

Makalah Seminar Tugas Akhir

APLIKASI ALQURAN DAN TERJEMAHAN PADA PERANGKAT GEGGAM MENGGUNAKAN J2ME

Wawan Setiawan^[1], Adian Fatchur Rochim, S.T,M.T^[2], Aghus Sofwan,S.T,M.T^[3]

Jurusan Teknik Elektro Fakultas Teknik Universitas Diponegoro

Jl.Prof.Soedharto, SH, Tembalang, Semarang 50275

Email : wawansetiawan83@yahoo.com

Abstrak- Zaman serba teknologi seperti saat ini, teknologi sudah digunakan dalam segala bidang, termasuk dalam bidang keagamaan. Seseorang selalu berusaha untuk meningkatkan kualitas ketakwaannya kepada Tuhan dengan beribadah. Salah satu ibadah dalam islam adalah membaca Alquran serta memahami isi dan kandungan dari alquran dengan memahami terjemahannya

Menggabungkan Alquran dan perangkat genggam seperti ponsel, Aplikasi Alquran dan Terjemahan dibuat menggunakan perangkat lunak JDK 1.5.5 dan J2ME Wireless Toolkit 2.2 serta Editor Teks untuk menuliskan programnya. Aplikasi ini mempermudah seseorang dalam membaca Alquran setiap saat. Terlebih lagi bila dilengkapi dengan terjemahannya, seseorang tidak hanya membaca tetapi juga mengerti dan memahami isi dari ayat-ayat Alquran.

Aplikasi ini berjalan pada Sony Ericsson K700i dan Nokia 6600 yang mendukung aplikasi Java. Ayat dan terjemahan ditampilkan dalam bentuk form. Interaksi user dengan perangkat genggam ditangani dengan mengimplementasikan interface CommandListener. Aplikasi ini mempermudah user dalam mempelajari Alquran dan terjemahannya.

Kata kunci : Alquran dan terjemahan, J2ME, perangkat genggam

I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang sangat pesat, memungkinkan kita untuk dapat berbagi informasi dengan cepat, mudah dan dapat diakses dari mana saja. Penggunaannya yang menyeluruh di segala bidang, baik bidang informasi itu sendiri juga berguna bagi bidang-bidang yang lainnya seperti agama, pendidikan, kesehatan, sosial, budaya, hiburan, bisnis atau pun pertahanan keamanan. Dalam kehidupan manusia, hubungan kita dengan Maha Pencipta sangatlah erat kaitannya. Hubungan diantara keduanya dapat diaplikasikan dalam bentuk ibadah.

Salah satu ibadah dalam agama islam adalah membaca Alquran dan terjemahannya. Alquran yang kita baca serta kita pahami isi yang terkandung dalam setiap ayat dengan mempelajari terjemahannya. Dengan kemajuan teknologi saat ini, dimungkinkan untuk dibuat aplikasi Alquran dan terjemahan yang akan membantu penggunaannya dalam memahami agama atau kaidah yang dapat kita gali dari alquran secara mendalam serta pemenuhan kebutuhan rohani bagi penggunaannya. Kita dapat membaca surat-surat yang ada di Alquran beserta terjemahannya. Selain itu, karena aplikasi Alquran dan terjemahan ini akan berjalan pada perangkat genggam, maka akan memudahkan untuk dibawa sehingga dapat digunakan kapanpun dibutuhkan.

Pada tugas akhir ini akan digunakan aplikasi perangkat lunak J2ME (Java 2 Micro Edition) untuk membuat Alquran dan terjemahan untuk dapat dijalankan dan digunakan pada perangkat genggam yang mendukung aplikasi Java.

1.2 Tujuan

Tujuan dari pembuatan Tugas Akhir ini adalah untuk membuat sebuah aplikasi Alquran dan terjemahan yang dapat dijalankan pada perangkat genggam yang mendukung aplikasi Java.

1.3 Pembatasan Masalah

Pada tugas akhir ini dibatasi pada masalah - masalah sebagai berikut :

1. Pembuatan aplikasi Alquran dan terjemahan dengan menggunakan bahasa pemrograman J2ME (Java 2 Micro Edition) yang akan berjalan pada perangkat genggam yang mendukung aplikasi Java.
2. Menampilkan surat 78 sampai surat 114 yang ada di Alquran beserta terjemahannya dalam bentuk lima ayat dan lima terjemahan.
3. Menampilkan suara Surat 78 An Naba' (Berita besar).

[1] Mahasiswa Teknik Elektro Undip

[2] Dosen Teknik Elektro Undip

II. Dasar Teori

2.1 Alquran

Alquran menurut bahasa berarti bacaan atau yang dibaca. Menurut istilah, Alquran adalah wahyu Allah SWT yang diturunkan kepada Nabi Muhammad melalui malaikat Jibril sebagai petunjuk bagi umat manusia. Alquran diturunkan untuk menjadi pegangan bagi mereka yang ingin mencapai kebahagiaan dunia dan akhirat. Alquran menggunakan bahasa Arab dan merupakan mu'izat bagi rasul. Sebagian besar ayat-ayat Alquran diturunkan di kota Mekah dan kota Madinah. Isi yang terkandung dalam Alquran terdapat 114 surat dan 30 juz.

