

APLIKASI MOBILE KAMUS ISTILAH LINUX

Lili Dwi Apriani¹, Maimunah²

¹Program Studi Teknik Informatika STMIK Nusa Mandiri Jakarta

²Program Studi Teknik Komputer Universitas Islam "45" Bekasi,

Email : lilidwiapriani@nusamandiri.ac.id

ABSTRACT

Android is an operating system is open source which provides freedom for developers to develop an application. With the advantages of the android operating system, will help a lot of Android-based smartphone users to enjoy a variety of applications, one such application is a dictionary of Linux. The main purpose of this application is to assist them in finding translation linux programming effectively and efficiently. The type of data used is secondary data where the data was obtained from reference books and literature related to this research. There are several stages in making this application, namely analysis, design, workmanship and evaluation of program models that have been produced. Linux is a dictionary application, can be used on smartphones based on Android version 2.2 (Froyo or Frozen Yoghurt) up to 4.1 Operating Systems namely Jelly Beans in order to test these applications can run smoothly.

Keywords: *Android, Dictionary, Linux, Mobile Programming*

ABSTRAK

Android merupakan sebuah sistem operasi yang bersifat *open source* yaitu memberikan kebebasan bagi para *developer* untuk mengembangkan sebuah aplikasi. Dengan kelebihan dari sistem operasi android, akan banyak membantu pengguna *smartphone* berbasis android untuk dapat menikmati beragam aplikasi, salah satu aplikasi tersebut adalah kamus Linux. Tujuan utama dari aplikasi ini adalah membantu mereka dalam menemukan terjemahan pemrograman linux secara efektif dan efisien. Adapun jenis data yang digunakan adalah jenis data sekunder dimana data diperoleh dari buku referensi dan literatur yang berhubungan dengan penelitian ini. Terdapat beberapa tahapan dalam pembuatan aplikasi ini, yaitu analisa, perancangan, implementasi dan pengujian terhadap model program yang telah dihasilkan. Aplikasi kamus Linux yang telah dirancang dapat digunakan pada *smartphone* yang berbasis android versi 2.2 (*Froyo* atau *Frozen Yoghurt*) sampai dengan 4.1 dengan *Sistem Operasi Jelly Beans*.

Kata Kunci : *Android, Kamus, Linux, Pemrograman Mobile*

1. Pendahuluan

Kamus merupakan sejenis buku yang menerangkan makna kata-kata. Kamus berfungsi untuk membantu seseorang mengenal perkataan baru. Selain menerangkan maksud kata, kamus juga mempunyai pedoman sebutan, asal-usul (*etimologi*) sesuatu perkataan dan juga contoh penggunaan bagi sesuatu perkataan. Kamus teknologi informasi merupakan sebuah kamus yang bisa digunakan untuk mencari istilah dalam teknologi informasi

seperti pada *computer* ataupun pada *handphone*.

Linux adalah *software system* operasi *open source* yang gratis untuk disebar luaskan di bawah lisensi *GNU*. Jadi pengguna diijinkan untuk menginstal pada komputer ataupun menyalin dan menyebarluaskan tanpa harus membayar. Salah satu media dalam pembelajaran Linux adalah kamus yang dapat digunakan untuk mencari arti dari kata-kata ataupun kode yang tidak diketahui. Pemanfaatan *mobile*

phone dalam bidang pendidikan dikenal dengan istilah *mobile learning*. *Mobile learning* merupakan sebuah paradigma baru yang telah menciptakan lingkungan pembelajaran baru yaitu pembelajaran yang didukung oleh *mobile phone* dan transmisi nirkabel berupa teknologi selular.

Mobile learning termasuk dalam kategori *E-Learning* sebagai bentuk pembelajaran yang memanfaatkan perangkat elektronik dan media digital bergerak. Seiring dengan perkembangan teknologi *mobile* yang sangat pesat, maka berbagai aplikasi untuk *mobile* banyak dikembangkan, baik untuk hiburan semata ataupun untuk pembelajaran diantaranya aplikasi *mobile* untuk istilah psikologi (Parno, 2011), istilah komputer (Mulyana, 2014) dan aplikasi kamus bahasa Inggris Indonesia Nias (Nalainia, 2013).

Penggunaan kamus pada perangkat *mobile* sangat praktis dibandingkan dengan penggunaan kamus konvensional berupa buku yang besar dan tebal pada umumnya ataupun penggunaan kamus pada perangkat komputer. Melalui kamus *mobile* jika ingin mencari arti dari sebuah kata dapat digunakan kapanpun dan dimanapun. Selain kamus bahasa, istilah-istilah di bidang teknologi informasi juga telah dibuat dalam aplikasi android dengan menggunakan metode *exact string matching*. *Exact String Matching*, merupakan pencocokan *string* secara tepat dengan susunan karakter dalam *string* yang dicocokkan dalam *string* yang sama. Penerapan metode *exact string matching* dalam proses pencarian kata

kamus teknologi informasi dapat dilakukan dengan cukup baik (Kacaribu, 2014).

Maksud dan tujuan dalam penelitian ini antara lain membuat aplikasi yang dapat berguna serta membantu dalam mempelajari istilah bahasa pemrograman Linux yang berbasis android serta memudahkan pengguna *handphone* untuk mencari makna kata atau mengartikan kata dalam memahami istilah dalam Linux menggunakan metode *Exact String Matching (ESM)*.

