

**APLIKASI PEMBELAJARAN BAHASA JEPANG
BERBASIS J2ME**

TUGAS AKHIR

Ditulis dan diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Ahli Madya Ilmu Komputer



Oleh :

Setiana Maya Habsari

M3206058

**PROGRAM DIPLOMA III ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SEBELAS MARET
SURAKARTA**

2009

HALAMAN PERSETUJUAN

**APLIKASI PEMBELAJARAN BAHASA JEPANG
BERBASIS J2ME**

Disusun Oleh

SETIANA MAYA HABSARI
NIM. M3206058

Tugas Akhir ini telah disetujui untuk dipertahankan
Di hadapan dewan penguji
pada tanggal _____

Pembimbing Utama

Irwan Susanto, S.Si, DEA
NIP. 19710511 199512 1 001

HALAMAN PENGESAHAN
APLIKASI PEMBELAJARAN BAHASA JEPANG
BERBASIS J2ME

Disusun Oleh

SETIANA MAYA HABSARI
NIM. M3206058

Di bimbing oleh
Pembimbing Utama

Irwan Susanto, S.Si, DEA
NIP. 19710511 199512 1 001

Tugas Akhir ini telah diterima dan disahkan
oleh dewan penguji Tugas Akhir
Program Diploma III Ilmu Komputer
pada hari _____ tanggal _____

Dewan Penguji

- | | | | |
|--------------|---|---|---|
| 1. Penguji 1 | Irwan Susanto, S.Si, DEA
NIP. 19710511 199512 1 001 | (|) |
| 2. Penguji 2 | Ristu Saptono, S.Si, M.T.
NIP. 19790210 200212 1 001 | (|) |
| 3. Penguji 3 | Bowo Winarno, M.Kom
NIP. 19810430 200812 1 001 | (|) |

Disahkan Oleh

Dekan
Fakultas MIPA UNS

Ketua Program Studi
DIII Ilmu Komputer UNS

Prof. Drs. Sutarno, M.Sc. PhD
NIP. 19600809 198612 1 001

Drs. Y.S. Palgunadi, M.Sc
NIP. 19560407 198303 1 004

ABSTRACT

Setiana Maya Habsari, 2009, **THE APPLICATION OF JAPANESE LEARNING BASED ON J2ME**, Manajemen Informatika, Diploma III Ilmu Komputer.

Japan is a country of advanced technology and will have attraction in the anime. In addition, Japan also has a tradition that is not less interesting. Therefore, many technology interests, anime, and Japan's tradition start to learn Japanese. The Learning of Japanese can be done with media phone, the Mobile Application. Java programming language, especially J2ME. The J2ME is able to be implemented to create a Mobile Application. The aim of this final project is to developed a software for Japanese Language based on J2ME.

Principal method of collecting data in this study are observation and literature study. Some steps are conducted. First, it is to design the menu, to construct the flow, the design UML diagram and finally to implement the design. The implementation is done by Wireless Toolkit 2.5.2 with JDK 6. The editor the program is TextPad.

The application of Japanese Learning is used as media for interested in learning Japanese.

Keywords :

Java Programming, J2ME, Learning, Japanese.

INTISARI

Setiana Maya Habsari, 2009, **APLIKASI PEMBELAJARAN BAHASA JEPANG BERBASIS J2ME**. Manajemen Informatika, Diploma III Ilmu Komputer.

Jepang adalah negara yang maju akan teknologi dan memiliki daya tarik dalam *anime*. Selain itu Jepang juga memiliki tradisi – tradisi yang tidak kalah menarik. Maka dari itu, banyak peminat teknologi, *anime*, dan tradisi – tradisi Jepang mulai mempelajari bahasa Jepang. Pembelajaran tersebut dapat dilakukan dengan media ponsel, yaitu *Mobile Application*. Bahasa pemrograman java khususnya J2ME, mampu diimplementasikan untuk membuat *Mobile Application*. Tugas akhir ini bertujuan untuk membuat aplikasi pembelajaran bahasa Jepang berbasis J2ME.

Metode pokok pengumpulan data dalam penelitian ini adalah observasi dan studi pustaka. Beberapa langkah telah dikerjakan. Pertama, mendesain menu, merancang aliran, mendesain diagram UML dan akhirnya untuk mengimplementasikan desain yang dirancang. Implementasinya telah dibuat dengan Wireless Toolkit 2.5.2 dan JDK 6, editor programnya menggunakan TextPad.

Program aplikasi pembelajaran bahasa Jepang ini dijadikan sebagai media pembelajaran bahasa Jepang bagi peminat bahasa Jepang.

Kata Kunci :

Pemrograman Java, J2ME, Pembelajaran, Bahasa Jepang.

MOTTO

Hal yang kamu anggap tidak sempurna bisa menjadi kesempurnaan bagi orang lain dan hal yang kamu anggap sempurna adalah hal yang paling tidak sempurna.
(Penulis)

PERSEMBAHAN

Tugas akhir ini penulis persembahkan pada :

1. Orang tua yang selalu memberikan dukungannya.
2. Adik – adik, nenek dan Keluarga besar Trah S yang selalu menemani dan memberikan ide – ide cemerlang dalam pembuatan tugas akhir ini.
3. Semua sahabat dan teman – teman yang telah memberikan motivasi
4. Semua pembaca tugas akhir ini.

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Pertama-tama marilah kita panjatkan puji syukur kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan Tugas Akhir (TA). Pembuatan laporan ini dimaksudkan guna memenuhi salah satu syarat untuk kelulusan Program Diploma III (D3) Manajemen Informatika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret Surakarta.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna karena keterbatasan kemampuan dan pengetahuan penulis serta adanya keterbatasan data yang diperoleh. Dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada :

1. Bapak Drs. Y.S. Palgunadi, M.Sc selaku Ketua Program DIII Ilmu Komputer FMIPA Universitas Sebelas Maret Surakarta.
2. Bapak Irwan Susanto, S.Si, DEA selaku Pembimbing di FMIPA UNS yang telah memberikan bimbingan kepada penulis.
3. Orang tua tercinta yang senantiasa memberikan motivasi dan perhatian.
4. Adik-adik, nenek dan keluarga besar Trah S yang selalu memberikan doa dan dukungan serta menemaniku dalam pembuatan laporan ini.
5. Sahabat-sahabatku ndong2com (NNC) yang selalu memberikan semangat, motivasi, dan perhatian.
6. Rekan Diploma III (D3) MI, TI, TK Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret Surakarta angkatan 2006.
7. Semua teman-teman di kampus FMIPA UNS khususnya jurusan Manajemen Informatika angkatan 2006.
8. Semua pihak yang telah membantu dalam penyusunan laporan ini yang tidak dapat penulis sebutkan satu per satu.

Semoga Allah SWT membalas semua kebaikan yang telah mereka berikan, amin. Akhir kata semoga laporan ini dapat menjadi manfaat bagi semua pihak

terutama bagi mahasiswa program studi Diploma III Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret dan pembaca pada umumnya.

Wassalamu'alaikum Wr. Wb.

Sukoharjo, 2 Juni 2009

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
ABSTRACT	iv
INTISARI	v
MOTTO	vi
PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	1
1.3 Pembatasan Masalah	1
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Metode Penelitian	2
1.6.1 Jenis dan Sumber Data	2
1.6.2 Metode Pengumpulan Data	2
1.6.3 Metode Analisa Sistem	3
1.6.4 Perencanaan Content Sistem	3
1.6.5 Implementasi Sistem	3
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Pemrograman Java	5
2.2 J2ME	7
2.3 Connected Limited Device Configuration (CLDC)	8
2.4 Mobile Information Device Profile (MIDP)	9

2.5 MIDlet	10
2.6 Unified Modeling Language (UML).....	10
2.7 Use Case Diagram.....	15
2.8 Activity Diagram.....	15
2.9 Class Diagram	16
2.10 Software Emulator	16
BAB III DESAIN DAN PERANCANGAN SISTEM	18
3.1 Spesifikasi Sistem	18
3.2 Gambaran Kerja Sistem	18
3.3 Use Case Diagram.....	19
3.4 Activity Diagram.....	19
3.5 Class Description	22
3.6 Class Diagram	29
BAB IV IMPLEMENTASI DAN ANALISA SISTEM	30
a. Langkah Langkah Pembuatan Sistem	30
i. Penggunaan TextField untuk Inputan	30
ii. Penggunaan File txt untuk Penyimpanan Data	31
iii. Penggunaan List untuk Menu.....	31
b. Keperluan Sistem	33
c. Hasil Pengujian Sistem	33
i. Tampilan Awal Aplikasi	33
ii. Tampilan Huruf Hiragana	34
iii. Tampilan Tutorial.....	35
iv. Tampilan Kamus	36
d. Evaluasi Sistem	38
4.4.1 Kelemahan dari Aplikasi Pembelajaran Bahasa Jepang	38
4.4.2 Kelebihan dari Aplikasi Pembelajaran Bahasa Jepang	39
BAB V KESIMPULAN DAN SARAN.....	40
5.1 Kesimpulan	40
5.2 Saran.....	40

DAFTAR PUSTAKA

DAFTAR TABEL

	Halaman
Tabel 3.1 Tabel Deskripsi Use Case Diagram Interface	19
Tabel 3.2 Tabel Deskripsi Class.....	23

