

Pembuatan Aplikasi Pembelajaran Bahasa Inggris Pada Handphone dengan J2ME

Yusni Nyura

*Jurusan Teknologi Informasi, Politeknik Negeri Samarinda
yusnichristian@yahoo.com*

Abstrak

Dalam perkembangan teknologi informasi sekarang ini telah banyak ditemukan penggunaan aplikasi yang berbasis Java pada beberapa perangkat mobile device yang bersifat inovatif. Java adalah bahasa pemrograman berorientasi objek yang dapat digunakan untuk pengembangan aplikasi mandiri, aplikasi berbasis internet maupun intranet serta aplikasi untuk perangkat-perangkat cerdas yang dapat berkomunikasi lewat Intranet/jaringan komunikasi. Dalam Pembuatan Aplikasi Pembelajaran Bahasa Inggris ini menggunakan J2ME (Java 2 Micro Edition) karena dapat digunakan untuk menangani pemrograman di dalam perangkat-perangkat kecil dalam hal ini adalah mobile device seperti pager, smart card, cell phone, handheld PDA, Handphone, dan termasuk ke dalam kategori software support hardware. Dalam aplikasi ini yang ditampilkan adalah pembelajaran Tenses pada bahasa Inggris beserta menu latihan soal Tenses, yang dapat dijalankan pada emulator yang telah disediakan yaitu pada device / alat mobile device yang sesungguhnya. Aplikasi yang dibuat bersifat informasi (menampilkan data).

Kata kunci: Pembelajaran Bahasa Inggris, Handphone, J2ME

1. Pendahuluan

Java adalah bahasa pemrograman yang disusun oleh James Gosling yang dibantu oleh rekan-rekannya seperti Patrick Naughton, Chris Warth, Ed Frank dan Mike Sheridan di suatu perusahaan perangkat lunak yang bernama Sun Microsystem pada tahun 1991. Kemudian pada perkembangan selanjutnya adalah Java dibagi menjadi tiga buah yaitu: J2SE, J2EE dan J2ME. J2ME merupakan sebuah kombinasi yang terbentuk antara sekumpulan interface Java yang sering disebut dengan Java API dengan JVM yang di desain khusus untuk alat (dalam hal ini mobile device).

J2ME sangat cocok digunakan untuk menangani pemrograman di dalam perangkat-perangkat kecil dalam hal ini adalah mobile device seperti Handphone, PDA, Pager dan sejenisnya. Sehingga kita sebagai user dapat membuat program atau aplikasi yang dapat diletakkan di atas perangkat mobile device, mulai dari aplikasi yang berupa permainan, music atau mungkin aplikasi yang bersifat informasi, seperti aplikasi pembelajaran.

Jika dilihat dewasa ini, perkembangan bahasa Inggris di tengah-tengah masyarakat sangatlah mengagumkan. bahasa Inggris seolah-olah telah menjadi bahasa kedua setelah bahasa Indonesia. Hal tersebut tidak bisa dipungkiri lagi karena pentingnya penggunaan bahasa Inggris dalam segala bidang, seperti percakapan, pendidikan, perdagangan dan lain-lain. Oleh karena itu bahasa Inggris penting untuk dipelajari. Tenses adalah waktu. Dalam mempelajari bahasa Inggris dan

dalam mempraktekannya seringkali bagi mereka yang menggunakannya dalam percakapan bahasa Inggris kurang sempurna dan kesulitan dalam penggunaan tenses ini.

Adapun bahasa pemrograman yang digunakan dalam pembuatan program ini adalah J2ME dengan standar CLDC 1.1 dan MIDP 2.0. Kemunculan versi MIDP 2.0 diharapkan bisa meningkatkan tawaran-tawaran aplikasi yang ber-platform independent.

2. Dasar Teori

2.1. Pengertian Java

Java adalah sebuah bahasa pemrograman yang diciptakan oleh James Gosling, seorang developer dari Sun Microsystem pada tahun 1991. Selanjutnya Java dikembangkan Sun Microsystem dan banyak digunakan untuk menciptakan Executable Content yang dapat didistribusikan melalui network.

Java adalah bahasa pemrograman Object-Oriented dengan unsur-unsur seperti bahasa C++ dan bahasa-bahasa lainnya yang memiliki libraries yang cocok untuk lingkungan internet. Java dapat melakukan banyak hal dalam melakukan pemrograman, seperti membuat animasi halaman web, pemrograman Java untuk Ponsel dan aplikasi interaktif. Java juga dapat digunakan untuk handphone, internet dan lain-lain.

2.2. Karakteristik-karakteristik Java

a. Sederhana

Bahasa pemrograman Java menggunakan sintaks yang mirip dengan bahasa C++ namun sintaks pada Java telah banyak diperbaiki, terutama dengan menghilangkan pointer yang rumit dan multiple inheritance. Java juga menggunakan automatic memory allocation dan garbage collection.

b. Berorientasi Objek

Java merupakan bahasa pemrograman berorientasi objek yang memungkinkan program untuk dibuat secara modular dan digunakan kembali.

c. Terdistribusi

Java dibuat untuk memudahkan distribusi aplikasi dengan adanya networking libraries yang terintegrasi dalam Java.

d. Interpreted

Program Java dijalankan menggunakan program Interpreter, yaitu Java Virtual Machine (JVM). Hal ini menyebabkan source code Java yang telah dikompilasi menjadi bytecodes dapat dijalankan pada berbagai platform.

e. Robust

Java mempunyai reliabilitas yang tinggi. Kompiler pada Java mempunyai kemampuan mendeteksi error yang lebih baik dibandingkan bahasa pemrograman yang lain. Java mempunyai Runtime Exception Handling untuk membantu mengatasi error pada pemrograman.

f. Secure

Sebagai bahasa pemrograman aplikasi internet dan terdistribusi, Java memiliki beberapa mekanisme keamanan untuk menjaga agar aplikasi tidak digunakan untuk merusak sistem komputer yang menjalankan aplikasi tersebut.

