

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka dalam penelitian ini mengenai penilaian tampilan, menu dan fitur-fitur dari beberapa aplikasi yang berupa kamus Android yang diambil dari jurnal maupun naskah publikasi yang sudah ada. Dimulai dari menilai segi tampilan, menu dan fitur dari jurnal ilmiah karangan Kusumawijaya dan Sekarwati (2013) dengan judul Aplikasi Kamus Istilah Komputer Pada Perangkat *Mobile* Berbasis Android, kemudian jurnal Kurniasih (2014) dengan judul Aplikasi Kamus Istilah Komputer Berbasis Android, dan yang terakhir Ayuningtyas (2011) dengan judul Aplikasi Kamus Kebidanan Berbasis Android.

2.1.1. Aplikasi Kamus Istilah Komputer Pada Perangkat *Mobile* Berbasis Android (Ike Putri Kusumawijaya dan Kemal Ade Sekarwati)

Aplikasi Kamus Istilah Komputer yang diciptakan oleh Kusumawijaya dan Sekarwati merupakan aplikasi kamus yang memuat istilah-istilah seputar ilmu komputer yang di dalamnya terdapat 4 menu utama seperti cari istilah, daftar istilah, info dan keluar. Pada menu cari istilah menampilkan fitur pencarian yang dapat digunakan oleh *user* untuk mencari istilah yang diinginkan, selanjutnya untuk menu daftar istilah menampilkan informasi daftar istilah, menu info menampilkan informasi petunjuk penggunaan aplikasi dan juga informasi tentang aplikasi, serta yang terakhir menu keluar yang berfungsi untuk menutup aplikasi. Aplikasi ini memiliki *splash screen* dan juga terdapat 100 istilah di dalamnya. Dilihat secara keseluruhan untuk pewarnaan tampilan tidak terlalu mencolok dan menarik untuk dilihat. Tampilan dari menu utama aplikasi kamus istilah komputer disajikan pada Gambar 2.1.



Gambar 2.1. *Interface* Menu Utama Aplikasi Kamus Istilah Komputer

2.1.2. Aplikasi Kamus Istilah Komputer Berbasis Android (Kurniasih)

Aplikasi Kamus Istilah Komputer berbasis Android yang diciptakan oleh Kurniasih merupakan aplikasi kamus yang memuat istilah-istilah komputer yang di dalamnya terdapat 3 menu utama seperti cari istilah, tentang, dan petunjuk. Pada menu cari istilah menampilkan fitur pencarian yang dapat digunakan oleh *user* untuk mencari istilah yang diinginkan, selanjutnya untuk menu tentang menampilkan informasi penjelasan aplikasi yang dibuat serta informasi versi Android yang digunakan untuk dapat menginstall aplikasi dan yang terakhir menu petunjuk menampilkan informasi petunjuk penggunaan aplikasi. Berbeda dari referensi sebelumnya aplikasi ini tidak memiliki *splash screen* sehingga ketika *user* pertama kali menjalankannya maka langsung menuju ke halaman utama, dilihat secara keseluruhan untuk tampilan tidak terlalu mencolok dan cukup menarik untuk dilihat. Tampilan dari menu utama Aplikasi Kamus Istilah Komputer disajikan pada Gambar 2.2.



Gambar 2.2. *Interface* Menu Utama Aplikasi Kamus Istilah Komputer

2.1.3. Aplikasi Kamus Kebidanan Berbasis Android (Vinanda U'un Ayuningtyas)

Aplikasi kamus kebidanan berbasis Android yang diciptakan oleh Ayuningtyas merupakan aplikasi kamus yang memuat istilah-istilah seputar ilmu kebidanan yang di dalamnya terdapat 3 menu utama seperti kamus kebidanan, profile, dan list kebidanan. Pada menu kamus kebidanan menampilkan fitur pencarian yang dapat digunakan oleh *user* untuk mencari istilah yang diinginkan, menu profile menampilkan informasi profil dari pembuat aplikasi, serta menu yang terakhir yaitu menu list kebidanan menampilkan daftar istilah kebidanan. Aplikasi ini memiliki *splash screen* di dalamnya. Dilihat secara keseluruhan pada sisi pewarnaan tampilan terlihat sangat sederhana dan kurang menarik untuk dilihat. Tampilan dari menu utama Aplikasi Kamus Istilah Kebidanan disajikan pada Gambar 2.3.



Gambar 2.3. *Interface* Menu Utama Aplikasi Kamus Kebidanan

Pada Tabel 2.1 akan ditampilkan tabel perbandingan dari ketiga jurnal ilmiah yang menjadi bahan referensi untuk pembuatan aplikasi Kamus Istilah Komputer Berbasis Android.

