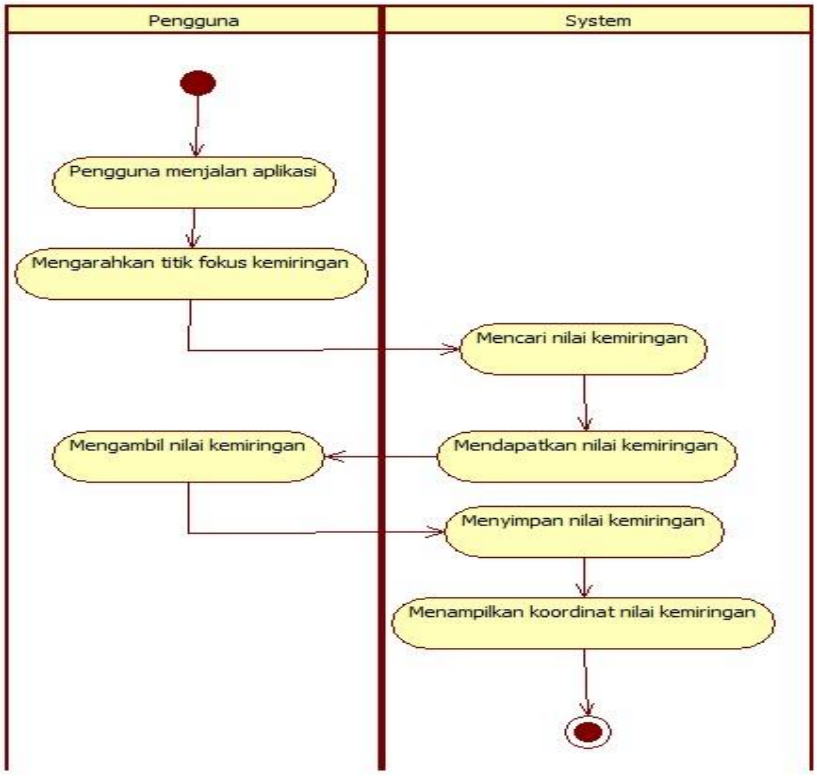
3. *Activity* DiagramLihat Tentang Aplikasi



Gambar 3.5 *activity* diagram lihat tentang aplikasi

Gambar diatas merupakan *activity diagram* yang memperlihakan aktivitas dan interaksi antara pengguna dan sistem untuk mengetaui tentang aplikasi pengukuran tinggi objek

4. *Activity Diagram* Mencari nilai koordinat atas atau bawah

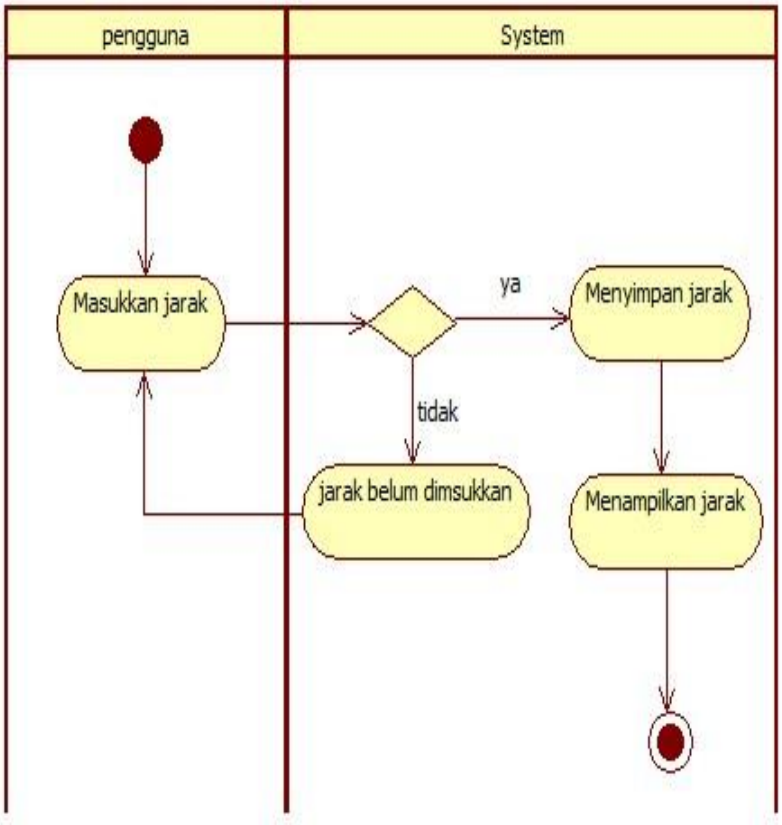


Gambar 3.6 *activity diagram* mendapatkan koordinat *accelerometer*

Gambar diatas merupakan *activity diagram*yang memperlihatkan aktivitas dan interaksi antara pengguna dan sistem untuk menjalankan aplikasi pengukuran tinggi objek

