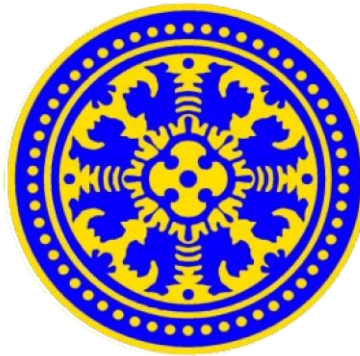


**Tugas Makalah**  
**Pemrograman Mobile**  
**TI024330**

**Desain dan Perancangan Aplikasi Menghitung Volume**  
**Bangun Ruang Berbasis Android**

Dosen :  
I Putu Agus Eka Pratama, ST MT



Oleh :

1. Tania Maria O.S	1404505098
2. Lalu Septian Adi Wijaya	1404505111

**Jurusan Teknologi Informasi**  
**Fakultas Teknik Universitas Udayana**

2015

## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang

Pada zaman era globalisasi sekarang ini kemajuan dibidang teknologi yang terus berkembang salah satunya adalah *operating system* atau sistem operasi pada *mobile device* seperti *handphone*. Diantara sistem operasi yang ada salah satu contohnya adalah android. Android merupakan sistem operasi untuk telepon seluler yang berbasis linux dan bersifat *open source*, sehingga menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi sendiri.

Matematika sebagai ilmu terapan yang juga merupakan salah satu ilmu dasar dari pengembangan sistem komputer. Bangun ruang dan bangun ruang merupakan salah satu anak pelajaran dari ilmu matematika. Bangun ruang merupakan suatu bidang 3 dimensi yang berbentuk satuan dasar luasnya sering ditulis dengan pangkat 3 ( $m^3$ ).

Bangun ruang atau biasa disebut juga sebagai bangunan tiga dimensi merupakan jenis bangun yang memiliki ruang serta sisi-sisi yang membatasinya. Jumlah serta bentuk dari setiap sisi yang ada menjadi cirikhas tersendiri dari sebuah bangun ruang. Karena bentuk dan jumlah rusuknya berbeda, maka setiap bangun ruang memiliki karakteristik sendiri yang menjadi ciri-ciri ataupun sifat dari bangun ruang tersebut.

Selama ini, untuk menghitung volume pada bangun ruang, masih dilakukan secara manual. Hal tersebut tentu saja membutuhkan waktu yang tidak sedikit dan membutuhkan ketelitian. Bahkan kemungkinan terjadi *human error* saat melakukan penghitungan cukup besar. Selain itu, ada pula aplikasi konverter yang telah tersedia untuk komputer. Aplikasi tersebut tentu saja tidak efektif untuk di bawa kemana saja. Maka dibutuhkan suatu media yang memungkinkan aplikasi tersebut dapat digunakan kapan saja dan dimana saja.

## **1.2 Rumusan Masalah**

Berdasarkan latar belakang masalah diatas, rumusan masalahnya adalah sebagai berikut:

1. Bagaimana merancang suatu perangkat lunak yang dapat membantu untuk mempelajari dan mengenal berbagai bentuk-bentuk bangun ruang.
2. Bagaimana merancang suatu perangkat lunak untuk menyelesaikan permasalahan dalam perhitungan bangun ruang.

## **1.3 Solusi**

Penulis tertarik untuk membuat dan mengembangkan sebuah aplikasi sederhana namun bermanfaat yang berbasis android berupa aplikasi perhitungan rumus dasar mencari volume bangun ruang. Pada aplikasi ini terdapat beberapa opsi pilihan bangun ruang seperti kubus, balok dan bola. Selain dapat melakukan proses penghitungan, pada aplikasi ini juga terdapat penjelasan singkat dari bangun ruang yang di pilih oleh *user* nantinya. Dibandingkan dengan membaca rumus dari buku, kelebihan dari aplikasi ini adalah dapat digunakan dimana saja dan kapan saja sehingga memudahkan *user* khususnya pelajar dalam proses pembelajaran yang disertai penjelasan singkat dari bangun ruang dan perhitungan yang akurat.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1    *Mobile Programming***

*Mobile device* adalah perangkat bergerak seperti *handphone*, laptop yang digunakan untuk mengakses jasa jaringan. <sup>[5]</sup>

*Mobile programming* atau pemrograman mobile adalah pemrograman yang ditujukan untuk pembuatan aplikasi diperangkat mobile yang dapat kita buat dengan menggunakan Java. Keterbatasan yang sangat umum dalam bahasa pemrograman untuk peralatan mobile adalah kendala dalam segala dalam hal sumber daya, seperti ukuran layar, memori, CPU, penyimpanan dan cara menginput data. Perbedaan tampilan juga disebabkan adanya perbedaan hardware dan API yang di gunakan.

- Bagian pendukung *Mobile Programming*

1.        Emulator

Aplikasi yang memungkinkan sebuah software dari sebuah platform berjalan di atas platform lain. Emulator menjalankan kode-kode software pada virtual machine. Sehingga software tersebut mengira bahwa Ia sedang berjalan di atas platform aslinya.

2.        WAP

Teknologi WAP (*Wireless Application Protocol*) merupakan sebuah metode untuk menghadirkan halaman web di dalam layar Ponsel (Telepon Seluler). Dengan begitu, maka bentuk informasi dari sebuah instansi tidak hanya ditampilkan dengan metode Web yang hanya dapat di akses melalui Browser, akan tetapi dapat ditampilkan dalam bentuk WAP yang dapat dilihat dengan ponsel.

#### **2.2    *Android***

Android (sistem operasi) merupakan sebuah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android

menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. <sup>[2]</sup>

Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google *Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

### **2.3    *Object Oriented***

*Object Oriented Programming* atau disingkat OOP merupakan paradigma pemrograman yang populer saat ini yang telah menggantikan teknik pemrograman berbasis prosedur. *Object Oriented Programing* yang berarti pula Pemrograman Berorientasi Objek sudah ditemukan sekitar tahun 1960 dan dikembangkan pada permulaan tahun 1970. <sup>[7]</sup>

Pemrograman Berorientasi Objek (*Object Oriented Programming*/OOP) merupakan pemrograman yang berorientasikan kepada objek, dimana semua data dan fungsi dibungkus dalam *class-class* atau *object-object*. Setiap *object* dapat menerima pesan, memproses data, mengirim, menyimpan dan memanipulasi data. Beberapa *object* berinteraksi dengan saling memberikan informasi satu terhadap yang lainnya.

Masing-masing *object* harus berisikan informasi mengenai dirinya sendiri dan dapat dihubungkan dengan *Object* yang lain. Pemrograman berorientasi objek berbeda dengan pemrograman prosedural yang hanya menggunakan satu halaman

kebawah untuk mengerjakan banyak perintah atau statement. Penggunaan pemrograman berorientasi objek sangat banyak sekali, contoh : java, php, perl, c#, cobol, dan lainnya.

## 2.4 UML

UML (*Unified Modeling Language*) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*).<sup>[3]</sup>

UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen- komponen yang diperlukan dalam sistem *software*. UML adalah salah satu *tool* / model untuk merancang pengembangan *software* yang berbasis *object oriented*. Beberapa *modeling tool* yang dipakai adalah bagian dari dasar UML, kependekan dari United Modeling Language. UML terdiri atas beberapa diagram, yaitu:<sup>[5]</sup>

### 1. *Use Case Diagram*

*Use Case* menggambarkan apa saja aktifitas yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. yang menjadi persoalan itu apa yang dilakukan bukan bagaimana melakukannya.

