



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Sistem

Hutahaean (2014:2), Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu.

Sutabri (2012:6), Sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.

Menurut Davis dalam Sunyoto (2014:32), Sistem terdiri dari bagian-bagian yang bersama-sama beroperasi untuk mencapai beberapa tujuan, dengan kata lain bahwa suatu sistem bukanlah merupakan suatu perangkat unsur-unsur yang dapat diidentifikasi sebagai kebersamaan yang mneyatu disebabkan tujuan atau sasaran yang sama.

2.1.2 Karakteristik Sistem

Sutabri (2012:13), Adapun karakteristik atau sifat-sifat tertentu dari sistem adalah sebagai berikut :

1. Komponen sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat-sifat sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan Supra sistem.

2. Batasan sistem

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan



sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.

3. Lingkungan luar sistem

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut dengan lingkungan luar sistem. Lingkungan luar sistem ini dapat menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan. Kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung sistem

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem yang lain. Keluaran suatu subsistem akan menjadi masukan untuk subsistem lain dengan melewati penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukkan sistem

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) sinyal (*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan diolah menjadi informasi.

6. Keluaran sistem

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, dimana informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainnya.



7. Pengolah sistem

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran sistem

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.3 Klasifikasi Sistem

Sutabri (2012:15), Sistem dapat diklasifikasikan dari beberapa sudut pandangan, yaitu :

a. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu suatu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan; sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia, dan lain sebagainya.

b. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang isebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.



c. **Sistem Deterministik dan Sistem Probabilistik**

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

d. **Sistem Terbuka dan Sistem Tertutup**

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.1.4 Pengertian Informasi

Hutahaean (2014:9), Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya.

Sutabri (2012:22), Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Sunyoto (2014:40), Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang.

2.1.5 Pengertian Sistem Informasi

Hutahaean (2014:13), Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.

Sunyoto (2014:47), Sistem informasi berisi informasi tentang orang-orang, tempat, dan hal-hal penting di dalam organisasi atau di lingkungan sekelilingnya.



Informasi sendiri berarti data yang telah dibentuk menjadi sesuatu yang memiliki arti dan berguna bagi manusia.

Sutabri (2012:38), Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu.

2.1.6 Komponen Sistem Informasi

Sutabri (2012:39), Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan, yaitu :

1. Blok masukan

Input mewakili data yang masuk ke dalam sistem informasi. Yang dimaksud dengan input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok teknologi

Teknologi merupakan *tool box* dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama yaitu teknisi, perangkat lunak, dan perangkat keras.



5. Blok basis data

Basis data merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan perangkat lunak digunakan untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*database management system*).

6. Blok kendali

Banyak hal dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan pada sistem itu sendiri, ketidakefisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah dan bila terlanjur terjadi maka kesalahan-kesalahan dapat dengan cepat diatasi.

2.1.7 Pengertian Rumah Sakit

Laksito (2014:51), Rumah sakit adalah sebuah institusi perawatan kesehatan profesional yang pelayanannya dilakukan oleh dokter, perawat, dan tenaga ahli kesehatan lainnya.

Rumah Sakit memberikan dua jenis dasar pelayanan kepada penderita yang dirawat di rumah sakit, yaitu :

1. Pelayanan yang diberikan kepada penderita sakit yang secara fisik tinggal di ruang perawatan rumah sakit, disebut pelayanan penderita rawat tinggal.
2. Pelayanan yang diberikan pada penderita sakit yang datang ke rumah sakit, yang tidak memerlukan tinggal di ruang perawatan rumah sakit disebut pelayanan penderita rawat jalan. Penderita rawat jalan termasuk penderita ambulatori, yaitu penderita yang tidak memerlukan tempat tidur di rumah sakit. Singkatnya, penderita ambulatori adalah penderita yang



noninstitusional, yang bertanggung jawab untuk memperoleh, menyimpan, dan mengonsumsi sendiri obatnya. Ia adalah atau tidak merupakan penderita rawat jalan, tergantung tempat ia menerima pengobatannya.

