

BAB II KAJIAN TEORI

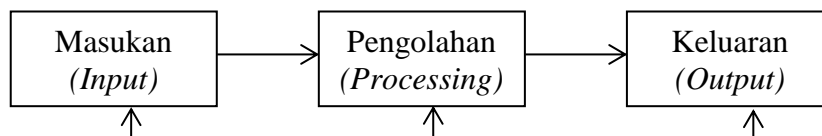
A. Deskripsi Teori

Beberapa teori yang digunakan sebagai dasar dari penelitian ini adalah (1) Sistem Informasi, (2) Sistem Informasi Berbasis Web, (3) *SMS Gateway*, (4) Tahap Pengembangan Rekayasa Perangkat Lunak, (5) *Software Requirements Specification* (SRS), (6) Prosedur, (7) Notasi Pemodelan Sistem, (8) *Framework Bootstrap* (9) Pelayanan Alva Edison Education (AEE) Private. Penjelasan lebih lanjut mengenai teori-teori tersebut adalah sebagai berikut.

1. Sistem Informasi

Menurut Kadir (2014:61-62), sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Elemen-elemen yang membentuk sebuah sistem yaitu tujuan, masukan, keluaran, proses, mekanisme, pengendalian, dan umpan balik. Selain itu, sistem juga berinteraksi dengan lingkungan dan memiliki batas.

Sedangkan menurut Scott (1996), sistem terdiri dari unsur-unsur seperti masukan (*input*), pengolahan (*processing*), serta keluaran (*output*). Model sistem tersebut seperti pada Gambar 1.



Gambar 1. Model Sistem

Informasi adalah data yang telah diolah menjadi bentuk yang berarti bagi penerimanya dan bermanfaat dalam mengambil keputusan saat ini atau mendatang (Supriyanto, 2005:243).

Menurut Murdowati (1998:1) definisi sistem merupakan sekumpulan komponen terintegrasi untuk mencapai suatu tujuan. Sedangkan berdasarkan pendekatan komponen, sistem merupakan kumpulan dari komponen-komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Dalam perkembangan sistem yang ada, sistem dibedakan menjadi dua jenis, yaitu sistem terbuka dan sistem tertutup. Sistem terbuka merupakan sistem yang dihubungkan dengan arus sumber daya luar dan tidak mempunyai elemen pengendali. Sedangkan sistem tertutup tidak mempunyai elemen pengontrol dan dihubungkan pada lingkungan sekitarnya.

Menurut Henry C. Lucas Jr. (1993:3) sistem informasi adalah sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan/atau untuk mengendalikan organisasi.

Menurut Kadir (2014:71-72) sistem informasi terdiri dari komponen-komponen sebagai berikut.

- a. Perangkat keras (*hardware*), yang mencakup peranti-peranti fisik seperti komputer dan printer.
- b. Perangkat lunak (*software*) atau program, yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
- c. Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.

- d. Orang, yakni semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.
- e. Basis data (*database*), yaitu kumpulan tabel, hubungan, dan lain-lain yang berkaitan dengan penyimpanan data.
- f. Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

Menurut Abdul Kadir (2008: 7), sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. Untuk dapat mempunyai arti data diolah sedemikian rupa sehingga dapat digunakan oleh penggunanya. Hasil pengolahan data inilah yang disebut sebagai informasi.

Berdasarkan uraian di atas maka dapat disimpulkan bahwa sistem informasi adalah data yang telah diolah dan mempunyai arti bagi penggunanya. Sehingga sistem informasi dapat didefinisikan sebagai prosedur-prosedur yang digunakan untuk mengolah data sehingga dapat digunakan oleh penggunanya.

2. Sistem Informasi Berbasis Web

Menurut Raharjo (2014: 20), sistem informasi berbasis web adalah suatu sistem informasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti internet atau intranet. *World Wide Web* atau WWW atau juga dikenal dengan *web* adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. *Web* ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet dari sekedar informasi “sampah”

atau informasi yang tidak berguna sama sekali sampai informasi yang serius, dari informasi yang gratisan sampai informasi yang komersial. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).

Sebuah *website* harus tersedia unsur unsur penunjangnya, yaitu sebagai berikut.

a. Nama domain (*Domain name/URL–Uniform Resource Locator*)

Pengertian nama domain atau biasa disebut dengan *Domain Name* atau URL adalah alamat unik di dunia internet yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain *domain name* adalah alamat yang digunakan untuk menemukan sebuah *website* pada dunia internet.

