BAB II

LANDASAN TEORI

2.1. Tinjauan Jurnal

Dalam penyusunan skripsi ini penulis merujuk pada beberapa jurnal tentang perancangan aplikasi *android* dan implementasi sistem *android.*

> Menurut Rini, dkk (2015:87), Kamus elektronik saat ini sedang populer terutama dikalangan generasi muda. Kemudahan dalam melakukan pencarian makna kata dengan aplikasi kamus elektronik menjadi pemicunya dari pada menggunakan kamus berbentuk buku. Kamus elektronik mengacu pada kamus yang digunakan secara elektronik baik dalam bentuk *compact disk* atau *online*. Perbedaan antara kamus cetak dengan kamus elektronik adalah bahwa kamus elektronik menangani lebih besar jumlah data yang lebih besar dan mengoperasikan terjemahan.
>
> Menurut Setiawan, dkk (2014:344), Melestarikan budaya bahasa daerah dapat dilakukan dengan banyak cara. Dengan kemajuan teknologi yang sangat pesat pada saat ini dapat diciptakan sebuah aplikasi yang dapat mengakomodir kebutuhan setiap pengguna akan sebuah aplikasi penerjemah yang mudah dibawa serta dapat digunakan kapanpun dan dimanapun dengan efektif. Aplikasi yang cukup memenuhi kriteria dalam hal tersebut adalah aplikasi penerjemah bahasa Jawa-Indonesia yang bebasis *android*. Aplikasi ini menyediakan informasi ilmu pengetahuan tentang kosakata dari bahasa Jawa diterjemahkan kedalam bahasa Indonesia.

Berdasarkan tinjauan jurnal diatas, penulis menyimpulkan bahwa aplikasi kamus saat ini sangat diperlukan sebagai media pencarian kata dalam bahasa daerah yang dapat digunakan untuk mempermudah adanya komunikasi. Dengan mempelajari bahasa daerah, secara langsung berarti kita telah berupaya untuk melesatrikan salah satu budaya bangsa yaitu bahasa daerah.

2.2. Konsep Dasar Pemprograman

Konsep dasar pemprograman merupakan peranan terpenting dalam merancang, menyusun dan mengembangkan suatu program, khususnya program aplikasi yang besar dan kompleks. Pendekatan terstruktur (*structure design*) dilakukan dengan cara memecah-mecah suatu masalah yang besar dan rumit menjadi beberapa masalah yang lebih kecil dalam bentuk modul-modul sehingga menjadi cukup mudah untuk ditangani.

> Yuswanto (2008:7) menerangkan bahwa:
> Program merupakan kata, ekspresi, pernyataan atau kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur berupa urutan langkah untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemprograman sehingga dapat dieksekusi oleh komputer. Pemprograman terstruktur merupakan proses kerja untuk mengimplementasikan urutan langkah dalam menyelesaikan suatu masalah dalam bentuk program, memiliki rancang bangun yang terstruktur dan tidak berbelit-belit sehingga mudah ditelusuri, dipahami dan dikembangkan oleh siapa saja.

2.3. Kamus

> Menurut Rini, dkk (2015:88), Ada beberapa istilah dalam bahasa arab yang dipakai untuk menyebut kamus yaitu mu'jam, qamus, fihris, mausu'ah (ensiklopedi) dan musrid (indeks, glosarium). Semua istilah tersebut mengarah kepada satu pengetian bahwasanya kamus adalah kumpulan kosakata yang dilengkapi makna atau arti dan keterangan lain yang bertujuan untuk menjelaskan informasi yang berhubungan dengan kata-kata yang termuat didalam daftar tersebut. Kesemua kosakata beserta maknanya disusun secara teratur, berurutan berdasarkan sistematika tertentu oleh penyusun kamus untuk mempermudah pengguna atau pembaca dalam memahami makna dan informasi tentang kata yang dicari.

Aplikasi kamus yang akan dirancang ini merupakan alat bantu untuk mencari arti kata bahasa Indonesia ke bahasa Jawa dan sebaliknya. Dengan adanya aplikasi kamus ini tentunya tidak perlu lagi membawa kamus dalam bentuk buku yang

tebal untuk mencari kata disetiap halamanya. Dapat dikatakan kamus dalam bentuk aplikasi ini lebih efisien dan praktis.

2.4. Bahasa Jawa

Rohman, dkk (2015:43) mengemukakan bahwa "Bahasa Jawa adalah bahasa yang digunakan penduduk suku bangsa Jawa di Jawa tengah, Yogyakarta dan Jawa timur."

