

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Program

Konsep dasar program berisi tentang teori-teori yang berhubungan dengan pengertian program, aplikasi dan rekam medis.

2.1.1. Pengertian Program

Menurut Yuswanto (2008:8) mengatakan bahwa “Program merupakan kata, ekspresi, pernyataan atau kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur, berupa urutan langkah untuk menyelesaikan masalah yang diimplementasikan dengan menggunakan bahasa pemrograman sehingga dapat dieksekusi oleh komputer”.

Menurut Kurniawan (2011:65) mengatakan bahwa “Program adalah kumpulan instruksi yang disusun dengan urutan nalar yang tepat untuk menyelesaikan suatu persoalan. Dalam pembuatan program, pemrogram mempunyai cara pandang terhadap eksekusi sebuah program yang disebut sebagai paradigma pemrograman.

Berdasarkan definisi diatas dapat disimpulkan bahwa Program adalah pernyataan yang disusun menjadi satu dengan urutan yang tepat untuk menyelesaikan permasalahan, pemrogram mempunyai cara pandang terhadap eksekusi sebuah program yang disebut paradigma pemrograman.

2.1.2. Aplikasi

Menurut Hendrayudi (2009:143) mengatakan bahwa “Aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu.

Menurut Arifin (2009:13) mengatakan bahwa “Aplikasi merupakan perangkat lunak pendukung yang memiliki fungsi tertentu untuk membantu pengguna dalam menyelesaikan tugas tertentu untuk membantu pengguna dalam menyelesaikan tugas tertentu.

Berdasarkan definisi diatas dapat disimpulkan bahwa Aplikasi adalah kumpulan perintah program yang dibuat untuk membantu pengguna dalam menyelesaikan tugas tertentu.

2.1.3. Rekam Medis

Menurut Hanafiah dan Amir (2008:258) mengatakan bahwa “Rekam Medis adalah rekam dalam bentuk tulisan atau gambaran aktivitas pelayanan yang diberikan oleh pemberi pelayanan medis/kesehatan kepada seorang pasien.

Menurut Wildan dan Hidayat (2008:28) mengatakan bahwa “Rekam Medis merupakan suatu keterangan baik secara tertulis maupun rekaman tentang identitas klien, hasil pengkajian, atau segala pelayanan dan tindakan medis yang diberikan kepada pasien. Rekam medis dapat diartikan sebagai dokumen atau catatan tentang keadaan pasien.

Berdasarkan definisi diatas dapat disimpulkan bahwa Rekam Medis adalah tulisan atau gambaran yang diberikan oleh pemberi pelayanan medis yang berisikan tentang identitas klien, hasil pengkajian dan tindakan medis yang diberikan kepada pasien.

2.2. Peralatan Pendukung (*Tools Program*)

Peralatan pendukung yang dimaksud untuk merancang aplikasi yaitu *Database*, Visual Basic.NET, MySQL, Xampp, Crystal Report, LRS (*Logical*

Relation Structure), ERD (*Entity Relationship Diagram*), *Activity Diagram* dan *Use Case Diagram*.

2.2.1. Basis Data/Database

Menurut Priyanto (2009:239) mengatakan bahwa “*database* (basis data) adalah sebuah penyimpanan informasi terpusat agar data tersebut mudah dikelola dan dicari kembali”.

Mengingat bahwa sistem informasi menyajikan informasi yang berasal dari satu maupun beberapa data yang diinputkan dan diolah, maka tentu diperlukan sebuah aplikasi untuk menyimpan, mengolah, dan menyajikan data dan informasi tersebut secara komputerisasi, yang dikenal dengan *database* (Sibero, 2013:12).

Sedangkan menurut Pratama (2014:17) mengatakan bahwa “Elemen basis data pada sistem informasi berfungsi sebagai media untuk penyimpanan data dan informasi yang dimiliki oleh sistem informasi bersangkutan. Setiap aplikasi dan sistem yang memiliki data didalamnya (dengan disertai proses manipulasi data berupa *insert*, *delete*, *edit/update*), pasti memiliki sebuah basis data”.

Berdasarkan definisi diatas, dapat disimpulkan bahwa basis data/*database* berfungsi sebagai tempat penyimpanan informasi yang berasal dari satu atau lebih data yang diinput dan diolah yang dibutuhkan aplikasi untuk penyimpanan, mengolah, dan menyajikan data dan informasi secara komputerisasi.

2.2.2. Visual Basic.Net

Menurut Wardana (2008:11) mengatakan bahwa “Visual Basic.Net adalah salah satu program berorientasi objek, selain itu ada pula program Java dan C++ yang juga berbasis objek”.

Menurut Hidayatullah (2012:8) mengatakan bahwa “Visual Basic.Net adalah salah satu dari kumpulan *tools* pemrograman yang terdapat pada paket visual studio.Net. Pada Visual Studio. NET terdapat beberapa *tools* pemrograman lain seperti : Visual C++, Visual C#.NET, dan Visual J#.NET”.

Berdasarkan definisi diatas, dapat disimpulkan bahwa visual basic.net adalah salah satu dari kumpulan *tools* pemrograman dan berorientasi objek yang terdapat dalam visual studio.Net.