Nama domain diperjual belikan secara bebas di internet dengan status sewa tahunan. Nama domain sendiri mempunyai identifikasi ekstensi/akhiran sesuai dengan kepentingan dan lokasi keberadaan *website* tersebut. Contoh nama domain berekstensi internasional adalah com, net, org, info, biz, name, ws. Contoh nama domain berekstensi lokasi Negara Indonesia adalah co.id (untuk nama domain *website* perusahaan), ac.id (nama domain *website* pendidikan), go.id (nama domain *website* instansi pemerintah), or.id (nama domain *website* organisasi).

b. Rumah tempat *website* (*Web hosting*)

Pengertian *Web Hosting* dapat diartikan sebagai ruangan yang terdapat

dalam harddisk tempat menyimpan berbagai data, file-file, gambar dan lain sebagainya yang akan ditampilkan di *website*. Besarnya data yang dapat dimasukkan tergantung dari besarnya ruangan harddisk yang disewa atau dipunyai, semakin besar *web hosting* semakin besar pula data yang dapat dimasukkan dan ditampilkan dalam *website*. *Web Hosting* juga diperoleh dengan menyewa. Besarnya hosting ditentukan ruangan harddisk dengan ukuran MB (*Mega Byte*) atau GB (*Giga Byte*). Lama penyewaan web hosting rata-rata dihitung per tahun. Penyewaan hosting dilakukan dari perusahaan-perusahaan penyewa *web hosting* yang banyak dijumpai baik di Indonesia maupun luar negeri.

c. Bahasa Program (*Scripts Program*)

Bahasa program adalah bahasa yang digunakan untuk menerjemahkan setiap perintah dalam *website* yang pada saat diakses. Jenis bahasa program sangat menentukan statis dinamis atau interaktifnya sebuah *website*. Semakin banyak ragam bahasa program yang digunakan maka akan terlihat *website* semakin dinamis dan interaktif serta terlihat bagus.

Beragam bahasa program saat ini telah hadir untuk mendukung kualitas *website*. Jenis bahasa program yang banyak dipakai para desainer *website* antara lain HTML, ASP, PHP, JSP, Java Scripts, Java applets. Bahasa dasar yang dipakai setiap situs adalah HTML sedangkan PHP, ASP, JSP dan lainnya merupakan bahasa pendukung yang bertindak sebagai pengatur dinamis dan interaktifnya situs. Bahasa program ASP, PHP, JSP atau lainnya dapat dibuat sendiri. Bahasa program ini biasanya digunakan untuk membangun portal berita, artikel, forum diskusi, buku tamu, anggota organisasi, email, *mailing list* dan lain sebagainya yang

memerlukan *update* setiap saat.

d. Desain *website*

Setelah melakukan penyewaan domain name dan web hosting serta penguasaan bahasa program (*scripts program*), unsur *website* yang penting dan utama adalah desain. Desain *website* menentukan kualitas dan keindahan sebuah *website*. Desain sangat berpengaruh kepada penilaian pengunjung akan bagus tidaknya sebuah *website*.

Untuk membuat *website* biasanya dapat dilakukan sendiri atau menyewa jasa *website designer*. Saat ini sangat banyak jasa *web designer*, terutama di kota-kota besar. Perlu diketahui bahwa kualitas situs sangat ditentukan oleh kualitas designer. Semakin banyak penguasaan *web designer* tentang beragam program atau *software* pendukung pembuatan situs maka akan dihasilkan situs yang semakin berkualitas, demikian pula sebaliknya. Jasa *web designer* ini yang umumnya memerlukan biaya yang tertinggi dari seluruh biaya pembangunan situs dan semuanya itu tergantung kualitas *designer*.

Berdasarkan uraian di atas maka dapat disimpulkan bahwa sistem informasi berbasis *web* adalah suatu sistem informasi yang diakses menggunakan penjelajah *web* melalui suatu jaringan seperti internet atau intranet. Sedangkan *website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya. Unsur-unsur penunjang dari sebuah *website* adalah nama domain (*domain name/URL–Uniform Resource Locator*), rumah tempat *website* (*web hosting*), bahasa program (*scripts program*),

dan desain *website*

3. SMS Gateway

a. Pengertian

Menurut Meyn (2014: 20), *SMS Gateway* adalah sebuah perangkat lunak atau platform yang menawarkan layanan transmit SMS, mentransformasikan pesan ke jaringan seluler dari media lain, atau sebaliknya, yang kemudian dipergunakan oleh penyedia jasa untuk mengatur SMS yang akan dikirim dan diterima. Hal ini memungkinkan pengiriman atau penerimaan pesan SMS dengan atau tanpa menggunakan handphone. *SMS Gateway* bekerja dengan cara menghubungkan handphone dengan sebuah komputer. Komunikasi antara handphone dengan komputer dilakukan dengan menggunakan bahasa pemrograman *AT-Command*.