Setelah mendapatkan nilai kemiringan titik atau titik bawah objek maka pengguna memasukkan jarak objek secara manual

5. *Activity* Diagram *M*asukkan Jarak

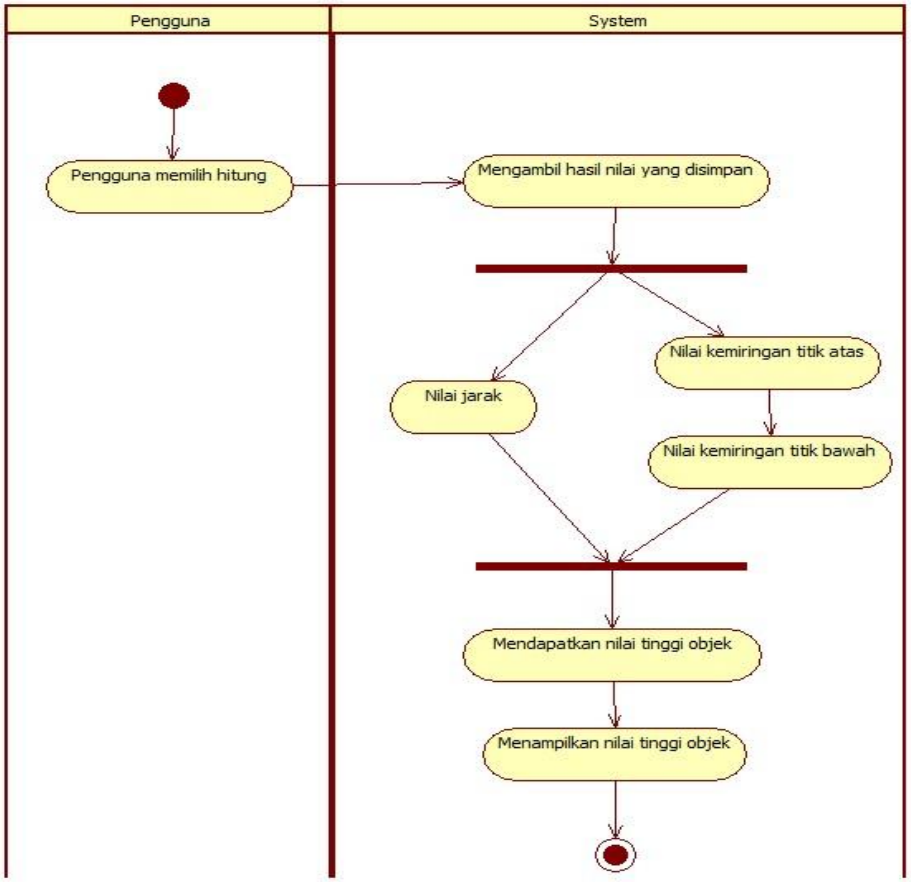


Gambar 3.7 *activity diagram* mesukkan jarak

Gambar diatas merupakan *activity diagram*yang memperlihatkan aktivitas dan interaksi antara pengguna dan sistem untuk mamasukkan jarak objek secara manual

Setalah memasukkan jarak objek maka pengguna dapat menjalankan proses perhitungan yang terdapat pada aplikasi pengukuraun tinggi objek.

