

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Sistem menurut sejarahnya berasal dari bahasa Yunani yaitu "*Sistema*" yang berarti kesatuan, yakni keseluruhan dari bagian-bagian yang mempunyai hubungan satu dengan yang lainnya. Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisir, saling berinteraksi, saling ketergantungan satu sama lainnya dan terpadu.

Dalam konsep dasar sistem ini akan dijelaskan tentang pengertian sistem, karakteristik sistem, klasifikasi sistem, pengertian informasi, pengertian sistem informasi, pengertian sistem informasi akuntansi, serta tentang pengertian SPP.

2.1.1. Pengertian Sistem

Menurut Jerry Fitzgrald dalam Puspitawati (2011a:1) menyatakan bahwa "Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu".

Sutabri (2012a:11) mengemukakan bahwa "sebuah sistem terdiri atas bagian-bagian atau komponen yang terpadu untuk suatu tujuan".

Sedangkan menurut Fatta (2007a:3) mengartikan sistem sebagai "suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain".

Dari beberapa definisi diatas, dapat disimpulkan bahwa sistem merupakan suatu kesatuan atau kumpulan komponen yang saling berhubungan, saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain untuk mencapai suatu tujuan yang sama.

2.1.2. Karakteristik Sistem

Menurut Sutabri (2012b : 20) “Sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem”.

Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan.

2. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan Luar Sistem (*Environtment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.

7. Pengolah Sistem (Proses)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*.

2.1.3. Klasifikasi Sistem

Klasifikasi sistem menurut Adzhar Susanto dalam Puspitawati (2011b : 6) adalah :

1. Sistem Terbuka dan Tertutup

Suatu sistem dikatakan terbuka apabila aktivitas didalam sistem tersebut dipengaruhi oleh lingkungannya, sedangkan suatu sistem dikatakan tertutup apabila aktivitas-aktivitas dalam sistem tersebut tidak terpengaruh oleh perubahan yang terjadi di lingkungannya.

2. Sistem Buatan Manusia dan Tuhan

Suatu sistem bila diklasifikasikan berdasarkan asalnya, sistem tersebut bisa diklasifikasikan sebagai sistem yang ada secara alamiah (buatan Tuhan)

atau buatan manusia. Kita adalah sistem yang secara alamiah demikian pula dengan pohon-pohon yang ada disekitar kita, sedangkan mobil merupakan buatan manusia.

3. Sistem Berjalan dan Konseptual

Suatu sistem yang belum diterapkan disebut sebagai sistem konseptual. Suatu sistem konseptual yang dapat diterima oleh pemakai sistem sehingga pemakai sistem tersebut menggunakannya untuk menunjang operasi sehari-hari maka sistem tersebut berubah menjadi sistem berjalan.

4. Sistem Sederhana dan Kompleks

Sebuah sistem yang sederhana merupakan sebuah sistem yang terbentuk dari sedikit tingkatan dan komponen atau subsistem serta hubungan antara mereka sangat sederhana. Sebuah sistem yang kompleks jelas terdiri dari banyak komponen dan tingkatan yang dihubungkan dalam berbagai cara yang berbeda.

5. Kinerja yang dapat dan tidak dapat dipastikan

Sebuah sistem yang dapat dipastikan kinerjanya artinya dapat ditentukan pada saat sistem akan sedang dibuat. Sedangkan sistem yang tidak dapat dipastikan kinerjanya artinya tidak dapat ditentukan dari awal tergantung kepada situasi yang dihadapi.

6. Sementara dan selamanya

Suatu sistem yang mungkin digunakan untuk selamanya mungkin juga digunakan untuk periode waktu tertentu.

7. Secara fisik dan abstrak

Sistem dapat dilihat dari wujudnya misalnya kendaraan bermotor, sedangkan yang abstrak seperti organisasi.

8. Sistem, subsistem dan supersistem

Subsistem adalah sistem yang lebih kecil dalam sebuah sistem, sedangkan supersistem adalah sistem yang lebih besar.

