

ABSTRAK

Judul Tugas Akhir : **Aplikasi Pembelajaran Dasar Bahasa Jepang Berbasis Android (Studi Kasus SMAN 2 Sungai Tarab)**

Nama Mahasiswa : **Lina Faizah Fitri**

Nim : **14 205 059**

Jurusan : **Manajemen Informatika**

Fakultas : **Ekonomi dan Bisnis Islam**

Dosen Pembimbing : **Adriyendi, M.Kom**

Bahasa Jepang merupakan salah satu bahasa yang banyak dipelajari. Banyak orang menganggap sulit untuk belajar bahasa Jepang. Padahal, jika bahasa Jepang itu dipelajari dengan cara yang berbeda, akan membuat orang tertarik untuk mempelajarinya.

Saat ini banyak sekolah menengah atas di Indonesia yang mempelajari bahasa Jepang berdasarkan kurikulum standar kompetensi 2004 bahasa Jepang Departemen Pendidikan Nasional. Akan tetapi masih banyak berbagai kalangan mengalami kesulitan mempelajari bahasa Jepang karena bentuk tulisan yang rumit dan tata bahasa sangat jauh berbeda dengan bahasa Indonesia.

Dengan pendekatan teknologi mobile berbasis android diharapkan informasi mengenai bahasa Jepang dapat tersampaikan kepada masyarakat pada umumnya dan siswa khususnya. Melalui aplikasi pembelajaran ini diharapkan akan memajukan dunia pendidikan.

Dalam penulisan tugas akhir ini menggunakan 3 macam metode penelitian yaitu penelitian lapangan yaitu wawancara dan tanya jawab kepada beberapa guru SMAN 2 Sungai Tarab, penelitian perpustakaan yaitu menggunakan buku sumber sebagai bahan acuan dan referensi serta penelitian labor dengan menggunakan pemograman Android Studio.

Kata Kunci : *Bahasa Jepang, Pembelajaran, Android.*

KATA PENGANTAR



Puji syukur penulis haturkan atas kehadiran Allah SWT. yang telah melimpahkan rahmat dan karunia-Nya sehingga pembuatan tugas akhir ini dapat terselesaikan. Shalawat dan salam buat junjungan umat, yakni Nabi Muhammad SAW. yang telah membawa manusia dari alam kebodohan ke alam yang penuh dengan ilmu pengetahuan seperti kita rasakan seperti sekarang ini.

Tugas akhir ini penulis susun untuk memberikan sumbangan pemikiran kepada almamater serta untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Diploma III Jurusan Manajemen Informatika Institut Agama Islam Negeri (IAIN) Batusangkar

Dalam penulisan tugas akhir ini penulis banyak mendapat bimbingan dan bantuan baik moril maupun materil dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. H. Kasmuri, MA selaku ketua IAIN Batusangkar.
2. Bapak Dr. Ulya Atsani, S.H.,M.Hum selaku Dekan Fakultas Ekonomi dan Bisnis Islam Institut Agama Islam Negeri Batusangkar .
3. Bapak Iswandi, M.Kom selaku ketua Jurusan Manajemen Informatika IAIN Batusangkar.
4. Bapak Adriyendi, M.Kom selaku pembimbing yang telah memberikan bimbingan dan arahan kepada penulis dalam pembuatan tugas akhir ini.
5. Seluruh dosen, staf, dan karyawan/ti IAIN Batusangkar.
6. Ibunda tercinta (Alm. Rosnaini), Ayahku tercinta (Alm.Marjunis), dan abang-abang tersayang (Een Abdul Rahman dan Ismail Abdul Salam) yang selalu memberikan semangat, dukungan, nasihat serta do'a yang tiada henti-

hentinya, sehingga penulis mempunyai semangat untuk menyelesaikan tugas akhir ini.

7. Onga dan Pak Onga (Ridwani dan Yunus) yang selalu memberikan doa, dukungan, dan nasihatnya, sehingga penulis bisa menyelesaikan tugas akhir ini
8. Kakak-kakak (Winny Rahma, Sefti Safar Indah, Silfia Rahayu) yang selalu membantu dan membimbing penulis dalam proses menyelesaikan tugas akhir ini.
9. Seluruh teman-teman keluarga besar Jurusan Manajemen Informatika angkatan 2014 yang selalu bersama-sama mengukir kenangan indah selama mengikuti perkuliahan di IAIN Batusangkar.
10. Rekan-rekan yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan motivasi dan semangat serta sumbangan pemikirannya kepada penulis sehingga selesainya tugas akhir ini.

Penulis sadar bahwasanya tugas akhir ini jauh dari kesempurnaan, oleh karena itu kritikan dan saran yang sifatnya membangun sangat penulis harapkan untuk kesempurnaan tugas akhir ini. Penulis juga berharap semoga penulisan tugas akhir ini memberikan manfaat kepada kita semua. Amin...

Akhirnya kepada Allah SWT jua lah Penulis bermohon dan bersujud semoga keikhlasan yang diberikan akan dibalas-Nya. Amin Ya Robbal'alam.

Batusangkar, Februari 2018

Penulis

DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	viii
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	2
C. Rumusan Masalah	2
D. Batasan Masalah.....	2
E. Tujuan Penelitian.....	3
F. Manfaat Penelitian.....	3
G. Metodologi Penelitian	3
H. Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	6
A. Gambaran Umum SMAN 2 Sungai Tarab	6
1. Sejarah Berdirinya SMAN 2 Sungai Tarab.....	6
2. Visi dan Misi SMAN 2 Sungai Tarab	6
3. Struktur Organisasi.....	8
B. Konsep Dasar Pembelajaran.....	8
1. Arti dan Makna Pembelajaran.....	8
2. Konsep Pembelajaran.....	8
C. Aplikasi	9
D. Bahasa Pemrograman Android.....	10
1. Mengetahui Android.....	10
2. Android: <i>Platform</i> Masa Depan	12
3. Versi Android.....	14
4. Fitur-Fitur Android.....	16

5.	Arsitektur Android	17
6.	Android SDK (Software Development Kit).....	20
7.	ADT (Android Development Tools).....	21
8.	Fundamental Aplikasi	22
9.	Android Studio	24
10.	Unified Modelling Language (UML).....	25
	a. Use Case Diagram	25
	b. Activity Diagram	27
	c. Class Diagram.....	29
E.	Konsep Dasar Bahasa Jepang.....	31
	1. Pengertian Bahasa Jepang	31
	2. Jenis-Jenis Huruf Jepang.....	32
	a. Kanji.....	33
	b. Hiragana.....	38
	c. Katakana	42
BAB III ANALISA DAN HASIL.....		44
A.	Analisa Kebutuhan Aplikasi.....	44
	1. Hardware	44
	2. Software	44
B.	Analisa Aplikasi	44
C.	Perancangan Sistem.....	45
	1. Actor.....	45
	2. Use Case Diagram.....	47
	3. Sequence Diagram.....	48
	4. Activity Diagram.....	50
	5. Class Diagram	52
	6. Struktur Program	53
D.	Desain Output Sistem.....	54
	1. Tampilan Menu Utama.....	54
	2. Tampilan Sub Menu Huruf Jepang	54
	3. Tampilan Sub Menu Kosakata	55
	4. Tampilan Sub Menu Percakapan.....	55

5. Tampilan Sub Menu Quiz	56
BAB IV PENUTUP	57
A. Kesimpulan.....	57
B. Saran.....	57

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2. 1 Struktur Organisasi SMAN 2 Sungai Tarab.....	8
Gambar 2. 2 Arsitektur Sistem Operasi Android.....	19
Gambar 2. 3 Android Studio	24
Gambar 3. 1 Rancangan Use Case Diagram	47
Gambar 3. 2 Sequence Diagram Huruf Jepang.....	48
Gambar 3. 3 Sequence Diagram Kosakata.....	48
Gambar 3. 4 Sequence Diagram Percakapan	49
Gambar 3. 5 Sequence Diagram Quiz.....	49
Gambar 3. 6 Activity Huruf Jepang	50
Gambar 3. 7 Activity Kosakata.....	51
Gambar 3. 8 Activiy Percakapan	51
Gambar 3. 9 Activity Quiz	52
Gambar 3. 10 Class Diagram	53
Gambar 3. 11 Struktur Program	53
Gambar 3. 12 Tampilan Menu Utama.....	54
Gambar 3. 13 Tampilan Sub Menu Huruf Jepang	54
Gambar 3. 14 Tampilan Sub Menu Kosakata	55
Gambar 3. 15 Tampilan Sub Menu Percakapan.....	55
Gambar 3. 16 Tampilan Sub Menu Quiz	56

DAFTAR TABEL

Tabel 2. 1 Use Case Diagram (Nugroho:2008).....	26
Tabel 2. 2 Activity Diagram (Nugroho:2008).....	28
Tabel 2. 3 Class Diagram (Rosa dan Shalahuddin: 2013)	30
Tabel 2. 4 Tabel Kanji <i>On-yomi</i>	37
Tabel 2. 5 Tabel Hiragana <i>Sei On</i>	38
Tabel 2. 6 Tabel Hiragana <i>Handaku On</i>	39
Tabel 2. 7 Tabel Hiragana <i>Daku On</i>	39
Tabel 2. 8 Tabel Hiragana <i>Yoo On</i>	40
Tabel 2. 9 Tabel Katakana	43
Tabel 3. 1 Tabel Peran Actor	47

BAB I

PENDAHULUAN

A. Latar Belakang

Bahasa Jepang merupakan salah satu bahasa yang banyak dipelajari. Banyak orang menganggap sulit untuk belajar bahasa Jepang. Padahal, jika bahasa Jepang itu dipelajari dengan cara yang berbeda, akan membuat orang tertarik untuk mempelajarinya.

Saat ini banyak sekolah menengah atas di Indonesia termasuk yang mempelajari bahasa Jepang berdasarkan kurikulum standar kompetensi 2004 bahasa Jepang Departemen Pendidikan Nasional. Akan tetapi masih banyak berbagai kalangan mengalami kesulitan mempelajari bahasa Jepang karena bentuk tulisan yang rumit dan tata bahasa sangat jauh berbeda dengan bahasa Indonesia. (Kusuma: 2015)

Bahasa Jepang menggunakan 3 (tiga) jenis karakter huruf yang berbeda dari karakter huruf yang biasa digunakan di dunia yaitu Alfabeth. Ketiga huruf dalam bahasa Jepang tersebut yaitu *Hiragana* (ひらがな), *Katakana* (カタカナ), dan *Kanji* (かんじ).

Salah satu sistem operasi ponsel yang terbuka dan dapat dikembangkan adalah android. Saat ini telah banyak aplikasi-aplikasi yang dibuat oleh para pengembang untuk meningkatkan dan mendukung kemampuan ponselnya. Kini kemajuan teknologi telah merajai berbagai kalangan dari anak-anak hingga dewasa. Dalam dunia pendidikan telah banyak dikembangkan aplikasi-aplikasi untuk mendukung pembelajaran bagi para pelajar agar lebih bisa menguasai dan paham akan pelajaran tersebut.

Berdasarkan permasalahan diatas, maka penulis tertarik untuk merancang sebuah aplikasi dalam penulisan Tugas Akhir (TA) yang berjudul

“APLIKASI PEMBELAJARAN DASAR BAHASA JEPANG BERBASIS ANDROID (STUDI KASUS SMAN 2 SUNGAI TARAB)”.

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, maka penulis dapat mengidentifikasi beberapa masalah. Diantaranya yaitu:

1. Sulitnya mempelajari bahasa Jepang karena bentuk tulisan yang rumit dan tata bahasa sangat jauh berbeda dengan bahasa Indonesia.
2. Belum ada aplikasi yang digunakan di SMAN 2 Sungai Tarab.

C. Rumusan Masalah

Berdasarkan dari latar belakang diatas maka penulis merumuskan permasalahannya yaitu bagaimana membuat suatu aplikasi bahasa Jepang menggunakan android?

D. Batasan Masalah

Agar penelitian tidak mengambang dan terarah kepada pokok permasalahan, maka penulis membatasi batasan terhadap masalah-masalah tersebut, adapun batasan-batasan masalah tersebut adalah sebagai berikut:

1. Aplikasi pembelajaran dasar Bahasa Jepang berbasis Android meliputi huruf Jepang, kosakata, percakapan, dan kuis
2. Aplikasi pembelajaran berbasis Android ini bisa dijalankan mulai dari versi android 4.0.
3. Aplikasi pembelajaran dasar Bahasa Jepang yang dibangun bersifat online dan offline.
4. Aplikasi pembelajaran dasar Bahasa Jepang yang dibangun tidak menggunakan huruf kanji.

E. Tujuan Penelitian

Dalam penelitian ini, penulis memiliki beberapa tujuan yang ingin di capai. Adapun tujuan tersebut adalah:

1. Untuk membuat aplikasi pembelajaran dasar Bahasa Jepang berbasis Android.
2. Sebagai alat bantu untuk memudahkan proses pembelajaran di SMAN 2 Sungai Tarab.

F. Manfaat Penelitian

Selain mempunyai tujuan, penelitian ini juga diharapkan dapat memberi manfaat terutama kepada peneliti sendiri, admin, dan pihak sekolah yang nantinya akan menggunakan aplikasi pembelajaran dasar Bahasa Jepang berbasis Android ini. Adapun manfaat tersebut adalah:

1. Menambah ilmu pengetahuan tentang aplikasi pembelajaran dasar Bahasa Jepang berbasis Android.
2. Dengan adanya aplikasi pembelajaran dasar Bahasa Jepang ini memudahkan proses pembelajaran di SMAN 2 Sungai Tarab.
3. Sebagai tambahan referensi bagi pembaca yang akan melakukan penelitian lebih lanjut.
4. Sebagai persyaratan untuk memperoleh gelar Ahli Madya Program Diploma III (D.3) Jurusan Manajemen Informatika pada IAIN Batusangkar.

G. Metodologi Penelitian

1. Studi Pustaka (*Library Research*)

Mempelajari buku, artikel dan situs internet serta referensi lain yang terkait dengan aplikasi pembelajaran dasar Bahasa Jepang berbasis android.

2. Penelitian Lapangan (*Field Research*)

Adalah suatu teknik penelitian yang dilakukan secara langsung pada objek penelitian, dimana untuk memperoleh data tersebut penulis menggunakan beberapa teknik yaitu:

a. Teknik Wawancara (*interview*)

Teknik wawancara (*interview*) yaitu dengan mengadakan wawancara langsung dengan pihak yang berwenang untuk mendapatkan penjelasan terhadap masalah yang menjadi objek pengamatan dan pembahasan.

b. Teknik Pengamatan (*observation*)

Teknik pengamatan (*observation*) yaitu dengan mengadakan peninjauan langsung terhadap pelaksanaan kegiatan yang menjadi data pembahasan.

3. Penelitian Labor (*Laboratory Research*)

Dalam penelitian ini penulis melakukan pengolahan data dengan menggunakan komputer sebagai alat bantu dalam pembuatan tugas akhir ini.

H. Sistematika Penulisan

Sistematika penulisan dapat diartikan sebagai cara yang digunakan dalam pembuatan laporan untuk memberikan gambaran isi tugas akhir yang terdiri dari pendahuluan, landasan teori, analisa dan hasil serta penutup.

Sistematika penulisan laporan tugas akhir ini dibagi dalam IV bab yang disusun sebagai berikut:

Bab I Pendahuluan, Bab ini merupakan penguraian mengenai latar belakang masalah, identifikasi masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

Bab II Landasan Teori, Bab ini berisi teori-teori dasar mengenai Aplikasi Pembelajaran Dasar Bahasa Jepang Berbasis Android (Studi Kasus SMAN 2 Sungai Tarab).

Bab III Analisa dan Hasil, Bab ini membahas analisa dan perancangan Aplikasi Pembelajaran Dasar Bahasa Jepang Berbasis Android (Studi Kasus SMAN 2 Sungai Tarab).

Bab IV Penutup, Bab ini berisi Kesimpulan yang didapat selama pembuatan laporan tugas akhir serta saran-saran yang akan menjadi masukan bagi perkembangan sistem selanjutnya.

BAB II

LANDASAN TEORI

A. Gambaran Umum SMAN 2 Sungai Tarab

1. Sejarah Berdirinya SMAN 2 Sungai Tarab

SMAN 2 Sungai Tarab berdiri pada bulan Juli tahun 2000. Kepala Sekolah pertamanya yaitu Bapak Drs. Sukardiman beliau menjabat hanya 2 (dua) tahun. Setelah itu digantikan dengan Bapak Drs. Yulisman sampai tahun 2009. Lalu Bapak Yulisman digantikan dengan Bapak Niviko. Beliau menjabat sampai tahun 2012. Setelahnya Bapak Niviko digantikan dengan Bapak Drs. Edial Yuspita, MM. Beliau menjabat sebagai kepala sekolah sampai dengan saat ini. Sudah 17 tahun SMAN 2 Sungai Tarab berdiri.

2. Visi dan Misi SMAN 2 Sungai Tarab

a. Visi

“Cerdas, Berprestasi, Berakhlak Mulia, dan Berwawasan Lingkungan”

b. Misi

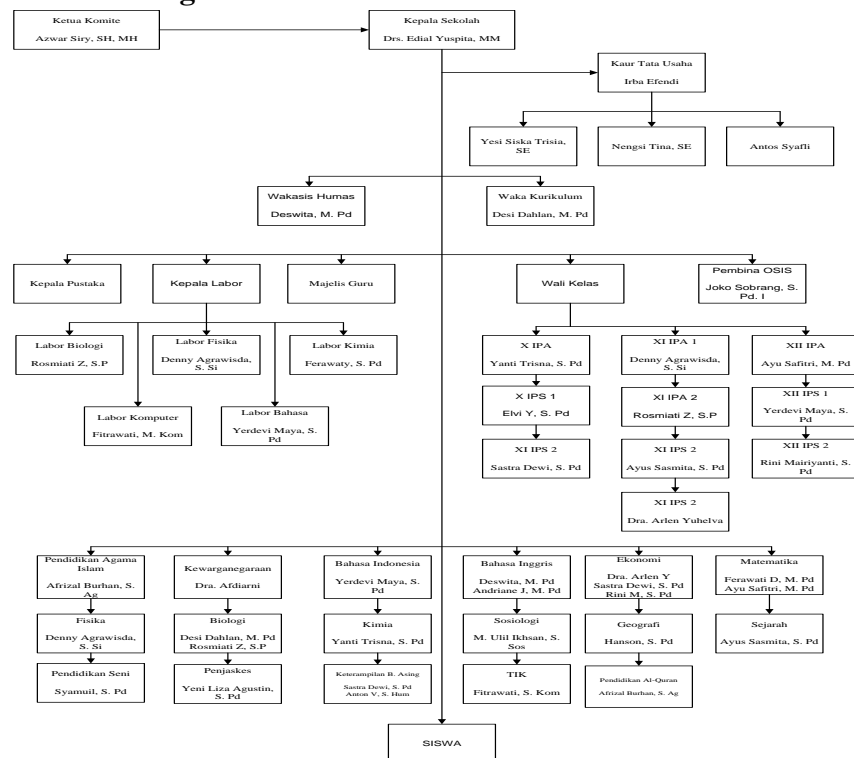
Misi SMAN 2 Sungai Tarab berdasarkan visi yang telah ditetapkan.