6. *Activity* DiagramMenghitung Tinggi

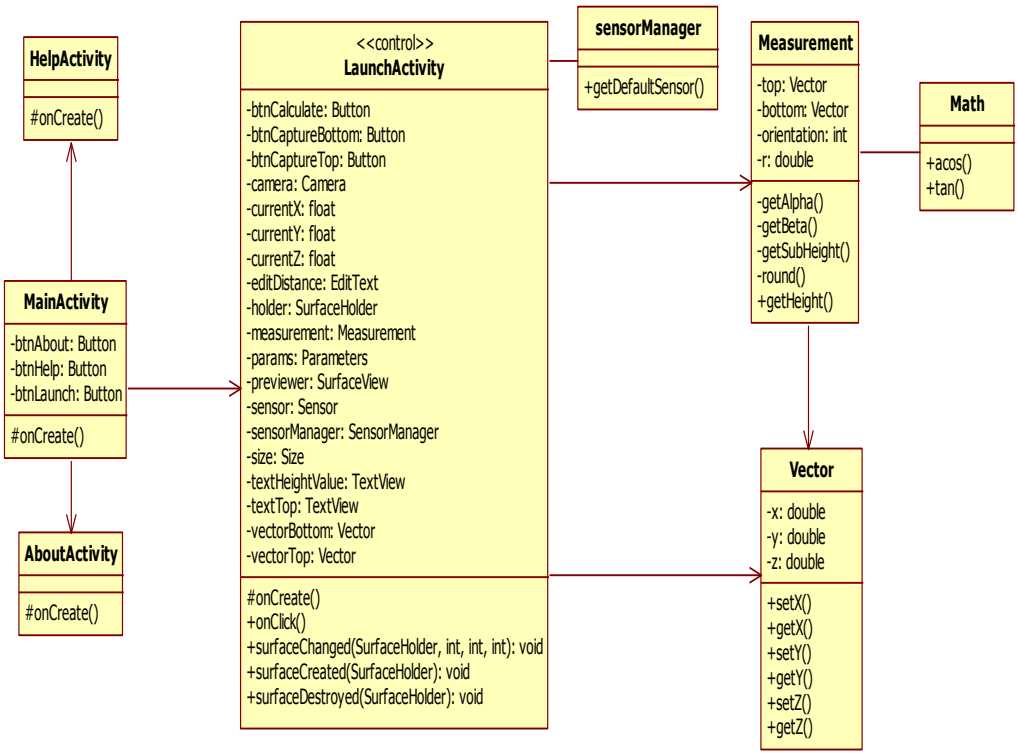


Gambar 3.8 *activity diagram* menghitung tinggi

Gambar diatas merupakan *activity diagram* yang memperlihatkan aktivitas dan interaksi antara pengguna dan sistem untuk memproses perhitungan dalam mencari hasil tinggi objek

3.1.3.2.3 Class Diagram

Diagram kelas (*Class Diagram*) merupakan diagram structural yang memodelkan sekumpulan kelas, interface, kolaborasi dan relasinya. Diagram kelas digambarkan dengan kotak yang pada dasarnya terbagi atas tiga bagian yaitu: nama kelas, atribut dan operasi. Diagram kelas digunakan untuk menggambarkan proses static dari aplikasi pengukuran tinggi objek ini. berikut ini adalah class diagram dari aplikasi pengukuran tinggi objek.



Gambar 3.9 Class Diagram

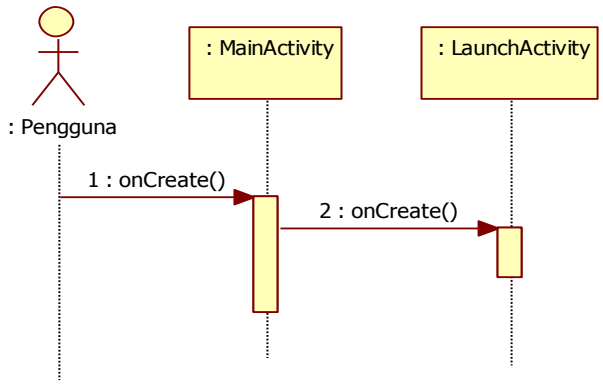
Tabel 3.15 Deskripsi Diagram Kelass (*Class Diagram*)

No	Kelas	Jenis Kelas	Deskripsi
1	LaunchActivity	*Control*	Kelas launchactivity merupakan kelas yang berfungsi untuk manjalankan aplikasi pengukuran tinggi objek dengan memanfaatkan *accelerometer*
2	Measurement	*Entity*	Kelas measurement merupakan kelas yang berfungsi untuk menghitung tinggi objek
3	Vector	*Entity*	Kelas vector merupakan kelas yang berfungsi untuk menyimpan nilai koordinat kemiringan *accelerometer* pada titik atas atau titik bawah objek
4	MainActivity	*Interface*	Kelas mainactivity merupakan kelas yang berfungsi untuk menjalankan aplikasi pada mobi *mobile*
5	HelpActivity	*Interface*	Kelas helpactivity merupakan kelas yang berfungsi untuk melihat petunjuk tentang cara menggunakan aplikasi pengukuran tinggi objek
6	AboutActivity	*Interface*	Kelas aboutactivity merupakan kelas yang berfungsi untuk melihat tentang aplikasi pengukuran tinggi objek

3.1.3.2.4 Sequence Diagram

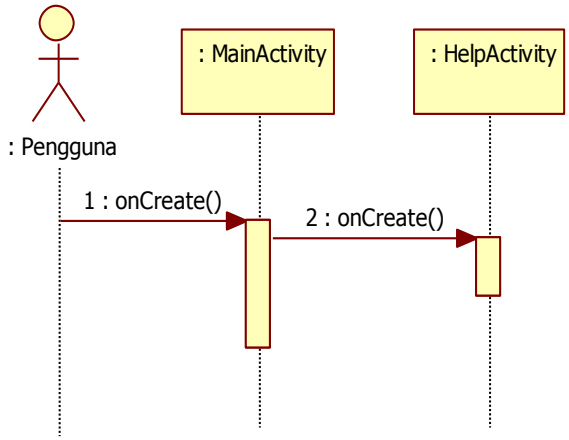
Sequence diagram digunakan untuk menggambarkan perilaku pada sebuahscenario. Diagram ini menunjukkan sejumlah contoh obyek dan message yangdiletakkan diantara obyek-obyek ini didalam use case. Biasa digunakan untukmenggambarkan scenario atau rangkaian langkah-langkah yang dilakukan sebagairespon dari sebuah event untuk menghasilkan output tertentu. Diawali dari sebuahaktivitas tertentu, kemudian berproses mengikuti urutan tertentu yang bisa terlihatmelalui *message* antar objeknya.