Dengan adanya software *SMS Gateway*, maka seseorang dapat mengatur SMS lewat PC atau laptop dengan mudah dan cepat. Selain ini *SMS Gateway* juga memungkinkan seseorang dapat mengirimkan pesan keluar secara banyak atau per grup dalam sekali kirim atau klik dengan jawaban SMS yang sudah diatur secara otomatis. Sebagai contoh SMS Registrasi, ketika seseorang melakukan registrasi ke suatu layanan tertentu, maka secara otomatis seseorang tersebut akan mendapat SMS balasan yang mengatakan bahwa ia telah terdaftar teregristrasi.

b. Cara Kerja SMS Gateway

Cara kerja *SMS Gateway* terbilang cukup mudah. Sebagai contoh saat seseorang ingin melakukan registrasi pada suatu aplikasi tertentu yang terhubung dengan layanan *SMS Gateway*, sebenarnya untuk mendapatkan balasan bahwa

orang tersebut telah teregistrasi, maka orang itu harus melewati tahapan sebagai berikut.

1) Tahap Pengiriman Pesan

Pada tahap ini SMS yang dikirim akan melewati jaringan seluler, pesan tersebut kemudian ditangkap dan diterima oleh layanan SMS *Gateway*. Pesan tersebut kemudian diproses di dalam sistem aplikasi SMS *Gateway*.

2) Tahap membalas pesan secara otomatis

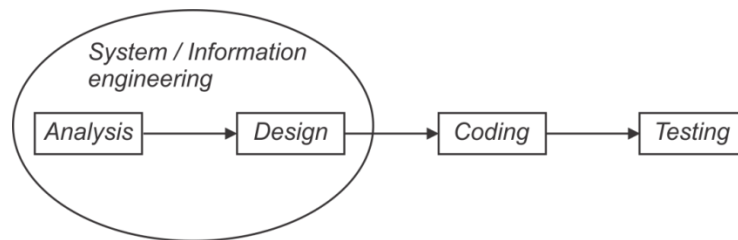
Di dalam sistem aplikasi, pesan akan dihubungkan dengan database server. Di dalam *database server* inilah semua pesan balasan telah disiapkan. Program secara otomatis akan mengambil data di dalam *database server* lalu mengirimkan pesan balasan kepada orang yang mengirimkan SMS itu.

Berdasarkan uraian di atas maka dapat disimpulkan bahwa SMS *Gateway* adalah sebuah perangkat lunak atau platform yang menawarkan layanan transmit SMS, mentransformasikan pesan ke jaringan seluler dari media lain, atau sebaliknya, yang kemudian dipergunakan oleh penyedia jasa untuk mengatur SMS yang akan dikirim dan diterima. Cara kerja SMS *Gateway* adalah tahap pengiriman pesan dan tahap membalas pesan secara otomatis.

4. Tahap Pengembangan Rekayasa Perangkat Lunak Waterfall

Menurut Pressman (2001:28-29) *Waterfall Model* atau disebut juga *classic life cycle* adalah suatu model yang digunakan dalam pengembangan perangkat lunak dengan pendekatan sistematis dan sekuensial. Tahapan dalam *Waterfall Model* adalah *analysis, design, coding, testing, dan support*. Tahapan dalam

Waterfall Model tersebut digambarkan dalam Gambar 2.



Gambar 2. *Waterfall Model*

Pengembangannya Metode *Waterfall* memiliki beberapa tahapan sebagai berikut.

a. Definisi dan Analisa Kebutuhan (*Requirements Analysis and Definition*)

Dalam langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini dapat melakukan sebuah penelitian wawancara atau *study literatur*. Seseorang sistem analisis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem komputer yang dapat melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirement* atau dapat dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan *system analyst* untuk menterjemahkan kedalam bahasa pemrograman.

b. Desain Sistem dan Perangkat Lunak (*System and Software Design*)

Proses desain akan menterjemahkan syarat kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi *interface* dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan

dokumen yang disebut *software requirement*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistemnya.

c. Implementasi Sistem

Coding merupakan penerjemahan desain dalam bahasa yang dapat dikenal oleh komputer. Dilakukan oleh *programmer* yang akan menterjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam arti penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan uji coba terhadap sistem yang telah dibuat tadi. Tujuan uji coba adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian dapat diperbaiki.

d. Uji Coba Sistem

Tahapan ini dapat dikatakan final dalam pembuatan sebuah sistem. Setelah melakukan analisa, desain dan pembuatan program, maka sistem yang sudah jadi akan diuji coba secara α test dan β test untuk mengetahui apakah program sudah bebas dari kesalahan.

