

BAB II

TINJAUAN PUSTAKA

II.1.1 Sistem

Menurut (Kusrini M.Kom; 2009:11) Sistem merupakan kumpulan elemen yang saling berkaitan yang bertanggung jawab memproses masukan (*input*) sehingga menghasilkan keluaran (*output*).

Menurut Richardus Eko Indrajit, (2001:2) Kata ‘sistem’ mengandung arti kumpulan dari komponen-komponen yang memiliki unsur keterkaitan antara satu dan lainnya. Sistem informasi merupakan suatu kumpulan dari komponen-komponen dalam perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran informasi. Dalam hal ini, teknologi informasi hanya merupakan salah satu komponen kecil saja dalam format perusahaan. Komponen-komponen lainnya adalah: proses dan prosedur, struktur organisasi, sumber daya manusia, produk, pelanggan, supplier, rekanan, dan lain sebagainya. Jadi, kehandalan suatu sistem informasi dalam perusahaan atau organisasi terletak pada keterkaitan antar komponen-komponen yang ada, sehingga dapat dihasilkan dan dialirkan suatu informasi yang berguna (akurat, terpercaya, detil, cepat, dan relevan) untuk lembaga yang bersangkutan.

II.1.2 Data

Menurut (Kusrini M.Kom; 2009:3) Data merupakan representasi dari fakta atau gambaran mengenai suatu objek atau kejadian, ambil contoh fakta mengenai biodata mahasiswa yang meliputi nama, alamat, jenis kelamin, agama yang dianut, dan lain-lain. Contoh lain dari fakta mengenai kejadian / transaksi dalam sebuah perusahaan dagang adalah seperti transaksi penjualan yang meliputi waktu transaksi, pelaku transaksinya (pelanggan, kasir), barang yang ditransaksikan, serta jumlah dan harganya. Data dinyatakan dengan nilai yang berbentuk angka, deretan karakter, atau simbol.

Menurut Zulkiffi A.M (1999:156) Data adalah catatan atas kumpulan fakta. Data merupakan bentuk jamak dari datum, berasal dari bahasa Latin yang berarti “sesuatu yang diberikan”. Dalam penggunaan sehari-hari data berarti suatu pernyataan yang diterima secara apa adanya. Pernyataan ini adalah hasil pengukuran atau pengamatan suatu variabel yang bentuknya dapat berupa angka, kata-kata, atau citra.

II.1.3. Informasi

Menurut (Jogiyanto HM; 1999:692) informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (event) yang nyata (fact) yang digunakan untuk pengambilan keputusan.

Menurut (Kusrini M.Kom; 2009:4) Suatu informasi berguna bagi Pembuat keputusan karena informasi bisa menurunkan ketidakpastian (meningkatkan pengetahuan) tentang hal yang sedang dipikirkan. Makna dari sebuah informasi tentu berbeda-beda antara seorang dengan lainnya, tergantung pada tingkat kepentingannya, misalnya informasi daftar pelanggan yang potensial akan sangat dibutuhkan oleh bagian marketing di suatu perusahaan guna meningkatkan penjualan produk, tetapi barangkali tidak akan menjadi perhatian dibagian personalia

Kegunaan informasi bagi seseorang juga sangat tergantung pada waktu. Pada suatu waktu tertentu informasi tersebut mungkin sangat diperlukan di lain hari, mungkin saja hal tersebut sudah tidak berguna sama sekali. Contohnya, informasi perbandingan harga barang akan sangat dibutuhkan oleh seseorang yang akan membeli barang tersebut. Namun saat ini dia sedang tidak mempertimbangkan untuk membeli barang tersebut, informasi tersebut menjadi kurang bermakna

II.1.4. Kualitas Informasi

Agar bisa menyediakan keluaran yang berguna untuk membantu manager atau para pengambil keputusan, sebuah sistem informasi harus mampu mengumpulkan data dan mentransformasikan data tersebut kedalam informasi yang memiliki kualitas-kualitas tersebut

Berikut karakteristik informasi yang berkualitas :

1. Relevan. Informasi yang disajikan sebaiknya terkait dengan keputusan yang akan diambil oleh pengguna informasi tersebut. Misalnya, seorang manager yang akan memberikan kredit kepada pelanggan bisa melihat laporan keuangan pelanggan tersebut karena laporan tersebut terkait dengan keputusan yang akan dibuat, yaitu memberikan atau tidak memberikan kredit kepada pelanggan tersebut.
2. Akurat. Kecocok antara informasi dengan kejadian-kejadian atau objek-objek yang diwakilinya. Misalnya, laporan inventaris yang tidak akurat menyebutkan bahwa terdapat 15 unit barang yang tersisadi gudang. Kenyataanya, masih ada 51 unit barang di dalam gudang.
3. Lengkap. Merupakan derajat sampai seberapa jauh informasi menyertakan kejadian-kejadian atau objek-objek yang berhubungan. Misalnya, penjualan selama satu hari yang seharusnya ada 150 transaksi di laporan hanya tercatat sebanyak 145 transaksi.
4. Tepat waktu. Informasi yang tidak tepat waktu akan menjadi informasi yang tidak berguna atau tidak dapat di gunakan untuk membantu pengambilan keputusan. Misalnya, informasi jadwal ujian seorang mahasiswa disampaikan setelah kegiatan ujian diselenggarakan. Informasi ini menjadi tidak berguna lagi.
5. Dapat dipahami. Hal tersebut terkait dengan bahasa dan cara penyajian informasi agar pengguna lebih mudah mengambil keputusan.