1. Sequence Diagram Menu Mulai



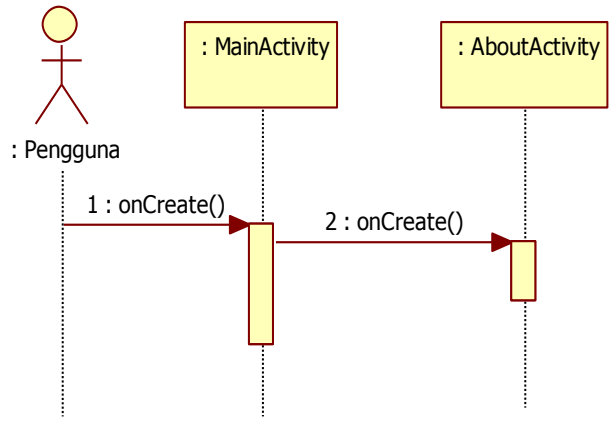
Gambar 3.10 Sequence Diagram Menu Mulai

2. Sequence Diagram Menu Petunjuk



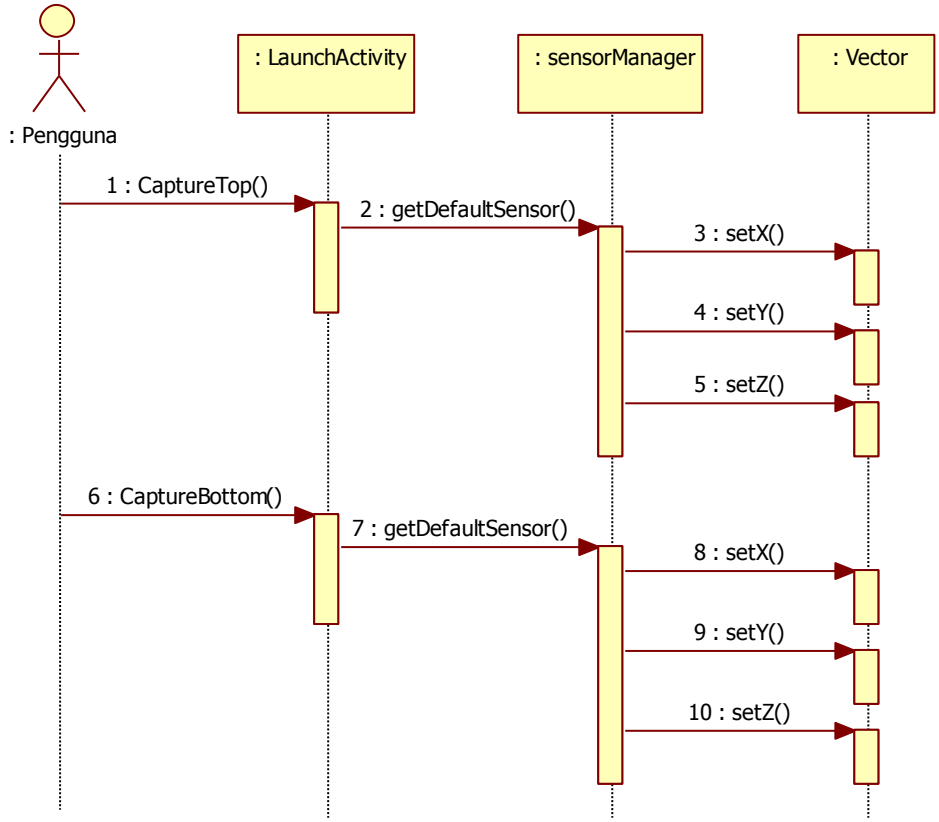
Gambar 3.11 Sequence Diagram Petunjuk

3. Sequence Diagram Menu Tentang



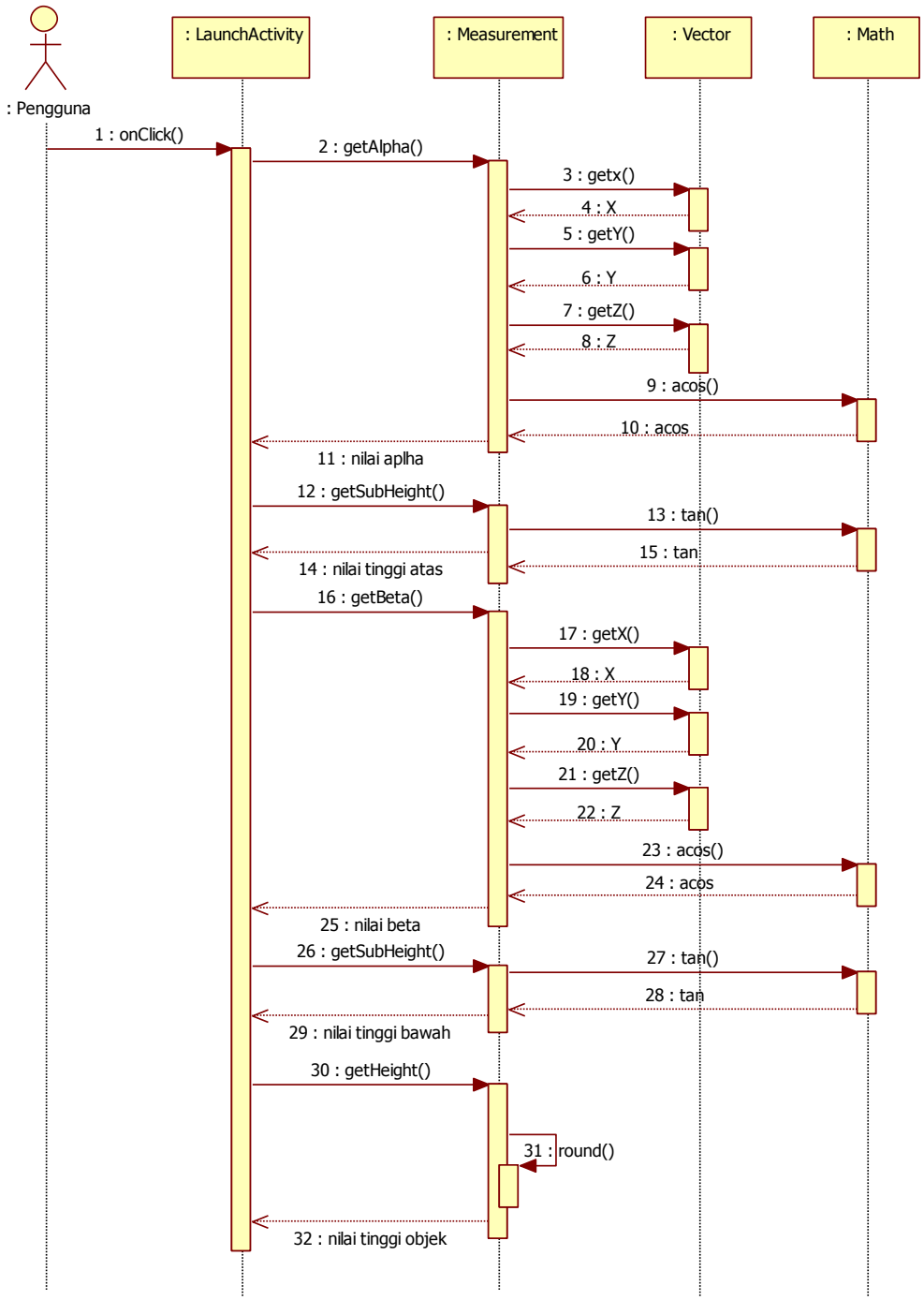
Gambar 3.12 Sequence Diagram Menu Tentang

4. Sequence Diagram Mencari Nilai Koordinat



Gambar 3.13 Sequence Diagram Mencari nilai koordinat