g. Architecture Neutral

Program Java tidak bergantung pada platform dimana program akan dijalankan. Cukup dibuat satu program yang dapat dijalankan pada berbagai platform dengan Java Virtual Machine.

h. Portable

Source code maupun program Java dapat dengan mudah dibawa ke berbagai platform berbeda tanpa harus dikompilasi ulang.

i. Performance

Kinerja Java sering kali dikatakan kurang, namun kinerja Java dapat ditingkatkan menggunakan compiler Java lain seperti buatan Inprise, Microsoft maupun Symantec yang menggunakan Just In Time Compilers (JIT).

j. Multithreaded

Java dapat membuat suatu program yang mampu melakukan beberapa pekerjaan secara sekaligus dan simultan.

k. Dynamic

Java dapat didesain untuk dapat dijalankan pada lingkungan yang dinamis. Perubahan suatu class dengan menambahkan properties ataupun metode

dapat dilakukan tanpa mengganggu program yang menggunakan class tersebut.

2.3. CARA KERJA JAVA

Lingkungan pemrograman Java menggunakan kompiler sekaligus interpreter agar dapat berjalan pada platform yang berbeda. Kompiler Java akan mentransformasikan kode-kode dalam bahasa Java ke dalam suatu bytecode. Bytecode adalah sekumpulan perintah hasil kompilasi yang kemudian dapat dieksekusi melalui sebuah mesin komputer abstrak yang disebut dengan JVM. JVM juga sering dinamakan sebagai interpreter, karena sifatnya yang selalu menerjemahkan kode-kode yang tersimpan dalam bytecode dengan cara baris demi baris.

2.4. JAVA 1

Pada awal perilisannya versi Java masih disebut dengan JDK (Java Development Kit). Dalam JDK semua kebutuhan untuk pengembangan program dan eksekusi program masih tergabung jadi satu. Penamaan ini berlaku sampai Java 1.1. Namun sekarang, setelah Java 1.2, Sun Microsystem menamainya dengan JSDK (Java Software Development Kit) dalam hal ini kebutuhan untuk pengembangan software dipisahkan dengan kebutuhan eksekusi. Bagian software yang digunakan untuk kebutuhan eksekusi program disebut dengan JRE (Java Runtime Environment). Selanjutnya Java 1.2. disederhanakan penamaannya menjadi Java 1.2.

2.5. JAVA 2

Sun Microsystem telah mendefinisikan tiga buah edisi dari Java 2, yaitu sebagai berikut:

a. J2SE (Java 2 Standard Edition)

Digunakan untuk mengembangkan aplikasi-aplikasi desktop dan applet (aplikasi yang dapat dijalankan di dalam browser web).

b. J2EE (Java 2 Enterprise Edition)

Digunakan untuk mengembangkan aplikasi-aplikasi berskala besar (enterprise), yaitu dengan melakukan pembuatan aplikasi-aplikasi di sisi server dengan menggunakan EJBs (Enterprise JavaBeans), aplikasi web dengan menggunakan Servlet dan JSP (JavaServer Pages) dan teknologi lainnya seperti CORBA (Common Object Request Broker Architecture) dan XML (Extensible Markup Language)

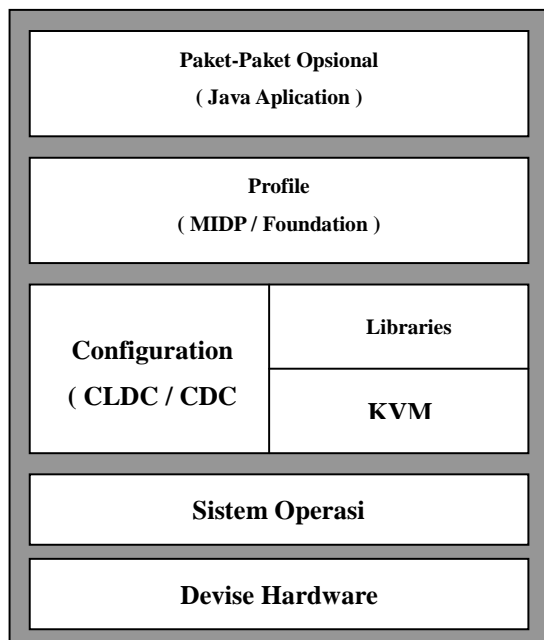
c. J2ME (Java 2 Micro Edition)

Digunakan untuk menangani pemrograman di dalam perangkat-perangkat kecil atau yang dapat berjalan di atas alat (dalam hal ini mobile device).

2.6. J2ME (Java2 Micro Edition)

Java2 Micro Edition atau yang biasa disebut J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak Java pada

barang elektronik beserta perangkat pendukungnya. Pada J2ME, jika perangkat lunak berfungsi baik pada sebuah perangkat maka belum tentu juga berfungsi baik pada perangkat yang lainnya. J2ME membawa Java ke dunia informasi, komunikasi dan perangkat komputasi selain perangkat komputer desktop yang biasanya lebih kecil dibandingkan perangkat komputer desktop. J2ME biasa digunakan pada telepon seluler, Pager, PDA dan sejenisnya. Teknologi J2ME juga memiliki beberapa keterbatasan terutama jika diaplikasikan pada ponsel. J2ME sangat tergantung pada perangkat atau device yang digunakan, bisa dari segi merek ponsel, maupun kemampuan ponsel dan dukungannya terhadap teknologi J2ME. Keterbatasan lainnya adalah pada ukuran aplikasi karena memori pada ponsel sangat terbatas.