9. Bisa beradaptasi dan tidak bisa beradaptasi

Berdasarkan fleksibelitasnya kita dapat membedakan karakteristik suatu sistem apakah sistem tersebut dapat beradaptasi terhadap perubahan yang terjadi di lingkungannya atau tidak.

2.1.4. Pengertian Informasi

Fatta (2007b:9) menjelaskan bahwa “Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang”.

Menurut Sutabri (2012c:29) “Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan”.

Menurut Pratama (2014a:9) menyatakan bahwa “Informasi didefinisikan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah sehingga memberikan nilai, arti dan manfaat”.

Dari beberapa pengertian diatas maka dapat disimpulkan bahwa informasi adalah hasil pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimannya dan digunakan untuk pengambilan keputusan.

2.1.5. Pengertian Sistem Informasi

Pratama (2014b:10) menjelaskan bahwa “Sistem informasi merupakan gabungan dari empat bagian utama. Keempat bagian utama tersebut mencakup

perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur, dan Sumber Daya Manusia (SDM) yang terlatih”.

Menurut Dianingrum dkk dalam Jurnal Telematika (2012a :105) menjelaskan bahwa “sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi”.

Sudirman dkk dalam Jurnal Algoritma Sekolah Tinggi Teknologi Garut (2012:2) menjelaskan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Dari beberapa pendapat diatas dapat disimpulkan bahwa sistem informasi adalah gabungan dari empat bagian utama yang saling berhubungan satu sama dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berguna.

2.1.6. Pengertian Sistem Informasi Akuntansi

Kusrini (2007:1) menjelaskan bahwa “Sistem informasi akuntansi dapat didefinisikan sebagai sistem informasi yang mengubah data transaksi bisnis menjadi informasi menjadi informasi keuangan yang berguna bagi pemakainya”.

Diana (2011:4) menyatakan bahwa ”sistem informasi akuntansi adalah sistem yg bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan transaksi keuangan”.

Sedangkan menurut TMBooks (2015 : 2) menyatakan “SIA merupakan sistem yang menyediakan informasi akuntansi keuangan beserta informasi lainnya yang diperoleh dari proses rutin transaksi akuntansi”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa sistem informasi akuntansi adalah sistem yang menyediakan informasi akuntansi yang bertujuan untuk mengumpulkan dan memproses data serta melaporkan informasi yang berkaitan dengan keuangan yang berguna bagi pemakainya.

2.1.7. Pengertian SPP

Dianingrum dkk dalam Jurnal Telematika (2012b :105) menjelaskan “ SPP (Sumbangan Penunjang Pendidikan) adalah iuran atau pembayaran setiap bulan dari siswa yang menjadi kewajiban bagi siswa di sekolah”.

Menurut Hayati dkk dalam Jurnal Teknologi Informasi dan Komunikasi (2013:66) menyimpulkan bahwa :

SPP dimaksudkan untuk membantu pembinaan pendidikan seperti yang ditunjukkan pada pasal 12 keputusan tersebut yakni untuk membantu penyelenggaraan sekolah, kesejahteraan personal, perbaikan sarana dan prasarana dan kegiatan supervisi.

Dari beberapa pendapat diatas dapat disimpulkan bahwa spp adalah iuran atau pembayaran wajib setiap bulan yang bertujuan untuk membantu pembinaan pendidikan.

2.2. Peralatan Pendukung (Tools System)

Tools System atau peralatan pendukung yang penulis gunakan dalam pembuatan Tugas Akhir ini meliputi *Unified Modelling Language (UML)* berupa *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Deployment Diagram*, *Entity Relationship Diagram (ERD)*, *Logical Record Structure (LRS)* beserta *Object Oriented Programming (OOP)*, *Netbeans*, *PhpMyAdmin*, *XAMPP*, *StarUML* dan peralatan lainnya yang membantu penulis dalam pembuatan Laporan Tugas Akhir.