5. Sequence Diagram Menghitung Tinggi Objek



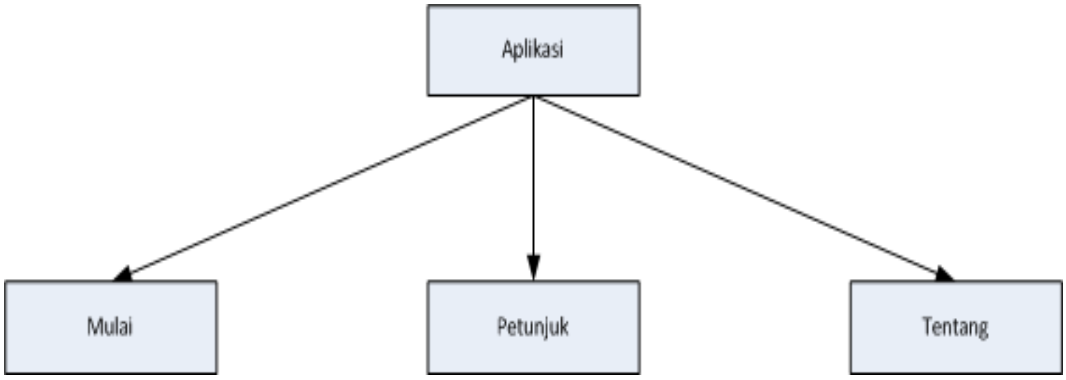
Gambar 3.14 Menghitung Tinggi Objek

3.2 Perancangan Sistem

Perancangan sistem bertujuan untuk menspesifikasikan aspek-aspek teknikyang menjadi solusi dalam perencanaan. Pada tahap ini perancanagn akandidefinisikan secara detail untuk mengatasi masalah-masalah yang lebih teknis,berkaitan dengan kegiatan implementasi seperti perancangan arsitektur sistem, dan perancangan antarmuka.