Tabel 2.1. Tabel Perbandingan Aplikasi Kamus Jurnal Ilmiah

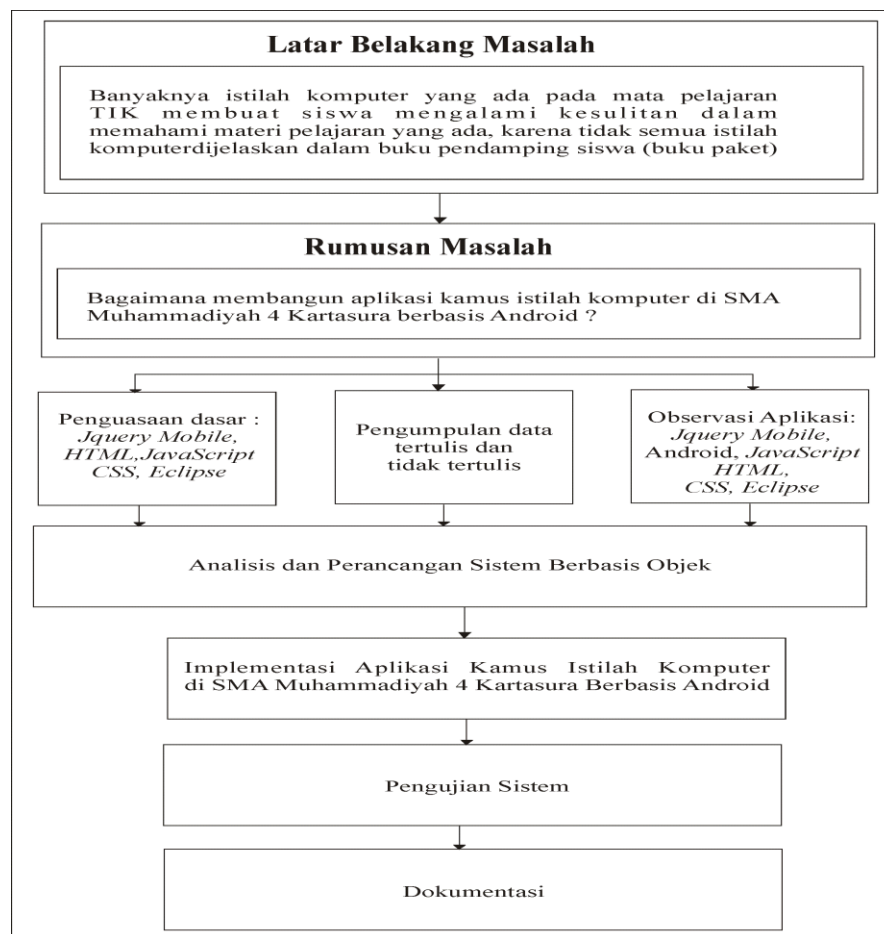
No	Judul	Menu Utama	Fitur	Software	Bahasa
1.	Aplikasi Kamus Istilah Komputer Pada Perangkat <i>Mobile</i> Berbasis Android	Daftar Istilah, Info, Keluar	Pencarian Istilah, Terdapat 100 Istilah, <i>Splash Screen</i>	<i>Eclipse IDE, JDK, Android SDK, AVD Android 2.1.</i>	<i>Java</i>
2.	Aplikasi Kamus Istilah Komputer Berbasis Android	Cari Istilah, Tentang, Petunjuk	Pencarian Istilah	<i>Eclipse IDE, ADT, Android SDK</i>	<i>Java</i>
3.	Aplikasi Kamus Kebidanan Berbasis Android	Kamus Kebidanan Profile, List Kebidanan	<i>Splash Screen</i>	<i>Eclipse IDE, JDK, Android SDK</i>	<i>Java</i>
4.	Membangun Aplikasi Kamus Istilah Komputer Di SMA Muhammadiyah 4 Kartasura Berbasis Android	Daftar Istilah, Petunjuk, Tentang, Profil Sekolah	Pencarian Istilah, Terdapat 825 Istilah, <i>Splash Screen, Welcome Screen, Keluar</i>	<i>Eclipse IDE, JDK, Android SDK, Notepad+ +</i>	<i>Java, HTML, CSS, JavaScript</i>

Penelitian pada Tugas Akhir dengan judul Membangun Aplikasi Kamus Istilah Komputer Di SMA Muhammadiyah 4 Kartasura Berbasis Android. Aplikasi terdiri dari 4 menu utama yaitu menu daftar istilah, menu petunjuk, menu tentang, dan menu profil sekolah serta terdapat beberapa fitur seperti pencarian istilah, *splash screen*, *welcome screen*, keluar dan berisi 825 istilah komputer. *Software* yang digunakan dalam pembuatan aplikasi ini

adalah *Eclipse IDE*, *JDK*, *Android SDK*, *Notepad++* serta dengan bahasa pemrograman *Java*, *HTML*, *CSS*, *JavaScript*.