1) Mewujudkan warga sekolah berakhlak mulia

- a) Mengembangkan perilaku yang mencerminkan insan beragama dan mengintegritaskan dalam setiap PBM.
- b) Mengintensifkan kegiatan yang bernuansa islami dengan melibatkan seluruh warga sekolah.
- c) Mengaplikasikan kurikulum nasional yang berbasis pada pendidikan dan budaya bangsa.
- d) Memberikan pelayanan prima dalam segala kegiatan setiap saat.

- 2) Mewujudkan warga sekolah yang berprestasi
 - a) Melaksanakan pembelajaran dan bimbingan secara efektif dengan menggunakan multimedia.
 - b) Meningkatkan pengetahuan, keterampilan, dan apresiasi peserta didik di bidang ilmu pengetahuan, teknologi, sosial budaya, dan seni dengan berpikir kritis, kreatif, kolaboratif, dan komunikatif.
 - c) Mengintensifkan bimbingan atau pelatihan secara berkelanjutan dengan optimalisasi penggunaan laboratorium, Green House, dan kerjasama berbagai pihak agar siswa mandiri, mampu bersaing dan mengikuti jenjang pendidikan yang lebih tinggi.
 - d) Melaksanakan event-event tertentu dalam berbagai event yang diikuti.
- 3) Berwawasan lingkungan
 - a) Mengembangkan kurikulum berbasis lingkungan.
 - b) Mewujudkan kultur sekolah yang berwawasan lingkungan untuk menunjang kenyamanan proses pembelajaran.
 - c) Menjadikan lingkungan sekolah yang bersih, indah, terpelihara, dan lestari dalam rangka mempertahankan sekolah adiwiyata.

3. Struktur Organisasi



Gambar 2. 1 Struktur Organisasi SMAN 2 Sungai Tarab

B. Konsep Dasar Pembelajaran

1. Arti dan Makna Pembelajaran

Pembelajaran menurut Syaiful Sagala (2003) ialah membelajarkan siswa menggunakan asas pendidikan maupun teori belajar merupakan penentu utama keberhasilan pendidikan. Pembelajaran merupakan proses komunikasi dua arah, mengajar dilakukan oleh pihak guru sebagai pendidik, sedangkan belajar dilakukan oleh peserta didik atau murid.

2. Konsep Pembelajaran

Pembelajaran mengandung arti setiap kegiatan yang dirancang untuk membantu seseorang mempelajari suatu kemampuan dan atau nilai yang baru. Pembelajaran mempunyai dua karakteristik yaitu *Pertama*, dalam proses pembelajaran

melibatkan proses mental siswa secara maksimal, bukan hanya menuntut siswa dalam proses berfikir. **Kedua**, dalam pembelajaran membangun suasana dialogis dan proses tanya jawab terus menerus yang diarahkan untuk memperbaiki dan meningkatkan kemampuan berfikir siswa, yang pada gilirannya kemampuan berfikir itu dapat membantu siswa memperoleh pengetahuan yang mereka konstruksi sendiri.

Proses pembelajaran menurut Dunkin dan Biddle dalam buku Syaiful Sagala (2003) berada pada empat variabel interaksi yaitu (1) variabel pertanda (*presage variables*) berupa pendidik; (2) konteks (*context variables*) berupa interaksi peserta didik dengan pendidik; dan (4) variabel produk (*product variables*) berupa perkembangan peserta didik dalam jangka pendek maupun jangka panjang.

Proses pembelajaran aktivitasnya dalam bentuk interaksi belajar mengajar dalam suasana interaksi edukatif, yaitu interaksi yang sadar akan tujuan, artinya interaksi yang telah dicanangkan untuk suatu tujuan tertentu setidaknya adalah pencapaian tujuan instruksional atau tujuan pembelajaran yang telah dirumuskan pada suatu pelajaran.

C. Aplikasi

Secara istilah pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Menurut kamus computer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. (Juansyah: 2015)

Pengertian aplikasi menurut Kamus Besar Bahasa Indonesia, “Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu”

D. Bahasa Pemrograman Android

1. Mengetahui Android

Menurut mAster.com (2012) Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak.

Menurut Safaat (2014) android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware*, dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia.

Pada saat perilis perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan open source pada perangkat mobile. Di lain pihak, Google merilis kodekode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan open platform perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama, yang mendapat dukungan penuh dari Google atau Google Mail Services (GMS) dan kedua adalah yang benar-

benar bebas distribusinya, tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD).

Sekitar September 2007 Google mengenalkan Nexus One, salah satu jenis smartphone yang menggunakan Android sebagai sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka, Android, perangkat mobile yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru. Pada masa saat ini kebanyakan vendor-vendor smartphone sudah memproduksi *smart phone* berbasis android, vendor-vendor itu antara lain HTC, Motorola, Samsung, LG, HKC, Huawei, Archos, Webstation Camangi, Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, LG, Acer, Philips, T-Mobile, Nexian, IMO, Asus dan masih banyak lagi vendor smartphone di dunia yang memproduksi android. Hal ini karena android itu adalah sistem operasi yang open source sehingga bebas didistribusikan dan dipakai oleh vendor manapun. Tidak hanya menjadi sistem operasi di smartphone, saat ini android menjadi pesaing utama dari Apple pada sistem operasi Tablet PC. Pesatnya pertumbuhan android selain faktor yang disebutkan di atas adalah 36 karena android itu sendiri adalah platform yang sangat lengkap baik itu sistem operasinya, aplikasi dan Tool Pengembangan, Market aplikasi android serta dukungan yang sangat tinggi dari komunitas *Open Source* di dunia, sehingga android terus

berkembang pesat baik dari segi teknologi maupun dari segi jumlah *device* yang ada di dunia.

2. Android: Platform Masa Depan

Android dipuji sebagai “*platform mobile* pertama yang lengkap, terbuka dan bebas”.

- a. Lengkap (*Complete Platform*). Para desainer dapat melakukan pendekatan yang komprehensif ketika mereka sedang mengembangkan platform android. Android merupakan sistem operasi yang aman dan banyak menyediakan tools dalam membangun software dan memungkinkan untuk peluang pengembangan aplikasi.
- b. Terbuka (*Open Source Platform*). Platform android disediakan melalui lisensi open source. Pengembang dapat dengan bebas untuk mengembangkan aplikasi. Android sendiri menggunakan Linux Kernel 2.6.
- c. *Free (Free Platform)*. Android adalah *platform/aplikasi* yang bebas untuk *develope*. Tidak ada liseni atau biaya royalti untuk dikembangkan pada *platform* android. Tidak ada biaya keanggotaan diperlukan. Tidak diperlukan biaya pengujian. Tidak ada kontrak yang diperlukan. Aplikasi untuk android dapat didistribusikan dan diperdagangkan dalam bentuk apapun.
- d. Android merupakan generasi baru *platform mobile, platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari android dilisensikan di bawah GNU, General Public Lisensi Versi 2 (GPLv2) yang sering dikenal dengan istilah “*copyleft*” lisensi dimana setiap perbaikan pihak ketiga harus terus jatuh di bawah terms. Android didistribusikan di bawah lisensi Apache Software

(ASL/Apache2) yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen handset khususnya) dapat memilih untuk meningkatkan platform tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apapun yang mereka inginkan. Pengembang aplikasi android diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan.

Pengembang memiliki beberapa pilihan ketika membuat aplikasi yang berbasis android. Kebanyakan pengembang menggunakan Eclipse yang tersedia secara bebas untuk merancang dan mengembangkan aplikasi android. Eclipse adalah IDE yang paling populer untuk pengembangan android, karena memiliki android plug-in yang tersedia untuk memfasilitasi pengembangan android. Selain itu Eclipse juga mendapat dukungan langsung dari Google untuk menjadi IDE pengembangan aplikasi android, ini terbukti dengan adanya penambahan plugins untuk eclipse untuk membuat project android dimana *source software* langsung dari situs resminya Google. Tetapi hal diatas tidak menutup kemungkinan untuk menggunakan IDE yang lain seperti Netbeans untuk melakukan pengembangan android.

Aplikasi android dapat dikembangkan pada sistem operasi berikut:

1. Windows XP, Vista/ Seven
2. Mac OSX (Mac OS X 10.4.8 atau lebih baru)
3. Linux

3. Versi Android

Telepon pertama yang memakai sistem operasi android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua vendor seluler didunia menggunakan android sebagai operating system. Adapun versi-versi android yang pernah dirilis adalah sebagai berikut:

a. Android versi 1.1

Pada 9 Maret 2009, Google merilis android versi 1.1. android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam, alarm, *voice search* (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email.

b. Android versi 1.5 (*Cupcake*)

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 (*Cupcake*). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengupload video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar dan keyboard pada layar yang dapat disesuaikan dengan sistem.

c. Android versi 1.6 (*Donut*)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang diintegrasikan, CDMA/EVDO, 802.1x, VPN, Gestures dan Text-to-speech engine, kemampuan

dial kontak, teknologi *text to change speech* (tidak tersedia pada semua ponsel, pengadaan resolusi VWGA).

d. Android versi 2.0/2.1 (*Eclair*)

Pada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (*Eclair*), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan *flash* untuk kamera 3.2 MP, digital zoom, dan Bluetooth 2.1.

e. Android versi 2.2 (*Froyo*)

Pada bulan Mei 2010 Android versi 2.2 Rev 1 diluncurkan. Android inilah yang sekarang sangat banyak beredar di pasaran, salah satunya adalah dipakai di Samsung FX tab yang sudah ada di pasaran. Fitur yang tersedia di Android versi ini sudah kompleks diantaranya adalah:

- 1) Kerangka aplikasi memungkinkan penggunaan dan penghapusan komponen yang tersedia
- 2) *Dalvik Virtual Machine* dioptimalkan untuk perangkat media
- 3) Grafik: grafik 2D dan grafis 3D berdasarkan libraries OpenGL
- 4) SQLite: untuk penyimpanan data
- 5) Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG,GIF)
- 6) GSM, Bluetooth, EDGE, 3G, dan Wifi (hardware independent)
- 7) Kamera, *Global Positioning System* (GPS), kompas dan *accelerometer* (tergantung hardware)

f. Android versi 2.3 (Gingerbread)

Android versi 2.3 diluncurkan pada Desember 2010, hal-hal yang direvisi dari versi sebelumnya adalah kemampuan seperti berikut:

- 1) *SIP-based VoIP*
- 2) *Near Field Communications (NFC)*
- 3) *Gyroscope dan sensor*
- 4) *Multiple cameras support*
- 5) *Mixable audio effects*
- 6) *Download Manager*

g. Android versi 3.0 (Honeycomb) dirilis Februari 2011 sebagai android 3.0 revisi 1 serta android versi 3.0 revision 2 telah dirilis pada Juli 2011.

h. Android versi 3.1 dirilis Mei 2011, sedangkan android 3.1 revisi 2 juga dirilis Mei 2011, serta android 3.1 revision 3 dirilis pada Juli 2011.

i. Android versi 3.2 dirilis Juli 2011.

j. Android versi 4.0 dirilis November 2011.

k. Android versi 4.1

l. Android versi 4.2

m. Android versi 4.3 Android versi 3.0 ke atas adalah generasi *platform* yang digunakan untuk tablet pc. Sementara versi 4.0 sudah merupakan *platform* yang bisa dipakai di smartphone dan tablet pc.

4. Fitur-Fitur Android

Android adalah sebuah sistem operasi mobile open source dan dapat dimanufaktur untuk dikustomisasi sehingga tidak ada konfigurasi yang pasti mengenai software dan hardware-nya. Akan tetapi, secara garis besar android sendiri mendukung fitur-fitur berikut ini menurut Wahana Komputer (2013:3) :

- a. Storage: Mendukung SQLite. SQLite adalah sebuah database relational lite (versi ringan) yang digunakan untuk penyimpanan data.
- b. Konektivitas: Mendukung koneksi GSM/EDGE, IDEN, CDMA, EVDO, UMTS, Bluetooth, Wifi, LTE, dan Wimax.
- c. Messaging: Mendukung SMS dan MMS.
- d. Web Browser: Web browser yang digunakan adalah browser berbasis open source Webkit, dengan engine java script Chrome V8.
- e. Media Support: Dukungan media meliputi file media bertipe H.263, H.264 (dalam bentuk 3GP atau MP4 container), MPEG-4 SP, AMR, AMR-WB, (3GP container), AAC, HE-AAC (dalam bentuk MP4 atau 3GP container), MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP.
- f. Dukungan Hardware: Sensor accelerometer, Camera, Kompas digital, Sensor proximity, dan GPS (Global Positioning System).
- g. Multi Touch: Mendukung layar dengan dukungan multi touch.
- h. Multitasking: Kemampuan untuk melaksanakan tugas secara bersamaan.
- i. Flash: Mendukung animasi Flash.
- j. Tethering: Mendukung berbagi koneksi internet.

5. Arsitektur Android

Secara garis besar Arsitektur Android dapat dijelaskan dan digambarkan sebagai berikut:

a. *Application and Widgets*

Application and Widgets ini adalah layer dimana kita berhubungan dengan aplikasi saja, dimana biasanya kita *download* aplikasi kemudian kita lakukan instalasi dan jalankan aplikasi tersebut. Di layer terdapat aplikasi inti termasuk klien

email, program SMS, kalender, peta, browser, kontak dan lain-lain. Semua aplikasi ditulis menggunakan bahasa pemrograman Java.

b. *Application Frameworks*

Android adalah “*Open Development Platform*” yaitu Android menawarkan kepada pengembang atau memberi kemampuan kepada pengembang untuk membangun aplikasi yang bagus dan inovatif. Pengembang bebas untuk mengakses perangkat keras, akses informasi *resources*, menjalankan *service background*, mengatur alarm, dan menambahkan status *notifications*, dan sebagainya. Pengembang memiliki akses penuh menuju API framework seperti yang dilakukan oleh aplikasi yang kategori inti. Arsitektur aplikasi dirancang supaya kita dengan mudah dapat menggunakan kembali komponen yang sudah digunakan (*reuse*).

Sehingga bisa kita simpulkan *Application Frameworks* ini adalah layer dimana para pembuat aplikasi melakukan pengembangan / pembuatan aplikasi yang akan dijalankan di sistem operasi Android, karena pada *layer* inilah aplikasi dapat dirancang dan dibuat, seperti *content-providers* yang berupa sms dan panggilan telepon. Komponen-komponen yang termasuk di dalam *Applications Frameworks* adalah sebagai berikut:

- 1) *Views*
- 2) *Content Provider*
- 3) *Resource Manager*
- 4) *Notification Manager*
- 5) *Activity Manager*

c. *Libraries*

Libraries ini adalah layer dimana fitur-fitur Android berada, biasanya para pembuat aplikasi mengakses *libraries* untuk

menjalankan aplikasinya. Berjalan di atas kernel, layer ini meliputi berbagai library C/C++ inti seperti Libc dan SSL, serta:

- 1) Libraries media untuk pemutaran media audio dan video.
- 2) Libraries untuk manajemen tampilan.
- 3) Libraries Graphics mencakup SGL dan OpenGL untuk grafis 2D dan 3D.
- 4) Libraries SQLite untuk dukungan database.
- 5) Libraries SSL dan Webkit terintegrasi dengan web browser dan security.
- 6) Libraries LiveWebcore mencakup modern web browser dengan engine embeded web view.
- 7) Libraries 3D yang mencakup implementasi OpenGL ES 1.0 API's.

Untuk lebih memahami mengenai bagaimana cara kerja android, Anda dapat melihat arsitektur android yang ditujukan pada gambar 2. 2.



Gambar 2. 2 Arsitektur Sistem Operasi Android

6. Android SDK (Software Development Kit)

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada platform android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang direlease oleh Google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi-netral, Android memberi anda kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *handphone/smartphone*. Beberapa fitur-fitur Android yang paling penting adalah:

- a. *Framework* aplikasi yang mendukung penggantian komponen dan *reusable*.
- b. Mesin *Virtual Dalvik* dioptimalkan untuk perangkat mobile.
- c. Integrated browser berdasarkan *engine open source* Webkit.
- d. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasarkan spesifikasi OpenGL ES 1,0 (Opsional akselerasi hardware).
- e. SQLite untuk penyimpanan data.
- f. Media Support yang mendukung audio, video, dan gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG,GIF), GSM Telephony (tergantung hardware).
- g. Bluetooth, EDGE, 3G, dan Wifi (tergantung hardware).
- h. Kamera, GPS, kompas, dan accelerometer (tergantung hardware).
- i. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plugin untuk IDE Eclipse.

7. ADT (Android Development Tools)

Android Development Tools (ADT) adalah *plugin* yang didesain untuk IDE Eclipse yang memberikan kita kemudahan dalam mengembangkan aplikasi android dengan menggunakan Eclipse IDE. Dengan menggunakan ADT untuk Eclipse akan memudahkan kita dalam membuat aplikasi project android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya, begitu juga kita dapat melakukan running aplikasi menggunakan Android SDK melalui eclipse, dengan ADT juga kita dapat melakukan pembuatan package android (.apk) yang digunakan untuk distribusi aplikasi android yang kita rancang.