Pelayanan rawat jalan dan rawat inap adalah salah satu bentuk dari pelayanan kedokteran. Secara sederhana, yang dimaksud rawat jalan adalah pelayanan kedokteran yang disediakan untuk pasien, bukan dalam bentuk rawat inap (hospitalisasi). Dibandingkan dengan pelayanan rawat inap, pelayanan rawat jalan ini memang tampak berkembang lebih pesat. Terdapat peningkatan angka multilasi pelayanan rawat jalan di rumah sakit, dua sampai tiga kali lebih tinggi dari peningkatan angka pelayanan rawat inap.

Secara umum, rawat jalan dan rawat inap adalah bagian dari pelayanan rumah sakit. Akan tetapi, dalam proses untuk mendapatkan informasi, pencatatan/pendokumentasian memiliki perbedaan isi. Pada rawat jalan pencatatan yang diperlukan hanya sebatas catatan atau dokumen tentang identitas pasien, pemeriksaan fisik, diagnosis/masalah, tindakan/pengobatan, dan pelayanan lain yang telah diberikan kepada pasien. Sedangkan dalam rawat inap catatan yang penting dalam dokumen medis adalah minimal memuat identitas pasien, pemeriksaan, diagnosis/masalah persetujuan tindakan medis (bila ada), tindakan/pengobatan dan pelayanan lain yang telah diberikan kepada pasien.

2.1.8 Pengertian Rumah Sakit Ibu dan Anak

Laksito (2014:51), Rumah Sakit Ibu dan Anak adalah rumah sakit untuk melayani dan merawat kesehatan ibu yang sakit kandungan dan kehamilan yang dilengkapi dengan fasilitas untuk melahirkan, pemeriksaan kehamilan, pemeriksaan ibu dan anak, serta berada di bawah pengawasan dokter.

Pelayanan pada Rumah Sakit Ibu dan Anak yang diberikan kepada pasien antara lain :

a. Preventif

Merupakan pelayanan untuk mencegah pasien terjangkit dari penyakit, hal ini dapat dilakukan dengan cara pemeriksaan rutin terhadap perkembangan bayi



dan ibu hamil, konsultasi kesehatan, penyuluhan tentang gizi ibu dan anak serta imunisasi dan KB

b. Kuratif

Merupakan usaha penyembuhan pada pasien dengan cara pengobatan dan perawatan berupa persalinan, pembedahan dan pengobatan.

c. Rehabilitasi

Merupakan tindakan penyembuhan kondisi fisik pasien setelah melampaui masa pengobatan berupa perawatan atau pemulihan kesehatan dan perawatan bayi.

2.1.9 Pengertian Sistem Informasi Rumah Sakit Ibu dan Anak Widiyanti

Sistem informasi Rumah Sakit Ibu dan Anak Widiyanti adalah kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang dapat mempermudah masyarakat dalam melakukan pendaftaran pelayanan rawat jalan, rawat inap dan beberapa pelayanan lainnya serta dapat memberikan informasi mengenai pelayanan apa saja yang ada di Rumah Sakit Ibu dan Anak Widiyanti.

2.2 Teori Khusus

2.2.1 Metodologi RUP (*Rational Unified Process*)

Menurut Rosa dan Shalahudin (2011:30), *Rational Unified Process (RUP)* merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri khas metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perangkat lunak.

RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan *Unified Model Language (UML)*. Melalui gambar dibawah dapat dilihat bahwa RUP memiliki, yaitu:

a. Dimensi Pertama

Digambarkan secara horizontal. Dimensi ini mewakili aspek-aspek dinamis dari pengembangan perangkat lunak. Aspek ini dijabarkan dalam tahapan