Berdasarkan uraian di atas maka dapat disimpulkan bahwa *Waterfall Model* adalah suatu model yang digunakan dalam pengembangan perangkat lunak dengan pendekatan sistematis dan sekuensial. Pengembangan Metode *Waterfall* memiliki beberapa tahapan yaitu definisi dan analisa kebutuhan, desain sistem dan perangkat lunak, implementasi sistem, dan uji coba sistem.

5. *Software Requirements Specification (SRS)*

Menurut Wisnu (2012), *Software Requirements Specification (SRS)* adalah

dokumen yang menjelaskan tentang berbagai kebutuhan yang harus dipenuhi oleh suatu *software*. Dokumen ini dibuat oleh *developer* (pengembang *software*) setelah menggali informasi dari calon pemakai *software*. Pembuatannya mengikuti standar yang ada dan paling dianggap benar oleh para praktisi rekayasa *software* di dunia. Oleh karena itu, standar yang akan dibahas di sini adalah standar dari IEEE, singkatan dari *Institute of Electrical and Electronics Engineers*.

SRS harus bermanfaat bagi pengguna, penyedia, atau perorangan. Adapun manfaat dari SRS antara lain sebagai berikut.

- a. Sebagai bentuk perjanjian antara pengguna dan penyedia tentang *software* apa yang akan dibuat.
- b. Mengurangi beban dalam proses pengembangan *software*.
- c. Sebagai bahan perkiraan biaya dan rencana penjadwalan.
- d. Sebagai dasar validasi dan verifikasi *software* di ujung penyelesaian proyek.
- e. Memfasilitasi transfer, semisal *software* tersebut ingin ditransfer ke pengguna atau mesin-mesin yang lain. Penggunaan lebih mudah jika ingin mentransfer *software* ke bagian-bagian lain dalam organisasinya. Bahkan, jika terjadi pergantian personil pengembang, proyek dapat mudah ditransfer ke personil baru dengan memahami SRS ini.
- f. Mendasari perbaikan produk *software* dikemudian hari. Jadi, SRS boleh diperbaiki dengan alasan dan mekanisme tertentu serta atas kesepakatan antara pengguna dan pengembang.
- g. Dengan menggunakan SRS, pengguna dapat menuangkan semua ide

terkait software dengan jelas dan akurat sehingga pengembang dapat memahami apa yang diinginkan pengguna dengan tepat. Standar ini akan membantu dalam mengembangkan *outline* SRS yang baku untuk pengguna, membantu membuat dokumen SRS dengan format dan isi yang standar (minimal), serta membantu mengembangkan rincian-rincian pendukung lainnya.

SRS harus memiliki jaminan kualitas perangkat lunak dan pengujian aplikasi. Jaminan perangkat lunak adalah aktivitas pelindung yang diaplikasikan pada seluruh proses perangkat lunak. Tujuan dari jaminan kualitas adalah untuk memberikan data yang diperlukan oleh manajemen dan menginformasikan masalah kualitas produk, sehingga dapat memberikan kepastian dan konfidensi bahwa kualitas produk dapat memenuhi sasaran, tidak hanya berkualitas menurut pengembang tapi juga berkualitas dan sesuai dengan keinginan pengguna (Nastiti, 2012: 35).

Menurut McCall dalam Nastiti (2012: 36), faktor-faktor penentu kualitas perangkat lunak adalah sebagai berikut.

- a. *Correctness*, sejauh mana suatu perangkat lunak memenuhi spesifikasi dan tujuan penggunaan perangkat lunak dari user.
- b. *Reliability*, sejauh mana keakuratan suatu perangkat lunak dalam melaksanakan fungsinya.
- c. *Efficiency*, banyaknya kode program yang dibutuhkan suatu perangkat lunak untuk melakukan fungsinya.
- d. *Integrity*, sejauh mana akses ke perangkat lunak dan data oleh pihak yang tidak berhak dapat dikendalikan.