Mengembangkan aplikasi android dengan menggunakan ADT di eclipse sangat dianjurkan dan sangat mudah untuk memulai mengembangkan aplikasi android. Berikut ini adalah versi ADT untuk eclipse yang sudah dirilis:

- a. ADT 12.00 (Juli 2011)
- b. ADT 11.00 (Juni 2011)
- c. ADT 10.0.1 (Maret 2011)
- d. ADT 10.0.0 (Februari 2011)
- e. ADT 9.0.0 (Januari 2011)
- f. ADT 8.0.1 (Desember 2010)
- g. ADT 8.0.0 (Desember 2010)
- h. ADT 0.9.9 (September 2010)
- i. ADT 0.9.8 (September 2010)
- j. ADT 0.9.7 (Mei 2010)
- k. ADT 0.9.6 (Maret 2010)
- l. ADT 0.9.5 (Desember 2009)
- m. ADT 0.9.4 (Oktober 2009)

Semakin tinggi platform android yang kita gunakan, dianjurkan menggunakan ADT yang lebih terbaru, karena biasanya munculnya platform baru diikuti oleh munculnya versi ADT yang terbaru.

Untuk melakukan instalasi ADT di eclipse dapat dilakukan secara online maupun offline.

8. Fundamental Aplikasi

Aplikasi android ditulis dalam bahasa pemrograman java. Kode java dikompilasi bersama dengan data *file resource* yang dibutuhkan oleh aplikasi, dimana prosesnya dipackage oleh tools yang dinamakan “apt tools” ke dalam paket android sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat diinstall di perangkat seluler.

Ada enam jenis komponen pada aplikasi android yaitu:

a. *Activities*

Suatu activity akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi android bisa jadi hanya memiliki satu activity, tetapi umumnya aplikasi memiliki banyak activity tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Satu activity biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai *user interface* (UI) saat aplikasi diperlihatkan kepada user. Untuk pindah dari satu activity ke activity lain kita dapat melakukannya dengan satu event, misalnya *clik* tombol, memilih opsi atau menggunakan triggers tertentu. Secara hirarki sebuah *windowsactivity* dinyatakan dengan *methodActivity setContentView().ContentView* adalah objek yang berada pada root hirarki.

b. *Services*

Service tidak memiliki *Graphic User Interface* (GUI), tetapi service berjalan secara background, sebagai contoh dalam memainkan musik, service mungkin memainkan musik atau mengambil data dari jaringan, tetapi setiap service harus

berada dalam kelas induknya. Misalnya, media player sedang memutar lagu dari list yang ada, aplikasi ini akan memiliki dua atau lebih activity yang memungkinkan user untuk memilih lagu misalnya, atau menulis sms sambil player sedang jalan. Untuk menjaga musik tetap di jalankan, activity player dapat menjalankan service. Service dijalankan pada thread utama dari proses aplikasi.

c. *Broadcast Receiver*

Broadcast receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh broadcast seperti notifikasi zona waktu berubah, baterai low, gambar telah selesai diambil oleh camera, atau pengubahan referensi bahasa yang digunakan. Aplikasi juga dapat menginisiasi broadcast misalnya memberikan informasi pada aplikasi lain bahwa ada data yang telah diunduh ke perangkat dan siap untuk digunakan.

Broadcast receiver tidak memiliki *user interface* (UI), tetapi memiliki sebuah activity untuk merespon informasi yang mereka terima, atau mungkin menggunakan *Notification Manager* untuk memberitahu kepada pengguna, seperti lampu latar atau *vibrating* (getaran) perangkat, dan lain sebagainya.

d. *Content Provider*

Content Provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam file sistem seperti database SQLite. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu activity, misalnya ketika kita menggunakan aplikasi yang membutuhkan peta (Map), atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi, maka disinilah fungsi *content provider*.

9. Android Studio

Android studio menurut Juansyah (2015) adalah IDE (Integrated Development Environment) resmi untuk pengembangan aplikasi Android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 Mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android.



Gambar 2. 3 Android Studio

Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT (*Android Development Tools*) plugin. Android studio memiliki fitur :

- a. Projek berbasis pada Gradle Build.
- b. Refactory dan pembenahan bug yang cepat.
- c. Tools baru yang bernama “Lint” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung Proguard And App-signing untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah.
- f. Didukung oleh Google Cloud Platfrom untuk setiap aplikasi yang dikembangkan.

10. Unified Modelling Language (UML)




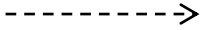
Nugroho (2005) berpendapat bahwa UML, merupakan bahasa untuk visualisasi, spesifikasi, konstruksi, serta dokumentasi. Sependapat dengan, Dharwiyanti (2003) yang menjelaskan UML seperti sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.


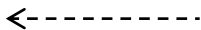
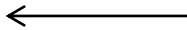
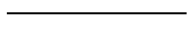
UML merupakan pemodelan berorientasi objek dalam merancang suatu sistem, akan tetapi dapat digunakan untuk pemodelan aplikasi prosedural. Pernyataan tersebut dikuatkan oleh Dharwiyanti (2003) dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun serta ditulis dalam bahasa pemrograman apapun, karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek.

a. *Use Case Diagram*

Use Case pada dasarnya merupakan unit fungsionalitas koheren yang diekspresikan sebagai transaksi-transaksi yang terjadi antara actor dan system. Kegunaan use case sesungguhnya adalah untuk mendefinisikan suatu bagian perilaku sistem yang bersifat koheren tanpa perlu menyikap struktur internal sistem / perangkat lunak yang sedang dikembangkan. Berikut adalah beberapa simbol di dalam use case diagram.

Tabel 2. 1 Use Case Diagram (Nugroho:2008)

No	Simbol	Nama	Keterangan
1		Actor	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		Use case	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
3		System	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
4		Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).


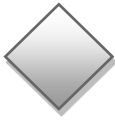



5		Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data objek yang ada di atasnya objek induk (<i>ancestor</i>).
6		Include	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
7		Extent	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan.
8		Assosiation	Menghubungkan antara objek satu dengan objek yang lainnya.

b. *Activity Diagram*

Activity Diagram sesungguhnya merupakan bentuk khusus dari *state machine* yang bertujuan memodelkan komputasi-komputasi dan aliran-aliran kerja yang terjadi dalam sistem atau perangkat lunak yang sedang dikembangkan. *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 50 paralel yang mungkin terjadi pada beberapa eksekusi. Berikut adalah beberapa simbol dalam *activity diagram*:

Tabel 2. 2 Activity Diagram (Nugroho:2008)

No	Simbol	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antar muka saling berinteraksi satu sama lain.
2		<i>Decision</i>	Pilihan untuk pengambilan keputusan.
3		<i>Initial Node</i>	Titik awal
4		<i>Activity Final Node</i>	Titik Akhir
5		<i>Fork</i>	Menunjuk kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.

c. *Class Diagram*


Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- 1) Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- 2) Operasi atau metode fungsi-fungsi yang dimiliki oleh suatu kelas.

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

- 1) Kelas main yaitu kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
- 2) Kelas yang menangani tampilan sistem (view) yaitu kelas yang mendefenisikan dan mengatur tampilan kepemakai.
- 3) Kelas yang diambil dari pendefenisian Use Case (controller) yaitu kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case.
- 4) Kelas yang diambil dari pendefinisian data (model) yaitu kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data. (Rosa dan Shalahuddin, 2013)

Tabel 2. 3 Class Diagram (Rosa dan Shalahuddin: 2013)

No	Simbol	Nama	Keterangan
1		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
2		Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		Asosiasi	Hubungan statis antar <i>class</i> yang menggambarkan <i>class</i> yang memiliki atribut dengan <i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain.
4		Generalization	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya

			objek induk (ancestor).
5	----->	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent).

E. Konsep Dasar Bahasa Jepang

1. Pengertian Bahasa Jepang

Bahasa dapat dikatakan sebagai salah satu ciri suatu bangsa. Bahasa Jepang sebagai salah satu bahasa yang memiliki kekhasan tersendiri, baik dilihat dari huruf maupun ungkapan-ungkapan bahasa yang digunakan dalam komunikasi (*“Analisis Kesalahan Penggunaan..”* 2010).

Sedangkan menurut Wikipedia Indonesia, ensiklopedia bebas berbahasa Indonesia, Bahasa Jepang adalah bahasa resmi di Jepang dan jumlah penutur 127 juta jiwa.

Bahasa Jepang memiliki karakteristik tertentu seperti penggunaan huruf, kosakata, sistem pengucapan, tata bahasa atau gramatika, dan ragam bahasanya. Apabila dibandingkan

dengan bahasa Indonesia, bahasa Jepang memiliki struktur pola kalimat yang berbeda. Struktur kalimat dalam bahasa Jepang berpola ‘Subjek – Predikat’ atau ‘Subjek – Objek – Predikat’, sedangkan struktur kalimat dalam bahasa Indonesia berpola ‘Subjek – Predikat – Objek’.

Pada saat ini bahasa Jepang, sebagaimana dikemukakan oleh Renariah (2002), merupakan salah satu bahasa asing yang banyak dipelajari di lembaga-lembaga pendidikan formal maupun non formal di Indonesia. Sebagai bahasa asing, tentu bahasa Jepang juga sama dengan bahasa - bahasa asing lainnya memiliki karakteristik-karakteristiknya yang harus diketahui dan dipahami oleh para pembelajar bahasa Jepang, hal ini sangat diperlukan untuk dapat mengkomunikasikan segala informasi kepada orang lain dengan bahasa yang kita pelajari, dalam arti bahwa kita harus dapat mengkomunikasikan dengan baik dan benar agar informasi yang kita sampaikan dapat diterima dengan baik oleh lawan bicara dalam bentuk lisan maupun bentuk tulisan, begitu pula sebaliknya.

2. Jenis-Jenis Huruf Jepang

Bahasa Jepang menggunakan tiga macam huruf yaitu huruf Hiragana, huruf Katakana, dan huruf Kanji. Pada zama dahulu pertama kali orang Jepang menggunakan huruf Kanji. Tetapi huruf Kanji yang berasal dari Cina tersebut awalnya merupakan huruf yang digunakan untuk menulis bahasa Cina, maka pada saat digunakana menulis bahasa Jepang terasa merepotkan. Oleh karena itu, orang Jepang membuat huruf Hiragana dan Katakana. Tetapi bukan berarti bahwa huruf Hiragana dan Katakana sama sekali tidak ada hubungannya dengan huruf Kanji, karena dua-duanya berasal dan dibuat dari huruf Kanji.

a. *Kanji*

Menurut sejarahnya Kanji adalah huruf yang berasal dari Cina, masuk ke Jepang sekitar abad ke IV – V, berjumlah kira-kira 50.000 huruf, kemudian dari beberapa kanji dikembangkan menjadi huruf hiragana dan katakana. Menurut Takebe (Renariah, 2002), kanji adalah salah satu jenis huruf yang dipergunakan dalam bahasa Jepang dan mempunyai ciri tersendiri terutama dalam cara baca dan cara penulisannya, oleh karena itu, kanji sering disebut sebagai huruf yang sangat rumit dan sukar untuk dipelajari namun demikian kanji merupakan salah satu huruf yang sangat penting dalam bahasa Jepang karena setiap huruf menyatakan arti. Di dalam bahasa Jepang kaya sekali akan kosa kata yang memiliki ucapan yang sama, tetapi dengan adanya kanji maka kesalahan pemahaman pengertian dapat dihindari. Takebe menyebut huruf kanji termasuk *Hyoo i moji* (表意文字), karena setiap huruf menyatakan arti. Bagi orang Jepang sendiri, apabila ia melihat kanji sepintas lalu maka meskipun ia belum mengetahui arti sebenarnya dan cara bacanya dari kanji yang ia lihat, akan tetapi secara sepintas lalu ia dapat dengan mudah memahami arti kata yang dimaksud.

Lebih lanjut Takebe menekankan bahwa dalam penggunaan kanji, kita tidak dapat sembarangan tetapi harus mengutamakan pada arti masing-masing kanji, karena hal tersebut sangat menentukan arti kanji yang dimaksud. Sehubungan dengan hal tersebut di atas, Takebe menambahkan bahwa suatu hal yang unik dan penting di dalam kanji adalah di dalam setiap kanji memiliki 3 unsur dasar yaitu 音、形、義 (*on, kei, gi* = bunyi, bentuk dan

arti), unsur-unsur ini tidak dimiliki dalam huruf lain, terutama huruf alfabet yang termasuk dalam *hyoo on moji*.

Telah banyak diakui oleh pembelajar bahasa Jepang dan tidak sedikit yang merasakannya bahwa Kanji adalah bidang yang sulit dipelajari dalam bahasa Jepang, terutama oleh para pembelajar bahasa Jepang yang tidak memiliki latar belakang “budaya kanji”, yang dalam bahasa Jepang disebut *hikanjiken* (非漢字圏), hal tersebut disebabkan oleh banyak faktor di antaranya adalah cukup banyak jumlah kanji yang harus diingat, cara baca kanji yang bervariasi, *hitsujun* (筆順 = cara menuliskannya) yang harus diperhatikan betul, serta pengetahuan kanji yang meliputi *bushu* (部首 = bagian kanji yang menentukan arti) serta *rikusho* (六書 = pembentukan dan pemakaian kanji).

1) Jumlah Kanji

Dalam kamus besar *Kanji* bahasa Jepang yaitu *kanwa daijiten* (漢和大辭典) yang merupakan kamus kanji terlengkap di Jepang, di dalamnya terdapat 450.000 huruf kanji. Bagi orang Jepang sendiri jumlah tersebut merupakan jumlah yang sangat banyak, terlebih-lebih bagi pembelajar bahasa Jepang yang tidak memiliki latar belakang kanji, dimana mereka terbiasa dengan huruf alfabet yang berjumlah hanya 26 huruf. Pada zaman Meiji, tepatnya tahun 1990, Departemen Kependidikan Jepang yaitu *Monbusho* (文部省) menetapkan jumlah kanji yang wajib dipelajari di Sekolah Dasar sebanyak 1200 kanji sebagai kanji standar, lalu 46 tahun kemudian tepatnya pada tanggal 16 November 1946 dengan maklumat

kabinet ditetapkan *Tooyoo Kanji* (当用漢字) . *Tooyoo Kanji* adalah kanji yang digunakan secara umum dalam kehidupan sehari-hari berjumlah 1850 kanji, yang di dalamnya meliputi *kyooiku kanji* (教育漢字) . *Kyooiku kanji* adalah kanji yang harus dipelajari oleh siswa SD dan SMP jumlahnya 881 kanji, kemudian seiring dengan perkembangan zaman maka kebutuhan kanji pun bertambah, oleh karena itu pada tanggal 1 Oktober 1981 ditetapkan *jooyoo kanji* (常用漢字) . *Jooyoo kanji* adalah kanji yang digunakan dalam kehidupan sehari-hari berjumlah 1945 kanji, jumlah ini berasal dari *tooyoo* kanji dengan penambahan 95 kanji.

2) Cara Membaca Kanji

Di dalam Kanji terdapat 2 (dua) cara baca kanji yaitu *kun-yomi* dan *on-yomi*. *Kun-yomi* adalah cara baca kanji secara ucapan asli bahasa Jepang, sedangkan *Onyomi* adalah cara baca kanji dengan meniru ucapan Cina tetapi dimodifikasikan sesuai dengan ucapan Jepang. Jumlah *Kun-yomi* dan *Onyomi* dalam setiap kanji berbeda-beda, ada kalanya kanji hanya memiliki *on-yomi* saja tanpa memiliki *kun-yomi* nya begitu pula sebaliknya bahkan ada pula *on-yomi* nya banyak ataupun *kun-yomi* nya lebih dari satu. Apabila kita amati *jooyoo kanji hyoo* (= daftar *jooyoo* kanji) maka kita akan tercengang, karena di dalamnya terdapat 2187 *on-yomi* dan 1900 *kun-yomi*, sehingga jumlah keduanya mencapai 4087, jumlah sebanyak ini bukanlah jumlah yang sedikit dan mudah untuk diingat.

Terutama dalam kanji yang memiliki *on-yomi* lebih dari satu macam, hal tersebut disebabkan karena kanji yang masuk ke Jepang mendapat pengaruh dari perubahan dan perkembangan jaman di Cina, sehingga di dalam *on-yomi* secara rinci dikenal dengan *goon* (吳音), *kan on* (漢音) dan *toon* (唐音). *Go on* adalah kanji yang masuk ke Jepang pada jaman Nara melalui korea dengan ucapan bahasa Cina pada abad ke 6-7. *Kan on* adalah kanji dengan ucapan bahasa Cina jaman dinasti Han, masuk ke Jepang pada jaman Nara dan Heian, kira-kira abad ke-8. *Too on* adalah ucapan bahasa Cina jaman pertengahan jaman dinasti Sung, Yuan dan Ming yang disebarkan ke Jepang pada jaman Kamakura, kira-kira pada abad ke-12. Oleh karena itu, tidaklah mengherankan bahwa di dalam kanji yang sama terdapat *on-yominya* lebih dari satu.

Contoh:

明日 /**Myoonichi**/ (*Goon*) = besok

説明 /**Setsumei**/ (*Kan on*) = penjelasan

明朝 /**Minchoo**/ (*Too on*) = dinasti Ming

行列 /**Gyooretsu**/ (*Go on*) = antrian

銀行 /**Ginkoo**/ (*Kan on*) = bank

行灯 /**Andon**/ (*Too on*) = lampion

東京 /**Tookyoo**/ (*Go on*) = Tokyo

京浜 /**Keihin**/ (*Kan on*) = Tokyo – Yokohama

北京 /**Pekin**/ (*Too on*) = Beijing

Kanji	On - yomi		
	Go on	Kan on	To on
明	Myoo	Mei	Min
行	Gyoo	Koo	An
京	Kyoo	Kei	Kin

Tabel 2. 4 Tabel Kanji *On-yomi*

Sebaliknya di dalam satu kanji, selain memiliki *on-yomi* yang bervariasi adakalanya juga memiliki *kun-yomi* lebih dari satu. Contoh : 上、生、明 dan sebagainya.

上 *Kun-yomi* : *ue, uwa, kami, a-garu, a-geru.*

生 *Kun-yomi* : *nama, ki, i-kiru, i-kasu, i-keru, u-mareru, u-mu, o-u, haeru, ha-yasu.*

明 *Kun-yomi* : *a-kari, aka-rui, aka-rui, aka-ramu, aki-raka, a-keru, aku, a-luru, a-kasu.*

Selain kanji-kanji yang memiliki *kun-yomi* dan *on-yomi* yang bervariasi, adapula kanji yang hanya memiliki *kun-yomi* saja, kanji-kanji tersebut dalam bahasa Jepang disebut *Kokuji* (国字), berjumlah 36 kanji, yaitu :

虱 (orosu)	畠 (hatake)	襷 (tasuki)
栃 (tochi)	杣 (soma)	鯰 (namazu)
働 (hataraki)	雫 (shizuku)	鋌 (byoo)
冨 (kogarashi)	辻 (tsuji)	榎 (kashi)
槓 (maki)	鰯 (iwashi)	俤 (omokage)

笹 (sasa)	料 (kirome-teru)	凧 (nagi)
峠 (tooge)	榊 (sakaki)	柶 (waku)
鰹 (katsuo)	糶 (kooji)	込 (ko-mu)
罎 (kiroguramu)	畑 (hatake)	袴 (kamishimo)
柶 (toga)	栴 (momiji)	鱈 (tara)
糶 (momi)	躰 (shitsuke)	鰯 (buri)
糶 (senchime-toru)	菊 (kiku)	麿 (maro)

b. *Hiragana*

Huruf hiragana melambangkan suku kata tunggal, dan digunakan untuk menulis kata-kata yang berasal dari Jepang asli. Huruf Hiragana terbentuk dari modifikasi dan penyederhanaan kanji, sehingga huruf hiragana coretannya melengkung dan tidak bersudut tajam, sehubungan dengan hal tersebut Takebe (Renariah, 2002:3) menyebutkan bahwa huruf hiragana digunakan mulai jaman Edo, yang pada awal mulanya digunakan oleh kaum wanita sehingga hiragana dikenal dengan huruf wanita, dalam bahasa Jepang disebut *onnade*. Huruf Hiragana berjumlah 46 huruf di dalamnya mencakup 5 vokal yaitu a i u e o sisanya adalah suku kata dengan deretan ka, sa, ta, na, ha, ma, ya ra, dan satu konsonan yaitu n serta satu kata bantu o yang kadangkala diucapkan wo. Asal usul huruf Hiragana dapat kita lihat dalam tabel berikut :

あ (安) a	い (以) i	う (宇) u	え (衣) e	お (於) o
か (加) ka	き (幾) ki	く (久) ku	け (計) ke	こ (己) ko
さ (左) sa	し (之) shi	す (寸) su	せ (世) se	そ (僧) so
た (太) ta	ち (知) chi	つ (川) tsu	て (天) te	と (止) to
な (奈) na	に (任) ni	ぬ (奴) nu	ね (祢) ne	の (乃) no
は (波) ha	ひ (比) hi	ふ (不) fu	へ (部) he	ほ (保) ho
ま (末) ma	み (美) mi	む (武) mu	め (女) me	も (毛) mo
や (也) ya		ゆ (由) yu		よ (与) yo
ら (良) ra	り (利) ri	る (留) ru	れ (礼) re	ろ (呂) ro
わ (和) wa				を (遠) wo
ん (天) n				

Tabel 2. 5 Tabel Hiragana *Sei On*

46 (empat puluh enam) huruf hiragana tersebut di atas disebut *sei on*. Huruf hiragana dapat dimodifikasi dengan menambahkan tanda tertentu dan menggabungkan dengan huruf lain sehingga dihasilkan bunyi yang berbeda. Bunyi tersebut, yaitu bunyi *handakuon*, *dakuon* dan *yoo on*.

Contoh :

さくら (*Sakura*) = bunga sakura

わたし (*Watashi*) = saya

ほんや (*Honya*) = toko buku

ぱ pa	ば ba	だ da	ざ za	が ga
ぴ pi	び bi	ぢ ji	じ ji	ぎ gi
ぷ pu	ぶ bu	づ zu	ず zu	ぐ gu
ぺ pe	べ be	で de	ぜ ze	げ ge
ぽ po	ぼ bo	ど do	ぞ zo	ご go

Tabel 2. 6 Tabel Hiragana *Handaku On*

Bunyi *dakuon* adalah bunyi huruf hiragana dasar dengan menambahkan [`] tanda tenten (titik dua yang diletakkan di sebelah kanan atas huruf hiragana dasar). Dan [o] tanda maru (lingkaran kecil yang diletakkan di sebelah kanan atas huruf hiragana dasar). *Daku on* berjumlah 25 (dua puluh lima) huruf hiragana, yang termasuk di dalamnya adalah sebagai berikut :

ぱ pa	ば ba	だ da	ざ za	が ga
ぴ pi	び bi	ぢ ji	じ ji	ぎ gi
ぷ pu	ぶ bu	づ zu	ず zu	ぐ gu
ぺ pe	べ be	で de	ぜ ze	げ ge
ぽ po	ぼ bo	ど do	ぞ zo	ご go

Tabel 2. 7 Tabel Hiragana *Daku On*

Contoh :

おどり (*odori*) = tarian

かぞく (*kazoku*) = keluarga

えんぴつ (*empitsu*) = pensil

Bunyi *yoo on* adalah bunyi huruf hiragana dasar dengan menambahkan huruf や (*ya*), ゆ (*yu*) dan よ (*yo*) yang ditulis lebih kecil daripada huruf dasar. Apabila huruf や (*ya*), ゆ (*yu*) dan よ (*yo*) ditulis sama dengan huruf dasar maka akan terbaca masing-masing hurufnya, misalnya huruf ひ や (*hiya*) berbeda dengan ひ や (*hya*).

Untuk itu perlu diperhatikan penulisan huruf tersebut. Huruf dasar yang menggunakan huruf や (*ya*), ゆ (*yu*) dan よ (*yo*) adalah huruf dasar urutan kedua, misalnya huruf き (*ki*), し (*shi*), ち (*chi*), に (*ni*), ひ (*hi*), み (*mi*), dan り (*ri*).

Contoh:

りや <i>rya</i>	みや <i>mya</i>	ひや <i>hya</i>
りゆ <i>ryu</i>	みゆ <i>myu</i>	ひゆ <i>hyu</i>
りよ <i>ryo</i>	みよ <i>myo</i>	ひよ <i>hyo</i>
にや <i>nya</i>	きや <i>kya</i>	しや <i>sha</i>
にゆ <i>nyu</i>	きゆ <i>kyu</i>	しゆ <i>shu</i>
によ <i>nyo</i>	きよ <i>kyo</i>	しよ <i>sho</i>
ぢや <i>ja</i>	じや <i>ja</i>	ぎや <i>gya</i>
ぢゆ <i>ju</i>	じゆ <i>ju</i>	ぎゆ <i>gyu</i>
ぢよ <i>jo</i>	じよ <i>jo</i>	ぎよ <i>gyo</i>
ちや <i>cha</i>	ぴや <i>pya</i>	びや <i>bya</i>
ちゆ <i>chu</i>	ぴゆ <i>pyu</i>	びゆ <i>byu</i>
ちよ <i>cho</i>	ぴよ <i>pyo</i>	びよ <i>byo</i>

Tabel 2. 8 Tabel Hiragana *Yoo On*

Contoh :

きゃく (*kyaku*) = tamu

ひゃくえん (*hyaku en*) = seratus yen

Huruf hiragana berfungsi untuk menuliskan :

1. bagian dari *yoogen* (用言) , yaitu kata kerja, kata sifat – i dan kata sifat na. sebagai *okurigana*.

Contoh :

読みます (*yomimasu*) = membaca

青い (*aoi*) = biru

静かな (*shizukana*) = sepi

2. *joshi* (助詞) , yaitu partikel/kata bantu

Contoh :

私は 毎晩 8 時 から 10 時 まで 復習 を します。

(*watashi wa maiban hachiji kara juuji made fukushuu o shimasu*).

Setiap malam saya mengulang pelajaran dari pukul 20.00 sampai pukul 22.00

3. *jodoshi* (助動詞) , yaitu kata kerja bantu

Contoh :

a) 私は学生です。

(*watashi wa gakusei desu*)

saya adalah mahasiswa

b) あの方は先生ではありません。

(*anokata wa sensei dewa arimasen*)

orang itu bukan guru

c) 弟は本屋で辞書を買いました。

(*ototo wa honya de jijo o kaimashita*)

adik laki-laki membeli kamus di toko buku

4. awalan dan akhiran sebagai penghalus kata (接頭辞、
接尾辞)

Contoh :

お手紙 (**o** tegami) = surat

ご両親 (**go** ryoushin) = orang tua

奥さん (**okusan**) = nyonya

c. *Katakana*

Huruf katakana sama seperti huruf Hiragana yaitu melambangkan suku kata tunggal, tetapi mempunyai fungsi yang berbeda dengan huruf hiragana. Huruf Katakana selain digunakan untuk menulis kata-kata yang berasal dari bahasa asing, juga digunakan untuk penekanan suatu kata yang berasal dari Jepang asli. Huruf Katakana juga terbentuk dari modifikasi kanji dengan cara mengambil salah satu bagian kanji, sehingga di dalam huruf katakana kita tidak akan menemukan coretannya yang melengkung seperti hiragana, katakana bentuk hurufnya terkesan kaku, karena setiap coretannya bersudut tajam sehingga katakana disebut sebagai huruf laki-laki .

Huruf Katakana berjumlah 45 (empat puluh lima) huruf, karena dalam di dalam huruf Katakana tidak terdapat huruf o yang berfungsi sebagai kata bantu. Susunan huruf katakana sama dengan huruf Hiragana, di dalamnya mencakup 5 (lima) vokal yaitu a i u e o dan sisanya adalah suku kata dengan deretan ka, sa, ta, na, ha, ma, ya ra, wa dan satu konsonan yaitu n.

ア (阿) a	イ (伊) i	ウ (宇) u	エ (江) e	オ (於) o
カ (加) ka	キ (幾) ki	ク (久) ku	ケ (介) ke	コ (己) ko
サ (散) sa	シ (之) shi	ス (須) su	セ (世) se	ソ (曾) so
タ (多) ta	チ (千) chi	ツ (川) tsu	テ (天) te	ト (止) to
ナ (奈) na	ニ (二) ni	ヌ (奴) nu	ネ (祢) ne	ノ (乃) no
ハ (八) ha	ヒ (比) hi	フ (不) fu	ヘ (部) he	ホ (保) ho
マ (末) ma	ミ (三) mi	ム (牟) mu	メ (女) me	モ (毛) mo
ヤ (也) ya		ユ (由) yu		ヨ (与) yo
ラ (良) ra	リ (利) ri	ル (流) ru	レ (礼) re	ロ (呂) ro
ワ (和) wa				
ン (尔) n				

Tabel 2. 9 Tabel Katakana

Huruf katakana berfungsi untuk menuliskan :

- 1) Kata-kata yang berasal dari bahasa asing
Contoh : パイロット (pairotto) = pilot
- 2) Nama orang dan tempat asing serta kata-kata benda asing.
Contoh :
nama orang アミル (Amiru) ジョン (John)
nama tempat インドネシア (Indonesia)
benda asing ボート (boat = perahu)
- 3) *Giseigo* (擬声語) dan *giongo* (擬態語) yaitu onomatope.
Contoh :
a) 犬は「ワンワン」となきます。
(inu wa “wanwan” to nakimasu)
anjing menggonggong
b) ピカピカ 光る。
(pikapika hikaru)
bersinar gemerlapan
- 4) Telegram
Contoh :
ニジュウイチニチ、サンジ ナリタクウコウニツク。
(nijuuichi nichu, sanji narita kuukoo ni tsuku)
Tiba di bandara Narita pada tanggal 21 pukul 15.00

BAB III ANALISA DAN HASIL

A. Analisa Kebutuhan Aplikasi

Alat penelitian yang digunakan dalam menyusun tugas akhir di bagi menjadi 2, yaitu hardware dan software. Alat yang digunakan sebagai berikut.

1. Hardware

- a. Motherboard Acer Aspire E1-432
- b. Processor Intel Celeron (R) @ 1.4 GHz
- c. Random Access Memory (RAM) 2 GB
- d. Flashdisk Kingston 64 GB
- e. Hardisk Drive 500 GB
- f. Smartphone Evercoss A7S RAM 512 MB

2. Software

- a. Sistem Operasi Windows 7 Ultimate 64-Bit
- b. Microsoft Office 2010
- c. Android Studio

B. Analisa Aplikasi

Dalam pembuatan aplikasi perangkat lunak dibutuhkan adanya suatu analisa dan perancangan dari sistem yang akan dibuat. Ini bertujuan agar aplikasi yang akan dibuat dapat selesai tepat waktu dan sesuai dengan tujuan awal. Kesalahan dalam analisa akan berimbas pada tahapan proyek selanjutnya dan mengakibatkan aplikasi tidak maksimal atau tidak sesuai dengan tujuan pembuatan aplikasi.

Tahapan analisis ini berisi identifikasi dan evaluasi permasalahan yang ada serta kebutuhan apa saja yang dibutuhkan dalam pembuatan aplikasi perangkat lunak. Dengan adanya analisa dan perancangan maka akan didapatkan suatu gambaran mengenai kebutuhan apa saja yang dibutuhkan dalam program aplikasi ini.

Langkah-langkah dalam penggunaan aplikasi dari awal hingga akhir adalah sebagai berikut:

1. User menjalankan aplikasi ke halaman utama
2. Di dalam halaman utama terdapat sebuah menu utama yang memiliki 4 (empat) buah tombol yaitu tombol Huruf, tombol Kosakata, tombol Percakapan, dan tombol Quiz.
3. Jika user memilih tombol Huruf maka user akan melihat tabel dari Hiragana dan Katakana. Jika user memilih salah satu huruf, boleh yang di Hiragana maupun Katakana, maka akan terdengar suara dari huruf yang dipilih tersebut.
4. Jika user memilih tombol Kosakata maka user akan melihat beberapa submenu dari Kosakata tersebut. Jika user memilih salah satu daftar kosakata, maka akan terdengar suara dari daftar kosakata yang dipilih tersebut.
5. Jika user memilih tombol Percakapan maka user akan melihat beberapa submenu dari percakapan tersebut. Jika user memilih salah satu dari percakapan, maka akan terdengar suara dari percakapan yang dipilih tersebut namun harus kita koneksikan ke internet terlebih dahulu.
6. Jika user memilih tombol Quiz maka user akan melihat salah satu daftar kosakata. Maka user harus menjawab apa yang ditunjuk dari kuis tersebut. Walaupun kita tidak tahu jawabannya kita bisa melompati untuk ke kuis selanjutnya.

C. Perancangan Sistem

1. Actor

Actor yang berperan dalam Aplikasi Pembelajaran Bahasa Jepang ini adalah pengguna (*user*) dari aplikasi yang akan dirancang. Peran *user* dapat dilihat pada tabel di bawah ini:

Actor	Peran
User	<ol style="list-style-type: none"> 1. Melihat huruf Jepang 2. Melihat daftar huruf hiragana dan katakana 3. Memilih salah satu daftar tabel huruf hiragana atau katakana 4. Mendengarkan salah satu huruf yang dipilih 5. Melihat kosakata 6. Memilih salah satu daftar kosakata 7. Mendengarkan daftar kosakata yang dipilih 8. Melihat percakapan 9. Memilih salah satu percakapan 10. Mendengarkan percakapan yang dipilih 11. Melihat kuis 12. Menjawab beberapa pertanyaan yang diajukan
Admin	<ol style="list-style-type: none"> 1. Melihat huruf Jepang 2. Melihat daftar huruf hiragana dan katakana 3. Memilih salah satu daftar tabel huruf hiragana atau katakana 4. Mendengarkan salah satu huruf yang dipilih 5. Melihat kosakata 6. Memilih salah satu daftar kosakata 7. Mendengarkan daftar kosakata yang dipilih 8. Menambahkan kosakata 9. Menghapus kosakata 10. Mengedit kosakata 11. Melihat percakapan 12. Mendengarkan percakapan yang dipilih 13. Menambah percakapan 14. Menghapus percakapan

	15. Mengedit percakapan 16. Melihat kuis 17. Menjawab beberapa pertanyaan yang diajukan 18. Menambah soal kuis 19. Menghapus soal kuis 20. Mengedit soal kuis
--	--

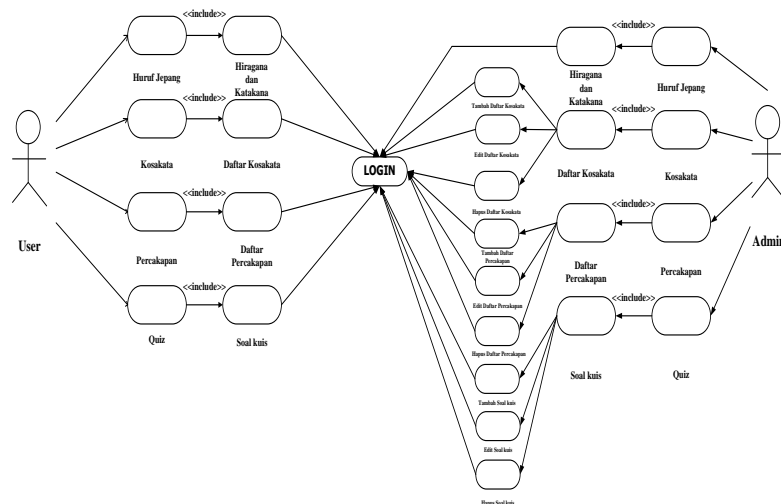
Tabel 3. 1 Tabel Peran Actor

2. Use Case Diagram

Diagram ini digunakan untuk menggambarkan user aplikasi dan perilaku user terhadap aplikasi. User yang diwakili oleh aktor sedangkan perilakunya diwakili oleh use case.

- a. Saat user menjalankan aplikasi, maka user akan masuk kepada menu utama aplikasi. User akan melihat submenu huruf, kosakata, percakapan dan quiz.
- b. User dapat melihat daftar huruf hiragana dan katakana, daftar kosakata, daftar percakapan dan kuis yang diajukan.

Untuk lebih jelasnya use case diagram user dapat dilihat pada Gambar 3.1.

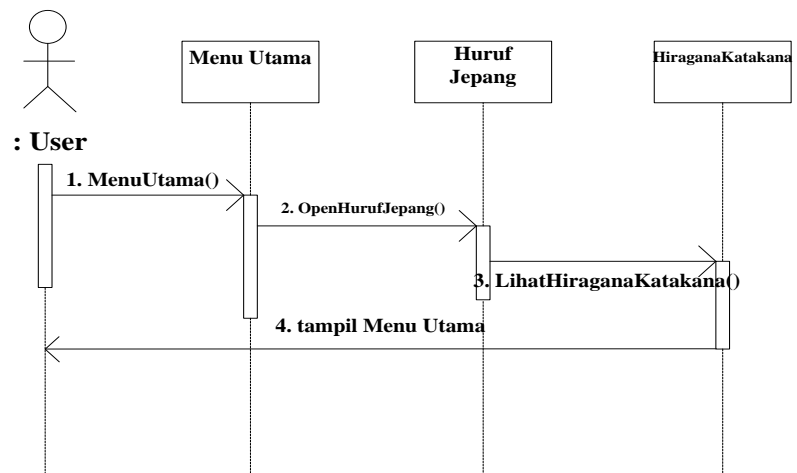


Gambar 3. 1 Rancangan Use Case Diagram

3. Sequence Diagram

a. Sequence Diagram Huruf Jepang

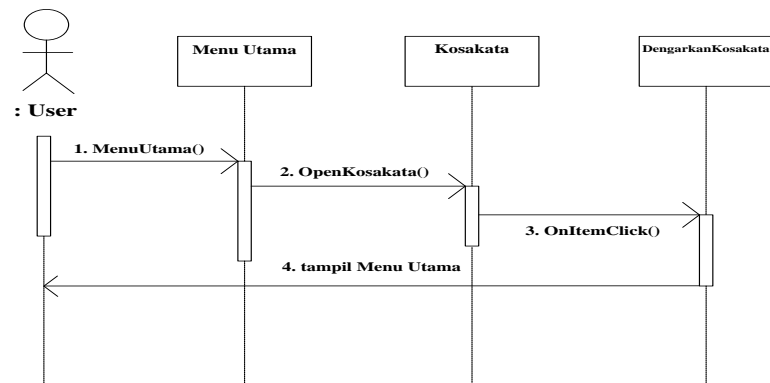
Untuk melihat daftar huruf, user harus masuk terlebih dahulu ke dalam menu utama aplikasi. Kemudian akan tampil daftar huruf hiragana dan katakana. Untuk lebih jelasnya dapat dilihat pada Gambar 3.2.



Gambar 3. 2 Sequence Diagram Huruf Jepang

b. Sequence Diagram Kosakata

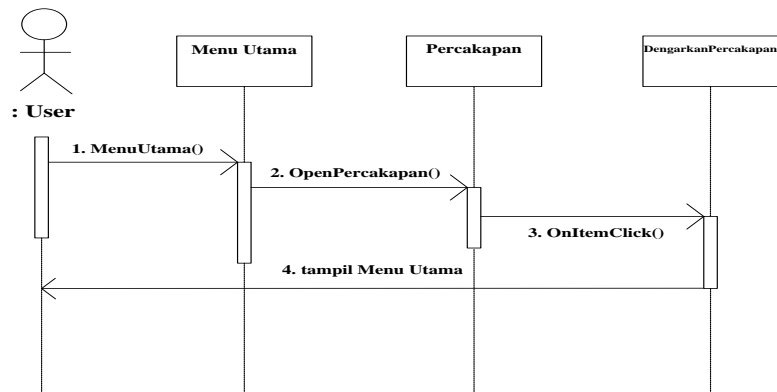
Untuk melihat daftar kosakata maka user harus masuk ke dalam menu kosakata. User bisa mendengarkan suara dari kosakata tersebut. Untuk lebih jelasnya dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Sequence Diagram Kosakata

c. Sequence Diagram Percakapan

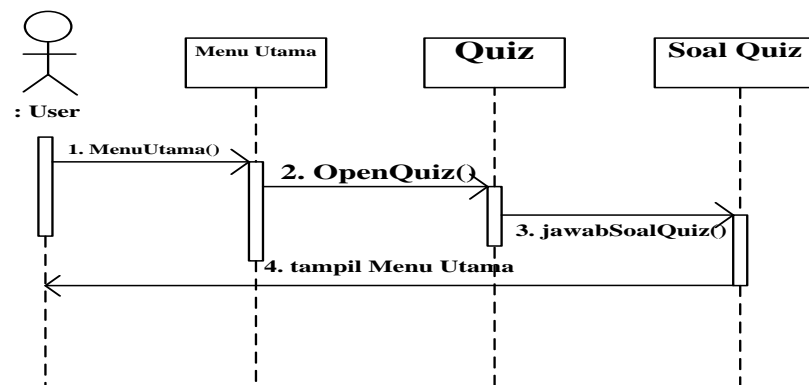
Untuk melihat daftar percakapan maka user harus masuk ke dalam menu kosakata. User akan melihat beberapa sub menu dari percakapan. Untuk lebih jelasnya dapat dilihat pada Gambar 3.4.



Gambar 3. 4 Sequence Diagram Percakapan

d. Sequence Diagram Quiz

Untuk melihat quiz maka user harus masuk ke dalam menu quiz. User akan melihat beberapa pertanyaan dari soal quiz yang diberikan. Untuk lebih jelasnya dapat dilihat pada Gambar 3.5.



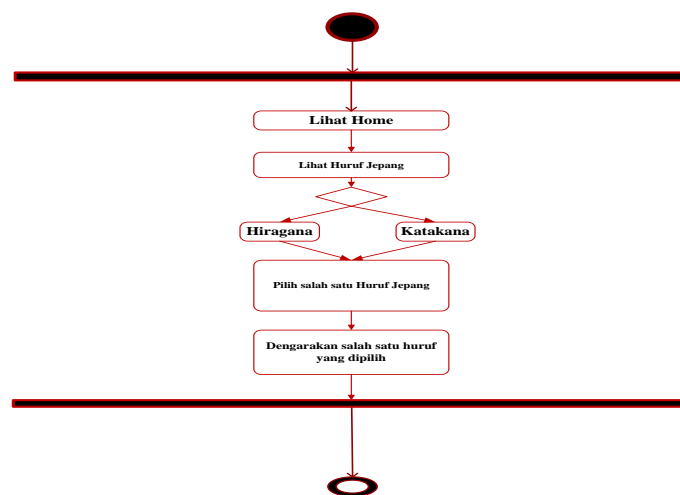
Gambar 3. 5 Sequence Diagram Quiz

4. Activity Diagram

Activity diagram menggambarkan proses-proses yang terjadi saat aktifitas dimulai sampai dengan aktifitas berhenti. Activity diagram ini mirip dengan flowchart diagram. Activity diagram dibagi 5 yaitu:

a. Activity Huruf Jepang

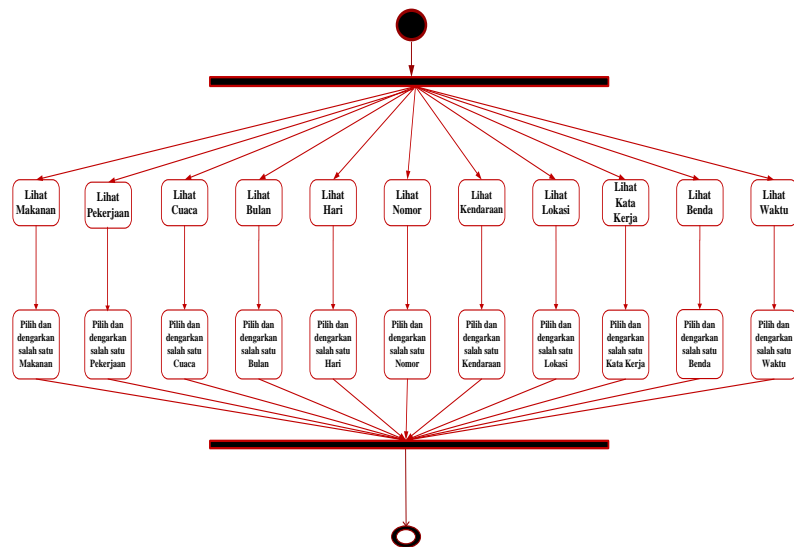
Activity Huruf Jepang menggambarkan aktifitas user dimulai dari menekan tombol Huruf sampai mendengarkan salah satu huruf hiragana / katakana. Untuk lebih jelasnya dapat dilihat pada Gambar 3.6.



Gambar 3. 6 Activity Huruf Jepang

b. Activity Kosakata

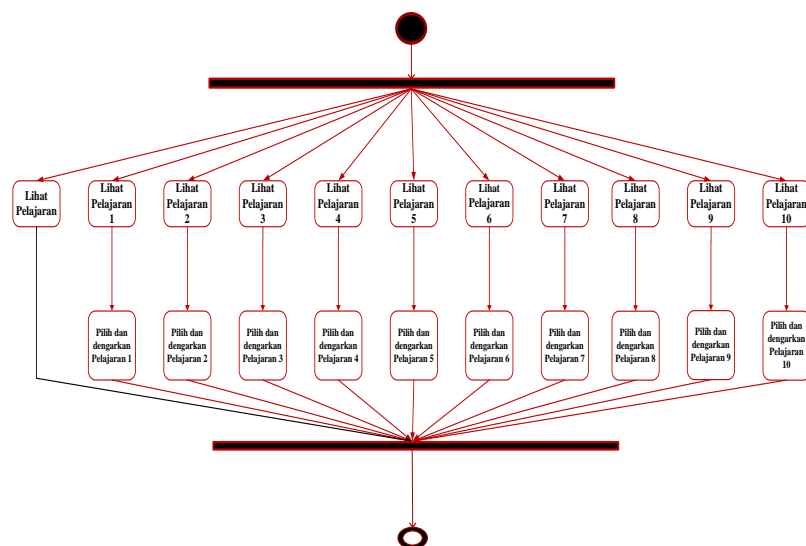
Activity kosakata menggambarkan aktifitas user dimulai dari menekan tombol kosakata sampai mendengarkan salah satu daftar kosakata. Untuk lebih jelasnya dapat dilihat pada Gambar 3.7.



Gambar 3. 7 Activity Kosakata

c. Activity Percakapan

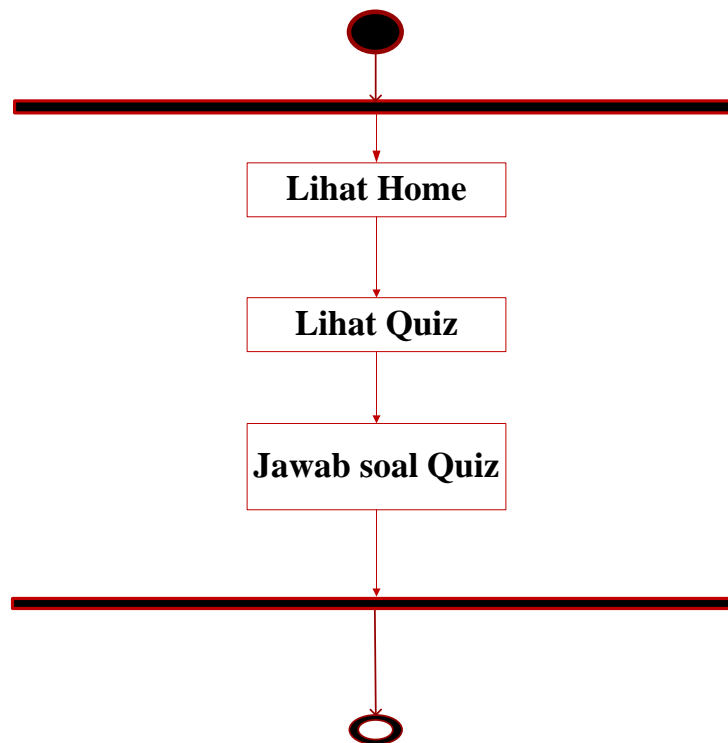
Activity percakapan menggambarkan aktifitas user dimulai dari menekan tombol percakapan sampai mendengarkan salah satu daftar percakapan. Untuk lebih jelasnya dapat dilihat pada Gambar 3.8 di bawah ini:



Gambar 3. 8 Activiy Percakapan

d. Activity Quiz

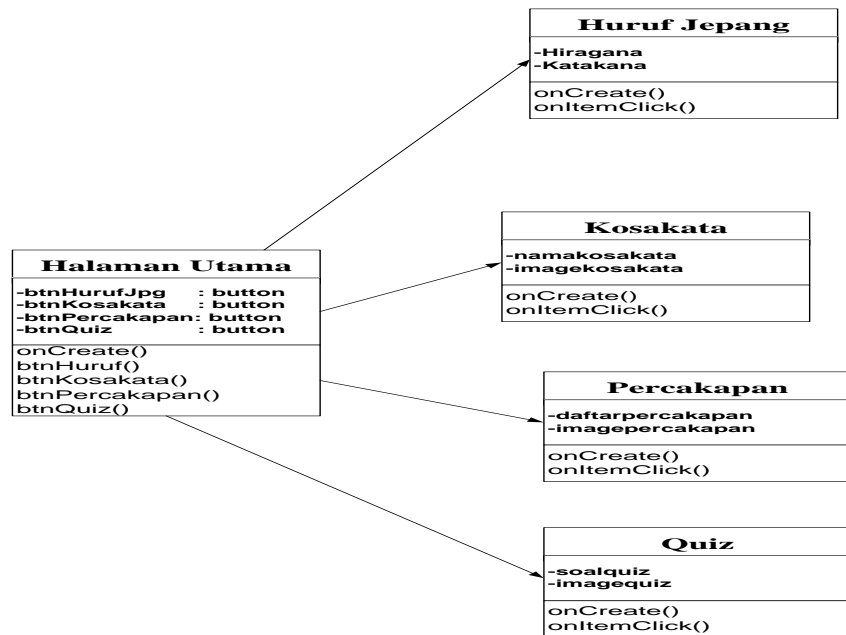
Activity percakapan menggambarkan aktifitas user dimulai dari menekan tombol Quiz sampai menjawab soal Quiz. Untuk lebih jelasnya dapat dilihat pada Gambar 3.9.



Gambar 3. 9 Activity Quiz

5. Class Diagram

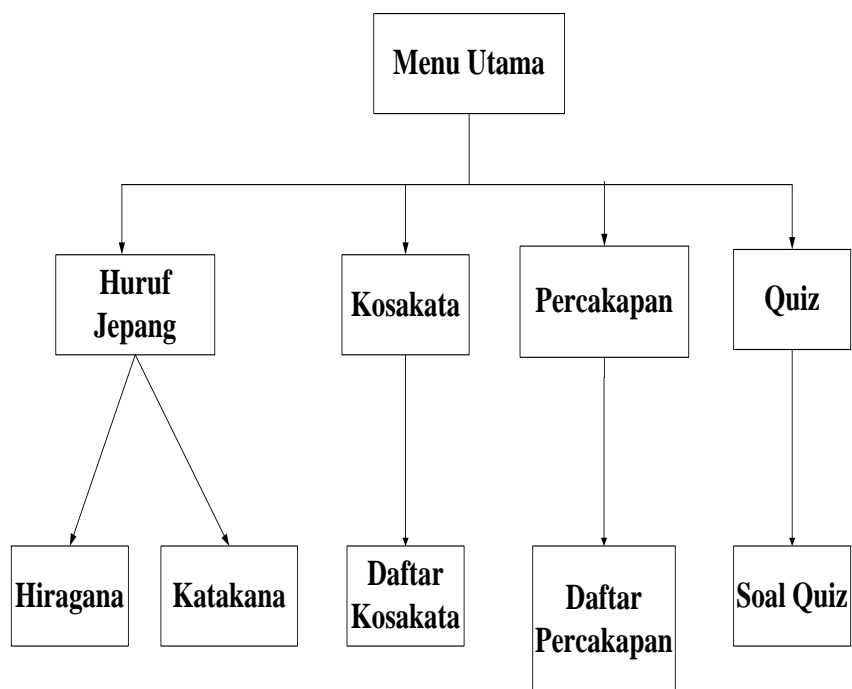
Class Diagram menunjukkan hubungan antar class atau tabel dalam sistem yang sedang dibangun dan bagaimana saling berkolaborasi untuk mencapai satu tujuan.. Untuk lebih jelasnya dapat dilihat pada Gambar 3.10.



Gambar 3. 10 Class Diagram

6. Struktur Program

Struktur program dari aplikasi dapat dilihat pada Gambar 3.11.