2. Bahan Dan Metode

2.1. Bahan

Bahan yang digunakan dalam pembuatan aplikasi kamus linux adalah :

1) Software

Perangkat Lunak (software) yang digunakan adalah

a. Eclipse

Eclipse merupakan tempat untuk membuat projek aplikasi android dan ada beberapa *device* yang harus diinstall di eclipse diantaranya Android SDK dan Android ADT.

b. Java JDK

Perangkat lunak yang digunakan oleh manajemen untuk membangun berbagai aplikasi Java.

c. Aplikasi

Aplikasi yang digunakan adalah aplikasi yang berbasis android sehingga program tersebut dapat digunakan untuk menjalankan fungsinya hanya pada sistem operasi berbasis android.

2) Komponen Mobile

a. Vendor : LG

- b. RAM : 512Mb
- c. OS : 4.1.2 (*Jelly Beans*)

Besarnya RAM yang akan digunakan akan mempengaruhi aplikasi yang akan dibangun, disarankan agar menggunakan RAM minimal 512Mb.

3) Spesifikasi Hardware

Perangkat keras yang dibutuhkan sebagai penunjang pembuatan aplikasi, berupa laptop dengan spesifikasi sebagai berikut :

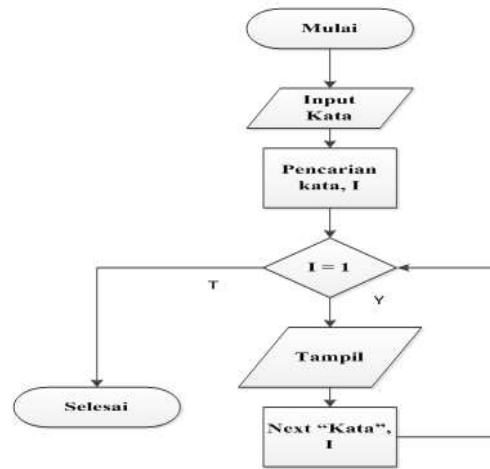
- a. Processor : Intel Core i3 2,53 GHz
- b. Memori : 4.00 GB DDR3
- c. Hard disk : 500 GB HDD SATA
- d. VGA : Intel HD Graphics

2.2. Metode

Metode yang digunakan dalam pembuatan aplikasi kamus Linux dalam penelitian ini adalah *Exact String Matching* (ESM). *Flowchart* metode ESM disajikan pada Gambar 5, dengan algoritma ESM sebagai berikut :

- a. Deklarasi sebuah variable kata yang akan dicari
- b. Deklarasi sebuah variable kata yang akan dimasukkan sebagai perbandingan kata
- c. Buat sebuah variable array
- d. Deklarasi variable *nMax* yang menampung panjang kata dari variable kata
- e. Lakukan perulangan dari nilai posisi 1 sampai dengan jumlah huruf dari variable kata.
- f. Ketika perulangan dilakukan, buat kondisi untuk membandingkan kata yang dicari dan kata yang dimasukkan,

jika ketemu maka proses berhenti dan jika tidak diproses lanjut dengan kata berikutnya.



Gambar 5 . Flowchart *Exact String Matching*(ESM)

Tahapan penelitian dalam aplikasi kamus Linux disajikan pada Gambar 6, terdiri dari 4 tahap, yaitu analisis kebutuhan, perancangan, implementasi dan pengujian. Dalam tahap analisis kebutuhan dilakukan penelitian dan pengamatan tentang apa yang menjadi permasalahan dan kebutuhan dalam kamus Linux. Dari permasalahan dan kebutuhan tersebut dapat disimpulkan solusi dari permasalahan tersebut dan membuat gambaran umum dari aplikasi yang akan dibuat.



Gambar 6. Tahapan Penelitian

Tahap perancangan merupakan pengembangan dari gambaran umum sistem. Dalam tahap perancangan dijelaskan lebih detail tentang isi dari aplikasi yang dibuat yaitu dengan membuat diagram *Unified Modelling Language* (UML) yang meliputi *use case*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram* serta membuat desain *input* dan *output*. Setelah tahap perancangan selesai maka dilakukan tahap implementasi yaitu menerjemahkan desain ke dalam *source code* berbasis Android.

Tahap pengujian dilakukan menggunakan *white box testing* dan *black box testing*. *White Box Testing* (pengujian kotak putih) yaitu menguji perangkat lunak dari segi desain dan kode program apakah mampu menghasilkan fungsi-fungsi, masukan dan keluaran yang sesuai dengan spesifikasi kebutuhan. *White Box Testing* (pengujian kotak putih) dilakukan memeriksa logik dari kode program.

Pembuatan kasus uji dapat mengikuti standar pengujian dari standar pemrograman yang seharusnya. Untuk menghitung Kompleksitas siklomatis ($V(G)$) digunakan persamaan:

$$V(G) = E - N + 2 \dots \dots \dots (1)$$

di mana E adalah jumlah tepi grafik aliran, dan N adalah jumlah *node* grafik aliran. *Black Box Testing* (pengujian kotak hitam) dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang

dibutuhkan. Kasus uji yang dibuat dengan kasus benar dan kasus salah.