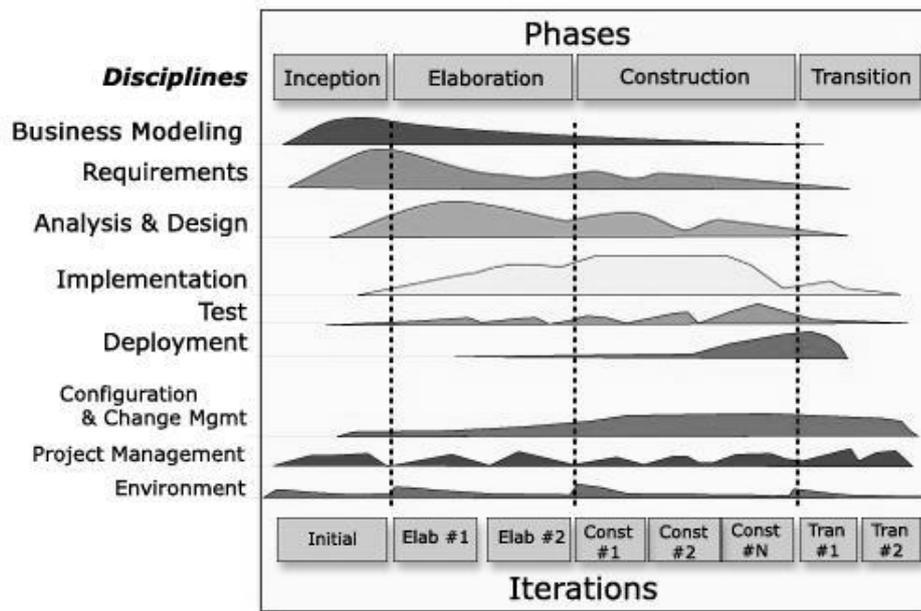


pengembangan atau fase. Setiap fase akan memiliki suatu *major milestone* yang menandakan akhir dari awal dari *phase* selanjutnya. Setiap *phase* dapat berdiri dari satu beberapa iterasi. Dimensi ini terdiri atas *Inception*, *Elaboration*, *Construction*, dan *Transition*.

b. Dimensi Kedua

Digambarkan secara vertikal. Dimensi ini mewakili aspek-aspek statis dari proses pengembangan perangkat lunak yang dikelompokkan ke dalam beberapa disiplin. Proses pengembangan perangkat lunak yang dijelaskan kedalam beberapa disiplin terdiri dari empat elemen penting, yakni *who is doing*, *what*, *how*, dan *when*. Dimensi ini terdiri atas:

Business Modeling, *Requirement, Analysis and Design*, *Implementation*, *Test*, *Deployment*, *Configuration*, dan *Change Management*, *Project Management*, *Environment*.



Gambar 2.1 Arsitektur Rational Unified Process

2.2.1.1 Penerapan Tahap Metodologi Pengembangan Perangkat Lunak dengan RUP

Menurut Rosa dan Shalahudin (2011:32) dalam *Rational Unified Process* terdapat empat tahap pengembangan perangkat lunak yaitu:



a. *Inception*

Pada tahap ini pengembangan mendefinisikan batasan kegiatan, melakukan analisis kebutuhan *user*, dan melakukan perancangan awal perangkat lunak (perancangan arsitektural dan *use case*). Pada akhir fase ini prototipe perangkat lunak versi *Alpha* harus sudah dirilis.

b. *Elaboration*

Pada tahap ini dilakukan perancangan perangkat lunak mulai dari menspesifikasikan fitur perangkat lunak hingga perilsan prototipe versi *Betha* dari perangkat lunak.

c. *Construction*

Pengimplementasian rancangan perangkat lunak yang telah dibuat dilakukan pada tahap ini. Pada akhir tahap ini, perangkat lunak versi akhir yang sudah disetujui administrator dirilis beserta dokumentasi perangkat lunak.

d. *Transition*

Instanlasi, deployment dan sosialisasi perangkat lunak dilakukan di tahap ini.

2.2.1.2 Aliran Kerja Utama RUP

Menurut Rosa dan Shalahudin (2011:32), adapun aliran kerja utama pada Metodologi RUP adalah sebagai berikut:

a. *Pemodelan Bisnis*

Mendeskripsikan struktur dan proses-proses bisnis organisasi.

b. *Kebutuhan*

Mendefinisikan kebutuhan perangkat lunak menggunakan metode *use case*.

c. *Analisis dan Perancangan*

Mendeskripsikan berbagai arsitektur perangkat lunak dari berbagai sudut pandang.

d. *Implementasi*

Menuliskan kode-kode program, menguji, dan mengintegrasikan unit-unit programnya.

e. *Pengujian*

Mendeskripsikan kasus uji, prosedur, dan alat ukur pengujian.



f. Deployment

Menangani konfigurasi sistem yang akan diserahkan.