6. Dapat dibandingkan. Sebuah informasi yang memungkinkan seorang pemakai untuk mengidentifikasi persamaan dan perbedaan antara dua objek atau kejadian yang mirip. Misalnya, membandingkan laporan pendapatan antara tahun 2006 dan 2007.

II.1.5. Sistem Informasi

Menurut (Kusrini M.Kom; 2009:11) suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan merupakan kegiatan strategi dari suatu organisasi, serta menyediakan laporan-laporan yang diperlukan untuk pihak luar.

Berdasarkan dukungan kepada pemakainya, sistem informasi dibagi menjadi

1. Sistem Pemrosesan Transaksi (*Transaction Processing System*) atau TPS
2. Sistem Informasi Manajemen (*Management Information System*) atau MIS
3. Sistem Otomasi Perkantoran (*Office Automation System / OAS*)
4. Sistem Pendukung Keputusan (*Decision Support System*) atau DSS
5. Sistem Informasi Eksekutif (*Executive Information System*) atau EIS
6. Sistem Pendukung Kelompok (*Group Support System*) atau GSS
7. Sistem Pendukung Cerdas (*Intelligent Support System*) atau ISS

Mengingat bahwa EIS, DSS, dan MIS digunakan untuk pendukung manajemen, maka ketiga sistem tersebut sering disebut Sistem Pendukung Manajemen (*management support system*) atau MSS

II.2. Sejarah Visual Basic

Menurut Harip Santoso (2006 : 3), diluncurkan *VB.Net 2002* (Pemrograman berbasis Teknologi.Net), dapat dianggap sebagai pertanda berakhirnya pengembangan *Visual Studio 6*, karena secara *defacto* VB6 sudah tidak dikembangkan lagi oleh *Microsoft*. Walaupun VB6 tidak dikembangkan lagi, tetapi kesinambungan masa hidup VB6 masih dijaga oleh *Microsoft*. Contohnya dengan diluncurkannya *Service Pack 5* untuk *Visual Studio 6* agar bahasa pemrograman yang terdapat pada *visual Studio 6* dapat mengakses *database SQL Server 2005*.

Kini teknologi pemrograman telah dialihkan ke pemrograman berbasis .NET, teknologi yang intinya adalah menulis program agar dapat dijalankan pada piranti lunak maupun keras yang berbeda tanpa harus mengubah program yang sudah dikompilasi. Misalnya, program yang dibuat pada sistem operasi *windows*, nantinya dapat dijalankan pada sistem operasi *Unix/Linux* dan sebaliknya, dimana hal ini dimungkinkan dengan diperkenalkannya konsep *.NET Framework*.

Service Pack yang dikeluarkan oleh *Microsoft* lebih pada penyediaan fasilitas piranti tambahan sebagai media untuk mengakses/membuat hubungan dengan piranti *software* yang baru, misalnya *SQL Server 2005*. jadi walaupun anda menginstal *Service Pack*, tidak berarti *Visual Studio 6* (VB 6) anda akan lebih canggih. itulah yang dimaksud dengan VB 6 tidak dikembangkan lagi oleh *Microsoft*.

Visual Basic 6 diluncurkan pada tahun 1998 dan terdiri dari tiga versi, yaitu *Learning Edition*, *Professional Edition*, dan *Enterprise Edition*. Adapun inti

peningkatan yang terdapat pada VB 6 dibandingkan dengan versi sebelumnya adalah, peningkatan akses data, penambahan *tools*, dan kontrol baru untuk mengakses database, misalnya *ADODC*.

Awal sejarah *Visual Basic* dimulai pada pertengahan tahun 1991 dengan diluncurkannya VB versi 1.0. yang merupakan pengembangan dari *Quick Basic*, yaitu bahasa pemrograman *Basic* yang dimiliki oleh *Microsoft* untuk penulisan program pada sistem operasi *DOS*. *QuickBasic* masih bersifat teks program, dan belum bersifat grafik. Singkatnya, piranti mouse belum dapat digunakan dalam kaitan dengan interaksi antara pengguna dan aplikasi program

VB 2.0 diluncurkan pada tahun 1992, dimana pada masa itu mulai diperkenalkan *ODBC (Open Database Connectivity)* sebagai metode untuk mengakses data, perkembangan VB 2.0 tidak terlalu nampak pada masa itu, karena bahasa pemrograman berbasis *xBase (dbase,foxpro,clipper)* mendominasi penulisan aplikasi program. Disamping itu sistem operasi *DOS* masih bergandengan erat dengan sistem operasi *Netware* yang dimiliki oleh *Novell* sebagai sistem operasi dasar pada aplikasi berbasis *LAN (Local Area Network)*. (Harip Santoso ; 2006 : 4).

II.3. Microsoft SQL Server

Menurut Harip Santoso (2006 : 5), sejarah *SQL Server* berbeda jauh dengan sejarah *Visual Basic*. Bila *Visual Basic* berasal dari pengembangan *QuickBasic* yang juga merupakan produk *Microsoft*, maka *SQL Server* adalah hasil kerja sama

antara *Microsoft* dengan *Sybase* untuk memproduksi sebuah software penyimpanan data (database) yang bekerja pada sistem operasi OS/2.