Gambar 3. 11 Struktur Program

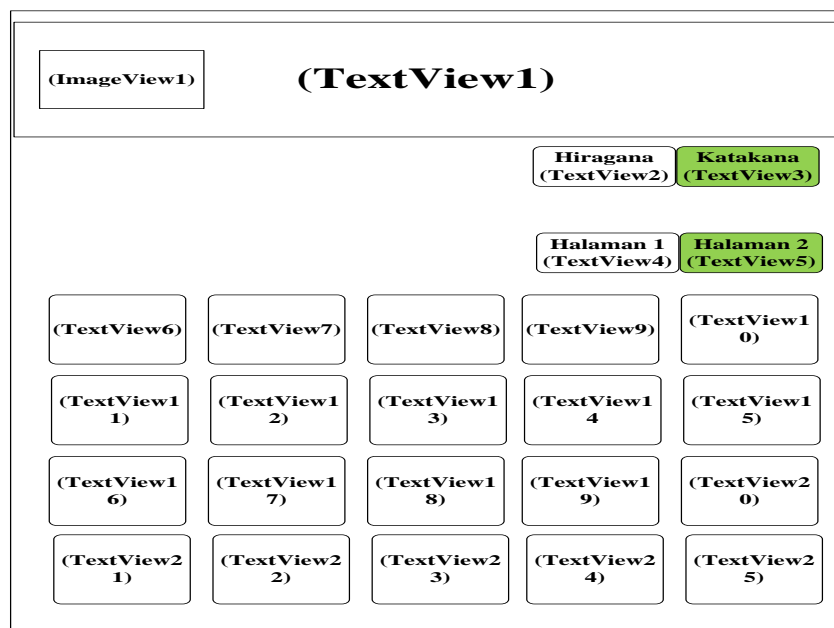
D. Desain Output Sistem

1. Tampilan Menu Utama



Gambar 3. 12 Tampilan Menu Utama

2. Tampilan Sub Menu Huruf Jepang






Gambar 3. 13 Tampilan Sub Menu Huruf Jepang

3. Tampilan Sub Menu Kosakata

(TextVie w1)	(TextVie w2)	(TextVie w3)	(TextVie w4)	(TextVie w5)	(TextVie w6)	(TextVie w7)	(TextVie w8)	(TextVie w9)	(TextVie w10)	(TextVie w11)
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									
Gambar Kosakata (ImageView1)	Nama Kosakata (ItemListView1)									

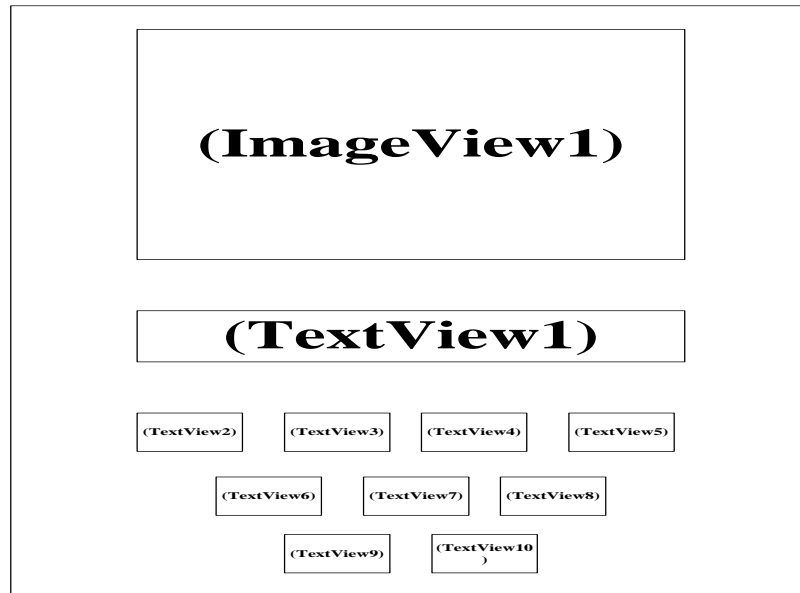
Gambar 3. 14 Tampilan Sub Menu Kosakata

4. Tampilan Sub Menu Percakapan

(TextV iew1)	(TextV iew2)	(TextV iew3)	(TextV iew4)	(TextV iew5)	(TextV iew6)	(TextV iew7)	(TextV iew8)	(TextV iew9)	(TextV iew10)	(TextV iew11)
(TextView12)										
(ImageView1)										
(ImageView2)	(TextView13)									
										
(ImageView3)	(TextView14)									
										
(ImageView4)	(TextView15)									
										

Gambar 3. 15 Tampilan Sub Menu Percakapan

5. Tampilan Sub Menu Quiz



Gambar 3. 16 Tampilan Sub Menu Quiz

BAB IV

PENUTUP

A. Kesimpulan

Dari hasil penelitian ini dapat penulis simpulkan:

1. Aplikasi Dasar Bahasa Jepang Berbasis Android ini dapat digunakan untuk menambah wawasan berbahasa Jepang dengan mengetahui huruf, kosakata, dan percakapan. Ditambah dengan kuis didalamnya.
2. Aplikasi ini juga dapat digunakan oleh user untuk memutar suara yang dipilih dari menu-menu yang sudah ada.
3. Diharapkan dengan adanya aplikasi ini akan meningkatkan kemampuan berbahasa Jepang siswa khususnya di SMAN 2 Sungai Tarab.

B. Saran

Penulis menyadari bahwa masih terdapat kekurangan dalam aplikasi ini. Untuk itu melalui bagian ini disarankan pengembangan dan penyempurnaannya, sebagai berikut:

1. Penambahan huruf *kanji* yang juga merupakan salah satu huruf Jepang selain *hiragana* dan *katakana*.
2. Penambahan menu inputan kata pada aplikasinya.

DAFTAR PUSTAKA

- Dharwiyanti, S. (2003). Kuliah Umum Ilmu Komputer.com Retrieved Juli 01, 2014, From Ilmukomputer.com: <http://www.ilmukomputer.com>
- Juansyah, Andi. “Pembangunan Aplikasi Child Tracker Berbasis Assisted–Global Positioning System (A-GPS) dengan Platform Android”. Jurnal Ilmiah Komputer dan Informatika (KOMPUTA). (online), (<http://elib.unikom.ac.id/files/disk1/673/jbptunikompp-gdl-andijuansy-33648-11-20.unik-a.pdf> , diakses pada tanggal 11 Februari 2018)
- Komputer, Wahana. *Step by Step menjadi Programmer Android*. 2013. Yogyakarta: Andi
- Kusuma, dkk. (2015). ”Aplikasi Pembelajaran Huruf Jepang Hiragana dan Katakana Berbasis Sistem Operasi Android”. Jurnal Teknik FTUP. (online). (https://litbangftup.files.wordpress.com/2015/10/jt-ftup-2015_4.pdf, diakses pada tanggal 27 Februari 2018)
- mAster.com. *Sekali Baca Langsung Inget Mengupas Lengkap All About Android*. 2012. Jakarta Pusat: Kuncikom
- Nugroho, A. Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek, Bandung : Informatika, 2005.
- Renariah.(2002).”Bahasa Jepang dan Karakteristiknya”. Jurnal Sastra Jepang Fakultas Sastra Universitas Kristen Maranatha. (online). (https://www.academia.edu/8451982/Bahasa_Jepang_dan_karakteristiknya, diakses pada tanggal 31 Juli 2017)
- Safaat, Nazruddin. *Android: Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. 2014. Bandung: Informatika
- Sagala, Syaiful. *Konsep dan Makna Pembelajaran: Untuk Membantu Memecahkan Problematika Belajar dan Mengajar*. 2003. Bandung: CV Alfabeta
- Tn.(2010).”Analisis Kesalahan Penggunaan Ungkapan “Yari-Morai” dan Pemerolehannya pada Pembelajar Bahasa Jepang”. Asosiasi Studi

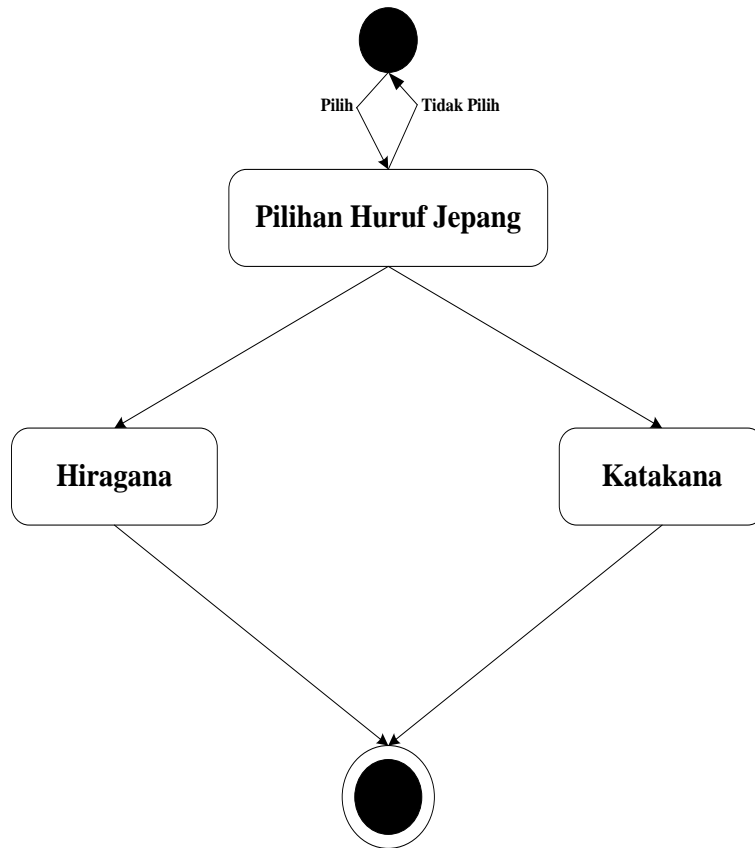
Pendidikan Bahasa Jepang Indonesia. (online). (<http://erepo.unud.ac.id>, diakses pada tanggal 31 Juli 2017)

https://id.wikipedia.org/wiki/Bahasa_Jepang. Diakses pada tanggal 22 November 2017

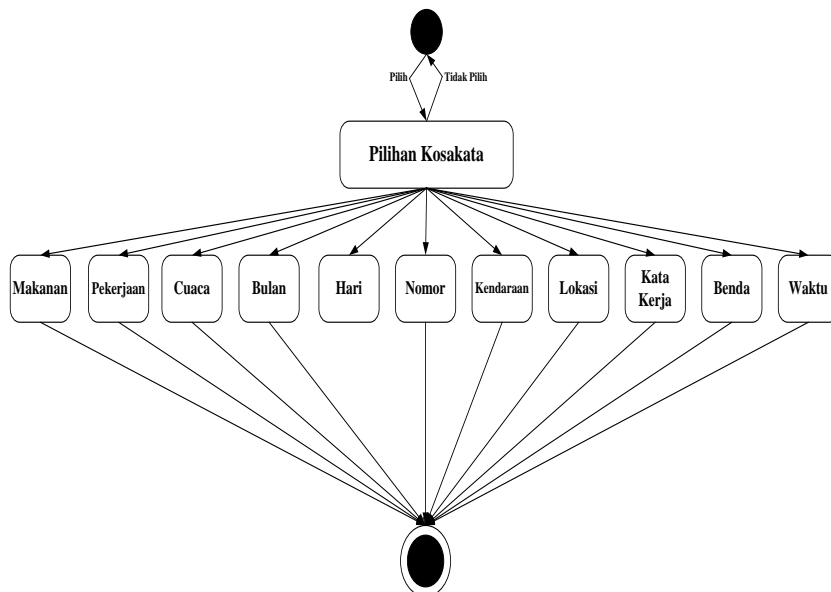
LAMPIRAN

STATECHART DIAGRAM

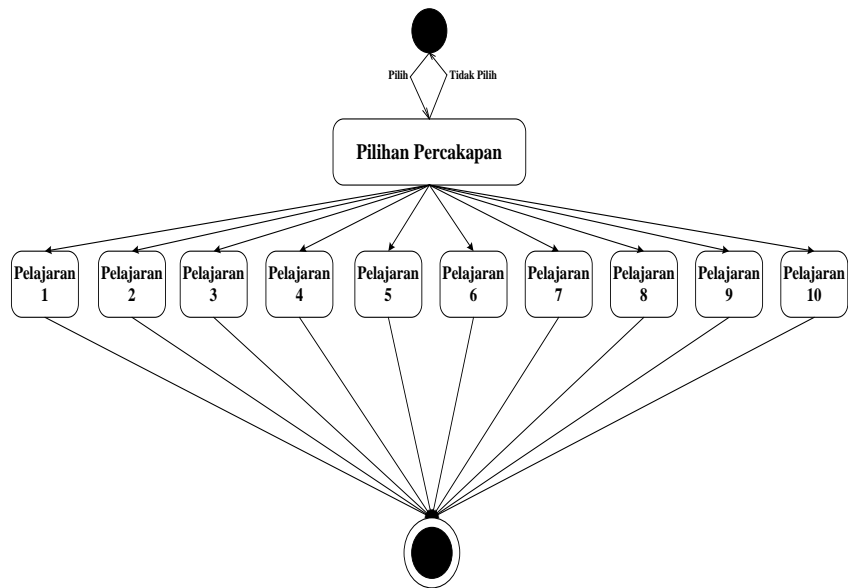
1. State Chart Diagram Huruf Jepang



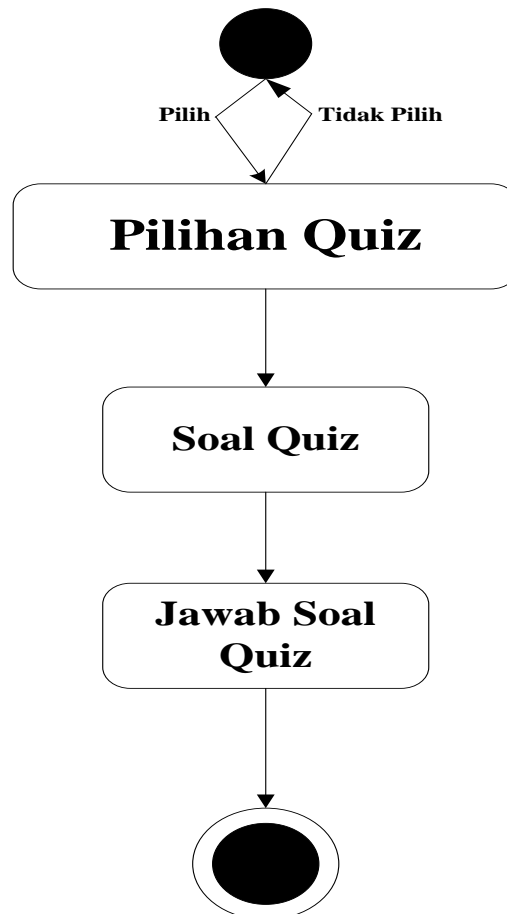
2. State Chart Diagram Kosakata



3. State Chart Diagram Percakapan



4. State Chart Diagram Quiz



LISTING PROGRAM

1. Huruf Jepang

a. Hiragana.html

```

<!DOCTYPE html>
<html lang="id">
<head>

    <title>Belajar Bahasa Jepang – Daftar Suku Kata Bahasa Jepang |
NHK WORLD RADIO JAPAN</title>

    <meta property="og:title" content="Belajar Bahasa Jepang –
Daftar Suku Kata Bahasa Jepang | NHK WORLD RADIO
JAPAN">

    <!-- /title -->
    <!-- css js -->
    <link rel="stylesheet" href="lesson/common/css/global-
parts.css">
    <link rel="stylesheet" href="lesson/common/css/base.css"
type="text/css">
    <link rel="stylesheet" href="lesson/common/css/content2.css"
type="text/css">
    <link rel="stylesheet" href="lesson/common/css/id.css"
type="text/css">

    <script src="hkworld/common/js/jquery-2.min.js"
charset="utf-8"></script>
    <script src="hkworld/common/js/common.js" charset="utf-
8"></script>
    <script src="hkworld/common/js/global-parts.js" charset="utf-
8"></script>
    <script src="hkworld/common/js/jquery.flexnav.min.js"
charset="utf-8"></script>

```

```

<script src="hkworld/common/js/jquery.matchHeight-min.js"
charset="utf-8"></script>
<!-- kana sound -->
<link rel="stylesheet" href="lesson/common/css/syllabary.css"
type="text/css">
<script src="hkworld/common/js/jquery.jplayer.min.js"
charset="utf-8"></script>
<script src="hkworld/common/js/syllabary.js"></script>

<!--[if IE 9]
<link rel="stylesheet" href="lesson/common/css/ie.css"
type="text/css">
<script src="hkworld/common/js/ie.js" charset="utf-8"></script>

<script src="hkworld/common/js/le_menu_id.js" charset="utf-
8"></script>
<!-- /css js -->
</head>
<body id="lesson-syllabary" class="lesson-content2">
<!-- mainBox -->
<div id="mainBox">

<!-- ***** -
->
<!-- ***** LESSON START ***** -
->
<!-- topic path -->

<div id="lesson-box">
<!-- EJ head -->
<!-- /EJ head -->

```

```

<!-- EJ body -->
<div id="ej-body">
  <div class="ej-wrapper">

    <!-- content -->
    <div id="content">
      <h1 align="center"><b>HIRAGANA</b></h1>
      <h1 id="con-title">Daftar Suku Kata Bahasa
Jepang</h1>

      <!-- lead box -->
      <div id="lead-box">
        <div class="lead-inner clearfix">
          
          <p>Bahasa Jepang punya 3 jenis karakter: Hiragana,
Katakana dan Kanji. Hiragana dan Katakana adalah simbol fonetik,
masing-masing mewakili satu suku kata. Mempelajari suku-suku kata
Jepang adalah langkah pertama untuk membaca dan menulis.</p>
          </div>
        </div>

      <!-- tab -->
      <nav id="tab" class="clearfix">
        <ul class="clearfix">
          <li class="active"><a
href="hiragana.html">Hiragana</a></li>
          <li><a href="katakana.html">Katakana</a></li>

```

```

        </ul>
    </nav>
    <div id="w-box">
        <p class="lead">Klik untuk memastikan
    pengucapannya! </p>

    <!-- tab syllabary -->
    <nav id="tab-syllabary" class="spDisplay
    clearfix">
        <ul class="clearfix">
            <li class="tab-syllabary-btn01 active"><a
    href="hiragana.html#page1">halaman 1</a></li>
            <li class="tab-syllabary-btn02"><a
    href="hiragana.html#page2">halaman 2</a></li>
        </ul>
    </nav>

    <!-- syllabary -->
    <div id="sy-box">
        <div id="page1" class="sy-inner">
            
            <ul id="hiragana1" class="kana-list">
                <!-- aa -->
                <li>
                    <div class="letter">
                        <div id="jquery_jplayer_0" class="jp-
    jplayer"></div>
                        <input id="jp_sound_0" type="hidden"

```

```

value="0">
    <div id="jp_container_0" class="jp-
audio" role="application" aria-label="media player">
        <div class="jp-play sy-play-button1">
あ</div>
            </div>
        </div>
    <div class="letter">
        <div id="jquery_jplayer_1" class="jp-
jplayer"></div>
        <input id="jp_sound_1" type="hidden"
value="1">
            <div id="jp_container_1" class="jp-
audio" role="application" aria-label="media player">
                <div class="jp-play sy-play-button1">
い</div>
                    </div>
                </div>
            <div class="letter">
                <div id="jquery_jplayer_2" class="jp-
jplayer"></div>
                <input id="jp_sound_2" type="hidden"
value="2">
                    <div id="jp_container_2" class="jp-
audio" role="application" aria-label="media player">
                        <div class="jp-play sy-play-button1">
う</div>
                            </div>
                        </div>
                    <div class="letter">
                        <div id="jquery_jplayer_3" class="jp-

