

BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Pada dasarnya sistem adalah sekelompok komponen dan elemen yang digabungkan menjadi satu untuk mencapai tujuan tertentu. Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan yang bertujuan menyediakan informasi untuk membantu mengambil keputusan manajemen operasi perusahaan dari hari ke hari serta menyediakan informasi yang layak untuk pihak di luar perusahaan. Istilah sistem secara umum dapat didefinisikan sebagai kumpulan hal atau elemen yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan.

2.1.1. Pengertian Sistem

Menurut (Jogiyanto, 2005) sistem adalah :“Suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”.

Berdasarkan uraian, penulis menyimpulkan bahwa sistem adalah serangkaian prosedur yang saling berhubungan untuk mencapai tujuan tertentu. Sistem mempunyai beberapa karakteristik sebagai berikut:

- A. Batasan (*Boundary*) Penggambaran dari suatu elemen atau unsur mana yang termasuk didalam sistem dan mana yang diluar sistem.

- B. Lingkungan (*Environment*) Segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
- C. Masukan (*input*) Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dimanipulasi oleh suatu sistem.
- d. Keluaran (*Output*) Sumber daya atau produk (informasi, laporan, dokumen, tampilan layar komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.

2.1.2. Karakteristik Sistem

Suatu sistem mempunyai ciri-ciri karakteristik yang terdapat pada sekumpulan elemen yang harus dipahami dalam mengidentifikasi pembuatan sistem. Adapun karakteristik sistem (Hutahaean, 2016) yang dimaksud adalah sebagai berikut:

1. Komponen

Sistem terdiri dari sejumlah komponen yang saling berinteraksi dan bekerja sama untuk membentuk satu kesatuan. Komponen sistem dapat berupa sub sistem atau bagi-bagian dari sistem.

2. Batasan Sistem (*Boundary*)

Daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luar dinamakan dengan Batasan sistem. Batasan sistem ini memungkinkan sistem dipandang sebagai satu kesatuan dan juga menunjukkan ruang lingkup (*scope*) dari sistem tersebut

3. Lingkungan Luar Sistem (*Environment*)

Apapun yang berada di luar batas dari sistem dan mempengaruhi sistem tersebut dinamakan dengan lingkungan luar sistem. Lingkungan luar yang

bersifat menguntungkan wajib dipelihara dan yang merugikan harus dikendalikan agar tidak mengganggu kelangsungan sistem.

4. Penghubungan Sistem (Interface)

Media penghubung diperlukan untuk mengalir sumber-sumber daya dari sub sistem ke sub sistem lainnya dinamakan dengan penghubung sistem.

5. Masukkan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem dinamakan dengan masukan sistem (*input*) dapat berupa perawatan dan masukan sinyal. Perawatan ini berfungsi agar sistem dapat beroperasi dan masukan sinyal adalah energi yang diproses untuk menghasilkan keluaran (*output*).

6. Keluaran Sistem (*Output*)

Hasil dari energi yang telah diolah dan diklasifikasikan menjadi keluaran yang berguna dinamakan dengan keluaran sistem (*output*). Informasi merupakan contoh keluaran sistem

7. Pengolah Sistem

Untuk mengolah masukan menjadi keluaran diperlukan suatu pengolah yang dinamakan dengan pengolah sistem.

8. Sasaran Sistem

Sistem pasti memiliki tujuan atau sasaran yang sangat menentukan input yang dibutuhkan oleh sistem dan keluaran yang dihasilkan

2.1.3 Klasifikasi Sistem

Menurut (Yakub, 2012), pada buku Pengantar Sistem Informasi, Sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya :

A. Sistem abstrak (*abstract system*)

Sistem Abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem teologia yang berisi gagasan tentang hubungan manusia dengan Tuhan merupakan contoh *abstract system*.

B. Sistem fisik (*physical system*)

Sistem fisik adalah sistem yang ada secara fisik, Sistem komputer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system*.

C. Sistem tertentu (*deterministic system*)

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.

D. Sistem tak tentu (*probabilistic system*)

Sistem tak tentu adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksikan dengan pasti.

E. Sistem tertutup (*close system*)

Sistem tertutup merupakan sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan, misalnya reaksi kimia dalam tabung terisolasi.

F. Sistem terbuka (*open system*)

Sistem ini adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan.