3. Hasil Dan Pembahasan

3.1 Hasil

A. Analisa kebutuhan

Aplikasi kamus linux yang dirancang merupakan kamus yang bersifat *mobile* dimana aplikasi kamus tersebut dijalankan pada *handphone* yang mendukung pemrograman android. Emulator yang digunakan adalah eclipse. Pengguna aplikasi kamus Linux dirancang dapat menggunakan perangkat *mobile*-nya untuk mencari kamus linux atau bahasa linux yang belum diketahui. Di dalam tampilan aplikasi, terdapat fitur sejarah linux, pencarian istilah, panduan penggunaan aplikasi dan menu keluar.

B. Perancangan

Aplikasi kamus Linux yang dirancang merupakan kamus yang bersifat *mobile*. Dalam pembuatan aplikasi kamus Linux meliputi perancangan model visualisasi (tampilan pada layar *handphone*) yang merupakan pilihan menu yang disediakan serta proses penerjemah kata berdasarkan algoritma yang digunakan. Perancangan *Unified Modelling Language* (UML) aplikasi kamus Linux sebagai berikut :

1. Use Case Diagram

Use Case Diagram digunakan untuk memodelkan proses berdasarkan perspektif pengguna sistem (Rosa dan Shalahuddin, 2013). Use case proses penerjemah kata di

dalam kamus Linux seperti dalam gambar 7 dan deskripsi sistem seperti dalam Tabel 1.

2. Activity Diagram Kamus

Activity diagram aplikasi kamus Linux seperti dalam Gambar 8.

3. Sequence Diagram

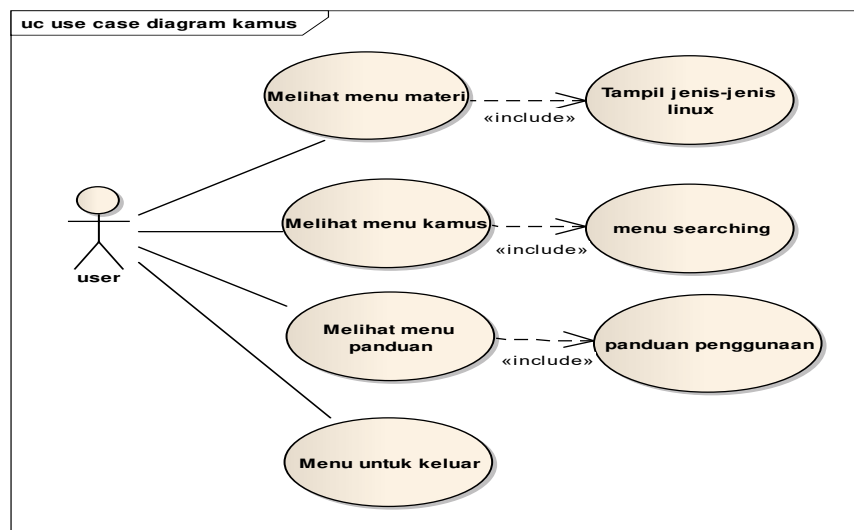
Aplikasi kamus linux mempunyai Sequence Diagram yang terdiri dari Menu Materi, Menu Kamus, Menu Panduan dan Menu Keluar seperti dalam Gambar 9,10,11, dan 12.

4. Class Diagram Menu Utama

Aplikasi kamus Linux mempunyai Class Diagram Menu Utama seperti dalam Gambar 13.

5. Deployment Diagram Kamus

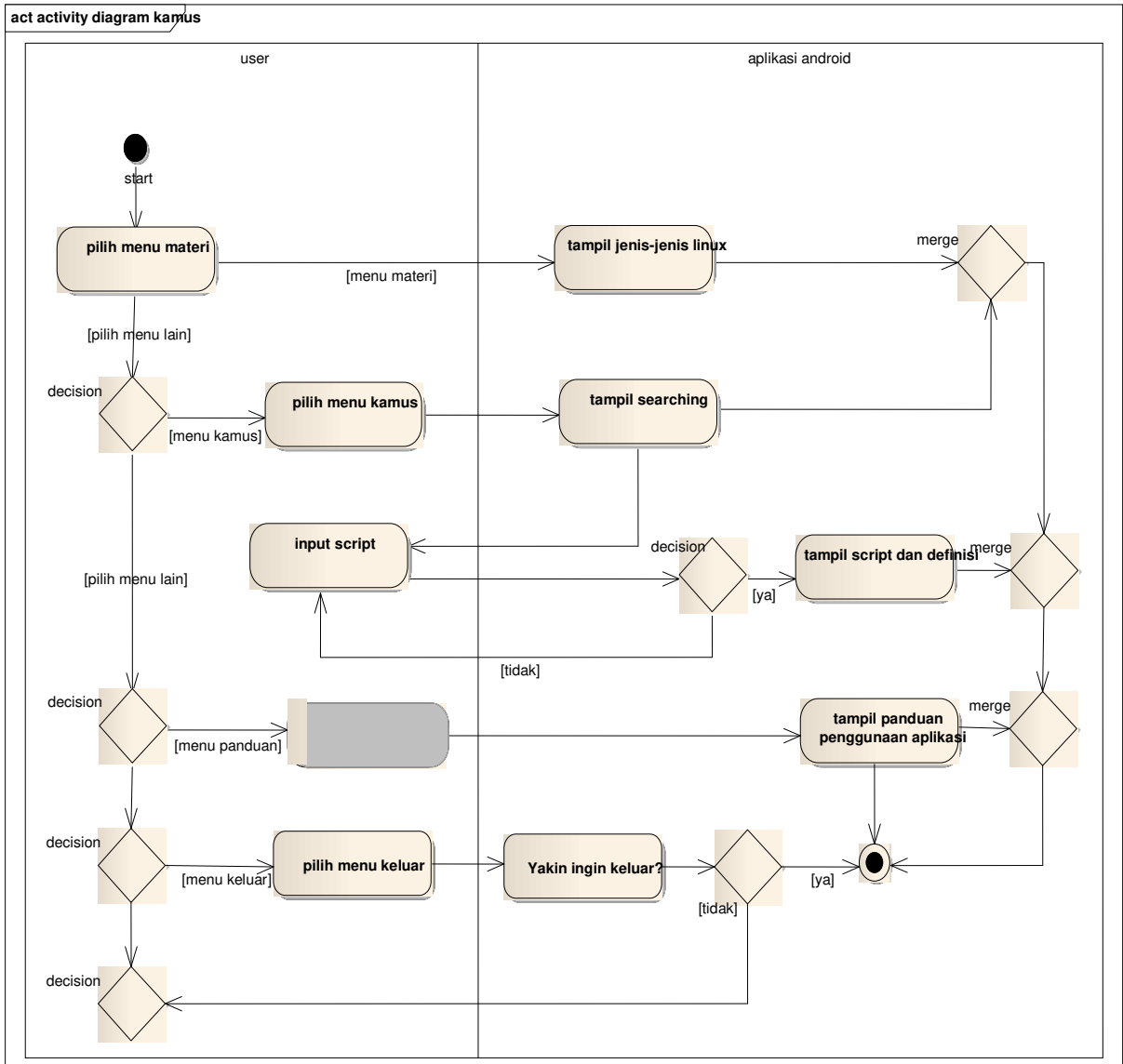
Deployment Diagram aplikasi kamus Linux seperti dalam Gambar 14. Rancangan user interface aplikasi Linux terdiri dari menu utama, menu materi, menu kamus, menu panduan dan menu keluar seperti dalam Gambar 15-19.



