

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian dengan tema yang sama sudah pernah dilakukan sebelumnya dengan judul “Implementasi E-Service Untuk Legalisir Ijazah dan Transkrip Nilai pada Universitas Budi Luhur” oleh Wahyu Pramusinto, Sri Mulyati, Safrina Amini pada tahun (2021). Penelitian ini didasari karena adanya permasalahan alumni dalam pengajuan legalisir yaitu proses legalisir yang dinilai tidaklah efisien dan menghabiskan banyak waktu, terutama untuk alumni yang sudah bekerja karena harus mendapatkan izin minimal 2 kali dari kantor untuk bisa melakukan legalisir. Terlebih lagi jika alumni telah tinggal di luar kota, atau bahkan di luar pulau Jawa. Tentu akan memerlukan biaya transportasi yang tidak sedikit. Untuk itu, disediakan alternatif yaitu pengimplementasian e-service berbasis website untuk layanan legalisir ijazah dan transkrip nilai, yang memudahkan alumni dalam pengajuan permohonan legalisir, serta memudahkan staff BAAK dalam menginformasikan status pengajuan legalisir dokumen akademik. Hasil dari penelitian ini ialah sebuah aplikasi berbasis website yang dapat digunakan oleh alumni untuk melegalisir ijazah dan transkrip nilai sebagai bagian dari peran Universitas Budi Luhur dalam melayani alumninya dimanapun alumninya berada. Sistem ini dibuat menggunakan Bahasa Pemrograman PHP dan database MariaDB. Model metode yang digunakan adalah model waterfall [4].

Penelitian lain menurut Wiwik Sri Wahyuni, Wagino, dan Dian Agustini (2023), dengan judul penelitian “Aplikasi Pendataan Alumni Dan Pelayanan Legalisir Online Pada Sma Negeri 1 Bati-Bati Kab. Tanah Laut Berbasis Web”. Aplikasi yang dirancang ini dikhususkan pada bagian Tata Usaha karena sistem yang sedang berjalan sekarang ini masih sistem manual yaitu pendataan alumni menggunakan aplikasi Microsoft Word dan file-file tersimpan dalam komputer tentang alumni belum tertata dengan baik dan data yang tersedia pada file-file hanya menyajikan data lulusan saja serta belum adanya sistem informasi yang lebih luas terkait informasi dari alumni. Bahasa pemrograman yang digunakan dalam pembuatan penelitian ini adalah PHP 5 dan MySQL. Hasil akhir dari penelitian ini adalah terciptanya aplikasi yang dapat mempermudah dalam proses pengolahan data, pelayanan,

mempermudah dalam penyimpanan, serta proses dalam pembuatan laporan [5].

Penelitian Irfan Syauqi, Sudargo, Ika Menarianti (2022) “Sistem Informasi Legalisir Online “Silegol” Untuk Mas Tahfidz Yanbu’ul Qur’an” dilatarbelakangi oleh sistematis legalisir ijazah pada MAS (Madrrasah Aliyah Swasta) Tahfidz Yanbu’ul Qur’an yang masih berjalan secara manual. Tersebarinya tempat tinggal alumni dari sekolah tersebut di berbagai kota dan pula tentunya menjadi hambatan tersendiri dalam pengajuan legalisir ijazah. Berdasarkan masalah itu maka dibuat layanan legalisir ijazah yang berguna untuk mempermudah alumni sebagai pemohon dan juga sekolah sebagai penyedia. SILEGOL memudahkan alumni yang berada jauh dari sekolah untuk mengajukan legalisir dari tempat tinggalnya tanpa harus datang ke sekolah secara langsung. Sekolah juga dimudahkan untuk mendapatkan informasi mengenai alumni dan memberikan akses untuk legalisir. Metode yang diterapkan dalam penelitian ini adalah metode waterfall dan menggunakan bahasa pemrograman PHP dan database SQL [6].

Penelitian lain dengan judul “Implementasi E-Legalisir Untuk Legalisir Ijazah dan Transkrip Online pada Fakultas Ilmu Komputer Universitas Mercu Buana” oleh Yudha Permana, Herry Derajad Wijaya (2020). Fakultas Ilmu Komputer Universitas Mercu Buana merupakan salah satu fakultas yang memiliki alumni yang banyak sehingga permintaan legalisir juga berbanding lurus dengan jumlah alumni. Sistem google form yang telah berjalan dinilai kurang efektif dalam pendataan dan pengajuan serta user interface yang kurang menarik. Terlebih memakan waktu untuk alumni dalam proses pengajuannya. Oleh karena itu, dilakukan penelitian yang menghasilkan sistem layanan legalisir online dengan tampilan user interface yang mudah dipahami. Metode pengembangan sistem yang akan digunakan adalah metode Extreme programming. Metode ini akan menjalankan secara urut tahapannya dari level perencanaan, level desain, level pengkodean dan level pengujian. Sistem ini dilakukan pengujian Blackbox Testing [7].

Untuk sistem informasi manajemen legalisir ijazah cukup banyak dilakukan oleh para peneliti sebelumnya, dengan berbagai model dan metode pilihan yang berbeda-beda serta fitur yang ditawarkan. Dengan ini, peneliti bermaksud untuk membuat sistem yang mana merupakan sistem dalam bidang layanan administratif legalisir ijazah dengan fitur berbeda atau tambahan yaitu pelayanan pengiriman ijazah melalui jasa ekspedisi dan transaksional pembayaran biaya legalisir. Sistem ini akan

memudahkan pemohon atau alumni untuk mengetahui status dari progress layanan legalisir. Sistem ini juga dikhususkan untuk mempermudah pemohon yang terkendala jarak jauh karena tidak perlu datang ke kampus berulang kali karena fitur layanan pengiriman paket lewat ekspedisi. Metode yang digunakan oleh peneliti adalah waterfall oleh Sommerville dan dengan bahasa pemrograman framework Laravel.

2.2 Landasan Teori

Landasan teori berisi hal-hal atau teori-teori yang berkaitan dengan permasalahan dan ruang lingkup permasalahan sebagai landasan dalam pembuatan laporan ini.

2.2.1 E-Service

Pengertian dari *e-service* sebagai penyediaan layanan melalui jaringan elektronik seperti internet. Selain itu, *e-service* sebagai: kunjungan awal dari homepage sampai layanan yang diminta komplit atau produk akhir telah disampaikan dan layak digunakan. *e-service* merupakan layanan online yang tersedia di Internet, dimana transaksi yang valid dari membeli dan menjual (pengadaan) adalah mungkin, yang bertentangan dengan website tradisional, dimana hanya informasi deskriptif yang tersedia, dan tidak ada transaksi online dimungkinkan.

2.2.2 Manajemen

Menurut Afandi (2018:1) Manajemen adalah proses kerja sama antar karyawan untuk mencapai tujuan organisasi sesuai dengan pelaksanaan fungsifungsi perencanaan, pengorganisasian, personalia, pengarahan, kepemimpinan, dan pengawasan[8]. Definisi manajemen menurut G. R. Terry (2018:2) yang 7 dialih bahasakan oleh R. Supomo dan Eti Nurhayati adalah “Manajemen adalah suatu proses yang khas yang terdiri atas tindakan-tindakan perencanaan, pengorganisasian, pengarahan dan pengendalian yang dilakukan untuk menentukan serta mencapai 18 sasaran yang telah ditentukan melalui pemanfaatan sumber daya manusia dan sumber-sumber lainnya[9].

2.2.1 Legalisasi

Legalisasi yang selanjutnya disebut legalisir adalah proses pengesahan atas salinan dokumen dengan tujuan agar dokumen tersebut sah di mata hukum. Umumnya dokumen resmi yang dilegalisir dapat

berupa ijazah, KTP, raport. Pengesahan dilakukan dengan cara pemberian cap stempel basah dan dibubuhi tanda tangan oleh petugas yang berwenang untuk memberikan bukti keaslian dokumen aslinya. Dokumen yang telah dilegalisir biasanya digunakan sebagai salah satu syarat administrasi pada saat-saat tertentu seperti pendaftaran sekolah ke jenjang berikutnya, daftar CPNS, melamar pekerjaan, atau keperluan lainnya. Menurut buku Aspek Hukum Peralihan Hak Milik Atas Satuan Rumah Susun oleh Dr. J. Andy Hartanto, S.H., M.H., I.r., M.M.T. (2018: 125), legalisasi adalah pengesahan yang dilakukan oleh Notaris terhadap akta dibawah tangan yang memberikan kepastian tentang tanggal penandatanganan, kebenaran dari pihak-pihak yang menandatangani, isi akta yang telah diketahui oleh para pihak. Legalisir berasal dari kata legal yang menurut KBBI berarti “sesuai dengan peraturan perundang-undangan atau hukum”. Dalam hal ini legalisir adalah proses pembubuhan cap stempel dan tanda tangan oleh pihak berwenang pada lembar fotokopi. Syarat untuk dapat melakukan legalisir adalah dengan membawa dokumen asli dan lembar fotokopian kemudian biaya legalisir. Biaya legalisir tiap tempat atau instansi berbeda-beda sesuai dengan kebijakan tempat masing-masing.

2.2.4 Website

Web merupakan sebuah sistem dengan informasi yang disajikan atau ditampilkan dalam bentuk teks, gambar, suara, dan sebagainya yang mana tersimpan dalam sebuah server web internet yang disajikan ke dalam bentuk hiperteks. Website adalah kumpulan halaman dalam suatu domain yang memuat mengenai berbagai informasi agar dapat dibaca dan dilihat oleh para pengguna internet melalui sebuah search engine atau mesin pencari. Informasi yang dapat disajikan dalam suatu website biasanya tentang konten gambar, ilustrasi, video, dan juga teks untuk berbagai macam keperluan.

Umumnya untuk tampilan awal suatu website dapat diakses melalui halaman utama atau yang disebut homepage dengan menggunakan suatu browser dengan cara menuliskan alamat URL secara lengkap dan tepat. Di dalam sebuah homepage atau halaman utama website juga memuat beberapa halaman web turunan yang terhubung satu dengan yang lainnya. Berikut website berdasarkan sifatnya:

1. Website dinamis

Disebut website dinamis karena kontennya yang dapat berubah-

ubah setiap saat. Faktor utama yang membuat website menjadi dinamis yaitu adanya Content Management System. Dengan adanya CMS, pihak administrator atau pemilik akses dapat mengupdate konten secara mudah[10].

2. Website Statis

Merupakan website yang isi kontennya jarang diubah karena memang tidak memerlukan perubahan yang sering. Interaksi Web dibagi ke dalam tiga langkah yaitu:

- a. Permintaan Pengguna mengirimkan permintaan ke server web, via halaman web yang ditampilkan pada browser web.
- b. Pemrosesan Server web menerima permintaan yang dikirimkan oleh pengguna kemudian memproses permintaan tersebut.
- c. Jawaban Browser menampilkan hasil dari permintaan pada jendela browser.

2.2.5 *Rekayasa Perangkat Lunak*

Rekayasa perangkat lunak adalah aplikasi yang berasal dari adanya pendekatan kuantifiabel, disiplin dan secara sistematis kepada proses, ekspansi dan pemeliharaan perangkat lunak yaitu aplikasi dari Rekayasa perangkat lunak itu sendiri. Definisi lain Rekayasa Perangkat Lunak adalah pembuatan dan penggunaan banyaknya prinsip suatu keahlian untuk membuat perangkat lunak yang bernilai ekonomis, handal dan berdaya guna pada mesin nyata.

Rekayasa Perangkat Lunak memiliki beberapa pokok Bahasa Penting yaitu :

- a. *Domain Engineering*
Kemampuan untuk memahami suatu permasalahan yang muncul dan dijadikan suatu proyek perangkat lunak.
- b. *Requirement Engineering*
Kemampuan untuk memahami kebutuhan seorang pengguna sekaligus dalam memecahkan permasalahan.
- c. *Software Design*
Kemampuan untuk memahami dan mengimplementasikan dalam perancangan perangkat lunak yang didalamnya terdapat aspek HCI (*Human Computer Interaction*).
- d. *Development*
Proses pengembangan yang melibatkan pembelajaran mengenai algoritma, Bahasa pemrograman yang di terapkan dan teknik yang didalamnya berkaitan dengan basis data dan sistem

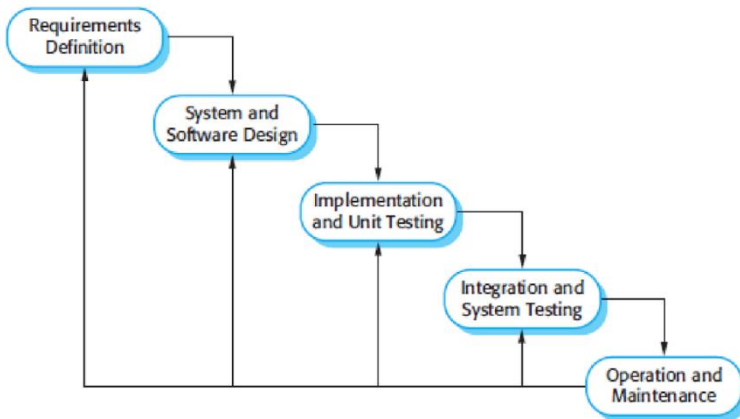
informasi.

e. *Maintenance*

Pemeliharaan dan pengembangan suatu sistem yang berguna untuk banyak orang[11].

1. Metode Pengembangan Sistem

Metode yang digunakan adalah metode *waterfall* menurut Ian Sommerville. Metode *waterfall* adalah suatu proses pengembangan perangkat lunak secara berurutan dan diibaratkan seperti air terjun karena kemajuannya dipandang terus mengalir kebawah melewati fase perencanaan, pemodelan, implementasi, dan pengujian[12]. Dalam proses implementasi metode *waterfall*, sebuah langkah dimulai dari tahapan pertama yang harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Penggunaan metode *waterfall* dapat memungkinkan adanya sedikit perubahan terhadap proyek yang sedang berlangsung. Metode ini mengusulkan sebuah pendekatan kepada pengembangan *software* yang sistematis dan sekuensial yang dimulai dari tingkat kemajuan sistem pada seluruh analisis, desain, kode, pengujian dan pemeliharaan. Metode pengembangan sistem *waterfall* dapat dilihat pada Gambar 2.1



Gambar 2.1 Metode Pengembangan Waterfall

Tahapan-tahapan dari metode waterfall menurut Sommerville,

a. Analisis Kebutuhan (*Requirements Definition*)

Tahapan metode waterfall yang pertama adalah mempersiapkan

dan menganalisa kebutuhan dari software yang akan dikerjakan. Informasi dan insight yang diperoleh dapat berupa dari hasil wawancara, survei, studi literatur, observasi, hingga diskusi.

- b. **Desain Aplikasi (*System and Software Design*)**
Tahap yang selanjutnya adalah pembuatan desain aplikasi sebelum masuk pada proses coding. Tujuan dari tahap ini, supaya mempunyai gambaran jelas mengenai tampilan dan antarmuka software yang kemudian akan dieksekusi oleh tim programmer.
- c. **Implementasi (*Implementation and Unit Testing*)**
Tahapan metode waterfall yang berikutnya adalah implementasi kode program dengan menggunakan berbagai tools dan bahasa pemrograman sesuai dengan kebutuhan tim dan perusahaan. Jadi, pada tahap implementasi ini lebih berfokus pada hal teknis, dimana hasil dari desain perangkat lunak akan diterjemahkan ke dalam bahasa pemrograman melalui tim programmer atau developer.
- d. **Integrasi dan Pengujian (*Integration and System Testing*)**
Tahap yang keempat, masuk dalam proses integrasi dan pengujian sistem. Pada tahap ini, akan dilakukan penggabungan modul yang sudah dibuat pada tahap sebelumnya. Setelah proses integrasi sistem telah selesai, berikutnya masuk pada pengujian modul. Yang bertujuan untuk mengetahui apakah perangkat lunak sudah sesuai dengan desain, dan fungsionalitas dari aplikasi apakah berjalan dengan baik atau tidak
- e. **Operasi dan Perbaikan (*Operation and Maintenance*)**
Tahapan metode *waterfall* yang terakhir adalah pengoperasian dan perbaikan dari aplikasi. Setelah dilakukan pengujian sistem, maka akan masuk pada tahap produk dan pemakaian perangkat lunak oleh pengguna (*user*). Untuk proses pemeliharaan, memungkinkan pengembang untuk melakukan perbaikan terhadap kesalahan yang ditemukan pada aplikasi setelah digunakan oleh user.

2. Metode Pengujian Sistem

Pengujian ini dilakukan pada sistem komputer (*computer-based system*) untuk mengurangi masalah yang berkaitan dengan antarmuka dan perancangan pada jalur penanganan terjadinya kesalahan antar sistem pada perangkat lunak [13]. Perangkat lunak dapat diuji dengan dua acara yaitu :

- a. *Black Box Testing* merupakan pengujian yang untuk menguji kelancaran pada program yang dikembangkan. Pengujian ini sangat penting supaya tidak terjadi kesalahan pada proses atau alur yang dibuat.
- b. *White Box Testing* merupakan pengujian mengenai detail perancangan yang menggunakan semua elemen program pada perangkat lunak seperti data internal, *loop*, logika keputusan, jalur). Data uji ini dikembangkan dengan mengetahui struktur internal dari perangkat lunak.

2.2.6 Pemrograman Berorientasi Objek


Metodologi pemrograman berorientasi objek merupakan pemrograman yang berorientasikan kepada objek, dimana semua data dan fungsi dibungkus dalam kelas-kelas atau objek-objek. Setiap objek dapat menerima pesan, memproses data, mengirim, menyimpan dan memanipulasi data. Beberapa objek berinteraksi dengan saling memberikan informasi satu terhadap yang lainnya.






Pada pemrograman berorientasi objek, UML digunakan untuk pemodelan sistem. UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya[14].

a. *Use Case Diagram*

Merupakan model UML gunanya untuk mengetahui hubungan antara user dengan sistem. Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan. Simbol use case dapat dilihat pada tabel 2.1.

Tabel 2. 1 Simbol Use Case Diagram


No.	Simbol	Nama	Keterangan
1.		<i>Use case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .


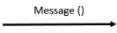



2.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case target</i> memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas

b. Sequence Diagram

Merupakan diagram yang digunakan untuk menampilkan dan menjelaskan interaksi antar objek pada sebuah sistem. *Sequence diagram* juga akan menampilkan sebuah perintah atau pesan yang dikirim. Simbol *Sequence Diagram* dapat dilihat pada tabel 2.2.

Tabel 2.2 Simbol Sequence Diagram







No.	Simbol	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmukayang saling berinteraksi.

2.		<i>Actor</i>	Menggambarkan <i>user</i> atau pengguna.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar <i>objek</i> yang memuat informasi – informasi tentang aktivitas yang terjadi.
4.		<i>Boundary</i>	Menggambarkan sebuah <i>form</i> .
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan Tabel.
6.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

c. ***Class Diagram***

Class diagram merupakan diagram yang digunakan untuk menampilkan beberapa kelas yang ada pada sebuah sistem yang dibangun, *class diagram* menunjukkan bagaimana class dalam sistem yang akan dibangun saling berkolaborasi untuk mencapai suatu tujuan (Wahyudi 2019). Simbol *Class diagram* dapat dilihat pada tabel 2.3.

Tabel 2. 3 Simbol Class Diagram

No.	Nama	Simbol	Keterangan
1.	<i>Generalization</i>		Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus)
2.	<i>Nary Association</i>		Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3.	<i>Class</i>		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4.	<i>Realization</i>		Operasi yang dilakukan oleh suatu objek
5.	<i>Dependency</i>		Relasi antarkelas dengan makna kebergantungan antarkelas
6.	<i>Association</i>		Diagram yang menghubungkan antara objek satu dengan objek lainnya.

2.2.7 Basis Data

Basis data merupakan kumpulan data yang saling terintegrasi serta berelasi secara *logic* serta dirancang untuk melengkapi informasi yang dibutuhkan[15].

A. *Database Management System (DBMS)*

Database Management System (DBMS) perangkat lunak computer yang berinteraksi dengan Pengguna, aplikasi lain, dan basis

data untuk mengambil dan menganalisis data. DBMS memungkinkan definisi, pembuatan, query, update, dan administrasi basis data[16]. DBMS menyediakan fasilitas antara lain:

1. *Data Definition Language (DDL)*

DDL memperbolehkan pengguna untuk mendeskripsikan *database*, misalnya merinci tipe dan batasan data yang akan disimpan dalam *database*. Terdapat tiga perintah penting dalam DDL, yaitu :

- a. *CREATE* adalah perintah yang digunakan untuk membuat, termasuk didalamnya membuat *database* baru, tabel baru, *view* baru, dan kolom baru. Contoh : *CREATE DATABASE* nama *database*.
- b. *ALTER* berfungsi untuk mengubah struktur tabel yang telah dibuat. Mencakup didalamnya mengubah nama tabel, menambah kolom, mengubah kolom, menghapus kolom, dan memberikan atribut pada kolom. Contoh : *Alter Table* nama tabel *ADD* nama kolom *datatype*.
- c. *DROP* perintah *DROP* berfungsi untuk menghapus *database* atau tabel. Contoh : *DROP DATABASE* nama *database*.

2. *Data Manipulation Language (DML)*

DML memperbolehkan pengguna untuk memanipulasi data dalam *database* yang telah dibuat. Terdapat 4 (empat) perintah penting dalam DML, yaitu :

- a. *INSERT* perintah ini digunakan untuk memasukkan data baru ke dalam sebuah tabel. Perintah ini tentu saja bisa dijalankan ketika *database* dan tabel sudah dibuat. Contoh: *INSERT INTO* nama tabel *VALUES* (data1, data2, dst..)
- b. *SELECT* perintah ini digunakan untuk mengambil dan menampilkan data dari tabel atau bahkan dari beberapa tabel dengan penggunaan relasi. Contoh : *SELECT* nama_kolom1, nama_kolom2 *FROM* nama tabel.
- c. *UPDATE* perintah ini digunakan untuk memperbaharui data pada sebuah tabel. Contoh : *UPDATE* nama_tabel *SET* kolom1=data1, kolom2=data2,.. *WHERE* kolom=data.
- d. *DELETE* perintah ini digunakan untuk menghapus data dari sebuah tabel. Contoh : *DELETE FROM* nama tabel *WHERE* kolom=data.



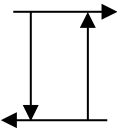
B. MySQL

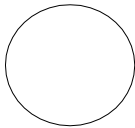
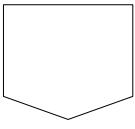
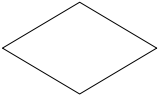
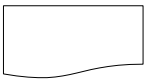
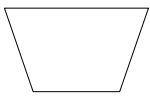
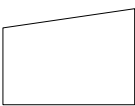
MySQL merupakan database server yang terkenal dan banyak digunakan untuk membangun sebuah aplikasi web yang databasenya digunakan sebagai sumber pengolahan data[9].

C. Flowchart

Flowchart atau biasa disebut dengan diagram alir merupakan suatu jenis diagram yang mempresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem. Flowchart sebagai bukti dokumentasi untuk menjelaskan gambaran yang logis pada sebuah sistem yang dibangun. Untuk menghubungkan satu proses ke proses selanjutnya dapat digambarkan dengan menggunakan garis penghubung [17]. Simbol Flowchart dapat dilihat pada table 2.4

Tabel 2. 4 Simbol Flowchart

No.	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Memulai dan mengakhiri suatu program
2.		<i>Input/Output</i>	Memasukan data maupun menunjukkan hasil dari suatu <i>process</i> tanpa tergantung dengan jenis peralatannya
3.		<i>Flow</i>	Menghubungkan antara simbol satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses. Simbol arus ini sering disebut juga dengan <i>connecting line</i> .

4.		<i>On-Page Connector Symbol</i>	Simbol penghubung yang digunakan untuk keluar- masuk atau penyambungan proses dalam halaman yang sama.
5.		<i>Off-Page Connector Symbol</i>	Simbol penghubung yang digunakan untuk keluar- masuk atau penyambungan proses dalam halaman atau lembar yang berbeda.
6.		<i>Decision</i>	Memilih proses berdasarkan kondisi yang ada.
7.		<i>Document</i>	Merupakan simbol untuk data yang terbentuk informasi.
8.		<i>Manual Operation</i>	Menunjukkan pengolahan yang tidak dilakukan oleh komputer/pc.
9.		<i>Manual Input</i>	Memasukan data secara manual <i>on-line</i> keyboard.