**BAB II**

**LANDASAN TEORI**

## **2. 1 Tinjauan Studi**

Penelitian tentang Harga Bibit Cengkeh Menggunakan Algoritma *Linier Regresi* telah dilakukan oleh beberapa peneliti terdahulu antara lain:

1. Nur Iksan, Yogi Pratama Putrab , Erika Devi Udayanti, 2018, dengan judul penelitian Nur Iksan, Yogi Pratama Putrab , Erika Devi Udayanti” Pada penelitian ini regresi linier diterapkan pada kasus prediksi penyediaan sparepart dengan tujuan untuk memperkirakan jumlah stok yang harus tersedia pada periode tertentu untuk masing-masing sparepart seperti kampas kopling, spion, ban motor, aki motor, dan kampas rem. Evaluasi prediksi dilakukan dengan MAP, MSE dan MAPE untuk tiap perhitungan sparepart sehingga dapat diketahui tingkat akurasi hasil dari prediksi tersebut
2. Rival Zunaidhi (2017), dengan judul penelitian “Aplikasi Peramalan Penjualan Menggunakan *Regresi Linier*” Peramalan penjualan merupakan hal yang paling utama untuk menentukan berapa banyak sebuah perusahaan memproduksi produk-produk mereka. Dengan peramalan tersebut melihat peluang penjualan yang ada pada pasar dengan cara memprediksi hasil yang akan datang berdasarkan hasil dari data penjualan yang telah lalu. Peramalan penjualan dapat dilakukan dengan berbagai cara dan metode peramalan. Salah satu metode yang digunakan dalam penelitian ini adalah metode regresi linier dengan model Time Series dengan menggunakan bahasa pemrograman visual basic 6.0. Regresi linier merupakan teknik / metode yang banyak digunakan dalam peramalan penjualan karena telah teruji ketepatan dalam peramalan. Dari aplikasi peramalan penjualan ini akan didapatkan hasil dari peramalan penjualan pada hari yang akan datang dan hasil standart error estimasi peramalan. Perhitungan dari hasil tersebut akan dibandingkan dengan perhitungan dalam microsoft excel. Terdapat pula pada aplikasi ini hasil yang lain berupa pelaporan penjualan tiap bulan dan laporan delivery order yang dapat langsung di cetak dan di export ke microsoft excel.

7

1. Penelitian ini yang dilakukan oleh Khoiriya Latifah, Ika Novita Dewi, Richardus Anggi P, Ika Amelia P, (2017), yang berjudul Neural Network Dan Svm Untuk Prediksi Jumlah Pemohon Passport Pada Kantor Imigrasi Kota Semarang, Algorithma prediksi atau peramalan yang digunakan pada penelitian ini adalah metode SVM dan Neural Network untuk melakukan prediksi jumlah pembuat paspor atau jumlah pengunjung kantor Imigrasi Kota Semarang. Support Vector Machine adalah teknik yang digunakan untuk melakukan prediksi dengan cara melakukan klasifikasi dan regresi. Tingkat akurasi menggunakan RMSE untuk menentukan jumlah pengunjung dengan metode SVM adalah sebesar 0.001. Komparasi dilakukan dengan metode Neural Network. Neural Netowrk adalah algorithma yang mampu menirukan cara berpikir manusia dengan melakukan komputasi berbasis kecerdasan komputasional. Neural Network adalah metode untuk memodelkan, memprediksikan, mendeteksi kesalahan dan mengontrol sistem yang memerlukan pendekatan desain dengan kecerdasan tiruan komputasional. Tingkat akurasi perhitungan jumlah pengunjung kantor Imigrasi dengan Neural Network Backpropagation dengan menggunakan RMSE adalah sebesar 0.011. Sehingga perbandingan tingkat akurasi antara SVM 0.001 lebih baik daripada menggunkan BPNN dengan tingkat akurasi 0.011.