Saat ini penggunaan bahasa Jawa hampir menyeluruh ke pelosok negeri di Indonesia bahkan ada beberapa negara tetangga seperti Suriname juga menggunakan bahasa Jawa. Pada dasarnya, penggunaan bahasa Jawa dibagi menjadi tiga bahasa yaitu:

1. Bahasa Jawa Ngoko (kasar)

   Bahasa jawa ngoko dipergunakan oleh anak- anak kepada anak-anak, orang yang sudah dekat (teman dekat) dan dipergunakan oleh orang yang lebih tua dari lawan bicara.

2. Bahasa Krama Madya

   Bahasa krama madya dipergunakan oleh murid kepada guru, orang yang lebih muda dengan orang yang lebih tua, anak kepada orang tua dan dipergunakan oleh pegawai terhadap pimpinanya.

3. Bahasa Krama Inggil

   Bahasa krama inggil dipergunakan untuk meninggikan derajat lawan bicara dan untuk merendahkan diri sendiri.

2.5. *Android*

Sadeli (2014:2) mengemukakan bahwa "*Android* merupakan sistem operasi berbasis *Linux* yang digunakan untuk perangkat *mobile* (*smartphone*) atau pun perangkat *tablet* (PDA). "

Android banyak mendapat pujian dari berbagai pihak sebagai *platform mobile* pertama yang lengkap. Kata lengkap dalam hal ini memungkinkan para perancang sistem *android* untuk melakukan pendekatan-pendekatan yang komprehensif ketika mereka sedang mengembangkan *platform android*. *Android* merupakan sistem operasi yang aman dan banyak menyediakan *tool* dalam membangun perangkat lunak dan memungkinkan sebuah peluang yang besar untuk pengembangan aplikasi. *Android* juga merupakan *platform* yang terbuka (*open source*), artinya pengembang dapat dengan bebas untuk mengembangakan aplikasi. Selain itu, *android* juga merupakan *platform* yang gratis (*free*) sehingga tidak ada lisensi atau biaya royalti untuk aplikasi yang dikembangkan pada *platform android*.

2.5.1. Sejarah singkat *Android*

Pada awalnya sistem aplikasi *android* dikembangkan oleh Android Inc. Yang kemudian dibeli Google pada tahun 2015. Dalam usaha mengembangkan *android*, pada tahun 2007 dibentuklah Open Handset Alliance (OHA), sebuah korsorsium dari beberapa perusahaan, yaitu Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel dan T-Mobile dengan

tujuan untuk mengembangkan standar terbuka untuk perangkat *mobile*. Pada tanggal 9 Desember 2008, diumumkan bahwa 14 anggota akan bergabung proyek *android*, termasuk Packet Video, ARM Holdings, Atheros Communication, Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp dan Vedafone Group Plc.

2.5.2. Kelebihan *Android*

Android merupakan *smartphone* yang benar-benar pintar. Di bawah ini merupakan paparan dari beberapa keunggulan *android* menurut Masruri (2013:3):

1. *Open Source*

   Android merupakan sistem operasi berbasis *Linux* yang berarti menyediakan *platform* terbuka bagi siapa saja yang ingin mengembangkanya. Hal ini terbukti dengan banyaknya versi *android* yang semakin tinggi versinya maka semakin canggih pula *android* tersebut.

2. Magnet Google

   Android merupakan salah satu aset yang dimiliki oleh Google. Jadi semua yang berhubungan dengan Google akan otomatis tersinkronisasi dengan ponsel *android* seperti: Youtube, Gmail, Google Maps, Google Talk dan masih banyak lagi.

3. *Multitasting*

   Ponsel *android* mampu mengerjakan tugas secara bersamaan tanpa harus menunggu aplikasi yang lain dihentikan, itulah yang dinamakan multitasting. Misalnya dapat melakukan *browsing* serta mendengarkan lagu dalam waktu yang bersamaan.

4. Dukungan dari vendor papan atas

Kecanggihan *android* mampu mengambil hati para vendor papan atas sehingga mampu melonjakan popularitas serta kemampuan *android*. Hadirnya *android* membawa berkah yang besar bagi para vendor karena dengan *android* penjualan produk mereka menjadi bertambah. Misalnya vendor dari samsung ketika ponsel mereka belum memakai sistem operasi *android* volume penjualanya bisa dikatakan kurang bagus di pasaran, akan tetapi setelah menggandeng *android* vendor ternama tersebut langsung mendapatkan tempat dihati masyarakat.