2.2.1.3 Aliran Kerja Pendukung RUP

Menurut Rosa dan Shalahudin (2011:31), adapun aliran kerja pendukung RUP adalah sebagai berikut:

- a. Manajemen konfigurasi dan perubahan
Mengendalikan perubahan dan memelihara artifak-artifak proyek.
- b. Manajemen proyek
Mendeskripsikan berbagai strategi pekerjaan dengan proses yang berulang.
- c. Lingkungan
Menangani infrastruktur yang dibutuhkan untuk mengembangkan sistem.

2.2.2 Metode Pengujian Perangkat Lunak

2.2.2.1 Pengertian Metode Pengujian

Shihab (2011), Metode pengujian adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

Pengujian perangkat lunak perlu dilakukan untuk mengevaluasi baik secara manual maupun otomatis untuk menguji apakah perangkat lunak sudah memenuhi persyaratan atau belum, dan untuk menentukan perbedaan antara hasil yang diharapkan dengan hasil sebenarnya.

2.2.2.2 Metode Pengujian

Perangkat lunak dapat diuji dengan dua cara, yaitu:

1. Pengujian dengan menggunakan data uji untuk menguji semua elemen program (data internal, loop, keputusan dan jalur). Data uji dibangkitkan dengan mengetahui struktur internal (kode sumber) dari perangkat lunak.
2. Pengujian dilakukan dengan mengeksekusi data uji dan mengecek apakah fungsional perangkat lunak bekerja dengan baik. Data uji dibangkitkan dari spesifikasi perangkat lunak.



2.2.2.3 Metode *Black Box Testing*

Shihab (2011), Pengujian *blackbox (blackbox testing)* adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum). Tahap pengujian merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak.

Black Box Testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program.

Ciri-ciri *black box testing*, yaitu:

1. *Black box testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.
2. *Black box testing* bukan teknik alternatif daripada *white box testing*. Lebih dari pada itu, ia merupakan pendekatan pelengkap dalam mencakup error dengan kelas yang berbeda dari metode *white box testing*.
3. *Black box testing* melakukan pengujian tanpa pengetahuan detail struktur internal dari sistem atau komponen yang dites. juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*.

2.2.3 Pengertian Basis Data

Badiyanto (2013:57), *Database* bisa dikatakan sebagai suatu kumpulan dari data yang tersimpan dalam tabel dan diatur atau diorganisasikan sehingga data tersebut bisa diambil atau dicari dengan mudah dan efisien.

2.2.4 UML (*Unified Modelling Language*)

Rosa dan Shalahuddin (2013:133), *UML (Unified Modeling Language)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.



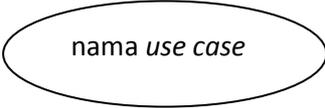
2.2.5 Diagram Use Case

Rosa dan Shalahuddin (2013:155), *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.

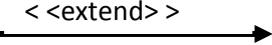
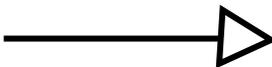
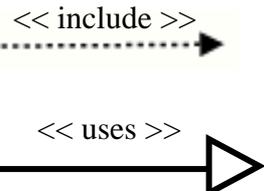
Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

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.

Tabel 2.1 Simbol diagram *use case*

No.	Simbol	Keterangan
1.	<p><i>Use case</i></p>  <p>nama use case</p>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.	<p>Aktor / actor</p>  <p>nama aktor nama_interface</p>	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; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.

Lanjutan Tabel 2.1 Simbol diagram *use case*

No.	Simbol	Keterangan
3.	Asosiasi / <i>association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	Ekstensi / <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i> -nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.
5.	Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya: Arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)
6.	Menggunakan / <i>include</i> / <i>uses</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .

(Sumber: Rosa dan Shalahuddin (2013:156))



2.2.6 Diagram Kelas

Rosa dan Shalahuddin (2013:141), Diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

2. Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.

3. Kelas yang diambil dari pendefinisian *use case* (*controller*)

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

4. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

Tabel 2.2 Simbol diagram kelas

No.	Simbol	Keterangan
1.	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: auto;"> <p>nama_kelas</p> <hr/> <p>+ atribut</p> <hr/> <p>+ operasi()</p> </div>	Kelas pada struktur sistem.