2.2.1. *Unified Modelling Language (UML)*

Rosa (2014a : 133) mengemukakan bahwa “*Unified Modelling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemograman berorientasi objek”.

Sedangkan menurut Hendini dalam Jurnal Khatulistiwa Informatika (2016:108) menjelaskan “*Unified Modeling Language (UML)* adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak”.

Dari beberapa pengertian diatas dapat disimpulkan bahwa *Unified Modeling Language (UML)* adalah salah satu standar bahasa spesifikasi yang dipergunakan untuk menggambarkan arsitektur dalam pemograman berorientasi objek.

2.2.2. *Use Case Diagram*

Use Case Diagram adalah *Diagram* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen (Pilone dalam Muslihudin,2016a :68).

Use Case Diagram adalah sebagai urutan langkah-langkah yang secara tindakan saling terkait, baik itu terkomputerisasi maupun manual (Whitten dalam Muslihudin,2016b :69).

Menurut Rosa (2014b : 155) *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. 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 diluar 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.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *usecase diagram* adalah urutan langkah-langkah suatu interaksi yang saling terkait antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

2.2.3. *Activity Diagram*

Activity Diagram adalah *Diagram* yang memperlihatkan gambaran dari aliran suatu aktivitas ke aktivitas lainya dalam suatu sistem (Muslihudin,2016c:63).

Menurut Rosa (2014c : 161) 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. Diagram aktivitas juga banyak digunakan untuk mendefinisikan ha-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

Dari beberapa pendapat diatas dapat disimpulkan bahwa *activity diagram* adalah gambaran aliran kerja (*workflow*) dalam suatu sistem.

2.2.4. Sequence Diagram

Menurut Rosa (2014d : 165) Diagram sekuen menggambarkan kelakuan objek pada *usecase* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *usecase* beserta metode-metode yang dimiliki kelas yang diinstalasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *usecase*.

2.2.5. Deployment Diagram

Deployment Diagram adalah *Diagram* yang bersifat statis, *Diagram* ini yang memperlihatkan konfigurasi saat aplikasi dijalankan memuat komponen-komponen yang ada didalamnya berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan (Muslihudin,2016d :63).

Menurut Rosa (2014e : 154) *Diagram deployment* atau *deployment Diagram* adalah diagram yang menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. *Diagram deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device*, *node*, dan *hardware*.

2. Sistem *client/server*
3. Sistem terdistribusi murni
4. Rekayasa ulang aplikasi

Dari beberapa pendapat diatas dapat disimpulkan bahwa *deployment diagram* adalah *Diagram* yang memperlihatkan konfigurasi komponen dalam aplikasi yang dijalankan berguna.

2.2.6. Entity Relationship Diagram (ERD)

Ladjamudin (2013:142) menyatakan “ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

Sedangkan menurut Rahmayu dalam Jurnal Khatulistiwa Informatika (2015a:161) “ERD merupakan *tool* analisis sistem pertama yang memusatkan pada data dan keterkaitan antar data serta pengorganisasian data”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa ERD adalah suatu model jaringan yang memusatkan pada data dan keterkaitan antar data serta pengorganisasian data yang disimpan dalam sistem secara abstrak.

2.2.7. Logical Record Structure (LRS)

Rahmayu dalam Jurnal Khatulistiwa Informatika (2015b:162) menyatakan bahwa “ LRS merupakan hasil dari pemodelan *Entity Relationship (ER)* beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”.

Sedangkan menurut Iskandar dalam Jurnal Basis Data (2008 :126) “*Logical Record Structure* terdiri dari *link -link* diantara tipe *record*. *Link* ini menunjukkan arah dari satu tipe *record* lainnya. Banyak *link* dari LRS yang diberi tanda *field-field* yang kelihatan pada kedua *link* tipe *record*”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa LRS merupakan hasil pemodelan dari *Entity Relationship (ER)* yang terdiri dari *link -link* diantara tipe *record*.