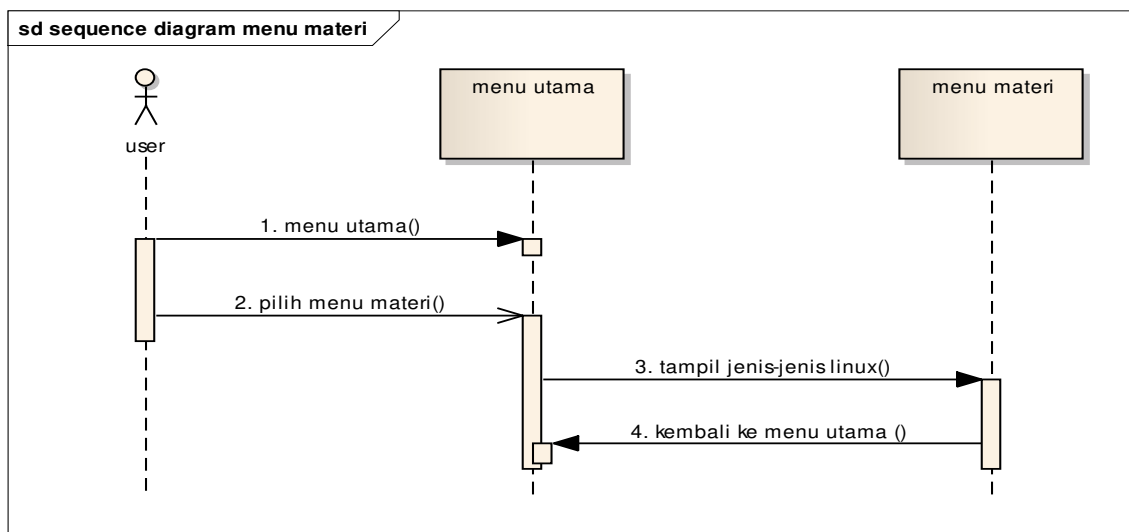
Gambar 7. Use Case Diagram Kamus

Tabel 1. Deskripsi Use Case Diagram Kamus

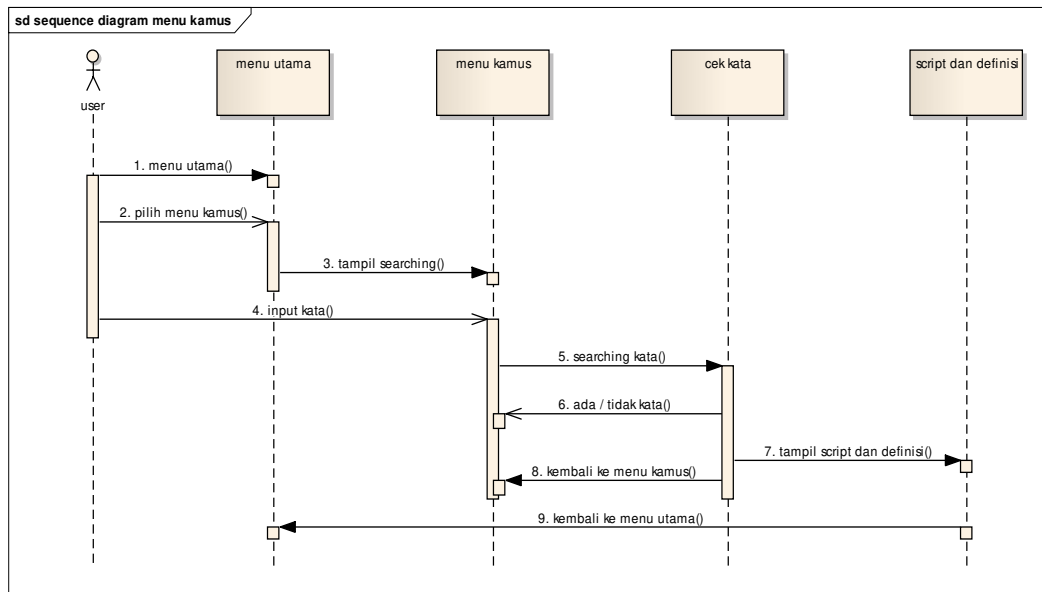
Use	Deskripsi
Actor	User
Alternatif Flow	Jika user tidak menemukan arti dari istilah akan muncul dialog “Terjemahan tidak ditemukan” maka user harus menghapus dan menggantinya.
Brief Description	Pencarian istilah <i>computer</i> untuk mendapatkan terjemahan dari istilah yang dicari.
Main Flow	Use case ini dimulai pada saat user mengklik tombol terjemahan.
Postcondition	User mendapatkan terjemahan dari istilah yang dicari.
Precondition	User menginputkan istilah yang dicari kemudian mengklik tombol terjemahan.