3.2.1 Perancangan Menu

Aplikasi pengukuran tinggi objek ini menggunakan struktur navigasi *Hierarchical Model.* Dimana menu utama adalah pusat menu atau navigasi yang merupakan penghubung kesemua fitur pada aplikasi.



Gambar 3.15 Menu Aplikasi Pengukuran Tinggi Objek

Tabel 3.16 Deskripsi Menu Aplikasi Pengukuran tinggi objek

No	Menu	Deskripsi
1	Mulai	Mulai merupakan menu untuk menjalan pengukuran tinggi objek yang akan dicari tingginya
2	Petunjuk	Petunjuk merupan bantuan cara menggunakan aplikasi pengukuran tinggi objek dengan memanfaatkan sensor *accelerometer* pada sistem operasi android
3	Tentang	Tentang merupakan apilkasi pengukuran tinggi objek

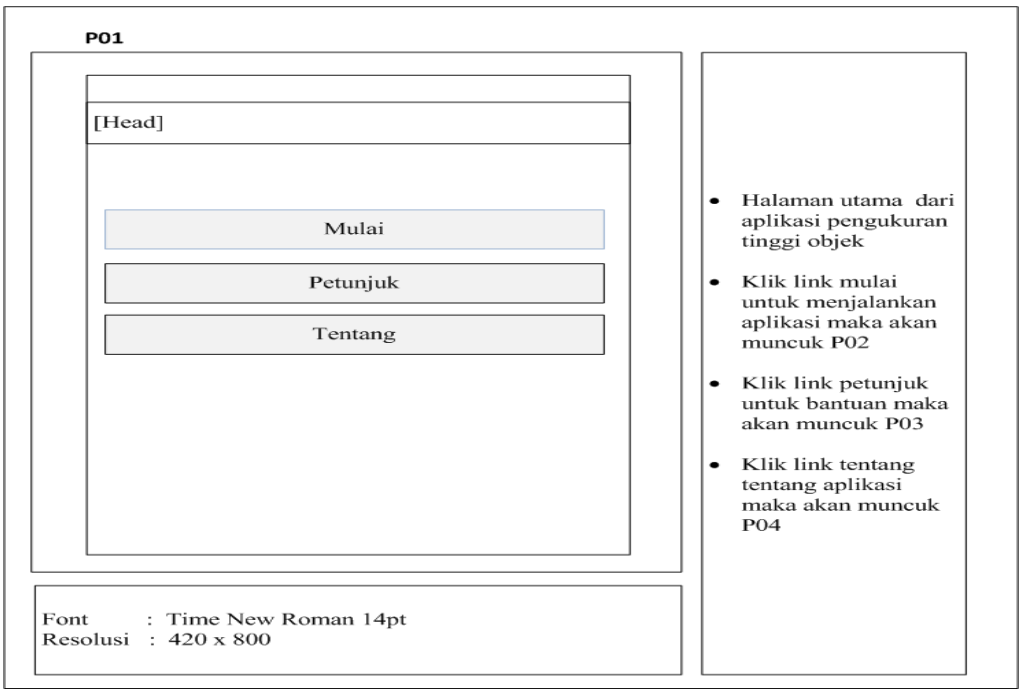
3.2.2 Peracangan Antarmuka

Perancangan antarmuka (*interface*) menggambarkan bagaimana perangkat lunak berkomunikasi dengan dirinya sendiri, dengan sistem yang berinteroperasi dengannya dan dengan manusia yang menggunkannya.