Sistem Operasi OS/2 merupakan sistem operasi baru dari hasil kerja sama antara *Microsoft* dengan *IBM*. Sistem operasi OS/2 mengenal bentuk-bentuk perintah DOS, sekaligus memiliki kemampuan *multitasking*. Untuk mendapat pengakuan pasar, maka *Microsoft* bekerja sama dengan *Ashton-Tate* yang telah dikenal dengan produk *dBase*. Sayangnya, kerja sama ini tidak berlangsung lama, karena setelah peluncuran *SQL Server* versi 1.0 pada tahun 1989 kerja sama itupun berakhir.

Peluncuran *SQL Server* 1.0 dilanjutkan dengan peluncuran *SQL Server* versi 1.1 pada tahun 1990. Adapun fitur terpenting dari produk ini adalah dukungan untuk *platform* baru dari sisi client yang dikenal sebagai sistem operasi *Windows 3.1*

Pada tahun 1991, *SQL Server* versi 1.11 diluncurkan dan berisi fasilitas perawatan database. Pada tahun yang sama, *Microsoft* mengembangkan sendiri sistem operasi *multi user* yang dikenal sebagai *Windows NT*, *Microsoft SQL Server* versi 4.2 dirilis pada tahun 1992 dan berisi tool administrator database berbasis *GUI-Windows*.

II.4. Database

Menurut Budi Raharjo (2011:3) istilah database banyak memiliki definisi. Untuk sebagian kalangan sederhana *database* diartikan sebagai kumpulan data

(buku, nomor telepon, daftar pegawai, dan lain sebagainya). Ada juga yang menyebut *database* dengan definisi lain yang lebih formal dan tegas. *Database* didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi, diambil dan dicari secara cepat.

Selain berisi data, *database* juga berisi *metadata*. Metadata adalah data yang menjelaskan tentang struktur dari data itu sendiri. Sebagai contoh, Anda dapat memperoleh informasi tentang nama-nama kolom dan tipe yang ditampilkan tersebut disebut *metadata*.

II.4.1. Pemodelan Data

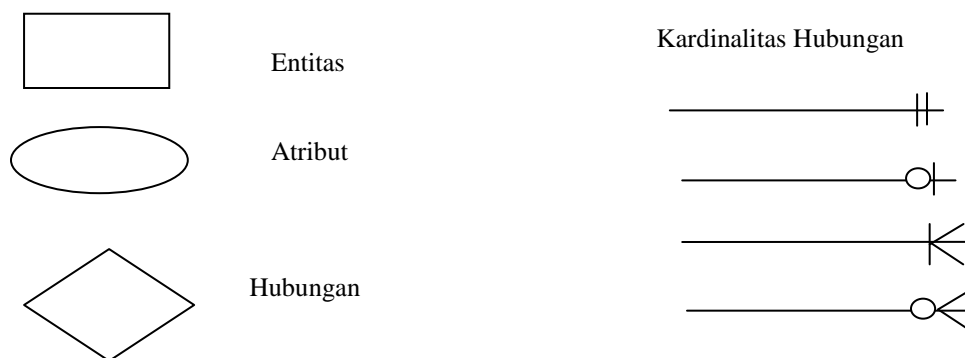
Menurut Abdul Kadir (2009:30) Pada Perancangan konseptual diperlukan suatu pendekatan yang digunakan untuk menggambarkan hubungan antar data. Hubungan tersebut dapat dinyatakan dalam bentuk E-R, karena model E-R adalah dasar penting dalam merancang database maka akan dijelaskan tentang gambaran tentang model E-R, penjelasan mengenai komponen-komponen yang menyusun model E-R, hingga cara penyusunan model E-R.

II.4.2. Model E-R

Menurut Abdul Kadir (2009:30) Model E-R adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas, atribut dan hubungan antar entitas. Huruf E sendiri menyatakan entitas dan R menyatakan hubungan (dari

kata *Relationship*). Model ini dinyatakan dalam bentuk diagram, itulah sebabnya model E-R sering disebut sebagai diagram E-R.

Model E-R melibatkan sejumlah notasi, beberapa notasi dasar dalam model E-R ditunjukkan pada gambar II.1, notasi-notasi tersebut diberikan hanya untuk memberikan suatu pengetahuan dasar.



Gambar II.1 : Sejumlah notasi pada model E-R

Sumber : Abdul Kadir (2009:31)