Gambar 2.1. Arsitektur J2ME

2.6.1. Konfigurasi

Konfigurasi merupakan bagian yang berisi JVM dan beberapa library kelas lainnya. Terdapat dua buah konfigurasi yang disediakan oleh Sun Microsystems, yaitu CLDC (Connected Limited Device Configuration) dan CDC (Connected Device Configuration). CLDC adalah perangkat dasar dari J2ME, spesifikasi dasar yang berupa library dan API yang diimplementasikan pada J2ME, seperti yang digunakan pada telepon seluler, pager dan PDA. CDC adalah spesifikasi dari konfigurasi J2ME. CDC merupakan komunitas proses pada JAVA yang memiliki standarisasi. CDC terdiri dari virtual machine dan kumpulan library dasar untuk dipergunakan pada profile industri. Perbandingan antara CLDC dan CDC dapat dilihat pada tabel 2.1:

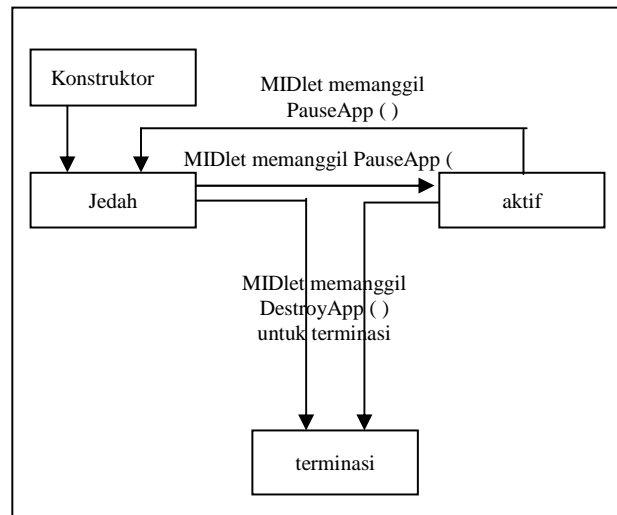
Tabel 2.1. Perbandingan antara CLDC dan CDC

CLDC	CDC
Mengimplementasikan sebagian dari J2SE	Mengimplementasikan seluruh fitur J2SE
JVM yang digunakan adalah KVM	JVM yang digunakan adalah CVM
Digunakan pada Handphone, Pager, PDA dengan memori terbatas (160-512 KB)	Digunakan pada internet tv, Nokia communicator, car TV dengan memori minimal 2 MB
Prosesor : 16/32 bit	Prosesor : 32 bit

2.7. MIDlet

MIDlet adalah aplikasi yang ditulis untuk MIDP. Aplikasi MIDlet adalah bagian dari kelas javax.microedition.midlet.MIDlet yang didefinisikan pada MIDP. MIDlet berupa kelas abstrak yang merupakan sub kelas dari bentuk dasar aplikasi sehingga antarmuka antar aplikasi J2ME dan aplikasi manajemen pada perangkat dapat terbentuk.

2.7.1. Daur Hidup MIDlet



Gambar 2.2. Alur Hidup MIDlet

MIDlet terdiri dari beberapa metode yang harus ada, yaitu Constructor, protected void startApp () throws MIDletStateChangeException, protected void pauseApp (), protected void destroyApp (boolean unconditional) throws MIDlet State Change Exception. Ketika MIDlet dijalankan maka akan diinisialisasi dengan kondisi *pause* dan dijalankan pauseApp (). Kondisi berikutnya adalah fungsi MIDlet dijalankan, yaitu pada startApp (). Metode yang ada tersebut diimplementasikan sebagai *protected*, hal ini dimaksudkan agar MIDlet lain tidak dapat memanggil metode tersebut. Pada saat pemakai keluar dari MIDlet, maka metode destroyApp () akan dijalankan sebelum MIDlet benar-benar tidak berjalan lagi. Metode notifyDestroyed () akan

dipanggil sebelum MIDlet benar-benar tidak berjalan lagi, `destroyApp()` akan memanggil `notifyDestroyed()` dan `notifyDestroyed()` akan memberitahu platform untuk menterminasi MIDlet dan membersihkan semua sumber daya yang mengacu pada MIDlet. Dalam implementasinya, MIDlet memiliki struktur direktori sebagai berikut:

1. **Src**, menyimpan source code untuk MIDlet dan kelas lain yang diperlukan
2. **Res**, menyimpan sumber daya yang dibutuhkan oleh MIDlet, seperti misalnya gambar icon
3. **Lib**, menyimpan file JAR atau ZIP yang berisi library tambahan yang dibutuhkan MIDlet
4. **Bin**, menyimpan file JAR, JAD dan file manifest yang berisi muatan komponen MIDlet

3. PERANCANGAN

3.1. Spesifikasi Sistem

Spesifikasi komputer yang digunakan adalah :

1. Prosesor Intel Pentium II 350 mhz
2. SDRAM 64 MB
3. Sistem Operasi Windows XP SP 2
4. Terinstall software J2ME
5. Ruang ksosong harddisk minimal 5 MB

3.2. Desain Antar Muka

Pada aplikasi ini akan menggunakan beberapa komponen dari J2ME seperti List, Ticker, Alert, Form, ChoiceGroup, StringItem dan Image. Flowchart di dalam aplikasi ini terbagi menjadi beberapa bagian seperti flowchart untuk menu utama, flowchart menu soal dan flowchart untuk tampilan Tenses yang ada dalam aplikasi ini.