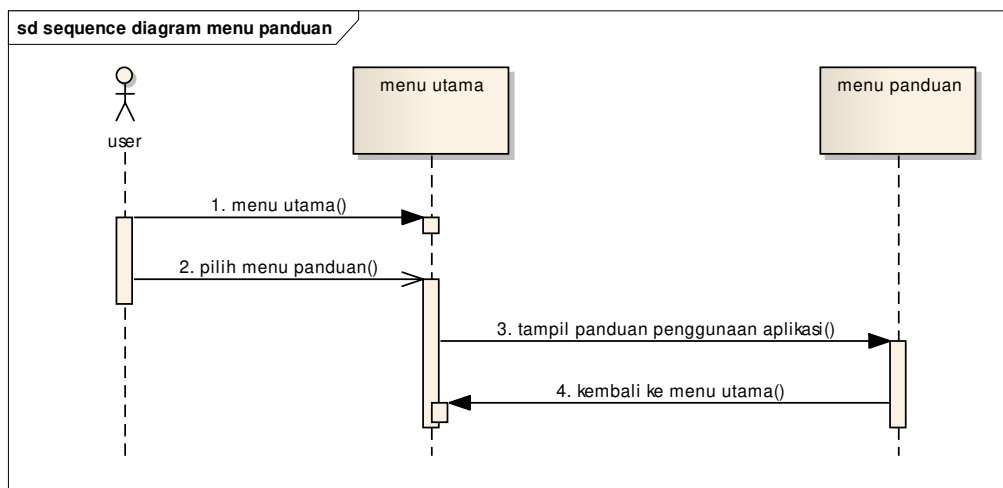
Gambar 8. Activity Diagram Aplikasi Kamus Linux