Lanjutan Tabel 2.2 Simbol diagram kelas

No.	Simbol	Keterangan
2.	Antarmuka / <i>interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	Asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6.	Kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.	Agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>).

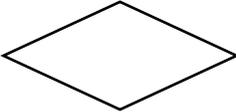
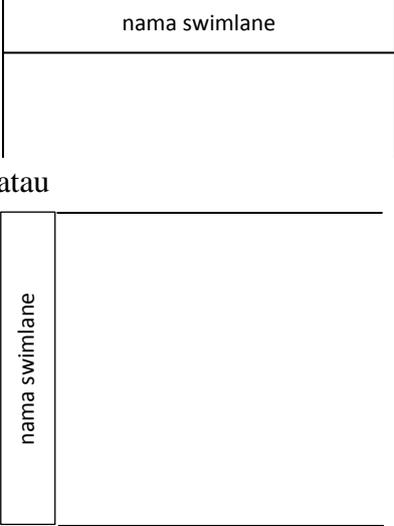
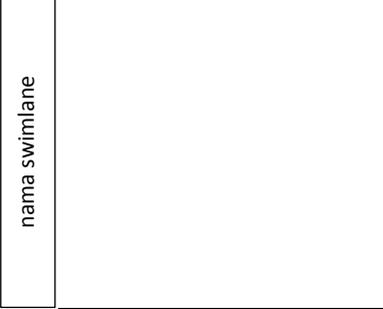
(Sumber: Rosa dan Shalahuddin (2013:146))

2.2.7 Diagram Aktivitas

Rosa dan Shalahuddin (2013:161), Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Berikut adalah simbol-simbol yang ada pada diagram aktivitas:



Tabel 2.3 Simbol-simbol diagram aktivitas

No.	Simbol	Keterangan
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.	Swimlane  atau 	Swimlane memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.

(Sumber: Rosa dan Shalahuddin (2013:162))

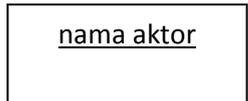
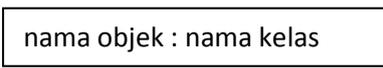


2.2.8 Diagram Sekuensial

Rosa dan Shalahuddin (2013:165), *Sequence diagram* atau diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirim dan diterima antar objek.

Banyaknya diagram sekuensial yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuensial sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuensial yang harus dibuat juga semakin banyak. Berikut adalah simbol-simbol yang ada pada diagram sekuensial:

Tabel 2.4 Simbol-simbol diagram sekuensial

No.	Simbol	Keterangan
1.	<p>Aktor</p>  <p>nama aktor</p> <p>atau</p>  <p>nama aktor</p> <p>tanpa waktu aktif</p>	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; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2.	<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.
3.	<p>Objek</p>  <p>nama objek : nama kelas</p>	Menyatakan objek yang berinteraksi pesan.
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.

(Sumber: Rosa dan Shalahuddin (2013:165))

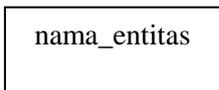
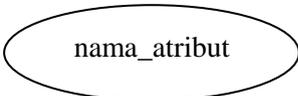
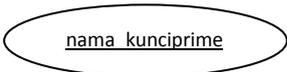
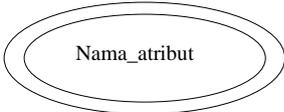


2.2.9 Pengertian *ERD* (*Entity Relationship Diagram*)

Rosa dan Shalahuddin, (2013:50), *ERD* adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. *ERD* digunakan untuk pemodelan basis data relasional.

Rosa dan Shalahuddin, (2013:50), menjelaskan simbol-simbol yang digunakan dalam *ERD*, yaitu:

Tabel 2.5 Simbol-simbol *Entity Relationship Diagram* (*ERD*)

No	Simbol	Keterangan
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya adar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4.	Atribut multivalai / <i>multivalve</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.