3.3. Flowchart Aplikasi

Aplikasi terdiri dari 2 menu utama yaitu menu Pembelajaran Tenses dan Menu Soal Tenses. Pada saat aplikasi dibuka pertama kali / ketika MIDlet dijalankan pertama kali maka akan ada proses inisialisasi (AMS dijalankan). Kemudian proses berikutnya adalah fungsi MIDlet diaktifkan yaitu pada `startApp()`, fungsinya agar MIDlet dalam keadaan aktif (layar aktif). Setelah itu akan muncul Alert sebagai pesan pembuka. Sampai disini user dihadapkan pada kondisi untuk memilih apakah ingin masuk ke menu utama atau tidak. Jika Tidak maka aplikasi / MIDlet dalam keadaan Paused. Setelah itu saat user keluar dari MIDlet maka metode `destroyApp()` akan dijalankan sebelum MIDlet benar-benar tidak dapat berjalan lagi dan selesai. Jika Ya maka akan muncul layar splash (layar yang dimunculkan beberapa saat tepat sebelum menu utama muncul). Setelah layar splash muncul, tampilan berikutnya adalah tampilan menu

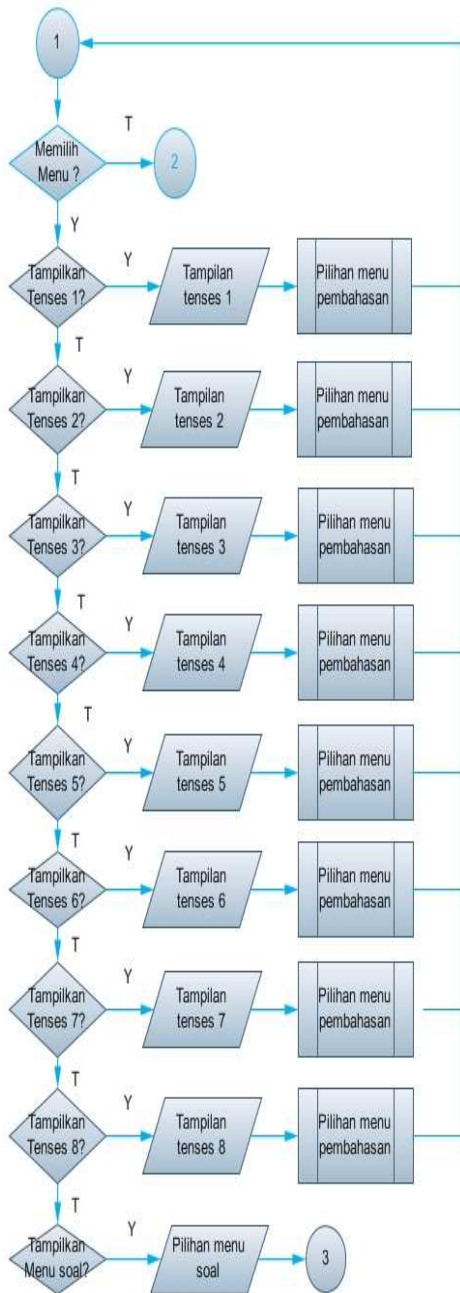
utama aplikasi. Sampai disini maka link flowchart akan berpindah ke nomor 1. Desain flowchart dapat dilihat pada gambar 3.1.:



Gambar 3.1. Flowchart aplikasi

3.3.1. Flowchart Menu Utama Aplikasi

Saat user masuk ke dalam tampilan menu utama aplikasi maka user akan dihadapkan pada sebuah kondisi untuk memilih menu tenses 1, 2, 3... atau menu soal. Jika user ingin keluar maka link flowchart akan berpindah ke nomor 2. Saat user ingin memilih menu soal tenses maka akan muncul tampilan pilihan menu soal dan sampai disini link flowchart berpindah ke nomor 3. (Gambar 3.2)

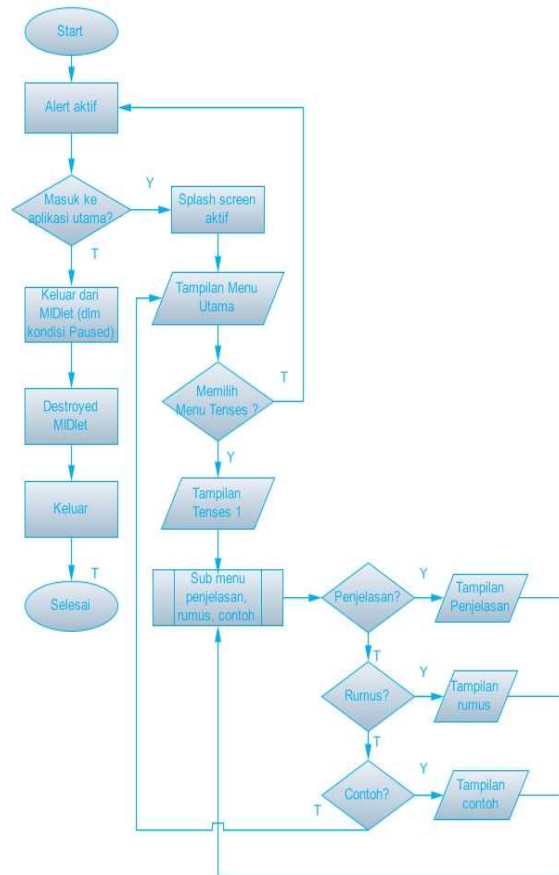


Gambar 3.2. Flowchart saat tampilan menu utama

3.3.2. Flowchart Tenses

Saat user masuk ke menu utama maka akan dihadapkan pada sebuah kondisi untuk melanjutkan ke tampilan berikutnya atau tidak. Jika Tidak maka aplikasi akan kembali ke tampilan awal yaitu di bagian Alert aktif. Jika user memilih untuk masuk ke salah satu menu tenses maka menu tenses akan ditampilkan dan sampai disini akan muncul 3 menu pembahasan yaitu penjelasan, rumus dan contoh. Di sini user dapat memilih menu pembahasan mana yang akan dilihat apakah menu penjelasan, rumus atau contoh. Jika user memilih menu pembahasan penjelasan maka akan muncul informasi berupa penjelasan dari tenses yang dipilih. Di layar

informasi ini user dapat kembali ke menu pembahasan sebelumnya untuk dapat memilih menu berikutnya. (Lihat gambar 3.3)



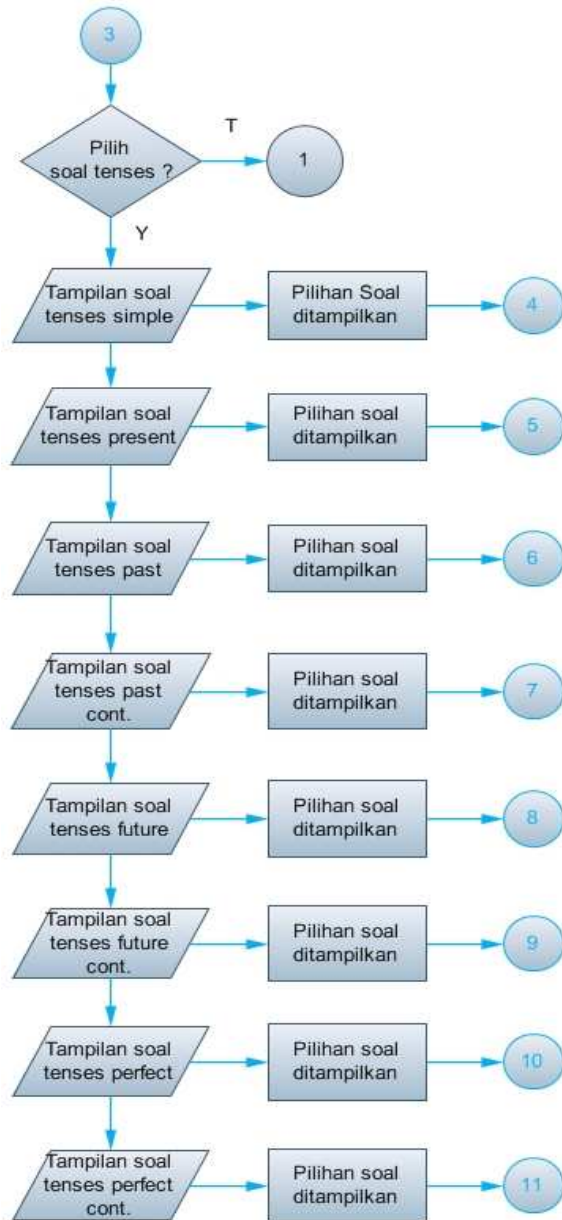
Gambar 3.3. Flowchart saat Tenses di pilih

3.3.3. Flowchart Menu Soal

Saat user memilih untuk masuk ke menu soal tenses maka akan muncul tampilan pilihan menu soal. Kemudian user akan dihadapkan pada sebuah kondisi untuk memilih soal tenses 1, 2, 3.....Jika Tidak maka program akan kembali ke link flowchart nomor 1. Misalnya user ingin memilih menu soal tenses simple maka akan muncul pilihan soal tenses simple. Saat muncul pilihan soal tenses simple maka link flowchart akan berpindah ke nomor 4. (Lihat Gambar 3.4)

3.3.4. FLOWCHART TAMPILAN SOAL TENSES

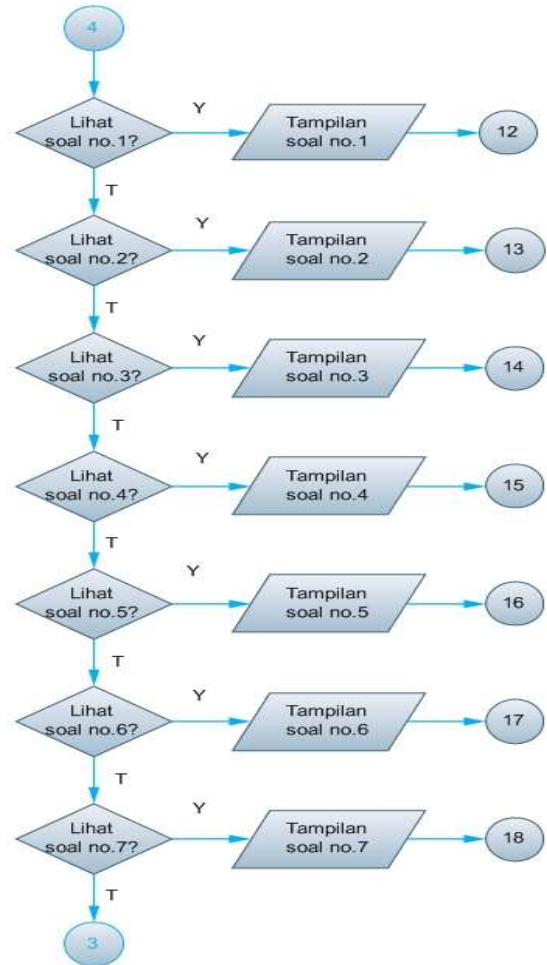
Dalam menu soal tenses terdiri dari 56 soal yang masing-masing tenses memiliki tujuh buah soal. Disini user dapat memilih soal nomor berapa yang ingin dilihat. Misalnya jika user ingin melihat soal nomor 1 maka user dapat memilih soal nomor 1 dan setelah itu akan muncul soal nomor 1 dan disini link flowchart akan berpindah ke nomor dua belas. Jika tidak user dapat memilih soal yang lainnya atau mungkin keluar dari tampilan soal tenses dan kembali ke link flowchart nomor 3. (Gambar 3.5)



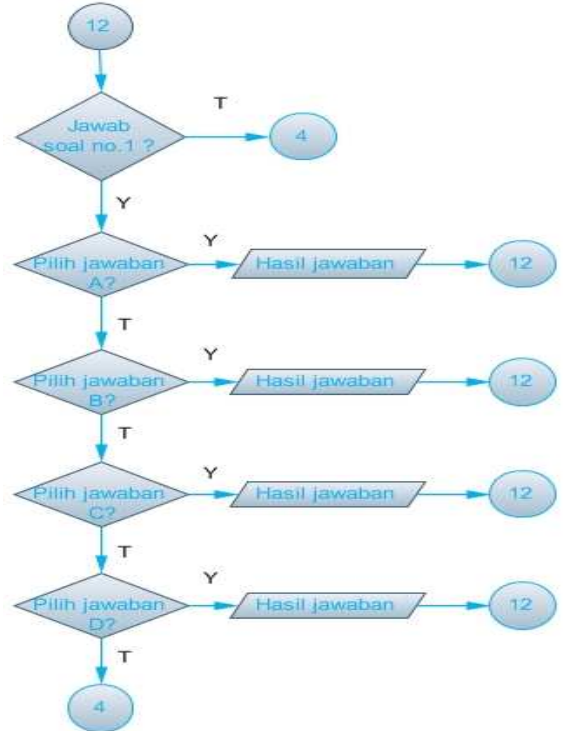
Gambar 3.4 Flowchart tampilan masuk ke menu soal

3.3.5. FLOWCHART SOAL

Saat user memilih soal nomor 1 maka akan muncul tampilan soalnya dan user juga akan dihadapkan pada kondisi yang dapat dipilih oleh user. Apakah user ingin keluar dari tampilan soal atau tidak. Jika TIDAK maka link flowchart akan berpindah ke nomor 4. Jika YA mak disini user harus menjawab dari pertanyaan yang muncul. Jika user menjawab pilihan a maka akan muncul layar informasi hasil jawaban yang menyatakan apakah jawaban yang dipilih benar atau salah. (Gambar.3.6). Flowchart saat menjawab soal



Gambar 3.5. Flowchart tampilan latihan soal tenses



Gambar.3.6. Flowchart saat menjawab soal

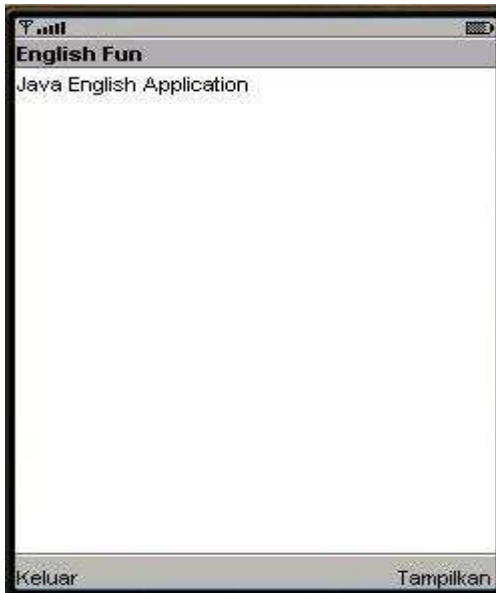
4. PEMBAHASAN DAN PENGUJIAN

4.1. Pembuatan Program

Kode program untuk J2ME ditulis dengan menggunakan editor teks, misalnya notepad, scite atau yang lainnya. Pertama, untuk membuat MIDlet maka langkah yang harus dilakukan adalah mengimpor paket javax.microedition.midlet, yaitu melalui perintah berikut: **import javax.microedition.midlet.*;** selanjutnya adalah, karena aplikasi yang dibuat merupakan aplikasi GUI (Graphical User Interface) maka yang harus dilakukan adalah mengimpor kelas-kelas yang terdapat pada paket javax.microedition.lcdui, dengan menuliskan perintah berikut: **import javax.microedition.lcdui.*;** kemudian paket berikutnya yang diimpor adalah import javax.microedition.io. Fungsinya berupa paket kerja fungsi koneksi secara umum: **import javax.microedition.io.*;** java.io merupakan paket yang menyediakan fungsi untuk input dan output sistem melalui aliran data, serialisasi dan sistem file, perintahnya adalah sebagai berikut: **import java.io.*;**

4.2. Tampilan Awal Program

Ketika user masuk / membuka aplikasi untuk pertama kali maka pada tampilan awal program akan ada Alert yang muncul. Alert adalah sejenis pesan yang tampil di layar yang menampilkan teks maupun gambar ke layar yang berguna untuk menginformasikan sesuatu ke user. Berikut adalah tampilan awal program:



Gambar 4.1. Tampilan awal program

Pada tampilan awal program (gambar 4.1.) terdapat dua buah command yaitu command keluar dan tampilkan. Dan untuk melanjutkan program maka user dapat memilih command tampilkan untuk masuk ke tampilan berikutnya, sehingga hasilnya

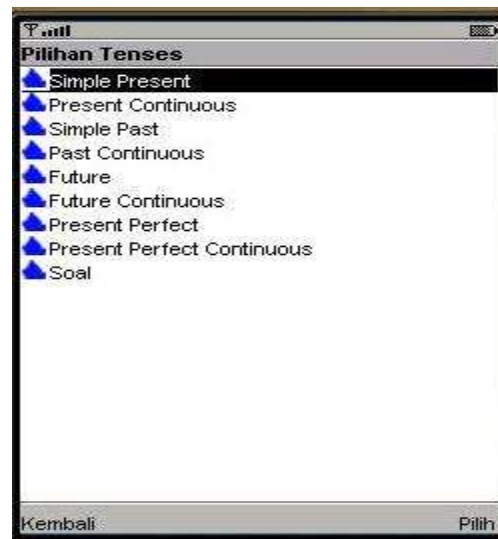
akan tampak seperti pada gambar 4.2. di bawah ini:



Gambar 4.2. Layar splash

4.2.1. Tampilan Menu Utama Program

Setelah layar splash muncul (selama 5 detik) maka akan tampil menu utama program yang berisikan pilihan menu tenses dan menu soal tenses.



Gambar 4.3. Tampilan Menu Utama

Berikut adalah listing program untuk menampilkan menu utama:

```

Image img;
try {
    img =
Image.createImage("/app.png") ;
} catch (Exception e) {
    img = null ;
}
listUtama = new List
("Pilihan Tenses", Choice.IMPLICIT);
listUtama.append ("Simple
Present", img);
listUtama.append ("Present
    
```

```
Continuous", img);
        listUtama.append ("Simple
Past", img);
        listUtama.append ("Past
Continuous", img);
        listUtama.append ("Future",
img);
        listUtama.append ("Future
Continuous", img);
        listUtama.append ("Present
Perfect", img);
        listUtama.append ("Present
Perfect Continuous", img);
        listUtama.append ("Soal",
img);
        listUtama.addCommand
(cmdPilih);
        listUtama.addCommand
(cmdKembali);
        listUtama.setCommandListener
(this);

        formAlert.setCommandListener(this);
        display.setCurrent(alert,
listUtama);
    }
}
```

Tampilan pada menu utama menggunakan fungsi komponen List yang akan menampilkan daftar item pilihan tenses. Terdapat tiga tipe List, yaitu tipe Exclusive, Implicit dan Multiple. Pada program ini digunakan List tipe Implicit karena List ini berupa List standar yang umumnya banyak digunakan. Seperti yang terdapat pada kode diatas maka tampilan menu utama dibuat dengan List tipe Implicit: `listUtama = new List ("Pilihan Tenses", Choice.IMPLICIT);`. Dan pada tampilan List tersebut juga terdapat proses untuk menambahkan gambar pada tampilan menu utama: `img = Image.createImage("/app.png")` ;. Pada kode: `listUtama.append ("Simple Present", img);` adalah proses penambahan menu pada List.

4.2.2. Tampilan Menu Tenses

Pada saat user berada pada tampilan menu utama (gambar 4.9) maka user dapat memilih salah satu dari tampilan menu tenses untuk ditampilkan. Misalnya user memilih untuk masuk ke tampilan menu Simple Present Tense, maka tampilannya adalah seperti pada gambar 4.4.

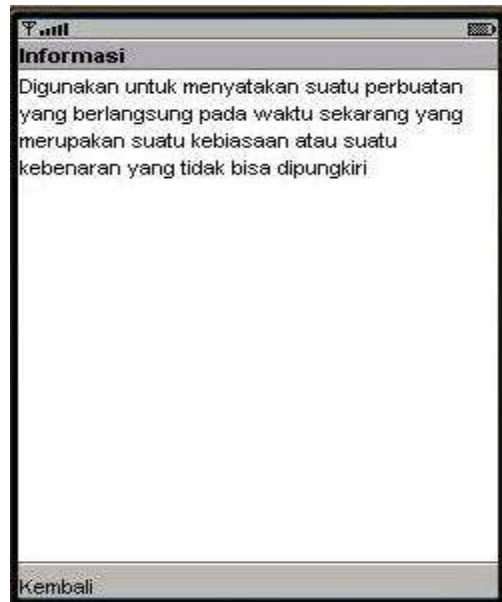


Gambar 4.4. Tampilan Menu Simple Present

Untuk tampilan menu pembahasan terdiri dari menu penjelasan, rumus dan contoh. Untuk tampilannya menggunakan fungsi komponen ChoiceGroup. Berikut adalah potongan listing programnya:

```
pilihan1 = new ChoiceGroup
("Pembahasan :",
Choice.EXCLUSIVE, menu1, null)
;
formMenu1.append (pilihan1) ;
```

Pada tampilan menu Simple Present Tenses (gambar 4.10) user dapat memilih menu pembahasan mana yang ingin dilihat apakah menu pembahasan penjelasan, rumus atau contoh. Misalnya user ingin melihat menu pembahasan penjelasan maka user dapat memilih command Pilih dan akan muncul layar informasi sebagai berikut:



Gambar 4.5. Layar informasi penjelasan

Berikut adalah potongan listing programnya:

```
private String teksTicker1 =
"Aplikasi Pembelajaran Tenses Bahasa
Inggris";

private String info1Penjelasan =
"Digunakan untuk menyatakan suatu
perbuatan yang berlangsung pada waktu
sekarang yang
merupakan suatu kebiasaan atau
suatu kebenaran yang tidak bisa
dipungkiri";

private String info1Rumus =
"(+) Subject + Verb 1 + Object
(-) Subject + do/does not +
Verb 1
(?) Do/Does + Subject +
Verb1";

private String info1Contoh =
```



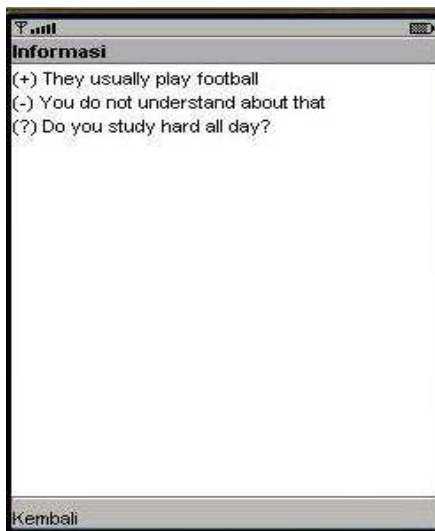
```

    "(+) They usually play football
        (-) You do not understand
about that
        (?) Do you study hard all
day ?";

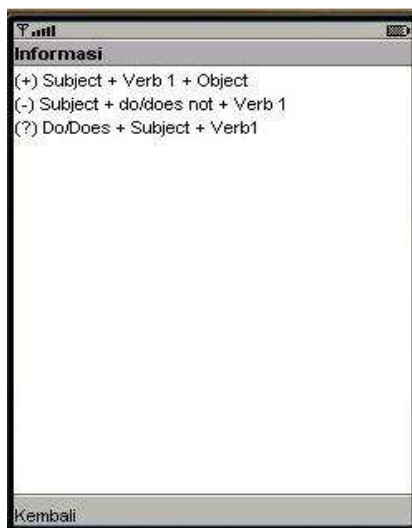
    private String[] menu1 = {
        "Penjelasan",
        "Rumus",
        "Contoh"
    } ;

