

BAB III

LANDASAN TEORI

3.1 Unified Modelling Language (UML)

Menurut Julius (2003:7), unified modelling language (UML) adalah bahasa standar yang digunakan untuk menjelaskan dan memvisualisasikan artifak dari prses analisis dan disain berorientasi objek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu sistem. Ada beberapa diagram yang disediakan dalam UML (Sholiq, 2006:7), antara lain:

- a. Diagram *use case* (*use case diagram*)
- b. Diagram aktivitas (*activity diagram*)
- c. Diagram sekuensial (*sequence diagram*)
- d. Diagram kolaborasi (*collaboration diagram*)
- e. Diagram kelas (*class diagram*)
- f. Diagram *statechart* (*statechart diagram*)
- g. Diagram komponen (*component diagram*)
- h. Diagram *deployment* (*deployment diagram*)

Terdapat notasi pada UML, antara lain :

- a. Actor

Actor adalah segala sesuatu yang berinteraksi dengan sistem aplikasi komputer. Jadi *actor* ini bisa berupa orang, perangkat keras, atau mungkin juga objek lain dalam sistem yang sama. Biasanya yang dilakukan oleh

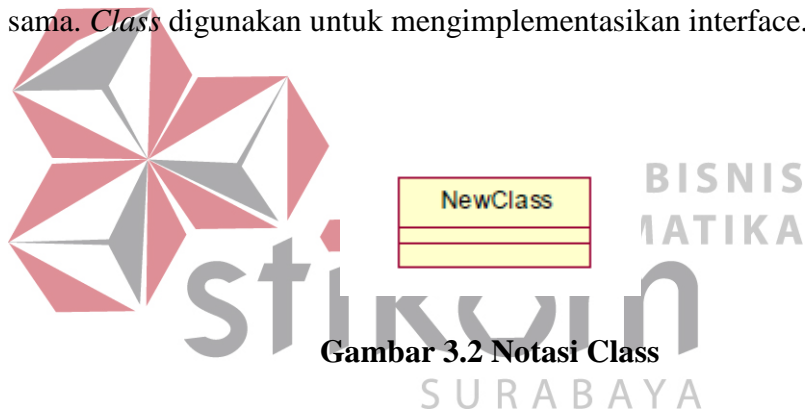
actor adalah memberikan informasi pada sistem dan memerintahkan sistem untuk melakukan sesuatu.



Gambar 3.1 Notasi Actor

b. Class

Class merupakan pembentuk utama dari sistem berorientasi objek karena *class* menunjukkan kumpulan objek yang memiliki atribut dan operasi yang sama. *Class* digunakan untuk mengimplementasikan interface.



Gambar 3.2 Notasi Class

c. Interface

Interface merupakan kumpulan operasi tanpa implementasi dari suatu *class*. Implementasi operasi dalam *interface* dijabarkan oleh operasi dalam *class*. Oleh karena itu keberadaan *interface* selalu disertai oleh *class* yang mengimplementasikan operasinya. *Interface* ini merupakan salah satu cara mewujudkan prinsip enkapsulasi dalam objek.



Gambar 3.3 Notasi Interface

d. Use case

Use case menjelaskan urutan kegiatan yang dilakukan *actor* dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan namun *use case* hanya menjelaskan apa yang dilakukan oleh *actor* dan sistem, bukan bagaimana *actor* dan sistem melakukan kegiatan tersebut.



Gambar 3.4 Notasi Use Case

e. Interaction

Interaction digunakan untuk menunjukkan baik aliran pesan atau informasi antar objek. Biasanya *interaction* ini dilengkapi juga dengan teks bernama *operation signature* yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.

