

BAB II

LANDASAN TEORI

2.1 Konsep Sistem Informasi

Peranan sistem informasi dalam suatu organisasi tidak diragukan lagi. Dukungannya dapat membuat sebuah perusahaan lain dengan mempergunakan sistem informasi. Keberadaan sistem informasi tidak hanya bermanfaat bagi perusahaan, tetapi juga bagi nasabah atau konsumen. Sebagai contoh, berkat sistem informasi para nasabah bank dengan mudah dapat mengambil uang di mesin-mesin ATM, memperoleh informasi saldo tabungan atau melakukan transfer melalui telepon, dan bahkan melakukan pemesanan barang melalui Internet. Sistem informasi adalah sistem yang mendukung fungsi operasional yang bersifat manajerial dengan kegiatan strategi untuk menyediakan informasi yang diperlukan untuk pengambilan keputusan (Kristy,2021).

2.1.1 Pengertian Sistem

Sistem merupakan kumpulan elemen – elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (output) yang diinginkan. Menurut (Ahmadar dan Taufik, 2021) Sistem dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu.

Maka, Suatu sistem yang baik harus mempunyai tujuan dan sasaran yang tepat karena hal ini akan sangat menentukan dalam mendefinisikan masukan yang dibutuhkan sistem dan juga keluaran yang dihasilkan.

2.1.2 Elemen Sistem

Menurut Kristanto (2018:2), terdapat elemen – elemen dalam sistem yang meliputi :

1. Tujuan Sistem

Tujuan sistem merupakan tujuan dari sistem dibuat. Tujuan sistem dapat berupa tujuan organisasi, kebutuhan organisasi, permasalahan yang ada dalam suatu organisasi maupun urutan prosedur untuk mencapai tujuan yang ingin diraih organisasi tersebut.

2. Batasan Sistem

Batasan sistem merupakan sesuatu yang membatasi sistem dalam mencapai tujuan sistem. Batasan sistem dapat berupa peraturan-peraturan yang ada dalam suatu organisasi, aspek biaya yang dikeluarkan, orang yang ada dalam sebuah organisasi, fasilitas yang baik sebagai sarana dan prasarana maupun batasan yang lain.

3. Kontrol Sistem

Kontrol atau pengawasan sistem merupakan pengawasan terhadap pelaksanaan pencapaian tujuan dari sistem tersebut. Kontrol sistem dapat berupa kontrol terhadap pemasukan data (input), kontrol terhadap

keluaran data (output), kontrol terhadap umpan balik dan lain sebagainya.

4. Masukan (Input)

Input atau masukan merupakan elemen dari sistem yang bertugas untuk menerima seluruh masukan data, dimana masukan tersebut dapat berupa jenis data, frekuensi pemasukan data, dan sebagainya.

5. Proses

Proses merupakan elemen dari sistem yang bertugas untuk mengolah atau memproses seluruh masukan data menjadi suatu informasi yang lebih berguna. Misalkan sistem produksi akan mengolah bahan baku yang berupa mentah menjadi bahan jadi yang siap untuk digunakan.

6. Keluaran (Output)

Output atau keluaran merupakan hasil dari input yang telah diproses oleh bagian pengolah dan merupakan tujuan akhir sistem. Output ini bisa berupa laporan grafik, diagram batang, dan sebagainya.

7. Umpan Balik

Umpan balik merupakan elemen dalam sistem yang bertugas untuk mengevaluasi bagian dari output yang dikeluarkan dimana elemen ini sangat penting demi kemajuan sebuah sistem. Umpan balik dapat diartikan sebagai perbaikan sistem, pemeliharaan sistem.

2.1.3 Pengertian Informasi

Menurut Pratama I Putu Agus Eka (2014 : 9) informasi merupakan hasil pengolahan data dari satu atau berbagai sumber, yang kemudian diolah, sehingga

memberikan nilai, arti, dan manfaat. Proses pengelolaan ini memerlukan teknologi. Berbicara mengenai teknologi memang tidak harus selalu berkaitan dengan komputer, namun komputer sendiri merupakan salah satu bentuk teknologi. Dengan kata lain, alat tulis dan mesin ketik pun dapat dimasukkan sebagai salah satu teknologi yang digunakan selain komputer dan jaringan komputer.