2.2 Kerangka Pemikiran

Kerangka pemikiran disajikan pada Gambar 2.4.



Gambar 2.4 Diagram Kerangka Pemikiran

Keterangan *Diagram Kerangka Pemikiran* :

1) Latar Belakang Masalah

Pokok permasalahan yang mendasari adalah banyaknya istilah yang ada pada mata pelajaran TIK (Teknologi Informasi dan Komunikasi) membuat siswa

mengalami kesulitan dalam memahami materi pelajaran yang ada, karena tidak semua istilah dijelaskan pada buku pendamping siswa (buku paket).

2) Rumusan Masalah

Bagaimana membangun aplikasi kamus istilah komputer di SMA Muhammadiyah 4 Kartasura Berbasis Android ?

3) Pengumpulan Data Tertulis dan Tidak Tertulis

Mengumpulkan semua data yang dibutuhkan, baik melalui *interview* dengan siswa dan guru TIK (Teknologi Informasi dan Komunikasi) yang bersangkutan, observasi atau studi literatur di SMA Muhammadiyah 4 Kartasura.

4) Penguasaan Dasar (*Jquery Mobile, JavaScript, HTML, CSS, Eclipse*)

Melakukan beberapa percobaan membuat aplikasi sederhana dengan menggunakan *Jquery Mobile* maupun *Eclipse* dengan tujuan agar dapat lebih memahami bahasa pemrograman *JavaScript, HTML, CSS* sehingga hasilnya menjadi lebih maksimal.

5) Observasi Aplikasi

Mengamati dan membandingkan beberapa aplikasi yang sudah ada, baik dari karya ilmiah, buku, atau *internet* yang dapat dijadikan referensi untuk membangun aplikasi.

6) Analisis dan Perancangan Sistem

Menganalisis dan merancang aplikasi yang akan dibangun seperti apa, bagaimana desainnya, dan apa saja isinya, sehingga aplikasi ini dapat membantu memecahkan permasalahan yang ada di SMA Muhammadiyah 4 Kartasura.

7) Implementasi

Membangun aplikasi kamus istilah komputer berbasis Android sesuai dengan data-data yang telah di dapatkan dari SMA Muhammadiyah 4 Kartasura.

8) Pengujian Sistem

Pengujian sistem ini dilakukan bertujuan untuk mengetahui jika ternyata masih ada kesalahan atau kekurangan pada sistem yang dibuat.

9) Dokumentasi

Tahap akhir, dimana aplikasi telah siap digunakan setelah melewati tahap pengujian dan membuat dokumentasi dari keseluruhan kegiatan penyusunan tugas akhir.

2.3 Landasan Teori

2.3.1 *Smartphone*

Smartphone merupakan *cellphone* yang menggabungkan fungsi-fungsi *Personal Digital Assistant (PDA)* seperti kalender, *personal schedule*, *address book*, dan memiliki kemampuan untuk mengakses *internet*, membuka *email*, membuat dokumen, bermain *game*, serta membuka aplikasi lainnya. Istilah *smartphone* merupakan istilah yang digunakan untuk mendeskripsikan *mobile device* yang menggabungkan fungsi *cellphone*, *audio player*, *digital camera*, *camcorder*, *Global Positioning System (GPS) receiver*, dan *Personal Computer (PC)* (Wijayanto, 2015).

2.3.2 Aplikasi

Aplikasi berasal dari kata *application* yang artinya penerapan, lamaran, penggunaan. Secara istilah aplikasi adalah program siap pakai yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain dan dapat digunakan oleh sasaran yang dituju (Ahmad, 2013:10 dikutip oleh Susilo, 2015).

Adapun karakteristik aplikasi adalah sebagai berikut :

- 1) Aplikasi merupakan elemen sistem logik dan bukan elemen sistem fisik seperti perangkat keras.
- 2) Elemen aplikasi bisa rusak.
- 3) Elemen aplikasi bisa direkayasa atau dikembangkan dan bukan dibuat di pabrik, seperti halnya perangkat keras.
- 4) Aplikasi tidak bisa dirakit atau disusun.

2.3.3 Kamus Istilah

Kamus merupakan koleksi referensi yang memuat informasi yang dibutuhkan. Kamus memuat kata dari satu bahasa atau subyek tertentu yang disusun menurut

abjad dengan memberikan keterangan yang berhubungan dengan aspek bahasa seperti arti, pengucapan, asal, pengejaan dan sebagainya (Parno dkk, 2011).

Kamus istilah termasuk kategori kamus khusus karena merujuk kepada kamus yang mempunyai fungsi khusus. Kamus ini berisi istilah-istilah khusus dalam bidang tertentu, fungsinya untuk kegunaan ilmiah (Kurniasih, 2014).

