

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Program

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

2.1.1 Pengertian Program

Menurut Yuswanto (2008:8) mendiskripsikan 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 bahasa pemrograman sehingga dapat dieksekusi oleh komputer”.

Menurut Kadir (2015:1) mendefinisikan bahwa, “Program adalah kumpulan instruksi yang ditujukan untuk komputer supaya peralatan tersebut dapat melakukan tindakan-tindakan yang dikehendaki oleh pemakai program (*user*)”.

Berdasarkan pengertian diatas penulis menyimpulkan Program adalah kumpulan instruksi yang ditujukan untuk komputer menyelesaikan masalah yang diimplementasikan dengan bahasa pemrograman sehingga dapat dieksekusi oleh komputer”.

2.1.2 Pengertian Aplikasi

Menurut Hendrayudi (2008:143) mendeskripsikan bahwa, “Aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu (khusus)”.

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

Berdasarkan pengertian diatas, penulis menyimpulkan aplikasi adalah kumpulan perintah program yang dibuat untuk membantu pengguna dalam menyelesaikan tugas tertentu dan pekerjaan-pekerjaan tertentu”.

2.1.3 Pengertian Penjualan

Menurut Siswosoediro (2008:61) mendefinisikan bahwa, ”Penjualan lansung (*direct selling*) adalah metode penjualan barang dan/atau jasa tertentu melalui jaringan pemasaran yang di kembangkan oleh mitra usaha yang bekerja”.

Menurut Sujarweni (2015:79) “ Penjualan tunai merupakan system yang berlaku oleh perusahaan dalam menjual barang dengan cara mewajibkan pembeli untuk melakukan pembayaran harga terlebih dahulu sebelum barang diserahkan kepada pembeli”.

Berdasarkan pengertian diatas, penulis menyimpulkan penjualan adalah metode penjualan barang dan/atau jasa tertentu melalui jaringan pemasaran oleh perusahaan dalam menjual barang dengan cara mewajibkan pembeli untuk melakukan pembayaran.

2.1.4 Pengertian Piutang Dagang

Menurut Johar (2008:51) “Piutang dagang berarti menghutangkan kepada pihak lain”.

Menurut Arief, Yanuar, dan Synthia (2009:161) "Piutang merupakan semua tagihan kepada seseorang ataupun badan usaha atau kepada pihak lainnya dalam satuan mata uang yang timbul sebagai akibat transaksi di masa lampau".

Dari pengertian diatas penulis menyimpulkan bahwa Piutang dagang berarti menghutangkan kepada pihak lain dalam satuan mata uang yang timbul sebagai akibat transaksi di masa lampau.

2.2 Peralatan Pendukung (*Tools Program*)

Peralatan pendukung mampu mendeskripsikan sistem yang sedang berjalan yaitu *Object Oriented Programming* (OOP) dan *Unified Modelling Language* (UML). *Unified Modelling Language* (UML) meliputi *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *Sequence Diagram*. Netbeans, PHPmyAdmin dan Xampp. Penjelasan mengenai peralatan pendukung akan dijelaskan lebih lanjut.

2.2.1 *Unified Modeling Language* (UML)

Menurut Nugroho (2010:6), "UML (*Unified Modeling Language*) adalah 'bahasa' pemodelan untuk sistem atau perangkat lunak yang berparadigma 'berorientasi objek'".

Menurut Rosa dan Shalahuddin (2014:133), "UML (*Unified Modeling Language*) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek".

Dari pengertian diatas penulis menyimpulkan bahwa UML (*Unified Modeling Language*) adalah 'bahasa' pemodelan untuk sistem atau perangkat

lunak yang mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur yang berparadigma ‘berorientasi objek’.

2.2.2 Database

Menurut Kusrini (2007:2) “Basis data adalah kumpulan data yang saling berelasi. Data sendiri merupakan fakta mengenai objek, orang, dan lain-lain. Data dinyatakan dengan nilai (angka, deretan karakter, atau simbol)”.

Basis data dapat didenifisikan dalam berbagai sudut pandang seperti berikut:

1. Himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa sehingga kelak dapat dimanfaatkan dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (*redundancy*) yang tidak perlu, untuk memenuhi kebutuhan.
3. Kumpulan *file/table/arsip* yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Menurut Pratama (2014:17), “Umumnya sebuah basis data memiliki satu atau beberapa buah *table*. Setiap *table* memiliki *field* masing-masing. Kedalam *table* dan *field* inilah data disimpan oleh pengguna melalui tatap muka aplikasi yang disediakan atau langsung melalui perintah diterminal (*command line*)”.

Dari pengertian diatas penulis menyimpulkan *Database* adalah kumpulan data yang saling berelasi. Data sendiri merupakan fakta mengenai objek, orang, dan lain-lain, data disimpan oleh pengguna melalui tatap muka aplikasi yang disediakan atau langsung melalui perintah diterminal (*command line*)”.

2.2.3. Visual Basic.Net

Menurut Wardana (2008:11), “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), “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: VisualC++, Visual C#.NET, dan Visual J#.NET”.