Diagram *Use Case* dekat kaitannya dengan kejadian-kejadian. Kejadian (skenario) merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan sistem.

### 2. *Sequence Diagram*

Diagram *Class* dan diagram *Object* merupakan suatu gambaran model statis. Namun ada juga yang bersifat dinamis, seperti Diagram *Interaction*. Diagram *sequence* merupakan salah satu diagram *Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan; message (pesan) apa yang dikirim dan kapan pelaksanaannya.

Diagram ini diatur berdasarkan waktu. Obyek-obyek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.

### 3. *Class Diagram*

*Diagram Class* memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan mereka. *Diagram Class* bersifat statis; menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan.

*Diagram Class* mempunyai 3 macam relationships (hubungan), sebagai berikut :

- *Association* Suatu hubungan antara bagian dari dua kelas. Terjadi *association* antara dua kelas jika salah satu bagian dari kelas mengetahui yang lainnya dalam melakukan suatu kegiatan. Di dalam diagram, sebuah *association* adalah penghubung yang menghubungkan dua kelas.
- *Aggregation* Suatu *association* dimana salah satu kelasnya merupakan bagian dari suatu kumpulan. *Aggregation* memiliki titik pusat yang mencakup keseluruhan bagian. Sebagai contoh : OrderDetail merupakan kumpulan dari Order.
- *Generalization* Suatu hubungan turunan dengan mengasumsikan satu kelas merupakan suatu *superclass* (kelas super) dari kelas yang lain. *Generalization* memiliki tingkatan yang berpusat pada *superclass*. Contoh : *Payment* adalah *superclass* dari *Cash*, *Check*, dan *Credit*.

## 2.5 **Java**

Java menurut definisi dari Sun adalah sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada computer stand alone ataupun pada lingkungan jaringan. Java 2 adalah generasi kedua dari java platform. Kata berdiri di atas sebuah mesin interpreter yang diberi nama Java Virtual Machine (JVM). JVM inilah yang akan membaca bytecode dalam file.class dari suatu program sebagai representasi langsung dari program yang berisi bahasa mesin. Oleh karena itu, bahasa

java disebut sebagai bahasa pemrograman yang portable karena dapat dijalankan pada berbagai sistem operasi, asalkan pada sistem operasi tersebut terdapat JVM. Agar sebuah program Java dapat dijalankan, maka file dengan ekstensi, java harus dikompilasi menjadi file bytecode. Untuk menjalankan bytecode tersebut dibutuhkan JRE (Java Runtime Environment) yang memungkinkan pemakai untuk menjalankan program Java, hanya menjalankan, tidak untuk membuat kode baru lagi. JRE berisi JVM dan library Java yang digunakan. <sup>[1][6]</sup>

Java memiliki beberapa versi library atau teknologi yang disebut juga sebagai edisi dari bahasa pemrograman Java. Tiga edisi utama dari library tersebut adalah Micro, Standard, dan Enterprise. J2ME (Java2 Micro Edition.) Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program Java Oak pertama, yang ditujukan sebagai pengendali sebuah peralatan dengan teknologi layar sentuh (touch screen), seperti pada PDA sekarang ini. Teknologi baru ini dinamai "7" (Star Seven). Setelah era Star Seven selesai, sebuah anak perusahaan TV kabel tertarik ditambah beberapa orang dari proyek The Green Project. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, Palo Alto. Perusahaan baru ini bertambah maju: jumlah karyawan meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada 7 rentang waktu ini juga ditetapkan pemakaian Internet sebagai medium yang menjembatani kerja dan ide di antara mereka. Pada awal tahun 1990-an, Internet masih merupakan rintisan, yang dipakai hanya di kalangan akademisi dan militer. Mereka menjadikan perambah (browser) Mosaic sebagai landasan awal untuk membuat perambah Java pertama yang dinamai Web Runner, terinspirasi dari film 1980-an, Blade Runner.

Pada perkembangan rilis pertama, Web Runner berganti nama menjadi Hot Java. Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar San Jose Mercury News pada tanggal 23 Mei 1995. Sayang terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah ruangan hotel Sheraton Palace. Tiga dari pimpinan utama proyek, Eric Schmidt dan George Paolini dari Sun Microsystems bersama Marc Andreessen, membentuk Netscape. Nama Oak,



diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "bapak java", James Gosling. Nama Oak ini tidak dipakai untuk versi release Java karena sebuah perangkat lunak sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling.

## 2.6 *Mobile Computing*

*Mobile Computing* muncul sebagai akibat dari perkembangan teknologi komputer, perangkat keras, dan perangkat lunak di dalamnya, termasuk juga jaringan komputer. <sup>[4]</sup>

- Definisi

Menurut salah satu ahli Uwe Velle dari *Association for Computing Machinery* (ACM) disebutkan bahwa *Mobile Computing* merupakan sebuah salah satu bentuk dari proses komputasi dan komputerisasi tanpa kabel (*wireless* / tanpa menggunakan kabel atau gelombang), sehingga pengguna dapat melakukannya di manapun dan kapanpun untuk bertukar data maupun berbagi informasi melalui komputer yang berukuran relatif kecil.

- Unsur – unsur *Mobile Computing*

Dari definisi diatas, terlihat bahwa *Mobile Computing* setidaknya memiliki tiga unsur sebagai berikut.

1. *Mobile Software*, yaitu sejumlah aplikasi dan system operasi khusus untuk *mobile computing*.
2. *Mobile Hardware*, yaitu sejumlah perangkat keras yang mendukung adanya *mobile computing*. Perangkat keras ini dapat berupa perangkat keras penghubung maupun komponen – komponen komputer yang mendukung adanya *mobile computing*.
3. *Mobile Communication*, yaitu sejumlah protocol jaringan, format data, dan teknologi terkait dengan pengoperasian *mobile computing*.

- Faktor yang mempengaruhi jalannya *Mobile Computing*

*Mobile computing* sangat membantu jalannya kehidupan manusia di abad ini. Meski demikian perlu diketahui sejumlah faktor yang dapat membatasi penggunaan *mobile computing* serta mempengaruhi jalannya *mobile computing*. Faktor – factor tersebut antara lain:

1. *Bandwith* jaringan

*Mobile computing* menggunakan teknologi jaringan *wireless* dengan beragam lebar pita jaringan (*bandwith*), dimulai dari GPRS, EDGE, hingga 4G.

2. Standardisasi keamanan

Salah satu teknologi keamanan yang umum digunakan di dalam *mobile computing* antara lain VPN (*Virtual Private Network*) yang salah satunya dimanfaatkan untuk memudahkan koneksi ke komputer di jaringan internet melalui internet dan SSL (*Secure Socket Layer*) yang dimanfaatkan pada saat transaksi perbankan secara online (*online banking*, *sms banking*, *e-banking*).

3. Konsumsi sumber daya energi

Sumber daya energi yang dimaksud disini adalah energi listrik yang umumnya tersimpan di dalam baterai. Perlu adanya penerapan *green computing*, dimana dengan konsumsi daya yang relatif rendah, performansi dan produktifitas tetap ataupun bertambah.

4. Interferensi dengan sinyal / gelombang lainnya

Memanfaatkan sinyal / gelombang untuk memperoleh koneksi merupakan ciri khas dari *wireless network* (nirkabel). Salah satunya yaitu menghindari adanya interferensi (tabrakan) dengan gelombang lainnya maupun benda / hal penghalang lainnya.

5. Interaksi dengan pengguna melalui tatap muka (*interface*) aplikasi

*Mobile computing* terdiri dari *mobile application*. *Mobile application* terdiri atas sejumlah pilihan aplikasi dan sistem operasi yang menunjang kegiatan *user* dalam melakukan *mobile computing*. Kemudahan ini dapat ditunjang dari bagaimana *user interface* (tatap muka aplikasi untuk pengguna) dirancang, disusun, diimplementasikan dengan tepat.

## 2.7 Android Studio

Android Studio adalah sebuah IDE untuk Android Development yang diperkenalkan google pada acara Google I/O 2013. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android Studio merupakan IDE resmi untuk pengembangan aplikasi Android.

Sebagai pengembangan dari Eclipse, Android Studio mempunyai banyak fitur-fitur baru dibandingkan dengan Eclipse IDE. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai build environment. Fitur-fitur lainnya adalah sebagai berikut :

- Menggunakan *Gradle-based build system* yang fleksibel.
- Bisa *mem-build multiple APK* .
- *Template support* untuk *Google Services* dan berbagai macam tipe perangkat.
- Layout editor yang lebih bagus.
- Built-in support untuk Google Cloud Platform, sehingga mudah untuk integrasi dengan Google Cloud Messaging dan App Engine.
- *Import library* langsung dari Maven repository
- dan masih banyak lagi lainnya

## 2.8 Matematika

Matematika (dari bahasa Yunani: - mathmatik) adalah studi besaran, struktur, ruang, dan perubahan. Para matematikawan mencari berbagai pola, merumuskan konjektur baru, dan membangun kebenaran melalui metode deduksi yang kaku dari aksioma-aksioma dan definisi-definisi yang bersesuaian. Melalui penggunaan penalaran logika dan abstraksi, matematika berkembang dari pencacahan, perhitungan, pengukuran, dan pengkajian sistematis terhadap bangun dan pergerakan benda-benda fisika.

Matematika praktis telah menjadi kegiatan manusia sejak adanya reka- man tertulis. Argumentasi kaku pertama muncul di dalam Matematika Yunani, terutama di dalam karya Euklides. Luas, luasan, atau area adalah besaran yang menyatakan

ukuran dua dimensi (dwigatra) suatu bagian permukaan yang dibatasi dengan jelas, biasanya suatu daerah yang dibatasi oleh kurva tertutup. Luas permukaan menyatakan luasan permukaan suatu benda padat tiga dimensi. Dalam aplikasi, luas permukaan bumi, yang dipakai dalam pengukuran lahan dan merupakan suatu luasan permukaan, kerap dianggap sebagai luas dua dimensi bidang ruang apabila luasan itu tidak terlalu besar relatif terhadap luas permukaan total bumi.

Satuan luas pokok menurut Sistem Internasional adalah meter persegi sedangkan menurut sistem Imperial adalah kaki persegi. Luas suatu bangun dua dimensi dapat dihitung dengan menggunakan elemen satuan luas berupa persegi (atau bentuk lain) yang diketahui ukurannya. Luas bangun yang akan diukur merupakan jumlah elemen satuan luas yang menutupinya. Untuk bangun-bangun yang memiliki keteraturan terdapat rumus-rumus yang dapat digunakan bergantung pada karakteristik bangun dua dimensi yang dimaksud.

### **2.8.1 Bangun Ruang**

Seperti yang kita ketahui bahwa bangun ruang merupakan sebutan untuk beragam jenis bangun tiga dimensi. Setiap jenis bangun ruang tersebut tentu memiliki ciri-ciri, sifat dan rumus - rumus tertentu.

Bangun ruang memiliki beberapa rumus dalam pembahasan makalah ini kami mengambil sebagian dari semua jenis bangun ruang diantaranya adalah:

#### **1) Kubus**

Kubus adalah bangun ruang tiga dimensi yang dibatasi oleh enam bidang sisi yang kongruen berbentuk bujur sangkar. Kubus memiliki 6 sisi, 12 rusuk dan 8 titik sudut. Kubus juga disebut bidang enam beraturan, selain itu juga merupakan bentuk khusus dalam prisma segiempat.

#### **2) Balok**

Balok adalah bangun ruang tiga dimensi yang dibentuk oleh tiga pasang persegi atau persegi panjang, dengan paling tidak satu pasang di antaranya berukuran berbeda. Balok memiliki 6 sisi, 12 rusuk dan 8 titik sudut. Balok yang dibentuk oleh enam persegi sama dan sebangun disebut sebagai kubus.

3) Bola

Sifat-sifat yang menjadi cirikhas dari bola adalah hanya memiliki satu buah sisi, tidak mempunyai titik sudut dan hanya mempunyai sebuah sisi lengkung yang tertutup