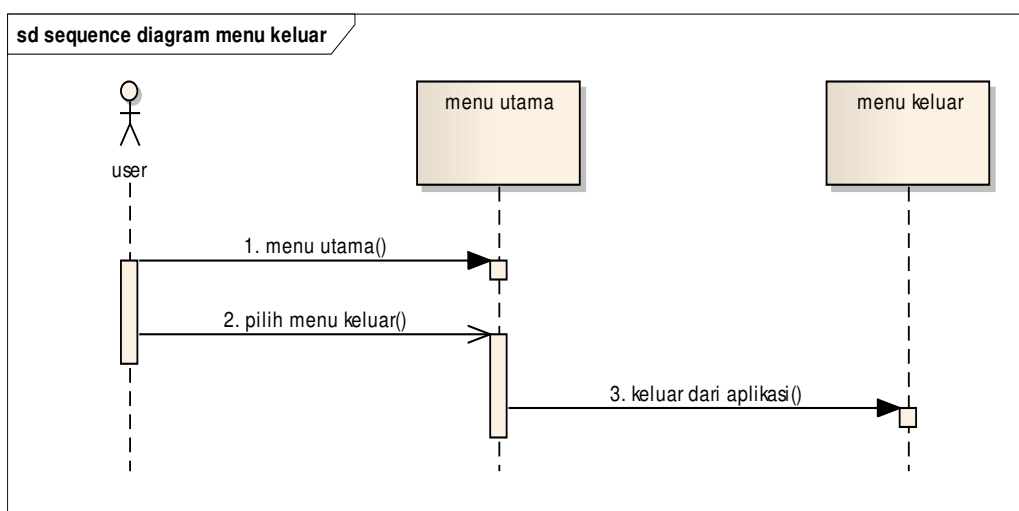
Gambar 9. Sequence Diagram Menu Materi



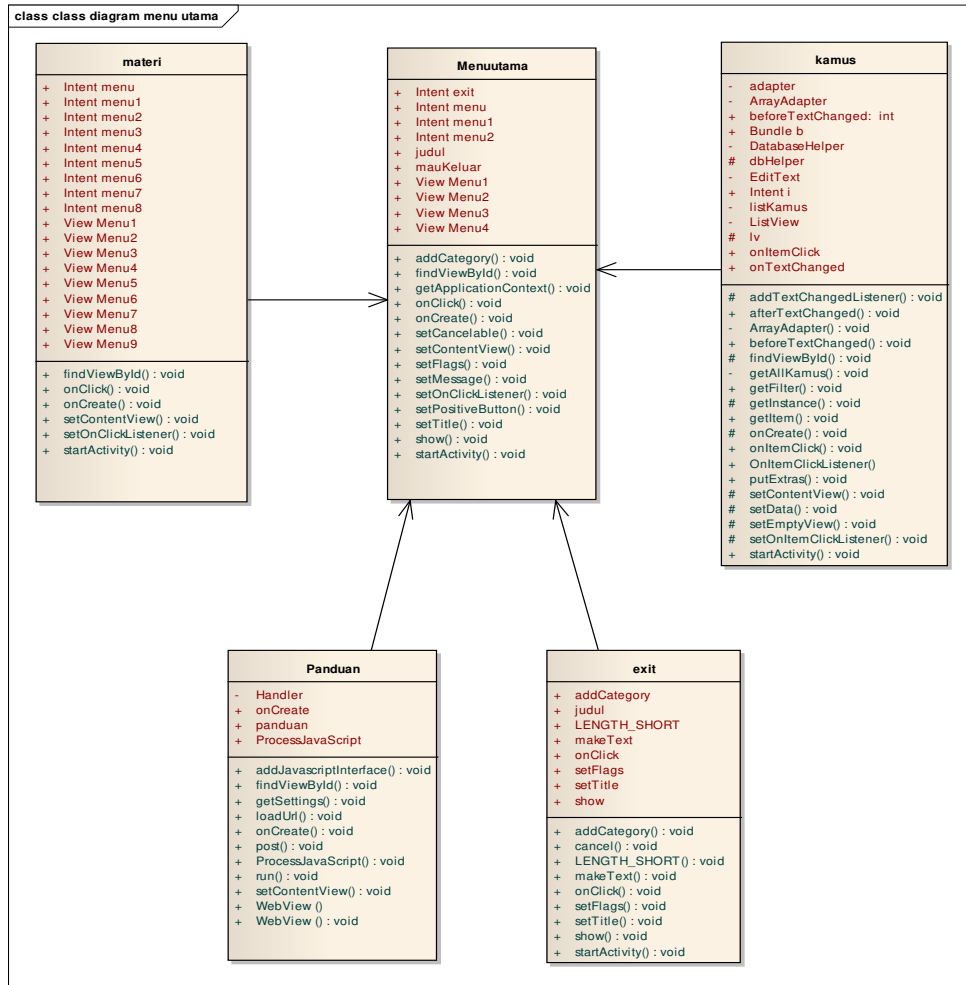
Gambar 10. *Sequence Diagram Menu Kamus*



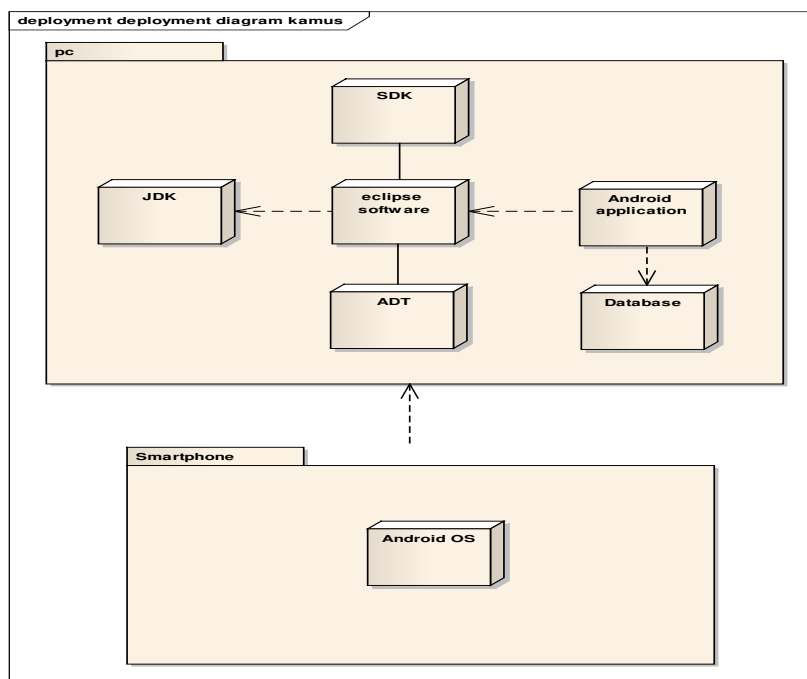
Gambar 11. *Sequence Diagram Menu Panduan*



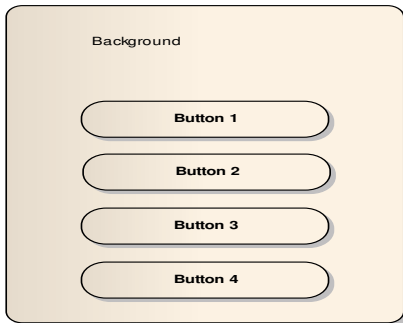
Gambar 12. *Sequence Diagram Menu Keluar*



Gambar 13. Class Diagram Menu Utama



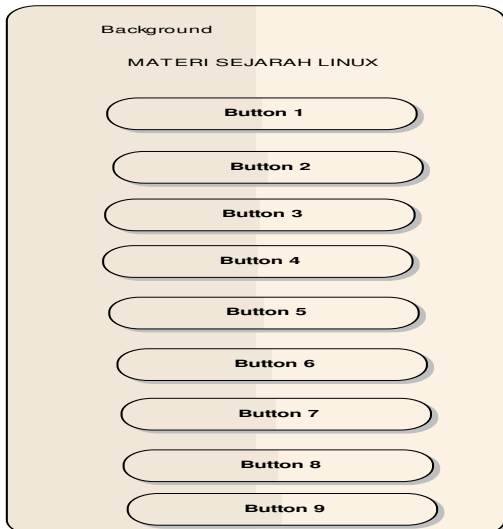
Gambar 14. Deployment Diagram Kamus



Gambar 15. *User Interface Menu Utama*



Gambar 19. *Interface Menu Keluar*



Gambar 16. *Interface Menu Materi*

C. Implementasi

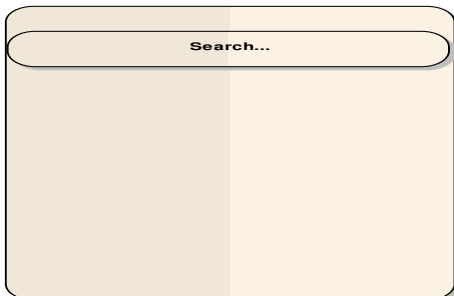
Implementasi dari rancangan yang telah dilakukan meliputi :

1. Menu utama :

Source code menu utama seperti dalam Gambar 20 dan tampilannya di Gambar 21.

```
Public class Memutama extends Activity
implements
OnClickListener{
Public void onCreate(Bundle savedInstanceState)
{super.onCreate(savedInstanceState);
setContentView(R.layout.memu);
```

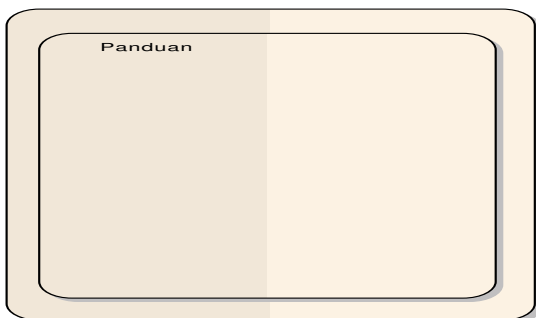
Gambar 20. Source Code Menu Utama



Gambar 17. *Interface Menu Kamus*



Gambar 21. Tampilan Menu Utama



Gambar 18. *Interface Menu Panduan*

2. Menu Materi

Source code menu materi seperti dalam Gambar 22 dan tampilannya di Gambar 23.

```

public class Materi extends Activity implements OnClickListener {
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.materi);
        View Menu1 = findViewById(R.id.button1);
        Menu1.setOnClickListener((OnClickListener) this);
        View Menu2 = findViewById(R.id.button2);
        Menu2.setOnClickListener((OnClickListener) this);
        View Menu3 = findViewById(R.id.button3);
        Menu3.setOnClickListener((OnClickListener) this);
        View Menu4 = findViewById(R.id.button4);
        Menu4.setOnClickListener((OnClickListener) this);
        View Menu5 = findViewById(R.id.button5);
        Menu5.setOnClickListener((OnClickListener) this);
        View Menu6 = findViewById(R.id.button6);
        Menu6.setOnClickListener((OnClickListener) this);
        View Menu7 = findViewById(R.id.button7);
        Menu7.setOnClickListener((OnClickListener) this);
        View Menu8 = findViewById(R.id.button8);
        Menu8.setOnClickListener((OnClickListener) this);
        View Menu9 = findViewById(R.id.button9);
        Menu9.setOnClickListener((OnClickListener) this);
    }
}

```

Gambar 22. Source Code Menu Materi

Untuk *interface* tampilan menu materi sebagai berikut :



Gambar 23. Tampilan Menu Materi

3. Tampilan Menu Kamus

Source code menu kamus seperti dalam Gambar 24 dan tampilannya di Gambar 25.

4. Tampilan Menu Panduan

Source code menu panduan seperti dalam Gambar 26 dan tampilannya di Gambar 27.

```

protected void onCreate(Bundle savedInstanceState){
    super.onCreate(savedInstanceState);
    setContentView(R.layout.kamus);
    lv = (ListView) findViewById(R.id.lv_data);
    lv.setEmptyView(findViewById(R.id.empty));
    search = (EditText) findViewById(R.id.search);
}

```

Gambar 24. Source Code Menu Kamus



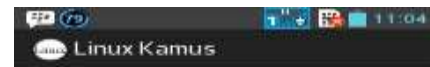
Gambar 25. Tampilan Menu Kamus

```

public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.panduan);
}

```

Gambar 26. Source Code Menu Panduan



Menu Panduan

PADA APLIKASI KAMUS LINUX INI TERDAPAT 3 MENU YAITU :

1. Menu Materi

Menu Materi berisi tentang jenis - jenis Sistem Operasi pada Linux beserta penjelasan sejarahnya.