5. *Easy Notification*

Android memberikan kemudahan untuk mengatur pemberitahuan, mulai dari terima sms, terima *email*, terima pesan *facebook* dan masih banyak lagi dengan nada yang berbeda-beda. Ponsel *android* sangat cocok digunakan untuk orang yang gemar bereksperimen dengan melakukan percobaan-percobaan baru didalamnya.

6. *Beautifull widget*

Dalam *homescreen* ponsel *android* dapat ditambahkan *widget* yang bermacam-macam seperti halnya dalam sebuah komputer. Dalam ponsel *android* dapat diberikan menu *shortcut* sebuah aplikasi. Hal itu tentunya sangat memudahkan dan dapat mempercepat waktu untuk melakukan perintah pada sebuah aplikasi.

7. *Modding*

Sistem operasi terbuka benar-benar dimanfaatkan untuk orang-orang yang gemar mengutak-atik atau modifikasi ponsel. Hal itu terbukti dengan adanya modifikasi dari versi ke versi. Ponsel *android* dapat dimodifikasi dari segi tampilan sampai dengan yang paling dalam, yaitu modifikasi ROM.

2.5.3. Versi *Android*

Beberapa versi *android* yang telah dirilis oleh Google menurut Sadeli (2014:2), antara lain:

1. *Android Versi 1.0 (Astro) API Level 1*

   Android versi ini dirilis pada tanggal 23 september 2008. Versi ini disematkan untuk pertama kali pada ponsel *smartphone* HTC Dream.

2. *Android Versi 1.1 (Bender) API Level 2*

   Android versi ini dirilis pada tanggal 9 Februari 2009. Versi ini memiliki beberapa fungsi seperti pembaruan *estetis* pada aplikasi, jam alarm, voice search, pengiriman pesan dengan *Gmail* dan pemberitahuan *email*.

3. *Android Versi 1.5 (Cupcake) API Level 3*

   Android versi ini dirilis pada pertengahan Mei 2009. Versi ini memiliki beberapa fungsi pembaruan seperti merekam dan menonton video, mengunggah video ke *youtube*, dukungan *bluetooth* A@DP, animasi layar, serta keyboard yang dapat disesuaikan dengan sistem.

4. *Android Versi 1.6 (Donut) API Level 4*

   Android versi ini dirilis pada 15 september 2009. Dalam versi ini terdapat penambahan serta perbaikan dari versi sebelumnya seperti penggunaan indikator baterai, *dial contact*, kamera, resolusi VWGA dan beberapa fitur lainya.

5. *Android Versi 2.0/2.1 (Eclair) API Level 5 – 7*

   Android versi ini dirilis tanggal 26 Oktober 2009. Pada versi ini terdapat penambahan fitur pengoptimalan *hardware*, peningkatan *Google Maps*, perubahan UI (*User Inerface*) dengan *browser* baru dan dukungan HTML5. Dukungan *flash* untuk kamera 3.2 MP, *digital zoom* dan pengembangan baru teknologi *bluetooth*

6. *Android Versi 2.2 Froyo (Froze Yoghurt) API Level 8*

   Android versi ini dirilis pada tanggal 20 Mei 2010. Pada versi ini terdapat dukungan *Adobe Flash* 10.1. Peningkatan kinerja lebih baik sampai dua kali lipat dari versi sebelumnya yang terintegrasi dengan V8 *JavaScript engine* yang dipakai Google Chrome, pemasangan aplikasi dalam CD card, WiFi Hotspot portable, *auto update* aplikasi *android market*.

7. *Android Versi 2.2 (Gingerbread) API Level 9 – 10*

   Android versi ini dirilis tanggal 6 Desember 2010. Terdapat peningakatan khusus pada aplikasi *game*, resolusi WXGA dan beberapa fitur lain. Pada versi ini memiliki pengguna terbanyak dibandingkan dengan versi sebelumnya.

8. *Android Versi 3.0/3.1 (Honeycomb) API Level 11 – 13*

   Android versi ini dirilis pada tanggal 22 Februari 2011. Versi ini mendukung ukuran layar yang lebih besar. UI (User Interface) pada *honeycomb* berbeda karena didesain khusus untuk *tablet*. *Tablet* pertama yang menggunakan versi ini adalah Motorola Xoom.