## **Tinjauan Pustaka**

### Paspor

Menurut Undang-Undang No. 9 Tahun 1992 tentang Keimigrasian, Paspor atau Surat Perjalanan Republik Indonesia (SPRI) adalah dokumen resmi yang dikeluarkan oleh pejabat yang berwenang yang memuat identitas diri pemegangnya dan berlaku untuk melakukan perjalanan antar Negara. Paspor Republik Indonesia adalah dokumen perjalanan yang diterbitkan oleh Direktorat Jenderal Imigrasi, Kementerian Hukum dan Hak Asasi Manusia, dan perwakilan Republik Indonesia di luar negeri. Paspor ini hanya diberikan kepada Warga Negara Indonesia.

Paspor biasanya diperlukan untuk perjalanan internasional karena harus ditunjukkan ketika memasuki perbatasan suatu negara, walaupun di negara tertentu ada beberapa perjanjian dimana warga suatu negara tertentu dapat memasuki negara lain dengan dokumen selain paspor. Paspor akan diberi cap (stempel) atau disegel dengan visa yang dilakukan oleh petugas negara tempat kedatangan. Beberapa macam Paspor Indonesia, yang masing-masing dikeluarkan oleh lembaga yang berbeda-beda.

1. Paspor umum (bersampul hijau, ada dua jenis yang berbeda jumlah halamannya), dikeluarkan oleh Direktorat Jenderal Imigrasi, Kementerian Hukum dan Hak Asasi Manusia
2. Paspor dinas/kedinasan (bersampul biru), dikeluarkan oleh Kementerian Luar Negeri.
3. Paspor diplomatik (bersampul hitam), dikeluarkan oleh Kementerian Luar Negeri. (http://id.wikipedia.org/wiki/Paspor\_Indonesia) Oktober 2018

Berikut sampel dataset Laporan Pemakaian, Permintaan dan pemohon pasport tahun 2018.

**Tabel 2.1** Data Permintaan dan Pemohon Paspor tahun 2018

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Bulan** | **Pemohon (X)** | **Permintaan (Y)** |
| 1 | Januari | 819 | 1.000 |
| 2 | Februari | 843 | 1.000 |
| 3 | Maret | 768 | 500 |
| 4 | April | 360 | 0 |
| 5 | Mei | 265 | 1.000 |
| 6 | Juni | 142 | 0 |
| 7 | Juli | 398 | 0 |
| 8 | Agustus | 434 | 1.000 |
| 9 | September | 434 | 3.500 |
| 10 | Oktober | 437 | 0 |
| 11 | November | 479 | 0 |
| 12 | Desember | 342 | 0 |

(Sumber : Kantor Imigrasi Kelas I Gorontalo, 2019 )

### 2.2.2 Data Mining

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai basis data besar (Turban,dkk.2005:3).

Istilah data mining dan knowledge discovery in databases (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lain. Dan salah satu tahapan dalam keseluruhan proses KDD adalah *data mining* . Proses KDD(*Knowlegge discovery in databases*) secara garis besar dapat dijelaskan sebagai berikut (Kusrini, 2009:7).



**Gambar 2.1** Tahapan *Knowledge Discovery in Databases*

1. Data *Selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam *knowledge data discovery* (KDD) dimulai. Data hasil seleksi yang akan digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

*2. Preprocessing* atau *Cleaning*

Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses *cleaning* pada data yang menjadi fokus *knowledge data discovery*. Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak juga dilakukan proses *enrichment*, yaitu proses memperkaya data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi.

3. *Transformation*

*Coding* adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses *coding* dalam *knowledge data discovery* merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. Data mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat tergantung pada tujuan dan proses KDD secara keseluruhan.

5. *Interpretation* atau *evaluation*

Pola informasi yang dihasilkan dari proses *data mining* perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut *interpretation*. Tahap ini mencakup pemeriksaan apakah pola informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada pada sebelumnya.

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu (Larose ,2005) :

1. Deskripsi

Deskripsi adalah menggambarkan pola dan kecenderungan yang terdapat dalam data yang memungkinkan memberikan penjelasan dari suatu pola atau kecenderungan tersebut.

1. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih kearah numeric dari pada kearah kategori. Model dibangun menggunakan record lengkap yang menyediakan nilai variabel target sebagai nilai prediksi.

3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, akan tetapi dalam prediksi nilai dari hasil akan ada di masa mendatang.

4. Klasifikasi

Klasifikasi adalah proses untuk menemukan model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep dengan tujuan memprediksikan kelas untuk data yang tidak diketahui kelasnya.

5. Pengklusteran

Pengklusteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. Kluster adalah kumpulan *record* yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan *record-record* dalam kluster lain.

1. Asosiasi

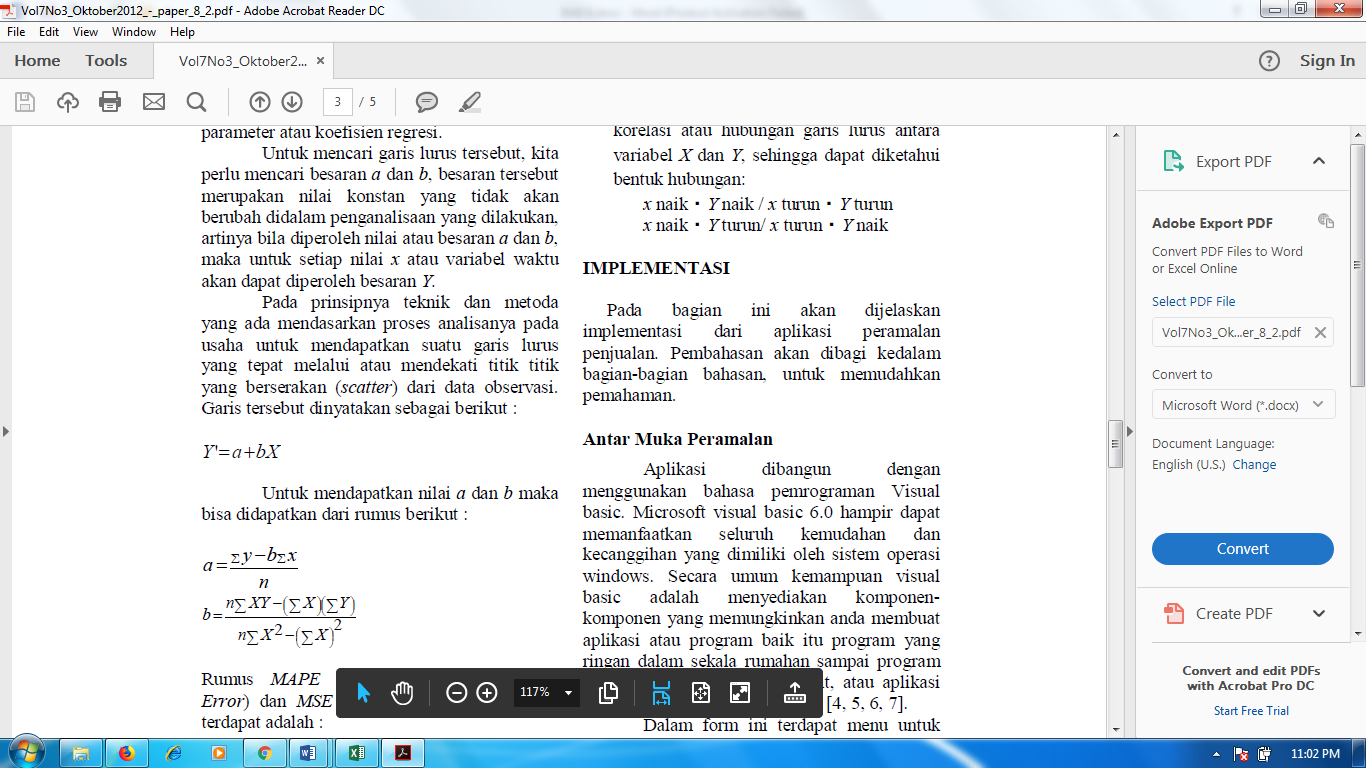
Asosiasi dalam *data mining* adalah menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih umum disebut analisis keranjang belanja.