```

Pada gambar diatas terdapat command kembali yang berfungsi agar user dapat kembali ke menu pembahasan sebelumnya untuk memilih menu pembahasan mana lagi yang ingin dilihat. Berikut adalah tampilan layar informasi untuk menu pembahasan rumus dan contoh:

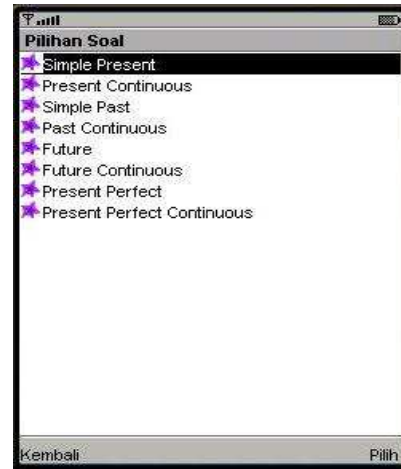


Gambar 4.6. Layar informasi contoh



Gambar 4.7. Layar informasi rumus

untuk masing-masing tenses sehingga total soal ada 56 soal. Berikut adalah tampilan menu soal tenses:



Gambar 4.8. Tampilan menu soal

4.3. IMPLEMENTASI SISTEM (PENGUJIAN)

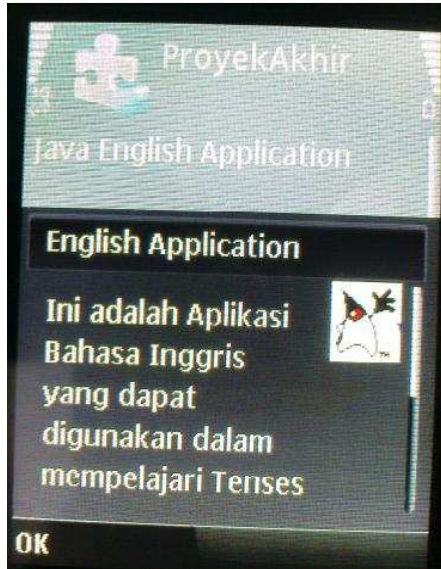
Aplikasi pembelajaran bahasa Inggris ini terdiri dari dua menu utama. Program aplikasi berukuran 343 KB pada komputer dan 40 KB pada handphone. Cara kerja aplikasi ini adalah ketika pertama kali menjalankan program, akan muncul Alert terlebih dahulu yang berfungsi untuk menampilkan informasi, dalam aplikasi ini informasinya adalah: Java English Application. Kemudian klik tombol pilihan “Tampilkan” sehingga akan muncul Splash screen, yaitu layar yang dimunculkan beberapa saat tepat sebelum aplikasi dimunculkan.



Gambar 4.9. Tampilan awal aplikasi

4.2.3. Tampilan Menu Soal Tenses

User dapat memilih untuk masuk ke menu soal tenses ketika berada pada tampilan menu utama (gambar 4.8). Menu soal terdiri dari 7 buah soal



Gambar 4.10. Tampilan Layar splash

Dari tampilan diatas, kemudian klik “OK” maka akan muncul menu utama aplikasi pembelajaran bahasa Inggris:



Gambar 4.11. Tampilan menu utama

4.4. INSTALASI MIDlet KE DALAM HANDPHONE

Terdapat beberapa cara untuk melakukan instalasi terhadap aplikasi (MIDlet) yang dibuat ke dalam handphone: infrared, bluetooth, maupun serial (kabel data). Berikut langkah yang perlu dilakukan:

1. Untuk membuat file JAR pilih menu **Project | Package | Create Package** dari

jendela Ktoolbar, maka file JAR akan terbentuk pada direktori bin. Untuk memindahkan MIDlet ke ponsel yang sesungguhnya dapat digunakan koneksi bluetooth, GPRS, ataupun inframerah. Dalam prakteknya, ada beberapa ponsel yang hanya membutuhkan file JAR saja, namun ada pula yang membutuhkan file JAD dan JAR dalam proses instalasi aplikasi pada ponsel.

2. Dari dalam handphone, pilih file JAR/JAD yang baru saja dicopy. Secara otomatis, AMS (Application Management Software) akan memunculkan menu “Install” pada layar handphone. Pilih menu tersebut untuk melalui proses instalasi. Sampai disini user sudah dapat menjalankan aplikasi tersebut dari handphone.

5. KESIMPULAN

Dari hasil pembuatan dan pengujian diambil kesimpulan bahwa :

1. Program dapat dijalankan pada emulator yang telah disediakan.
2. Program dapat dijalankan pada device / alat mobile device yang sesungguhnya.
3. Pembelajaran tenses bahasa Inggris pada pembuatan aplikasi ini dapat ditampilkan, dan bersifat informasi (menampilkan data)

6. DAFTAR PUSTAKA

1. Echols, J.M., Dan Shadily, H.(Eds), 1989, “*Kamus Inggris-Indonesia*”, PT. Gramedia Pustaka Umum, Jakarta
2. Raharjo Budi, Haryanto Imam dan Haryono Arif, 2007, “*Tuntunan Pemrograman Java Untuk Handphone*”, Informatika, Bandung.
3. Sholahudin, M., Rosa, A. S., 2006, “*Pemrograman J2ME Belajar Cepat Pemrograman Perangkat Telekomunikasi Mobile*”, Informatika, Bandung
4. Tim Penelitian Dan Pengembangan Wahana Komputer, 2005, “*Pengembangan Aplikasi Sistem Informasi Akademik Berbasis SMS Dengan Java*”, Salemba Infotek, Jakarta
5. Wicaksono Ady, 2002, “*Pemrograman Aplikasi Wireless dengan Java*”, PT. Elex Media Komputindo Kelompok Gramedia, Jakarta