- e. *Usability*, usaha yang diperlukan untuk mempelajari, mengoperasikan, menyiapkan input, dan mengartikan output dari perangkat lunak.
- f. *Maintainability*, usaha yang diperlukan untuk menetapkan dan memperbaiki kesalahan dalam program.
- g. *Testability*, usaha yang diperlukan dalam pengujian program untuk memastikan bahwa program melaksanakan fungsi yang ditetapkan.
- h. *Flexibility*, usaha yang diperlukan untuk memodifikasi program operasional.
- i. *Portability*, usaha yang diperlukan untuk memindahkan program dari perangkat keras/lingkungan sistem perangkat lunak tertentu ke yang lainnya.
- j. *Reusability*, tingkat kemampuan program/bagian dari program yang dapat dipakai ulang dalam aplikasi lainnya, berkaitan dengan paket dan lingkup dari fungsi yang dilakukan oleh program.
- k. *Interoperability*, usaha yang diperlukan untuk menggabungkan satu sistem dengan yang lainnya.

Menurut Pressman (2010:30), pengujian *software* adalah metode yang dilakukan untuk menjelaskan tentang pengoperasian perangkat lunak yang terdiri dari perangkat pengujian, metode pengujian dan pelaksanaan pengujian. Ada 2 jenis pengujian, yaitu:

a. Pengujian *Alpha*

Pengujian *Alpha* dilakukan pada sisi pengembang. Software digunakan pada pengaturan yang natural dengan pengembang yang memandang sisi pemakai dan merekam semua kesalahan dan masalah pemakaian. Pengujian *Alpha* bertujuan untuk mengidentifikasi dan menghilangkan sebanyak mungkin masalah sebelum

akhirnya sampai ke pengguna, dilakukan setelah *software* selesai oleh orang-orang yang tidak terlibat dalam pengembangan dan memang ahli dibidangnya menggunakan formulir evaluasi resmi.

b. Pengujian *Betha*

Pengujian *Betha* dilakukan kepada satu atau lebih pengguna *software* dalam lingkungan yang sebenarnya, pengembang tidak terlibat pengujian ini. Pengguna merekam semua masalah (nyata atau imajiner) yang ditemui selama pengujian dan melaporkan pada pengembang pada interval waktu tertentu. Berdasarkan hasil pengujian *Betha*, dicari persentase masing-masing jawaban pertanyaan dengan menggunakan rumus sebagai berikut (Pressman, 2010:30).

$$Y = (P/Q) \times 100\%$$

Keterangan:

Y = Nilai persentase

P = Banyaknya jawaban responden tiap pertanyaan

Q = Jumlah responden


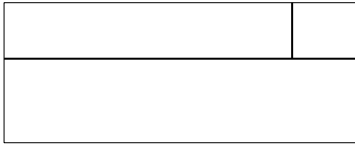
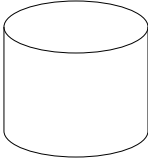

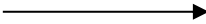
Berdasarkan uraian di atas maka dapat disimpulkan bahwa *Software Requirements Specification* (SRS) adalah dokumen yang menjelaskan tentang berbagai kebutuhan yang harus dipenuhi oleh suatu *software*. Faktor-faktor penentu kualitas perangkat lunak adalah *correctness*, *reliability*, *efficiency*, *integrity*, *usability*, *maintainability*, *testability*, *flexibility*, *portability*, *reusability*, dan *interoperability*.

6. Prosedur

Menurut Yogiarto (1995), prosedur adalah urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa (*what*) yang harus dikerjakan,

siapa (*who*) yang mengerjakan, kapan (*when*) dikerjakan, dan bagaimana mengerjakannya. Simbol yang digunakan dalam prosedur disajikan pada Tabel 1.

Tabel 1. Simbol-simbol Prosedur

Simbol	Keterangan
	Mulai / Selesai
	Proses
	Tempat Penyimpanan
	Dokumen
	Penghubung

7. Notasi Pemodelan Sistem

a. *Use Case Diagram*

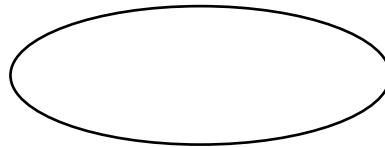
1) Pengertian

Use case diagram ialah sebuah grafis yang menggambarkan interaksi antara sistem dan pengguna. Dengan kata lain *use case diagram* secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna (*user*) mengharapkan interaksi dengan sistem itu. *Use case* secara naratif digunakan untuk secara tekstual menggambarkan skueni langkah-langkah

dari setiap interaksi.