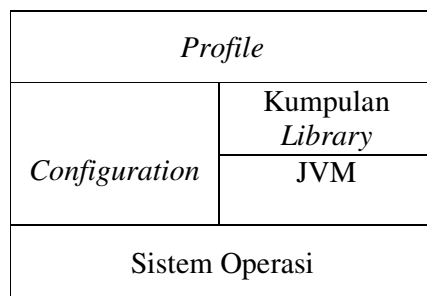
2.2 Terjemahan Alquran

Alquran menggunakan bahasa Arab dan merupakan mu'izat bagi rasul. Terjemahan Alquran merupakan kandungan yang ada dari alquran yang menggunakan bahasa Indonesia ataupun bahasa lain tanpa merubah isi dan kandungan yang ada didalamnya.

2.3 J2ME

Java 2 Micro Edition (J2ME) merupakan salah satu bagian dari teknologi Java yang dikembangkan untuk memungkinkan aplikasi – aplikasi Java bisa diimplementasikan pada peralatan *handheld* (perangkat genggam) yang tidak mampu mendukung secara penuh implementasi menggunakan J2SE.

Dalam J2ME dibagi menjadi dua bagian yang dikenal dengan istilah *configuration* dan *profile*. Arsitektur J2ME dapat dilihat pada gambar 2.3.



Gambar 2.3. Arsitektur J2ME

Configuration mendefinisikan lingkungan kerja J2ME *runtime*. Karena setiap perangkat genggam memiliki fitur – fitur yang berbeda – beda, *configuration* dirancang untuk menyediakan *library* standar yang mengimplementasikan fitur standar dari sebuah perangkat genggam. Sedangkan *profile* menyediakan implementasi – implementasi tambahan yang sangat spesifik dari sebuah perangkat genggam.

[1] Mahasiswa Teknik Elektro Undip

[2] Dosen Teknik Elektro Undip

Profil dan konfigurasi yang paling populer yang disediakan oleh Sun adalah *Mobile Information Device Profile* (MIDP) dan *Connected Limited Device Configuration* (CLDC).

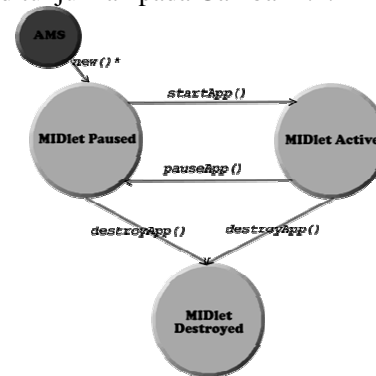
2.4 MIDlet

MIDlet adalah aplikasi yang ditulis untuk MIDP. Aplikasi MIDlet adalah bagian dari kelas `javax.microedition.midlet.MIDlet` yang didefinisikan pada MIDP. Kelas ini merupakan kelas utama dalam sebuah MIDlet, yaitu semua aplikasi MIDlet harus merupakan turunan dari kelas ini.

2.4.1 Siklus Hidup MIDlet

Perangkat *mobile* berinteraksi dengan MIDlet menggunakan perangkat lunaknya sendiri, yang disebut *Application Management Software* (AMS). AMS merupakan perangkat lunak atau program bawaan yang ada dalam perangkat genggam tempat dimana suatu MIDlet berjalan yang mengatur manajemen aplikasi MIDlet yang ada. AMS bertanggung jawab dalam menginisialisasi, memulai, menghentikan sementara, melanjutkan, dan memusnahkan MIDlet. Dalam melakukan manajemen MIDlet ini, suatu MIDlet dapat berada dalam tiga keadaan yang dikendalikan melalui metode kelas MIDlet, yaitu *active*, *paused* dan *destroyed*.

Jika digambarkan, perubahan status dalam daur hidup sebuah MIDlet adalah seperti yang ditunjukkan pada Gambar 2.4.



* - creates new MIDlet instance using the MIDlet's no args constructor

Gambar 2.4 Siklus hidup MIDlet

2.4.1.1 Paused

Status *paused* terjadi ketika MIDlet selesai diinisialisasi dan tidak melakukan aksi apapun.

2.4.1.2 *Active*

Status *active* terjadi ketika MIDlet sedang aktif / berjalan dengan normal, yaitu setelah memanggil fungsi MIDlet.startApp()

2.4.1.3 *Destroyed*

Status *destroyed* terjadi ketika MIDlet berhenti berjalan (identik dengan *exit*), sehingga seluruh sumber daya yang digunakan akan dibebaskan. Status ini terjadi ketika berhasil dilakukan pemanggilan fungsi MIDlet.destroyApp() atau MIDlet.notifyDestroyed().

2.4.2 Eksekusi sebuah MIDlet

Umumnya urutan eksekusi sebuah MIDlet adalah sebagai berikut:

- AMS menginisialisasi sebuah objek MIDlet, sehingga MIDlet berada pada status *Paused*.
- Ketika AMS memutuskan bahwa MIDlet sudah waktunya dijalankan, maka AMS memanggil fungsi MIDlet.startApp() sehingga MIDlet akan mulai aktif berfungsi.
- Ketika AMS memutuskan bahwa MIDlet harus dinonaktifkan sementara, maka AMS akan memanggil fungsi MIDlet.pauseApp() sehingga MIDlet akan berhenti bekerja.
- Ketika AMS memutuskan bahwa MIDlet harus dinonaktifkan selamanya, maka AMS akan memanggil fungsi MIDlet.destroyApp() sehingga alokasi memori yang digunakan oleh MIDlet yang bersangkutan akan dibersihkan.