Lanjutan Tabel 2.5 Simbol-simbol *Entity Relationship Diagram (ERD)*

No	Simbol	Keterangan
4.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
5.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

(Sumber : Rosa dan Shalahuddin (2008 :75))

2.2.10 Kamus Data

Rosa dan Shalahuddin (2013:73), Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan dan keluaran dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data biasanya berisi :

- a) Nama-nama dari data
- b) Digunakan pada – merupakan proses-proses yang terkait data
- c) Deskripsi – merupakan deskripsi data
- d) Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Rosa dan Shalahuddin, (2013:73), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

**Table 2.6** Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	Dan
3	[]	baik ...atau...
4	{ } ⁿ	n kali diulang/ bernilai banyak
5	()	data operasional
6	*...*	batas komentar

(Sumber : Rosa dan Shalahudin (2013:73))

2.3 Teori Program

2.3.1 Pengenalan PHP

Edy dkk. (2014:49), *PHP* awalnya dikembangkan oleh seorang programmer bernama Rasmus Lerdorf pada tahun 1995, namun semenjak itu slelau dikembangkan oleh kelompok independen yang disebut *Group PHP* dan kelompok ini juga yang mendefinisikan standar de facto untuk *PHP* karena tidak ada spesifikasi formal. Saat ini pengembangannya dipimpin oleh Andi Gutmans dan Zeev Suraski.

Yang menyebabkan *PHP* banyak dipakai oleh banyak orang adalah karena *PHP* adalah perangkat lunak bebas (*open source*) yang dirilis di bawah lisensi *PHP*. Artinya untuk menggunakan bahasa pemrograman ini gratis, bebas dan tidak terbuka.

Untuk web, *PHP* adalah bahasa scripting yang bisa dipakai untuk tujuan apaun. Di antaranya cocok untuk pengembangan aplikasi web berbasis server (*server-side*) mana *PHP* nantinya dijalankan di server web.

Setiap kode *PHP* akan dieksekusi oleh runtime *PHP*, hasilnya adalah kode *PHP* yang dinamis tergantung kepada script *PHP* yang dituliskan. *PHP* dapat digunakan dibanyak server web, sistem operasi dan platform.

Selain itu digunakan juga di sistem manajemen database relasional (RDBMS). Semuanya ini bisa diperoleh gratis, dan *Group PHP* menyediakan kode sumber lengkap bagi pengguna untuk membangun, menyesuaikan dan mengutak-atik sesuai fungsi yang mereka inginkan.



2.3.1.1 Pengertian *PHP*

Badiyanto (2013:32), *PHP* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam *HTML/PHP* banyak dipakai untuk membuat situs *web* dinamis.

2.3.1.2 Tipe Data *PHP*

Edy dkk. (2014:65), menjelaskan variabel dasar yang dapat diakomodasi di *PHP*, yaitu :

1. Boolean

Boolean adalah tipe data paling standar, ini hanya menyatakan kebenaran, apakah *TRUE* (benar) atau *FALSE* (salah). Untuk menyatakan sebuah boolean, anda bisa menggunakan kata kunci *TRUE* atau *FALSE*.

2. Integer

Sebuah integer adalah bilangan bulat (bukan pecahan), baik negatif atau positif. Misalnya { ..., -2, -1, 0, 1, 2, ... }.

Integer bisa dituliskan dalam satuan desimal (berbasis 10), heksadesimal (berbasis 16), atau oktal (berbasis 8) dan bisa juga ditambahi tanda plus atau minus (- atau +) tapi penandaan ini opsional saja.

3. Floating Point

Floating point atau nomor pecahan atau juga bilangan real, bisa didefinisikan dengan syntax berikut ini :

```
<?php
$a = 3.652;
$b = 3.2.e3;
$c = 7E - 10;
?>
```

4. String

String adalah untaian karakter, sehingga karakter sama seperti byte. Ada 256 karakter yang bisa dijadikan string. Sebuah string bisa didefinisikan dengan berbagai cara, tapi yang paling mudah adalah dengan satu petik.



5. Null

Nilai NULL mempresentasikan variabel tanpa value. NULL adalah nilai yang mungkin untuk NULL. Sebuah variabel akan menjadi null jika :

- a. Di alokasikan konstanta NULL
- b. Variabel yang belum diset ke nilai apapun.
- c. Menerima unset().