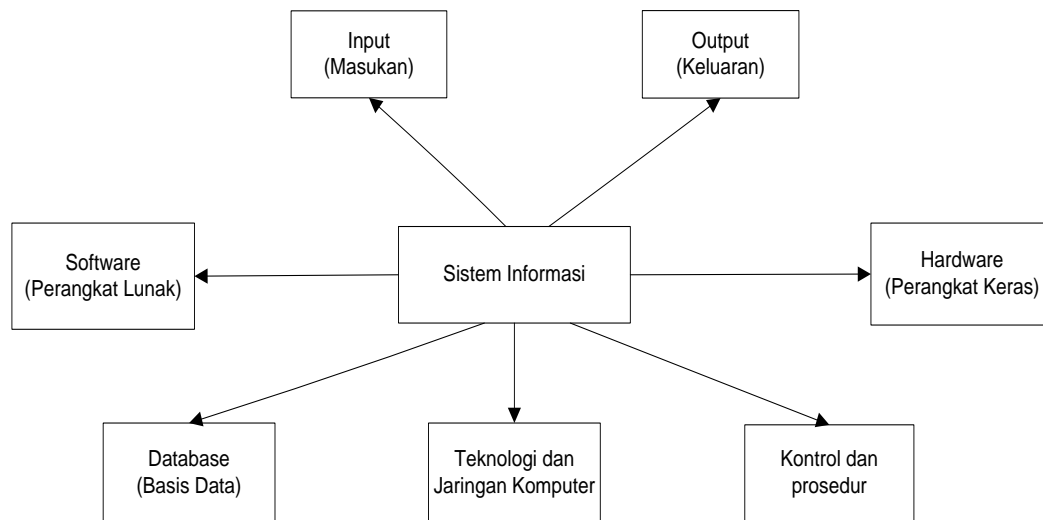
2.1.4 Pengertian Sistem Informasi

Menurut Kristanto (2018:12) menyimpulkan bahwa “Sebuah sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut”.

Sedangkan Menurut Kadir (2018:8) pengertian “Sistem informasi adalah jumlah komponen (manusia, komputer, teknologi informasi dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan”.

Di dalam sistem informasi, data sangat berpengaruh penting karena data akan dimasukkan sebagai formulir-formulir, prosedur-prosedur dan bentuk data lainnya. Maka, sistem informasi dapat diartikan sebagai sekumpulan prosedur pada sebuah organisasi yang akan memberikan informasi bagi pengambil keputusan sebagai pengendali organisasi. Sehingga sebagai sebuah sistem yang mengolah data menjadi informasi yang akan disajikan dan digunakan oleh pengguna, maka sistem informasi merupakan sebuah sistem yang kompleks.

Bukan hanya computer saja yang bekerja, namun juga manusia. Manusia dalam hal ini menggunakan seluruh ide, pemikiran, perhitungan, untuk dituangkan ke dalam sistem informasi yang digunakan. Sistem informasi dikembangkan sesuai dengan kebutuhan pengguna. Ini berarti ada banyak jenis sistem informasi dengan tujuan berbeda. Berikut adalah komponen-komponen pada sistem informasi :



Gambar 2.1 Komponen-komponen di Dalam Sistem Informasi

(Sumber : *Sistem Informasi dan Implementasinya* Pratama I Putu Agus Eka, 2014:14)

1. *Input* (masukan)

Komponen *input* ini berfungsi untuk menerima semua input (masukan) dari pengguna. Inputan yang diterima dalam bentuk data. Data ini berasal dari satu maupun beberapa buah sumber.

2. *Output* (keluaran)

Komponen *output* berfungsi untuk menyajikan hasil akhir ke pengguna sistem informasi. Informasi yang disajikan ini merupakan hasil dari

pengolahan data yang telah diinputkan sebelumnya. Pada komponen *output*, informasi yang disajikan disesuaikan dengan data yang diinputkan dan fungsionalitas dari sistem informasi bersangkutan.

3. *Software* (Perangkat Lunak)

Komponen perangkat lunak ini melakukan proses pengolahan data, penyajian informasi, penghitungan data, dan lain-lain. Komponen perangkat lunak mencakup sistem informasi, aplikasi dan driver.

4. *Hardware* (Perangkat Keras)

Komponen perangkat keras ini meliputi komputer server beserta komponen didalamnya, komputer portable beserta komponen didalamnya, mobile device, dan lain-lain. Termasuk juga didalamnya hub, switch, router, yang berperan didalam jaringan komputer.

5. *Database*

Komponen basis data berfungsi untuk menyimpan semua data dan informasi kedalam satu atau beberapa tabel. Setiap tabel memiliki field masing-masing. Setiap tabel memiliki fungsi penyimpanan masing-masing, serta antar tabel dapat juga terjadi relasi (hubungan).

