BAB II

LANDASAN TEORI

2.1. Pengertian Perangkat Lunak Aplikasi

Menurut Abdul Kadir dan Terra CH. Triwahyuni (Pengenalan Teknologi Informasi 2003:230) Perangkat lunak aplikasi (*application software*) adalah program yang biasa dipakai oleh pemakai untuk melakukan tugas-tugas yang spesifik, misalnya untuk membuat dokumen, memanipulasi data, atau membuat laporan keuangan.

Sebuah program komputer tanpa tersosialisasi dengan dokumentasinya maka belum disebut perangkat lunak aplikasi (*application software*). Sebuah perangkat aplikasi juga sering disebut dengan sistem perangkat lunak aplikasi. Sistem berarti kumpulan komponen yang saling terkait dan mempunyai satu tujuan yang ingin dicapai.

Sistem Perangkat Lunak Aplikasi berarti sebuah sistem yang memiliki komponen berupa perangkat lunak yang memiliki hubungan satu sama lain untuk memenuhi kebutuhan pelanggan (c*ustomer).* Pelanggan (c*ustomer)* adalah orang atau organisasi yang dengan sukarela mengeluarkan uang untuk memesan atau membeli perangkat lunak. *User* atau pemakai perangkat lunak adalah orang yang memiliki kepentingan untuk memakai atau menggunakan perangkat lunak untuk atau memudahkan pekerjaannya.

2.2. Pengertian Android

Menurut Nazruddin Safaat H (Pemrograman aplikasi *mobile smartphone* dan *tablet PC* berbasis android 2012:1) Android adalah sebuah sistem operasi pada *mobile* berbasis *linux* yang mencakup sistem operasi , *middleware* dan aplikasi.

Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan apilakasi mereka. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, *Google* Inc. membeli Android Inc., 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 perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode–kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Didunia 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 *Open Handset Distribution (OHD).*

Kelebihan-kelebihan Android dibandingkan dengan *mobile platform* yang ada dipasaran, yaitu :

1. Merupakan pengembang platform bebas yang berbasis linux dan *open source.*
2. Sebuah komponen berbasis arsitektur yang terinspirasi oleh *mashup* internet. Bagian dari satu aplikasi dapat dikembangkan, digantikan, ditingkatkan komponennya sesuai dengan versi yang diinginkan sehingga dapat memunculkan berbagai kreativitas pada *mobile application.*
3. Memiliki layanan berbasis lokasi menggunakan GPS atau triangulasi menara sel untuk menyesuaikan lokasi Anda berada.
4. Memungkinkan Anda untuk memanfaatkan kekuatan penyimpanan lokal untuk berhubungan dengan komputasi dan sinkronisasi.
5. Android dioptimalkan untuk perangkat daya rendah dan rendah memori, yang ada pada *platform* sebelumnya tidak terpikirkan.
6. Mendukung grafis dan suara berkualitas tinggi dan lebih halus.
7. Memiliki probabilitas diberbagai perangkat keras saat ini dan masa depan.
8. Program yang terisolasi dapat memberikan tingkat stabilitas sistem yang tidak terlihat sebelumnya pada *smartphone.*
9. Android menawarkan berbagai cara untuk berinteraksi dengan *mobile application* bersama dengan dasar-dasar teknis untuk membuatnya.
10. Dalam pengembangannya, hanya diperlukan satu komputer yang ter-*install SDK* Android dan emulator.

Adapun kekurangan-kekurangan Android adalah sebagai berikut:

1. *Handset* Android dibuat oleh banyak vendor dengan ukuran layar yang berbeda, sebagian menggunakan *keyboard* fisik, sebagian lagi hanya menggunakan *touch screen.* Versi Android yang terpasang di *handset* juga berbeda-beda, hal ini menyebabkankan pengalaman pemakai Android tidak seragam, hal ini juga sekaligus menyulitkan *programmer* dalam membuat program, karena apabila versi sistem operasi tidak sama, program tersebut tidak bisa di-*install*.
2. Android adalah produk *open source* dari Google dan Open Handset Alliance, karena itu aplikasi yang ada dalam Android sangat menyatu dengan layanan Google, contohnya *email* menggunakan Gmail, kalender menggunakan Google Calender, bahkan sampai *contact* pun disimpan di Gmail Contact. Bagi penggemar Google ini merupakan keunggulan sendiri, namum bagi pengguna yang tidak familiar dengan produk Google, hal ini cukup merepotkan karena mereka harus menyesuaikan keadaan.
3. Android cukup aktif dikembangkan, hampir tiap tahun mereka merilis system operasi baru, pada tahun 2010 Google merilis *Froyo (2.2), Gingerbread (2.3)* dan *Honeycomb (3.0),* pada tahun 2011 merilis *Ice Cream Sandwich (4.0).* Namun dari sisi *vendor* atau *operator,update* ke *user* membutuhkan waktu berbulan-bulan, sebagai contoh, Froyo dirilis pada Mei 2010, namun pengguna Android kebanyakan baru meng-*update* setelah September.

.

2.2.1. Android SDK

Menurut Nazruddin Safaat H (Pemrograman aplikasi *mobile smartphone* dan *tablet PC* berbasis android 2012:5) Android SDK (*Software Development Kit*) adalah tools API (*Application Programming Interface*) yang diperlukan untuk pengembangan atau pembangunan suatu aplikasi pada *platform* android menggunakan bahasa pemrograman *Java*.

Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* android menggunakan bahasa pemrograman *Java*. Sebagai *platform* aplikasi *netral*, android memberi kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan 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 opensource 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 berbagai format gambar (MPEG4, H.264, MP3,AAC, AMR, JPG, PNG, GIF), GSM *Telephony* (tergantung *hardware*).
g. GSM, *Bluetooth*, EDGE, 3G, dan WiFi (tergantung *hardware*).

h. Kamera, *Global Positioning System* (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*.

2.2.2. *ADT* (Android Development Tools)

Menurut Nazruddin Safaat H (Pemrograman aplikasi *mobile smartphone* dan *tablet PC* berbasis android 2012: 6) *ADT* (*Android Development Tools*) adalah *plugin* yang didesain untuk *IDE Eclipse* yang memberikan kita kemudahan dalam mengembangkan aplikasi android dengan menggunakan *IDE Eclipse.*

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 adalah versi ADT untuk eclipse yang sudah dirilis :

a. ADT 12.0.0 (*July* 2011)

b. ADT 11.0.0 (*June* 2011)

c. ADT 10.0.1 (*March* 2011)

d. ADT 10.0.0 (*February* 2011)

e. ADT 9.0.0 (*January* 2011)

f. ADT 8.0.1 (*December* 2010)

g. ADT 8.0.0 (*December* 2010)

h. ADT 0.9.9 (*September* 2010)

i. ADT 0.9.8 (*September* 2010)

j. ADT 0.9.7 (*May* 2010)

k. ADT 0.9.6 (*March* 2010)

l. ADT 0.9.5 (*December* 2009)

m. ADT 0.9.4 (*October* 2009)

2.2.3. Versi Android

Telepon pertama yang memakai system operasi android adalah *HTC Dream*, yang dirilis pada Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua *vendor seluler* di dunia mengggunakan android sebagai *operating system*.

Adapun versi-versi android yang pernah dirilis adalah sebagai berikut:

a) Versi 1.0 dirilis 23 September 2008

b) Versi 1.1 dirilis 9 Maret 2009

c) Versi 1.5 (*Cupcake*)

d) Versi 1.6 (*Donut*)

e) Versi 2.0 / 2.1 (*Eclair*)

f) Versi 2.2 (*Froyo:Frozen Yoghurt*)

g) Versi 2.3 (*Gingerbread*) yaitu versi yang digunakan saat membangun aplikasi ini.

h) Versi 3.0 / 3.1 / 3.2 (*Honeycomb*)

i) Versi 4.0 (*IceCreamSandwich*)

j) Versi 4.1 (*Jelly Bean*)

2.2.4. Database SQLite

Menurut Nazruddin Safaat H (Pemrograman aplikasi mobeli *smartphone* dan *tablet PC* berbasis *android* 2012:4) android memiliki fasilitas untuk membuat database yang dikenal dengan SQLite yaitu salah satu software yang embedded yang sangat popular, kombinasi SQL interface dan penggunaan memory yang sangat sedikit dengan kecepatan yang sangat cepat. SQLite di android termasuk dalam Android runtime, sehingga setiap versi dari android dapat membuat database dengan SQLite.

Dalam sistem android terdapat beberapa teknik untuk melakukan penyimpanan data. Teknik yang umum digunakan adalah sebagai berikut:

1. Shared prefences yaitu menyimpan data beberapa nilai (value) dalam bentuk groups key yang dikenal dengan prefences.
2. Files yaitu menyimpan data dalam file, dapat berupa menulis ke file atau membaca dari file.
3. SQLite Database, yaitu menyimpan data dalam bentuk database.
4. Content Providers, yaitu menyimpan data dalam bentuk content providers service.