2.2.3. MySQL

MySQL adalah salah satu jenis *database server* yang terkenal. MySQL termasuk jenis RDBMS (*Relational Database Management System*). MySQL ini mendukung bahasa pemrograman PHP. MySQL juga mempunyai query atau bahasa SQL (*Structured Query Language*) yang simple dan menggunakan *escape character* yang sama dengan PHP (Kurniawan, 2010:16).

Menurut Anhar (2010:21) mengatakan bahwa “MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Database Management System*) atau DBMS dari sekian banyak DBMS, seperti Oracle, MS SQL, Postagre SQL, dan lain-lain.

Sedangkan menurut Hidayatullah (2012:21) mengatakan bahwa “MySQL atau dibaca “My Sekuel” dengan adalah suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data. MySQL pertama dikembangkan oleh MySQL AB yang kemudian diakuisi *Sun Microsystem* dan terakhir dikelola oleh Oracle Cooperation”.

Berdasarkan definisi diatas dapat disimpulkan bahwa MySQL adalah satu jenis *database server* dan termasuk jenis RDBMS (*Relational Database Management System*) yang menjalankan fungsi pengolahan data.

2.2.4. XAMPP

XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolah data MySQL di komputer lokal. XAMPP berperan sebagai server web pada komputer anda. XAMPP juga dapat disebut sebuah Cpanel server virtual, yang dapat membantu anda melakukan preview sehingga dapat memodifikasi website tanpa harus online atau terakses dengan internet (Wicaksono, 2008:7).

Menurut Imansyah (2010:4) mengatakan bahwa “XAMPP adalah *installer* yang membundel Apache, PHP dan MySQL untuk windows dalam satu paket, dengan menginstal XAMPP Anda bisa menjadikan komputer menjadi server”.

Sedangkan menurut Pratama (2014:440) mengatakan bahwa “XAMPP adalah aplikasi web server bersifat instan (siap saji) yang dapat digunakan baik di sistem operasi Linux maupun di sistem operasi windows”.

Berdasarkan definisi diatas dapat disimpulkan bahwa XAMPP adalah aplikasi web server yang berfungsi menjalankan website berbasis PHP yang menggunakan pengolah data MySQL di komputer lokal.

2.2.5. Crystal Report

Menurut Kusrini dan Kuniyo (2007:264) mengatakan bahwa “*Crystal Report* merupakan program dapat digunakan untuk membuat, menganalisis dan menerjemahkan informasi yang terkandung dalam *database* atau program ke dalam

berbagai jenis laporan yang sangat fleksibel. Beberapa kelebihan *Crystal Report* adalah :

1. Pembuatan laporannya tidak terlalu rumit sehingga memungkinkan pemrogram pemula sekalipun untuk membuat laporan tanpa harus melibatkan kode pemrograman.
2. Terintegrasi dengan berbagai bahasa pemrograman lain sehingga memungkinkan pemrogram memanfaatkannya dengan keahliannya sendiri-sendiri.
3. Fasilitas impor hasil laporan yang mendukung format yang populer seperti *Microsoft Word, Excell, Acces, Adobe Acrobat Reader*, HTML dan sebagainya

Menurut Junindar (2008:12) mengatakan bahwa “*Crystal Report* adalah program *third party* (pihak ketiga, artinya diluar *microsoft* dan pemakai) untuk membuat laporan pada aplikasi *Windows* dan *Web*”.

Berdasarkan definisi diatas dapat disimpulkan bahwa *Crystal Report* adalah program yang digunakan untuk membuat, menganalisis dan menejermahkan informasi yang terkandung dalam *database* dan untuk membuat laporan pada aplikasi *windows* dan *web*.

2.2.6. LRS (*Logical Relational Structure*)

Menurut Frieyadie (2007:13) mengatakan bahwa “LRS merupakan hasil dari pemodelan *Entity Relational Ship* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”. Dalam pembuatan LRS terdapat 3 hal yang dapat mempengaruhi menurut Frieyadie (2007:13) yaitu :

1. Jika tingkat hubungan (*cardinality*) satu pada satu (*one-to-one*), maka digabungkan dengan entitas yang lebih kuat (*strong entity*), atau digabungkan dengan entitas yang memiliki atribut yang lebih sedikit.
2. Jika tingkat hubungan (*cardinality*) satu pada banyak (*one-to-many*), maka hubungan relasi atau digabungkan dengan entitas yang tingkat hubungannya banyak.
3. Jika tingkat hubungan (*cardinality*) banyak pada banyak (*many-to-many*), maka hubungan relasi tidak akan digabungkan dengan entitas maupun, melainkan menjadi sebuah LRS.