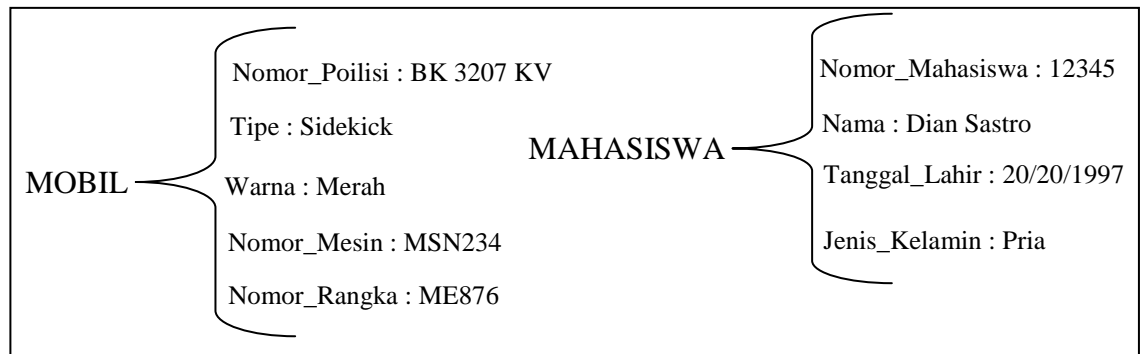
1. Entitas

Yang dimaksud dengan entitas adalah sesuatu dalam dunia nyata yang keberadaannya tidak bergantung pada yang lain. Sebagai contoh, setiap pegawai dalam sebuah organisasi adalah sebuah entitas. Entitas dapat berupa sesuatu yang nyata ataupun abstrak (berupa suatu konsep). Secara lebih rinci dijelaskan bahwa entitas dapat berupa seseorang, sebuah tempat, sebuah objek, sebuah kejadian atau suatu konsep.

2. Atribut

Setiap entitas dinyatakan dalam sejumlah atribut. Atribut adalah properti atau karakteristik yang terdapat pada setiap entitas. Sebagai contoh, pada gambar

II.2, terdapat entitas MOBIL yang mengandung atribut Nomor_Polisi, Tipe, Warna, Nomor_Rangka dan Nomor_Mesin. Selain itu terdapat entitas MAHASISWA yang mengandung atribut Nomor_Mahasiswa, Nama, Tanggal_Lahir, dan Jenis_Kelamin.

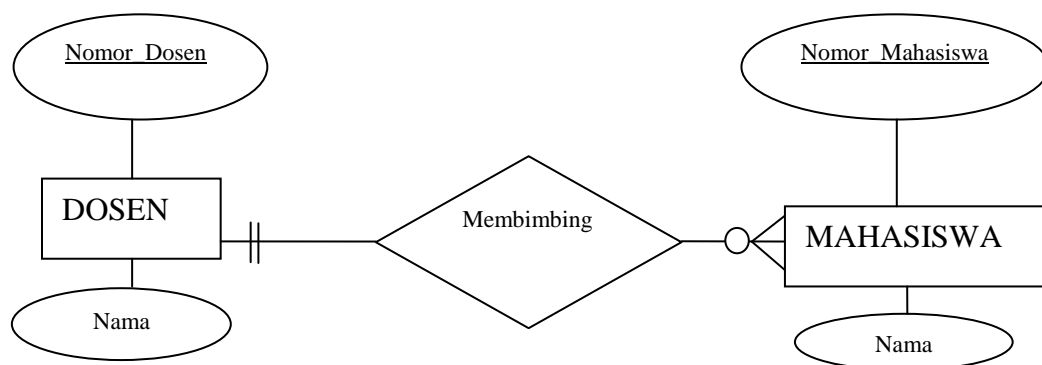


Gambar II.2 : Contoh Entitas dan Atribut

Sumber : Abdul Kadir (2009:32)

3. Hubungan (*Relationship*)

Hubungan (*Relationship*) menyatakan ketertarikan antara beberapa tipe entitas. Sebagai contoh , tipe entitas MAHASISWA dan DOSEN mempunyai hubungan yang mencerminkan bahwa seorang mahasiswa memiliki dosen pembimbing akademis. Gambar II.3 menunjukkan hubungan tersebut.



Gambar II.3 : Contoh Hubungan antara tipe entitas

Sumber : Abdul Kadir (2009:45)

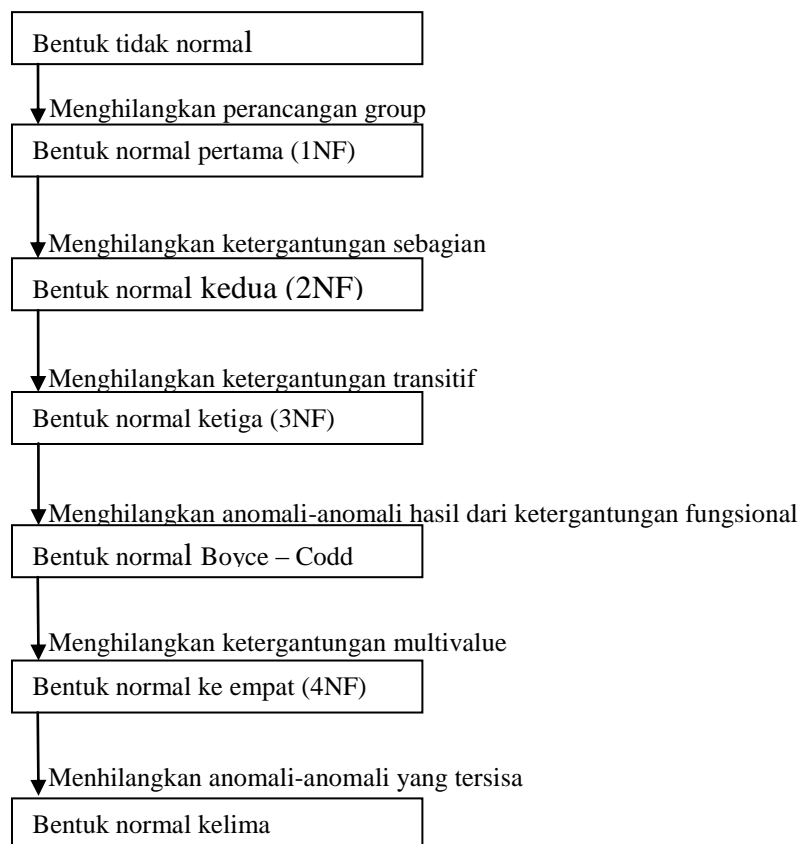
II.4.3. Normalisasi

Menurut Marlinda (2004:115) Normalisasi merupakan proses pengorganisasian file untuk menghilangkan grup elemen yang berulang menjadi tabel-tabel yang menjadi entity dan relasinya. Pada proses ini selalu dituju pada beberapa kondisi apakah ada kesulitan pada saat menambah (insert), menghapus (delete), mengubah (update), membaca (retrieve) pada suatu database.

Menurut Abdul Kadir (2009:116) Normalisasi adalah proses yang digunakan untuk menentukan pengelompokan atribut-atribut dalam sebuah relasi sehingga diperoleh relasi yang berstruktur baik. Dalam hal ini yang dimaksud dengan relasi yang berstruktur baik adalah relasi yang memenuhi dua kondisi berikut

1. Mengandung redundansi sedikit mungkin, dan
2. Memungkinkan baris-baris dalam relasi disispkan, dimodifikasi dan dihapus tanpa menimbulkan kesalahan atau ketidakkonsistenan.

Normalisasi sendiri dilakukan melalui sejumlah langkah. Setiap langkah berhubungan dengan bentuk normal (*normal form*) tertentu. Gambar II.4 berikut ini akan memperlihatkan hubungan keenam bentuk normal tersebut.



Gambar II.4 : Langkah-langkah dalam normalisasi

Sumber : Abdul Kadir (2009:118)

II.5. Unified Modeling Language (UML)

Menurut Rosa A.S & M. Shalahuddin (2011 : 118) Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling*

Language (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML hanya berfungsi untuk melakukan pemodelan, jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metode berorientasi objek.

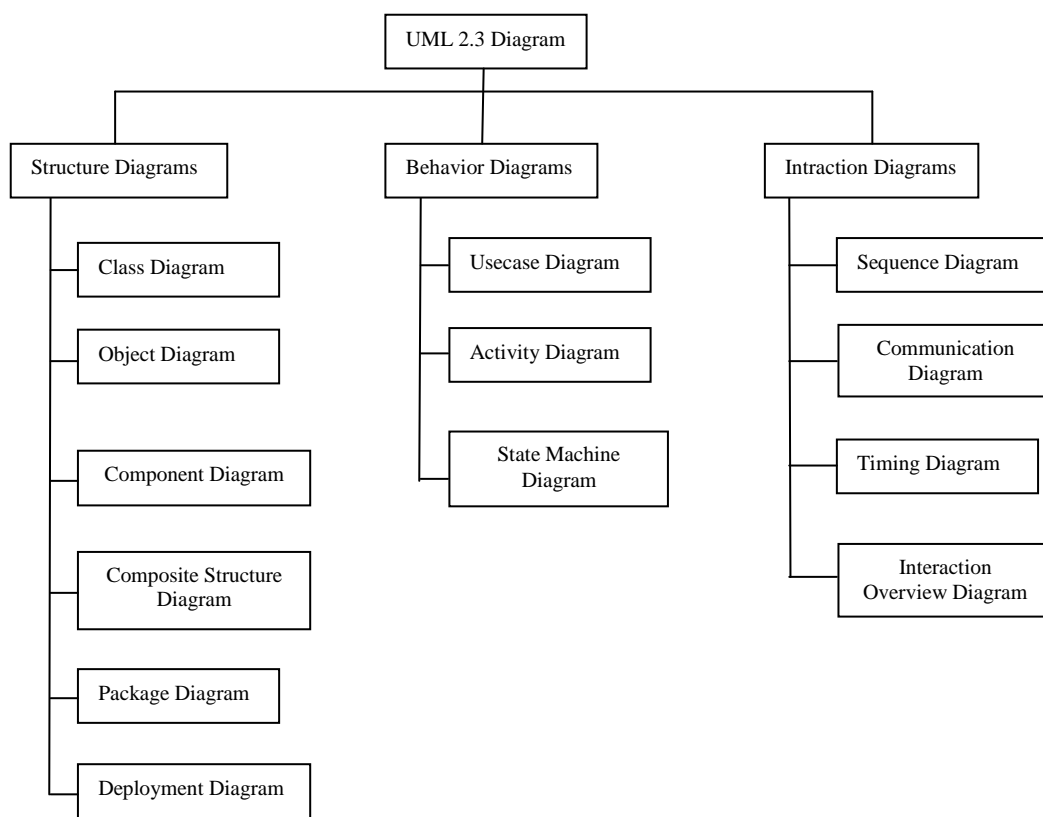
Menurut Prabowo Pudjo Widodo & Herlawati (2011 : 6) UML diaplikasikan untuk maksud tertentu, biasanya antara lain :

1. Merancang perangkat Lunak.
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Blok pembangunan utama UML adalah diagram. Beberapa diagram ada yang rinci (jenis *timing diagram*) dan lainnya ada yang bersifat umum (misalnya diagram kelas). Para pengembang sistem berorientasi objek menggunakan bahasa model untuk menggambarkan, membangun dan mendokumentasikan sistem yang mereka rancang. UML memungkinkan para anggota team untuk bekerja sama dengan bahasa model yang sama dengan mengaplikasikan beragam sistem. Intinya UML merupakan alat komunikasi yang konsisten dalam mendukung para pengembang sistem saat ini.