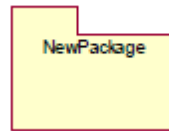


Gambar 3.5 Notasi Interaction

f. Package

Package adalah kontainer atau wadah konseptual yang digunakan untuk mengelompokkan elemen-elemen dari sistem yang sedang dibangun,

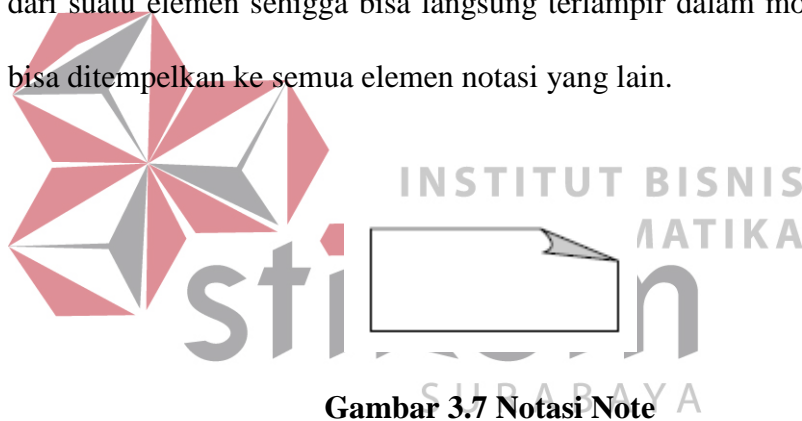
sehingga bisa dibuat model yang lebih sederhana. Tujuannya adalah untuk mempermudah penglihatan (*visibility*) dari model yang sedang dibangun.



Gambar 3.6 Notasi Package

g. Note

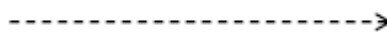
Note digunakan untuk memberikan keterangan dan komentar tambahan dari suatu elemen sehingga bisa langsung terlampir dalam model. Note ini bisa ditempelkan ke semua elemen notasi yang lain.



Gambar 3.7 Notasi Note

h. Dependency

Dependency merupakan relasi yang menunjukkan bahwa perubahan pada salah satu elemen memberi pengaruh pada elemen lain. Elemen yang ada di bagian tanda panah adalah elemen yang tergantung pada elemen yang ada di bagian tanpa tanda panah.



Gambar 3.8 Notasi Dependency

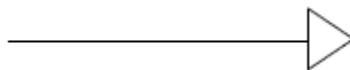
i. Association

Association menggambarkan navigasi antar *class* (*Navigation*), beberapa banyak objek lain yang bisa berhubungan dengan satu objek (*Multiplicity* antar *class*), dan apakah suatu *class* menjadi bagian dari *class* lainnya (*Aggregation*).

Gambar 3.9 Notasi Association

j. Generalization

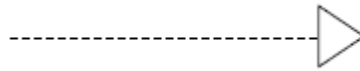
Generalization menunjukkan hubungan antara elemen yang lebih umum ke elemen yang lebih spesifik. Dengan *generalization*, *class* yang lebih spesifik (*subclass*) akan menurunkan atribut dan operasi dari *class* yang lebih umum (*superclass*), atau "*subclass is a superclass*". Dengan menggunakan notasi *generalization* ini konsep *inheritance* dari prinsip hirarki dimodelkan



Gambar 3.10 Notasi Generalization

k. Realization

Realization menunjukkan hubungan bahwa elemen yang ada di bagian tanpa panah akan merealisasikan apa yang dinyatakan oleh elemen yang ada di bagian dengan panah. Misalnya merealisasikan *package*, *component* merealisasikan *class* atau *interface*.



Gambar 3.11 Notasi Realization

3.2 Visual Studio 2005

Microsoft Visual Basic .NET adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak di atas sistem .NET Framework, dengan menggunakan bahasa BASIC. Dengan menggunakan alat ini, para programmer dapat membangun aplikasi Windows Forms, Aplikasi web berbasis ASP.NET, dan juga aplikasi command-line. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti Microsoft Visual C++, Visual C#, atau Visual J#), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio .NET. Bahasa Visual Basic .NET sendiri menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari Microsoft Visual Basic versi sebelumnya yang diimplementasikan di atas .NET Framework. Peluncurannya mengundang kontroversi, mengingat banyak sekali perubahan yang dilakukan oleh Microsoft, dan versi baru ini tidak kompatibel dengan versi terdahulu. Bahasa Visual Basic 2005 memperkenalkan fitur-fitur baru, yakni:

- a. Bawaan .NET Framework 2.0:
 - o Generics
 - o Partial class, sebuah metode yang dapat digunakan untuk mendefinisikan beberapa bagian dari sebuah kelas di dalam sebuah berkas, lalu menambahkan definisinya di lain waktu sangat berguna

khususnya ketika mengintegrasikan kode pengguna dengan kode yang dibuat secara otomatis.

- o Nullable Type.
- b. Komentar XML yang dapat diproses dengan menggunakan beberapa alat bantu seperti NDoc untuk membuat dokumentasi secara otomatis.
- c. Operator overloading.
- d. Dukungan terhadap tipe data bilangan bulat tak bertanda (*unsigned integer*) yang umumnya digunakan di dalam bahasa lainnya.

3.3 SQL Server

Microsoft SQL Server adalah sebuah sistem manajemen basis data relasional (RDBMS) produk Microsoft. Bahasa kueri utamanya adalah Transact-SQL yang merupakan implementasi dari SQL standar ANSI/ISO yang digunakan oleh Microsoft dan Sybase. Umumnya SQL Server digunakan di dunia bisnis yang memiliki basis data berskala kecil sampai dengan menengah, tetapi kemudian berkembang dengan digunakannya SQL Server pada basis data besar.

Microsoft SQL Server dan Sybase/ASE dapat berkomunikasi lewat jaringan dengan menggunakan protokol TDS (Tabular Data Stream). Selain dari itu, Microsoft SQL Server juga mendukung ODBC (Open Database Connectivity), dan mempunyai driver JDBC untuk bahasa pemrograman Java. Fitur yang lain dari SQL Server ini adalah kemampuannya untuk membuat basis data *mirroring* dan *clustering*. Pada versi sebelumnya, MS SQL Server 2000 terserang oleh cacing komputer SQL Slammer yang mengakibatkan kelambatan akses Internet pada tanggal 25 Januari 2003.

3.4 Debit Air

Dalam hidrologi dikemukakan, debit air sungai adalah, tinggi permukaan air sungai yang terukur oleh alat ukur permukaan air sungai. Pengukurannya dilakukan tiap hari, pada jam-jam tertentu. Faktor utama yang mempengaruhi ketinggian air sungai, adalah curah hujan yang terjadi di hulu tempat alat pengukur permukaan air sungai ditempatkan. Karena curah hujan merupakan data deret waktu yang memiliki komponen musiman, dan siklus tahunan dengan karakteristik musim hujan panjang (kemarau pendek), atau kemarau panjang (musim hujan pendek).

Untuk mendapatkan debit banjir rancangan dapat dihitung dengan menggunakan metode rasional. Data yang digunakan adalah data curah hujan harian dan data tata guna lahan. Data ini kemudian ditransformasikan menjadi intensitas hujan jam-jaman menggunakan berbagai rumus yang mendukung, seperti metode Monobo, metode untuk menghitung debit puncak salah satu faktor yang mempengaruhi adalah intensitas hujan. Adapun besarnya intensitas hujan untuk berbagai kala ulang 1, 2, 5, 10, 15, 25 (tahun) adalah 1,925 mm/jam; 4,106 mm/jam; 5,564 mm/jam; 6,598 mm/jam; 7,005 mm/jam; 7,454 mm/jam; dan 7,914 mm/jam. Tanah atau tata guna lahan suatu daerah irigasi juga sangat mempengaruhi besarnya debit puncak yang terjadi pada waktu datangnya hujan dimana hujan tidak lagi mengalami infiltrasi melainkan melimpah sebagai aliran permukaan.

3.5 Dundas Map For Windows Forms

Merupakan tool untuk memasukan peta spasial ke dalam windows form (vb.net). Dundas Map dapat memanggil data-data yang ber-*extensi* peta spasial contohnya .shp. Dengan penggunaan Dundas map ini, peta-peta spasial dapat menjadi program yang dinamis.