2.3.1.3 Skrip PHP

Badiyanto (2013:32), PHP merupakan bahasa scripting yang terpasang pada HTML, dan skrip *HTML (Hyper Text Markup Language)* adalah bahasa standar untuk membuat halaman *web*. Berikut contoh kode *PHP* yang berada di dalam kode *HTML* (disimpan dengan ekstensi .php) :

```
<HTML>
<HEAD>
    <TITLE>Belajar PHP</TITLE>
</HEAD>
<BODY>
    <p>Belajar PHP</p>
    <?php
        $kata="selamat datang";
        Echo $kata;
    ?>
</BODY>
</HTML>
```

Bila dijalankan melalui *browser*, kode tersebut membentuk tampilan seperti berikut:



Gambar 2.2 Tampilan *PHP* Sederhana

2.3.2 Pengertian *MySQL*

Badiyanto (2013:57), *MySQL* merupakan sebuah database server *SQL multiuser* dan *multi threaded*.

Edy dkk. (2014 : 102). *MySQL* adalah sebuah *software database*. *MySQL* merupakan tipe data relasional yang artinya *MySQL* menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan.“

2.3.3 Pengertian *HTML*

Kadir (2013:5), *HTML (Hypertext Markup Language)* adalah suatu bahasa pemrograman dalam bentuk skrip yang dapat digunakan untuk menyusun halaman web.

Edy dkk. (2014:2), *HTML* adalah bahasa yang mengatur bagaimana tampilan isi dari situs web, di dalam html ada tag-tag dimana tag berfungsi menyediakan informasi berkaitan dengan sifat dan struktur konten serta referensi untuk gambar dan media lainnya.



2.3.4 Pengertian CSS

Sya'ban, Wahyu (2010:37), *CSS (Cascading Style Sheet)* adalah suatu bahasa yang dikhususkan untuk mengatur gaya atau *layout* sebuah halaman *web*.

2.3.5 Pengenalan Adobe Dreamweaver CS6

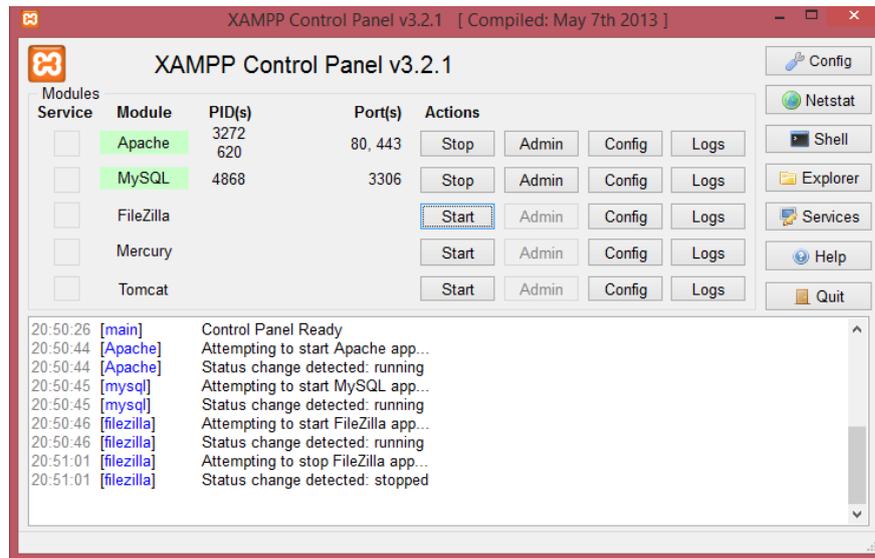
Sadeli (2014:12), *Dreamweaver* merupakan suatu perangkat lunak *web editor* keluaran *Adobe System* yang digunakan untuk membangun dan mendesain suatu *website* dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya.

Dreamweaver CS6 tergabung dalam paket *Adobe Creative Suite (CS)* yang di dalamnya terdapat paket desain grafis, *video editing*, dan pengembangan *web* aplikasi.

Pada *Dreamweaver CS6* terdapat beberapa fitur baru dari versi sebelumnya seperti *Fluid Grid Layout* yang dapat mengatur multi halaman *web* dengan mudah, *Web Fonts Manager* manajemen yang memungkinkan anda untuk menggunakan font yang tersedia di *web server*, *CSS Style Panel* tambahan untuk membuat grafis seperti bayangan pada elemen tertentu, *gradient*, membuat sudut oval dan lain sebagainya, *CSS Transitions* spesial efek pada elemen, *phone GAP* memungkinkan anda untuk membangun aplikasi *smartphone* menjadi lebih mudah dari sebelumnya.