II.5.1 Diagram-Diagram UML

Menurut Rosa A.S & M. Shalahuddin (2011 : 120) Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar II.5 di bawah ini



Gambar II.5 : Diagram UML

Sumber : Rosa A.S & M. Shalahuddin (2011 : 121)

Berikut ini penjelasan singkat dari pembagian kategori tersebut

1. *StructureDiagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction Diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

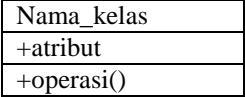


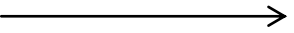
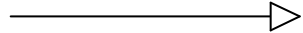
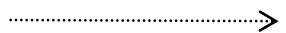
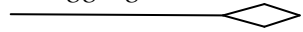
A. *Class Diagram*

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.

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

Berikut gambar II.6 menerangkan simbol-simbol pada diagram kelas :

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur sistem
<p>Antarmuka / <i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
<p>Asosiasi / <i>association</i></p>  <p>Asosiasi berarah/ <i>directed association</i></p> 	<p>Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i></p> <p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i></p>
<p>Generalisasi</p> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
<p>Kebergantungan</p> 	Relasi antar kelas dengan makna kebergantungan antar kelas
<p>Agregasi / <i>aggregation</i></p> 	Semua bagian (<i>whole part</i>)

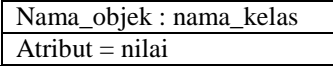

Gambar II.6 : Diagram Kelas

Sumber : Rosa A.S & M. Shalahuddin (2011 : 124)

B. *Object Diagram*

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Pada diagram objek harus dipastikan semua kelas yang sudah didefinisikan pada diagram kelas harus dipakai objeknya, karena jika tidak, pendefinisian kelas itu tidak dapat dipertanggungjawabkan. Untuk apa mendefinisikan sebuah kelas sedangkan pada jalannya sistem, objeknya tidak pernah dipakai.

Berikut adalah gambar II.7 menerangkan simbol-simbol diagram objek

Simbol	Deskripsi
Objek 	Objek dari kelas yang berjalansaat sistem dijalankan
Link 	Relasi antar objek

Gambar II.7 : Diagram Paket

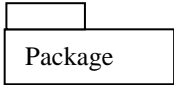
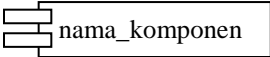
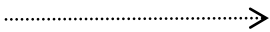


Sumber : Rosa A.S & M. Shalahuddin (2011 : 124)

C. *Component Diagram*

Diagram komponen atau component diagram dibuat untuk menunjukkan organisasi dan ketergantungan di antara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem. Komponen dasar yang biasanya ada dalam suatu sistem adalah sebagai berikut :

- 1) Komponen *user interface* yang menangani tampilan
- 2) Komponen *bussiness procesiing* yang menangani fungsi-fungsi proses bisnis
- 3) Komponen data yang menangani manipulasi data
- 4) Komponen *security* yang menangani keamanan sistem

Komponen lebih terfokus pada penggolongan secara umum fungsi-fungsi yang diperlukan, berikut gambar II.8 yang menerangkan simbol-simbol yang ada pada diagram komponen

Simbol	Deskripsi
Package 	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih komponen
Komponen 	Komponen Sistem
Kebergantungan / <i>dependency</i> 	Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai
Antar muka / <i>interface</i> 	Sama dengan konsep <i>interface</i> pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen
Link 	Relasi antar komponen

Gambar II.8 : Diagram Komponen

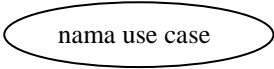

Sumber : Rosa A.S & M. Shalahuddin (2011 : 126)


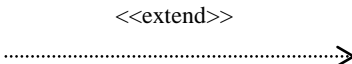
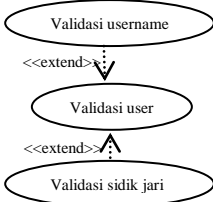
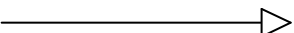
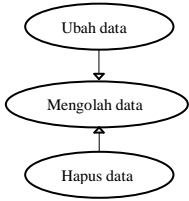
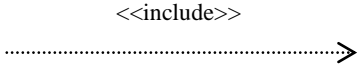
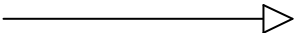
D. Use Case Diagram