2. Menu Kamus

Pada Menu Kamus pengguna dapat

Gambar 27. Tampilan Menu Panduan

5. Tampilan Menu Keluar

Source code menu panduan seperti dalam Gambar 28 dan tampilannya di Gambar 29.

```

public void onClick(DialogInterface arg0,int arg1){
Intent exit = new Intent (Intent.ACTION_MAIN);
exit.addCategory(Intent.CATEGORY_HOME);
exit.setFlags (Intent.FLAG_ACTIVITY_EXCLUDE_FROM_RECENTS
);
Toast.makeText (getApplicationContext(), "Terima Kasih
Telah
Menggunakan Aplikasi Ini",
Toast.LENGTH_SHORT).show();
startActivity(exit);    } })

```

Gambar 28. Source Code Menu Keluar



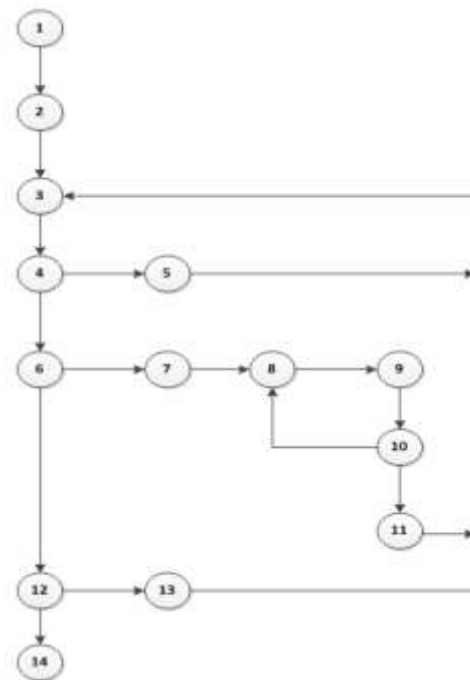
Gambar 29. Tampilan Menu Keluar

D. Pengujian

Setelah aplikasi berhasil dijalankan pada *emulator*, tahap selanjutnya yaitu menguji aplikasi *mobile*. Pengujian dilakukan dengan menggunakan *whitebox testing* seperti dalam Gambar 30.

Kompleksitas siklomatis yang dihasilkan adalah $V(G)=17 - 14+2$ maka $V(G)=5$. Hasil pengujian satu set jalur untuk

grafik alir dan pengujian *blackbox* disajikan pada Tabel 2 dan Tabel 3.



Gambar 30. Whitebox Testing

Tabel 2. Hasil Pengujian Jalur Grafik Alir

Anak Panah (E)	Simpul Lingkaran (N)
Jalur 1	1-2-3-4-5-14-3-14
Jalur 2	1-2-3-6-7-8-9-10-11-3-14
Jalur 3	1-2-3-6-7-8-9-10-8-9-10-11-3-14
Jalur 4	1-2-3-12-13-3-14
Jalur 5	1-2-3-14

Tabel 3. Pengujian *Blackbox*

Input / Event	Proses	Output / Next Stage	Hasil Pengujian
Tombol Materi	Menampilkan halaman materi	Tampil layout jenis-jenis linux	Sesuai
Tombol Kamus	Menampilkan halaman kamus	Tampil layout search	Sesuai
Tombol Panduan	Menampilkan halaman panduan	Tampil layout panduan	Sesuai
Tombol Keluar	Menampilkan halaman keluar	Tampil layout keluar	Sesuai

3.2 Pembahasan

Hasil pengujian diperoleh bahwa metode ESM dapat melakukan proses pencarian kamus linux dengan baik seperti halnya aplikasi android untuk kamus teknologi informasi. Metode ESM merupakan pencocokan string secara tepat dengan susunan karakter dalam string yang dicocokkan memiliki jumlah maupun urutan karakter dalam string yang sama. Proses pencocokan string dengan menggunakan ESM sebenarnya tidak berbeda dengan proses pencocokan biasa, bedanya hanya pada pemeriksaan kata. Hasil pengujian *blackbox* pada tabel 3 menyatakan bahwa aplikasi yang telah dirancang telah sesuai dengan yang telah dirumuskan di awal. Hal tersebut menunjukkan bahwa penggunaan metode ESM dalam pembangunan kamus linux menghasilkan kamus yang cukup baik seperti halnya yang telah dilakukan oleh Kacaribu(2014).

4. Kesimpulan

Berdasarkan pembahasan diatas maka penulis dapat mengambil beberapa kesimpulan sebagai berikut :

1. Penerapan metode *exact string matching* dalam proses pencarian kamus Linux layak digunakan.

2. Dengan adanya aplikasi kamus mobile Linux ini dapat membantu pengguna dalam menterjemahkan istilah Linux dan tanpa harus membawa kamus yang berbentuk buku.

Daftar Pustaka

- Kacaribu Andrico Dewanto. 2014. Perancangan Kamus Teknologi Informasi Berbasis Android Menggunakan Metode *Exact String Matching* (ESM). *Pelita Informatika Budi Dharma*, 22(1) : 157-160.
- Mulyana, Herlan. 2014. Aplikasi Mobile Kamus Istilah Komputer Berbasis Android. *Jurnal PIKSEL*, Volume 2 No. 1: 27-34
- Parno, dkk. 2011. Aplikasi *Mobile* Kamus Istilah Psikologi Berbasis Android 2.2. *Proceeding PESAT (Psikologi, Ekonomi, Sastra, Arsitektur & Sipil)* 4 : 121-128.
- Rosa dan Shalahuddin. 2013. *Rekayasa Perangkat Lunak*. Bandung : Informatika
- Sarumaha Nalainia. 2013. Aplikasi Kamus Bahasa Inggris-Indonesia-Nias Berbasis *Mobile* Android Dengan Algoritma *String Matching*. *Pelita Informatika Budi Dharma* 5(1) : 8-14.