

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Aplikasi**

Aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Contoh utama aplikasi adalah pengolah kata, lembar kerja, memanipulasi foto, merancang rumah dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket disebut sebagai suatu paket atau *suite* aplikasi (*application suite*). Contohnya adalah Microsoft Office dan OpenOffice.org, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja dan beberapa aplikasi lainnya. Aplikasi-aplikasi dalam suatu paket biasanya memiliki atarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah. Jenis-jenis *Software* Aplikasi :

- a. *Software* aplikasi hiburan, contohnya yaitu winamp untuk mendengarkan musik, games dan sebagainya untuk hiburan.
- b. *Software* aplikasi pendidikan yaitu *software* digunakan untuk mempelajari atau mereferensikan tentang pendidikan atau pengetahuan.

- c. *Software* aplikasi bisnis yaitu *software* yang digunakan untuk aplikasi bisnis.
- d. *Software* aplikasi khusus
- e. *Software* aplikasi untuk produktivitas kerja. (Abdullah dan Erlina ; 2012 : 38-39)

## **II.2. Gambar**

Secara matematis gambar merupakan fungsi menerus dari intensitas cahaya pada bidang dua dimensi. Suatu gambar digital terbentuk dari kumpulan titik pembentuk gambar yang disebut dengan *pixel*. *Pixel* merupakan sampel dari pemandangan yang mengandung intensitas cahaya yang dinyatakan dalam bilangan bulat. Sehingga gambar *digital* adalah kumpulan *pixel* yang dinotasikan dalam bentuk bilangan pada sejumlah baris dan kolom. Gambar yang dimaksud di dalam penulisan ini adalah gambar diam, yaitu gambar tunggal yang tidak bergerak. (Eka Ardianto, et al. ; 2011 : 110-117)

## **II.3. Editor**

Menurut Kamus Besar Bahasa Indonesia (2008 : 374). Editor adalah para pengguna yang bertugas memeriksa suntingan dan memastikan bahwa kriteria dasar revisi diperiksa telah terpenuhi. Bila kriteria telah dipenuhi, maka Editor dapat menandai revisi tersebut sebagai revisi diperiksa.

#### **II.4. Foto**

Foto adalah sebuah hasil dari kegiatan pendokumentasian atau bisa dikatakan sebagai salah satu usaha manusia untuk membuat seakan-akan waktu berhenti sejenak dan kemudian merekamnya dalam satu bingkai. Kehadirannya dalam penyuguhan suatu produk media masa cetak yaitu berupa surat kabar telah mengambil fungsi dan perannya sendiri. Foto mampu hadir dengan kekuatan visual yang dapat menggantikan seribu kata. (Erlangga ; 2012 : 2)

#### **II.5. Android**

*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 piranti lunak untuk ponsel/*smartphone*. Kemudian untuk mengembangkan Android, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. (Murtiwiyati dan Lauren ; 2013 : 1-10)

#### **II.6. Eclipse**

*Eclipse* adalah sebuah IDE (Integrated Development Environment) untuk mengembangkan perangkat lunak dan 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 Mac OS X.

b. *Multi language* :

Eclipse dikembangkan dengan bahasa pemrograman Java. Akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.

c. *Multi role* :

Selain sebagai IDE untuk pengembangan aplikasi, Eclipse bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan *web* dan lain sebagainya. (Murtiwiyati dan Lauren ; 2013 : 1-10)

## **II.7. Android SDK (*Software Development Kit*)**

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Sebagai platform aplikasi netral, Android memberi kesempatan bagi semua orang untuk membuat aplikasi yang dibutuhkan, yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*. Beberapa fitur-fitur Android yang paling penting adalah :

- a. Mesin *Virtual Dalvik* yang dioptimalkan untuk perangkat *mobile*.
- b. *Integrated browser* berdasarkan *engine open source WebKit*.
- c. Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, grafis 3D berdasarkan spesifikasi *opengl ES 1.0* (Opsional akselerasi perangkat keras).

- d. SQLite untuk penyimpanan data (*database*).
- e. Media yang mendukung *audio*, *video*, dan gambar.
- f. *Bluetooth*, EDGE, 3G dan WiFi.
- g. Kamera, GPS, dan kompas.
- h. Lingkungan *Development* yang lengkap dan kaya termasuk perangkat *emulator*, *tools* untuk *debugging*, profil dan kinerja memori, dan *plugin* untuk IDE Eclipse. (Lindung , Yunus Dwi ; 2012)

## II.8. AVD (*Android Virtual Device*)

AVD (*Android Virtual Device*) merupakan *emulator* untuk menjalankan aplikasi android. Setiap AVD terdiri dari:

- a. Sebuah profil perangkat keras yang dapat mengatur pilihan untuk menentukan fitur *hardware emulator*. Misalnya, menentukan apakah menggunakan perangkat kamera, apakah menggunakan keyboard QWERTY fisik atau tidak, berapa banyak memori internal, dan lain-lain.
- b. Sebuah pemetaan versi Android, maksudnya kita menentukan versi dari platform Android akan berjalan pada *emulator*.
- c. Pilihan lainnya, misalnya menentukan *skin* yang kita ingin gunakan pada *emulator*, yang memungkinkan untuk menentukan dimensi layar, tampilan, dan sebagainya. Kita juga dapat menentukan *SD Card virtual* untuk digunakan dengan di *emulator*. (Lindung , Yunus Dwi ; 2012)

## II.9. XML

XML merupakan dasar terbentuknya *web service* yang digunakan untuk mendeskripsikan data. Pada *level* paling *detail web service* secara keseluruhan dibentuk diatas XML. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan komunikasi aplikasi eksternal dengan partner luaran. Dengan standarisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain. (Deviana ; 2011 : 61)

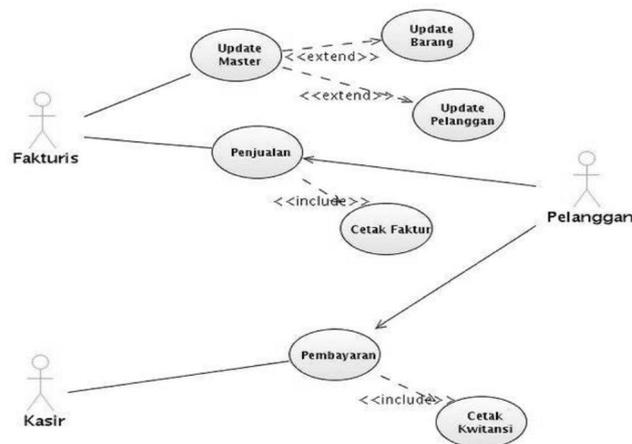
## II.10. Pengertian UML

UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodalan umum dalam industri perangkat lunak dan pengembangan sistem (Windu dan Grace ; 2013 : 81). Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

### II.10.1. Use Case Diagram

*Use case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Windu

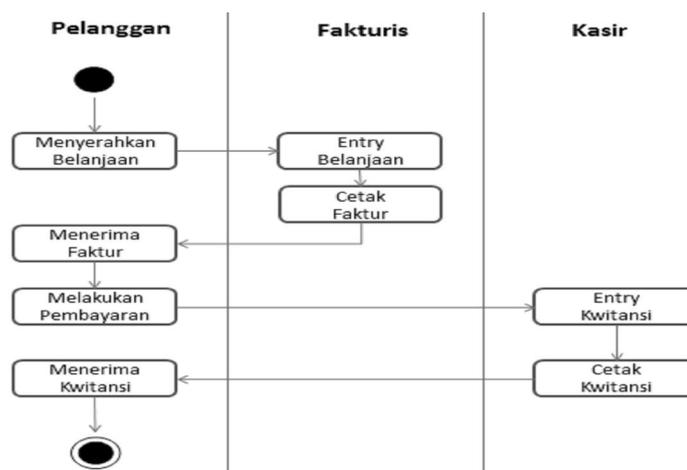
dan Grace ; 2013 : 81). Contoh pembuatan *use case* diagram dapat dilihat pada gambar II.1. berikut :



**Gambar. II.1. Use Case Diagram**  
(Sumber : Windu dan Grace ; 2013 : 83)

### II.10.2. Activity Diagram

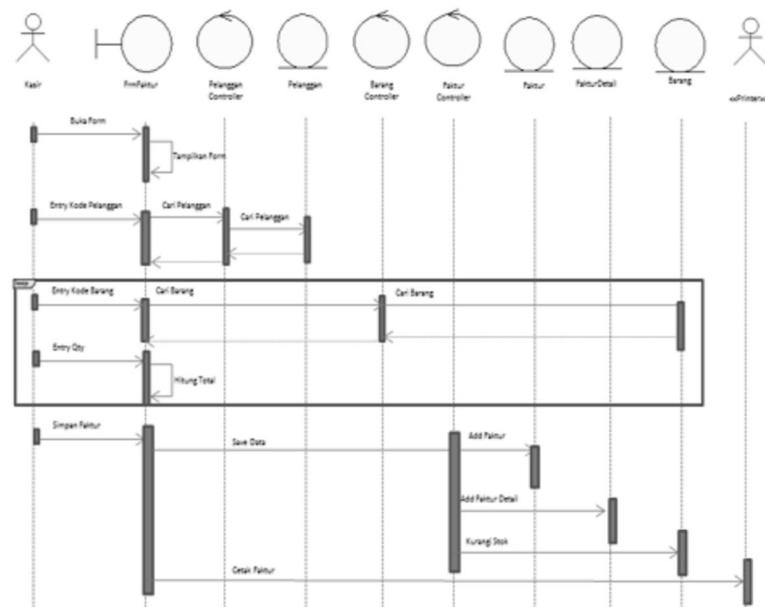
*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis (Windu dan Grace ; 2013 : 81). Contoh pembuatan *activity diagram* dapat dilihat pada gambar II.2. berikut :



**Gambar. II.2. Activity Diagram**  
(Sumber : Windu dan Grace ; 2013 : 83)

### II.10.3. Sequence Diagram

*Sequence diagram* menggambarkan kelakuan obyek pada *use case* dengan mendeskripsikan waktu hidup obyek dan pesan yang dikirimkan dan diterima antar obyek (Windu dan Grace ; 2013 : 81). Contoh pembuatan *sequence diagram* dapat dilihat pada gambar II.3. :

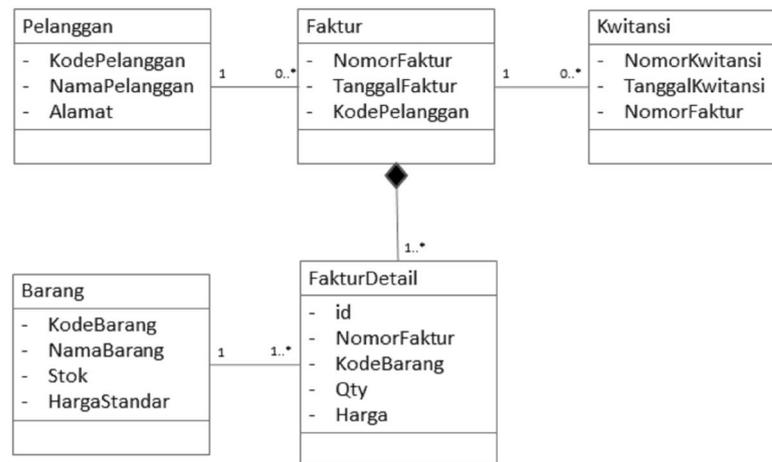


**Gambar. II.3. Sequence Diagram**  
(Sumber : Windu dan Grace ; 2013 : 84)

### II.10.4. Class Diagram

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan obyek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), *Relasi*, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), dan *Visibility*, tingkat akses

objek eksternal kepada suatu operasi atau atribut. Hubungan antar Kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau kardinaliti (Windu dan Grace ; 2013 : 81). Contoh pembuatan *class diagram* dapat dilihat pada gambar II.4. berikut :



**Gambar. II.4. Class Diagram**  
(Sumber : Windu dan Grace ; 2013 : 83)