2.1.4 Pengertian Informasi

Informasi merupakan hal yang sangat penting bagi perusahaan dalam mengambil setiap pengambilan keputusan. Secara Etimologi, Informasi berasal dari bahasa Perancis kuno yaitu *informacion* (tahun 1387) yang diambil dari bahasa latin *informationem* yang berarti “garis besar, konsep, ide”

Pengertian menurut (Krismaji, 2015) Informasi adalah “data yang telah diorganisasi dan telah memiliki kegunaan dan manfaat”. Hal serupa disampaikan oleh (B. Romney & Steinbart, 2015) : Informasi (*information*) adalah data yang telah dikelola dan diproses untuk memberikan arti dan memperbaiki proses pengambilan keputusan. Sebagaimana perannya, pengguna membuat keputusan yang lebih baik sebagai kuantitas dan kualitas dari peningkatan informasi

2.3. Metode SJF (Shortest Job First)

2.3.1. SJF-Non Preemptive

Menurut . (Pangera, Ariyus, & Sigit Suyantoro, 2005) Penjadwalan SJF-Non Preemptive Merupakan penjadwalan yang *Non-Preemptive* dan tidak berprioritas, jarang digunakan, menggunakan asumsi waktu jalan proses sudah diketahui, menjadwalkan proses dengan waktu terpendek dijalankan lebih dahulu dan time around time rendah. masalah yang terjadi tidak dapat mengetahui ukuran job saat job masuk dan untuk proses yang datang tidak bersamaan menggunakan penetapan dinamis

2.3.2 Tahapan Penyelesaian Menggunakan Metode Shortest Job First

Berikut ini contoh tabel *Shortest Job First Non Preemptive* ditunjukkan pada tabel 2.1

Tabel 2.1 *Shortest Job First Non Preemptive*

Proses	Waktu
P1	8
P2	8
P3	4
P4	4

Sumber : (Santika & Hansun, 2014). Implementasi Algoritma Shortest Job First dan Round Robin pada Sistem Penjadwalan Pengiriman Barang

Jika urutan pengerjaannya :

- Job A, B, C, D
- Job B, C, D, A

Dengan pengerjaan job berdasarkan urutan (a), maka berturut-turut waktu yang dibutuhkan untuk proses A, B, C, D adalah 8, 12, 16, 20 sehingga dapat dihitung waktu rata-rata = $(8 + 12 + 16 + 20) / 4 = 14$.

Bila job yang dikerjakan berdasarkan (b), yaitu dengan shortest job first, maka waktu yang dibutuhkan untuk proses B, C, D, A adalah 4, 8, 12, 20 atau rata-rata = $(4 + 8 + 12 + 20) / 4 = 11$.

Pada algoritma ini setiap proses yang ada di ready queue akan dieksekusi berdasarkan burst time terkecil. Hal ini mengakibatkan waiting time yang pendek untuk setiap proses dan karena hal tersebut maka waiting time rata-ratanya juga menjadi pendek, sehingga dapat dikatakan bahwa algoritma ini adalah algoritma yang optimal

2.3.3 Manfaat dari Metode *SJF Non-Preemptive*

Salah satu manfaat sistem penjadwalan pengiriman barang metode Shortest Job First adalah prioritas jadwal pengiriman barang berdasarkan jarak terdekat dan beban barang yang sedikit sehingga waktu tunggu untuk pengiriman selanjutnya tidak memakan waktu lama

2.4. Rapid Application Development (RAD)

Menurut (Sarosa & Samiaji, 2017) menyatakan bahwa “Rapid Application Development (RAD) adalah salah satu metodologi yang pertama kali muncul untuk menjawab masalah kelambatan penyelesaian pengembangan system informasi pada SDLC konvensional”.

Menurut (S. & Shalahuddin, 2014) menjelaskan bahwa “Rapid Application Development (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek”.

Dari kedua defenisi di atas, dapat penulis simpulkan bahwa metode Rapid Application Development (RAD) merupakan suatu metode pengembangan system yang dapat mempersingkat waktu pengerjaan dari sebuah sistem dengan tahapan - tahapannya yang sangat terstruktur.

Metode RAD membagi tim pengembang menjadi beberapa tim untuk mengerjakan beberapa komponen masing-masing tim pengerjaan dapat dilakukan

a) Pemodelan Bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa saja yang terkait proses bisnis, informasi apa saja yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yang terkait informasi itu.

b) Pemodelan Data

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

c) Pemodelan Proses

Mengimplementasikan fungsi bisnis yang sudah didefenisikan terkait dengan pendefenisian data.

d) Pembuatan Aplikasi

Mengimplementasikan pemodelan proses dan data menjadi program.

Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

e) Pengujian dan pergantian

Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya

2.5. Perangkat Lunak Yang Digunakan

2.5.1. XAMPP

Menurut (Madcoms, 2016), sekarang ini banyak paket *software* instalasi *web server* yang disediakan secara gratis diantaranya menggunakan XAMPP. Dengan menggunakan paket *software* instalasi ini, maka sudah dapat melakukan beberapa instalasi *software* pendukung *web server*, yaitu *Apache*, *PHP*, *phpMyAdmin*, dan *database* MySQL. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, MySQL *database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, MySQL, PHP, dan *Perl*. Program ini tersedia dalam GNU *General Public License* dan bebas digunakan untuk umum.