6. Kontrol dan prosedur

Komponen kontrol berfungsi untuk mencegah terjadinya beragam gangguan dan ancaman terhadap data dan informasi yang ada di dalam sistem informasi, termasuk juga sistem informasi itu sendiri beserta fisiknya. Komponen prosedur mencakup semua prosedur dan aturan

yang harus dilakukan dan wajib ditaati bersama, guna mencapai tujuan yang diinginkan.

7. Teknologi dan Jaringan Komputer

Komponen jaringan komputer berperan didalam menghubungkan sistem informasi dengan sebanyak mungkin pengguna, baik melalui kabel jaringan (*wired*) maupun tanpa kabel (*wireless*).

2.1.5 Siklus Hidup Pengembangan Sistem (SDLC)

SDLC atau *Software Development Life Cycle* atau sering disebut juga *Sistem Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (Rosa A S dan M. Shalahuddin (2014 : 26).

Tahap-tahap yang ada pada SDLC secara global adalah sebagai berikut :

1. Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*sistem concept developmet*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

3. Perencanaan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya.

4. Analisis kebutuhan (*requirements analysis*)

Menganalisis kebutuhan pemakai sistem perangkat lunak (user) dan pengembangan kebutuhan user.

5. Desain (*design*)

Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan.

7. Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional.

8. Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada user) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

Mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas user.

2.2 Alat Bantu Perancangan Sistem

Langkah-langkah yang dilakukan pada tahap merancang suatu sistem informasi dan program adalah membuat usulan pemecahan masalah secara logikal sesuai dengan permasalahan yang ada. Dan alat bantu yang digunakan dalam membantu pemecahan masalah dalam membuat sistem ini antara lain adalah:

1. Diagram Alir Data (DFD):

DFD digunakan untuk menggambarkan alur data dalam sistem dan hubungan antara proses, data, dan entitas eksternal. DFD membantu dalam memahami bagaimana data bergerak di dalam sistem dan memodelkan proses bisnis dengan jelas.

2. Diagram Use Case:

Diagram Use Case digunakan untuk menunjukkan interaksi antara aktor (pengguna) dengan sistem dan menggambarkan fungsionalitas sistem

dari perspektif pengguna. Use Case membantu dalam mengidentifikasi kebutuhan fungsional sistem.

3. Diagram ER (Entity Relationship):

Diagram ER digunakan untuk menggambarkan model hubungan antara entitas dalam basis data. Diagram ini membantu dalam merancang struktur basis data dan mengidentifikasi ketergantungan antar entitas.

4. Mockup dan Prototipe:

Mockup dan prototipe digunakan untuk membuat tampilan visual dan fungsional sistem sebelum implementasi sebenarnya. Ini membantu dalam mengumpulkan umpan balik dari pengguna dan mengklarifikasi kebutuhan mereka.

5. Bahasa pemrograman dan framework:

Bahasa pemrograman dan framework digunakan untuk mengembangkan aplikasi berbasis kode. Pemilihan bahasa pemrograman yang tepat dapat mempengaruhi kinerja dan skalabilitas sistem.

6. Perangkat lunak pengelola basis data:

Perangkat lunak seperti MySQL, PostgreSQL, atau Oracle digunakan untuk mengelola basis data. Pemilihan DBMS yang sesuai mempengaruhi keamanan, kinerja, dan kehandalan sistem.

7. Alat pengujian:

Alat pengujian digunakan untuk melakukan pengujian fungsionalitas dan kinerja sistem sebelum diterapkan secara menyeluruh. Pengujian ini membantu memastikan sistem berjalan dengan baik dan sesuai dengan kebutuhan.

8. Alat pemantauan dan pelaporan:

Alat pemantauan dan pelaporan digunakan untuk memantau kinerja sistem secara real-time dan menghasilkan laporan terkait. Ini membantu dalam mengidentifikasi masalah dan memperbaiki masalah dengan cepat.

9. Perangkat lunak kolaborasi:

Perangkat lunak kolaborasi digunakan untuk memfasilitasi komunikasi dan kolaborasi antara tim pengembang dan pemangku kepentingan. Alat ini memungkinkan tim untuk bekerja bersama secara efisien.

Dengan menggunakan alat-alat bantu perancangan sistem ini, tim pengembang dapat merancang dan mengembangkan sistem informasi secara terstruktur, efisien, dan sesuai dengan kebutuhan pengguna serta tujuan organisasi.