2.2.8. Object oriented Programming (OOP)

Supardi (2008: 225) mengemukakan bahwa “*Object Oriented Programming (OOP)* adalah cara berpikir baru terhadap pemograman, dimana program terdiri atas objek objek yang mempunyai metode dan properti”.

Wibowo dalam Jurnal Khatulistiwa Informatika (2015: 152) mengemukakan bahwa “*OOP (Object Oriented Programing)* merupakan suatu cara atau paradigma pemrograman yang berorientasikan kepada objek”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa OOP merupakan cara berpikir baru terhadap pemograman yang berorientasi pada objek dalam mrnganalisa sistem dan permasalahan pemrograman.

2.2.9. Netbeans

Menurut Wahana Komputer (2015 : 20) “*Netbeans 8.0* adalah salah satu aplikasi IDE yang digunakan oleh *developer software* komputer untuk menulis, meng-*compile*, mencari kesalahan, dan untuk menyebarkan program. *Netbeans* mempunyai sekumpulan *software* modul yang dipakai untuk membuat suatu aplikasi”.

Prasetyo dalam Jurnal Ilmiah DASI (2015 : 31) menjelaskan bahwa “*Netbeans* merupakan IDE (*Integreted Development and Environment*) untuk membuat aplikasi *java*. Karena didukung dengan fasilitas *drag and drop* komponen yaitu *rapid application development* (pemrograman berbasis visual)”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *Netbeans* adalah suatu program aplikasi yang digunakan untuk membuat aplikasi *java*.

2.2.10. *PhpMyAdmin*

Wahana Komputer (2015:13) menyatakan “*PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP, yang digunakan untuk menangani administrasi *MySql* melalui *World Wide Web*”.

Menurut Purwanto (2012:5) menyimpulkan bahwa: *PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *MySQL* melalui Jejaring Jagat Jembar (*World Wide Web*). *PhpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya (mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perijinan (*permissions*), dan lain-lain.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *MySQL*.

2.2.11. *XAMPP*

Menurut Wahana Komputer (2015:55) mengemukakan bahwa “*XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySql*, *PHP*, *Perl*. *XAMPP* adalah *tool* yang menyediakan paket perangkat lunak dalam satu buah paket”.

Sedangkan Hendrianto dalam Jurnal JNS- *Indonesian Journal on Networking and Security* (2014:59) menjelaskan bahwa “*XAMPP* merupakan paket php berbasis open source yang dikembangkan oleh sebuah komunitas *Open Source*”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *XAMPP* adalah paket perangkat lunak dalam satu buah paket php berbasis *open source*.

2.2.12. *StarUML*

Iswari dalam Jurnal ULTIMATICS (2015:73) menjelaskan bahwa “*StarUML* merupakan proyek *open source* untuk mengembangkan *platform Unified Modelling Language* (UML) atau *Model Driven Architecture* (MDA) yang cepat, fleksibel, dapat diperluas, memiliki banyak fitur, dan tidak dipungut biaya”.

Menurut Triandini (2012:1) “*StarUML* adalah *platform* pemodelan perangkat lunak yang mendukung UML (*Unified Modelling Language*)”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *StarUML* adalah proyek *open source* untuk mengembangkan *platform* pemodelan perangkat lunak yang mendukung UML (*Unified Modelling Language*) yang fleksibel.

2.2.13. *IReport*

Menurut Samsudin (2012:8) menyatakan bahwa “*Ireport* adalah sebuah *tool* yang digunakan untuk membuat *design* laporan pada *Jasper Reports*”.

Supriyatno (2010:182) menjelaskan bahwa “*Ireport* merupakan *software* yang digunakan untuk membuat dan mendesain *template report*/laporan dengan tampilan GUI, sehingga anda tidak perlu mengetik secara manual *file XML* untuk membuat *template reportnya*”.

Dari beberapa pendapat diatas dapat disimpulkan bahwa *Ireport* adalah *software* yang digunakan untuk membuat dan mendesain *template report*/laporan pada *Jasper Reports*.