2.5.2. PHP

Menurut (Madcoms, 2016), PHP adalah pemrograman interpreter yaitu proses penerjemahan baris kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan atau sering disebut suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan istilah *open source* yaitu pengguna dapat mengembangkan kode-kode fungsi PHP sesuai dengan kebutuhannya. Dapat ditarik kesimpulan bahwa PHP adalah bahasa pemrograman yang digunakan secara luas untuk menangani pembuatan dan pengembangan sebuah situs *web* dan bisa digunakan bersamaan dengan HTML.

2.5.3. HTML (Hyper Text Markup Language)

Menurut (Anhar, 2010), HTML (*Hypertext Markup Language*) adalah sekumpulan simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang digunakan untuk menampilkan halaman pada *web browser*. Pada dokumen HTML yang termasuk sistem hypertext, kita tidak harus membaca dokumen tersebut secara urut dari atas ke bawah atau sebaliknya, tetapi kita dapat menuju topik tertentu secara langsung dengan menggunakan teks penghubung yang akan membawa anda ke suatu topik atau dokumen lain secara langsung. Dokumen ini umumnya berisi informasi atau interface aplikasi di dalam internet. Ada dua cara untuk membuat sebuah web page yaitu dengan HTML editor atau dengan editor teks biasa seperti notepad.

2.5.4. CSS (Cascading Style Sheet)

Menurut (Jayan, 2010), CSS (*Cascading Style Sheet*) adalah suatu teknologi yang digunakan untuk mengatur tampilan dokumen HTML, contohnya seperti pengaturan jarak antar baris, teks, warna dan format border, bahkan penampilan file

gambar. CSS ini terutama digunakan untuk mengatur style elemen yang ada dalam halaman *web*, mulai dari mengatur format text sampai layout. CSS dapat memberikan *style* yang konsisten pada elemen tertentu dan memberi kemudahan dalam pengaturan style dalam halaman *web*.

2.5.5. JavaScript

Menurut Andre Pratama (Pratama, 2020) mengatakan JavaScript adalah bahasa pemrograman web yang bersifat Client Side Programming Language dan dikembangkan oleh Brendan Eich dari Netscape. JavaScript bersifat Client Side Programming Language yaitu tipe bahasa pemrograman yang pemrosesannya dilakukan oleh client. Aplikasi client yang dimaksud merujuk kepada web browser seperti Google Chrome dan Mozilla Firefox

2.5.6. Basis Data

Menurut (Indrajani, 2015), basis data adalah kumpulan data yang saling berhubungan secara logis dan didesain untuk mendapatkan data yang dibutuhkan oleh suatu organisasi.

2.5.7. MySQL

Menurut (Raharjo, 2016) “MySQL merupakan RDBMS (atau server database) yang mengelola database dengan cepat menampung dalam jumlah sangat besar dan dapat di akses oleh banyak user”.

2.6. Peralatan Pendukung Sistem

2.6.1. UML (Unified Modelling Language)

Menurut (A. S. & Shalahuddin, 2018), Unified Modeling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML 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 pendapat yang dikemukakan Rosa dan Shalahuddin tersebut dapat ditarik kesimpulan bahwa UML adalah sebuah bahasa yang berdasarkan gambar untuk memvisualisasikan, menspesifikasikan, dan membangun arsitektur dari sebuah sistem pengembangan perangkat lunak berbasis Objek (Object Oriented Programming)

2.7. Model-Model Diagram UML

1. Use Case Diagram

Menurut (A.S & Shalahuddin, 2016) *Use case diagram* Merupakan pemodelan untuk kelakuan (behavior) 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. Sebuah *use case* mempresentasikan sebuah interaksi antara actor dengan sistem. *Use case* menggambarkan kata kerja seperti *login* ke sistem, *maintenance user* dan sebagainya. Oleh karena itu, *use case diagram* dapat membantu menganalisa kebutuhan suatu sistem.

2. Skenario

Menurut (Verdi, 2012) skenario adalah sebuah dokumentasi terhadap kebutuhan fungsional dari sebuah sistem. *Form* skenario merupakan penjelasan penulisan *use case* dari sudut pandang *actor*.

3. Activity Diagram

Menurut (Sukamto & Shalahudin, 2014) diagram aktivitas atau *activity diagram* adalah menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan oleh actor. Dengan demikian *activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan *actor*, jadi aktivitas yang dapat dilakukan oleh sistem