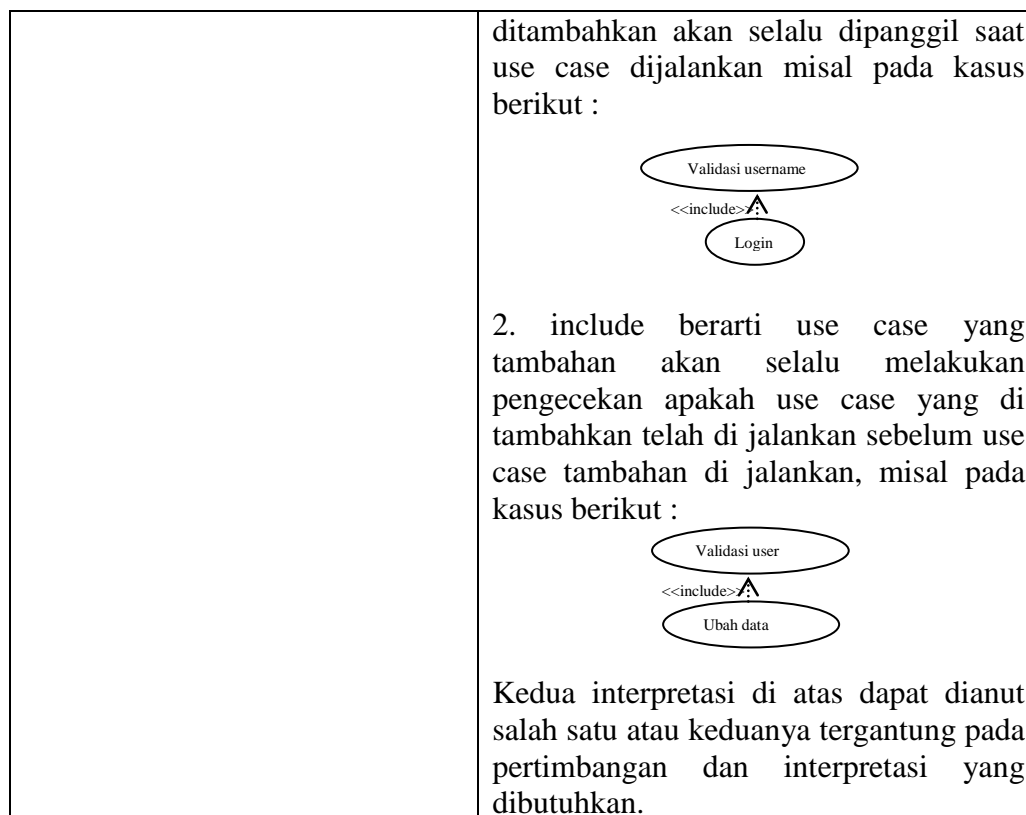
Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behaviour*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. 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*.

- 1) 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 dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- 2) *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut gambar II.9 menerangkan simbol-simbol pada diagram *use case*

Simbol	Deskripsi
Use case 	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>
Aktor / <i>actor</i> 	Orang, proses, atau sistem yang lain berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi

nama aktor	yang akan di buat itu sendiri
Asosiasi / <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> , atau <i>usecase</i> memiliki interaksi dengan aktor
Ekstensi / <i>extend</i> 	Relasi usecase 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 inheritance pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan misal  arah panah mengarah pada <i>use case</i> yang ditambahkan
Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya misalnya :  Arah panah mengarah pada use case yang menjadi generalisasinya (umum)
Menggunakan / <i>include / uses</i>  	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini Ada 2 sudut pandang yang cukup besar mengenai include di usecase 1. include berarti use case yang

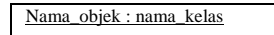

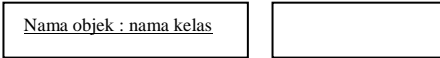




Gambar II.9 : Diagram Use case

Sumber : Rosa A.S & M. Shalahuddin (2011 : 131)

E. Communication Diagram

Diagram komunikasi mengelompokkan message pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram komunikasi yang dituliskan adalah operasi / metode yang di jalankan antara objek yang satu dengan objek lainnya secara keseluruhan, oleh karna itu dapat di ambil dari jalanya interaksi pada semua diagram sekuen. Berikut adalah gambar II.10 yang menerangkan simbol-simbol yang ada pada diagram komunikasi :

Simbol	Deskripsi
Objek 	Objek yang melakukan interaksi pesan
Link 	Relasi antar objek yang menghubungkan objek satu dengan lainnya atau dengan dirinya sendiri 
Arah pesan / stimulus 	Arah pesan yang terjadi, jika pada suatu link ada dua arah pesan yang berbeda, maka arah juga digambarkan dua arah pada dua sisi link 

Gambar II.10 : Diagram Komunikasi


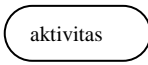



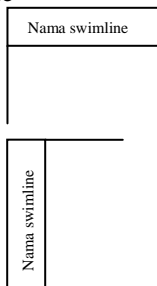
Sumber : Rosa A.S & M. Shalahuddin (2011 : 140)

F. Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

- 1) Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan
- 2) Urutan atau pengelompokan tampilan dari sistem/user interface dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan
- 3) Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujiannya.

Berikut adalah gambar II.11 yang menggambarkan simbol-simbol yang ada pada diagram aktivitas :

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / decesion 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / join 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane  atau	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Gambar II.11 : Diagram Aktivitas


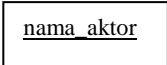

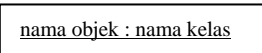

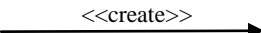
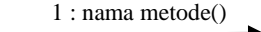
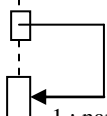
Sumber : Rosa A.S & M. Shalahuddin (2011 : 134)

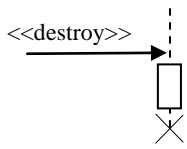
G. Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Banyaknya diagram objek yang digambarkan adalah sebanyak pendefinisian use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalanya pesan sudah dicakup dapa diagram sekuen sehingga semakin banyak use case

yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.