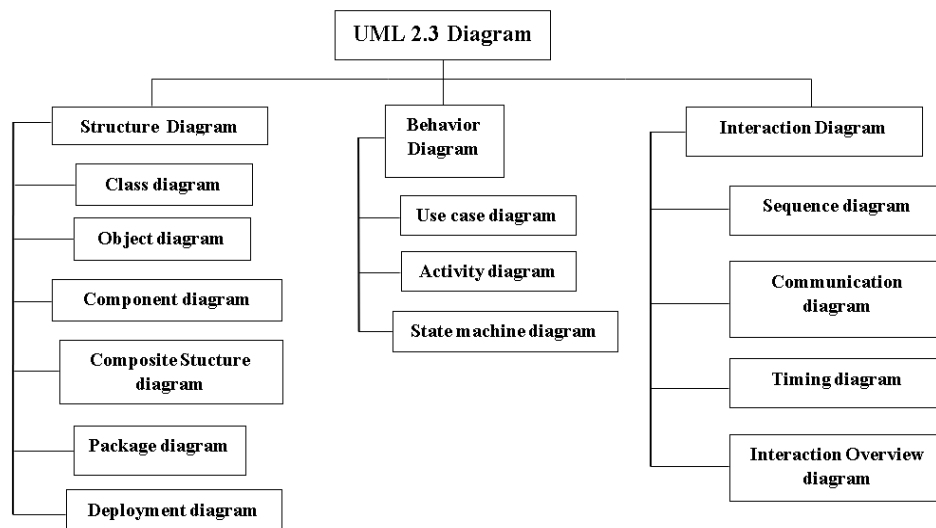
2.2.1 UML (*Unified Modelling Language*)

Menurut Rosa A. S dan M. Shalahuddin (2014 : 133) UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di

dunia industri untuk mendefinisikan *requirement*, membuat analisis dan disain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

2.2.2 Diagram UML

UML (*Unified Modeling Language*) terdiri dari 13 macam diagram yang di kelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini:



Gambar 2.2 : Proses Rekayasa Perangkat lunak

(Sumber: Rosa A. S dan M. Shalahuddin (2014 : 140) “Rekayasa Perangkat Lunak”)

- a. Structure diagram, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Structure diagram terdiri dari class diagram, object diagram, component diagram, composite structure diagram, package diagram dan deployment diagram.

- b. Behavior diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Behavior diagram terdiri dari Use case diagram, Activity diagram, State Machine System.
- c. Interaction diagram yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. Interaction diagram terdiri dari Sequence Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.

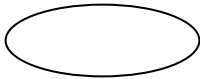



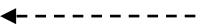
2.2.3 Jenis - Jenis Diagram UML

UML (*Unified Modeling Language*) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. Secara fisik, UML adalah sekumpulan spesifikasi yang dikeluarkan oleh OMG (Rosa A. S dan M. Shalahuddin, 2014).

2.2.4 Use Case Diagram

Menurut Rosa A. S dan M. Shalahuddin (2014 : 155) *Use case diagram* merupakan pemodelan untuk kelakuan (*behavior*). *Use case* mendiskripsikan 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.

Komponen pembentuk atau simbol yang ada pada diagram *Use Case* dapat dilihat pada Tabel 2.1 berikut ini.

No.	Simbol	Nama	Keterangan
1		Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2		Actor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3		Asosiasi	Komunikasi antara aktor dan usecase yang berpartisipasi pada usecase atau usecase memiliki interaksi dengan aktor.
4		<<Extend>>	Relasi usecase tambahan ke sebuah usecase yang ditambahkan dapat berdiri sendiri walau tanpa usecase tambahan itu.
5		<<Include>>	Relasi usecase dimana proses bersangkutan akan dilanjutkan ke proses yang dituju.

Tabel 2.1. Simbol Pada *Use Case Diagram*

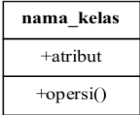




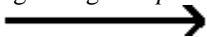

(Sumber: Rosa A. S dan M. Shalahuddin (2014 : 155) “*Rekayasa Perangkat Lunak*”)

2.2.5 Class Diagram

Menurut Rosa A. S dan M. Shalahuddin (2014 : 141), *Class diagram* menggambarkan struktur sistem dari segi pendefinisikan kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. *Class diagram* juga menjelaskan hubungan antar *class* dalam

sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan.

Adapun simbol-simbol yang terdapat dalam *Class diagram* adalah seperti pada Tabel 2.2 berikut ini :

No.	Simbol	Deskripsi
1	<p>Kelas</p> 	Kelas pada struktur sistem.
2	<p>antarmuka/<i>interface</i></p>  <p>nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek.
3	<p>asosiasi/<i>association</i></p> 	Relasi antar kelas dengan makna umum, dan juga disertai dengan <i>multiplicity</i> .
4	<p>asosiasi berarah/ <i>directed association</i></p> 	Relasi antarkelas bermakna umum, asosiasi biasanya disertai dengan <i>multiplicity</i>
5	<p>Generalisasi</p> 	Relaksi antarkelas yang bermakna generalisasi-spesialisasi (umum-khusus)
6	<p>kebergantungan/<i>dependency</i></p> 	Relasi antar kelas yang bermakna kebergantungan antarkelas
7	<p>agregasi/<i>aggregation</i></p> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)






Tabel 2.2. Simbol Pada *Class Diagram*

(Sumber: Rosa A. S dan M. Shalahuddin, 2014)

2.2.6 Activity Diagram

Menurut Rosa A. S dan M. Shalahuddin (2014 :161) activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses kerja 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.

Adapun simbol-simbol yang terdapat dalam *activity diagram* adalah seperti pada Tabel 2.3 berikut ini :

No.	Gambar	Nama	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2.		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
3.		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4.		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan.
5.		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.



Tabel 2.3. Simbol Pada Activity Diagram

(Sumber: Rosa A. S dan M. Shalahuddin, 2014)

2.2.7 Sequence Diagram

Menurut Rosa A. S dan M. Shalahuddin (2014 : 165) *Sequence diagram* menggambarkan kelakuan *usecase* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karna itu untuk menggambarkan *sequence diagram* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diintansiasi menjadi objek itu. Membuat *sequence diagram* juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Simbol-simbol yang terdapat pada *Sequence Diagram* dapat dilihat pada Tabel 2.4 berikut ini :

No.	Gambar	Nama	Keterangan
1.		<i>Object / Participant</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi untuk dapat menyederhanakan diagram.
2.		<i>Message</i>	Message adalah perpindahan control dari satu participant ke participant lainnya.

Tabel 2.4 Simbol Pada *Sequence Diagram*

(Sumber: Rosa A. S dan M. Shalahuddin, 2014)


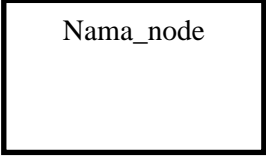


2.2.8 Deployment Diagram

Menurut Rosa A. S dan M. Shalahuddin (2014 : 154) Diagram *deployment* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut :

- a. Sistem tambahan (*embedded sistem*) yang menggambarkan rancangan *device, node, hardware*.
- b. Sistem *client/server*.
- c. Sistem terdistribusi murni.
- d. Rekayasa perangkat lunak.

Simbol-simbol yang ada pada diagram *deployment* dapat dilihat pada table

2.5 sebagai berikut :

No	Gambar	Nama	Keterangan
1.		Package	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih <i>node</i>
2.		<i>node</i>	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika didalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan dengan komponen yang telah diidentifikasi sebelumnya pada diagram komponen.
3.		Ketergantungan/ <i>dependency</i>	Ketergantungan antar <i>node</i> arah panah mengarah pada <i>node</i> yang dipakai
4.		<i>Link</i>	Relasi antar <i>node</i>

Tabel 2.5. Simbol Pada Diagram Deployment

(Sumber: Rosa A. S dan M. Shalahuddin, 2014)

2.2 Database

Pengertian database menurut Bambang Hariyanto (2004) adalah : *"kumpulan data (elementer) yang secara logic berkaitan dalam mempresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi dalam system tertentu"*.

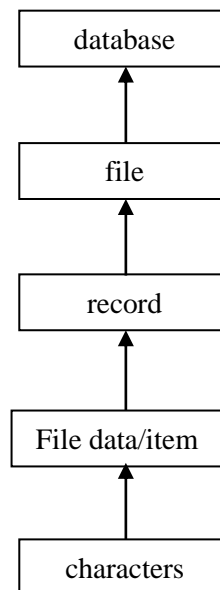
Dari definisi diatas maka dapat disimpulkan bahwa database adalah kumpulan dari item data yang saling berhubungan satu dengan yang lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, yang kelak dapat dimanfaatkan kembali dengan cepat dan mudah. (miranti dan susanti, 2014:105)

2.3.1 Alasan Diperlukan Database

Alasan diperlukannya database dalam sistem informasi dapat dilihat pada poin-poin dibawah ini :

1. Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
2. Menentukan kualiatas informasi : akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
3. Mengurangi duplikasi data (data redudancy)
4. Hubungan data dapat ditingkatkan (data relatability)
5. Mengurangi pemborosan tempat simpanan luar

Untuk lebih jelasnya dapat dilihat pada gambar 2.3 jenjang data dibawah ini :



Gambar 2.3 Jenjang Data

Dimana:

1. Characters : merupakan bagian data yang terkecil, dapat berupa karakter numerik, huruf ataupun karakter-karakter khusus (special characters) yang membentuk suatu item data/field.
2. Field : merepresentasikan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya. Kumpulan dari field membentuk suatu record
 - a. field name : harus diberi nama untuk membedakan field yang satu dengan lainnya
 - b. field representation : tipe field (karakter, teks, tanggal, angka, dsb), lebar field (ruang maksimum yang dapat diisi dengan karakter-karakter data).
 - c. field value : isi dari field untuk masing-masing record.

3. Record : Kumpulan dari field membentuk suatu record. Record menggambarkan suatu unit data individu yang tertentu. Kumpulan dari record membentuk suatu file. Misalnya file personalia, tiap-tiap record dapat mewakili data tiap-tiap karyawan.
4. File : File terdiri dari record-record yang menggambarkan satu kesatuan data yang sejenis. Misalnya file mata pelajaran berisi data tentang semua mata pelajaran yang ada.
5. Database : Kumpulan dari file / tabel membentuk suatu database.(miranti dan susanti, 2018:105-106)

2.4 E-commerce

E-Commerce adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa yang dilakukan oleh konsumen melalui sistem elektronik seperti internet , *www (World Wide Web)*, atau jaringan komputer lainnya. *E-commerce* dapat melibatkan transfer dana elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan data otomatis (Himawan R.P et al, 2020).

2.4.1 Komponen E-Commerce

Pada *e-commerce* terdapat beberapa komponen yang terkait, dimana komponen-komponen ini membentuk sebuah mekanisme pasar *e-commerce*, yaitu (Himawan R.P et al, 2020):

1. Konsumen

Konsumen disini berbicara mengenai para pengguna internet yang dijadikan sebagai target pasar yang potensial untuk diberikan berbagai macam penawaran baik berupa produk, jasa maupun informasi oleh penjual.

2. Penjual

Penjual merupakan pihak yang menawarkan produk, jasa atau informasi kepada para konsumen baik secara individu maupun organisasi. Proses penjualan dapat dilakukan dengan menggunakan media *website* yang dimiliki oleh penjual.

3. Produk

Pada transaksi *e-commerce*, sesungguhnya produk yang ditawarkan adalah produk digital. Hal ini dikarenakan konsumen tidak melihat secara langsung produk yang ditawarkan dalam bentuk fisik melainkan hanya merupakan gambar visualisasi dalam bentuk katalog produk dalam halaman *website*.

4. *Front end*

Front end merupakan aplikasi web yang dapat berinteraksi dengan para pengguna secara langsung. Beberapa proses bisnis yang terdapat pada *front end* ini antara lain adalah katalog, keranjang belanja (*shopping cart*), mesin pencari (*search engine*), dll.

5. Infrastruktur

Infrastruktur pasar yang menggunakan media elektronik meliputi penggunaan perangkat keras, perangkat *lunak* dan juga sistem jaringan komputer seperti penggunaan jaringan komunikasi internet.

6. Back end

Back end merupakan bentuk aplikasi yang secara tidak langsung berperan sebagai pendukung dari aplikasi *front end*. Dimana semua aktifitas yang berkaitan dengan pemesanan barang, manajemen pengelolaan produk, proses pembayaran dan pengiriman barang termasuk dalam proses bisnis *back end*.

7. Partner Bisnis

Partner bisnis merupakan pihak yang dapat melakukan kolaborasi atau kerjasama dengan para produsen. Contoh partner bisnis adalah seperti bank yang dapat memudahkan proses pembayaran yang dilakukan oleh para konsumen baik via transfer ataupun *mobile banking (m-banking)*.

8. Support services

Support services dalam konteks e-commerce juga mencakup layanan pelanggan yang responsif dan efisien untuk membantu mengatasi masalah atau pertanyaan konsumen. Selain itu, layanan pengiriman dan logistik yang handal juga merupakan bagian penting dari support services agar produk dapat sampai ke tangan konsumen dengan tepat waktu dan dalam kondisi baik.

2.5 Dasar Jaringan

2.5.1 Website

Website merupakan kumpulan halaman web yang saling terhubung dan file-filenya saling terkait. Web terdiri dari page atau halaman, dan kumpulan halaman yang dinamakan homepage. Homepage berada pada posisi teratas, dengan halaman –halaman terkait berada di bawahnya. Biasanya setiap halaman di bawah homepage disebut child page, yang berisi hyperlink ke halaman lain dalam web.

Website awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hyperlink, yang memudahkan surfer atau pengguna internet melakukan penelusuran informasi di internet. Informasi yang disajikan dengan web menggunakan konsep multimedia, informasi dapat disajikan dengan menggunakan banyak media, seperti teks, gambar, animasi, suara, atau film (Aryato Didik, at al, 2020).

2.5.2 Internet

Internet adalah sebagai jaringan komputer yang sangat luas dan besar dan mendunia, menghubungkan pemakai komputer dari satu negara ke negara lain di seluruh dunia, dimana di dalamnya terdapat berbagai sumber informasi dan fasilitas– fasilitas layanan internet yaitu diantaranya (Hastanti R.P et al, 2019):

1. *Browsing atau surfing* yaitu kegiatan “berselancar” di internet, kegiatan ini seperti layaknya berjalan-jalan di *mal* sambil melihat-lihat ke toko-toko tanpa membeli apapun.

2. *Elektronik mail (E-mail)*, fasilitas ini digunakan untuk berkirim surat dengan orang lain, tanpa mengenal batas, waktu, ruang bahkan birokrasi. *Searching* yaitu kegiatan mencari data atau informasi tertentu di internet.
3. *Chatting*, fasilitas ini digunakan untuk berkomunikasi secara langsung dengan orang lain di Internet. Pada umumnya fasilitas ini sering digunakan untuk bercakap-cakap atau mengobrol di internet *World Wide Web (WWW)*.
4. *Newsgroup*, fasilitas ini digunakan untuk berkoferensi jarak jauh, sehingga dapat menyampaikan pendapat dan tanggapan dalam internet.
5. *Download*, adalah proses mengambil file dari komputer lain melalui internet ke komputer kita. Selain yang diatas masih banyak lagi fasilitas-fasilitas yang terdapat di internet.

2.6 PHP

Php pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama form interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari *web*. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis 2.0 ini, interpreter PHP sudah di implemetasikan dalam program C. Didalam rilis ini juga ikut disertakan modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan. Kemudian pada tahun 1997 sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP 3.0.

Pada Juni 2004 Zend merilis PHP 5.0 dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman kearah paradigma berorientasi objek. (Priyanto Hidayatullah dan Jauhari Khairul Kawistara, 2014 : 232).

2.6.1 Kelebihan PHP (Personal Home Page)

PHP berbasis *server side scripting*. PHP sendiri dapat melakukan tugas-tugas yang dilakukan dengan mekanisme CGI seperti mengambil, mengumpulkan data dari *database*, *meng-generate* halaman dinamis, atau bahkan menerima dan mengirim *cookie*. Dan yang menjadi keutamaan PHP itu sendiri adalah PHP bias digunakan di berbagai *operating sistem*, diantaranya *Linux*, *Unix*, *Windows*, dan lain-lain. (Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2014 : 232).

2.7 MySQL

MySQL adalah salah satu perangkat lunak Database Management System (DBMS) yang bersifat open-source, berarti dapat digunakan secara gratis dan dikembangkan oleh Oracle Corporation. MySQL menggunakan bahasa query SQL (Structured Query Language) untuk mengelola dan mengakses basis data. DBMS ini digunakan untuk membuat, mengelola, dan memanipulasi database relasional, yang merupakan kumpulan tabel yang berisi data terstruktur.

MySQL sangat populer dan sering digunakan di berbagai aplikasi, situs web, dan sistem informasi karena kinerja tinggi, kehandalan, dan fleksibilitasnya dalam mengelola data.

2.7.1 Sekilas Mengenai MySQL

MySQL adalah Relational Database Management Sistem (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namaun tidak boleh dijadikan produk turunan yang bersifat close source atau komersial (*Sophan Sophian, 2014*).

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama yaitu, SQL(*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan/seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis (*Sophan Sophian, 2014*).

2.7.2 Tipe Data dalam MySQL

2.7.2.1 Tipe Data Numerik

MySQL menggunakan seluruh tipe data numerik standar ANSI. Berikut ini adalah tipe data numerik yang biasanya digunakan beserta penjelasannya.