2.5 Pemrograman GUI (*Graphical User Interface*) pada J2ME

Berbeda dengan sebuah komputer, perangkat genggam semacam telepon genggam, *Palm*, dan *organizer* memiliki berbagai keterbatasan, misalnya ukuran layar yang jauh lebih kecil dan masukan input yang tidak berupa *mouse*. Keterbatasan – keterbatasan ini menyebabkan perlunya ada teknik pemrograman GUI yang berbeda dengan teknik pemrograman yang umumnya digunakan pada aplikasi – aplikasi yang dijalankan pada sebuah komputer. Fungsi – fungsi untuk pemrograman GUI pada J2ME ditangani oleh MIDP.

2.5.1 Jenis Fungsi – Fungsi API pada MIDP untuk GUI

Fungsi – fungsi untuk pembuatan antarmuka berbasis *window* (GUI) yang disediakan oleh MIDP terbagi atas dua level, yaitu *highlevel* dan *lowlevel*. Semua fungsi – fungsi untuk antarmuka berbasis *window* ini ditangani oleh paket javax.microedition.lcdui.

Pada level pemrograman yang lebih tinggi (*highlevel*) yang merupakan level pemrograman yang dianjurkan, aplikasi MIDlet yang dibuat akan lebih portabel dengan banyak perangkat genggam yang berbeda. Kelas – kelas untuk manajemen pemrograman yang lebih tinggi akan diturunkan dari javax.microedition.lcdui.Screen. Aplikasi alquran dan terjemahan pada perangkat genggam ini menggunakan pemrograman *high level* ini untuk pembuatan antarmukanya.

Pada level pemrograman yang lebih rendah (*lowlevel*), fungsionalitas yang didapatkan akan lebih spesifik ke jenis perangkat genggam yang digunakan. Hal ini memungkinkan kurangnya portabilitas MIDlet yang dibuat.

2.5.2 Model Pemrograman GUI pada MIDP

Untuk menampilkan sesuatu pada layar, misalnya tulisan, form, dan sebagainya harus mengakses *display* dari perangkat genggam yang dapat diakses dengan kelas javax.microedition.lcdui.Display. Objek kedua adalah *screen*, yang berbeda dengan *display*. *Screen* digunakan untuk menangani komponen – komponen GUI, misalnya *window*, form, *dialogbox*, dan menangani masukan dari pengguna misalnya dari *keypad*. Objek *screen* ini direpresentasikan dengan kelas javax.microedition.lcdui.Screen. Pada setiap waktu hanya satu objek *screen* yang bisa ditampilkan oleh objek *Display*, untuk menampilkan digunakan fungsi setCurrent() pada objek *Display*.

2.5.3 Bekerja Dengan *Display*

Kelas *Display* merupakan kelas yang menyediakan fungsi – fungsi untuk manajemen layar pada perangkat genggam, menampilkan objek *screen*, dan menyediakan informasi tentang properti dari perangkat genggam yang digunakan. Akses ke layar dapat diacu dengan fungsi statik getDisplay() pada kelas Display.

```
Public static Display getDisplay
(MIDlet m)
```

Penggunaan fungsi ini umumnya dilakukan di dalam fungsi startApp(). Setelah mendapatkan objek *Display* dengan fungsi getDisplay(), maka digunakan fungsi setCurrent() untuk menentukan objek *screen* mana yang akan ditampilkan.

```
Public void setCurrent (Displayable
YourScreen)
```

2.5.4 Bekerja Dengan Screen

Objek *screen* menyediakan fungsionalitas untuk interaksi antara pengguna dengan perangkat genggam. Objek *screen* mempunyai empat jenis objek turunan, yaitu *TextBox*, *Alert*, *List*, dan *Form*. Kelas `javax.microedition.lcdui.Screen` merupakan kelas abstrak dengan dua karakteristik, yaitu objek *screen* bisa memiliki *title* dan *ticker*.

2.5.5 Bekerja Dengan Form

Dengan form dimungkinkan untuk menampilkan beberapa komponen GUI seperti daftar pilihan dan masukan teks (*textbox*) dalam satu layar. Form diimplementasikan oleh kelas `javax.microedition.lcdui.Form`. Form dapat menampung komponen – komponen yang disebut item dalam satu layar. Yang termasuk dalam item adalah *textfield*, *images*, *date fields*, *gauge*, dan *choice groups*. Item – item tersebut merupakan implementasi dari kelas turunan *Item* (`javax.microedition.lcdui.Item`). Ada dua konstruktor form yang umum digunakan, yaitu:

```
public Form(String title)
public Form(String title, Item[]
items)
```

2.5.6 Kelas Item (javax.microedition.lcdui.Item)

Komponen – komponen yang bisa diletakkan dalam sebuah form merupakan komponen – komponen yang memiliki kelas yang merupakan kelas turunan dari kelas abstrak `javax.microedition.lcdui.item`. Komponen – komponen tersebut adalah:

- *Image* dan *ImageItem*
Kelas ini menyediakan komponen grafik untuk manipulasi gambar.
- *StringItem*
Kelas ini menyediakan komponen teks string yang tidak bisa diedit oleh pengguna.