3.2.2.1 Perancangan Tampilan

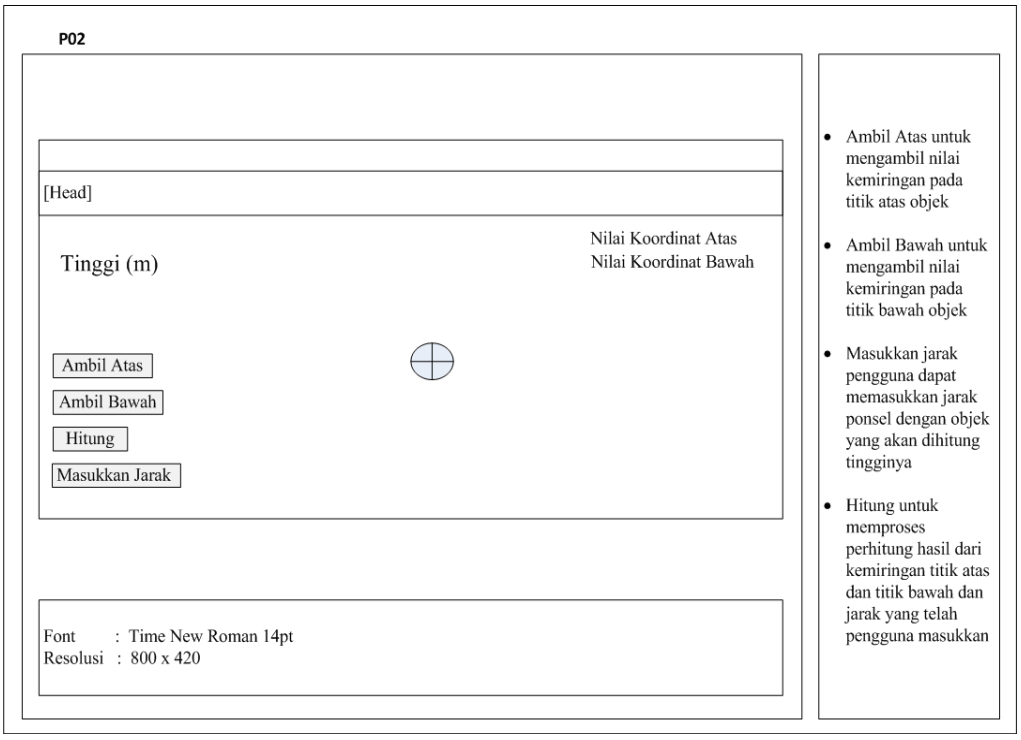
Perancangan *form* mendeskripsikan rencana tampilan dari setiap *form* yang akan digunakan pada tampilan aplikasi *mobile* pengukuran tinggi objek. Adapun gambaran dari perancangan antarmuka aplikasi *mobile* pengukuran tinggi objek adalah sebagai berikut:

1. **Perancangan menu utama**



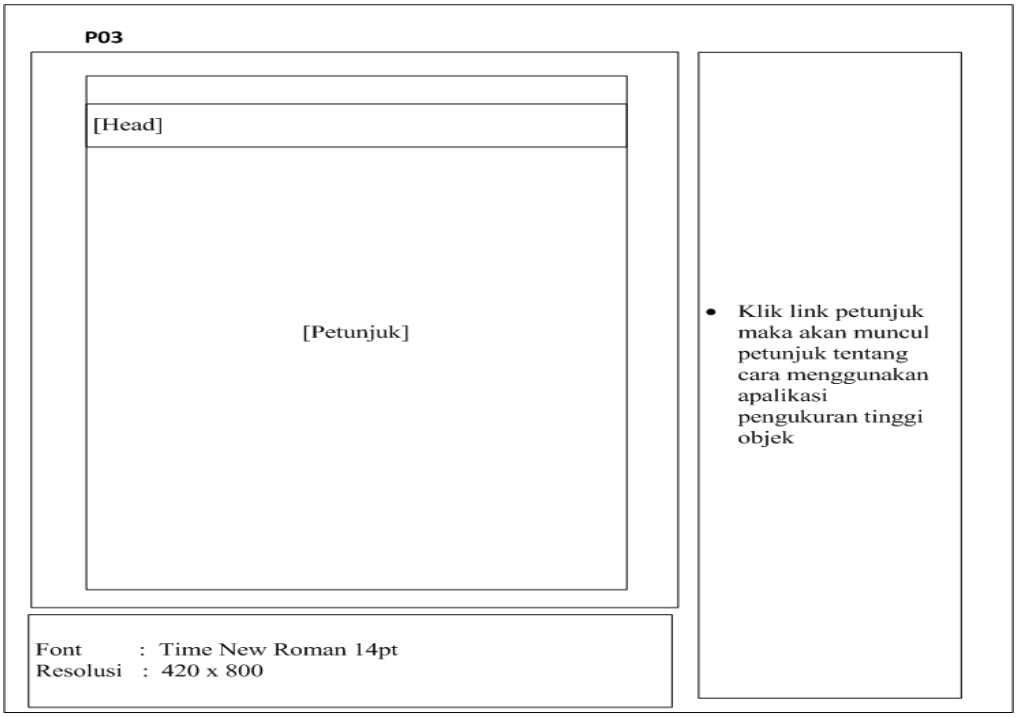
Gambar 3.16 Tampilan Menu Utama

2. Perancangan menu mulai



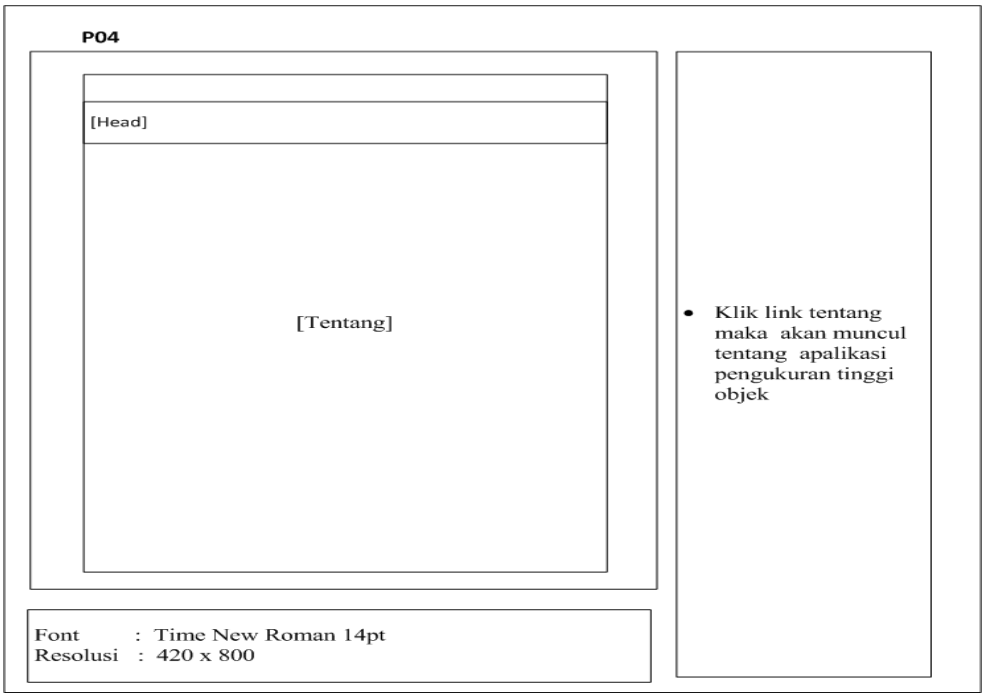
Gambar 3.17 Tampilan Mulai

3. Perancangan menu petunjuk



Gambar 3.18 Tampilan Petunjuk

4. Perancangan menu tentang

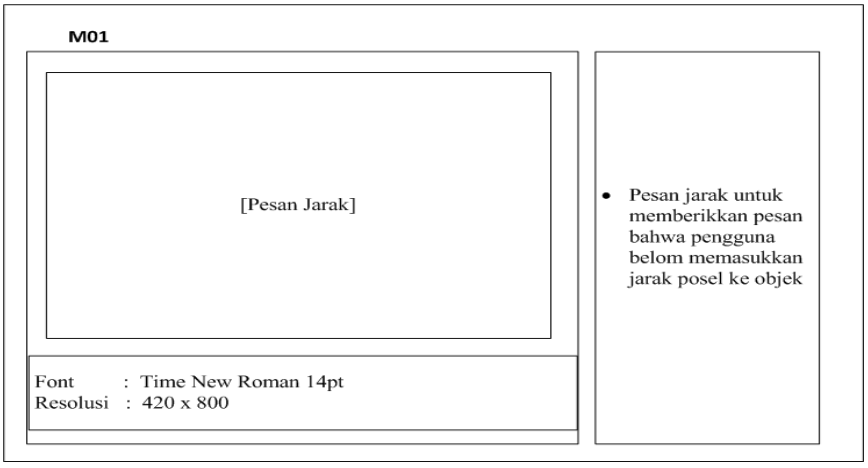


Gambar 3.19 Tampilan Tentang

3.2.2.2 Perancangan Pesan

Pesan yang dirancang dalam konteks aplikasi yang dikembangkan merupakan pesan peringatan yang keluar jika pengguna melakukan aktifitas yang memicu *event* tertentu.

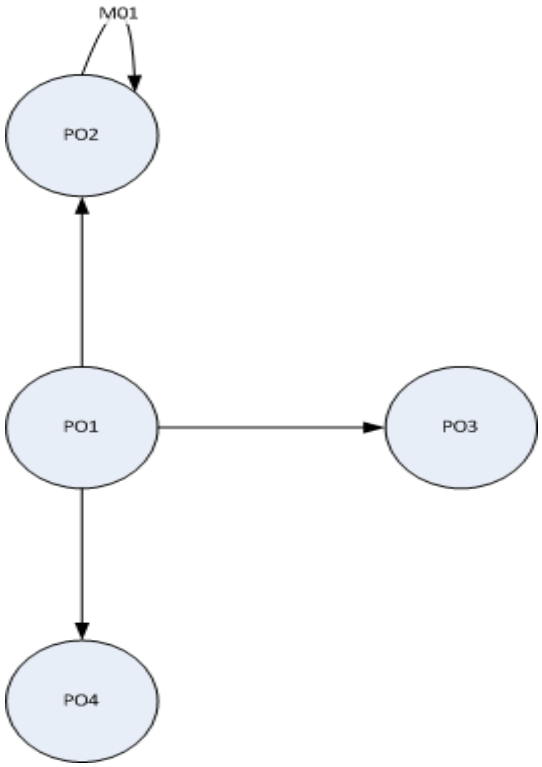
1. Pesan peringatan jika belum memasukkan jarak



Gambar 3.20 Pesan Jarak

3.2.2.3 Perancangan Jaringan Semantik

Jaringan semantik merupakan alur dari menu-menu yang terdapat didalam sebuah aplikasi yang dibangun. Jaringan semantik yang terdapat pada aplikasi pengukuran tinggi objek ini adalah.



Gambar 3.21 Jaringan Semantik