Dari pengertian diatas penulis menyimpulkan Visual Basic.Net adalah salah satu program berorientasi objek yang mempunyai beberapa *tool* pemrograman yang terdapat pada paket Visual Studio.NET. Pada Visual Studio.NET terdapat beberapa *tools* pemrograman lain, seperti: VisualC++, Visual C#.NET, dan Visual J#.NET”.

2.2.4. 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 (*Structure Query Language*) yang *simple* dan menggunakan *escape character* yang sama dengan PHP (Kurniawan, 2010:16).

Menurut Anhar (2010:12) 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, *Portagre SQL*, dan lain-lain.

Dari pengertian diatas penulis menyimpulkan MySQL adalah salah satu perangkat lunak sistem manajemen basis data SQL (*Database Management System*). MySQL termasuk jenis RDBMS (*Relational Database Management System*). MySQL ini mendukung bahasa pemrograman PHP.

2.2.5. 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 analisis sistem bersama-sama dengan pengguna. Pada tahap selanjutnya, berdasarkan *use case* ini, analisis menyusun model data dan model proses (Al Fatta 2007:91)

Menurut Sukamto dan Shalahuddin (2015:156) mendefinisikan bahwa, “*Use case* atau *diagram use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

Dari pengertian diatas penulis menyimpulkan *Use Case* merupakan metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks, pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat”.

2.2.6. 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 Shalahuddin (2015:161) mendeskripsikan bahwa, “*Activity Diagram* adalah menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”.

Dari pengertian diatas penulis menyimpulkan, *Activity Diagram* adalah sebuah diagram alur kerja yang menjelaskan berbagai kegiatan pengguna (atau sistem), menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau menu yang ada pada perangkat lunak”.

2.2.7. Pengertian Xampp

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

Menurut Pratama (2014:440), “Xampp adalah aplikasi *web server* bersifat instan (siap saji) yang dapat digunakan baik di sistem operasi Linux maupun dari sistem operasi Windows”.

Dari pengertian diatas penulis dapat menyimpulkan bahwa, Xampp adalah *installer* yang membundel *Aphace*, PHP, dan MySQL bersifat instan (siap saji) yang dapat digunakan baik untuk Windows dan Linux.

2.2.8. ERD (*Entity Relationship Diagram*)

Entity Relationship Diagram adalah diagram yang menggambarkan keterkaitan antartabel beserta dengan *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* di dalamnya). Setiap tabel umumnya memiliki keterkaitan hubungan. Keterkaitan

antartabel ini bisa disebut dengan Relasi. Terdapat tiga buah jenis relasi antar tabel didalam bagan ERD (Pratama, 2014:49).

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_Katagori, Kode_Penulis, Nama_Penulis, Judul, Penerbit) dan tabel Katagori (Kode_Katagori, Nama_Katagori, Alamat). *Field* Kode_Katagori memiliki keterkaitan (relasi) satu ke satu pada table Buku dan table Katagori.

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

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

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

Sebagai contoh, sebuah sistem informasi sekolah memiliki penggunaan guru dan siswa di dalamnya. Sistem informasi ini memiliki sebuah database bernama sisfosekolah dengan tiga buah tabel didalamnya. 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.

Menurut Al Fatta (2009:27) menyatakan bahwa “(*Entity Relationship Diagram*) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD merupakan model jaringan data yang menekankan pada struktur dan hubungan antar data *store* pada DFD. Diagram hubungan entitas, atau yang lebih dikenal E-R, adalah notasi grafik dari sebuah

model data atau sebuah model jaringan yang menjelaskan tentang data yang disimpan (*storage data*) dalam sistem secara abstrak. Diagram hubungan entitas tidak menyatakan bagaimana memanfaatkan data, membuat data, mengubah data, dan menghapus data.

Dari pengertian diatas penulis menyimpulkan ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antartabel beserta dengan *field-field* di dalamnya yang menggunakan susunan data yang saling berelasi, diagram hubungan entitas tidak menyatakan bagaimana memanfaatkan data, membuat data, mengubah data, dan menghapus data.

2.2.9. LRS (*Logical Record Structure*)

Menurut Frieyadie (2007:13) LRS merupakan permodelan *Entity Relationship* (ER) beserta atributnya sehingga bisa terlihat hubungan antar entitas.

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 bahwa 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).

Dari pengertian diatas penulis menyimpulkan LRS (*Logical Record Structure*) adalah permodelan *Entity Relationship* (ER) beserta atributnya, diagram ER akan mengikuti pola / aturan permodelan tertentu dalam kaitannya dengan konversi ke LRS.

2.2.10. Crystal Report

Menurut Kusrini dan Koniyo (2007:264) mengatakan bahwa “Crystal Report merupakan program yang 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 pemrograman pemula sekalipun untuk membuat laporan tanpa harus melibatkan kode pemrograman.
2. Terintegrasi dengan berbagai bahasa pemrograman lain sehingga memungkinkan pemrograman 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”.

Dari pengertian diatas, penulis menyimpulkan Crystal Report adalah program *third party* (pihak ketiga, artinya diluar *microsoft* dan pemakai) untuk

membuat laporan pada aplikasi Windows dan *Web*, menganalisis dan menerjemahkan informasi yang terkandung dalam *database* atau program ke dalam berbagai jenis laporan yang sangat fleksibel.