2.5.6.1 Image

Objek *Image* digunakan untuk meletakkan objek gambar yang bisa bersifat *immutable* atau *mutable*. Gambar yang bersifat *immutable* artinya

tidak bisa diubah – ubah oleh aplikasi, dan umumnya bersifat gambar yang diambil dari sebuah file gambar, URL, atau tempat lain. Sebaliknya, gambar yang bersifat *mutable* bisa diubah – ubah oleh aplikasi. Konstruktor dari kelas *Image* ini bersifat statik, sehingga tidak menggunakan operator new untuk membuat sebuah objek *Image* melainkan mengacu ke fungsi createImage() pada objek *Image* ini:

```
public static void
createImage(String name)
public static void
createImage(Image img)
public static void createImage(int
width, int height)
```

2.5.6.2 ImageItem

ImageItem merupakan objek untuk menampilkan gambar seperti objek *image*, hanya saja *ImageItem* dilengkapi dengan adanya fasilitas pengaturan layout atau struktur tempat gambar pada layer. Metode yang ada pada *ImageItem* adalah membuat sebuah *ImageItem* baru dengan gambar *img* dan *layout*.

```
ImageItem(String label, Image img,
int layout)
```

2.5.6.3 StringItem

StringItem menampilkan sebuah label statis dan sebuah pesan yang berupa teks. Metode yang ada pada *StringItem* adalah membuat sebuah *StringItem* nama *StringItem*nya dan teksnya.

```
StringItem(String label, string
teks)
```

2.6 Manajemen Event

Ketika terjadi interaksi antara pengguna dengan perangkat genggam semacam telepon genggam, *Palm*, dan *organizer*, maka akan dihasilkan suatu *event*. Misalnya ketika memilih suatu menu atau memasukkan teks ke dalam objek *TextField*, maka sistem akan memproduksi sebuah *event* yang kemudian aplikasi akan diberi tahu bahwa telah terjadi suatu *event* sebagai wujud interaksi dari pengguna sehingga aplikasi MIDlet bisa melakukan sesuatu berdasarkan *event* tersebut.

2.6.1 Penanganan Level Tinggi untuk Event

Untuk bisa menangani *event* pada level tinggi (*high level*), dapat dilakukan dengan mengimplementasikan *interface* *CommandListener* yang dideklarasikan pada paket `javax.microedition.lcdui`. Ketika mengimplementasikan *interface* *CommandListener*, maka harus

mengimplementasikan fungsi `commandAction()` dalam *interface* tersebut.

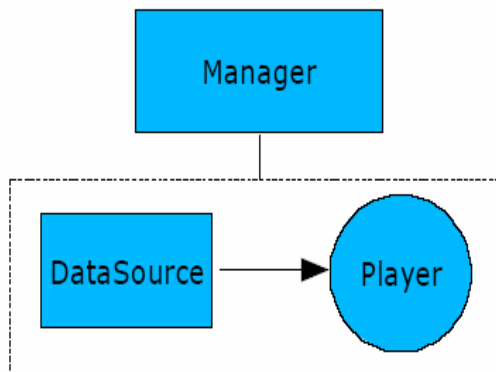
2.6.2 Command dan Interface CommandListener

Kelas `Command` (`javax.microedition.lcdui.Command`) merupakan kelas yang mengenkapsulasi informasi sebuah aksi. Dari informasi yang dibungkus oleh kelas ini, maka aplikasi bisa menentukan aksi apa yang dilakukan, tipenya bagaimana dan lain sebagainya. Terdapat tiga fungsi yang berkaitan dengan manajemen *event* dengan objek `Command`, yaitu:

```
public void addCommand(Command cmd)
Mengasosiasikan objek GUI yang bersangkutan
dengan objek Command cmd.
Public void removeCommand (Command
cmd)
Menghapuskan asosiasi objek GUI yang
bersangkutan dengan objek Command cmd.
public void setCommandListener
(CommandListener l)
Mengasosiasikan objek GUI yang bersangkutan
dengan interface CommandListener l.
```

2.7 Mobile Media API (MMAPI)

Mobile Media API (MMAPI) memberikan dukungan pada *MIDlet* dalam penggunaan media seperti audio dan video. Penggunaan sebuah media ditangani oleh dua object : *Manager*, *DataSource* dan *Player*.



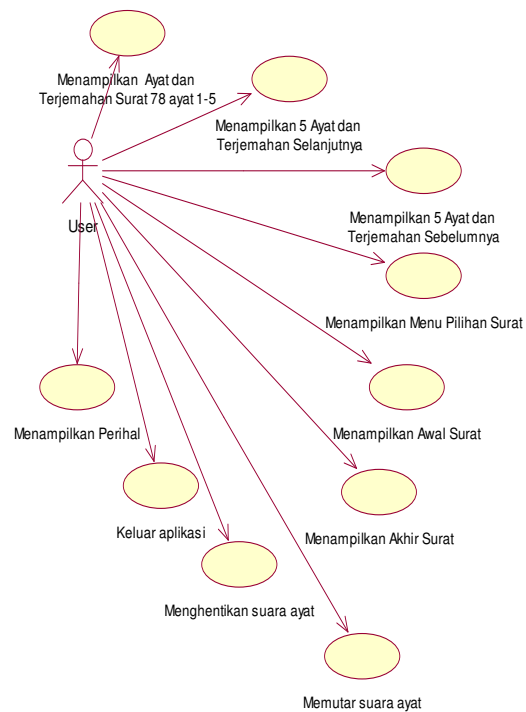
Gambar 2.5 Hirarki pada MMAPI

III. Perancangan Aplikasi Alquran dan Terjemahan pada Perangkat Genggam

3.1 Diagram Use-Case

Dalam pembuatan *diagram use-case* dari aplikasi Alquran dan terjemahan ini pertama kali ditentukan aktor yang akan menggunakan aplikasi (bersifat eksternal) dan aliran – aliran kerja (*workflow*) dalam aplikasi (bersifat internal).

