

BAB II

TINJAUAN PUSTAKA

II.1. Sistem Pakar

Sistem pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek (*emulates*) kemampuan pengambilan keputusan (*decision making*) seorang pakar. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah. Sebuah sistem pakar umumnya tidak memiliki pengetahuan lain diluar area pengetahuannya kecuali jika diprogram dan dimuat ke dalam sistem. (Rika Rosnelly; 2012: 2)

II.1.1 Karakteristik Sistem Pakar

Sistem pakar umumnya dirancang untuk memenuhi beberapa karakteristik umum berikut ini :

1. Kinerja sangat baik (*high performance*)

Sistem harus mampu memberikan respon berupa saran (*advice*) dengan tingkat kualitas yang sama dengan seorang pakar atau melebihinya.

2. Waktu respon yang baik (*adequate respon time*)

Sistem juga harus mampu bekerja dalam waktu yang sama baiknya (*reasonable*) atau lebih cepat dibandingkan dengan seorang pakar dalam menghasilkan keputusan. Hal ini sangat penting terutama pada sistem waktu nyata (*real-time*).

3. Dapat dipahami (*understandable*)

Sistem harus mampu menjelaskan langkah-langkah penalaran yang dilakukannya seperti seorang pakar.

4. Dapat diandalkan (*good reliability*)

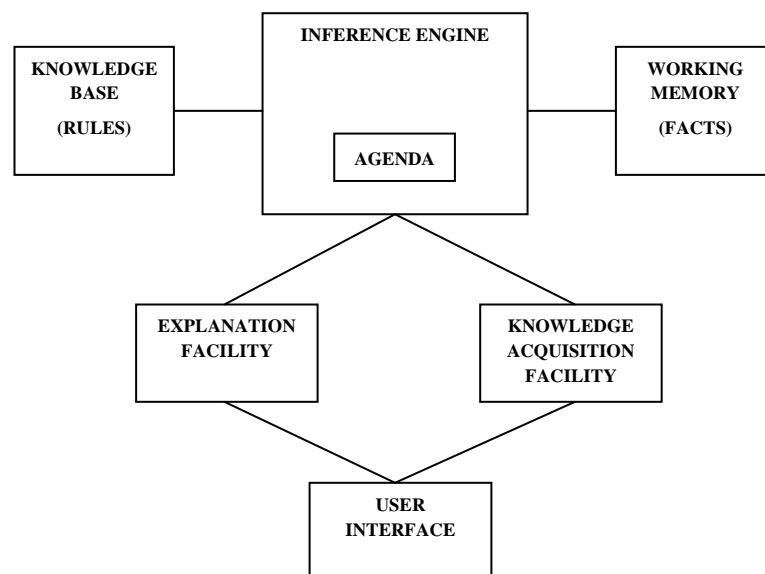
Sistem harus dapat diandalkan dan tidak mudah rusak / *crash*.

5. Fleksibel (*flexibility*)

Sistem harus menyediakan mekanisme untuk menambah, mengubah, dan menghapus pengetahuan. (Rika Rosnelly; 2012: 20-21)

II.1.2 Struktur Sistem Pakar

Adapun struktur sistem pakar dapat dilihat pada gambar II.1. berikut ini :



Gambar II.1. Struktur Sistem Pakar
(Sumber : Rika Rosnelly, 2012 : 13)

Komponen yang terdapat dalam struktur sistem pakar ini adalah *knowledge base (rules)*, *inference engine*, *working memory*, *explanation facility*, *knowledge acquisition facility*, *user interface*.

1. *Knowledge Base* (Basis Pengetahuan)

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Komponen sistem pakar disusun atas dua elemen dasar, yaitu fakta dan aturan. Fakta merupakan informasi tentang objek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui. Pada struktur sistem pakar diatas, *knowledge base* disini untuk menyimpan pengetahuan dari pakar berupa rule / aturan (if <kondisi> then <aksi> atau dapat juga disebut condition-action rules).

2. *Inference Engine* (Mesin Inferensi)

Mesin inferensi merupakan otak dari sebuah sistem pakar dan dikenal juga dengan sebutan *control structure* (struktur control) atau rule interpreter (dalam sistem pakar berbasis kaidah). Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi disini adalah processor pada sistem pakar yang mencocokkan bagian kondisi dari rule yang tersimpan di dalam *knowledge base* dengan fakta yang tersimpan di *working memory*.

3. *Working Memory*

Berguna untuk menyimpan fakta yang dihasilkan oleh inference engine dengan penambahan parameter berupa derajat kepercayaan atau dapat juga dikatakan sebagai global database dari fakta yang digunakan oleh rule-rule yang ada.

4. *Explanation Facility*

Menyediakan kebenaran dari solusi yang di hasilkan kepada user (*reasoning chain*).

5. *Knowledge Acquisition Facility*

Meliputi proses pengumpulan, pemindahan dan perubahan dari kemampuan pemecahan masalah seorang pakar atau sumber pengetahuan terdokumentasi ke program komputer, yang bertujuan untuk memperbaiki atau mengembangkan basis pengetahuan.

6. *User Interface*

Mekanisme untuk memberi kesempatan kepada user dan sistem pakar untuk berkomunikasi. Antar muka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai.
(Rika Rosnelly; 2012: 13-15)

II.1.3 Manfaat Sistem Pakar

Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat diberikannya, diantaranya :

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat daripada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Andal.
8. Meningkatkan kapabilitas sistem komputer.
9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
10. Bisa digunakan sebagai media pelengkap dalam penelitian.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar. (T. Sutojo; 2011: 160)

II.1.1 Kekurangan Sistem Pakar

Selain manfaat, ada juga beberapa kekurangan yang ada pada sistem pakar diantaranya :

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar. (T. Sutojo; 2011: 161)

II.2. Kopi

Pada mulanya, tanaman kopi belum dibudidayakan secara sempurna oleh penduduk, melainkan masih tumbuh liar di hutan-hutan dataran tinggi. Di Indonesia, tanaman kopi diperkenalkan pertama kali oleh VOC antara tahun 1696-1699. Awalnya, penanaman kopi hanya bersifat coba-coba (penelitian). Namun, karena hasilnya memuaskan dan dipandang cukup menguntungkan sebagai komoditas perdagangan maka VOC menyebarkan bibit kopi ke berbagai daerah agar penduduk dapat menanamnya. Kemudian, perkebunan besar pun didirikan dan akhirnya tanaman kopi tersebar ke daerah Lampung, Sumatera Barat, Sumatera Utara, Sumatera Selatan, dan daerah lain di Indonesia (Sri Najiyati.,Danarti; 2008: 1-2)

II.2.1 Fisiologi Tanaman Kopi

Kopi termasuk kelompok tanaman semak belukar dengan genus *Coffea*. Kopi termasuk ke *dalam famili Rubiaceae, subfamili Ixoroideae, dan suku Coffea*. Genus *Coffea* terbagai menjadi dua subgenus, yakni *Coffea* dan *Baracoffea*.

A. Akar

Tanaman kopi merupakan jenis tanaman berkeping dua (dikotil) dan memiliki akar tunggang. Pada akar tunggang, ada beberapa akar kecil yang tumbuh ke samping (melebar) yang sering disebut akar lebar. Pada akar lebar ini tumbuh akar rambut, bulu-bulu

akar, dan tudung akar. Tudung akar berfungsi untuk melindungi akar ketika menghisap unsur hara dari tanah.

B. Batang dan Percabangan

a. Cabang Primer (Plagiotrop)

Cabang primer yang terdapat pada batang kopi berada pada ruas ke-5 atau ke-6 dari leher akar. Cabang primer berfungsi sebagai cabang reproduksi serta tempat tumbuhnya cabang sekunder dan cabang balik. Cabang primer tumbuh ke samping dengan arah mendatar. Cabang ini tidak dapat berkembang biak dan hanya satu kali tumbuh, selanjutnya mati.

b. Cabang Sekunder

Cabang sekunder tumbuh di cabang primer. Cabang sekunder berfungsi sebagai tempat tumbuhnya cabang reproduksi atau ranting bunga dan buah. Berbagai arah tumbuh cabang sekunder di antaranya samping bawah, samping atas, dan membentuk cabang-cabang lain (cabang kipas).

c. Cabang Reproduksi (*Orthotrop*)

Tempat tumbuhnya cabang *orthotrop* berada di batang tanaman. Bentuk, arah tumbuh, dan sifatnya hampir sama dengan batang tanaman. Cabang ini berguna sebagai cabang reproduksi dan dapat menggantikan fungsi batang (dapat mengeluarkan ranting yang dapat menghasilkan buah). Dari

cabang orthotrop akan tumbuh cabang-cabang lain yang biasa disebut cabang air (wiwilan).

d. Cabang Balik

Cabang balik tumbuh di cabang primer. Sama seperti cabang sekunder, cabang ini juga sebagai cabang reproduksi. Selain itu, cabang balik juga berfungsi sebagai tempat tumbuhnya bunga dan buah. Arah pertumbuhannya cabang balik agak berbeda dengan jenis cabang lainnya, yaitu ke arah mahkota tajuk.

e. Cabang Kipas

Tempat tumbuhnya cabang kipas berada di bagian ujung cabang primer. Cabang reproduksi ini memiliki sifat seperti batang tanaman. Arah pertumbuhan cabang kipas mengarah ke samping dan atas.

C. Daun

Salah satu panduan untuk membedakan jenis tanaman kopi adalah dengan memerhatikan bentuk dan fisik daun. Secara umum, daun kopi berbentuk seperti telur, bergaris ke samping, bergelombang (talang air), hijau pekat, kekar, dan meruncing di bagian ujungnya. Daun tumbuh dan tersusun secara berdampingan di ketiak batang, cabang, dan ranting. Sepasang daun terletak di bidang yang sama di cabang dan ranting yang tumbuh mendatar.

D. Bunga

Bunga kopi terbentuk pada akhir musim hujan dan akan menjadi buah hingga siap petik pada awal musim kemarau. Setelah penyerbukkan, kopi akan menghasilkan kuntum bunga. Setiap ketiak daun menghasilkan 2-4 kelompok bunga. Setiap kelompok bunga menghasilkan 4-6 kuntum bunga, sehingga setiap ketiak daun menghasilkan 8-24 kuntum bunga. Kuntum bunga kopi berukuran kecil yang tersusun dari kelopak bunga, mahkota bunga, benang sari, tangkai putik, dan bakal buah. Kelopak bunga berwarna hijau. Mahkota bunga terdiri dari 3-8 helai daun. Sementara itu, benang sari terdiri dari 5-7 helai. Tangkai putik terdiri dari dua sirip berukuran kecil yang panjang.

E. Buah

Buah kopi mentah berwarna hijau muda. Setelah itu, berubah menjadi hijau tua, lalu kuning. Buah kopi matang (*ripe*) berwarna merah atau merah tua. Daging buah kopi yang sudah matang penuh mengandung lendir dan senyawa gula yang rasanya manis. Jika di dalam buah kopi hanya terdapat satu biji kopi dan bentuknya bulat memanjang biasanya disebut kopi jantan.

F. Penyerbukan

Tanaman kopi termasuk tanaman yang dapat melakukan penyerbukan sendiri (*self fertile*). Keberhasilan tanaman kopi untuk berbunga hingga menjadi buah sangat dipengaruhi oleh iklim

(musim hujan atau kemarau). Penyerbukan umumnya terjadi setelah musim hujan. Penyerbukan dipengaruhi oleh musim secara umum. Bunga umumnya muncul ketika tanaman kopi berumur sekitar 2-2,5 tahun. Sementara lama waktu perubahan bunga menjadi buah tergantung dari jenis kopi yang di tanam (Edy Panggabean; 2011: 10-18).

II.2.2 Hama dan Penyakit pada Tanaman Kopi

Tanaman kopi harus dihindarkan dari serangan hama dan penyakit. Hal ini dikarenakan faktor tersebut dapat menurunkan produksi dan mutu kopi yang dihasilkan. Bahkan, akibat serangan hama dan penyakit dapat menyebabkan tanaman tidak mau berbuah sama sekali, atau bahkan sering menyebabkan kematian (Sri Najiyati.,Danarti;2008:147).

II.2.2.1 Hama

1. Hama pada Akar (Nematoda)

Hama nematoda merusak akar-akar halus dan kecil. Hama ini umumnya menyerang tanaman kopi yang masih muda (berumur 1-2 tahun). Berikut beberapa hama nematoda yang sering dijumpai di tanaman kopi adalah *Pratylenchus coffeae*, *Meloidogyne spp.*, *Radopholus similes*, dan *Heterodera marioni*.

2. Hama pada Batang dan Dahan

Hama penggerek batang berupa larva dari beberapa jenis kumbang, seperti penggerek merah (*Zeuzera coffeae*). Jenis penggerek merah

memiliki ciri fisik khusus yaitu adanya bulatan warna kuning di bagian dada dan perut. Ukuran hama ini dapat mencapai 5 mm.

Hama penggerek betina bisa bertelur hingga 1.000 butir. Ciri utama telur hama ini diantaranya berwarna kuning, berkelompok kecil, dan biasanya terletak di bagian dalam kulit cabang. Perubahan telur menjadi larva terjadi cukup cepat. Setelah menjadi larva, hama ini menggerek menuju batang cabang dan berkembang biak menjadi kepompong.

3. Hama pada Ranting

Hama penggerek ranting di antaranya *Xylosandrus morstati* dan *Xylosandrus morigerus*. Keduanya sering disebut bibik dahan. Cara merusak dan berkembang biak hama ini dengan menggerek kulit dan membuat lubang kevil di dalam kayu. Lubang tersebut merupakan tempat bertelurnya hama betina. Perkembangan telur menjadi mago sekitar tiga minggu. Makanannya berupa cendawan *Ambrosia* yang ada di dalam lubang. Kumbang atau hama penggerek ini biasanya tidak ada pada musim kemarau.

4. Hama pada Buah

Jenis hama pada buah yang paling sulit diberantas diantaranya *Hypothenemus hampei* dan *Stephanoderes hampei*. Kerugian akibat hama tersebut dapat mencapai 80%. Hama tersebut pertama kali dijumpai di Bogor pada tahun 1909. Awalnya, induk hama menggerek buah kopi, lalu masuk ke dalam buah dan bertelur. Satu indukan dapat menggerek 20 buah dalam waktu 2-6 hari. Setelah 5-10 hari, telur

menetas menjadi larva. Setelah itu, larva berkembang menjadi kepompong (pupa) selama 6-16 hari. Selesai perkembangan itu, terjadilah perkawinan antara jantan dan betina di dalam biji kopi. Selanjutnya, betina akan pindah dan kembali bertelur di dalam lubang buah. Siklus ini terjadi selama lebih dari dua bulan.

5. Hama pada Daun, Tunas, dan Bunga

Hama yang sering menyerang daun, tunas, dan bunga tanaman kopi biasanya berupa kutu, seperti *Pseudococcus citri*, *Coccus viridis*, *Ferrissia virgata*. Berbagai kutu ini merusak tanaman kopi dengan menghisap cairan zat makanan di tunas muda, daun muda, ranting, dompolan bunga, dan buah menggunakan paruhnya.

II.2.2.2 Penyakit

1. Penyakit Karat Daun

CLR (*Coffea Leaf Rust*) merupakan penyakit karat daun yang disebabkan oleh *Hemileia vastatrix* melalui penyebaran angin atau udara. Sporanya berbentuk seperti kutil. Penyakit ini umumnya menyerang daun muda dengan kondisi kelembaban lingkungan yang tinggi. Penyakit ini sangat terkenal ke ganasannya, khususnya untuk perkebunan kopi arabika. Penyakit ini jarang menyerang tanaman kopi robusta. Pasalnya, perkebunan robusta umumnya berada di dataran rendah yang memiliki kelembaban lingkungan yang rendah.

2. Penyakit Bercak Daun

Penyakit bercak daun disebabkan oleh *Cercospora coffeicola*. Penyakit ini bisa menyerang kopi jenis arabika dan robusta. Serangannya dimulai dari daun, lalu merusak bagian buah. Penyakit ini biasanya menyerang saat pembibitan atau persemaian. Terdapat bercak berwarna coklat dan coklat tua berbentuk bulat.

3. Penyakit Daun Hangus

Penyakit daun hangus disebabkan oleh *Root dauw*. Ciri-cirinya adalah adanya lapisan berwarna hitam dipermukaan daun, lalu terdapat kumpulan semut di bagian daun. Suhu daun yang terserang mengalami kenaikan dan menyebabkan kelayuan.

4. Penyakit pada Bunga dan Buah

Penyakit pada bunga biasanya disebabkan oleh iklim yang ekstrem atau kondisi tanah yang sangat lembab dan basah. Salah satu penyakit pada bunga yang banyak dijumpai adalah bunga bintang. Penyakit ini umumnya menyerang tanaman kopi jenis arabika.

Sementara itu, penyakit yang sering terjadi di bagian buah adalah penyakit rontok buah. Penyakit ini dapat disebabkan oleh siklus tumbuh yang abnormal dan iklim ekstrem.

5. Penyakit Bercak Hitam

Penyakit bercak hitam pada buah disebabkan oleh *Cephaleuros coffea*. Mikroorganisme ini pada umumnya menyerang buah kopi yang belum matang. Ciri-cirinya terdapat bercak-bercak hitam pada buah kopi

muda(berwarna hijau), dan terdapat rambut-rambut halus dan butiran-butiran berwarna merah di buah kopi (Edy Panggabean; 2011:74-86).

II.3. Metode Dempster Shafer

Teori *Dempster-Shafer* (*DST*) merupakan teori matematika dari *evidence*. Teori ini dapat memberikan sebuah cara untuk menggabungkan *evidence* dari beberapa sumber dan mendatangkan atau memberikan tingkat kepercayaan (direpresentasikan melalui fungsi kepercayaan) dimana mengambil dari seluruh *evidence* yang tersedia. Teori tersebut pertama kali dikembangkan oleh Arthur P. Dempster dan Glenn Shafer.

Ada berbagai macam penalaran dengan model yang lengkap dan sangat konsisten, tetapi pada kenyataannya banyak permasalahan yang tidak dapat terselesaikan secara lengkap dan konsisten. Ketidakkonsistenan yang tersebut adalah akibat adanya penambahan fakta baru. Penalaran yang seperti itu disebut dengan penalaran *non monotonis*. Untuk mengatasi ketidakkonsistenan tersebut maka dapat menggunakan penalaran dengan teori *Dempster-Shafer*.

Secara umum teori *Dempster-Shafer* ditulis dalam suatu interval : [*Belief,Plausibility*]. *Belief* (*Bel*) adalah ukuran kekuatan *evidence* dalam mendukung suatu himpunan proposisi. Jika bernilai 0 maka mengindikasikan bahwa tidak ada *evidence*, dan jika bernilai 1 menunjukkan adanya kepastian. *Plausibility* (*Pl*) dinotasikan sebagai :

$$Pl(s) = 1 - Bel(\neg s)$$

Plausibility juga bernilai 0 sampai 1. Jika yakin akan $\neg s$, maka dapat dikatakan bahwa $Bel(\neg s)=1$, dan $Pl(\neg s)=0$

Pada teori *Dempster-Shafer* dikenal adanya *frame of discrement* yang dinotasikan dengan θ . *Frame* ini merupakan semesta pembicaraan dari sekumpulan *hipotesis*. Tujuannya adalah mengaitkan ukuran kepercayaan elemenelemen θ . Tidak semua *evidence* secara langsung mendukung tiap-tiap elemen.

Untuk itu perlu adanya probabilitas fungsi densitas (m). Nilai m tidak hanya mendefinisikan elemen-elemen θ saja, namun juga semua subsetnya. Sehingga jika θ berisi n elemen, maka subset θ adalah 2^n . Jumlah semua m dalam subset θ sama dengan 1. Apabila tidak ada informasi apapun untuk memilih hipotesis, maka nilai : $m\{\theta\} = 1,0$ (Anis Mistanti, 2014).

Apabila diketahui X adalah subset dari θ , dengan m_1 sebagai fungsi densitasnya, dan Y juga merupakan subset dari θ dengan m_2 sebagai fungsi densitasnya, maka dapat dibentuk fungsi kombinasi m_1 dan m_2 sebagai m_3 , yaitu :

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) \cdot m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) \cdot m_2(Y)} \dots \dots \dots (1)$$

Dimana :

m_1 = Densitas untuk gejala pertama

m_2 = Densitas untuk gejala kedua

m_3 = Kombinasi dari kedua Densitas diatas

\emptyset = Himpunan Kosong

XYZ = Himpunan Evidence

II.4. Basis Data

Menurut Bunafit Nugroho (2005: 72), Basisdata dapat juga disebut *database*, adalah sekumpulan informasi yang sangat kompleks yang berguna untuk mengatur semua data yang ada di dalamnya sehingga dapat diakses oleh pengguna dengan mudah dan cepat.

Dalam basisdata, data yang ada tidak hanya diletakkan dan disimpan begitu saja dalam sebuah media penyimpanan, akan tetapi dikelola dengan sebuah sistem pengaturan basisdata yang sering disebut dengan Database Management System (DBMS).

II.5. SQL Server R2008

SQL Server 2008 adalah sebuah terobosan baru dari Microsoft dalam bidang *database*. SQL Server adalah DBMS (Database Management System) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. SQL Server 2008 dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa SQL Server 2008 membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data.

Microsoft merilis SQL Server 2008 dalam beberapa versi yang disesuaikan dengan segment-segment pasar yang dituju. Versi-versi tersebut adalah sebagai berikut. Menurut cara pemrosesan data pada prosesor maka Microsoft mengelompokkan produk ini berdasarkan 2 jenis yaitu :

- Versi 32-bit(x86), yang biasanya digunakan untuk komputer dengan single prosesor (Pentium 4) atau lebih tepatnya prosesor 32 bit dan sistem operasi Windows XP.
- Versi 64-bit(x64), yang biasanya digunakan untuk komputer dengan lebih dari satu prosesor (Misalnya Core 2 Duo) dan system operasi 64 bit seperti Windows XP 64, Vista, dan Windows 7.

Sedangkan secara keseluruhan terdapat versi-versi seperti berikut ini:

- Versi Compact, ini adalah versi “Tipis” dari semua versi yang ada. Versi ini seperti versi desktop pada SQL Server 2000. Versi ini juga digunakan pada handled drvice seperti Pocket PC, PDA, SmartPhone, Tablet PC.
- Versi Express, ini adalah versi “Ringan” dari semua versi yang ada(tetapi versi ini berbeda dengan versi compact) dan paling cocok untuk latihan para pengembang aplikasi. Versi ini memuat Express Manager standar, integrasi dengan CLR dan XML.

II.6. Visual Basic 2010

Visual Basic 2010 merupakan salah satu bagian dari produk pemrograman terbaru yang dikeluarkan oleh *Microsoft*, yaitu *Microsoft Visual Studio 2010*. Visual Studio merupakan produk pemrograman

andalan dari *Microsoft Corporation*, dimana didalamnya berisi beberapa jenis IDE pemrograman seperti Visual Basic, Visual C++, Visual Web Developer, Visual C#, dan Visual F#.

Semua IDE pemrograman tersebut sudah mendukung implementasi .Net Framework terbaru, yaitu .Net Framework 4.0 yang merupakan pengembangan dari .Net Framework 3.5. adapun database standart yang disertakan adalah Microsoft Sql Server 2008 Express.

Visual Basic 2010 merupakan versi perbaikan dan pengembangan dari versi pendahulunya, yaitu Visual Basic 2008. Beberapa pengembangannya yang terdapat didalamnya antara lain dukungan terhadap *Library* terbaru dari *Microsoft* (Wahana Komputer : 2011).

II.7. *Unified Modelling Language (UML)*

Menurut Munawar (2005: 17), UML (*Unified Modeling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

Berikut ini adalah beberapa diagram UML (Rosa A. S, M. Shalahuddin; 2014: 141-167) :

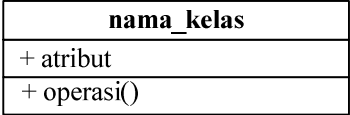
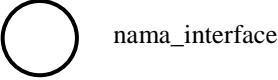

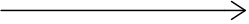

1. *Class Diagram*


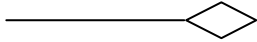
Diagram kelas atau *Class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
- Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas

Berikut adalah simbol-simbol yang ada pada diagram kelas :

Tabel II.1. Simbol-Simbol Diagram Kelas

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Generalisasi / <i>Generalization</i> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)

Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas
Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua bagian (<i>whole-part</i>)
1	Satu dan hanya satu
0.....*	Boleh tidak ada atau 1 atau lebih
1.....*	1 atau lebih
0.....1	Boleh tidak ada, maksimal 1
n.....n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Rosa A.S., M. Shalahuddin, 2014 : 141-147)

2. Use Case Diagram

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (behaviour) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

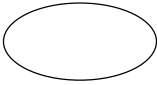
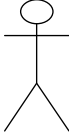

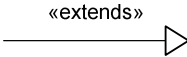
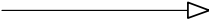
- Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol

aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

- *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case* :

Tabel II.2. Simbol-simbol Diagram *Use Case*

Simbol	Keterangan
<p><i>Use Case</i></p> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>
<p>Aktor / Actor</p> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama actor
<p>Asosiasi / Association</p> 	Komuniikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor
<p>Ekstensi / Extends</p> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan
<p>Generalisasi / Generalization</p> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<p>Menggunakan / Include / Uses</p>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini







(Sumber : Rosa A.S., M. Shalahuddin, 2014 : 156-158)


3. Activity Diagram

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

Berikut adalah simbol-simbol yang ada pada diagram aktivitas :

Tabel II.3. Simbol-simbol Diagram Aktivitas

Simbol	Keterangan
Status Awal 	Status awal aktivitas sistem sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>Decision</i> 	<i>Decision</i> , atau pilihan untuk mengambil keputusan.
Penggabungan / <i>Join</i> 	Arah tanda panah alur proses.

Status Akhir 	Titik akhir atau akhir dari aktivitas.
---	--

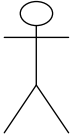
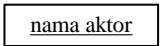
(Sumber : Rosa A.S., M. Shalahuddin, 2014 : 162)


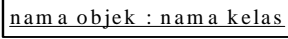
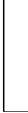
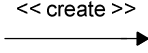
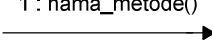
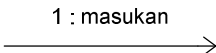
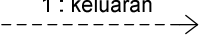
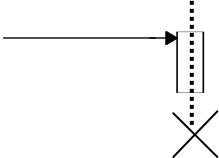
4. *Sequence Diagram*

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 yang ada pada *use case*.

Berikut adalah simbol-simbol yang ada pada diagram sekuen :

Tabel II.4. Simbol-simbol Diagram Sequence

Simbol	Keterangan
Aktor / <i>Actor</i>  Atau  Tanpa waktu aktif	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor

<p>Garis hidup / <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya</p>
<p>Pesan tipe create</p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p> 	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri</p>
<p>Pesan tipe send</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe return</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang di akhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i></p>

(Sumber : Rosa A.S., M. Shalahuddin, 2014 : 165-167)