Berikut adalah gambar II.12 yang menerangkan simbol-simbol yang ada pada diagram sekuen :

Simbol	Deskripsi
<p>Aktor</p>  <p>nama aktor</p> <p>atau</p>  <p>nama_aktor</p> <p>tampa waktu aktif</p>	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya di nyatakan menggunakan kata benda di awali frase nama aktor</p>
<p>Garis hidup / lifeline</p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p>  <p>nama objek : nama kelas</p>	<p>Menyatakan objek yang berinteraksi pesan</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi pesan</p>
<p>Pesan tipe create</p>  <p><<create>></p>	<p>Objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p>  <p>1 : nama metode()</p>	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri</p>  <p>1 : nama_metode()</p>

	Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang di panggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
<p>Pesan tipe send</p> <p>1 : masukan -----></p>	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainya, arah panah mengarah pada objek yang dikirim
<p>Pesan tipe return</p> <p>1 : keluaran -----></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>Pesan tipe destroy</p> <p><<destroy>> </p>	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy

Gambar II.12 : Diagram Sequence

Sumber : Rosa A.S & M. Shalahuddin (2011 : 138)

II.6 Sejarah Singkat Perusahaan

Setelah masa kemerdekaan Tahun 1945 banyak anggota tentara maupun keluarganya yang mengalami sakit dan berdomisili di Medan memanfaatkan fasilitas kesehatan rumah sakit swasta yang ada disekitar Medan. Karena rumah sakit tentara satu-satunya yang ada di Sumatera Utara hanya ada di Pematang Siantar (merupakan peninggalan tentara Belanda) sementara jumlah anggota yang memanfaatkan fasilitas kesehatan ini terus bertambah dari hari kehari, untuk itu para pejuang kemerdekaan maupun dokter tentara yang ada di Medan berpikir perlu

adanya fasilitas kesehatan (Rumah sakit) khusus tentara di Kota Medan ini. Pada tahun 1950 atas prakarsa dokter militer yang diketuai Letkol dr. Moh. Majoedin mendirikan sebuah Tempat Perawatan Asrama (TPA) yang berlokasi di Jalan Banteng 2A Medan. TPA ini dipergunakan untuk merawat anggota Tentara maupun keluarga yang menderita penyakit ringan, sedangkan untuk penyakit berat dirawat di RST Pematang Siantar. TPA ini memiliki fasilitas 10 tempat tidur, laboratorium kecil, kamar obat, kamar suntik, kamar bedah kecil serta dapur.

Pada tahun 1951 Letkol Dr. Moh Majoedin sekaligus selaku Kepala Dinas Kesehatan Tk I menerima penyerahan 4 buah bangsal Rumah Sakit Verenigde Deli Maatschkapy (VDM), yaitu RS PTPN II sekarang (Dahulu RS PTP IX / Tembakau Deli) yang sebelumnya dipergunakan oleh Belanda untuk merawat Tentara Belanda yang sakit dan berlokasi di Jalan Putri Hijau Medan.

Dengan diserah terimaknya VDM tersebut maka TPA berubah menjadi Rumah Sakit Putri Hijau, yang untuk pertama kalinya pada Tahun 2006 seorang wanita memimpin Rumah Sakit Tk II Putri Hijau Kesdam I/Bukit Barisan yaitu Kolonel dr. Titut Sri Endartini, MARS.

Rumah Sakit Tk II Putri Hijau Kesdam I/BB, sekarang menjadi rumah sakit kebanggaan prajurit di wilayah Kodam I / BB dan sekaligus sebagai rumah sakit rujukan wilayah barat Indonesia dengan di dukung sarana dan prasarana yang makin memadai dan tenaga dokter spesialis dan sub spesialis yang makin lengkap. Dalam era reformasi, tuntutan masyarakat pada pelayanan yang lebih baik makin meningkat, sehingga berbagai upaya penyempurnaan dan perbaikan kualitas

pelayanan kepada masyarakat terus dilakukan secara bertahap, sehingga sekarang telah terakreditasi 5 pelayanan dasar sesuai dengan sertifikat akreditasi Keputusan Menteri Kesehatan RI Nomor : HK. 03. 05 / III / 760 / 11 tanggal 17 Maret 2011.

II.7 Visi dan Misi

II.7.1 Visi

Menjadi Rumah Sakit dambaan warga TNI dan masyarakat dikawasan barat Negara Kesatuan Republik Indonesia.

II.7.2 Misi

1. Memberikan dukungan dan pelayanan kesehatan yang tepat, akurat bagi personil beserta keluarga TNI secara profesional.
2. Turut berperan meningkatkan derajat kesehatan masyarakat dengan memberikan pelayanan dan pendidikan kesehatan yang bermanfaat secara optimal, sesuai kemampuan masyarakat.
3. Memberikan pelayanan kesehatan yang cepat, tepat dan akurat bagi personil TNI, PNS dan keluarganya.
4. Memberikan dukungan kesehatan yang handal dalam mendukung tugas pokok TNI.

5. Memberikan pelayanan yang ramah, terjangkau dan professional bagi masyarakat Medan dan sekitarnya dengan memanfaatkan fasilitas lebih yang ada di Rumah Sakit Tk II Putri Hijau Kesdam I / BB
6. Turut meningkatkan pengembangan pendidikan kesehatan di Kota Medan dan sekitarnya.

II.7.3 Motto

Kerja ikhlas dan tuntas pasien puas agar menjadi Rumah Sakit dambaan masyarakat luas.