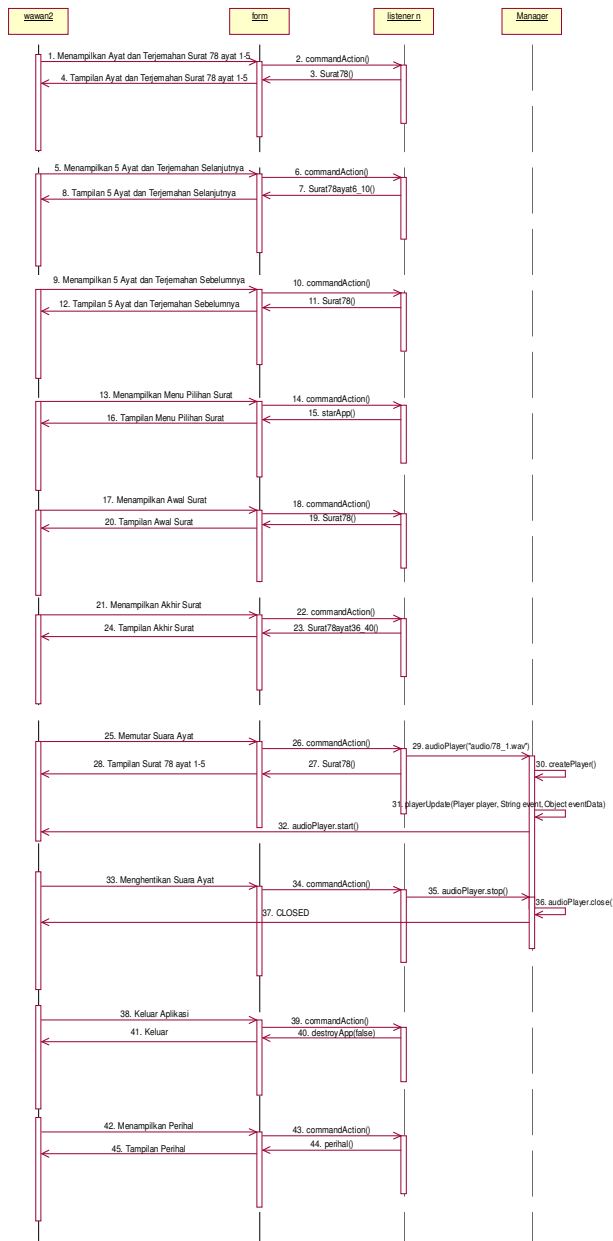
Diagram use-case dari aplikasi Alquran dan terjemahan ini ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Use-Case Aplikasi Alquran dan Terjemahan pada Perangkat Genggam.

3.2 Diagram Sequence

Diagram sequence memperlihatkan *event – event* yang terjadi sepanjang aliran kerja yang ada pada *use-case*, yang menyangkut objek apa yang dibutuhkan aliran, pesan mana yang dikirimkan suatu objek ke objek lainnya, serta bagaimana urutan pesan – pesan yang dikirimkan, yang berurutan sepanjang berjalannya waktu. *Diagram sequence* dari aplikasi Alquran dan terjemahan ini ditunjukkan pada Gambar 3.2.

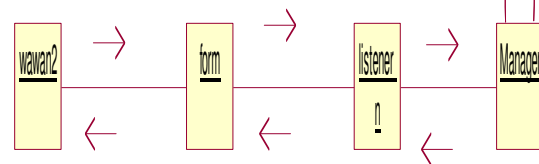


Gambar 3.2 Diagram sequence Aplikasi Alquran dan Terjemahan pada Perangkat Genggam.

3.4 Diagram Collaboration

Diagram collaboration memperlihatkan *event – event* yang terjadi sepanjang aliran kerja yang ada pada *use-case*, yang menyangkut objek apa yang dibutuhkan aliran, pesan mana yang dikirimkan suatu objek ke objek lainnya, serta bagaimana urutan pesan – pesan yang dikirimkan, dengan berfokus pada relasi – relasi yang terjadi antara objek yang satu dengan objek – objek yang lainnya. *Diagram collaboration* dari aplikasi alquran ini ditunjukkan pada Gambar 3.3.

- 1: Menampilkan Ayat dan Terjemahan Surat 78 ayat 1-5
- 2: commandAction()
- 3: Surat78()
- 4: Tampilan Ayat dan Terjemahan Surat 78 ayat 1-5
- 5: Menampilkan 5 Ayat dan Terjemahan Selanjutnya
- 6: commandAction()
- 7: Surat78ayat6_10()
- 8: Tampilan 5 Ayat dan Terjemahan Selanjutnya
- 9: Menampilkan 5 Ayat dan Terjemahan Sebelumnya
- 10: commandAction()
- 11: Surat78()
- 12: Tampilan 5 Ayat dan Terjemahan Sebelumnya
- 13: Menampilkan Menu Pilihan Surat
- 14: commandAction()
- 15: starApp()
- 16: Tampilan Menu Pilihan Surat
- 17: Menampilkan Awal Surat
- 18: commandAction()
- 19: Surat78()
- 20: Tampilan Awal Surat
- 21: Menampilkan Akhir Surat
- 22: commandAction()
- 23: Surat78ayat36_40()
- 24: Tampilan Akhir Surat
- 25: Memutar Suara Ayat
- 26: commandAction()
- 27: Surat78()
- 28: Tampilan Surat 78 ayat 1-5
- 29: audioPlayer("audio/78_1.wav")
- 30: createPlayer()
- 31: playerUpdate(Player player, String event, Object eventData)
- 32: audioPlayer.start()
- 33: Menghentikan Suara Ayat
- 34: commandAction()
- 35: audioPlayer.stop()
- 36: audioPlayer.close()
- 37: CLOSED
- 38: Keluar Aplikasi
- 39: commandAction()
- 40: destroyApp(false)
- 41: Keluar
- 42: Menampilkan Perihal
- 43: commandAction()
- 44: perihal()
- 45: Tampilan Perihal