2.3.4 Android

Android merupakan sebuah sistem operasi pada *handphone* yang bersifat terbuka dan berbasis pada sistem operasi *Linux*. Android bisa digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam piranti bergerak. Awalnya, Google Inc membeli Android Inc, pendatang baru yang membuat piranti lunak untuk ponsel. 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 dibawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Adapun ikon dari sistem operasi Android sering disebut dengan sebutan robot hijau (Fadlullah, 2012:25 dikutip oleh Susilo, 2015).

2.3.4.1 Perkembangan versi- versi Android

Maulana (2014) memaparkan penjelasan berbagai macam versi Android seperti dibawah ini:

2.3.4.1.1 Android versi 1.1

Android versi ini dirilis pada tanggal 9 Maret 2009. Android versi 1.1 merupakan Android awal yang dimana versi ini baru memberikan sentuhan di beberapa aplikasinya seperti sistem antar muka bagi pengguna (*user interface*) yang lebih baik, serta beberapa aplikasi yang lain.

2.3.4.1.2 Android versi 1.5 (*Cupcake*)

Pada pertengahan Mei 2009 Android kembali mengalami perubahan versi. Android versi 1.1 kemudian disempurnakan dengan Android versi 1.5 atau yang dikenal dengan sebagai Android *Cupcake*. Perubahan yang terjadi pada sistem operasi Android *Cupcake* bisa dibilang cukup banyak, seperti mempunyai kemampuan untuk merekam dan menonton video melalui *camcorder*, dapat meng-*upload* video ke *Youtube* dan *upload* gambar ke *Picasa*, penambahan fitur *soft-keyboard* dengan fungsi *text prediction*, adanya *Bluetooth A2DP* dan *AVRCP support*, mempunyai kemampuan otomatis untuk terhubung ke perangkat *Bluetooth* dengan jarak tertentu serta penambahan *Widget* baru dan folder dapat dikumpulkan di layar *home*.

2.3.4.1.3 Android versi 1.6 (*Donut*)

Android ini dirilis pada September 2009. Versi ini memperbaiki kesalahan *reboot* dalam OS sebelumnya. Selain itu Android *Donut* memiliki fitur-fitur tambahan seperti peningkatan pada Android *market*, integrasi kamera, perekam video, dan tampilan galeri. Aplikasi galeri yang baru memungkinkan pengguna memilih banyak foto untuk dihapus. Aplikasi *voice search* yang diperbaharui menjadi lebih cepat merespon dan integrasi dengan aplikasi yang lain termasuk kemampuan mencari kontak, aplikasi *search* yang ditingkatkan untuk bisa mencari *bookmarks*, *history*, kontak, dan *web* dari layar *home*, peningkatan dukungan teknologi untuk *CDMA/EVDO.802.1x*, *VPNs* dan mesin *text to speech*, mendukung resolusi layar *WVGA* serta perbaikan kecepatan di aplikasi pencarian dan aplikasi kamera.

2.3.4.1.4 Android versi 2.0/ 2.1 (*Eclair*)

Android *Eclair* dirilis pada 3 Desember 2009. Perubahan yang ada antara lain adalah pengoptimalan *hardware*, peningkatan *Google Maps 3.1.2*, perubahan UI (*User Interface*) dengan *browser* baru dan dukungan *HTML 5*, daftar kontak yang baru, dukungan *flash* untuk kamera 3,2 MP, digital *Zoom*, dan *Bluetooth 2.1*. Android *Eclair* merupakan Android pertama yang mulai dipakai oleh banyak *smartphone*, fitur utama *Eclair* yaitu perubahan total struktur dan tampilan *user interface*, peningkatan pada aplikasi virtual *keyboard*, *Bluetooth 2.1*, *Live Wallpapers*.

2.3.4.1.5 Android versi 2.2 *Froyo (Frozen Yoghurt)*

Android versi 2.2 dirilis pada 20 Mei 2010. Sistem operasi dengan julukan *Froyo* ini melakukan beberapa *update* dan juga pembenahan pada tampilannya seperti optimalisasi kecepatan dan performa Android OS, integrasi dengan *google chrome* versi 8 *javascript* ke dalam aplikasi *browser*, peningkatan dukungan *microsoft exchange*, peningkatan aplikasi *launcher* dengan *shortcuts* menuju aplikasi *phone* dan *browser*, USB *tethering* dan *WiFi hotspot* functionality, penambahan pilihan untuk menonaktifkan akses data jaringan *mobile*, aplikasi Android *market* yang telah diperbaharui dengan fitur *update* otomatis, berbagai kontak melalui *bluetooth*, mendukung file *upload* di aplikasi *browser* dan dukungan terhadap aplikasi *adobe flash*.

2.3.4.1.6 Android versi 2.3 (*Gingerbread*)

Android *Gingerbread* dirilis pada 6 Desember 2010. Perubahan-perubahan umum yang di dapat dari Android versi ini antara lain peningkatan kemampuan permainan (*gaming*), peningkatan fungsi *copy paste*, layar antar muka (*User Interface*) yang di desain ulang, dukungan format *video* VP8 dan *WebM*, efek *audio* baru (*reverb*, *equalization*, *headphone virtualization* dan *bass boost*), dukungan kemampuan *Near Field Communication* (NFC), dan dukungan jumlah kamera yang lebih dari satu.

2.3.4.1.7 Android versi 3.0 (*Honeycomb*)

Android *Honeycomb* dirancang khusus untuk Tablet PC. Android versi ini mendukung ukuran layar yang lebih besar. *User Interface* pada *Honeycomb* juga berbeda karena sudah didesain untuk tablet. *Honeycomb* juga mendukung *multiprocessor* dan juga akselerasi perangkat keras (*hardware*) untuk grafis.

2.3.4.1.8 Android versi 4.0 (*Ice Cream Sandwich*)

Pada tanggal 19 Oktober 2011, diperkenalkan Android versi 4.0 yang membawa fitur *Honeycomb* untuk *smartphone*, menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan pengguna dan kontrol, kontak jaringan sosial terpadu, perangkat tambahan fotografi, pencarian

email secara *offline*, dan berbagai informasi dengan menggunakan NFC. Ponsel pertama yang menggunakan sistem operasi ini adalah Samsung Galaxy Nexus.

2.3.4.1.9 Android versi 4.1 (*Jellybean*)

Beberapa fitur yang ada dalam sistem operasi ini antara lain *Photo Sphere* yang dapat membuat gambar yang besar seperti halnya pada *Google Street View*. Pada dasarnya fitur ini merupakan pengembangan dari *Photo Panorama*, *Gesture Typing* (hampir sama dengan aplikasi pihak ketiga yang bernama *Swipe*, *Multiple User Support* (mendukung banyak pengguna untuk *login*, jadi pada saat *device* digunakan secara bergantian, pengguna tidak perlu menghapus akun Google yang telah dimiliki sebelumnya), *Wireless Screen Sharing*, *Daydream*, *Expandable Notifications* (penyempurnaan pada fitur sebelumnya yaitu pemakai *device* bisa langsung menanggapi maupun berinteraksi langsung dengan notifikasi dari sebuah aplikasi hanya dari panel notifikasi) dan *Google Now*.

2.3.4.1.10 Android versi 4.4 (*Kitkat*)

Android *Kitkat* dirilis pada tanggal 28 Oktober 2013. Android *Kitkat* mempunyai beberapa pembaruan yang lebih baik dari versi sebelumnya seperti perintah suara lebih sederhana dan pintar karena pengguna tidak harus menyentuh ponsel untuk melakukan pencarian, mengirim pesan, atau menerima arahan dari *Google Map*, pada menu utama pengguna hanya perlu mengatakan ' OK Google ' untuk memberikan perintah dengan kerja spesifik yang presisi kemudian adanya *Multi-Tasking* yang lebih cepat, visual yang lebih baik pada versi Android sebelumnya, Selain itu, ada tombol tambahan pada kunci layar untuk memungkinkan pengguna bermain *game* dan bisa langsung masuk ke dalam menu galeri pada waktu bersamaan, adanya *Photo Editor* pada Android *Kitkat* pun diperbarui. Terdapat beberapa fitur efek, *filter*, dan *tools* untuk menyesuaikan gambar. Hebatnya, Photo Editor juga memungkinkan pengguna mengembalikan hasil editan foto ke bentuk semula, *Google Hangouts* juga ditambahkan pada sistem operasi ini. Hangouts mampu mengelompokkan semua SMS, MMS, *video call*, dan *chatting* dengan Google Hangouts pada satu tempat.

2.3.5 *HTML (Hypertext Markup Language)*

HTML adalah kependekan dari *Hyper Text Markup Language*, merupakan sebuah bahasa *scripting* yang berguna untuk menuliskan halaman *web*. Pada halaman *web*, *HTML* dijadikan sebagai bahasa *script* dasar yang berjalan bersama berbagai bahasa *scripting* pemrograman lainnya.

Semua *tag-tag HTML* bersifat dinamis, artinya kode *HTML* tidak dapat dijadikan sebagai *fileexecutable* program. Hal ini disebabkan *HTML* hanyalah sebuah bahasa *scripting* yang dapat berjalan apabila dijalankan di dalam *browser* (pengaksesan *web*), *browser-browser* yang mendukung *HTML* antara lain adalah *Internet Explorer*, *Netscape Navigator*, *Opera*, *Mozilla* dan lain-lain. Jadi pada saat ingin membuka halaman yang berasal dari *HTML* dapat dilihat dari bentuk pengkodeanya dengan cara mengklik menu *view source*, maka disana akan ditampilkan semua *tag* beserta isi dari halaman *web* tersebut (Nugroho, 2004 dikutip oleh Yuniyanti, 2016).

2.3.6 *CSS (Cascading Style Sheets)*

CSS (Cascade Style Sheet) adalah sebuah fitur yang diperkenalkan sejak *HTML* versi 4.0 dan berfungsi untuk menangani masalah tampilan pada *HTML* seperti jenis, ukuran dan warna *font*, posisi teks, batas tulisan atau *margin*, warna *background*, dan sebagainya.

Dari sisi manajemen dan perawatan, penggunaan *CSS* dipandang lebih praktis karena para *web developer* tidak perlu membuka setiap *file* dalam sebuah situs untuk melakukan perubahan.

Hal penting yang perlu diperhatikan adalah cara meletakkan *CSS* dan juga bahasa berbasis *web* lain untuk memudahkan manajemen *file*, *editing*, dan *maintenance*. Banyak di antara *programmer web* yang belum menyadari aspek-aspek penting ini menyisipkan *CSS*, *JavaScript*, *VB Script*, *PHP*, maupun *ASP* langsung ke dalam dokumen *HTML (embedded script)* (Madcoms, 2008 dikutip oleh Yuniyanti, 2016).

2.3.7 *JQuery Mobile*

Jquery adalah *Javascript Library*, kumpulan kode atau fungsi *Javascript* siap pakai, sehingga memudahkan dan mempercepat *developer* dalam membuat kode *JavaScript*. *JQuery Mobile* adalah cara aman agar tetap bisa bertahan di dunia *mobile programming*, karena dukungan *multi-platform* mulai dari Android, IOS sampai Blackberry, tidak hanya untuk perangkat *mobile* tapi bisa untuk membuat *responsive website* serta digunakan oleh 57% *website* di seluruh dunia dengan kecepatan pertumbuhan yang stabil dan selalu meningkat tanpa menunjukkan pengurangan (Gliser, 2013 dikutip oleh Ajie, 2014).

2.3.8 *SDK (Software Development Kit)*

SDK (Software Development Kit) merupakan alat bantu dan API (*Application Programming Interface*) yang diperlukan untuk memulai pengembangan aplikasi pada *platform* Android menggunakan bahasa pemrograman *Java*.

Pada Android SDK ini terdiri dari *debugger*, *libraries*, *handset emulator*, dokumentasi dan tutorial. SDK memungkinkan pengembang untuk membuat aplikasi untuk *platform* Android. SDK Android mencakup proyek sampel dengan kode sumber, perangkat pengembangan, emulator dan perpustakaan yang diperlukan untuk membangun bahasa pemrograman *Java* dan berjalan di Dalvik, mesin virtual yang dirancang khusus untuk penggunaan *embedded* yang berjalan di atas kernel Linux (Fadlullah, 2011:18 dikutip oleh Susilo, 2015)

2.3.9 *JDK (Java Development Kit)*

Java Development Kit (JDK) merupakan sebuah produk yang dikembangkan oleh *Oracle* yang ditujukan untuk para *developer java*. JDK dilengkapi dengan banyak komponen untuk melakukan pemrograman. JDK juga berisi paket *Java Runtime Environment (JRE)* yang lengkap, biasanya disebut *private runtime* dari JRE reguler dan dilengkapi dengan konten tambahan, yaitu terdiri atas *Java Virtual Machine* dan semua *class library* yang ada di *environment* produk dan juga *library* tambahan yang berguna untuk *developer* (Fadlullah, 2011 dikutip oleh Susilo, 2015).

2.3.10 JavaScript

JavaScript diperkenalkan pertama kali oleh *Netscape* pada tahun 1995. Pada awalnya bahasa yang sekarang disebut *JavaScript* ini dulunya dinamai “*LiveScript*” yang berfungsi sebagai bahasa sederhana untuk *browser Netscape Navigator 2* yang sangat populer pada saat itu. Kemudian sejalan dengan sedang giatnya kerjasama antara *Netscape* dan *Sun* (pengembang bahasa pemrograman “*Java*”) pada masa itu, maka *Netscape* memberikan nama “*JavaScript*” kepada bahasa tersebut pada tanggal 4 Desember 1995. Pada saat yang bersamaan *Microsoft* sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “*Jscript*” di *browser* milik mereka yaitu *Internet Explorer 3*. *JavaScript* sendiri merupakan modifikasi dari bahasa pemrograman C++ dengan pola penulisan yang lebih sederhana dari bahasa pemrograman C++.

JavaScript adalah bahasa pemrograman yang sederhana karena bahasa ini tidak dapat digunakan untuk membuat aplikasi ataupun *applet*. Dengan *JavaScript developer* dapat dengan mudah membuat sebuah halaman *web* yang interaktif. Program *JavaScript* dituliskan pada file HTML (*.htm*.html)) (Hardjono, 2006:4 dikutip oleh Nandari dan Sukadi, 2014).

2.3.11 ADT(Android Development Tools)

ADT merupakan kepanjangan dari *Android Development Tools* yang menjadi penghubung antara IDE Eclipse dengan Android SDK. ADT ini adalah sebuah *plugin* untuk Eclipse yang didesain untuk membangun aplikasi Android baru, membuat *user interface*, menambahkan komponen berdasarkan *framework* API Android, *debug* aplikasi dan menjalankan emulator Android (Fadlullah, 2011:18 dikutip oleh Susilo, 2015).

2.3.12 IDE Eclipse

Eclipse adalah sebuah *IDE (Integrated Development Environment)* untuk mengembangkan perangkat lunak agar dapat dijalankan di semua *platform (platform-independent)*. Berikut ini adalah sifat dari *Eclipse* :

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

Eclipse pada saat ini merupakan salah satu *IDE* favorit dikarenakan gratis dan bersifat *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* adalah kemampuannya untuk dapat dikembangkan oleh *user* dengan komponen yang dinamakan *plug-in* (Almuzakki, 2013).



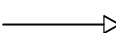



2.4 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’ (Nugroho, 2010). Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks demikian rupa sehingga lebih mudah dipelajari dan dipahami.

2.4.1 Use Case Diagram

Use case diagram yang menggambarkan *actor, use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML *use case* (Haviluddin, 2011). Simbol-simbol yang digunakan pada *use case diagram* disajikan pada Tabel 2.2.

Tabel 2.2 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Aktor</i>	Idealization orang eksternal, proses, atau hal yang berinteraksi dengan sistem, subsistem atau kelas.
2.		<i>Use case</i>	Sebuah <i>use case</i> menggambarkan interaksi dengan <i>actor</i> sebagai urutan pesan antara sistem dan aktor satu atau lebih.
3.		<i>System Boundary</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
4.		<i>Generalization</i>	Hubungan antara <i>use case</i> umum dan <i>use case</i> yang lebih spesifik yang mewarisi dan menambahkan fitur.
5.		<i>Comunication Association</i>	Jalur komunikasi antara <i>actor</i> dan <i>usecase</i> yang berpartisipasi didalam
6.		<i>Extend</i>	Penyisipan perilaku tambahan kedalam basis <i>use case</i> yang tidak tahu tentang hal itu.
7.		<i>Include</i>	Penyisipan perilaku tambahan kedalam basis <i>use case</i> yang secara eksplisit menggambarkan penyisipan.

Sumber : Kusumo (2004:3)

2.4.2 *Class Diagram*

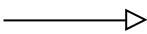

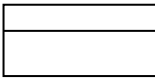

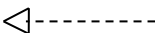
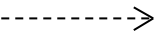
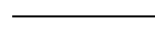
Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class* diagram membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Selama tahap desain, *class* diagram berperan dalam menangkap struktur dari semua kelas yang membentuk arsitektur sistem yang dibuat (Haviluddin, 2011).

Class memiliki tiga area pokok :

- 1) Nama (dan *stereotype*)
- 2) Atribut
- 3) Metoda

Simbol-simbol yang digunakan pada *class diagram* disajikan pada Tabel 2.3

Tabel 2.3 Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagai perilaku dan struktur data dari objek yang ada diatas objek induk (<i>ancestor</i>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari dua objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang terbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung pada elemen yan tidak mandiri.
7.		<i>Association</i>	Untuk menghubungkan objek satu dengan objek yang lainnya.





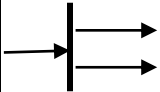
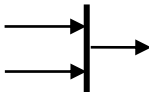
Sumber : Kusumo (2004:3)

2.4.3 *Activity Diagram*

Activity diagram menggambarkan aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku

sistem untuk aktivitas (Haviluddin, 2011). Simbol-simbol yang digunakan pada *activity diagram* disajikan pada Tabel 2.4

Tabel 2.4 Simbol *Activity Diagram*


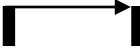

No	Gambar	Nama	Keterangan
1.		<i>Activity State</i>	Aktivitas yang mewakili pelaksanaan dalam pernyataan dalam prosedur atau pelaksanaan kegiatan dalam alur kerja.
2.		<i>Branch/Merge</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
3.		<i>Initial State</i>	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
4.		<i>Final State</i>	Status akhir yang dilakukan sistem.
5.		<i>Fork</i>	Percabangan yang menunjukkan aliran pada activity diagram.
6.		<i>Join</i>	Penggabungan yang menjadi arah aliran pada activity diagram.

Sumber : Kusumo (2004:4)

2.4.4 *Sequence Diagram*

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence* diagram adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case* diagram (Haviluddin, 2011). Simbol-simbol yang digunakan pada *sequence* diagram ditampilkan pada Tabel 2.5.