## BAB III

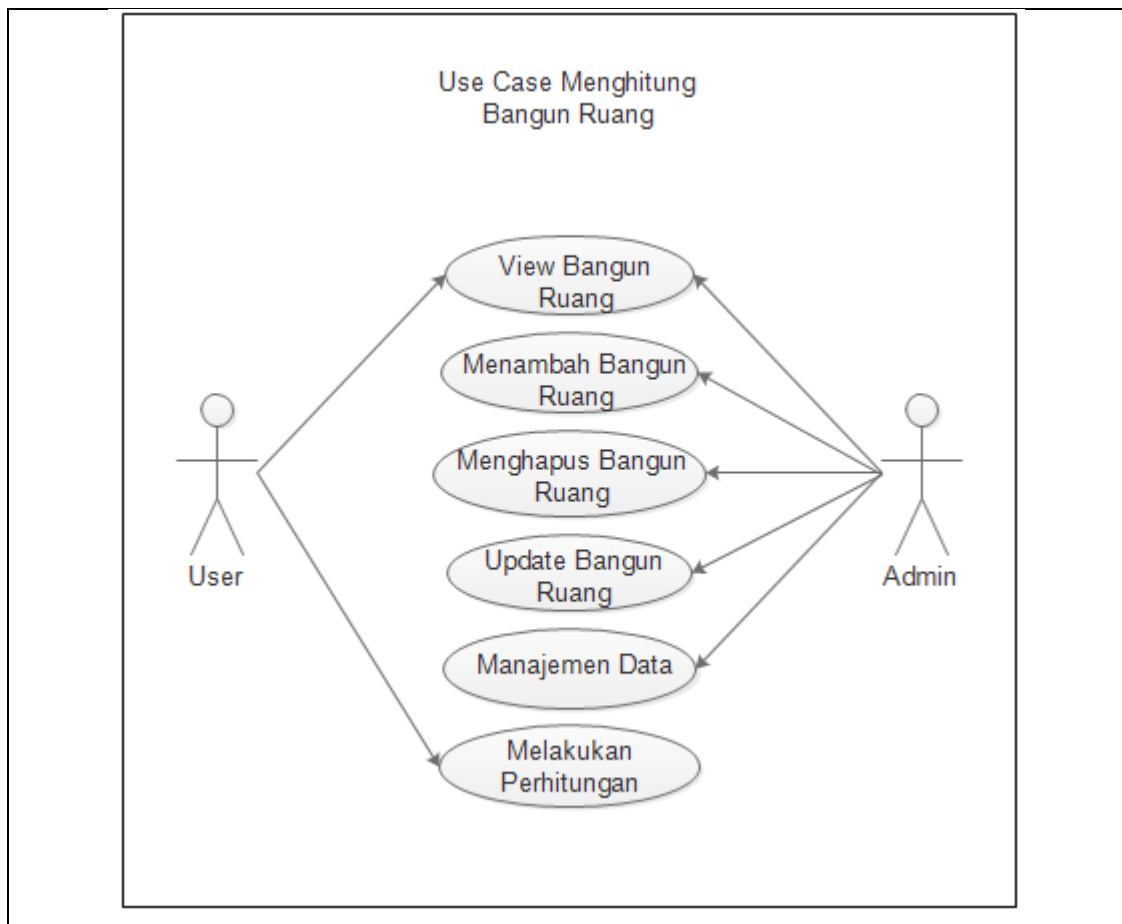
### PEMBAHASAN

#### 3.1 *Design Solusi*

Desain dan perancangan aplikasi menghitung bangun ruang berbasis android akan dijelaskan melalui beberapa diagram, yaitu *diagram use case*, *diagram sequence* dan *diagram class* serta desain *interface* juga akan dipaparkan.

##### 3.1.1 *Use Case Diagram*

Perancangan aplikasi menghitung bangun ruang dalam bentuk *use case diagram* akan dijelaskan sebagai berikut.



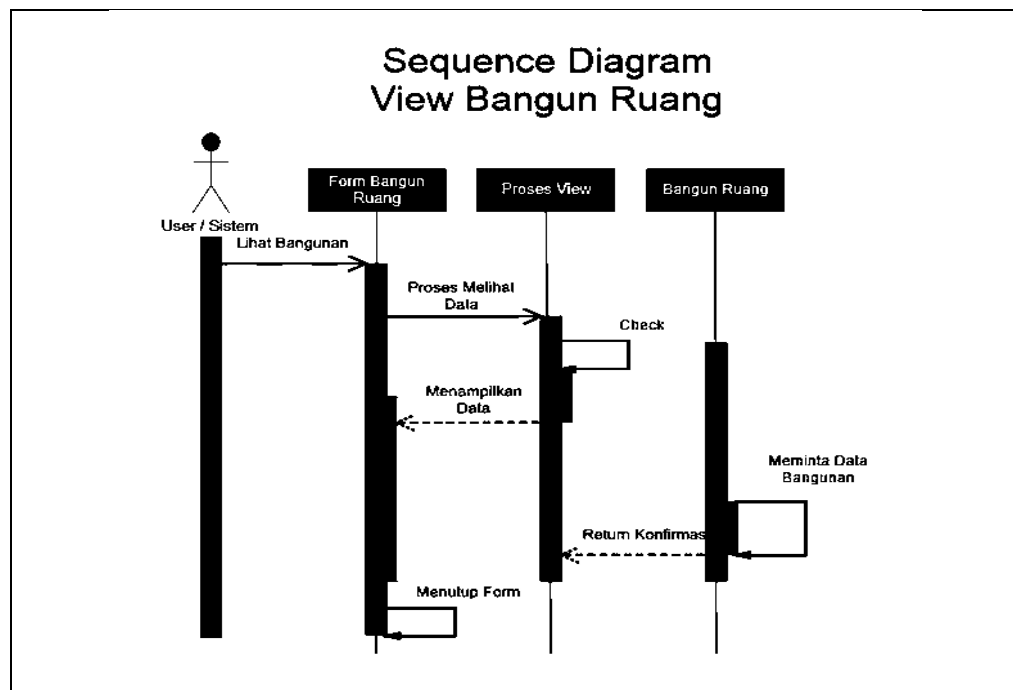
Pada gambar di atas telah di paparkan beberapa bagian pada *use case diagram* aplikasi menghitung bangun ruang. Bagian-bagian akses yang diberikan yaitu view bangun ruang, menambah bangun ruang, menghapus bangun ruang, update bangun

ruang, manajemen data dan melakukan perhitungan. *User* memiliki 2 hak akses yaitu untuk view bangun ruang dan melakukan perhitungan. Admin memiliki hak ases yaitu untuk view bangun ruang, menambah bangun ruang, menghapus bangun ruang, update bangun ruang, dan melakukan manajemen data.

### 3.1.2 Sequence Diagram

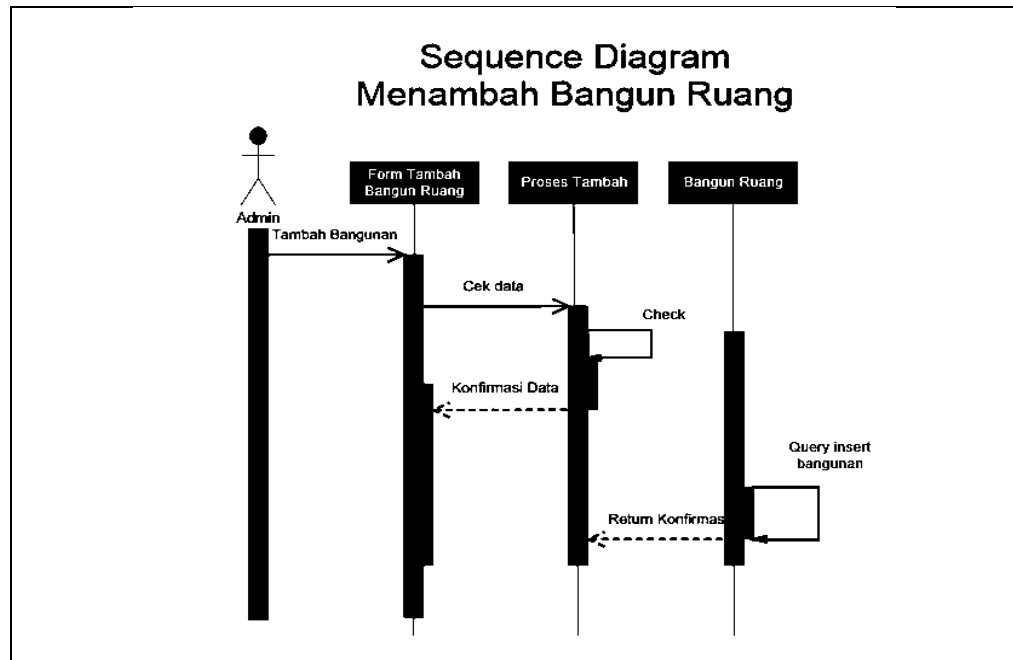
*Sequence diagram* adalah suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi diantara obyek-obyek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh obyek- obyek yang melakukan suatu tugas atau aksi tertentu.