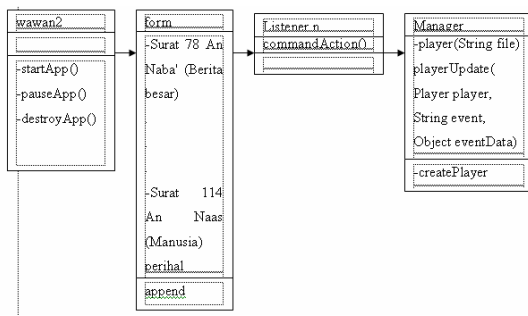
- 4: Tampilan Ayat dan Terjemahan Surat 78 ayat 1-5
- 3: Surat78()
- 30: createPlayer()
- 8: Tampilan 5 Ayat dan Terjemahan Selanjutnya
- 7: Surat78ayat6_10()
- 31: playerUpdate(Player player, String event, Object eventData)
- 12: Tampilan 5 Ayat dan Terjemahan Sebelumnya
- 11: Surat78()
- 36: audioPlayer.close()
- 16: Tampilan Menu Pilihan Surat
- 15: starApp()
- 20: Tampilan Awal Surat
- 19: Surat78()
- 24: Tampilan Akhir Surat
- 23: Surat78ayat36_40()
- 32: CLOSE
- 28: Tampilan Surat 78 ayat 1-5
- 27: Surat78()
- 37: audioPlayer.stop()
- 41: Keluar
- 40: destroyApp()
- 42: Menampilkan Perihal
- 44: perihal()

Gambar 3.3 Diagram collaboration Aplikasi Alquran dan Terjemahan pada Perangkat Genggam.

3.5 Class Diagram

Class diagram digunakan untuk menampilkan beberapa kelas yang ada dalam perangkat lunak Alquran dan terjemahan perangkat genggam serta memberikan gambaran / diagram statis tentang perangkat lunak dan relasi – relasi yang ada di dalamnya.

Class diagram dari aplikasi Alquran dan terjemahan ini ditunjukkan pada Gambar 3.4.



Gambar 3.4 Diagram Class Aplikasi Alquran dan Terjemahan pada Perangkat Genggam.

IV. Implementasi dan Pengujian Aplikasi Alquran dan Terjemahan pada Perangkat Genggam

Dalam implementasi aplikasi ini terlebih dahulu persiapkan ayat, dan terjemahan dari ayat tersebut. Ayat yang ditampilkan penulis berupa gambar, dan dalam gambar itu mengandung lima ayat. Begitu pula terjemahan yang ditampilkan mengandung lima terjemahan. Kemudian membuat gambar ayat yang disimpan dalam folder resource

Setelah pembuatan gambar ayat dan terjemahan selanjutnya dilakukan perancangan tampilan menu – menu yang digunakan dalam implementasi program ini. Menu – menu yang digunakan antara lain terdiri dari menu pilihan surat, menu Surat 78 An Naba' (Berita besar), Surat 79 An Naazi'at (Malaikat yang mencabut), sampai menu Surat 114 An Naas (Manusia), dan menu perihal. Tampilan dari masing – masing menu dapat dilihat pada penjelasan di bawah ini.

4.1.1 Menu Pilihan Surat

Tampilan dari menu Pilihan Surat adalah:



Gambar 4.1. Tampilan menu pilihan surat

Proses membuat list pilihan surat dan perihal dengan tipe exclusive ditunjukkan pada program berikut.

```
list = new List("PILIHAN SURAT", Choice.EXCLUSIVE);
```

Proses penambahan menu pada list dengan menambahkan gambar pada list ditunjukkan pada program berikut.

```
list.append("Surat 78 An Naba' (Berita besar)", img);
.
.
list.append("Surat 114 An Naas (Manusia)", img);
list.append("perihal ", img);
```

Proses menambahkan gambar pada list ditunjukkan pada program berikut.

```
try {
    img = Image.createImage("/qq.png");
} catch (Exception e) {
    img = null;
}
```

Membuat command pilih dan command exit ditunjukkan pada program berikut.

```
cmdPilih = new Command("Pilih", Command.EXIT, 2);
cmdKeluar = new Command("exit", Command.EXIT, 1);
```

Proses penambahan command Pilih dan command Exit pada list ditunjukkan pada program berikut.

```
list.addCommand(cmdKeluar);
list.addCommand(cmdPilih);
list.setCommandListener(this);
;
```

Proses list ditampilkan dilayar ditunjukkan pada program berikut .

```
display.setCurrent(list);
);
```

Program untuk menentukan pilihan yang dipilih dengan menggunakan command pilih adalah:

```
else if(c==cmdPilih) {
    int indeks = list.getSelectedIndex();
};
```

```

if
(indeks==0)
{ surat78();}
.
.
.
else if
(indeks==36)
{
surat114();}
else if
(indeks==37)
{ perihal();}
}

```

Jika memilih command exit maka akan keluar dari program. Program untuk keluar dari program adalah:

```

if (c == cmdKeluar) {
destroyApp(false);
}

```

4.1.2 Menu Surat 78 An Naba' (Berita besar)

Menu Surat 78 An Naba' (Berita besar) digunakan untuk menampilkan ayat 1-5 pada surat 78 beserta terjemahannya, seperti ditunjukkan gambar 4.2.



Gambar 4.2. Tampilan menu Surat 78 An Naba' (Berita besar)

Membuat form dengan nama Surat 78 An Naba' (Berita besar) ditunjukkan pada program berikut.

```

Form = new Form("surat 78
An Naba' (Berita besar)");

```

Membuat command back, next, awal, akhir, indeks, play, stop dan exit ditunjukkan pada program berikut.

```

cmdKembali = new
Command("back", Command.EXIT, 1);
cmdNext = new
Command("next", Command.EXIT, 2);
cmd1 = new
Command("awal", Command.EXIT, 3);
cmd2 = new
Command("akhir", Command.EXIT, 3);

```

```

cmd3 = new
Command("indeks", Command.EXIT, 2);
cmd6 = new
Command("exit", Command.EXIT, 3);
cmd7 = new
Command("play", Command.EXIT, 3);
cmd8 = new
Command("stop", Command.EXIT, 3);

```

Membuat instance dari kelas StringItem ditunjukkan pada program berikut.

```

si = new
StringItem("Artinya:", arti,
Item.PLAIN);
si.setFont(Font.getFont
(Font.FACE_MONOSPACE,
Font.STYLE_ITALIC |
Font.STYLE_UNDERLINED,
Font.SIZE_MEDIUM));
Membuat instance dari kelas
ImageItem ditunjukkan pada program
berikut

```

```

ii = new
ImageItem("Surat 78:1-
5 ",
imageItem.LAYOUT_NEWLIN
E_AFTER, null);

```

Proses membuat image dengan nama 78_1.png pada direktori res ditunjukkan pada program berikut .

```

try {
image =
Image.createImage("/78_1.png");
} catch (IOException ioe) {
ioe.printStackTrace();
}

```

Proses mengakses file dengan nama pada direktori res sebagai input stream ditunjukkan pada program berikut

```

InputStream is =
getClass().getResourceA
sStream("78_1.txt");
try{
StringBuffer sb = new
StringBuffer();
int chr,
i=0;
while
((chr=is.read()) !=-1)
sb.append((char) chr);
arti=sb.toString();
} catch (Exception e){
}

```


Menampilkan ayat 1-5 Surat 78 An Naba' (Berita besar) dan terjemahan pada form Surat 78 An Naba' (Berita besar)

```
form.append(ii);
form.append(si);
```

Menambahkan command back, next, indeks, awal, akhir, play, stop dan exit pada form Surat 78 An Naba' (Berita besar).

```
form.addCommand(cmdKembali);

form.addCommand(cmdNext);
    form.addCommand(cmd1);
    form.addCommand(cmd2);
    form.addCommand(cmd3);
    form.addCommand(cmd7);
```

```
form.addCommand(cmdexit);
```

Menampilkan semua objek yang ada form ditunjukkan pada program berikut.

```
display.setCurrent(form);
```

Pembuatan objek Player bergantung pada file musik yang dimainkan. Pada aplikasi ini jenis file musik yang dimainkan adalah file wave. Karena class wawan2 adalah implementasi dari antarmuka (interface) PlayerListener, oleh karenanya metode public void playerUpdate(Player player, String event, Object eventData) harus diimplementasikan (harus ada).

```
public void player(String
file) throws Exception{
    audioPlayer=Manager.createPl
ayer(getClass().getResourceA
sStream(file), "audio/x-
wav");
    audioPlayer.addPlayerListene
r(this);
    audioPlayer.setLoopCount(-
1);
    audioPlayer.prefetch();
    audioPlayer.realize();
    audioPlayer.start();
}
public void
playerUpdate(Player player,
String event, Object
eventData) {
    if(event.equals(PlayerListen
er.STARTED)&&new
Long(0L).equals((Long)eventD
ata)) {
        }else
if(event.equals(PlayerListener.CLOSED
)) {
        }
}
```

Membuat instance listener78_1call yang merupakan interface CommandListener yang digunakan untuk menangkap event yang dibuat oleh command di class listener 78_1().

```
listener78_1call=(CommandListener)n
ewlistener78_1();
```

```
form.setCommandListener(liste
ner78_1call);
```

Buat inner class listener 78_1() yang merupakan event yang dibuat oleh command.

```
class
listener78_1 ()
```

Pada bagian kiri bawah layar terdapat command back yang digunakan untuk kembali ke menu pilihan surat bila berada menu Surat 78 An Naba' (Berita besar) dan pindah ke lima ayat sebelumnya selain dari menu Surat 78 An Naba' (Berita besar)

```
if(c == cmdKembali) {
display.setCurrent(list);
    if (c == cmdKembali) {
```

```
surat78ayat6_10();}
```

Command next akan menampilkan lima ayat sesudahnya yaitu Surat 78 An Naba' (Berita besar) ayat 6-10

```
else if(c==cmdNext) {
```

```
surat78ayat6_10();}
```

Command awal akan menampilkan ayat pertama dari Surat 78 An Naba' (Berita besar)

```
else if(c==cmd1) {
    surat78(); }
```

Command akhir akan menampilkan ayat terakhir dari Surat 78 An Naba' (Berita besar) ayat 6-10

```
else if(c==cmd2) {
```

```
surat78ayat36_40();}
```

Command play akan menampilkan suara lima ayat yaitu Surat 78 An Naba' (Berita besar) ayat 1-5

```
else if(c==cmd7)
```

```
{
    form.addCom
mand(cmd8);
    try{
```

```
player("audio/78_1.wav"
```

```
);
```

```

    }catch
(Exception e) {
    e.printStackTrace();
}
}

```

Command stop akan menghentikan suara lima ayat yaitu Surat 78 An Naba' (Berita besar) ayat 1-5

```

else if(c==cmd8)
{
try{
audioPlayer.stop();

if(audioPlayer != null) {
audioPlayer.close();
}
}catch
(Exception e) {
}

form.removeCommand(cmd8);
}

```

Jika memilih command exit maka akan keluar dari program. Program untuk keluar dari program adalah:

```

if (c == cmdexit) {
destroyApp(false);
}

```

Command indeks akan menampilkan menu pilihan surat.

```

else {
display.setCurrent(list);}
}
}

```

4.1.3 Menu Perihal

Tampilan dari menu perihal dapat dilihat gambar 4.3 dibawah ini.



Gambar 4.3. Tampilan menu perihal.

Membuat form dengan nama perihal ditunjukkan pada program berikut.

```

Form = new
Form("perihal");

```

Membuat command back ditunjukkan pada program berikut.

```

cmdKembali = new
Command("back", Command.EXIT, 1);

```

Membuat instance dari kelas StringItem ditunjukkan pada program berikut.

```

si = new
StringItem("Artinya:", arti,
Item.PLAIN);
si.setFont(Font.getFont(
(Font.FACE_MONOSPACE,
Font.STYLE_ITALIC |
Font.STYLE_UNDERLINED,
Font.SIZE_MEDIUM));

```

Proses mengakses file dengan nama pada direktori res sebagai input stream ditunjukkan pada program berikut

```

InputStreamis=getClass(
).getResourceAsStream("
1.txt");
try{
StringBuffer sb =
new StringBuffer();
int chr, i=0;
while
((chr=is.read()) !=-1)
sb.append((char)
chr);

arti=sb.toString();
}catch (Exception
e){
}

```

Menampilkan perihal ditunjukkan pada program berikut.

```

form.append(si);

```

Menambahkan command back pada form perihal ditunjukkan pada program berikut .

```

form.addCommand(cmdKembali);

```

Menampilkan semua objek yang ada form ditunjukkan pada program berikut.

```

display.setCurrent(form);

```

Membuat instance listener12call yang merupakan interface CommandListener yang digunakan untuk menangkap event yang dibuat oleh command di class listener12.

```

        listener12call=(CommandListener
) new listener12();
        form.setCommandListener(listene
r12call);

```

Buat inner class listener12() yang merupakan event yang dibuat oleh command.

```
class listener12()
```

Pada bagian kiri bawah layar terdapat command back yang digunakan untuk kembali ke menu pilihan surat.

```
if (c == cmdKembali) {
```

```
display.setCurrent(list);}
```

V. Penutup

5.1 Kesimpulan

Dari pembahasan dan penulisan Tugas Akhir yang berjudul Aplikasi Alquran dan Terjemahan Pada Perangkat Genggam, dapat diambil beberapa kesimpulan dan saran.

1. Aplikasi Alquran dan Terjemahan ini berjalan pada Nokia 6600 dan Sony Ericsson K700i yang mendukung aplikasi java.
2. Aplikasi ini menampilkan ayat dan terjemahan dalam bentuk form.
3. Interaksi user dengan perangkat genggam dapat ditangani dengan mengimplementasikan interface CommandListener.
4. Aplikasi Alquran dan Terjemahan ini maka dapat digunakan sebagai alat untuk mempermudah membaca Alquran dan memahami kandungan dari Alquran dengan kita membaca terjemahannya pada perangkat genggam yang mendukung aplikasi java.

5.2 Saran

Adapaun saran untuk pengembangan selanjutnya adalah:

1. Pada aplikasi Alquran dan Terjemahan ini menampilkan ayat-ayat surat 78 sampai surat 114 dalam bentuk gambar dan terjemahannya dalam bentuk teks, karena ukuran file gambar yang besar, maka untuk pengembangan selanjutnya digunakan dengan bentuk teks untuk menampilkan ayatnya.

DAFTAR PUSTAKA

- [1] Nugroho, A., *Rational Rose untuk Pemodelan Berorientasi Objek*, Informatika, Bandung, 2005.
- [2] Shalahuddin, S dan Rosa A.S., *Pemrograman J2ME Belajar Cepat Pemrograman Perangkat*

Telekomunikasi Mobile., Informatika, Bandung, 2006.

- [3] Suyoto, *Membuat Sendiri Aplikasi Ponsel*, Gava Media, Yogyakarta, 2005.
- [4] Syafi'I, A., *Pendidikan Agama Islam 2*, Yudhistira, Jakarta, 1994.
- [5] Wicaksono, A., *Pemrograman Aplikasi Wireless dengan Java*, PT Elex Media Komputindo, Jakarta, 2002.
- [6] <http://java.sun.com/j2me/>
- [7] <http://pocketquran.com/>