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Platform Java.....	5
Gambar 2.2 Arsitektur J2ME.....	7
Gambar 2.3 MIDP User Interface.....	10
Gambar 2.4 Sebuah Class dari Model UML.....	11
Gambar 2.5 Sebuah <i>Interface</i> /antar-muka.....	12
Gambar 2.6 <i>Collaboration</i>	12
Gambar 2.7 <i>Use Case</i>	12
Gambar 2.8 Nodes.....	13
Gambar 2.9 <i>Dependency</i>	13
Gambar 2.10 <i>Association</i>	14
Gambar 2.11 <i>Generalizations</i>	14
Gambar 2.12 <i>Realization</i>	14
Gambar 2.13 Tampilan Sistem WTK.....	17
Gambar 3.1 Use Case Diagram Interface Sistem.....	19
Gambar 3.2 Activity Diagram Huruf Jepang.....	20
Gambar 3.3 Activity Diagram Tutorial Pembelajaran Bahasa Jepang.....	21
Gambar 3.4 Activity Diagram Kamus Jepang-Indonesia.....	22
Gambar 3.5 Class Diagram.....	29
Gambar 4.1 Tampilan Aplikasi Sebelum Dijalankan.....	33
Gambar 4.2 Tampilan Awal Aplikasi.....	34
Gambar 4.3 Tampilan Huruf Hiragana.....	34
Gambar 4.4 Tampilan Huruf Katakana.....	35
Gambar 4.5 Tampilan Tutorial Tanggal dan Hari.....	35
Gambar 4.6 Tampilan Tutorial Kebudayaan Jepang.....	36
Gambar 4.7 Tampilan Kamus.....	36
Gambar 4.8 Tampilan Kamus.....	37
Gambar 4.9 Tampilan Kamus saat Dua Terjemahan yang Ditemukan.....	37
Gambar 4.10 Tampilan Kamus saat Terjemahan tidak Ditemukan.....	38

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Jepang merupakan salah satu negara maju dalam bidang teknologi, misalnya dalam pengembangan robotic. Selain itu, Jepang juga memiliki daya tarik dalam bidang kebudayaan maupun dalam dunia seni, terutama *anime*. *Anime* bisa didefinisikan yaitu menceritakan kisah dengan menggunakan media gambar seperti manga ataupun kartun.

Untuk mendapatkan informasi terbaru dari kedua hal tersebut terkadang didapati informasi–informasi yang masih berbahasa Jepang. Peminat anime tak jarang mendapatkan *anime* favoritnya yang belum diterjemahkan dalam bahasa Inggris maupun bahasa Indonesia, sehingga harus menunggu sampai *anime* tersebut diterjemahkan. Hal ini tentu saja menghambat mereka untuk memperoleh informasi–informasi terbaru dari hal – hal yang mereka gemari.

Dari aspek di atas tercetus ide untuk membuat sebuah aplikasi pembelajaran bahasa jepang, untuk memudahkan peminat teknologi maupun peminat *anime* berkomunikasi dan mengetahui informasi terbaru dalam bidang yang mereka gemari. Aplikasi ini diberi nama Tanabata dikarenakan Tanabata merupakan salah satu nama perayaan yang menarik di Jepang.

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan di atas, maka dapat dirumuskan masalah “Bagaimana membuat aplikasi pembelajaran bahasa jepang berbasis J2ME?”.

1.3 Pembatasan Masalah

Dalam pembuatan tugas akhir ini penulis hanya akan membuat aplikasi pembelajaran bahasa jepang meliputi huruf–huruf Jepang, tutorial pembelajaran bahasa Jepang serta kamus Jepang–Indonesia dan Indonesia-Jepang.

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk membuat sebuah aplikasi pembelajaran bahasa jepang untuk *handphone* berbasis J2ME.

1.5 Manfaat Penelitian

Adapun manfaat penelitian yang diperoleh adalah

1. Bagi Penulis

Mengaplikasikan ilmu yang telah penulis dapatkan di perkuliahan serta membantu dunia pendidikan untuk pengenalan bahasa jepang.

2. Bagi Pengguna (*User*)

Sebagai sarana pembelajaran berbahasa khususnya bahasa jepang sehingga peminat bahasa jepang dapat mempelajarinya dimana saja.

1.6 Metode Penelitian

1.6.1 Jenis dan Sumber Data

Dalam penelitian ini terdapat dua jenis data yang digunakan

1. Data Primer

Data yang langsung diperoleh dari sumber atau obyek yang sedang diteliti. Dalam hal ini, data primer diperoleh dari pembelajaran jepang online maupun dari modul – modul bimbingan bahasa jepang yang akan digunakan sebagai acuan pembuatan aplikasi pembelajaran bahasa jepang.

2. Data Sekunder

Data diperoleh dengan berbagai macam kepustakaan yang berkaitan dengan pembuatan aplikasi, baik yang diperoleh melalui internet maupun studi pustaka.

1.6.2 Metode Pengumpulan Data

Dalam penelitian tugas akhir ini dilakukan dengan beberapa metode antara lain

1. Metode Observasi

Metode yang dilakukan dengan pengamatan langsung tentang bagaimana pembelajaran bahasa Jepang.

2. Metode Studi Pustaka

Metode atau teknik pengumpulan data menggunakan pustaka – pustaka yang telah ada untuk digunakan sebagai referensi atau bahkan sebagai bahan perbandingan.

1.6.3 Metode Analisa Sistem

Metode analisa sistem menggunakan analisa sistem penguraian yaitu dari sistem informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan dan hambatan-hambatan yang terjadi serta kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan.

1.6.4 Perencanaan *Content* Sistem

Perencanaan *content* sistem disusun untuk menentukan fasilitas-fasilitas yang ada dalam aplikasi pembelajaran bahasa Jepang tersebut yaitu huruf-huruf Jepang, tutorial dan kamus Jepang-Indonesia, Indonesia-Jepang sehingga sistem menjadi powerful dengan fasilitas – fasilitas yang bermanfaat.

1.6.5 Implementasi Sistem

Tahap ini merupakan kegiatan untuk mengimplementasikan rancangan yang telah disusun agar dapat diwujudkan. Tahapan implementasi sistem mencakup pengkodean program dan deployment program. Setelah tahap ini berakhir maka akan sampai pada tahap evaluasi sistem yaitu menguji sistem dan memastikan bahwa sistem tersebut sudah benar sesuai dengan alur sistem secara keseluruhan.

1.7 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini menguraikan tentang latar belakang, Rumusan Masalah, Pembatasan Masalah, Tujuan Penelitian, Manfaat Penelitian, Metode Penelitian dan Sistematika Penulisan.

BAB II LANDASAN TEORI

Bab ini menguraikan tentang Pemrograman Java, J2ME, *Connected Limited Device Configuration* (CLDC), *Mobile Information Device Profile* (MIDP), MIDlet, *Unified Modeling Language* (UML), *Use Case Diagram*, *Activity Diagram*, *Class Diagram* dan Software Emulator

BAB III DESAIN DAN PERANCANGAN SISTEM

Bab ini menguraikan tentang Spesifikasi Sistem, Gambaran Kerja Sistem, *Use Case Diagram*, *Activity Diagram*, *Class Description* dan *Class Diagram*.

BAB IV IMPLEMENTASI DAN ANALISA SISTEM

Bab ini menguraikan tentang Langkah Langkah Pembuatan Sistem, Keperluan Sistem dan Hasil Pengujian Sistem.

BAB V KESIMPULAN DAN SARAN

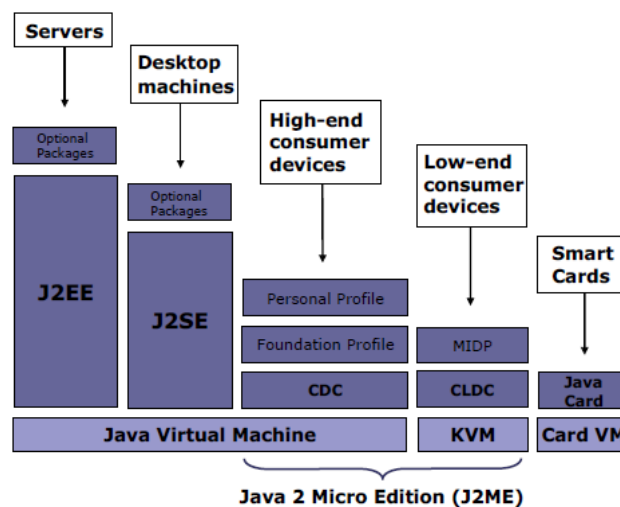
Bab ini menguraikan tentang Kesimpulan dan Saran.

BAB II LANDASAN TEORI

2.1 Pemrograman Java

Pada awalnya, Java ditujukan untuk pemrograman device kecil berupa chip. Karena suatu hal, perkembangannya menjadi terancam. Karena fitur – fitur java cocok dengan pengembangan internet, maka java diarahkan untuk pemrograman web. Pada perkembangannya, Java telah dikembangkan menjadi sebuah bahasa universal yang mendukung pemrograman dekstop.

Penamaan java terinspirasi ketika anggota tim sedang minum kopi di sebuah kedai. Salah seorang dari mereka bercanda tentang kopi dan teringat pada Pulau Jawa. Akhirnya, mereka sepakat untuk menamakan bahasa java dengan logo secangkir kopi panas. Perkembangan java dapat dilihat pada situs yang dikelola Sun Microsystem pada halaman web <http://www.sun.java.com> secara gratis. Gambar *platform* java ditunjukkan pada Gambar 2.1.



Gambar 2.1 *Platform* Java

Platform java terdiri dari kumpulan *library*, JVM, kelas – kelas *loader* yang dipaket dalam sebuah lingkungan rutin java, sebuah *compiler*, *debugger*, dan perkakas lain yang dipaket dalam *Java Development Kit* (JDK). Java 2 adalah generasi yang sekarang sedang berkembang dari *platform* java. Agar sebuah

program java dapat dijalankan, maka *file* dengan ekstensi Java harus dikompilasi menjadi *file bytecode*. Untuk menjalankan *bytecode* ini, dibutuhkan JRE (*Java Runtime Environment*) yang memungkinkan pemakai untuk menjalankan program java, tidak untuk membuat kode baru lagi. JRE berisi JVM dan library java yang digunakan. (Shalahuddin dan Rosa, 2008)

Java 2 *Platform* dibagi menjadi 3 kategori, yaitu

1. *Java 2 Standard Edition (J2SE)*

Kategori ini digunakan untuk mengembangkan dan menjalankan aplikasi java berbasis PC.

2. *Java 2 Enterprise Edition (J2EE)*

Kategori ini digunakan untuk mengembangkan dan menjalankan aplikasi java pada lingkungan enterprise, dengan fungsi – fungsi seperti *Enterprise Java Bean (EJB)*, *Servlet* dan *Java Server Page (JSP)*.

3. *Java 2 Micro Edition (J2ME)*

Kategori ini digunakan untuk mengembangkan dan menjalankan aplikasi java berbasis *handheld device*, seperti *Personal Digital Assistant (PDA)*, *handphone* dan *pocketPC*.

Java digunakan dalam programming karena memiliki keunggulan – keunggulan yang tidak dimiliki oleh bahasa lain. Berikut beberapa aspek yang menjadikan Java sebagai bahasa pemrograman yang digemari saat ini.

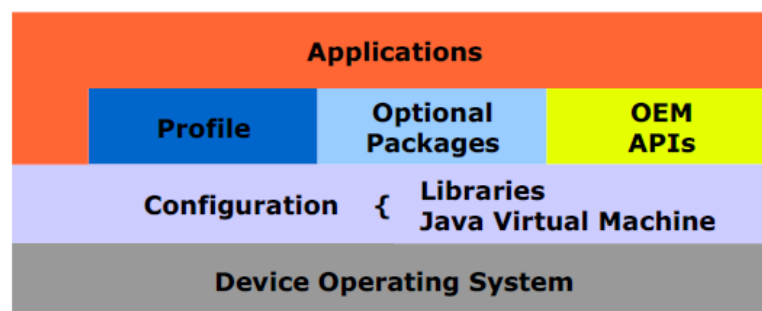
1. Java bersifat sederhana dan relatif mudah digunakan. Hal ini disebabkan Java dikembangkan dari bahasa C++ yang bersifat *object oriented* dengan menghilangkan beberapa fitur dan menambahkan fitur – fitur yang dianggap penting.
2. Java berorientasi objek (*object oriented*) dalam menyusun program. Java membagi program menjadi objek – objek, kemudian objek – objek tersebut di modelkan sesuai karakteristik masing – masing. Selanjutnya, Java akan mengatur hubungan antar objek tersebut.
3. Java bersifat terdistribusi, yaitu dapat digunakan pada jaringan yang luas, tidak hanya pada personal komputer (PC).

4. Java bersifat *Multiplatform* sehingga Java dapat digunakan dalam berbagai sistem operasi, baik Windows, Linux, Apple dan lainnya tanpa melakukan banyak konfigurasi.
5. Java bersifat *Multithread*, yaitu mampu mengerjakan beberapa proses dalam waktu yang hampir bersamaan.

2.2 J2ME

J2ME adalah satu set spesifikasi dan teknologi yang fokus kepada perangkat konsumen. Perangkat ini memiliki jumlah memori yang terbatas, menghabiskan sedikit daya dari baterai, layar yang kecil dan bandwidth jaringan yang rendah. Dengan berkembangnya perangkat mobile konsumen dari telepon, PDA, kotak permainan ke peralatan-peralatan rumah, Java menyediakan suatu lingkungan yang portable untuk mengembangkan dan menjalankan aplikasi pada perangkat ini.

Program J2ME, seperti semua program JAVA adalah diterjemahkan oleh VM. Program-program tersebut dikompil ke dalam *bytecode* dan diterjemahkan dengan JVM. Ini berarti bahwa program-program tersebut tidak berhubungan langsung dengan perangkat. J2ME menyediakan suatu *interface* yang sesuai dengan perangkat. Aplikasi-aplikasi tersebut tidak harus dikompil ulang supaya mampu dijalankan pada mesin yang berbeda. Arsitektur J2ME dapat dilihat di Gambar 2.2.



Gambar 2.2 Arsitektur J2ME

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 dekstop yang biasanya lebih kecil dibandingkan perangkat komputer dekstop. J2ME biasa digunakan pada telepon seluler, pager, PDA dan sejenisnya.

J2ME adalah bagian dari J2SE, karena itu tidak semua *library* yang ada pada J2SE dapat digunakan pada J2ME. Tetapi J2ME mempunyai beberapa *library* khusus yang tidak dimiliki J2SE. J2ME sangat tergantung pada device yang digunakan terutama jika diaplikasikan pada ponsel misalnya dari segi merk ponsel, maupun kemampuan ponsel dan dukungannya terhadap teknologi J2ME. Selain itu juga memiliki keterbatasan pada ukuran aplikasi, karena *memory* ponsel sangat terbatas selain itu ada juga ponsel yang tidak mengizinkan aplikasi J2ME dengan alasan keamanan.

J2ME dibagi dalam *configuration* dan *profile*. *Configuration* adalah Java *library* minimum dan kapabilitas yang dipunya oleh para pengembang J2ME, yang maksudnya sebuah *mobile device* dengan kemampuan java akan optimal untuk menjadi sesuai. *Configuration* hanyalah mengatur hal-hal tentang kesamaan sehingga dapat dijadikan ukuran kesesuaian antar-*device*. Dalam J2ME telah didefinisikan dua buah konfigurasi yaitu *Connected Limited Device Configuration* (CLDC) untuk perangkat kecil dan *Connected Device Configuration* (CDC) untuk perangkat yang lebih besar.

Profile berbeda dengan *Configuration*, *profile* membahas sesuatu yang spesifik untuk sebuah perangkat. Sebagai contoh misalnya sebuah sepeda dengan merk tertentu tentu mempunyai ciri spesifik dengan sepeda lain. (Shalahuddin dan Rosa, 2008). Profil merupakan bagian perluasan dari konfigurasi, jadi profil menyediakan kelas – kelas yang tidak terdapat pada konfigurasi. Contoh profil adalah *Mobile Information Device Profile* (MIDP). (Yuniar, 2008).

2.3 Connected Limited Device Configuration (CLDC)

CLDC merupakan perangkat dasar dari J2ME, spesifikasi dasar yang berupa *library* dan API yang diimplementasikan pada J2ME, seperti yang digunakan pada

telepon selular, pager, dan PDA. Perangkat tersebut dibatasi dengan keterbatasan memori, sumber daya, dan kemampuan memproses. Spesifikasi CLDC pada J2ME adalah spesifikasi minimal dari *package*, kelas, dan sebagian fungsi JVM yang dikurangi agar dapat diimplementasikan dengan keterbatasan sumber daya pada alat – alat tersebut, JVM yang digunakan disebut *Kiloibyte Virtual Machine* (KVM). (Shalahuddin dan Rosa, 2008)

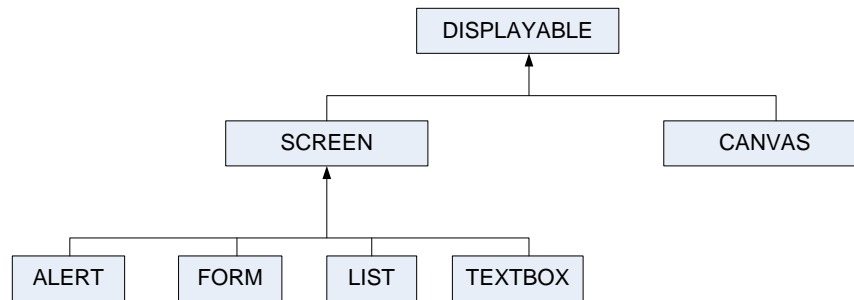
Spesifikasi CLDC mengidentifikasikan piranti yang mempunyai karakteristik sbb :

1. Untuk menjalankan Java dibutuhkan memori 128 kilobytes.
2. Untuk alokasi memori runtime dibutuhkan memori 32.
3. Antara muka pengguna sangat terbatas.
4. Konektifitas jaringan, biasanya wireless, dengan lebar pita (bandwidth) terbatas dan akses yang terputus – putus.
5. Untuk platform Java tersedia memori 160 -512 kb.
6. Prosesor 16/ 32 bit.
7. Konsumsi daya rendah, sering menggunakan baterai yang dapat diisi ulang. (Suyoto, 2007).

2.4 Mobile Information Device Profile (MIDP)

MIDP adalah profile yang mendefinisikan model aplikasi yang bisa dijalankan pada device dengan resource terbatas dan memungkinkannya untuk digunakan oleh beberapa aplikasi MIDP secara bersama – sama. MIDP mendefinisikan lingkungan terpadu tempat dimana sejumlah MIDlet bisa berjalan (Rangsang, 2008). MIDP memiliki lapisan diatas CLDC, API tambahan untuk daur hidup aplikasi, antarmuka, jaringan dan penyimpanan persisten.

Arsitektur antarmuka MIDP dapat dilihat pada Gambar 2.3.



Gambar 2.3 MIDP *User Interface*

2.5 MIDlet

MIDlet adalah sebutan untuk aplikasi yang berjalan dengan konfigurasi MIDP. Sebuah aplikasi adalah sebuah MIDlet. Java memiliki beberapa format aplikasi yang berakhiran `-let`, MIDlet untuk aplikasi java jenis MIDP, servlet untuk aplikasi java jenis client-server, dan applet untuk jenis aplikasi Java yang ditempatkan ke dalam aplikasi web. Sebuah project J2ME dikatakan terbentuk sebagai sebuah MIDP, sedangkan sebuah project J2ME bisa tersusun dari beberapa aplikasi MIDlet. (Rangsang, 2008)

2.6 Unified Modeling Language (UML)

UML adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML, dapat dibuat model untuk semua jenis aplikasi piranti lunak. Aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa – bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C.

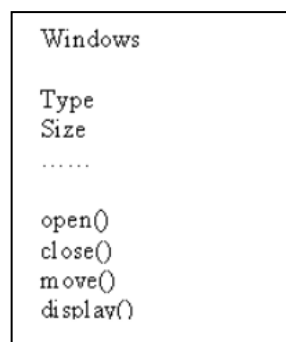
Untuk dapat memahami UML membutuhkan bentuk konsep dari sebuah bahasa model, dan mempelajari 3 (tiga) elemen utama dari UML seperti building

block, aturan-aturan yang menyatakan bagaimana building block diletakkan secara bersamaan, dan beberapa mekanisme umum (*common*). Terdapat tiga macam yang terdapat dalam *building block* adalah kategori benda, hubungan, dan diagram. Benda adalah abstraksi yang pertama dalam sebuah model, hubungan sebagai alat komunikasi dari benda-benda, dan diagram sebagai kumpulan / group dari benda-benda.

1. Benda

Benda merupakan hal yang sangat mendasar dalam model UML, juga merupakan bagian paling statik dari sebuah model, serta menjelaskan elemen-elemen lainnya dari sebuah konsep dan fisik. Bentuk dari beberapa benda antara lain :

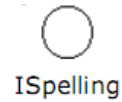
- a. Sebuah kelas yang diuraikan sebagai sekelompok dari *object* yang mempunyai *attribute*, operasi, hubungan yang semantik. Sebuah kelas mengimplementasikan 1 atau lebih *interfaces*. Sebuah kelas dapat digambarkan sebagai sebuah persegi panjang, yang mempunyai sebuah nama, *attribute*, dan metoda pengoperasiannya, seperti terlihat pada Gambar 2.4.



Gambar 2.4 Sebuah Class dari model UML

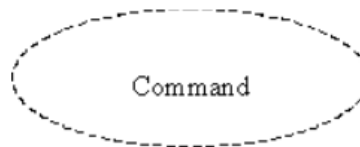
- b. Menggambar *interface* merupakan sebuah antar-muka yang menghubungkan dan melayani antar kelas dan elemen. *Interface* mendefinisikan sebuah set/kelompok dari spesifikasi pengoperasian, umumnya digambarkan dengan sebuah lingkaran

yang disertai dengan namanya. Sebuah antar-muka berdiri sendiri dan umumnya merupakan pelengkap dari kelas atau komponen seperti pada Gambar 2.5.



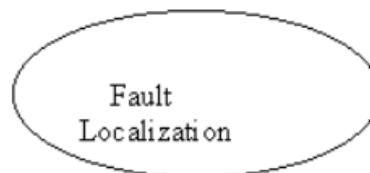
Gambar 2.5 Sebuah *interface*/antar-muka

- c. *Collaboration* yang didefinisikan dengan interaksi dan sebuah kumpulan/kelompok dari kelas-kelas/elemen-elemen yang bekerja secara bersama-sama. *Collaborations* mempunyai struktur dan dimensi. Pemberian sebuah kelas memungkinkan berpartisipasi didalam beberapa *collaborations* dan digambarkan dengan sebuah *elips* dengan garis terpotong-potong seperti pada Gambar 2.6.



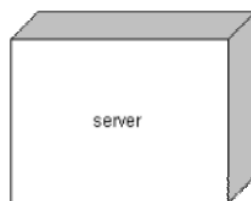
Gambar 2.6 *Collaborations*

- d. Sebuah *use case* adalah rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. *Use case* digunakan untuk membentuk tingkah-laku benda dalam sebuah model serta direalisasikan oleh sebuah *collaboration*. Umumnya *use case* digambarkan dengan sebuah *elips* dengan garis yang solid, biasanya mengandung nama, seperti yang terlihat pada Gambar 2.7.



Gambar 2.7 *Use Case*

- e. Sebuah node merupakan fisik dari elemen-elemen yang ada pada saat dijalankannya sebuah sistem, contohnya adalah sebuah komputer, umumnya mempunyai sedikitnya *memory* dan *processor*. Sekelompok komponen mungkin terletak pada sebuah node dan juga mungkin akan berpindah dari node satu ke node lainnya. Umumnya node ini digambarkan seperti kubus serta hanya mengandung namanya, dapat dilihat pada Gambar 2.8.



Gambar 2.8 Nodes

2. Hubungan/*Relations*

Ada empat macam hubungan didalam penggunaan UML, yaitu :

a. *Dependency*

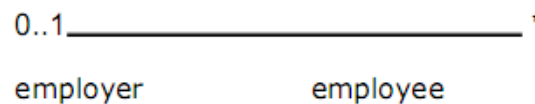
Dependency adalah hubungan semantik antara dua benda yang mana sebuah benda berubah mengakibatkan benda satunya akan berubah pula. Umumnya sebuah *dependency* digambarkan sebuah panah dengan garis terputus-putus seperti pada Gambar 2.9.



Gambar 2.9 *Dependency*

b. *Association*

Association adalah hubungan antar benda struktural yang terhubung diantara obyek. Kesatuan obyek yang terhubung merupakan hubungan khusus, yang menggambarkan sebuah hubungan struktural diantara seluruh aatau sebagian. Umumnya *assosiation* digambarkan dengan sebuah garis yang dilengkapi dengan label, nama dan status hubungannya seperti dapat dilihat pada Gambar 2.10.



Gambar 2.10 Association

c. *Generalization*

Generalization adalah menggambarkan hubungan khusus dalam obyek anak yang menggantikan obyek induk. Dalam hal ini, obyek anak memberikan pengaruhnya dalam hal struktur dan tingkah lakunya kepada obyek induk. Digambarkan dengan garis panah seperti pada Gambar 2.11.



Gambar 2.11 Generalizations

d. *Realization*

Realization merupakan hubungan semantik antara pengelompokan yang menjamin adanya ikatan diantaranya. Hubungan ini dapat diwujudkan diantara *interface* dan kelas atau *elements*, serta antara *use cases* dan *collaborations*. Model dari sebuah *realization* seperti terlihat dalam Gambar 2.12.



Gambar 2.12 Realization

3. Diagram

UML terdiri atas pengelompokan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model. UML mendefinisikan diagram berikut :

- a. *Use Case Diagram*
- b. *Class Diagram*
- c. *Statechart Diagram*
- d. *Activity Diagram*
- e. *Sequence Diagram*

- f. *Collaboration Diagram*
- g. *Component Diagram*
- h. *Deployment Diagram*

2.7 Use Case Diagram

Use case diagram merupakan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case diagram* merepresentasikan sebuah interaksi antara aktor dengan sistem. Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal. Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*.

Sebuah *use case* juga dapat meng-*extend use case* lain dengan *behaviour*-nya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

2.8 Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level secara umum.

Sebuah aktivitas dapat direalisasikan oleh *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

2.9 Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi dan lain-lain. *Class* memiliki tiga area pokok yaitu nama, atribut dan metoda. Atribut dan metoda dapat memiliki salah satu sifat antara lain :

1. *Private*, tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected*, hanya dapat oleh *class* yang bersangkutan dan anak-anak yang mewarisinya.
3. *Public*, dapat dipanggil siapa saja.

Adapun hubungan antar *class* terdiri dari :

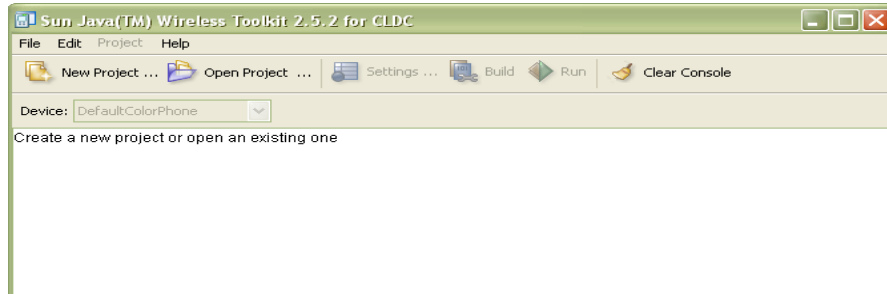
1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain. Panah navigability menunjukkan arah query antar *class*.
2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas..”).
3. Pewarisan, yaitu hubungan hirarkis antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metoda *class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-passing dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram*.

2.10 Software Emulator

Software emulator yang digunakan adalah Wireless Toolkit (WTK). Berbeda dengan J2SE yang tidak memiliki aplikasi khusus, J2ME memiliki aplikasi yang menunjukkan identitas J2ME. Aplikasi tersebut adalah Wireless

Toolkit (WTK), sebuah aplikasi yang dibuat oleh Sun Microsystems, Inc. Untuk keperluan membuat dan menjalankan aplikasi J2ME di komputer. (Rangsang, 2008)

Tampilan dari sistem WTK dapat dilihat di Gambar 2.13.



Gambar 2.13 Tampilan Sistem WTK

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Spesifikasi Sistem

Dalam pembuatan aplikasi pembelajaran bahasa jepang *mobile* ini sistem dirancang dan dibuat dengan tujuan untuk mempelajari bahasa jepang terutama untuk berkomunikasi dengan bahasa tersebut. Adapun aplikasi ini meliputi huruf jepang, dan kamus Jepang-Indonesia. Spesifikasi *handphone* yang dapat mengaplikasikan aplikasi ini adalah *handphone* yang memiliki fitur java ber-MIDP 2.0 dan CLDC 1.0.

3.2 Gambaran Kerja Sistem

Aplikasi ini merupakan sebuah sistem pembelajaran bahasa jepang yang dirancang untuk mempelajari bahasa jepang. Pengguna aplikasi dapat mempelajari huruf jepang, tutorial pembelajaran serta kamus Jepang – Indonesia dan kamus Indonesia - Jepang.

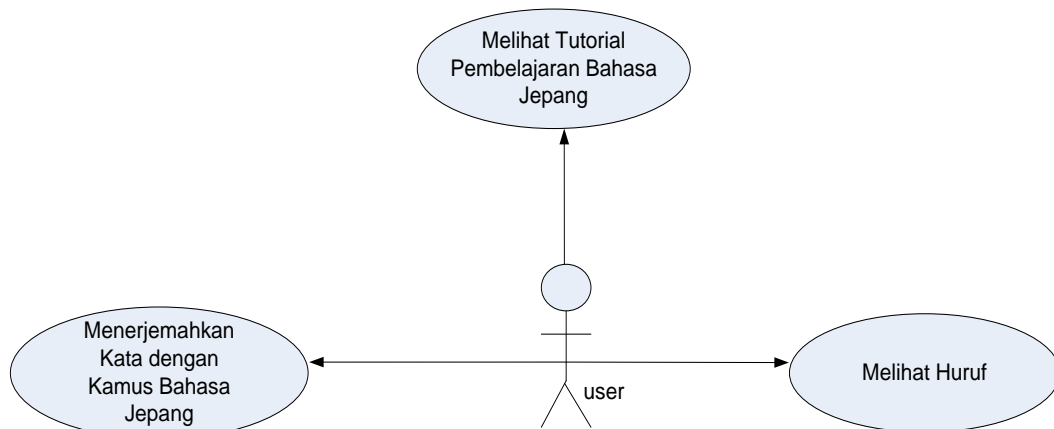
Huruf Jepang adalah suatu proses yang dilakukan oleh pengguna untuk melihat macam – macam huruf jepang, sebagai contoh yaitu huruf hiragana, katakana dan kanji. Selain itu juga terdapat penjelasan singkat mengenai ketiga huruf tersebut.

Tutorial merupakan suatu proses yang dilakukan oleh pengguna untuk melihat tutorial yang dibagi menjadi beberapa pokok bahasan seperti memperkenalkan diri, menanyakan usia dan berbagai pokok bahasan lainnya. Tutorial tersebut didapatkan dari pembelajaran bahasa Jepang online dan juga didapatkan dari bimbingan belajar.

Kamus adalah suatu proses yang dilakukan oleh pengguna untuk menerjemahkan kata. Pengguna akan menginputkan sebuah kata berbahasa Jepang maupun Indonesia kemudian pengguna akan mendapatkan terjemahannya kata tersebut dalam bahasa Jepang maupun Indonesia.

3.3 Use Case Diagram

Use Case Diagram mendefinisikan fitur umum yang dimiliki oleh sistem. Interaksi antara aktor dan sistem, dalam hal ini aktor merupakan *user* sedangkan sistem adalah aplikasi pembelajaran bahasa Jepang *mobile* dapat dilihat pada Gambar 3.1.



Gambar 3.1 *Use Case Diagram Interface Sistem*

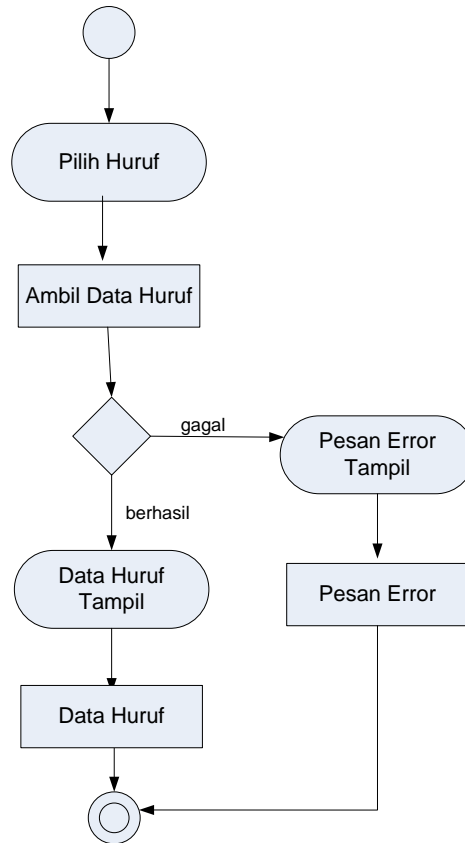
Tabel 3.1 Tabel Deskripsi *Use Case Diagram Interface*

Aktor	<i>User</i>	
Deskripsi	<i>Use Case</i>	Mempelajari bahasa Jepang
	Aktor	<i>User</i>
		<ul style="list-style-type: none"> - <i>User</i> dapat melihat huruf Jepang - <i>User</i> dapat melihat tutorial - <i>User</i> dapat menerjemahkan kata dengan kamus Jepang - Indonesia

3.4 Activity Diagram

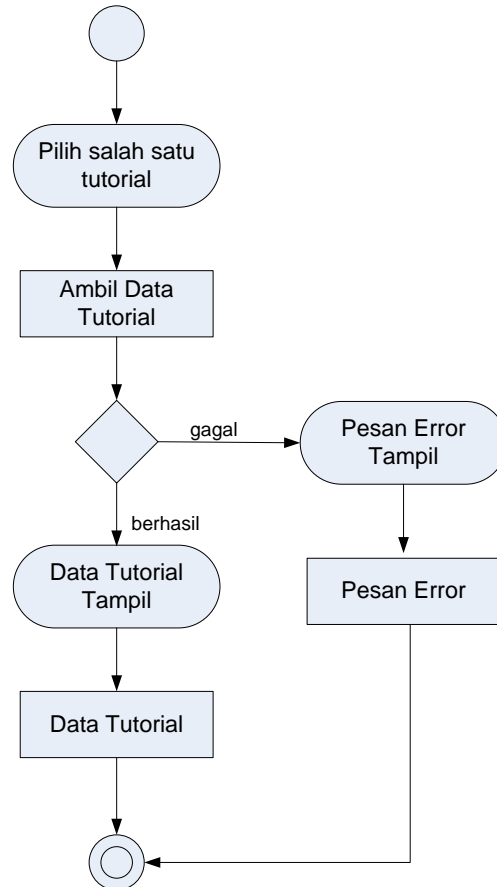
Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Rancangan dari alur aktivitas aplikasi pembelajaran bahasa jepang *mobile* yang ada pada aplikasi ini dapat dilihat pada gambar berikut :



Gambar 3.2 *Activity Diagram* Huruf Jepang

Gambar 3.2 menggambarkan alur aktifitas dari pilihan huruf yang ada. Saat pengguna memilih salah satu dari huruf jepang misalkan hiragana maka akan mengakses data dari huruf hiragana dan menampilkannya di layar.

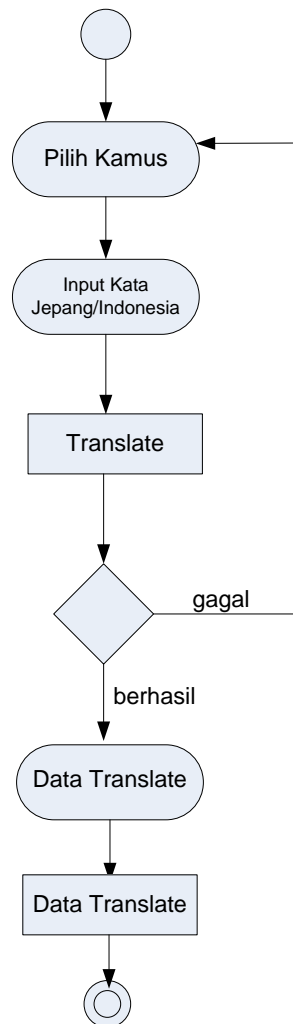


Gambar 3.3 *Activity Diagram* Tutorial Pembelajaran Bahasa Jepang

Gambar 3.3 menggambarkan alur aktifitas dari pilihan tutorial yang ada. Saat pengguna memilih salah satu dari tutorial misalkan perkenalan maka akan mengakses data dari tutorial perkenalan dan menampilkannya di layar. Untuk alur aktifitas dari pilihan tutorial yang lain juga sama seperti alur dari tutorial perkenalan.

Alur aktifitas kamus dapat dilihat pada Gambar 3.4. Pengguna memilih menu kamus kemudian diminta inputan berupa kata dalam bahasa Jepang maupun dalam bahasa Indonesia. Kemudian sistem akan mencari arti kata tersebut dalam bahasa Jepang ataupun bahasa Indonesia. Jika satu kata terdeteksi menjadi salah satu kata dalam bahasa Jepang dan kata dalam bahasa

Indonesia maka akan menghasilkan dua output berupa arti kata tersebut dalam bahasa Indonesia dan bahasa Jepang.



Gambar 3.4 *Activity Diagram* Kamus Jepang-Indonesia, Indonesia-Jepang

3.5 Class Description

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi

objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class Description adalah penjelasan tiap kelas mengenai *attribute* dan *method* yang terdapat pada *class* tersebut, untuk aplikasi pembelajaran bahasa jepang ini sendiri. *Class Description* dapat dilihat pada tabel dibawah ini.

Tabel 3.2 Deskripsi *Class*

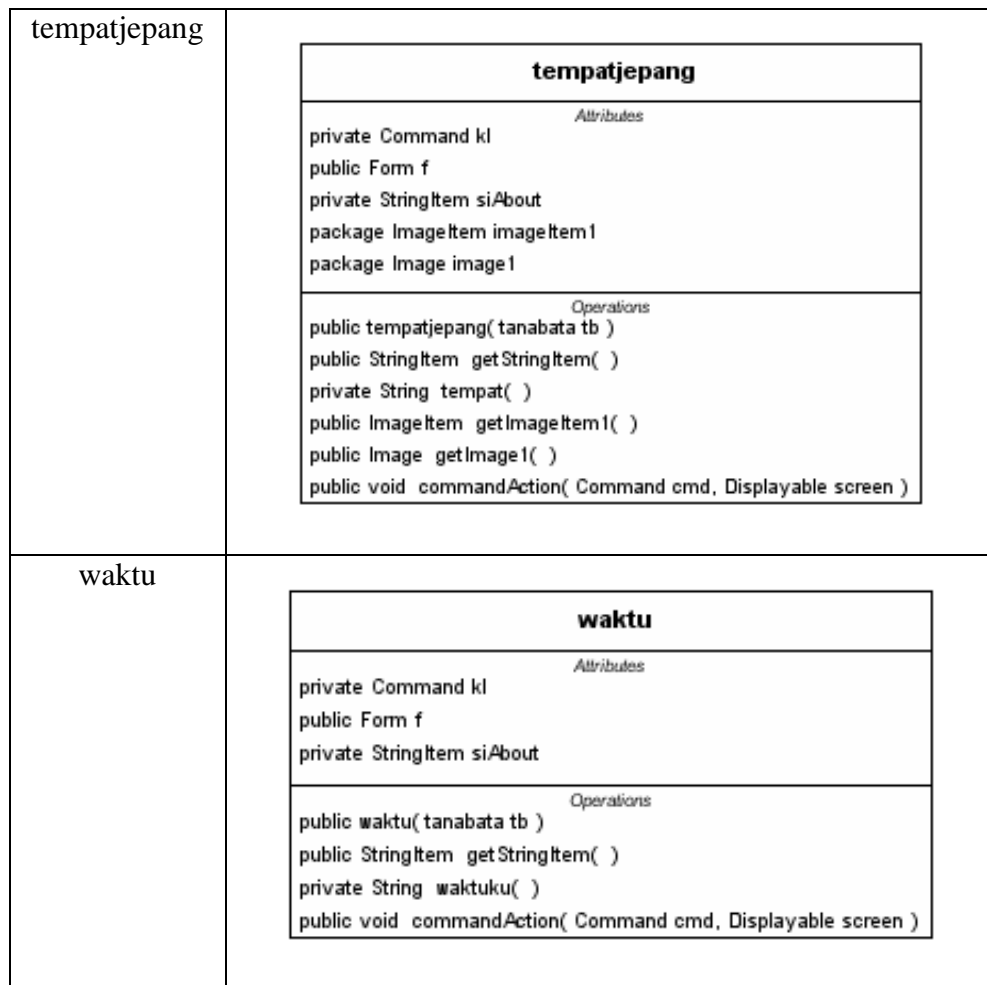
Nama Kelas	Properti dan Method			
tanabata	<table border="1" style="margin: auto;"> <thead> <tr> <th data-bbox="732 701 1225 750" style="text-align: center;">tanabata</th> </tr> </thead> <tbody> <tr> <td data-bbox="732 750 1225 925" style="text-align: center;"><i>Attributes</i></td> </tr> <tr> <td data-bbox="732 925 1225 1115" style="text-align: center;"><i>Operations</i></td> </tr> </tbody> </table> <pre> package Display display package List list package Image img package Command cmExit package Command cmBack public tanabata() public void startApp() public void pauseApp() public void destroyApp(boolean unconditional) public void commandAction(Command c, Displayable s) public void exitMIDlet() </pre>	tanabata	<i>Attributes</i>	<i>Operations</i>
tanabata				
<i>Attributes</i>				
<i>Operations</i>				
about	<table border="1" style="margin: auto;"> <thead> <tr> <th data-bbox="691 1193 1267 1243" style="text-align: center;">about</th> </tr> </thead> <tbody> <tr> <td data-bbox="691 1243 1267 1368" style="text-align: center;"><i>Attributes</i></td> </tr> <tr> <td data-bbox="691 1368 1267 1507" style="text-align: center;"><i>Operations</i></td> </tr> </tbody> </table> <pre> private Command kl public Form f private StringItem siAbout public about(tanabata tb) public StringItem getStringItem() private String tentang() public void commandAction(Command cmd, Displayable screen) </pre>	about	<i>Attributes</i>	<i>Operations</i>
about				
<i>Attributes</i>				
<i>Operations</i>				

angka	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">angka</th> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>private Command kl public Form f private StringItem siAbout</pre> </td> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>public angka(tanabata tb) public StringItem getStringItem() private String usia() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </table>	angka	<i>Attributes</i>	<pre>private Command kl public Form f private StringItem siAbout</pre>	<i>Operations</i>	<pre>public angka(tanabata tb) public StringItem getStringItem() private String usia() public void commandAction(Command cmd, Displayable screen)</pre>
angka						
<i>Attributes</i>						
<pre>private Command kl public Form f private StringItem siAbout</pre>						
<i>Operations</i>						
<pre>public angka(tanabata tb) public StringItem getStringItem() private String usia() public void commandAction(Command cmd, Displayable screen)</pre>						
Budaya	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">budaya</th> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>private Command kl public Form f private StringItem siAbout package ImageItem imageItem2 package Image image3 package ImageItem imageItem3 package Image image4 package ImageItem imageItem4 package Image image5 package ImageItem imageItem5 package Image image6</pre> </td> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>public budaya(tanabata tb) public StringItem getStringItem() private String kebudayaan() public ImageItem getImageItem2() public Image getImage3() public ImageItem getImageItem3() public Image getImage4() public ImageItem getImageItem4() public Image getImage5() public ImageItem getImageItem5() public Image getImage6() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </table>	budaya	<i>Attributes</i>	<pre>private Command kl public Form f private StringItem siAbout package ImageItem imageItem2 package Image image3 package ImageItem imageItem3 package Image image4 package ImageItem imageItem4 package Image image5 package ImageItem imageItem5 package Image image6</pre>	<i>Operations</i>	<pre>public budaya(tanabata tb) public StringItem getStringItem() private String kebudayaan() public ImageItem getImageItem2() public Image getImage3() public ImageItem getImageItem3() public Image getImage4() public ImageItem getImageItem4() public Image getImage5() public ImageItem getImageItem5() public Image getImage6() public void commandAction(Command cmd, Displayable screen)</pre>
budaya						
<i>Attributes</i>						
<pre>private Command kl public Form f private StringItem siAbout package ImageItem imageItem2 package Image image3 package ImageItem imageItem3 package Image image4 package ImageItem imageItem4 package Image image5 package ImageItem imageItem5 package Image image6</pre>						
<i>Operations</i>						
<pre>public budaya(tanabata tb) public StringItem getStringItem() private String kebudayaan() public ImageItem getImageItem2() public Image getImage3() public ImageItem getImageItem3() public Image getImage4() public ImageItem getImageItem4() public Image getImage5() public ImageItem getImageItem5() public Image getImage6() public void commandAction(Command cmd, Displayable screen)</pre>						
date	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">date</th> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>private Command kl public Form f private StringItem siAbout</pre> </td> </tr> <tr> <td style="padding: 5px; text-align: center;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>public date(tanabata tb) public StringItem getStringItem() private String time() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </table>	date	<i>Attributes</i>	<pre>private Command kl public Form f private StringItem siAbout</pre>	<i>Operations</i>	<pre>public date(tanabata tb) public StringItem getStringItem() private String time() public void commandAction(Command cmd, Displayable screen)</pre>
date						
<i>Attributes</i>						
<pre>private Command kl public Form f private StringItem siAbout</pre>						
<i>Operations</i>						
<pre>public date(tanabata tb) public StringItem getStringItem() private String time() public void commandAction(Command cmd, Displayable screen)</pre>						

hurufhira	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">hurufhira</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px; vertical-align: top;"> <div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre> </td> </tr> <tr> <td style="padding: 5px; vertical-align: top;"> <div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufhira(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </tbody> </table>	hurufhira	<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre>	<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufhira(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre>
hurufhira				
<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre>				
<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufhira(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre>				
hurufkata	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">hurufkata</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px; vertical-align: top;"> <div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre> </td> </tr> <tr> <td style="padding: 5px; vertical-align: top;"> <div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufkata(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </tbody> </table>	hurufkata	<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre>	<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufkata(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre>
hurufkata				
<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Attributes</i></div> <pre>private Command kl public Form f package StringItem stringItem package ImageItem imageItem package ImageItem imageItem1 package Image image2 package Image image3 package Display display</pre>				
<div style="text-align: right; font-size: small; margin-bottom: 5px;"><i>Operations</i></div> <pre>public hurufkata(tanabata tb) public StringItem getStringItem() private String hiragana() public ImageItem getImageItem() public ImageItem getImageItem1() public Image getImage2() public Image getImage3() public void commandAction(Command cmd, Displayable screen)</pre>				

kamus	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">kamus</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 2px 5px;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>private Command kl public Form f private TextField text private TextField arti1 private TextField arti2 private TextField pesan private Command submit = new Command("Translate", Command.OK, 2) private Alert alert private byte array1 [0..*] = new byte[10000] private byte temp[0..*] = new byte[1000] private int begin private int length private String firstlastj[0..*] = new String[42] private String firstlastij[0..*] = new String[42]</pre> </td> </tr> <tr> <td style="text-align: center; padding: 2px 5px;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>public kamus(tanabata tb) public void commandAction(Command cmd, Displayable screen) private String findWord(String kata, boolean ji) private String findWord(String kata, String file) private int binarySearchWord(int low, int high, String word) private int searchLastWord(int l, String word) private int cariAwal(int awal) private int cariAkhir(int akhir) private int compare(byte a[0..*], int awal, int akhir, String kata) private int cariFirstLast(String a[0..*], String kata) private int cariIndexFirstLast(String a[0..*], String kata)</pre> </td> </tr> </tbody> </table>	kamus	<i>Attributes</i>	<pre>private Command kl public Form f private TextField text private TextField arti1 private TextField arti2 private TextField pesan private Command submit = new Command("Translate", Command.OK, 2) private Alert alert private byte array1 [0..*] = new byte[10000] private byte temp[0..*] = new byte[1000] private int begin private int length private String firstlastj[0..*] = new String[42] private String firstlastij[0..*] = new String[42]</pre>	<i>Operations</i>	<pre>public kamus(tanabata tb) public void commandAction(Command cmd, Displayable screen) private String findWord(String kata, boolean ji) private String findWord(String kata, String file) private int binarySearchWord(int low, int high, String word) private int searchLastWord(int l, String word) private int cariAwal(int awal) private int cariAkhir(int akhir) private int compare(byte a[0..*], int awal, int akhir, String kata) private int cariFirstLast(String a[0..*], String kata) private int cariIndexFirstLast(String a[0..*], String kata)</pre>
kamus						
<i>Attributes</i>						
<pre>private Command kl public Form f private TextField text private TextField arti1 private TextField arti2 private TextField pesan private Command submit = new Command("Translate", Command.OK, 2) private Alert alert private byte array1 [0..*] = new byte[10000] private byte temp[0..*] = new byte[1000] private int begin private int length private String firstlastj[0..*] = new String[42] private String firstlastij[0..*] = new String[42]</pre>						
<i>Operations</i>						
<pre>public kamus(tanabata tb) public void commandAction(Command cmd, Displayable screen) private String findWord(String kata, boolean ji) private String findWord(String kata, String file) private int binarySearchWord(int low, int high, String word) private int searchLastWord(int l, String word) private int cariAwal(int awal) private int cariAkhir(int akhir) private int compare(byte a[0..*], int awal, int akhir, String kata) private int cariFirstLast(String a[0..*], String kata) private int cariIndexFirstLast(String a[0..*], String kata)</pre>						
kanji	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; padding: 5px;">kanji</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 2px 5px;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>private Command kl public Form f private StringItem siAbout</pre> </td> </tr> <tr> <td style="text-align: center; padding: 2px 5px;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;"> <pre>public kanji(tanabata tb) public StringItem getStringItem() private String kanji() public void commandAction(Command cmd, Displayable screen)</pre> </td> </tr> </tbody> </table>	kanji	<i>Attributes</i>	<pre>private Command kl public Form f private StringItem siAbout</pre>	<i>Operations</i>	<pre>public kanji(tanabata tb) public StringItem getStringItem() private String kanji() public void commandAction(Command cmd, Displayable screen)</pre>
kanji						
<i>Attributes</i>						
<pre>private Command kl public Form f private StringItem siAbout</pre>						
<i>Operations</i>						
<pre>public kanji(tanabata tb) public StringItem getStringItem() private String kanji() public void commandAction(Command cmd, Displayable screen)</pre>						

kenal	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">kenal</th> </tr> <tr> <td style="padding: 5px;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;">private Command kl public Form f private StringItem siAbout</td> </tr> <tr> <td style="padding: 5px;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;">public kenal(tanabata tb) public StringItem getStringItem() private String kenalan() public void commandAction(Command cmd, Displayable screen)</td> </tr> </table>	kenal	<i>Attributes</i>	private Command kl public Form f private StringItem siAbout	<i>Operations</i>	public kenal(tanabata tb) public StringItem getStringItem() private String kenalan() public void commandAction(Command cmd, Displayable screen)
kenal						
<i>Attributes</i>						
private Command kl public Form f private StringItem siAbout						
<i>Operations</i>						
public kenal(tanabata tb) public StringItem getStringItem() private String kenalan() public void commandAction(Command cmd, Displayable screen)						
kerja	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">kerja</th> </tr> <tr> <td style="padding: 5px;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;">private Command kl public Form f private StringItem siAbout</td> </tr> <tr> <td style="padding: 5px;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;">public kerja(tanabata tb) public StringItem getStringItem() private String kerjaan() public void commandAction(Command cmd, Displayable screen)</td> </tr> </table>	kerja	<i>Attributes</i>	private Command kl public Form f private StringItem siAbout	<i>Operations</i>	public kerja(tanabata tb) public StringItem getStringItem() private String kerjaan() public void commandAction(Command cmd, Displayable screen)
kerja						
<i>Attributes</i>						
private Command kl public Form f private StringItem siAbout						
<i>Operations</i>						
public kerja(tanabata tb) public StringItem getStringItem() private String kerjaan() public void commandAction(Command cmd, Displayable screen)						
negara	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">negara</th> </tr> <tr> <td style="padding: 5px;"><i>Attributes</i></td> </tr> <tr> <td style="padding: 5px;">private Command kl public Form f private StringItem siAbout</td> </tr> <tr> <td style="padding: 5px;"><i>Operations</i></td> </tr> <tr> <td style="padding: 5px;">public negara(tanabata tb) public StringItem getStringItem() private String negara() public void commandAction(Command cmd, Displayable screen)</td> </tr> </table>	negara	<i>Attributes</i>	private Command kl public Form f private StringItem siAbout	<i>Operations</i>	public negara(tanabata tb) public StringItem getStringItem() private String negara() public void commandAction(Command cmd, Displayable screen)
negara						
<i>Attributes</i>						
private Command kl public Form f private StringItem siAbout						
<i>Operations</i>						
public negara(tanabata tb) public StringItem getStringItem() private String negara() public void commandAction(Command cmd, Displayable screen)						



3.6 Class Diagram

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi dan lain-lain untuk aplikasi pembelajaran bahasa jepang ini sendiri. *Class Diagram* dapat dilihat pada Gambar 3.6.

BAB IV IMPLEMENTASI DAN EVALUASI SISTEM

4.1 Langkah Langkah Pembuatan Sistem

4.1.1 Penggunaan TextField untuk Inputan

TextField adalah sebuah objek untuk memasukkan inputan berupa teks ke dalam form masukan. Memang ada kesamaan serta perbedaan antara TextBox dan TextField. Namun, yang paling mendasar adalah TextBox yang merupakan subkelas dari *Screen* sedangkan TextField adalah subkelas Item dan pada saat ditampilkan, TextField bisa bersama Item yang lainnya sedangkan TextBox hanya dapat ditampilkan sendirian. Persamaannya adalah keduanya menggunakan batasan (*constraint*) yang sama dan keduanya memiliki kapasitas maksimum.

Untuk Implementasi TextField sendiri dapat dilihat dari source dibawah :

```
public kamus(tanabata tb){
    t = tb;
    kl = new Command("Kembali", Command.EXIT, 8);
    f = new Form("Kamus");
    text = new TextField("Masukan Kata Jepang/Indonesia", "", 50,
    TextField.ANY);
    arti1 = new TextField("Jepang - Indonesia", "", 500,
    TextField.UNEDITABLE);
    arti2 = new TextField("Indonesia - Jepang", "", 500,
    TextField.UNEDITABLE);
    pesan = new TextField("Maaf", "kata tidak ditemukan", 500,
    TextField.UNEDITABLE);
```

4.1.2 Penggunaan File txt untuk media penyimpanan.

File txt dipilih karena memiliki memory yang kecil. Untuk Implementasinya sendiri dapat dilihat dari source dibawah :

```
private String findWord(String kata, boolean ji)
{
    int index;
    if(ji){
        index = cariIndexFirstLast(firstlastji, kata);
        if(index>=0){
            return findWord(kata, "/kamus/kamusji"+index+".txt");
        }
    }else {
        index = cariIndexFirstLast(firstlastij, kata);
        if(index>=0){
            return findWord(kata, "/kamus/kamusij"+index+".txt");
        }
    }
    return null;
}
```

4.1.3 Penggunaan List untuk Menu

List menyediakan fungsi memilih elemen dalam List. Elemen tersebut dapat berupa teks, *string*, ataupun gambar. Komponen List mengimplementasikan antarmuka (interface) Choice yang mendefinisikan tiga tipe yang dapat diimplementasikan pada sebuah List. Untuk menu dalam aplikasi ini menggunakan Choice.IMPLICIT.

Untuk Implementasi List sendiri dapat dilihat dari source dibawah :

```
public tanabata() {
    display = Display.getDisplay(this);
}
```

```
try {
    img = Image.createImage("/image/a.png");
} catch (java.io.IOException e) {
}

cmBack = new Command("Kembali", Command.BACK, 1);
cmExit = new Command("Keluar", Command.EXIT, 1);
list = new List(".:Menu Utama Tanabata.:", List.IMPLICIT);
list.append("Huruf Jepang", img);
list.append("  Hiragana", null);
list.append("  Katakana", null);
list.append("  Kanji", null);
list.append("Tutorial", img);
list.append("  perkenalan", null);
list.append("  menanyakan asal", null);
list.append("  angka", null);
list.append("  tempat penting", null);
list.append("  keterangan waktu", null);
list.append("  pekerjaan", null);
list.append("  tanggal dan hari", null);
list.append("  kebudayaan jepang", null);
list.append("Kamus", img);
list.append("Tentang", img);
list.addCommand(cmBack);
list.addCommand(cmExit);
list.setCommandListener(this);
display.setCurrent(list);
```

4.2 Keperluan Sistem

Untuk pembuatan aplikasi pembelajaran bahasa jepang ini diperlukan perangkat keras dan perangkat lunak sebagai penunjang pembuatan aplikasi adapun keperluan untuk pembuatan aplikasi ini adalah sebagai berikut :

1. Keperluan perangkat lunak untuk pengguna :
 - a. MIDP 2.0
 - b. CLDC 1.0
2. Keperluan pembuatan sistem
 - a. Java SDK 1.6.2 sebagai *java development kit*.
 - b. Sun Java Wireless Toolkit 2.5.2 sebagai *compiler* dari *source code* yang dibuat.
 - c. TextPad sebagai text editor untuk *source code* java.
 - d. Bluetooth sebagai alat untuk mengirim aplikasi ke perangkat handphone.

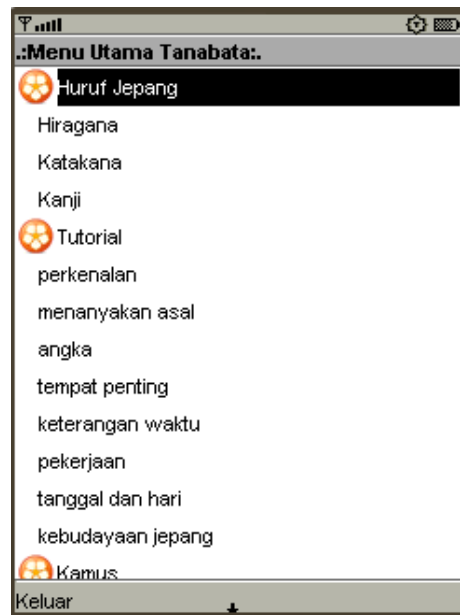
4.3 Hasil Pengujian Sistem

4.3.1 Tampilan Menu Utama

Saat pertama kali dijalankan maka akan muncul tampilan menu utama, dapat dilihat pada Gambar 4.2.



Gambar 4.1 Tampilan Aplikasi Sebelum Dijalankan.



Gambar 4.2 Tampilan Awal Aplikasi.

4.3.2 Tampilan Huruf Hiragana

Dalam menu huruf jepang terdapat tiga menu antara lain hiragana, katakana dan kanji. Ketiga menu tadi akan terhubung dengan *class* yang memunculkan file penjelasan mengenai huruf hiragana, katakana dan kanji, yang dapat dilihat pada Gambar 4.3 dan Gambar 4.4.



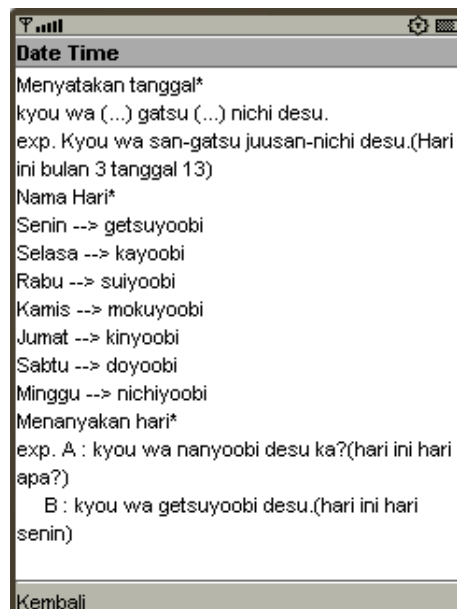
Gambar 4.3. Tampilan Huruf Hiragana.



Gambar 4.4. Tampilan Huruf Katakana

4.3.3 Tampilan Tutorial

Dalam menu tutorial terdapat berbagai macam tutorial misalnya mengenai percakapan dalam bahasa Jepang, tempat penting di Jepang dan kebudayaan – kebudayaan yang ada di Jepang. Menu tutorial memanggil *class* yang akan memunculkan file tutorial seperti pada Gambar 4.5.



Gambar 4.5. Tampilan Tutorial Tanggal dan Hari



Gambar 4.6. Tampilan Tutorial Kebudayaan Jepang

4.3.3 Tampilan Kamus

Dalam kamus Jepang-Indonesia dan Indonesia-Jepang, pengguna dapat menginputkan dalam kata berbahasa Jepang maupun kata yang berbahasa Indonesia seperti pada Gambar 4.7 dan Gambar 4.8.



Gambar 4.7. Tampilan Kamus.



Gambar 4.8. Tampilan Kamus.

Jika ada dua kata yang ditemukan maka akan muncul dua hasil terjemahan dari satu kata tersebut seperti pada Gambar 4.9.



Gambar 4.9. Tampilan Kamus Saat Dua Terjemahan yang Ditemukan.

Jika kata tidak ditemukan maka akan muncul pemberitahuan kata tidak ditemukan seperti pada Gambar 4.10.



Gambar 4.9. Tampilan Kamus Saat Terjemahan Tidak Ditemukan.

4.3 Evaluasi Sistem

4.3.1 Kelemahan dari Aplikasi Pembelajaran Bahasa Jepang

Adapun kelemahan yang masih ada dari aplikasi pembelajaran bahasa Jepang antara lain belum adanya contoh kalimat dalam huruf Jepang. Selain itu juga belum adanya contoh huruf pada penjelasan huruf kanji. Tutorial yang disediakan belum lengkap sehingga belum dapat memuaskan para pengguna.

Pembendaharaan kata pada kamus Jepang-Indonesia maupun Indonesia-Jepang belum begitu banyak sehingga masih banyak kata yang belum dapat diterjemahkan.

4.3.2 Kelebihan dari Aplikasi Pembelajaran Bahasa Jepang

Kelebihan yang ada pada aplikasi pembelajaran bahasa Jepang antara lain aplikasi ini merupakan aplikasi offline sehingga pengguna dapat menggunakan aplikasi ini dimana saja tanpa perlu memperhatikan sinyal GPRS di tempat mereka sedang berada dan pengguna tidak perlu mensetting GPRS pada *handphone*.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan data yang telah dikumpulkan dan analisa terhadap aplikasi pembelajaran bahasa jepang yang telah dapat disimpulkan :

1. Aplikasi Pembelajaran Bahasa Jepang ini dibuat untuk mempermudah peminat bahasa jepang dalam mempelajari bahasa jepang, selain itu penggunaan media Handphone sebagai media pembelajaran membuat peminat bahasa jepang semakin tertarik untuk mempelajari bahasa jepang.
2. Aplikasi ini tidak menggunakan media koneksi GPRS (*General Packet Radio Service*) sehingga aplikasi ini dapat dipelajari dimana saja tanpa perlu memperhatikan koneksi GPRS di wilayahnya.
3. Aplikasi ini berjalan pada ponsel dengan profil MIDP 2.0 dan CLDC 1.0, untuk tipe dibawahnya yaitu MIDP 1.0 tidak bisa.

5.2 Saran

Perlu pengembangan lebih lanjut untuk menu kamus dan tutorial sehingga aplikasi memenuhi keinginan pengguna, pembendaharaan kata untuk kamus lebih diperbanyak lagi, selain itu perlu juga dilakukan pemantauan terhadap memori yang digunakan aplikasi agar tidak terjadi *over flow memory*.

DAFTAR PUSTAKA

- Budi Raharjo, Imam Heryanto, Arif Haryono.2007.*Tuntunan Pemrograman Java untuk Handphone*.Bandung:Penerbit Informatika.
- M. Shalahuddin, Rosa A.S.2008.*Pemrograman J2ME (Belajar Cepat Pemrograman perangkat Telekomunikasi Mobile)*.Bandung:Penerbit Informatika.
- Purnama, Rangsang.2008.*Pemrograman J2ME Tingkat Dasar*.Surabaya:Penerbit Gitamedia Press.
- Setianto, Dwi.2006.*Panduan Lengkap Pemrograman Celluler Device dengan J2ME*.Yogyakarta:Penerbit Ardana Media.
- Siregar, Ivan Michael.2007.*Membangun Aplikasi Chat Lewat GPRS dengan J2ME Menggunakan NetBeans IDE 5.0*.Bandung:Penerbit Gava Media.
- Supardi, Yuniar.2007.*Pemrograman Database dengan Java dan MySQL*.Jakarta:Penerbit PT Elex Media Komputindo, Kelompok Kompas Gramedia.
- Supardi, Yuniar.2008.*Pemrograman Handphone dengan J2ME*.Jakarta:Penerbit PT Elex Media Komputindo.
- Suyoto.2007.*Membuat Sendiri Aplikasi Ponsel*.Bandung:Penerbit Gava Media.