9. *Android Versi 4.0 – 4.0.4 ICS (Ice Cream Sandwich) API Level 14 -15*

   Android versi ini dirilis pada tanggal 19 Oktober 2011. *Smartphone* yang pertam kali menggunakan sistem operasi ini adalah Samsung Galaxy Nexus. Versi ini memiliki fitur baru seperti membuka kunci dengan pengenalan wajah, jaringan sosial dan beberapa fitur tambahan lainya. Secara teori versi sebelumnya seperti *Gingerbread* dapat di *upgrade* ke versi ini.

10. *Android Versi 4.1 – 4.3.1 (Jelly Bean)API Level 16 – 18*

    Android versi ini banyak sekali fitur penambahanya seperti *widget* baru, *photo sphere*, *power control lock screen widget* dan beberapa fitur lainya. *Smartphone* yang pertama kali menggunakan versi ini adalah LG Google Nexus 4. *Android* versi ini dirilis pada tanggal 24 Juli 2013.

11. *Android Versi 4.4 – 4.4.4 (KitKat) API Level 19*

    Android versi ini dirilis pada tanggal 31 Oktober 2013. Beberapa pembaruan seperti antarmuka dengan *bar status* , navigasi transparan pada layar depan, sensor *batching*, *step detector* dan *counter API*, peningkatan audio, aksesibilitas API dan beberapa fitur lainya.

12. *Android Versi 5.0 - 5.1 (Lollipop) API Level 21 - 22*

    Android versi ini dirilis pada tanggal 17 Oktober 2014. Versi ini memfokuskan pada desain dan performa sehingga lebih nyaman digunakan. Ditambah dengan kapasitas baterai yang lebih tinggi dari pada versi sebelumnya

2.5.4. *Android Sofware Developmnet Kit* (SDK)

Android Sofware Developmnet Kit (SDK) adalah *tools Application Programming Interface* (API) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform android* menggunakan bahasa pemograman java. SDK *android* berisi *debugger, library, emulator*, dokumentasi contoh kode program dan tutorial. *Android* memberi kesempatan untuk membuat aplikasi yang dibutuhkan, namun bukan merupakan aplikasi bawaan *smartphone*.

2.6. *Java*

Menurut Hermawan (2011:20) "*Java* merupakan bahasa pemprograman untuk membangun aplikasi pada sistem aplikasi *android*." Oleh karen itu, untuk membangun aplikasi pada sistem operasi ini diperlukan dasar tentang pemprograman *java. Java* merupakan pemprograman berorientasi *object*. Oleh karena itu, setiap konsep yang diimplementasikan dalam *java* berbentuk dalam kelas. Kelas ini mendefinisikan *object-object* yang memiliki kesamaan perilaku dan keadaan. Pada *java* terdapat kumpulan kelas standar yang dikenal dengan *Aplication Programing Interface (API) Java*, selain itu dapat juga dideskripsikan kelas sendiri sesuai kebutuhan.

2.7. *UML*

UML (*Unified Modelling Language)* suatu metode pemodelan secara visual untuk sarana perancangan sistem berorientasi objek. *UML* merupakan suatu bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga pendokumentasian sistem *software*. Saat ini *UML* sudah menjadi bahasa standar dalam penulisan *blue print software*.

2.7.1. Fungsi *UML*

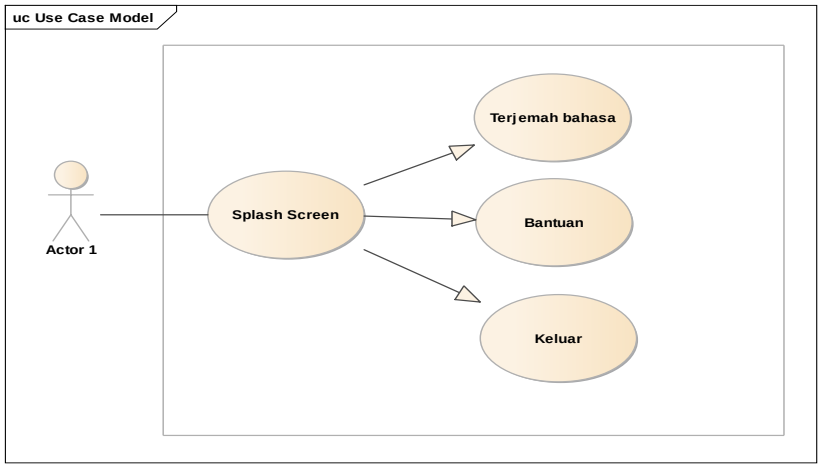
Fungsi dari penggunaan *UML* diantaranya:

1. Dapat memberikan bahasa pemodelan visual kepada pengguna dari berbagai macam pemprograman maupun proses rekayasa.
2. Dapat menyatukan praktek-praktek terbaik yang ada dalam pemodelan.
3. Dapat memberikan model yang siap untuk digunakan, merupakan bahasa pemodelan visual yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah.
4. Dapat berguna sebagai *blue print*, sebab sangat lengkap dan detail dalam perancangannya yang nantinya akan diketahui informasi yang detail mengenai koding suatu program.
5. Dapat memodelkan sistem yang berkonsep orientasi objek, jadi tidak hanya digunakan untuk memodelkan perangkat lunak *(software)* saja.
6. Dapat menciptakan suatu bahasa pemodelan yang nantinya dapat dipergunakan oleh manusia maupun oleh mesin.

2.7.2. Jenis-jenis diagram *UML*

1. *Use case Diagram*

Use case diagram yaitu salah satu jenis diagram pada *UML* yang menggambarkan interaksi antara sistem dan aktor. *Use case* diagram juga dapat mendeskripsikan tipe interaksi antara si pemakai sistem dengan sistemnya.

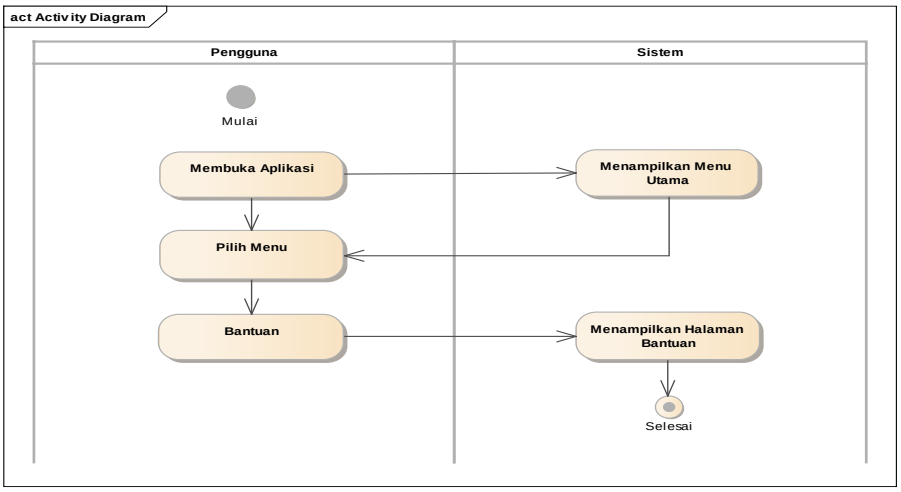


Sumber: Dokumentasi pribadi

Gambar II.1 .Contoh *Use case* diagram

2. *Activity Diagram*

Activity diagram atau diagram aktivitas yaitu salah satu jenis diagram pada *UML* yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.

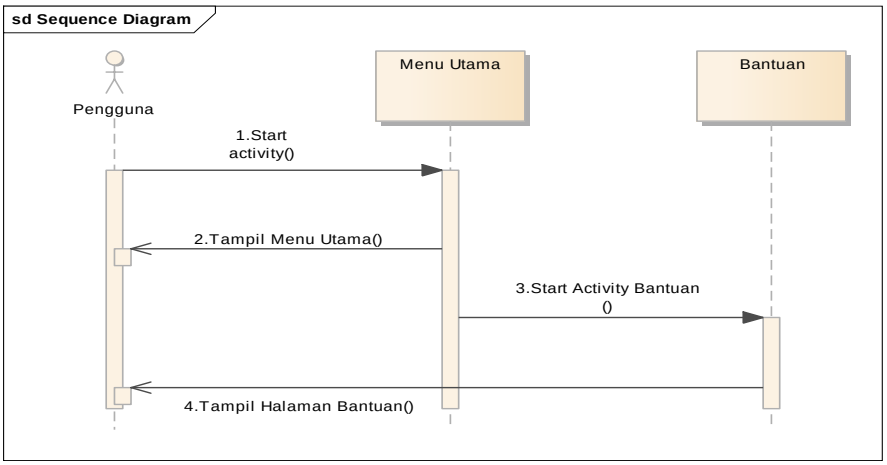


Sumber: Dokumentasi pribadi

Gambar II.2 Contoh *Activity* diagram

3. *Sequence Diagram*

Sequence diagram menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence diagram* juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada *use case diagram.*