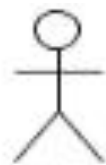
2) Simbol Utama *Use Case* dan Aktor

Use case jika digambarkan secara grafik berbentuk elips seperti dibawah ini :



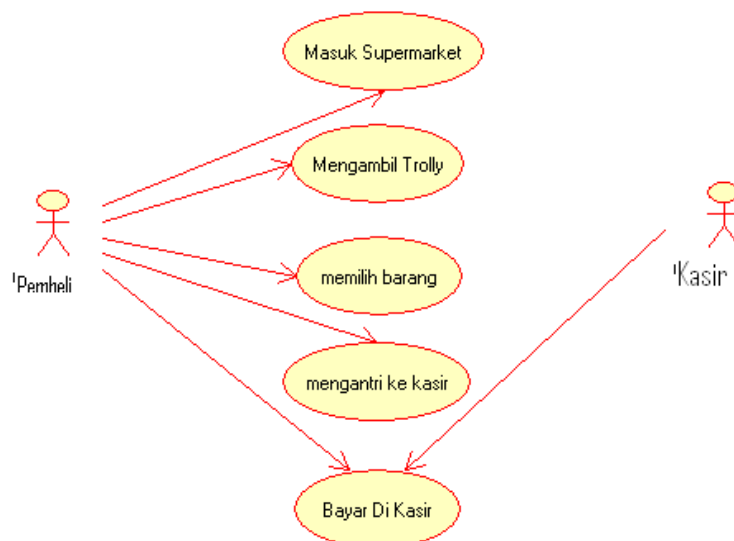
Gambar 3. *Use Case*

Aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.



Gambar 4. Aktor

Contoh *Use Case Diagram* ditampilkan pada Gambar 5.



Gambar 5. Contoh *Use Case Diagram*





b. Data Flow Diagram (DFD)

1) Pengertian

Data Flow Diagram (DFD) adalah diagram yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, di mana data disimpan, proses apa yang dihasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut (Kristanto, 2008:61).

2) Simbol Data Flow Diagram

- a) *User / terminator* : Kesatuan di luar sistem (*external entity*) yang memberikan input ke sistem atau menerima output dari sistem berupa orang, organisasi, atau sistem lain.
- b) *Proses* : Aktivitas yang mengolah input menjadi output.
- c) *Data Flow* : Aliran data pada sistem (antar proses, antara *terminator* dan proses, serta antara proses dan *data store*).
- d) *Data Store* : Penyimpanan data pada database, biasanya berupa tabel.

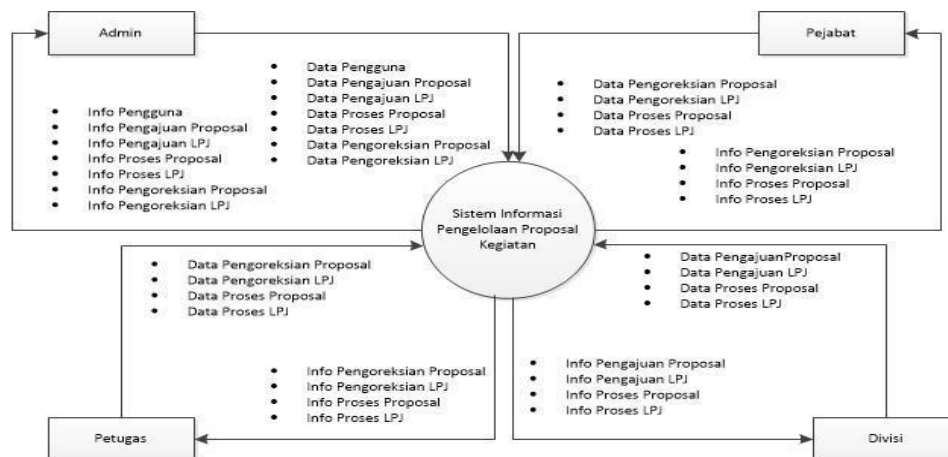
<i>Terminator</i>	
<i>Proses</i>	
<i>Data Flow</i>	
<i>Data Store</i>	

Gambar 6. Simbol Data Flow Diagram

3) Jenis-jenis Data Flow Diagram

a) Diagram Konteks (*Context Diagram*)

Diagram konteks merupakan diagram yang paling sederhana dari sebuah sistem informasi yang menggambarkan aliran data dari kesatuan luar ke dalam sistem dan sebaliknya. Contoh diagram konteks ditampilkan pada Gambar 7.