#### 1. View



Gambar diatas menjelaskan bagaimana proses yang terjadi ketika *user* dan admin melihat informasi pada aplikasi menghitung bangun ruang. Kemudian akan di proses untuk melihat data informasi yang ada pada aplikasi tersebut untuk menghubungkan ke *database*. Setelah proses menghubungkan ke *database*, maka aplikasi akan memberikan konfirmasi berupa tampilan data – data bangun ruang.

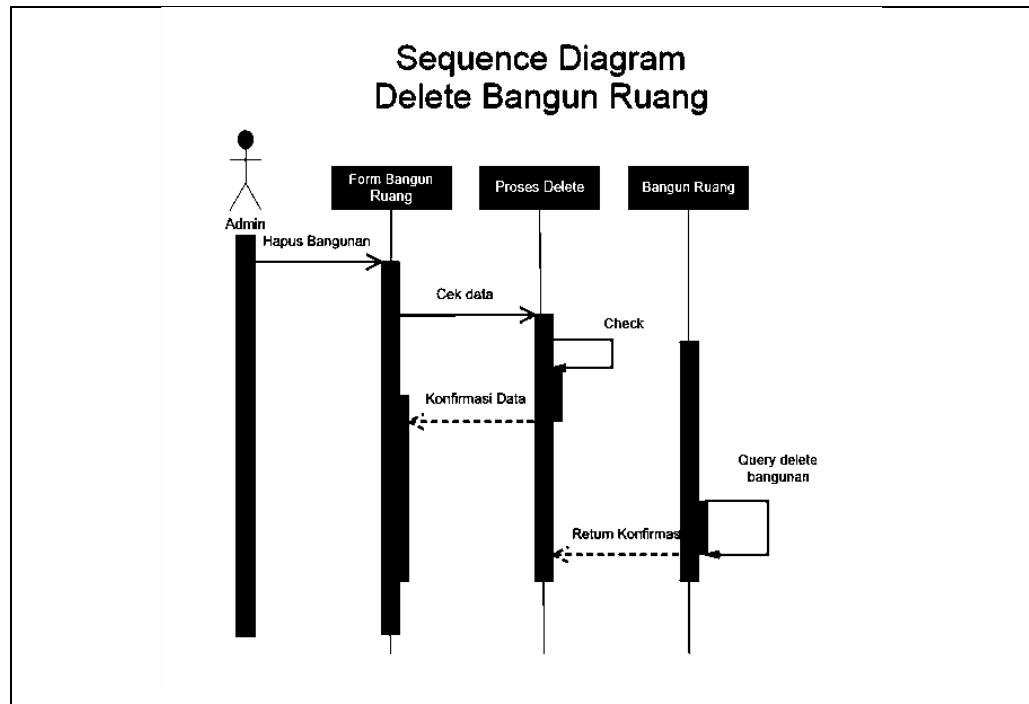
## 2. Menambah Bangun Ruang



*Sequence* ini diawali dari *admin* menginputkan detail-detail bangun ruang yang akan ditambahkan, dimulai dari menginput data bangun ruang kemudian data bangunan tersebut dikirim ke *control*, *control* akan menghubungkan ke *database* dan memanggil *query insert* data, *database* akan memberikan timbal balik yang akan diteruskan ke *interface (boundary)*.

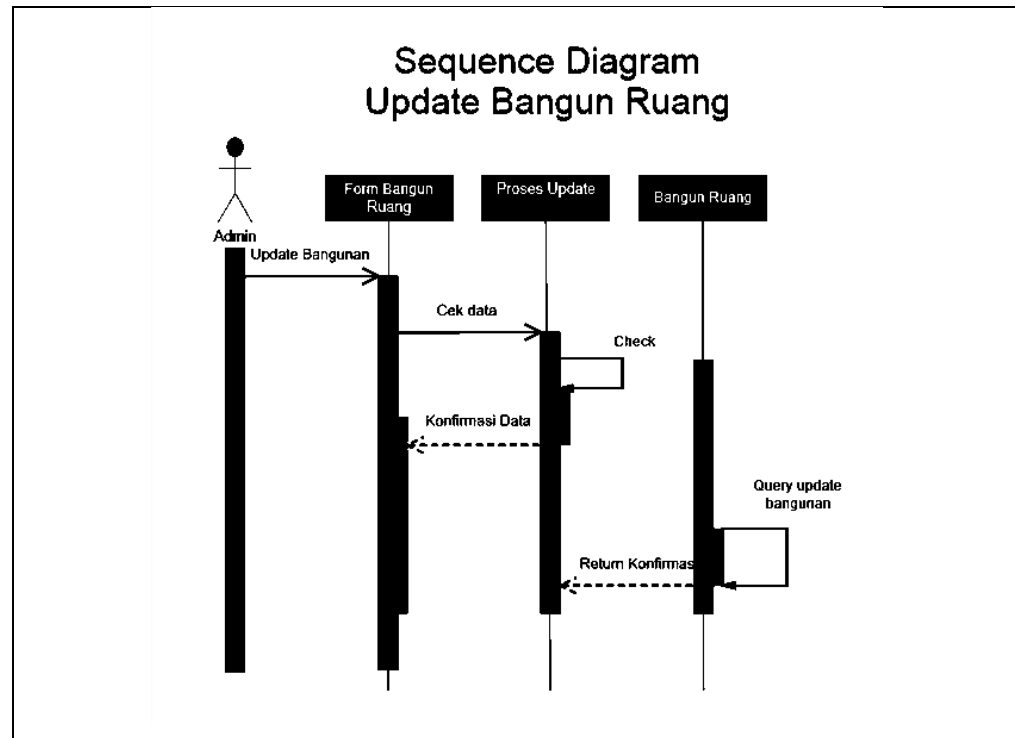


### 3. Menghapus Bangun Ruang



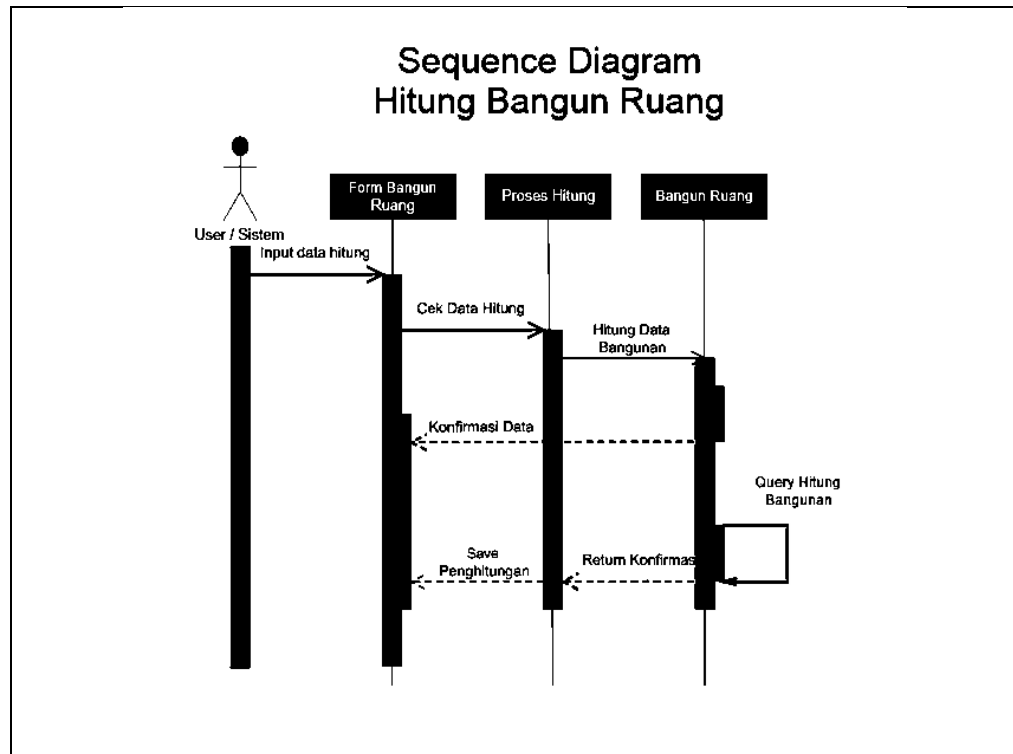
Gambar diatas menjelaskan bagaimana proses yang terjadi ketika *admin* melakukan penghapusan informasi bangun ruang pada aplikasi menghitung bangun ruang. Kemudian akan di proses untuk melihat ke halaman data yang ada pada tempat tersebut untuk menghubungkan ke *database*. Setelah proses menghubungkan ke *database*, maka aplikasi akan memberikan konfirmasi berupa tampilan kata “sukses menghapus”.