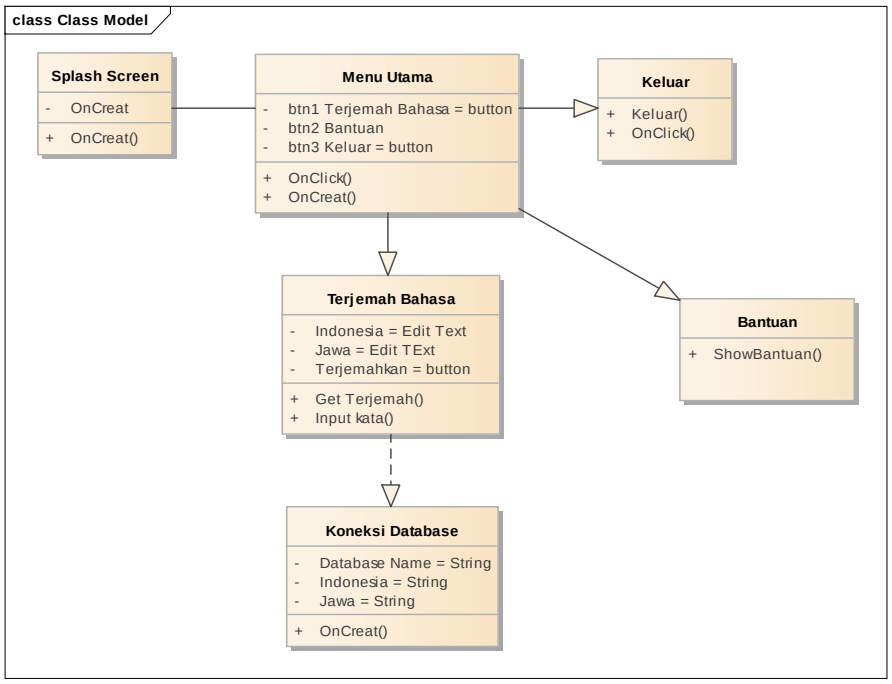


Sumber: Dokumentasi pribadi

Gambar II.3 .Contoh *Sequence* diagram

4. *Class* Diagram

Class diagram yaitu salah satu jenis diagram pada *UML* yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut.



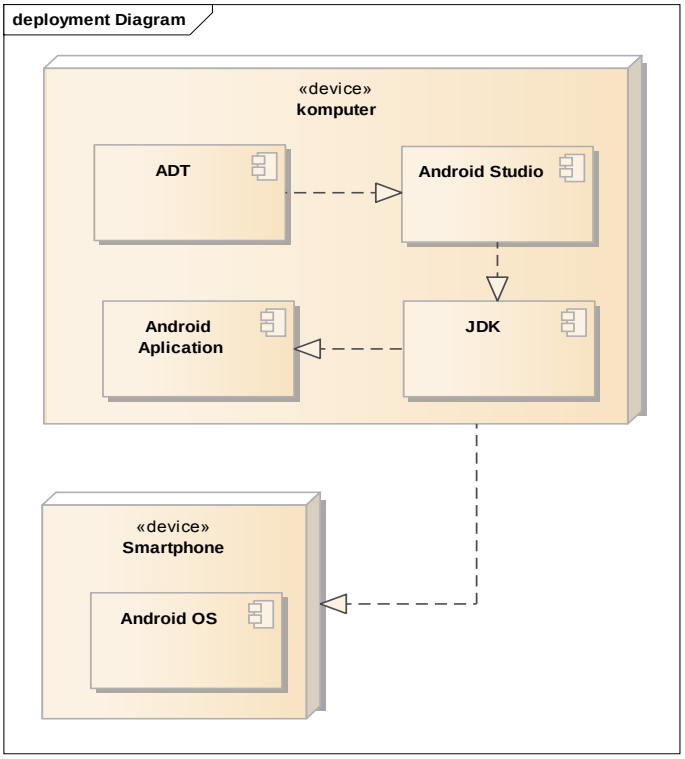
Sumber: Dokumentasi pribadi

Gambar II.4 .Contoh *Class* diagram

5. *Deployment Diagram*

Deployment diagram yaitu salah satu diagram pada *UML* yang menunjukan tata letak suatu sistem secara fisik. Dapat juga dikatakan untuk menampilkan bagian-bagian *softwere* yang terdapat pada *hardwere* dan digunakan untuk menerapkan suatu sistem dan hubungan antara

komponen *hardwere*. Jadi *Deployment diagram* intinya untuk menunjukan letak *softwere* pada *hardwere* yang digunakan sistem.



Sumber: Dokumentasi pribadi

Gambar II.5. Contoh *Deployment diagram*

2.8. *SQLite*

Menurut Andi (2013:195) "*SQLite* merupakan sebuah sistem manajemen basis data rasional yang bersifat *ACID-complaint*. *Software* ini ditulis menggunakan bahasa C. Protokol komunikasi utama yang digunakan adalah melalui pemanggilan API langsung melalui bahasa pemprograman". Mekanisme seperti ini tentu membawa keuntungan karena dapat mereduksi *overhead* dan *latency time*, yang intinya adalah lebih sederhana. Seluruh elemen basis data (definisi data, tabel, indeks dan data) disimpan sebagai sebuah file. Salah satu

keunggulanya adalah hanya memakan sedikit memori saat dijalankan. *SQLite* saat ini tersedia di semua perangkat *android*. Pengguna tidak memerlukan *setup database* administrasi menggunakan *database SQLite*, tetapi pengguna hanya perlu mendefinisikan *SQL statement* untuk membuat dan *mengupdate database*. Setelah itu, *database* otomatis diatur oleh *platform android* untuk penggunanya.

2.8.1. *SQLite Database*

SQLiteDatabase merupakan *class* dasar untuk *database SQLite* di *android*. Disini pengguna bisa membuka *query*, *update* dan menutup *database*. *SQLiteDatabase* mempunyai banyak *method*, seperti *method insert*( ), *update*( ) dan *delete*( ). Ada juga *execSQL*( ) yang bisa mengeksekusi SQL secara langsung. Nilai *Object ContentValues* memungkinkan mendefinisikan *key/value. Key* merepresentasikan identitas kolom tabel, sedangkan *value* merepresentasikan konten dari *record* tabel di kolom. *ContenValues* bisa digunakan untuk *insert* dan *update entri database.*

2.9. Metode Algorima

Dalam penulisan laporan skripsi ini, penulis menggunakan metode algoritma pencarian. Algoritma pencarian merupakan algoritma yang menerima sebuah argumen, data atau kunci dengan langkah mencari rekaman pada data atau kunci yang telah tersedia. Setelah proses pencarian dilaksanakan, akan diperoleh salah satu dari dua kemungkinan, yaitu data yang dicari ditemukan (*successful*) atau tidak ditemukan (*unsuccessful*). Macam teknik pencarian data antara lain :

1. Pencarian Biner

Terdapat metode pencarian pada data terurut yang paling efisien yaitu metode pencarian bagi dua atau pencarian biner (*binary search*). Metode ini digunakan untuk kebutuhan pencarian dengan waktu yang cepat. Prinsip pencarian dengan membagi data atas dua bagian. Data yang disimpan didalam larik harus sudah terurut. Berikut merupakan contoh algoritmanya:

```
Class BinaryNode {
void printInOrder( )
{
if( left != null )
left.printInOrder( ); // Left
System.out.println( element ); // Node
if( right != null )
right.printInOrder( ); // Right

}
}

class BinaryTree {
public void printInOrder( )
{
if( root != null )
root.printInOrder( ); }}
```

2. Pencarian Sequensial

Pencarian sequensial juga dapat dibilang pencarian linear yang merupakan metode pencarian yang paling sederhana. Pencarian sequensial adalah proses membandingkan setiap elemen larik satu per satu secara beruntun, mulai dari elemen pertama sampai elemen yang dicari ditemukan atau seluruh elemen sudah diperiksa. Algoritma pencarian berurutan dapat dicontohkan sebagai berikut:

```
#include <stdio.h>
#include <conio.h>
void main(){
int data[8] = {8, 10, 6, 2, 11, 7, 1, 100};
int cari;
int flag=0;
printf("masukkan data yang ingin dicari = ");scanf("%d",&cari);
for(int i=0;i<8;i++){
if(data[i] == cari) flag=1;
} if(
flag==1) printf("Data ada!\n");
else printf("Data tidak ada!\n");
getch();
return 1;
}
```

2.10. Pengujian sistem

Untuk mengetahui kekurangan yang ada pada aplikasi yang penulis buat, dibutuhkan adanya pengujian sistem. Dalam pembuatan aplikasi ini penulis menggunakan dua macam pengujian sistem yaitu:

1. *White Box Testing*

   Menurut Shalahuddin (2014:276) "Pengujian *White Box* yaitu menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan dan keluaran yang sesuai dengan spesifikasi kebutuhan." Pengujian *white box* dilakukan dengan memeriksa lojik dari kode program. Pembuatan kasus uji bisa mengikuti standar pengujian dari standar pemograman yang seharusnya. Contoh dari pengujian *white box* misalkan menguji alir (dengan menelusuri) dan pengulangan (*looping*) pada logika pemrograman. *White box* testing merupakan pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara prosedural untuk membagi pengujian kedalam beberapa kasus pengujian. Secara sekilas dapat diambil kesimpulan *white box* testing merupakan petunjuk untuk mendapatkan program yang benar secara keseluruhan.

   Dalam teknik ini terdapat persamaan yang digunakan yaitu:

   $V(G) = E - N + 2$

   Keterangan:

   V (G) = *Cyclomatic Complexity* untuk *flow graph G*

   E = Jumlah *edge* (panah)

N = Jumlah *node* (lingkaran)

2. *Black Box Tasting*

Black box tasting merupakan pengujian yang dilakukan hanya dengan mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat kotak hitam. Kita hanya bisa melihat penampilan luarnya saja tanpa tahu ada apa dibalik bungkus hitamnya. Sama seperti pengujian black box. Mengevaluasi hanya dari tampilan luarnya *(Interface*) atau fungsionalitasnya, tanpa mengetahui apa sesungguhnya yang terjadi dalam proses detilnya (hanya mengetahui *input* dan *output*).

2.11. Peralatan Pendukung

Dalam penulisan laporan skripsi ini penulis menggunakan beberapa peralatan pendukung yang berupa perangkat keras dan perangkat lunak. Peralatan pendukung tersebut antara lain:

1. *Android Studio*

Android studio merupakan lingkungan pengembangan baru dan terintegrasi penuh yang dirilis oleh Google untuk sistem operasi *android. Android studio* dirancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan juga memberi alternatif lain selain *Eclipse* yang saat ini menjadi IDE (*Integrated Development Environment*) yang paling banyak dipakai.

2. *Photoscape*

Photoscape merupakan perangkat lunak editor citra yang dikembangkan oleh MOOII Tech, Korea. Konsep dasar *Photoscape* adalah "Mudah dan Menyenangkan" sehingga memungkinkan pengguna untuk dengan mudah mengedit foto yang diambil dari kamera digital atau kamera ponsel. *Photoscape* menyediakan antarmuka pengguna sederhana untuk melakukan tambahan foto umum termasuk penyesuaian warna, memotong, mengubah ukuran, percetakan dan animasi GIF.

3. *Enterprise Architect*

Enterprise Architect merupakan sebuah *platform* visualisasi untuk merancang dan membangun sebuah sistem perangkat lunak dan untuk pemodelan proses bisnis. *Enterprise Architect* adalah sebuah alat yang sangat maju yang mencakup seluruh aspek dari mulai siklus pengembangan, memberikan penelusuran secara mendetail dari mulai awal tahap mendesain sampai mengontrol perkembangan, pemeliharaan, pengujian dan perubahan kontrol. *Tools Enterprise Architect* dikembangkan oleh Sparx Systems, sebuah perusahaan Australia dengan inovasi dan pengembangan dalam pemodelan UML. Sparx System adalah anggota dari Object Management Group (OMG).

4. Perangkat Keras (*Harware*)

Berikut merupakan spesifikasi perangkat keras yang digunakan:

a) Komputer

Penulis menggunakan laptop HP dengan spesifikasi sebagai berikut:

1.	System Model	:	HP Pavilion g4 Notebook PC
2.	*Processor*	:	AMD A4 3305 M
3.	Chip Type	:	AMD Radeon Ghaphics Processor
4.	Ram	:	2048 MB
5.	Sistem Operasi	:	Windows 7 Profesional 32-bit
6.	*Hard Disk*	:	500 GB
7.	*Display Graphic*	:	HD 6480 G

b) Ponsel *Android* / *Smartphone*

Penulis menggunakan *Android* Asus Zenfone 5 dengan spesifikasi sebagai berikut:

1.	Chipset	:	Intel Atom Z2580
2.	Ram	:	1 GB
3.	Versi *Android*	:	*Kitkat*
4.	CPU	:	Dual-Core 2 GHZ
5.	Ukuran layar	:	720 X 1280 pixels, 5.0 inches