Gambar 7. Contoh Diagram Konteks

b) *DFD* Level n

DFD Level n merupakan diagram yang digunakan untuk menggambarkan diagram hasil pengembangan dari diagram konteks ke dalam komponen yang lebih detail. Nilai n merupakan banyaknya angka atau digit yang digunakan untuk penomoran proses yang ada. Contoh *DFD* Level 1 disajikan pada Gambar 8.

c. ER Diagram

1) Pengertian Model E-R

Menurut Kadir (2008), Model E-R adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas, atribut, dan hubungan antar entitas, huruf E sendiri menyatakan entitas dan menyatakan hubungan (dari kata *relationship*). Model ini dinyatakan dalam bentuk diagram. Itulah sebabnya model E-R seringkali juga disebut sebagai diagram E-R. Perlu diketahui bahwa model seperti ini tidak mencerminkan bentuk fisik yang nantinya akan disimpan dalam *database*, melainkan hanya bersifat konseptual. Itulah sebabnya model ER tidak bergantung pada produk DBMS yang akan digunakan.

2) Komponen dalam ER Diagram

a) Entitas

Entitas adalah suatu objek yang dapat dibedakan dari lainnya dan dapat diwujudkan dalam basis data.

b) Hubungan (relasi/*Relationship*)

Hubungan dua jenis entitas dan direpresentasikan sebagai garis lurus yang menghubungkan dua entitas.

c) Atribut

Memberikan informasi lebih rinci tentang jenis entitas.

3) Kardinalitas Rasio

Kardinalitas rasio menjelaskan jumlah maksimum hubungan antara satu entitas dengan entitas lainnya.

a) One to One (1 : 1)

Setiap anggota entitas A hanya boleh berhubungan dengan satu anggota entitas B, begitu pula sebaliknya.

b) One to Many (1 : M)

Setiap anggota entitas A dapat berhubungan dengan lebih dari satu anggota entitas B tetapi tidak sebaliknya.

c) Many to Many (M : M)

Setiap entitas A dapat berhubungan dengan banyak entitas himpunan entitas B dan demikian pula sebaliknya.

Berdasarkan uraian di atas yang akan digunakan dalam penulisan adalah notasi pemodelan sistem *Data Flow Diagram* (DFD). *Data Flow Diagram* (DFD) adalah diagram yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem. Jenis-jenis DFD adalah diagram konteks dan diagram level n.

8. Framework Bootstrap

Framework adalah sebuah *tools* untuk memudahkan para programmer dalam membuat aplikasi berbasis *website*. Di dalam *framework* terdapat fungsi, *plugin*, dan konsep untuk membentuk suatu sistem tertentu. Dengan menggunakan *framework*, sebuah aplikasi akan tersusun dan terstruktur dengan rapi (Davies, 2011).

Bootstrap adalah sebuah *framework* yang berfungsi untuk mengubah tampilan halaman *website* menjadi lebih menarik dan *responsive* sehingga lebar halaman *website* yang ditampilkan menyesuaikan dengan lebar *browser* (Cavaliere, dkk., 2015). Bootstrap terdiri dari kode yaitu *Cascading Style Sheet* (CSS) dan

JQuery (*Javascript Query*). CSS pada Bootstrap berfungsi untuk mengatur tampilan antarmuka, seperti: tipografi, formulir, tombol, navigasi, dan antarmuka halaman *website*. Sedangkan fungsi JQuery pada Bootstrap adalah memberi efek animasi pada halaman *website* (DeDeo, dkk., 2013).

9. Alva Edison Education Private

Alva Edison Education Private merupakan Lembaga Bimbingan bersifat non formal yang melayani bimbingan belajar privat untuk tingkat SD, SMP, SMK, dan Umum. Lembaga Bimbingan ini berdiri pada tanggal 1 Januari 2015 dan didirikan oleh Nisa Fauziah seorang mahasiswa tingkat akhir Prodi Matematika Universitas Negeri Yogyakarta. Lembaga ini telah beroperasi pesat di Yogyakarta. Awalnya hanya mempunyai 20 siswa dan sekarang sudah 50 siswa dengan rata-rata 300 pertemuan dalam sebulan.

Berikut adalah fasilitas-fasilitas dari Alva Edison Education Private.

a. Konsep Tempat

Alva Edison Education Private memiliki fasilitas tutor datang ke rumah siswa. Jadi siswa tidak perlu untuk keluar rumah. Selain itu siswa juga dapat melakukan bimbingan di rumah makan, cafe, dsb. Jika siswa menghendaki melakukan bimbingan di kantor Alva Edison Education Private.

b. Konsep Materi

Materi bimbingan sesuai dengan permintaan siswa, sehingga tutor menjelaskan sesuai materi yang telah didapatkan siswa di sekolah. Selain itu juga diberikan fasilitas pemberian ringkasan materi dan latihan soal agar siswa lebih menguasai materi.

c. Konsep Tentor

Tentor Alva Edison Education Private adalah mahasiswa semester akhir atau alumnus UNY. Selain itu tentor ber-IPK di atas 3.00. Tentor-tentor yang mengajar sesuai dengan bidangnya masing-masing. Selain itu siswa juga dapat memilih tentor sesuai keinginan siswa.

d. Konsep Administrasi

Pembayaran administrasi di bimbingan ini menggunakan sistem mengambil minimal 4x pertemuan dibayarkan diawal. Pembayaran dapat menggunakan via transfer atau pengambilan biaya oleh staf Alva Edison Education Private ke rumah siswa atau ditempat umum. Pembayaran juga dapat dilakukan di kantor.

Keunggulan yang dimiliki antara lain:

- a. *Free* ringkasan materi
- b. *Free* latihan soal dan pembahasan
- c. Jadwal fleksibel
- d. Tentor datang ke rumah
- e. Siswa mendapat tentor sesuai dengan keinginannya
- f. Tentor mahasiswa/i semester akhir dengan IPK >3,00 atau alumnus UNY
- g. Tentor dengan *background* pendidikan PGSD (Pendidikan Guru Sekolah Dasar), Pendidikan Matematika/Matematika, Pendidikan Fisika/Fisika, Pendidikan Kimia/Kimia, Pendidikan Biologi/Biologi, Pendidikan Geografi, Pendidikan Sosiologi, Pendidikan Ekonomi, Pendidikan Bahasa Indonesia, Pendidikan Bahasa Inggris.

Proses administrasi Alva Edison Education Private masih dilakukan secara

manual. Contohnya pada proses pendaftaran siswa yang masih manual dengan cara siswa yang ingin mendaftar harus bertemu dengan admin, selanjutnya mengisi formulir pendaftaran siswa, memilih mata pelajaran, menentukan jadwal, dan membayar biaya bimbingan les sesuai paket yang telah dipilih secara tunai. Siswa diberikan kwitansi sebagai bukti pembayaran dan kartu kontrol siswa sebagai presensi sewaktu melakukan bimbingan les. Setelah itu akan diproses untuk menentukan tutor sesuai mata pelajaran dan jadwal yang diinginkan siswa. Bimbingan les akan dilakukan pada jadwal yang diinginkan siswa dan di tempat yang diinginkan siswa pula. Selain pada proses pendaftaran, proses pencetakan jadwal dan administrasi yang lain juga masih menggunakan sistem yang manual, sehingga dapat berdampak dalam segala hal, seperti laporan administrasi yang terlambat, pelayanan yang tidak efisien terhadap siswa, lamanya proses pendaftaran siswa, serta keterlambatan pembayaran paket les karena siswa dan orang tua siswa tidak mengetahui apakah paket lesnya masih atau sudah habis.

B. Kerangka Berpikir

Berdasarkan uraian permasalahan di depan, dapat dianalisa beberapa kekurangan dari sistem pelayanan Bimbingan Alva Edison Education Private sebagai berikut.

1. Penyimpanan data pendaftaran dan pembayaran masih menggunakan media penyimpanan manual. Informasi yang diperlukan sering kurang efektif.
2. Proses administrasi yang masih manual dan membutuhkan proses yang lama menjadi tidak efisien.
3. Keterlambatan pembayaran paket les karena siswa dan orang tua siswa tidak

mengetahui apakah paket lesnya masih atau sudah habis.

Rencana sistem yang akan dibuat untuk Alva Edison Education Private ini adalah sebuah sistem yang digunakan untuk menangani beberapa hal sebagai berikut.

1. Menangani proses pendaftaran online
2. Menangani proses pembayaran
3. Menangani proses pengolahan data siswa, data nilai, data pembayaran, data jadwal, dan data tentor
4. Memberikan informasi jadwal bimbingan dan nilai kepada siswa secara online
5. Memberikan informasi tentang laporan administrasi kepada staff Alva Edison Education Private.
6. Penggunaan SMS *Gateway* meliputi, cek jadwal tentor dan siswa, cek jumlah tagihan pembayaran, cek sisa paket les, cek nilai, dan *reminder* sisa paket les.