2.3.6 Pengenalan XAMPP

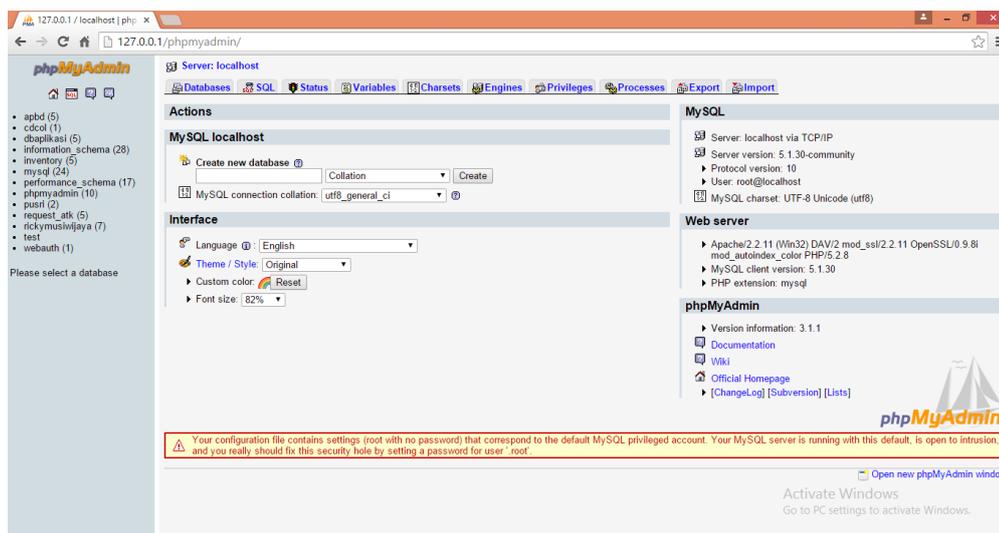
Nugroho (2013:1), menjelaskan *XAMPP* adalah paket program *web* lengkap yang dapat Anda pakai untuk belajar pemrograman *web*, khususnya *PHP* dan paket ini dapat didownload secara gratis dan legas.



Gambar 2.3 Tampilan XAMPP

2.3.7 PHPMYAdmin

Nugroho (2013:71) menjelaskan, *phpMyAdmin* adalah tools *MySQL Client* berlisensi *Freeware*, Anda dapat mengunduhnya secara gratis pada situs resminya di www.phpmyadmin.net.



Gambar 2.4 Tampilan phpMyAdmin



2.3.8 JavaScript

Badiyanto (2013:27) *JavaScript* adalah bahasa pemrograman *java*, perintah-perintahnya ditulis dengan kode yang disebut skrip. *Java* adalah Bahasa pemrograman berorientasi objek, sedangkan *script* adalah serangkaian intruksi program.

Edy dkk. (2014:129), “*JavaScript* adalah bahasa scripting client side / kode pemrograman yang bisa dimasukkan atau disisipkan ke halaman HTML.

Sidik, Betha (2011:1), *JavaScript* merupakan bahasa *script*, bahas ayang tidak memerlukan kompiler untuk menjlankannya, cukup dengan *interpreter*. Tidak perlu ada proses kompilasi terlebih dahulu agar program dapat dijalankan.

Tipe data yang dpat diproses dalam *Javascript*, dibedakan menjadi :

a. *Numerik*

Adalah data berupa bilangan atau angka yang dapat dikenai operasi matematika.

b. *String*

Adalah semua data berupa huruf, tanda baca, angka, dan rangkaian huruf atau angka.

c. *Boolean*

Adalah data yang terdiri dari dua nilai, benar atau salah, *true* atau *false*.

d. *Tanggal*

Adalah data khusus yang menyatakan tentang waktu – tanggal dan jam.

e. *Objek*

Adalah tipe data yang mendefinisikan objek.

f. *RegExp*

Adalah tipe data yang digunakan untuk mendefinisikan data kosong (*null*).

g. *Null dan undefined*

Null untuk mendefinisikan data kosong sedangkan *undefined* untuk data yang belum didefinisikan.