2.2.5. Pengertian Eclipse

Menurut Nazruddin Safaat H (Pemrograman aplikasi mobeli *smartphone* dan *tablet PC* berbasis *android* 2012:16) *Eclipse* adalah sebuah IDE *(Integrated Development Environment)* untuk pengembangan java/android yang *free* dan dapat di-*download* di http://www.eclipse.org/downloads/, versi eclipse yang ada

sekarang sudah banyak seperti *Eclipse Helios* (eclipse versi 3.6), *Eclipse Galileo* (eclipse versi 3.5), dan *Eclipse Ganymede* (Eclipse versi 3.4), kita dapat melakukan instalasi salah satu versi, versi eclipse 3.4 sudah *support* dengan *Android Development Tools (ADT)* untuk membuat eclipse dapat digunakan untuk *coding project Android*.

Berikut ini adalah sifat dari Eclipse:

a. Multi-*platform* : Target sistem operasi Eclipse adalah *Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X*.
b. Multi-*language* : Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis Bahasa pemrogrman lainnya, seperti *C/C++*, *Cobol,Phyton,Perl,PHP* dan lain sebagainya.
c. *Multi-role* : Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, tes perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu *IDE* favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*

2.3. Pengertian Java

Menurut Abdul Kadir (Dasar Pemrograman Java 2 2005:2) Java adalah bahasa pemrograman serbaguna, Java dapat digunakan untuk membuat suatu program sebagaimana anda membuatnya dengan bahasa seperti *Pascal* atau *C++*.

Yang lebih menarik, Java juga mendukung sumber daya *Internet* yang saat ini populer, yaitu *World Wide Web* atau yang sering di sebut *web* saja. Java juga mendukung aplikasi klien/server, baik dalam jaringan *local (LAN)* maupun jaringan berskala luas *(WAN).*

Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari *Oracle* dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada *C* dan C++ namun dengan sintaksis model objek yang lebih sederhana. Aplikasi-aplikasi berbasis *Java* umumnya dikompilasi ke dalam *p-code (bytecode)* dan dapat dijalankan pada berbagai Mesin Virtual *Java (JVM). Java* merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi *Java* mampu berjalan di beberapa platformsistem operasi yang berbeda, *Java* dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini *Java* merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

2.4. Pengertian MySQL

Menurut Budi Raharjo (2011 :21) MySQL merupakan *software* RDBMS (atau server database) yang dapat mengelola database dengan sangata cepat, dapat menampung data dalam jumlah sangata besar, dapat diakses oleh banyak user (multi user), dan dapat melakukan suatu proses secara sinkron atau berbarengan (multi-threaded).

Saat ini, MySQL banyak digunakan diberbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai industry, baik industri kecil, menengah, maupun besar.

Lisensi MySQL terbagi menjadi dua. Anda dapat menggunakan MySQL sebagai produk *open source* dibawah GNU General Public Lisensi (gratis) atau dapat membeli lisensi dari versi komersialnya. MySQL versi komersial tentu memiliki nilai lebih atau kemampuan-kemampuan yang tidak disertakana pada versi gratis. Pada kenyataanya, untuk keperluan industri menengah kebawah, versi gratis masih dapat digunakan dengan baik.

Beberapa keunggulan dari MySQL yaitu :

a) Cepat, handal dan Mudah dalam penggunaannya
b) Didukung oleh berbagai bahasa
c) Mampu membuat tabel berukuran sangat besar
d) Lebih Murah
e) Melekatnya integrasi PHP dengan MySQL

2.5. Pengertian PHP

Menurut Abdul Kadir (2008 : 2) yang dimaksud dengan PHP merupakan singkatan dari *Hypertext Preprocessor*, Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server.

Hasilnyalah yang dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, Anda bisa menampilkan isi database ke halaman *web*. Pada prinsispnya

PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Clod Fusion*, ataupun Perl. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara command line. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun browser. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya. Yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi tool yang disebut "Personal Home Page". Paket inilah yang menjadi cikal bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI versi 2. Pada versi inilah pemograman PHP juga bisa berkomunikasi dengan database dan melakukan perhitunganperhitungan yang kompleks sambil jalan.

2.6. Pengertian JSON

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer.

Format ini dibuat berdasarkan bagian dariBahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data. JSON terbuat dari dua struktur :

- Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), table hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
- Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagailarik (*array*), vector (*vector*), daftar (*list*) atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya semua bahsa pemrograman *modern* mendkung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini.

2.7. Pengertian UML

Menurut Rosa A. S (Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek 2013:137) UML (*Unified Modeling Language*) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram.

UML (*Unified Modeling Language*) muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak yang berbasis objek *UML* merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. *UML* tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang

membutuhkan pemodelan. Bagian-bagian utama dari *UML* adalah *View*, *Diagram*, *model element*, dan *general mechanism*.

Adapun jenis Diagram dalam *UML* antara lain :

1) *Use case* Diagram

   Diagram *Use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. *Use ase* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa yang berhak menggunakan fungsi-fungsi itu.

2) *Activity* Diagram

   Diagram Aktivitas Menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahawa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

3) *Class* Diagram

   Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancanga di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

4) *Squence* Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario pada *use case*.

5) *Component* Diagram

Diagram komponen dibuat untuk menunjukan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem.

6) *Deployment* Diagram

Diagram *deployment* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

a. Sistem tambahan (*embedded system)* yang menggambarkan rancangan *device*, *node,* dan *hardware.*
b. Sistem *client/server.*
c. Sistem terdistribusi murni
d. Rekayasa ulang aplikasi

Sedangkan perangkat lunak yang mendukung pembuatan diagaram *UML* yaitu:

StarUML (http://*StarUML*.sourceforge.net/en/)

StarUML adalah sebuah proyek *Staropen source* untuk mengembangkan cepat, *fleksibel, extensible, featureful,* dan bebas-tersedia *UML* / *platform MDA* berjalan pada *platform Win32* dan *Win64*.Tujuan dari proyek *StarUML* adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga platform yang menarik adalah pengganti alat *UML* komersial seperti *Rational Rose*, dan sebagainya.

2.8. Pengertian Panduan

Menurut Ernawati Waridah, S.S (Kamus Bahasa Indonesia 2013:207) Panduan adalah petunjuk.

2.9. Pengertian Menulis

Menurut Ernawati Waridah, S.S (Kamus Bahasa Indonesia 2013:185) Menulis adalah membuat huruf dengan pena.

2.10. Pengertian Hiragana

Menurut Cuk Yuana (Step by Step Belajar Bahasa Jepang Dasar 2013 : xxiii) Hiragana adalah huruf abjad yang digunakan untuk menuliskan kosakata yang berasal dari Jepang sendiri.

Hiragana (ひらがな、平仮名) mulai digunakan secara luas pada abad ke-10 Masehi. Tulisan Hiragana ini adalah tulisan yang memiliki bentuk yang sangat halus yang juga pada zaman dahulu dikenal sebagai tulisan wanita atau onna de (女手), karena pada zaman dahulu para wanita di Jepang sering menggunakan huruf-huruf Hiragana untuk penulisan bahasa Jepang. Selain itu Hiragana

memiliki kegunaan tersendiri, seperti untuk menulis akhiran kata, menulis kata keterangan, digunakan untuk situasi yang formal, menulis untuk bahan bacaan anak-anak seperti buku teks, animasi dan komik serta perkataan dimana huruf Kanjinya lama tidak digunakan atau bahkan sudah tidak diketahui.

あ	い	う	え	お
か	き	く	け	こ
さ	し	す	せ	そ
た	ち	つ	て	と
な	に	ぬ	ね	の
は	ひ	ふ	へ	ほ
ま	み	む	め	も
や		ゆ		よ
ら	り	る	れ	ろ
わ				を
ん				

Gambar 2.1 Huruf Hiragana

Sumber : http://nihongo.as.ua.edu/hiragana.htm

2.11. Pengertian Katakana

Menurut Cuk Yuana (Step by Step Belajar Bahasa Jepang Dasar 2013 : xxiv) Katakana adalah huruf abjad yang digunakan untuk menuliskan kosakata yang berasal dari serapan bahasa asing (baik dari bahasa Inggris, Jerman, Indonesia, Spanyol dsb).

Katakana (カタカナ) kebalikan dari huruf Hiragana, karena Katakana ini sering digunakan para laki-laki di Jepang pada zaman dahulu untuk penulisan bahasa Jepang. Katakana diciptakan sekitar tahun 800 M Huruf-huruf Katakana

tentu saja berbeda dengan huruf-huruf Hiragana, huruf Hiragana terlihat atau memiliki bentuk yang sangat halus sedangkan huruf-huruf Katakana memiliki bentuk yang tegak dan lurus. katakana bisa digunakan untuk penulisan lambang bunyi atau pengucapan. Katakana juga sering digunakan untuk kata-kata bahasa asing yang sudah diserap menjadi bahasa Jepang.



Gambar 2.2 Huruf Katakana

Sumber : http://nihongo.as.ua.edu/katakana.htm