```

```

jplayer"></div>
        <input id="jp_sound_3" type="hidden"
value="3">
        <div id="jp_container_3" class="jp-
audio" role="application" aria-label="media player">
            <div class="jp-play sy-play-button1">
                え</div>
            </div>
        </div>
    </li>

```

b. Katakana.html

```

<!DOCTYPE html>
<html lang="id">
<head>
    <title>Belajar Bahasa Jepang – Daftar Suku Kata Bahasa Jepang |
NHK WORLD RADIO JAPAN</title>
    <!-- /title -->
    <!-- css js -->
    <link rel="stylesheet" href="lesson/common/css/global-
parts.css">
    <link rel="stylesheet" href="lesson/common/css/base.css"
type="text/css">
    <link rel="stylesheet" href="lesson/common/css/content2.css"
type="text/css">
    <link rel="stylesheet" href="lesson/common/css/id.css"
type="text/css">

    <script src="hkworld/common/js/jquery-2.min.js"
charset="utf-8"></script>

```

```

<script src="hkworld/common/js/common.js" charset="utf-
8"></script>
<script src="hkworld/common/js/global-parts.js" charset="utf-
8"></script>
<script src="hkworld/common/js/jquery.flexnav.min.js"
charset="utf-8"></script>
<script src="hkworld/common/js/jquery.matchHeight-min.js"
charset="utf-8"></script>

<!-- kana sound -->
<link rel="stylesheet" href="lesson/common/css/syllabary.css"
type="text/css">
<script src="hkworld/common/js/jquery.jplayer.min.js"
charset="utf-8"></script>
<script
src="hkworld/common/js/jplayer/syllabary.js"></script>

<!--[if IE 9]
<link rel="stylesheet" href="lesson/common/css/ie.css"
type="text/css">
<script src="hkworld/common/js/ie.js" charset="utf-8"></script>

<script src="hkworld/common/js/le_menu_id.js" charset="utf-
8"></script>
<!-- /css js -->
</head>
<body id="lesson-syllabary" class="lesson-content2">
<!-- mainBox -->
<div id="mainBox">

<!-- ***** _

```

```

->
<!-- ***** LESSON START ***** -
->
<!-- topic path -->
<div id="lesson-box">

<div id="ej-body">
<div class="ej-wrapper">

<!-- content -->
<div id="content">

<h1 align="center"><b>KATAKANA </b></h1>
<h1 id="con-title">Daftar Suku Kata Bahasa
Jepang</h1>

<!-- lead box -->
<div id="lead-box">
<div class="lead-inner clearfix">

<p>Bahasa Jepang punya 3 jenis karakter: Hiragana,
Katakana dan Kanji. Hiragana dan Katakana adalah simbol fonetik,
masing-masing mewakili satu suku kata. Mempelajari suku-suku kata
Jepang adalah langkah pertama untuk membaca dan menulis.</p>
</div>
</div>

<!-- tab -->
<nav id="tab" class="clearfix">

```



```

<ul class="clearfix">
  <li><a href="hiragana.html">Hiragana</a></li>
  <li class="active"><a
href="katakana.html">Katakana</a></li>
</ul>
</nav>
<div id="w-box">
  <p class="lead">Klik untuk memastikan
pengucapannya! </p>

<!-- tab syllabary -->
<nav id="tab-syllabary" class="spDisplay
clearfix">
  <ul class="clearfix">
    <li class="tab-syllabary-btn01 active"><a
href="#page1">halaman 1</a></li>
    <li class="tab-syllabary-btn02"><a
href="#page2">halaman 2</a></li>
  </ul>
</nav>

<!-- syllabary -->
<div id="sy-box">

  <div id="page1" class="sy-inner">
    
    <ul id="katakana1" class="kana-list">
      <!-- aa -->

```

```

<li>
  <div class="letter">
    <div id="jquery_jplayer_0" class="jp-
jplayer"></div>
    <input id="jp_sound_0" type="hidden"
value="0">
    <div id="jp_container_0" class="jp-
audio" role="application" aria-label="media player">
      <div class="jp-play sy-play-button1">
ア </div>
    </div>
  </div>
  <div class="letter">
    <div id="jquery_jplayer_1" class="jp-
jplayer"></div>
    <input id="jp_sound_1" type="hidden"
value="1">
    <div id="jp_container_1" class="jp-
audio" role="application" aria-label="media player">
      <div class="jp-play sy-play-button1">
イ </div>
    </div>
  </div>
  <div class="letter">
    <div id="jquery_jplayer_2" class="jp-
jplayer"></div>
    <input id="jp_sound_2" type="hidden"
value="2">
    <div id="jp_container_2" class="jp-
audio" role="application" aria-label="media player">
      <div class="jp-play sy-play-button1">

```

```

ウ</div>
    </div>
</div>
<div class="letter">
    <div id="jquery_jplayer_3" class="jp-
jplayer"></div>
    <input id="jp_sound_3" type="hidden"
value="3">
    <div id="jp_container_3" class="jp-
audio" role="application" aria-label="media player">
        <div class="jp-play sy-play-button1">
エ</div>
    </div>
</div>
<div class="letter">
    <div id="jquery_jplayer_4" class="jp-
jplayer"></div>
    <input id="jp_sound_4" type="hidden"
value="4">
    <div id="jp_container_4" class="jp-
audio" role="application" aria-label="media player">
        <div class="jp-play sy-play-button1">
オ</div>
    </div>
</div>
</li>

```

2. Kosakata

a. benda.java

```

package com.iainbsk.linafaizahfitri.bjepang.kosakata;

import android.app.Activity;

```

```

import android.content.res.TypedArray;
import android.os.Bundle;
import android.support.v4.app.ListFragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import com.iainbsk.linafaizahfitri.bjepang.CustomArrayAdapter;
import com.iainbsk.linafaizahfitri.bjepang.R;

public class benda extends ListFragment {

    Activity mActivity;
    private CustomArrayAdapter myCustomArrayAdapter;

    public static benda newInstance() {
        benda fragment = new benda();
        return fragment;
    }

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        mActivity = activity;
    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup
container, Bundle savedInstanceState) {
        View rootView = inflater.inflate(R.layout.activity_page1,
container, false);
        return rootView;
    }

    @Override
    public void onActivityCreated(Bundle savedInstanceState) {
        super.onActivityCreated(savedInstanceState);

        TypedArray values_imgs =

```

```

getResources().obtainTypedArray(R.array.bendaimgs);
    int len = values_imgs.length();
    int[] resIds = new int[len];
    for (int i = 0; i < len; i++)
        resIds[i] = values_imgs.getResourceId(i, 0);

    values_imgs.recycle();

    String[] values = getResources().getStringArray(R.array.benda);
    String[] values_description =
getResources().getStringArray(R.array.bendadescription);
    String[] values_description_2 =
getResources().getStringArray(R.array.bendadescription2);

    myCustomArrayAdapter = new
CustomArrayAdapter(getActivity(), R.layout.list_row, values,
values_description, values_description_2, resIds, 1);
    setListAdapter(myCustomArrayAdapter);
}
}

```

string.xml

```

//benda
<array name="bendaimgs">
    <item>@drawable/buku</item>
    <item>@drawable/kursi</item>
    <item>@drawable/meja</item>
    <item>@drawable/kipasangin</item>
    <item>@drawable/papan_tulis</item>
    <item>@drawable/pena</item>
    <item>@drawable/pensil</item>
    <item>@drawable/kertas</item>
    <item>@drawable/koran</item>
</array>

<string-array name="benda">
    <item>ほん</item>
    <item>いす</item>
    <item>つくえ</item>
    <item>せんぷき</item>
    <item>こくばん</item>
    <item>ぺん</item>
    <item>えんぴつ</item>

```

```

    <item>かみ</item>
    <item>しんぶん</item>
</string-array>

<string-array name="bendadescription">
    <item>hon</item>
    <item>isu</item>
    <item>tsukue</item>
    <item>senpuki</item>
    <item>kokuban</item>
    <item>pen</item>
    <item>enpitsu</item>
    <item>kami</item>
    <item>shinbun</item>
</string-array>

<string-array name="bendadescription2">
    <item>buku</item>
    <item>kursi</item>
    <item>meja</item>
    <item>kipas angin</item>
    <item>papan tulis</item>
    <item>pena</item>
    <item>pensil</item>
    <item>kertas</item>
    <item>koran</item>
</string-array>

```

3. Percakapan

PelajaranActivity.java

```
package com.iainbsk.linafaizahfitri.bjepang.penysunankalimat;
```

```

import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentPagerAdapter;
import android.support.v4.view.ViewPager;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.webkit.WebView;

```

```

import com.iainbsk.linafaizahfitri.bjepang.R;

public class PelajaranActivity extends AppCompatActivity {
    private static final String HTML_IMG_REGEX =
"(?i)<[/]?[ ]?img(.)*?>";
    private static final String BACKGROUND_COLOR = "#f6f6f6" ;
    private static final String QUOTE_BACKGROUND_COLOR =
"#e6e6e6" ;
    private static final String QUOTE_LEFT_COLOR = "#a6a6a6";
    private static final String TEXT_COLOR = "#000000" ;
    private static final String BUTTON_COLOR = "#52A7DF" ;
    private static final String SUBTITLE_COLOR = "#666666";
    private static final String SUBTITLE_BORDER_COLOR = "solid
#ddd";
    private static final String CSS = "<head>" +
    "<style type='text/css'> "
    + "body {max-width: 100%; margin: 0.3cm; font-family: sans-
serif-light; color: " + TEXT_COLOR + "; background-color:" +
BACKGROUND_COLOR + "; line-height: 150%} "
    + "* {max-width: 100%; word-break: break-word}"
    + "h1, h2 {font-weight: normal; line-height: 100%} "
    + "h1 {font-size: 140%; margin-bottom: 0.1em} "
    + "h2 {font-size: 110%} "
    + "a {color: #0099CC}" +
    "table {" +
    " border: 1px solid #ccc;" +
    " border-collapse: collapse;" +
    " margin: 0;" +
    " padding: 0;" +
    " width: 100%;" +
    " table-layout: fixed;" +
    "}" +
    "table caption {" +
    " font-size: 1.5em;" +
    " margin: .5em 0 .75em;" +
    "}" +
    "table tr {" +
    " background: #f8f8f8;" +
    " border: 1px solid #ddd;" +
    " padding: .35em;" +
    "}" +
    "table th," +
    "table td {" +
    " text-align: center;" +

```

```

}" +

```



```

    }"+
    ".container {"+
        " border: 2px solid #dedede;" +
        " background-color: #f1f1f1;" +
        " border-radius: 5px;" +
        " padding: 5px;" +
        " margin: 5px 0;" +
    }"+
    ""+
    ".darker {"+
        " border-color: #ccc;" +
        " background-color: #ddd;" +
    }"+
    ""+
    ".container::after {"+
        " content: '\\";" +
        " clear: both;" +
        " display: table;" +
    }"+
    ""+
    ".container img {"+
        " float: left;" +
        " max-width: 60px;" +
        " width: 100%;" +
        " margin-right: 20px;" +
        " border-radius: 50%;" +
    }"+
    ""+
    ".container img.right {"+
        " float: right;" +
        " margin-left: 20px;" +
        " margin-right: 0;" +
    }"+
    ""+
    ".time-right {"+
        " float: right;" +
        " color: #aaa;" +
    }"+
    ""+
    ".time-left {"+
        " float: left;" +
        " color: #999;" +
    }"

+ "h1 a {color: inherit; text-decoration: none}"
+ "img {height: auto} "

```

```

+ "pre {white-space: pre-wrap;} "
+ "blockquote {border-left: thick solid " +
QUOTE_LEFT_COLOR + "; background-color:" +
QUOTE_BACKGROUND_COLOR + "; margin: 0.5em 0 0.5em 0em;
padding: 0.5em} "
+ "p {margin: 0.8em 0 0.8em 0} "
+ "p.subtitle {color: " + SUBTITLE_COLOR + "; border-
top:1px " + SUBTITLE_BORDER_COLOR + "; border-bottom:1px "
+ SUBTITLE_BORDER_COLOR + "; padding-top:2px; padding-
bottom:2px; font-weight:800 } "
+ "ul, ol {margin: 0 0 0.8em 0.6em; padding: 0 0 0 1em} "
+ "ul li, ol li {margin: 0 0 0.8em 0; padding: 0} "
+ "div.button-section {padding: 0.4cm 0; margin: 0; text-align:
center} "
+ ".button-section p {margin: 0.1cm 0 0.2cm 0}"
+ ".button-section p.marginfix {margin: 0.5cm 0 0.5cm 0}"
+ ".button-section input, .button-section a {font-family: sans-
serif-light; font-size: 100%; color: #FFFFFF; background-color: " +
BUTTON_COLOR + "; text-decoration: none; border: none; border-
radius:0.2cm; padding: 0.3cm} "
+ "</style><meta name='viewport' content='width=device-
width'/></head>";
private static final String BODY_START = "<body>";
private static final String BODY_END =
"<br/><br/></body></html>";
private static final String TITLE_START = "<h1><a href='";
private static final String TITLE_MIDDLE = "'>";
private static final String TITLE_END = "</a></h1>";
private static final String SUBTITLE_START = "<p
class='subtitle'>";
private static final String SUBTITLE_END = "</p>";
private static final String BUTTON_SECTION_START = "<div
class='button-section'>";
private static final String BUTTON_SECTION_END = "</div>";
private static final String BUTTON_START = "<p><input
type='button' value='";
private static final String BUTTON_MIDDLE = "' onclick='";
private static final String BUTTON_END = "'/></p>";
// the separate 'marginfix' selector in the following is only needed
because the CSS box model treats <input> and <a> elements differently
private static final String LINK_BUTTON_START = "<p
class='marginfix'><a href='";
private static final String LINK_BUTTON_MIDDLE = "'>";
private static final String LINK_BUTTON_END =
"</a></p><br><br>";
private static final String IMAGE_ENCLOSURE = "[@]image/";

```

```

/**
 * The {@link android.support.v4.view.PagerAdapter} that will provide
 * fragments for each of the sections. We use a
 * {@link FragmentPagerAdapter} derivative, which will keep every
 * loaded fragment in memory. If this becomes too memory intensive, it
 * may be best to switch to a
 * {@link android.support.v4.app.FragmentStatePagerAdapter}.
 */
private SectionsPagerAdapter mSectionsPagerAdapter;

/**
 * The {@link ViewPager} that will host the section contents.
 */
private ViewPager mViewPager;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_pelajaran);

    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    // Create the adapter that will return a fragment for each of the three
    // primary sections of the activity.
    mSectionsPagerAdapter = new
SectionsPagerAdapter(getSupportFragmentManager());

    // Set up the ViewPager with the sections adapter.
    mViewPager = (ViewPager) findViewById(R.id.container);
    mViewPager.setAdapter(mSectionsPagerAdapter);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.menu_pelajaran, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.action_home) {

```

```

mViewPager.setCurrentItem(0);
    return true;
}
if (id == R.id.action_pelajaran1) {
mViewPager.setCurrentItem(1);
    return true;
}
if (id == R.id.action_pelajaran2) {
mViewPager.setCurrentItem(2);
    return true;
}
if (id == R.id.action_pelajaran3) {
    mViewPager.setCurrentItem(3);
    return true;
}
if (id == R.id.action_pelajaran4) {
    mViewPager.setCurrentItem(4);
    return true;
}
if (id == R.id.action_pelajaran5) {
    mViewPager.setCurrentItem(5);
    return true;
}
if (id == R.id.action_pelajaran6) {
    mViewPager.setCurrentItem(6);
    return true;
}
if (id == R.id.action_pelajaran7) {
    mViewPager.setCurrentItem(7);
    return true;
}
if (id == R.id.action_pelajaran8) {
    mViewPager.setCurrentItem(8);
    return true;
}
if (id == R.id.action_pelajaran9) {
    mViewPager.setCurrentItem(9);
    return true;
}
if (id == R.id.action_pelajaran10) {
    mViewPager.setCurrentItem(10);
    return true;
}

return super.onOptionsItemSelected(item);
}

```

```

/**
 * A placeholder fragment containing a simple view.
 */
public static class PlaceholderFragment extends Fragment {
    /**
     * The fragment argument representing the section number for this
     * fragment.
     */
    private static final String ARG_SECTION_NUMBER =
"section_number";

    public PlaceholderFragment() {
    }

    /**
     * Returns a new instance of this fragment for the given section
     * number.
     */
    public static PlaceholderFragment newInstance(int sectionNumber)
    {
        PlaceholderFragment fragment = new PlaceholderFragment();
        Bundle args = new Bundle();
        args.putInt(ARG_SECTION_NUMBER, sectionNumber);
        fragment.setArguments(args);
        return fragment;
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup
container,
        Bundle savedInstanceState) {
        WebView webView;

        View rootView = inflater.inflate(R.layout.fragment_pelajaran,
container, false);
        //      TextView textView = (TextView)
rootView.findViewById(R.id.section_label);
        //      textView.setText(getString(R.string.section_format,
getArguments().getInt(ARG_SECTION_NUMBER)));
        webView = (WebView) rootView.findViewById(R.id.webView);
        webView.getSettings().setJavaScriptEnabled(true);
        webView.getSettings().setSupportZoom(true);
        webView.getSettings().setBuiltInZoomControls(true);
        webView.getSettings().setDisplayZoomControls(false);

```

```

//String title="<h1>Pelajaran
"+getArguments().getInt(ARG_SECTION_NUMBER)+"</h1>";
int nomor;
String contentText = null;
String title=null;
nomor=getArguments().getInt(ARG_SECTION_NUMBER);
int nomorpel=nomor-1;
title="<h1>Pelajaran "+nomorpel+"</h1>";
switch (nomor) {
    case 1:
        contentText = getString(R.string.pelajaran);
        title="<h1>Pelajaran </h1>";
        break;
    case 2:
        contentText = getString(R.string.pelajaran1);

        break;
    case 3:
        contentText = getString(R.string.pelajaran2);
        break;
    case 4:
        contentText = getString(R.string.pelajaran3);
        break;
    case 5:
        contentText = getString(R.string.pelajaran4);
        break;
    case 6:
        contentText = getString(R.string.pelajaran5);
        break;
    case 7:
        contentText = getString(R.string.pelajaran6);
        break;
    case 8:
        contentText = getString(R.string.pelajaran7);
        break;
    case 9:
        contentText = getString(R.string.pelajaran8);
        break;
    case 10:
        contentText = getString(R.string.pelajaran9);
        break;
    case 11:
        contentText = getString(R.string.pelajaran10);
        break;
}
// contentText = getString(R.string.pelajaran1);