Tabel 2.5 Simbol *Sequence Diagram*

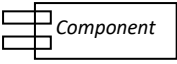
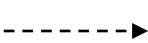
No	Gambar	Nama	Keterangan
1.		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Message</i>	<i>Message</i> ditampilkan sebagai anak panah dari lifeline dari satu objek ke objek yang lain.
3.		<i>Return</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.

Sumber : Kusumo (2004:4)

2.4.5 *Component Diagram*

Component diagram menggambarkan struktur fisik dari kode, pemetaan pandangan logis dari kelas proyek untuk kode aktual di mana logika ini dilaksanakan (Haviluddin, 2011). Simbol-simbol yang digunakan pada *component diagram* disajikan pada Tabel 2.6.

Tabel 2.6 Simbol *Component Diagram*

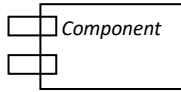
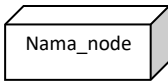

No	Gambar	Nama	Keterangan
1.		<i>Component</i>	Sebuah komponen melambangkan sebuah entitas <i>software</i> dalam sebuah sistem.
2.		<i>Dependency</i>	Sebuah <i>Dependency</i> digunakan untuk menotasikan relasi antara dua komponen.

Sumber : Kusumo (2004:4)

2.4.6 *Deployment Diagram*

Deployment diagram memberikan gambaran dari arsitektur fisik perangkat lunak, perangkat keras, dan artefak dari sistem. *Deployment* diagram dapat dianggap sebagai ujung spektrum dari kasus penggunaan, menggambarkan bentuk fisik dari sistem yang bertentangan dengan gambar konseptual dari pengguna dan perangkat berinteraksi dengan sistem (Haviluddin, 2011). Simbol-simbol yang digunakan pada *deployment diagram* disajikan pada Tabel 2.7.

Tabel 2.7 Simbol *Deployment Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Component</i>	Komponen-komponen yang ada diletakkan didalam <i>node</i> .
2.		<i>Node</i>	<i>Node</i> menggambarkan bagian-bagian <i>hardware</i> dalam sebuah sistem.
3.		<i>Association</i>	Sebuah <i>association</i> digambarkan sebagai sebuah garis yang menghubungkan dua <i>node</i> yang mengindikasikan jalur komunikasi antara element-elemen <i>hardware</i> .

Sumber : Kusumo (2004:4)

2.5 Pengujian *Five View*

Pengujian *five view* adalah pengujian yang sifatnya deskriptif dimana *software* yang diuji dinilai melalui lima sudut pandang atau kategori yang berbeda melalui penilaian dari *expertise* atau orang yang berpengalaman dari masing-masing sudut pandang (Naik dan Tripaty, 2008). Lima sudut pandang tersebut adalah sebagai berikut:

2.5.1 *User View*

Kualitas menyangkut sejauh mana produk memenuhi kebutuhan dan harapan pengguna dan apakah suatu produk cocok untuk digunakan. Pendapat ini bersifat sangat *personal*. Sebuah produk berkualitas baik jika memuaskan sebagian besar pengguna. Hal ini berguna untuk mengidentifikasi fitur dari produk yang pengguna anggap penting. Pandangan ini dapat mencakup banyak unsur subjek, seperti kegunaan, keandalan, dan ke-efisiensi.

2.5.2 *Value-based View*

Value-base View merupakan penggabungan dari dua konsep yaitu keunggulan dan kelayakan. Kualitas adalah ukuran dari keunggulan dan nilai adalah ukuran layak. Beberapa banyak pengguna bersedia membayar untuk tingkat kualitas tertentu.

Kualitas tidak berarti jika produk memenuhi nilai ekonomi. Pandangan berbasis nilai antara biaya dan kualitas.

2.5.3 *Manufacturing View*

Pandangan ini berkaitan dengan faktor dalam industri manufaktur, apakah produk memenuhi persyaratan atau tidak. Setiap penyimpangan dari persyaratan yang dinilai mengurangi kualitas produk. Konsep proses memainkan peran kunci. Produk yang dibuat harus orisinal sehingga biaya berkurang, misal biaya pembangunan dan biaya pemeliharaan.

Kesesuaian dengan persyaratan dan spesifikasi menyebabkan keseragaman dalam produk tapi beberapa berpendapat bahwa keseragaman tersebut tidak menjamin kualitas. Kualitas produk dapat secara bertahap ditingkatkan dengan memperbaiki proses.

2.5.4 *Product View*

Jika sebuah produk diproduksi dengan sifat internal (misalnya bahan dan tindakan) yang baik, maka produk akan memiliki sifat eksternal atau *output* yang baik dan dapat dieksplorasi hubungan antara sifat internal dan kualitas eksternal.

2.5.5 *Transcendental View*

Kualitas menurut pandangan ini adalah sesuatu yang dapat dikenali melalui pengalaman tapi tidak dapat selalu digambarkan. Objek atau *software* yang bagus itu menonjol dan dapat dengan mudah dikenali.