#### 4. *Update Bangun Ruang*



Untuk meng-*update* data bangun ruang, misalnya bangun ruang yang tersedia, akan berhubungan dengan *sequence* ini. Sama halnya dengan *insert* dan *update* sebelumnya, hanya saja berbeda fungsi *query* yang dijalankan.

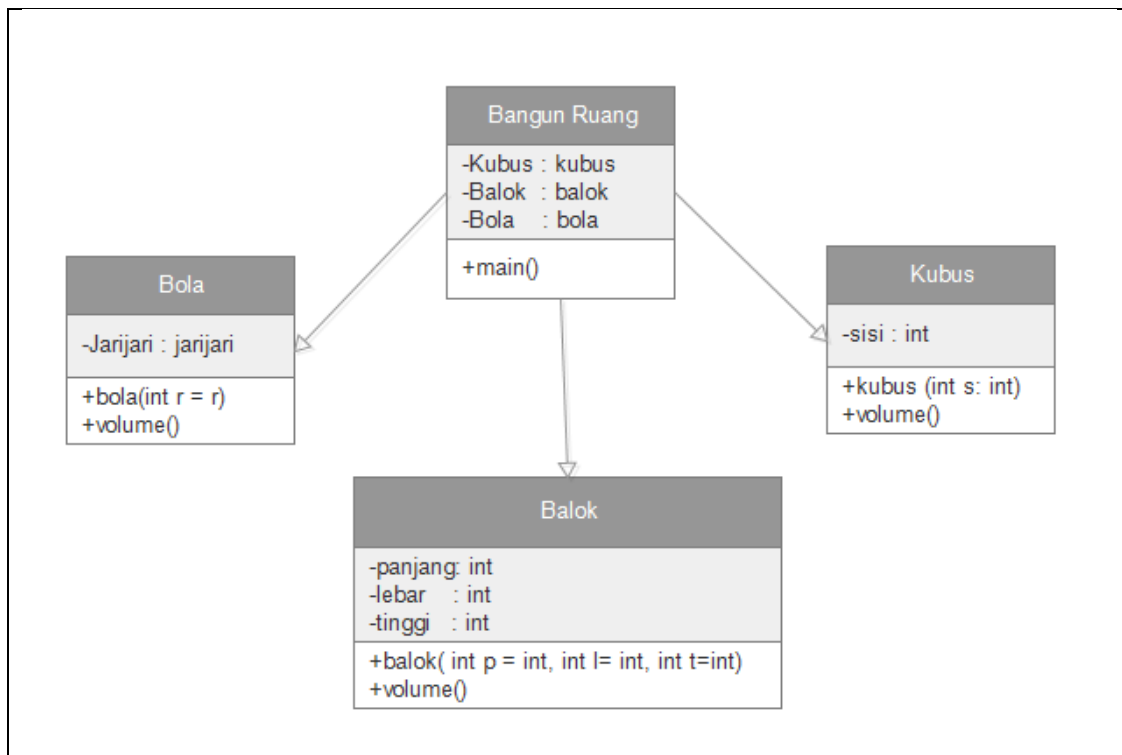
## 5. Melakukan Perhitungan



Gambar diatas menggambarkan bagaimana sistem melakukan perhitungan terhadap data volume yang diinputkan oleh *user*. *User* akan menginputkan bilangan yang akan dihitung sebagai volume, kemuadia sistem akan memproses data tersebut sehingga menampilkan hasil dari data yang diinputkan *user*.

### 3.1.3 Class Diagram

*Class* Diagram adalah diagram UML yang menggambarkan kelas-kelas dalam sebuah sistem dan hubungannya antara satu dengan yang lain, serta dimasukkan pula atribut dan operasi. Berikut gambar untuk *Class* Diagram.



Class diagram diatas merupakan class diagram yang dimiliki oleh aplikasi menghitung bangun ruang. Pada class diagram ini terdapat empat kelas yaitu bangunruang, bola, kubus dan balok. Class bangun ruang terdiri dari kubus, balok dan bola. Pada class bola terdapat jari –jari yang nantinya akan diinputkan untuk menghitung volume bola. Pada class kubus terdapat sisi terdapat sisi untuk diinputkan. Pada class balok terdapat panjang, lebar dan tinggi.

### 3.2 Implementasi

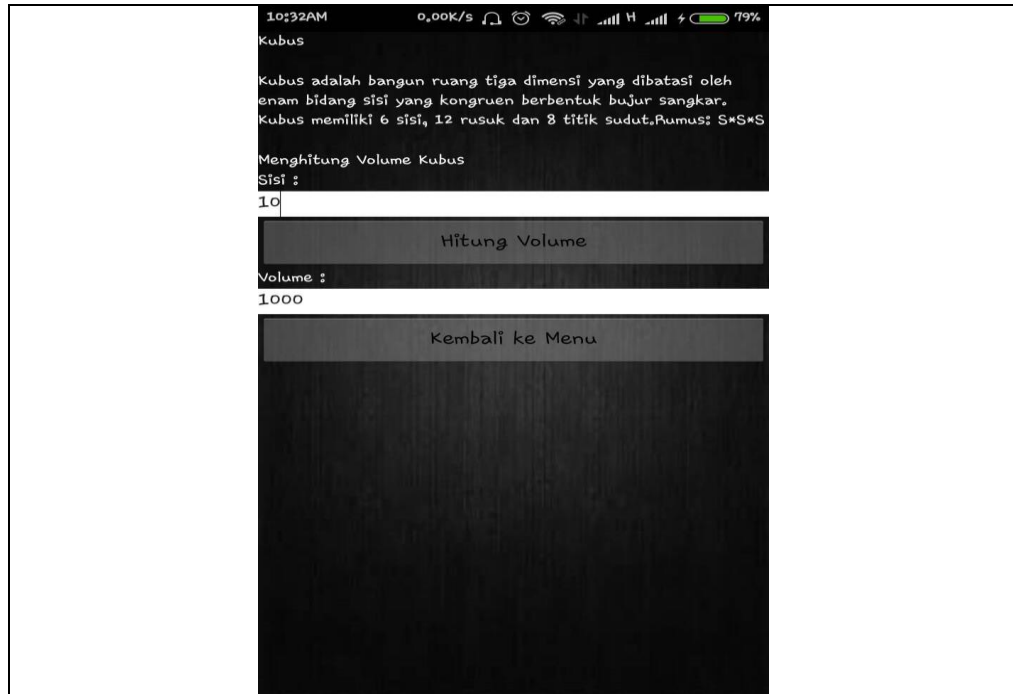
Pada bagian ini akan dipaparkan implementasi berupa aplikasi android dari program yang telah dibuat menggunakan Android Studio.

## 1. Tampilan Awal



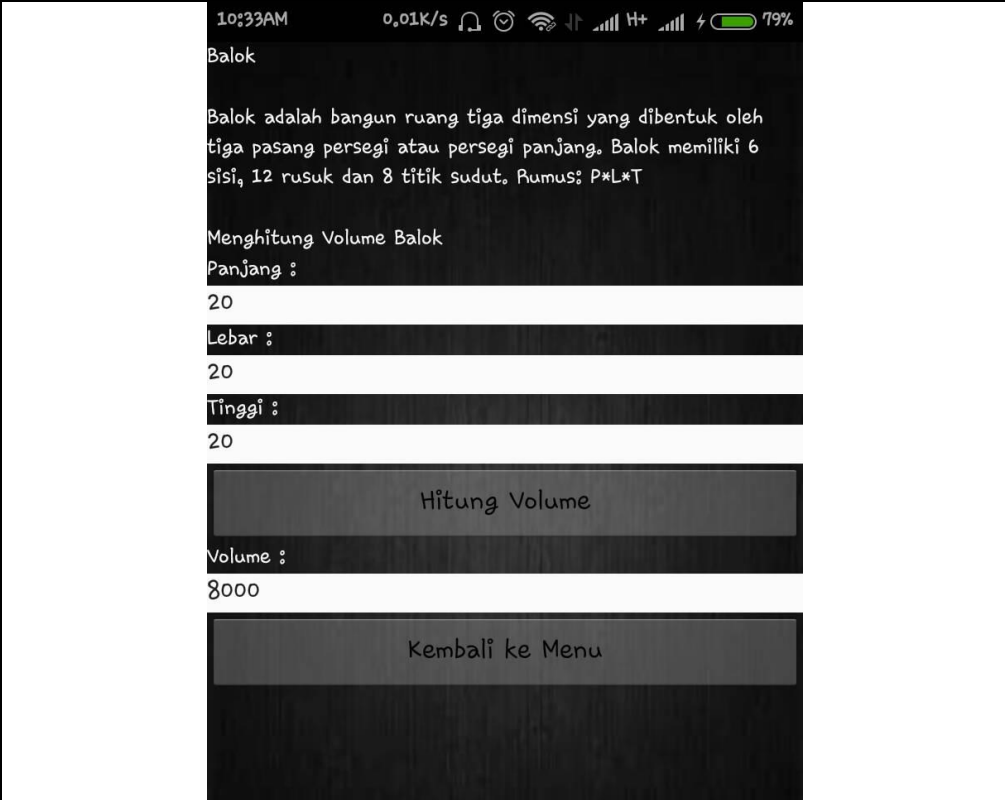
Gambar diatas merupakan tampilan awal dari aplikasi menghitung bangun ruang. Terdapat pilihan bangun ruang yang dapat dipilih oleh *user*, bangun ruang yang dapat dipilih oleh user antara lain kubus, balok dan bola. Tampilan awal ini juga terdapat menu tentang yang berisi tentang informasi pembuat program dan menu keluar.

## 2. Kubus



Gambar diatas merupakan tampilan yang akan diasilkan jika *user* memilih menu bangun ruang kubus. Gambar diatas akan menampilkan penjelasan tentang kubus dan *text box* yang akan diinputkan oleh *user* untuk menghitung volume kubus.

### 3. Balok



10:33AM 0.01K/s 79%

Balok

Balok adalah bangun ruang tiga dimensi yang dibentuk oleh tiga pasang persegi atau persegi panjang. Balok memiliki 6 sisi, 12 rusuk dan 8 titik sudut. Rumus:  $P \times L \times T$

Menghitung Volume Balok

Panjang :  
20

Lebar :  
20

Tinggi :  
20

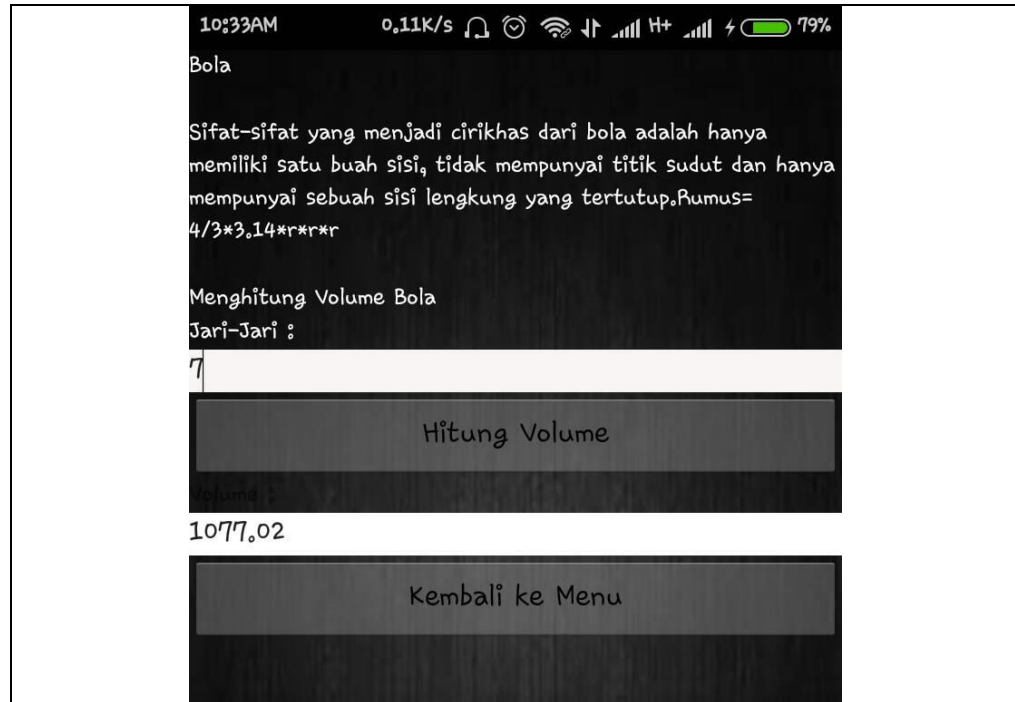
Hitung Volume

Volume :  
8000

Kembali ke Menu

Gambar diatas merupakan tampilan yang akan diasilkan jika *user* memilih menu bangun ruang balok. Gambar diatas akan menampilkan penjelasan tentang balok dan *text box* yang akan diinputkan oleh *user* untuk menghitung volume balok.

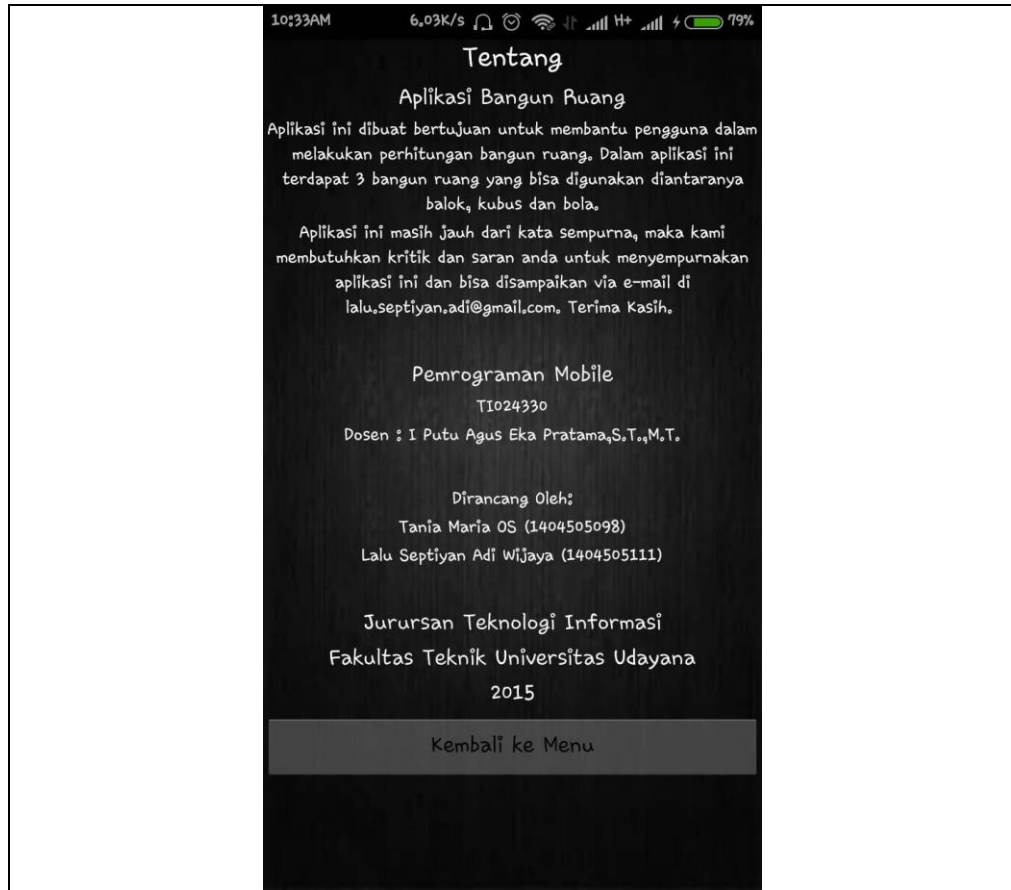
#### 4. Bola



Gambar diatas merupakan tampilan yang akan dihasilkan jika *user* memilih menu bangun ruang bola. Gambar diatas akan menampilkan penjelasan tentang bola dan *text box* yang akan diinputkan oleh *user* untuk menghitung volume bola.



## 5. Tentang



Gambar diatas merupakan tampilan yang dihasilkan oleh menu tentang jika menu tersebut dipilih oleh user. Menu tentang tersebut akan menghasilkan tampilan berupa informasi tentang pembuat program dan definisi dari aplikasi bangun ruang.

### 3.3 Kesimpulan dan Saran

Dalam pembuatan aplikasi menghitung bangun ruang ini dapat diambil sebuah kesimpulan dan tentunya penulis masih memerlukan saran demi kesempurnaan aplikasi ini dimasa yang akan datang.

#### 3.3.1 Kesimpulan

Kesimpulan Dari uji coba pada aplikasi ini,, ternyata aplikasi Penghitungan Rumus Matematika Bangun ruang cukup menarik, dan dapat digunakan oleh *user* khususnya pelajar untuk menghitung Rumus Dasar Matematika, serta dapat membantu proses pembelajaran. Pada aplikasi ini juga terdapat kalkulator

penghitungan berdasarkan rumus-rumus yang ada, untuk mendapatkan hasil yang akurat dari penghitungan rumus tersebut. Aplikasi Penghitungan Matematika Bangun ruang sangat mudah digunakan karena dapat di-instal dalam sistem operasi android berbagai versi dan ukurannya pun sangat kecil sehingga tidak banyak mengurangi space memory. Pembuatan Aplikasi ini juga memberikan manfaat bagi penulis sehingga penulis dapat lebih memahami dan mengerti lebih jauh tentang perkembangan software pada handphone android.

### **3.3.2 Saran**

Pembuatan Aplikasi ini dapat memudahkan user dalam perhitungan rumus matematika untuk menghitung Volume bangun ruang. Namun terbatasnya pilihan bangun ruang yang dapat di hitung, sehingga *user* tidak sepenuhnya mempelajari berbagai macam perhitungan pada ap-likasi ini. Oleh sebab itu, penulis menyarankan agar dapat mengembangkan dan menyempurnakan aplikasi ini agar lebih baik lagi serta memperbaiki aplikasi ini jika ternyata ditemukan kesalahan dalam perhitungannya.

## DAFTAR PUSTAKA

- [1] Anonim1. 2011. *Pengenalan Bahasa Java*. Diakses melalui <http://julio.staff.ipb.ac.id/files/2011/12/JENI-Intro1-Bab02-Pengenalan-Bahasa-JAVA.pdf> pada 14 Mei 2016
- [2] Anonim2. *Pengertian Android*. Diakses melalui <http://repository.usu.ac.id/bitstream/123456789/43703/4/Chapter%20II.pdf> pada 14 Mei 2016
- [3] Dharwiyanni, Sri dan Romi Satria Warhono. 2003. *Pengantar Unified Modeling Language (UML)*. Diakses melalui <http://mirror.unej.ac.id/iso/dokumen/ikc/yanti-uml.pdf> pada 14 Mei 2016.
- [4] Eka Pratama, I Putu Agus. 2014. *Smart City Beserta Dengan Cloud Computing Dan Teknologi – Teknologi Pendukung Lainnya*. Informatika: Bandung.
- [5] Jasa, Lie. 2015. *Mobile Programming*. Diakses melalui <http://liejasa.dosen.narotama.ac.id/files/2011/09/Mobile-Programming-pertemuan-ke-1-27-Februari-2012.pdf> pada 14 Mei 2016
- [6] Novianto. 2011. *Pengenalan Bahasa Pemrograman Java*. Diakses melalui [file:///C:/Users/user/Downloads/1\\_Pengenalan\\_Java.pdf](file:///C:/Users/user/Downloads/1_Pengenalan_Java.pdf) pada 14 Mei 2016.
- [7] Rolliawati, Dwi. 2014. *Pemrograman Berorientasi Objek*. Diakses melalui <http://rolliawati.dosen.narotama.ac.id/files/2014/03/PEMROGRAMA-N-BERORIENTASI-OBYEK.pdf> pada 14 Mei 2016.