No.	Tipe Data	Deskripsi
1.	INT	Nilai integer yang bisa bertanda atau tidak. Jika bertanda, maka rentang yang diperbolehkan adalah -2147483548 sampai 2147483647, sedangkan jika tidak bertanda maka rentangnya dari 0 sampai 4294967295.
2.	TINYINT	Nilai integer yang sangat kecil. Rentangnya -128 sampai 127 untuk yang bertanda dan 0 – 255 untuk yang tidak bertanda.
3.	SMALLINT	Nilai integer yang sangat kecil dengan rentang -31768 sampai 32767 untuk yang bertanda sedangkan untuk yang tidak bertanda dari 0 – 65535.

4.	MEDIUMINT	Integer dengan ukuran sedang dengan rentang -8388608 sampai 8388607 atau 0 sampai 16777215
5.	BIGINT	Integer dengan ukuran besar dengan rentang -9223372036854775808 sampai 9223372036854775807 atau 0 sampai 18446744073709551615.
6.	FLOAT	Bilangan pecahan dengan panjang M dan jumlah desimal D. Presisi desimalnya bisa sampai 24 digit. Defaultnya Float (10,2). Bilangan float selalu bisa bertanda.
7.	DOUBLE	Adalah Bilangan pecahan dengan panjang M dan jumlah desimal D. Presisi desimalnya bisa sampai 53 digit. Defaultnya Double (16,4). Bilangan float selalu bisa bertanda.
8.	DECIMAL	Adalah bilangan pecahan dan harus didefinisikan M dan D-nya. Setiap desimal membutuhkan tempat 1 byte. Sinonim dari DECIMAL adalah NUMERIC.

Tabel 2.7 Tipe Data Numerik

(Sumber : Priyanto Hidayatullah dan Jauhari Khairul Kawistara, 2014)

2.7.2.2 Tipe Data Tanggal dan Waktu

Berikut ini adalah data tanggal dan waktu yang paling umum digunakan di dalam MySQL :

No.	Tipe Data	Deskripsi
1.	DATE	Adalah tipe data tanggal dengan format YYYY-MM-DD antara 1000-01-01 and 9999-12-31.
2.	DATETIME	Adalah kombinasi tanggal dan waktu dengan format YYYY-MM-DD HH:MM:SS dan rentang data antara 1000-01-01 00:00:00 sampai dengan 9999-12-31 23:59:59.
3.	TIMESTAMP	Sebuah penanda waktu antara 1 Januari 1970 tengah malam sampai dengan tahun 2037. Formatnya mirip dengan DATETIME tetapi tanpa pembatas di antara angkanya.
4.	TIME	Menyimpan waktu dalam format HH:MM:SS.

5.	YEAR	Menyimpan data tahun dalam format 2 atau 4 digit. Jika M diisi dengan nilai 2, maka rentang tahunnya dari 1970-2069 sedangkan jika M diisi dengan nilai 4 maka YEAR bisa bernilai 1901 sampai dengan 2155. Defaultnya nilai M adalah 4.
----	------	---

Tabel 2.8 Tipe Data Tanggal dan Waktu

(Sumber : Priyanto Hidayatullah dan Jauhari Khairul Kawistara, 2014)

2.7.2.3 Tipe Data String

Berikut ini adalah data *string* yang paling umum digunakan di dalam MySQL :

No.	Tipe Data	Deskripsi
1.	CHAR	String dengan ukuran tetap. Ukurannya antara 1 sampai 255 karakter.
2.	VARCHAR	String dengan ukuran bervariasi antara 1 sampai dengan 255 karakter.
3.	TEXT	String dengan ukuran maksimum 65535 karakter. String yang tersimpan di dalam TEXT dianggap tidak case sensitive. Untuk kapasitas yang lebih kecil bisa menggunakan TINYTEXT dengan kapasitas maksimal 255 karakter.
4.	BLOB	<i>Binary Large Objects</i> (BLOB) adalah tipe data untuk menyimpan data binary dalam jumlah besar. Untuk menyimpan data yang lebih kecil bisa menggunakan TINYBLOB sedangkan untuk kapasitas yang lebih besar bisa menggunakan MEDIUMBLOB dan LONGBLOB
5.	ENUM	<i>Enumerasi</i> atau sebuah <i>list</i> (daftar). Misalnya nilai terbatas hanya boleh dengan nilai tertentu saja maka bisa membuat daftar. Misalnya saja nilai itu hanya bisa terdiri dari A-E, maka bisa membuatnya ENUM.

Tabel 2.9 Tipe Data String

(Sumber : Priyanto Hidayatullah dan Jauhari Khairul Kawistara, 2014)