```

```

        //webView.loadData(customHtml, "text/html", "UTF-8");
        String TEXT_HTML = "text/html";
        webView.loadDataWithBaseUrl("", generateHtmlContent(title,
contentText), TEXT_HTML, "UTF-8", null);
        return rootView;
    }

}

public class SectionsPagerAdapter extends FragmentPagerAdapter {

    public SectionsPagerAdapter(FragmentManager fm) {
        super(fm);
    }

    @Override
    public Fragment getItem(int position) {
        // getItem is called to instantiate the fragment for the given page.
        // Return a PlaceholderFragment (defined as a static inner class
below).
        return PlaceholderFragment.newInstance(position + 1);
    }

    @Override
    public int getCount() {
        // Show 3 total pages.
        return 11;
    }
}

private static String generateHtmlContent(String title, String
contentText) {
    StringBuilder content = new
StringBuilder(CSS).append(BODY_START);
    content.append(title);
    content.append(contentText);
    content.append(BODY_END);
    return content.toString();
}
}

```

string.xml

```

<string name="title_pelajaran">Home</string>
<string name="title_pelajaran1">Pelajaran 1</string>
<string name="title_pelajaran2">Pelajaran 2</string>
<string name="title_pelajaran3">Pelajaran 3</string>
<string name="title_pelajaran4">Pelajaran 4</string>
<string name="title_pelajaran5">Pelajaran 5</string>
<string name="title_pelajaran6">Pelajaran 6</string>
<string name="title_pelajaran7">Pelajaran 7</string>
<string name="title_pelajaran8">Pelajaran 8</string>
<string name="title_pelajaran9">Pelajaran 9</string>
<string name="title_pelajaran10">Pelajaran 10</string>

<string name="title_activity_pelajaran">PelajaranActivity</string>
<string name="title_home">Home</string>
<string name="title_dashboard">Dashboard</string>
<string name="title_notifications">Notifications</string>

<string name="section_format">Pelajaran: %1$d</string>
<string name="pelajaran" translatable="false">
  <![CDATA[ <p> Para Tokoh<br/>
    <img src=file:///android_asset/images/pemain.png />
    <br/>
  ]]>
</string>
<string name="pelajaran1" translatable="false"><![CDATA[
  <p>Anna adalah mahasiswi asing dari Thailand. Hari ini ia berjumpa
  pendampingnya di universitas untuk pertama kalinya.</p>
  <img src=file:///android_asset/images/le1_mp_01.jpg /><br/>
  <p>Kata kunci : <b>WATASHI WA ANNA DESU</b></p>

  <div class=container>
    <img src=file:///android_asset/images/anna.png alt=Avatar
    style=width:100% hight:30px;>
    <p>はじめまして。私はアンナです。
  <br/>HAJIMEMASHITE, WATASHI WA ANNA DESU.</p>
    <span class=time-right>Senang bertemu denganmu, Saya
  adalah Anna.</span>
    <audio controls>
    <source
  src=https://www.nhk.or.jp/lesson/update/mp3/le1_v_sc01.mp3>
    </audio>
  </div>
  <div class=container darker>
    <img src=file:///android_asset/images/sakura.png class=right

```



```

style=width:100%;>
    <p>はじめまして。さくらです。 <br/>HAJIMEMASHITE,
SAKURA DESU.</p>
    <span class=time-left>Hajimemashite.Saya Sakura.</span>
    <audio controls>
    <source
src=https://www.nhk.or.jp/lesson/update/mp3/le1_v_sc02.mp3>
    </audio>
</div>
<div class=container>
    <img src=file:///android_asset/images/anna.png alt=Avatar
style=width:100% hight:30px;>
    <p>よろしくお願ひします。 <br/>YOROSHIKU ONEGAI
SHIMASU.</p>
    <span class=time-right>Senang berkenalan dengan
Anda.</span>
    <audio controls>
    <source
src=https://www.nhk.or.jp/lesson/update/mp3/le1_v_sc03.mp3>
    </audio>
</div>
<div class=container darker>
    <img src=file:///android_asset/images/sakura.png class=right
style=width:100%;>
    <p>こちらこそ。 <br/>KOCHIRAKOSO.</p>
    <span class=time-left>Senang berjumpa dengan Anda
juga.</span>
    <audio controls>
    <source
src=https://www.nhk.or.jp/lesson/update/mp3/le1_v_sc04.mp3>
    </audio>
</div>
<p><b><i>Petunjuk tata bahasa</i></b></i></p>
<p><i>Kata Benda-A</i> <b>WA</b> <i>Kata Benda-B</i>
<b>DESU</b></p>
(A adalah B.)<br/>
WA adalah partikel penanda topik.<br/>
DESU setelah B membentuk predikat.<br/>
Contoh: WATASHI WA ANNA DESU. (Saya adalah Anna.)<br/></p>
]]></string>

```

4. Quiz

GameActivity.java

```

package com.iainbsk.linafaizahfitri.bjepang;

import android.content.Intent;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.graphics.Point;
import android.graphics.Typeface;
import android.graphics.drawable.Drawable;
import android.os.Bundle;
import android.support.v4.app.NavUtils;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.LinearLayout;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.Collections;
import java.util.logging.Level;
import java.util.logging.Logger;

public class GameActivity extends AppCompatActivity {

    TextView guess_text;
    LinearLayout main_layout;
    LinearLayout[] letter_line, answer_line;
    ImageView image;
    String[] images;
    int cur_image = 0;
    int score_image = 0;
    String correct_answer = "";
    ArrayList<Point> link_list;
    static final int letters_count = 21;
    int letters_in_answer = 0;

    /*Edit 25/4/2558 */
    /*prefs to save the game*/
    SharedPreferences save;

```

SharedPreferences.Editor **editor**;

@Override

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.game_layout);
    Toolbar toolbar= (Toolbar) findViewById(R.id.app_bar);
    setSupportActionBar(toolbar);

    getSupportActionBar().setHomeButtonEnabled(true);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    //initialization
    initView();

    loadImage(images[cur_image]);
    correct_answer =
    getUpperNameWithoutExtensionAndSpaces(images[cur_image]);
    fillLetterButtons(correct_answer);

    generateAnswerButtons(getNameWithoutExtension(images[cur_image]))
;

}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }
    if(id==android.R.id.home){
        NavUtils.navigateUpFromSameTask(this);
    }
    return super.onOptionsItemSelected(item);
}

void generateNewLevel(){
    score_image++;
    initLinking();
    cur_image++;
}
```

```

        loadImage(images[cur_image]);
        letters_in_answer = 0;
        correct_answer =
getUpperNameWithoutExtensionAndSpaces(images[cur_image]);
        fillLetterButtons(correct_answer);

generateAnswerButtons(getNameWithoutExtension(images[cur_image]))
;
    guess_text.setText("Benar" + ": " + score_image + "/" +
images.length);
    }

void skipNewLevel(){
    initLinking();
    cur_image++;
    loadImage(images[cur_image]);
    letters_in_answer = 0;
    correct_answer =
getUpperNameWithoutExtensionAndSpaces(images[cur_image]);
    fillLetterButtons(correct_answer);

generateAnswerButtons(getNameWithoutExtension(images[cur_image]))
;
    }

void initLinking() {
    link_list = new ArrayList<Point>();
    for (int i = 0; i < 3 * letters_count; i++) {
        link_list.add(i, new Point(0, 0));
    }
}

void initViews() {
    main_layout = (LinearLayout) findViewById(R.id.main_layout);
    answer_line = new LinearLayout[3];
    answer_line[0] = (LinearLayout)
findViewById(R.id.answer_line_1);
    answer_line[1] = (LinearLayout)
findViewById(R.id.answer_line_2);
    answer_line[2] = (LinearLayout)
findViewById(R.id.answer_line_3);
    letter_line = new LinearLayout[3];
    letter_line[0] = (LinearLayout) findViewById(R.id.letter_line_1);
    letter_line[1] = (LinearLayout) findViewById(R.id.letter_line_2);
    letter_line[2] = (LinearLayout) findViewById(R.id.letter_line_3);
}

```

```

image = (ImageView) findViewById(R.id.question_image);

initLinking();

Button button = (Button) findViewById(R.id.btn_skip);
button.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        skipNewLevel();
    }
});
/*Get SharedPreferences*/
/*Context.MODE_PRIVATE = 0*/
save = getSharedPreferences("SAVE_GAME", 0);
editor = save.edit();
if (save.contains("continue") && save.getBoolean("continue",
false)) {
    images = save.getString("images", null).replaceAll("\\",
""").split(",");
    cur_image = save.getInt("currentImage", 0);
    score_image = save.getInt("currentScore", 0);
    for (int i = 0; i < images.length; i++) {
        System.out.println(images[i]);
    }
} else {
    images = ShuffleImages(getImagesFromAssets());
    cur_image = 0;
    score_image = 0;
}

guess_text = (TextView) findViewById(R.id.guess_text);
guess_text.setText("Benar" + ": " + score_image + "/" +
images.length);

}

String[] getImagesFromAssets() {
    String[] img_files = null;
    try {
        img_files = getAssets().list("pictures");
    } catch (IOException ex) {

Logger.getLogger(GameActivity.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return img_files;
}

```

```

void loadImage(String name) {
    try {
        InputStream ims = getAssets().open("pictures/" + name);
        Drawable d = Drawable.createFromStream(ims, null);
        image.setImageDrawable(d);
    } catch (IOException ex) {
        return;
    }
}

void generateAnswerButtons(String answer) {
    //Remove all new images[cur_image]
    for (int i = 0; i < answer_line.length; i++) {
        answer_line[i].removeAllViews();
    }

    int cur_word = 0;
    String[] result_split = answer.split("_");
    for (int i = 0; i < result_split.length; i++) {
        if (cur_word == 0) {
            for (int j = 0; j < result_split[cur_word].length(); j++) {
                //System.out.println(j);
                addAnswerButton(answer_line[0]);
            }
        } else {
            for (int j = 0; j < result_split[cur_word].length(); j++) {
                addAnswerButton(answer_line[i]);
            }
        }
        cur_word++; //System.out.println(cur_word);
    }
    setOnAnswerButtonsClickListeners();
}

void addAnswerButton(LinearLayout l) {
    LinearLayout.LayoutParams params = new
    LinearLayout.LayoutParams(LinearLayout.LayoutParams.WRAP_CONTENT,
    LinearLayout.LayoutParams.WRAP_CONTENT);
    params.width = (int)
    getResources().getDimension(R.dimen.btn_size_width);
    params.height = (int)
    getResources().getDimension(R.dimen.btn_size_height);
    params.setMargins(5,5,5,5);
    Button b = new Button(this);
    b.setLayoutParams(params);
}

```

```

        b.setText("");
        b.setTextColor(Color.BLACK);

b.setTextSize(getResources().getDimension(R.dimen.text_answer_line));

b.setBackgroundColor(getResources().getColor(R.color.colorPrimaryLight));
    l.addView(b);
}

void fillLetterButtons(String name) {
    //clear letter lines
    for (int i = 0; i < letter_line.length; i++) {
        letter_line[i].removeAllViews();
    }
    String nameWithoutSpaces = name.toUpperCase().replaceAll("_",
    "");
    String alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    String letters = ShuffleLetters(nameWithoutSpaces
        + ShuffleLetters(alphabet).substring(0, letters_count -
nameWithoutSpaces.length()));
    for (int i = 0; i < letters_count; i++) {
        Button b = new Button(this);
        b.setText(Character.toString(letters.charAt(i)));
        LinearLayout.LayoutParams params = new
LinearLayout.LayoutParams(LinearLayout.LayoutParams.WRAP_CONTENT,
WRAP_CONTENT,
        LinearLayout.LayoutParams.WRAP_CONTENT);
        b.setLayoutParams(params);
        params.weight = 1;
        params.setMargins(5,5,5,5);
        Typeface tf = Typeface.createFromAsset(getAssets(),
fonts/Roboto-Regular.ttf);
        b.setTypeface(tf);
        b.setTextColor(Color.WHITE);

b.setTextSize(getResources().getDimension(R.dimen.text_letter_line));

b.setBackgroundColor(getResources().getColor(R.color.colorPrimary));

    if (i < letters_count / 3) {
        letter_line[0].addView(b);
    } else if ((letters_count / 3 <= i) && (i < 2 * letters_count / 3)) {
        letter_line[1].addView(b);
    } else if ((2 * letters_count / 3 <= i) && (i < letters_count)) {
        letter_line[2].addView(b);
    }
}

```

```

    }
    }
    setOnLettersClickListeners();
}

void setOnLettersClickListeners() {
    for (int i = 0; i < letter_line.length; i++) {
        for (int j = 0; j < letter_line[i].getChildCount(); j++) {
            final Button b = (Button) letter_line[i].getChildAt(j);
            final Point letter_pos = new Point(i, j);
            b.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    if (!b.getText().toString().equals("") &&
letters_in_answer < correct_answer.length()) {
                        letters_in_answer++;
                        System.out.println("Huruf: " + letters_in_answer);
//Debug letter_in_answer
                        int free_pos = getFirstFreeAnswerButtonPosition();
                        Button a = findButtonByPos(answer_line, free_pos);
                        a.setText(b.getText());
                        hideButton(b);
                        link_list.set(free_pos, letter_pos);
                        checkPlayersAnswer(letters_in_answer);
                    }
                }
            });
        }
    }
}

void checkPlayersAnswer(int current_pos){
    if(current_pos == correct_answer.length()){
        if(isCorrectAnswer()){
            Toast.makeText(getApplicationContext(), "Benar!!
"+getNameWithoutExtension(images[cur_image]),
Toast.LENGTH_LONG).show();
            if(score_image < images.length-1){
                generateNewLevel();
            }else{
                Intent intent = new
Intent(GameActivity.this,EndActivity.class);
                startActivity(intent);
            }
        }else{
            Toast.makeText(getApplicationContext(), "Salah!!!",

```



```

Toast.LENGTH_LONG).show();
    }
}

boolean isCorrectAnswer() {
    String player_answer="";
    for(int i =0; i< answer_line.length;i++){
        for(int j = 0; j<answer_line[i].getChildCount();j++){
            Button b = (Button)answer_line[i].getChildAt(j);
            player_answer = player_answer+b.getText().toString();
        }
    }
    if(player_answer.equals(correct_answer)){
        return true;
        //System.out.println(true);
    }else{
        //System.out.println(false);
        return false;
    }
}

int getFirstFreeAnswerButtonPosition() {
    int pos = 0;
    boolean isFound = false;
    for (int i = 0; i < answer_line.length; i++) {
        for (int j = 0; j < answer_line[i].getChildCount(); j++) {
            Button b = (Button) answer_line[i].getChildAt(j);
            if (b.getText().toString().equals("") && !isFound) {
                pos = getAnswerButtonPos(i, j);
                isFound = true;
            }
        }
    }
    return pos;
}

void setOnAnswerButtonsClickListeners(){
    for (int i = 0; i < answer_line.length; i++) {
        for (int j = 0; j < answer_line[i].getChildCount(); j++) {
            final Button b = (Button) answer_line[i].getChildAt(j);
            final int al_pos = getAnswerButtonPos(i, j);
            b.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v) {
                    if (!b.getText().toString().equals("")) {

```

```

        Button a = (Button)
letter_line[link_list.get(al_pos).x].getChildAt(link_list.get(al_pos).y);
        showButton(a, b.getText().toString());
        b.setText("");
        letters_in_answer--;
    }
    }
    });
}
}

int getAnswerButtonPos(int row, int col) {
    int pos = 1;
    for (int i = 0; i <= row; i++) {
        if (i < row) {
            pos += answer_line[i].getChildCount();
        } else {
            pos += col;
        }
    }
    return pos;
}

Button findButtonByPos(LinearLayout[] line, int pos) {
    if (pos <= line[0].getChildCount()) {
        return (Button) line[0].getChildAt(pos - 1);
    } else if (pos <= line[0].getChildCount() + line[1].getChildCount()) {
        return (Button) line[1].getChildAt(pos - line[0].getChildCount() -
1);
    } else {
        return (Button) line[2].getChildAt(pos - line[0].getChildCount() -
line[1].getChildCount() - 1);
    }
}

//delete .jpg or .png
String getNameWithoutExtension(String name) {
    return name.substring(0, name.indexOf('.'));
}

//All uppercase text and _ replace ""
String getUpperNameWithoutExtensionAndSpaces(String name) {
    return
getNameWithoutExtension(name.toUpperCase().replaceAll("_", ""));
}

String ShuffleLetters(String s){

```

```

    ArrayList<Character> char_list = new ArrayList<Character>();
    for(int i=0; i< s.length();i++){
        char_list.add(s.charAt(i));
    }
    Collections.shuffle(char_list);
    s = TextUtils.join("",char_list);
    return s;
}

String[] ShuffleImages(String[] imgs) {
    ArrayList<String> string_list = new ArrayList<String>();
    for (int i = 0; i < imgs.length; i++) {
        string_list.add(imgs[i]);
    }
    Collections.shuffle(string_list);
    return string_list.toArray(imgs);
}

void hideButton(Button b) {
    b.setText("");
    b.setVisibility(View.INVISIBLE);
}

void showButton(Button b, String text) {
    b.setVisibility(View.VISIBLE);
    b.setText(text);
}

@Override
public void onBackPressed() {
    saveCurrentGame();
    Intent menu_intent = new Intent(GameActivity.this,
MainActivity.class);
    startActivity(menu_intent);
    finish();
}

@Override
public void onPause() {
    saveCurrentGame();
    super.onPause();
}

void saveCurrentGame() {
    editor.putBoolean("gameSaved", true);
    editor.putBoolean("gameSaved", true);
}

```

```
    editor.putString("images", arrayToString(images));
    editor.putInt("currentImage", cur_image);
    editor.putInt("currentScore", score_image);
    editor.commit();
}
String arrayToString(String[] name) {
    StringBuilder sb = new StringBuilder();
    for (String s : name) {
        sb.append(s).append(',');
    }
    if (name.length != 0) {
        sb.deleteCharAt(sb.length() - 1);
    }
    return sb.toString();
}
}
```