Sebuah model sistem yang digambarkan dengan sebuah diagram ER akan mengikuti pola / aturan permodelan tertentu dalam kaitannya dengan konversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan – aturan berikut ini : setiap entitas akan diubah kebentuk kotak, sebuah atribut relasi disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada diagram ER 1:M (relasi bersatu dengan cardinality M) atau tingkat hubungan 1:1 (relasi bersatu dengan cardinality yang saling membutuhkan referensi), sebuah relasi dipisah dalam sebuah kotak tersendiri (menjadi entitas baru) jika tingkat hubungannya M:M (many to many) dan memiliki foreign key sebagai primary key yang diambil dari kedua entitas yang sebelumnya saling berhubungan (Hasugian, 2012:608).

Berdasarkan definisi diatas dapat disimpulkan bahwa LRS merupakan hasil dari pemodelan *Entity Relational Ship* (ER), mengikuti pola/aturan permodelan tertentu dalam kaitannya dengan konversi ke LRS, perubahan yang terjadi mengikuti aturan-aturannya.

2.2.7. UML (*Unified Modelling Language*)

Menurut Nugroho (2010:6) mengatakan bahwa “UML (*Unified Modelling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek’”.

Menurut Sukanto dan M.Shalahuddin (2014:133) mengatakan bahwa “UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”.

Berdasarkan definisi diatas dapat disimpulkan bahwa UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan untuk pemodelan sistem atau perangkat lunak berparadigma berorientasi objek.

2.2.8. ERD (*Entity Relationship Diagram*)

ERD adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis. Entitas biasanya menggambarkan jenis informasi yang sama. Dalam entitas digunakan untuk menghubungkan antar entitas yang sekaligus menunjukkan hubungan antar data. Pada akhirnya ERD bisa juga digunakan untuk menunjukkan aturan-aturan bisnis yang ada pada sistem informasi yang akan dibangun (Fatta, 2007:121).

ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antartabel beserta *field-field* di dalamnya pada suatu *database* sistem. Sebuah *database* memuat minimal sebuah tabel dengan sebuah atau beberapa buah *field* (kolom) di dalamnya. Namun pada kenyataannya, *database* lebih sering memiliki lebih dari satu buah tabel (dengan beberapa *field* didalamnya). Keterkaitan

antartabel ini biasa disebut dengan relasi terdapat tiga buah jenis relasi antar tabel di dalam bagan ERD (Pratama, 2014:49).

Ketiga relasi tersebut yaitu:

1. *One to One* (Satu ke Satu)

Relasi ini menggambarkan hubungan satu *field* pada tabel pertama ke satu *field* pada tabel kedua. Relasi ini paling sederhana. Sebagai contoh, pada sistem informasi perpustakaan terdapat tabel Buku (dengan *field* Kode_Buku, Kode_Kategori, Kode_Penulis, Judul, Penerbit) dan tabel Kategori (Kode_Kategori, Nama_Kategori, Alamat).

Field Kode_Kategori memiliki keterkaitan (relasi) satu ke satu pada tabel Buku dan tabel Kategori.

2. *One to Many* (Satu ke Banyak)

Relasi ini menggambarkan hubungan satu *field* pada tabel pertama ke dua atau beberapa buah *field* di tabel kedua.

3. *Many to Many* (Banyak ke Banyak)

Sebagai contoh, sebuah sistem informasi sekolah memiliki pengguna guru dan siswa di dalamnya. Sistem informasi ini memiliki sebuah database bernama sisfosekolah dengan tiga buah tabel di dalamnya. Ketiga tabel tersebut adalah tabel Guru (memuat *field* NIP, Nama_Guru, Jabatan, Pangkat_Golongan, Alamat), tabel Mata Pelajaran (memuat *field* Kode_Mata_Pelajaran, Nama_Mata_Pelajaran), dan tabel Mengajar (memuat *field* NIP, Kode_Mata_Pelajaran, Kelas).

Berdasarkan definisi diatas dapat disimpulkan bahwa ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antar

tabel dan *field-field* dalam suatu *database* sistem dan dikembangkan berdasarkan teori matematika.

2.2.9. *Activity Diagram*

Menurut Triandini dan Suardika (2012:37) mengatakan bahwa “*Activity Diagram* adalah sebuah diagram alur kerja yang menjelaskan berbagai kegiatan pengguna (atau sistem), orang yang melakukan masing-masing aktivitas dan aliran sekuensial dari aktivitas-aktivitas tersebut.”

Menurut Sukamto dan M.Shalahuddin (2015:161) mengatakan bahwa “*Activity Diagram* adalah menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

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 ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berdasarkan pengertian diatas penulis menyimpulkan, *Activity Diagram* adalah suatu yang 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 tapi aktivitas yang dapat dilakukan oleh sistem.

2.2.10. Use Case Diagram

Use Case merupakan metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks. *Use Case* menambahkan detail untuk kebutuhan yang telah dituliskan pada definisi sistem kebutuhan. *Use Case* dikembangkan oleh analis sistem bersama-sama dengan pengguna. Pada tahap selanjutnya, berdasarkan *use case* ini, analis menyusun model data dan model proses (Fatta 2007:91).

Menurut Sukamto dan M.Shalahuddin (2015:156) mengatakan bahwa “*Use Case* atau diagram use case 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. 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.

Berdasarkan definisi diatas dapat disimpulkan *Use Case* adalah metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks dan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat.