



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perangkat Lunak

Rosa dan Shalahuddin (2018:2), “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”. Sedangkan menurut Nurcholish (2018:14), “Perangkat lunak adalah perintah (program komputer) yang bila dieksekusi memberikan fungsi dan *performance* seperti yang diinginkan”.

Dari pernyataan diatas dapat disimpulkan bahwa perangkat lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan bila dieksekusi memberikan fungsi dan *performance* seperti yang diinginkan.

2.1.2 Metode Pengembangan Perangkat Lunak

Penelitian ini menggunakan metode pengembangan perangkat lunak *Waterfall*. Menurut Rosa dan Shalahuddin (2018:28) metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*).

Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

a. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak ini diperlukan didokumentasikan.

**b. Desain**

Desain perangkat lunak adalah proses multi langkah yang focus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak ini yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program computer sesuai dengan yang telah dibuat pada tahap desain. Desain perangkat lunak ini juga perlu di dokumentasikan.

d. Pengujian

Pengujian fokus pada perangkat lunak dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung dan pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk perangkat lunak yang baru.

2.1.2.1 Analisis PIECES

Suharto (2018:38) menjelaskan bahwa Analisis PIECES (*Performance, Information, Economy, control, Eficiency, dan Service*) merupakan teknik untuk mengidentifikasi dan memecahkan permasalahan yang terjadi pada sistem



informasi. Dari analisis ini akan menghasilkan identifikasi masalah utama dari suatu sistem serta memberikan solusi dari permasalahan tersebut.

Analisis PIECES terdiri dari:

1. Analisis Kinerja Sistem (*Performance*) Kinerja adalah suatu kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur dengan jumlah produksi (*throughput*) dan waktu yang digunakan untuk menyesuaikan perpindahan pekerjaan (*response time*).
 2. Analisis Informasi (*Information*) Informasi merupakan hal penting karena dengan informasi tersebut pihak manajemen (*marketing*) dan *user* dapat melakukan langkah selanjutnya. Apabila kemampuan sistem informasi baik, maka *user* akan mendapatkan informasi yang akurat, tepat waktu dan relevan sesuai dengan yang diharapkan.
 3. Analisis Ekonomi (*Economy*) Pemanfaatan biaya yang digunakan dari pemanfaatan informasi. Peningkatan terhadap kebutuhan ekonomis mempengaruhi pengendalian biaya dan peningkatan manfaat. Saat ini banyak perusahaan dan manajemen mulai menerapkan *paperless system* (meminimalkan penggunaan kertas) dalam rangka penghematan. Oleh karena itu dilihat dari penggunaan bahan kertas yang berlebihan dan biaya iklan di media cetak untuk media publikasi, sistem ini dinilai kurang ekonomis.
 4. Analisis Pengendalian (*Control*) Analisis ini digunakan untuk membandingkan sistem yang dianalisa berdasarkan pada segi ketepatan waktu, kemudahan akses, dan ketelitian data yang diproses.
 5. Analisis Efisiensi (*Efficiency*) Efisiensi berhubungan dengan bagaimana sumber tersebut dapat digunakan secara optimal. Operasi pada suatu perusahaan dikatakan efisien atau tidak biasanya didasarkan pada tugas dan tanggung jawab dalam melaksanakan kegiatan.
 6. Analisis Pelayanan (*Service*) Peningkatan pelayanan memperlihatkan kategori yang beragam. Proyek yang dipilih merupakan peningkatan pelayanan yang
-



lebih baik bagi manajemen (*marketing*), *user* dan bagian lain yang merupakan simbol kualitas dari suatu sistem informasi.

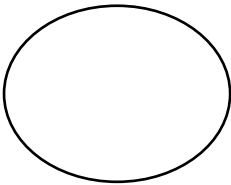
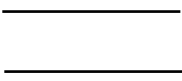
2.2 Teori Khusus

2.2.1 Pengertian DFD (*Data Flow Diagram*)

Kurniawan (2020:11), "*Data Flow Diagram* (DFD) adalah diagram arus data yang menggunakan simbol-simbol untuk menyajikan entitas, proses, arus data dan penyimpanan yang berkaitan dengan suatu sistem dengan tujuan untuk memberikan indikasi mengenai bagaimana data ditransformasi pada saat data bergerak melalui sistem dan menggambarkan fungsi-fungsi (dan sub fungsi) yang mentransformasikan aliran data". Sedangkan menurut Rosa dan Shalahuddin (2018:70), "*Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)".

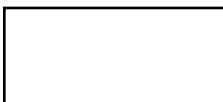

Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.1 Notasi-notasi DFD

| Notasi | Keterangan |
|---|--|
|  | Proses atau fungsi atau prosedur; Pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi ilmiah yang harusnya menjadi fungsi atau prosedur di dalam kode. Catatan : Nama yang diberikan pada sebuah proses biasanya berupa kata kerja. |
|  | File atau basis data atau penyimpanan (<i>storage</i>); Pada pemodelan perangkat lunak yang akan |



Lanjutan Tabel 2.1 Notasi-notasi DFD

| Notasi | Keterangan |
|---|--|
| | <p>diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi ilmiah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan table-tabel pada basis data (<i>Entity Relationship Diagram (ERD)</i>, <i>Conceptual Data Model (CDM)</i>, <i>Physical Data Model (PDM)</i>).</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p> |
|  | <p>Entitas luar (<i>External Entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>Catatan :</p> <p>Nama yang digunakan pada masukan (<i>input</i>) biasanya berupa kata benda.</p> |
|  | <p>Aliran data; Merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>Catatan :</p> <p>Nama yang digambarkan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.</p> |

(Sumber : Rosa dan Shalahuddin, 2018:71-72)



Rossa dan Shalahuddin menjelaskan bahwa berikut ini merupakan tahapan-tahapan perancangan dengan menggunakan DFD :

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*.

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang *breakdown* harus di- lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 harus sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

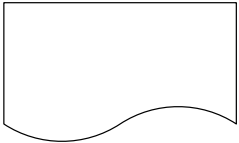
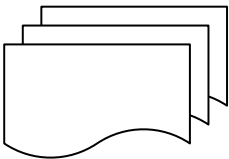
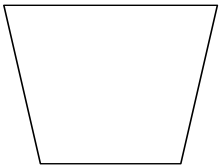

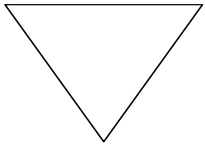
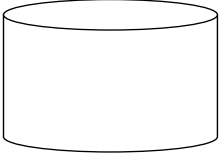
4. Membuat DFD Level 3 dan seterusnya

DFD Level 3,4,5 dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau level 2.


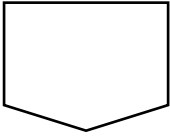
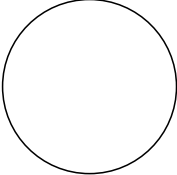
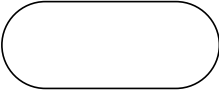
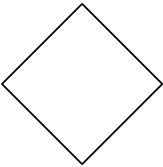


2.2.2 Pengertian *Blockchart*

Kristanto (2018:75), “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses, maupun transaksi dengan menggunakan simbol-simbol tertentu”.

Tabel 2.2 Simbol-simbol *Blockchart*

| Simbol | Arti |
|---|---|
|  | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/ bendel/ berkas atau cetakan. |
|  | Multi dokumen |
|  | Proses Manual |
|  | Proses yang dilakukan oleh komputer |
|  | Menandakan dokumen yang diarsipkan (arsip manual) |
|  | Data Penyimpanan (<i>Storage</i>) |

Lanjutan Tabel 2.2 Simbol-simbol *Blockchart*

| Simbol | Arti |
|---|---|
|  | Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik |
|  | Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain |
|  | Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama |
|  | Terminasi yang menandakan awal dan akhir dari suatu aliran |
|  | Pengambilan keputusan (<i>decision</i>) |
|  | Layar peraga (monitor) |
|  | Pemasukan data secara manual |

(Sumber :Kristanto, 2018:75-77)



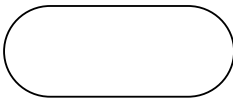
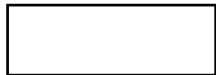
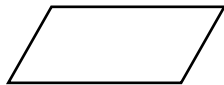
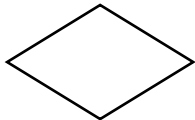
2.2.3 Pengertian Bagan Alir (*Flowchart*)

Kaharu dan Sakina (2016:36),”*Flowchart* adalah tools yang sangat fleksibel yang memungkinkan untuk meneliti proses yang kompleks dalam mengidentifikasi masalah potensial dan peluang pengembangan.”

Kadir (2017:39) menjelaskan bahwa diagram (*flowchart*) memiliki dua kategori, yaitu diagram alir sistem (*system flowchart*) dan diagram alir program (*program flowchart*).


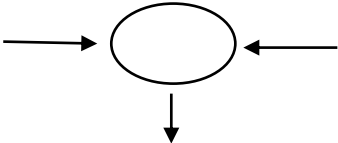
- a. Diagram alir sistem digunakan oleh analis sistem untuk menggambarkan aliran data atau struktur *file* di dalam suatu sistem.
- b. Diagram alir program digunakan oleh pemrogram untuk menjelaskan langkah-langkah di dalam suatu program.

Tabel 2.3 Simbol-simbol *Flowchart* menurut Abdul Kadir

| Simbol | Keterangan |
|---|--|
|  | Tanda yang menyatakan awal atau akhir diagram alir. Tulisan yang diletakkan di dalamnya bias berupa “Mulai” atau “Selesai”. Kadangkala, tanda ini tidak disertakan dalam diagram alir. |
|  | Tanda yang menyatakan suatu proses atau perhitungan. Umum digunakan untuk menyatakan pemberian nilai ke suatu variable. |
|  | Tanda yang menyatakan operasi masukan (membaca dari keyboard) atau keluaran (menampilkan informasi ke layar). |
|  | Tanda yang digunakan sebagai pengambilan keputusan. Di dalamnya berisi suatu kondisi yang bias bernilai ya atau tidak. |



Lanjutan Tabel 2.3 Simbol-simbol *Flowchart* menurut Abdul Kadir

| Simbol | Keterangan |
|---|---|
|  | Menyatakan arah aliran pemrosesan |
|  | Berfungsi sebagai tanda penghubung yang menyatakan arah aliran pemrosesan. Pada contoh disamping, dua aliran menuju ke satu aliran. Namun, seringkali gambar lingkaran tidak disertakan |

(Sumber : Abdul Kadir, 2017:18-19)

2.2.4 Pengertian ERD (*Entity Relational Diagram*)

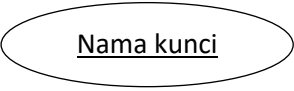
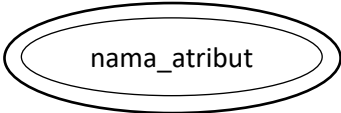
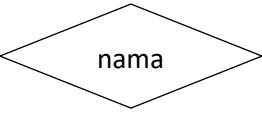
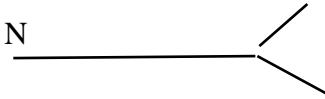
Anisya dan Allazi (2017:83), “*Entity Relationship Diagram* adalah relasi yang berisikan komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut. Sedangkan menurut Rosa dan Shalahuddin (2018:50), “ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain”. Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen :

Tabel 2.4 Simbol-simbol ERD

| Simbol | Deskripsi |
|--|--|
| Entitas / <i>entity</i> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> nama_entitas </div> | Entitas merupakan data inti yang akan disimpan; bakal table pada basis data; benda yang memiliki data dan harus disimpan datanya lebih ke kata benda dan belum merupakan nama table. |
| Atribut kunci primer | Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id; kunci |



Lanjutan Tabel 2.5 Simbol-simbol ERD

| Simbol | Deskripsi |
|---|---|
|  | primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama). |
| Atribut multivali / <i>multilevel</i>  | Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki lebih dari satu. |
| Relasi  | Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja. |
| Asosiasi / <i>association</i>  | Penghubung antara relasi dan entitas dimana 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, 2018:50-51)

2.2.5 Pengertian Kamus Data

Rosa dan Shalahuddin (2018:73), “Kamus Data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.



Berikut adalah simbol-simbol yang digunakan dalam Kamus Data :

Tabel 2.5 Simbol-simbol Kamus Data

| Simbol | Keterangan |
|------------------|---------------------------------|
| = | Disusun atau terdiri dari |
| + | Dan |
| [] | Baik...atau... |
| { } ⁿ | N kali diulang/ bernilai banyak |
| () | Data Opsional |
| *...* | Batas komentar |

(Sumber : Rosa dan Shalahuddin, 2018:74)

2.2.6 Metode Pengujian Perangkat Lunak

2.2.6.1 Pengertian Metode Pengujian

Cholifah et al. (2018:207),”Pengujian adalah satu set aktifitas yang direncanakan dan sistem untuk menguji atau mengevaluasi kebenaran yang diinginkan.

2.2.6.2 Metode Pengujian

Secara umum pola pengujian perangkat lunak adalah sebagai berikut:

1. Pengujian dimulai dari level komponen hingga integrasi antar komponen menjadi sebuah sistem
2. Teknik pengujian berbeda-beda sesuai dengan berbagai isi atau unit uji dalam waktu yang berbeda-beda pula bergantung padaa pengujian pada bagian mana yang dibutuhkan.
3. Pengujian dilakukan oleh pengembang perangkat lunak, dan jika untuk proyek besar, pengujian bisa dilakukan oleh tim uji yang tidak terkait dengan tim pengembang perangkat lunak (*independent test group* (ITG)).
4. Pengujian dan penirkutuan (*debugging*) merupakan aktivitas yang berbeda tetapi penirkutuan (*debugging*) harus diakomondasikan pada berbagai strategi pengujian.



2.2.6.3 *Back-Box Testing* (Pengujian Kotak Hitam)

Menurut Cholifah et al. (2018:207), “Metode *Blackbox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi”. Sedangkan menurut Rosa dan Shalahuddin (2018:275), “*Black-box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan”.

Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji coba yang dilakukan adalah :

1. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
2. Jika *user* memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalkan nama pemakai benar tapi kata sandi salah, atau sebaliknya atau keduanya salah.

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Pane et al. (2020:132), “Aplikasi adalah program yang dibuat dengan tujuan melaksanakan fungsi sesuai dengan kegunaan aplikasinya, penggunaanya dan jenis aplikasi itu sendiri. Sedang menurut Syifani dan Dores (2018:23), “Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari *user* (pengguna)”.

Berdasarkan dari pengertian diatas dapat disimpulkan bahwa, aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan



melaksanakan tugas khusus dari *user* (pengguna) dengan tujuan melaksanakan fungsi sesuai dengan kegunaan aplikasinya.

2.3.2 Pengertian Pengolahan Data

Haerani dan Robiyanto (2019:104), “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”. Sedangkan menurut Kaharu dan Sakina (2016:31), “Pengolahan data yaitu proses perhitungan/trasformasi data *input* menjadi informasi yang mudah dipahami atau sesuai yang diinginkan”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa, pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan dan mudah dipahami.

2.3.3 Pengertian Evaluasi

Setiawan (2018:269), ”Evaluasi dapat diartikan sebagai proses menentukan kriteria standar, melakukan pengukuran dan penilaian serta mengambil keputusan berdasarkan kriteria tersebut”. Sedangkan menurut Rukajat (2018:2), “Evaluasi adalah suatu proses penentuan keputusan tentang kualitas suatu objek atau aktivitas dengan melibatkan pertimbangan nilai berdasarkan data dan informasi yang dikumpulkan, dianalisis dan ditafsirkan secara sistematis”.

Dari pernyataan diatas dapat disimpulkan bahwa evaluasi adalah proses menentukan kriteria standar, melakukan pengukuran dan penilaian serta mengambil keputusan berdasarkan data dan informasi yang dikumpulkan dianalisis dan ditafsirkan secara sistematis.

2.3.4 Pengertian Beasiswa

Setiyowati et al. (2019:95), “Beasiswa adalah pemberian berupa bantuan keuangan yang diberikan kepada perorangan, mahasiswa atau pelajar yang digunakan demi keberlangsungan pendidikan yang ditempuh”. Sedangkan menurut Sulistyو dan Winiarti (2015:965), “Beasiswa merupakan pemberian



bantuan keuangan maupun pendidikan perorangan, mahasiswa atau pelajar yang digunakan demi keberlangsungan pendidikan yang ditempuh bagi yang memiliki prestasi dibidang akademik, non akademik dan kemampuan ekonominya lemah, serta telah memenuhi syarat-syarat yang ditentukan pihak pemberi beasiswa”.

Dari definisi diatas dapat disimpulkan bahwa beasiswa adalah pemberian bantuan berupa keuangan yang diberikan kepada perorangan yang digunakan demi keberlangsungan pendidikan yang ditempuh bagi yang memiliki prestasi dibidang akademik, non akademik dan kemampuan ekonominya lemah, serta telah memenuhi syarat-syarat yang ditentukan pihak pemberi beasiswa.

2.3.5 Pengertian Website

Haerani dan Robiyanto (2019:104), “*Website* adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses diseluruh dunia selama terkoneksi dengan jaringan internet”. Sedangkan menurut Azis et al. (2019:), “ *Website* merupakan layanan atau alat tukar menukar data, informasi yang menggunakan konsep *client-server* di mana antara pengguna dan administrator dapat saling memberikan data atau informasi yang dapat memudahkan keduanya.

Dari definisi diatas dapat disimpulkan bahwa, *website* merupakan halaman layanan atau alat tukar menukar data, informasi yang disediakan melalui jalur internet.

2.3.6 Pengertian CSR (*Corporate Social Responsibility*)

Said (2018:4), “CSR adalah kewajiban sosial swasta atau perusahaan kepada masyarakat dan pemerintah sebagai dampak dari ekspansi bisnisnya yang dimungkinkan telah mengganggu keseimbangan lingkungan dan sosial kemasyarakatan dimana mereka menjalankan aktivitasnya. Sedangkan menurut Elmayati (2016:9), “*Corporate Social Responsibility (CSR)* adalah komitmen perusahaan atau dunia bisnis untuk berkontribusi dengan pengembangan ekonomi yang berkelanjutan dengan memperhatikan tanggung jawab sosial.



Jadi dapat disimpulkan bahwa CSR adalah kewajiban sosial swasta atau perusahaan kepada masyarakat dan pemerintahan sebagai bentuk tanggung jawab perusahaan terhadap keseimbangan lingkungan dan sosial kemasyarakatan dimana mereka menjalankan aktivitasnya

2.3.7 Pengertian Aplikasi Pengolahan Data dan Evaluasi Penerima Program Beasiswa Ayo Sekolah Berbasis Website pada CSR (*Corporate Social Responsibility*) PT Bukit Asam, Tbk

Berdasarkan pengertian yang telah dijelaskan diatas dapat disimpulkan bahwa aplikasi pengolahan data dan evaluasi penerima program beasiswa ayo sekolah merupakan suatu perangkat lunak yang dibuat untuk mengolah data dan evaluasi penerima bantuan pendidikan ayo sekolah pada CSR (*Corporate Social Responsibility*) atau pertanggungjawaban sosial perusahaan PT Bukit Asam, Tbk di bidang pendidikan.

2.4 Teori Program

2.4.1 Basis Data (*Database*)

Anisya dan Allazi (2017:83), “Database adalah kumpulan data/infomasi yang teratur berdasarkan kriteria tertentu yang saling berhubungan”. Sedangkan menurut Swara dan Pebriadi (2016:30), “Basis data atau Database adalah kumpulan informasi yang disusun dan merupakan suatu kesatuan yang utuh yang disimpan di dalam perangkat lunak (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak.

Dari beberapa definisi di atas penulis menyimpulkan bahwa basis data (*database*) adalah kumpulan data/informasi yang saling berhubungan yang disimpan di dalam perangkat lunak (komputer) secara sistematis sehingga dapat diolah menggunakan perangkat lunak

2.4.1.1 Pengertian *MySQL*

Anisya dan Allazi (2017:84), “MySQL(bisa dibaca dengan mai-es-ki-el atau bisa juga mai-se-kuel) adalah suatu perangkat lunak database relasi (*Relational Database Management System* atau *DBMS*), seperti halnya ORACLE,



POSTGRESQL, MSSQL, dan sebagainya”. Sedangkan menurut Syifani dan Dores (2018:26), “MySQL adalah sebuah implementasi dari sistem manajemen basis data relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License)”.

Dari beberapa definisi di atas penulis menyimpulkan bahwa, MySQL adalah suatu perangkat lunak database relasi yang merupakan implementasi dari sistem manajemen basis data relasional (*RDBMS*) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*).

2.4.2 Web Editor

2.4.2.1 Pengertian *Sublime Text*

Syifani dan Dores (2018:25), “Sublime Text adalah aplikasi *editor* untuk kode dan teks yang dapat berjalan diberbagai *platform operating system* dengan menggunakan teknologi Phyton API. Sedangkan menurut Supono dan Putratama (2018:14), “*Sublime Text* merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-edit suatu aplikasi”.

Dari definisi diatas dapat disimpulkan bahwa, *sublime text* adalah perangkat lunak *text editor* yang digunakan untuk membuat atau meng-edit kode dan teks suatu aplikasi yang dapat berjalan diberbagai *platform operating system* dengan menggunakan teknologi Phyton API.

2.4.3 Bahasa Pemrograman

Azis et al. (2019:7), “ Bahasa pemrograman adalah suatu jenis aplikasi komputer yang digunakan dalam kegiatan membuat program/aplikasi komputer (*software*)”.

2.4.3.1 Pengertian HTML

Abdulloh (2016:2), “HTML singkatan dari *Hyper Text Markup Language*, yaitu skrip yang berupa tag-tag untuk membuat dan mengatur struktur *website*”. Sedangkan menurut Azis (2019:5), “HTML atau *Hyper Text Markup Language*



merupakan standar bahasa pemrograman semi terstruktur yang dibuat dalam bentuk tag-tag yang menyusun setiap elemen *website* yang dapat diakses atau ditampilkan menggunakan *browser*”.

Dari pengertian diatas dapat disimpulkan bahwa, HTML merupakan standar bahasa pemrograman semi terstruktur yang dibuat dalam bentuk tag-tag untuk membuat dan mengatur struktur *website* yang dapat diakses atau ditampilkan menggunakan *browser*.

2.4.3.2 Pengertian PHP

Abdulloh (2016:3), “PHP singkatan dari *Hypertext Prerocessor* yang merupakan *server-side programming*, yaitu Bahasa pemrograman yang diproses di sisi server”. Sedangkan menurut Anisya dan Allazi (2017:84), “*PHP: Hypertext Preprocessor* adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML”.

Dari pengertian diatas dapat disimpulkan bahwa, PHP adalah Bahasa pemrograman yang dapat ditanamkan atau disisipkan ke dalam HTML.

2.4.3.3 Pengertian Java Script

Azis (2019:10), “*JavaScript* adalah bahasa yang digunakan untuk membuat program yang digunakan agar dokumen HTML yang ditampilkan dalam browser menjadi lebih interaktif, tidak hanya indah saja”. Sedangkan menurut IpanRipai (2017:3) *JavaScript* adalah bahasa *script* yang ditempelkan pada kode HTML dan proses pada sisi klien, sehingga kemampuan dokumen HTML menjadi lebih luas”.

Dari pengertian diatas dapat disimpulkan bahwa, *JavaScript* adalah Bahasa script yang ditempelkan pada kode HTML dan proses pada sisi klien, sehingga kemampuan dokumen HTML menjadi lebih luas dan interaktif.



2.4.3.4 Pengertian CSS

Abdulloh (2016:2), “CSS singkatan dari *Cascading Style Sheets*, yaitu skrip yang digunakan untuk mengatur desain *website*”. Sedangkan menurut IpanRipai (2017:3), “*Cascading Style Sheet* (CSS) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur”.

Dari pengertian diatas dapat disimpulkan bahwa, CSS adalah skrip yang digunakan untuk mengatur desain *website* agar lebih terstruktur.

2.4.4 Web Server

Abdulloh (2018:5), “*Web Server* merupakan perangkat lunak yang berfungsi untuk menerima permintaan (*Request*) melalui protocol HTTP atau HTTPS dari client kemudian mengirimkan kembali dalam bentuk halaman-halaman web”. Sedangkan menurut Rusli et al. (2019:1), “*Web server* adalah sebuah program yang menggunakan *client/server* dan *World Wide Hypertext Transfer Protocol* (HTTP) dan melayani file yang membentuk halaman *web* untuk pengguna *web* atau memberikan konten atau layanan kepada pengguna akhir melalui internet”.

2.4.4.1 Pengertian XAMPP

Abdulloh (2016:7), “*XAMPP* adalah salah satu paket *installer* yang berisi *apache* yang merupakan *web server* tempat menyimpan file-file diperlukan *website*, *Phpmyadmin* sebagai aplikasi yang digunakan untuk perancangan *database* MySQL. Sedangkan menurut Nurcholish (2018:23), ”*XAMPP* adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program”.

Berdasarkan pengertian diatas dapat disimpulkan bahwa, *XAMPP* adalah perangkat lunak bebas yang mendukung banyak sistem operasi dan merupakan *web server* tempat menyimpan file-file diperlukan *website*.