### *Regresi Linear* Sederhana

Pola yang ditunjukkan dengan analisa regresi yang sederhana mengasumsikan bahwa hubungan diantara 2 variabel dapat dinyatakan dengan suatu garis lurus. Notasi regresi sederhana yang merupakan pola garis lurus itu menurut Sofyan Assauri (l,h.35) dinyatakan sebagai berikut.

*Y = a + bX*………………………………………………………………(2.1)

Dimana *Y* adalah variabel yang diramalkan, *x* adalah variabel waktu, serta a dan b adalah parameter atau koefisien regresi. Untuk mencari garis lurus tersebut, kita perlu mencari besaran *a* dan *b*, besaran tersebut merupakan nilai konstan yang tidak akan berubah di dalam penganalisaan yang dilakukan, artinya bila diperoleh nilai atau besaran *a* dan *b*, maka untuk setiap nilai *x* atau variabel waktu akan dapat diperoleh besaran *Y*. Untuk mendapatkan nilai *a* dan *b* maka bisa didapatkan dari rumus berikut :



……………………… …(2.2)

………………………….(2.3)

## **Contoh Penerapan Algoritma Linear Regresi**

## Seorang Engineer ingin mempelajari Hubungan antara Suhu Ruangan dengan Jumlah Cacat yang diakibatkannya, sehingga dapat memprediksi atau meramalkan jumlah cacat produksi jika suhu ruangan tersebut tidak terkendali. Engineer tersebut kemudian mengambil data selama 30 hari terhadap rata-rata (mean) suhu ruangan dan Jumlah Cacat Produksi.

**Penyelesaian**

Penyelesaiannya mengikuti Langkah-langkah dalam Analisis Regresi Linear Sederhana adalah sebagai berikut :

**Langkah 1 : Penentuan Tujuan**

 Memprediksi Jumlah Cacat Produksi jika suhu ruangan tidak terkendali

**Langkah 2 : Identifikasikan Variabel Penyebab dan Akibat**

**Varibel Faktor Penyebab (X)** : Suhu Ruangan,  
**Variabel Akibat (Y)**: Jumlah Cacat Produksi

**Langkah 3 : Pengumpulan Data**

Berikut ini adalah data yang berhasil dikumpulkan selama 30 hari (berbentuk tabel) :

**Tabel 2.2** Data Training

|  |  |  |
| --- | --- | --- |
| **Tanggal** | **Rata-rata Suhu Ruangan** | **Jumlah Cacat** |
| 1 | 24 | 10 |
| 2 | 22 | 5 |
| 3 | 21 | 6 |
| 4 | 20 | 3 |
| 5 | 22 | 6 |
| 6 | 19 | 4 |
| 7 | 20 | 5 |
| 8 | 23 | 9 |
| 9 | 24 | 11 |
| 10 | 25 | 13 |
| 11 | 21 | 7 |
| 12 | 20 | 4 |
| 13 | 20 | 6 |
| 14 | 19 | 3 |
| 15 | 25 | 12 |
| 16 | 27 | 13 |
| 17 | 28 | 16 |
| 18 | 25 | 12 |
| 19 | 26 | 14 |
| 20 | 24 | 12 |
| 21 | 27 | 16 |
| 22 | 23 | 9 |
| 23 | 24 | 13 |
| 24 | 23 | 11 |
| 25 | 22 | 7 |
| 26 | 21 | 5 |
| 27 | 26 | 12 |
| 28 | 25 | 11 |
| 29 | 26 | 13 |
| 30 | 27 | 14 |

**Langkah 4 : Hitung X², Y², XY dan total dari masing-masingnya**

Berikut ini adalah tabel yang telah dilakukan perhitungan X², Y², XY dan totalnya

**Tabel 2.3** perhitungan X², Y², XY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tanggal** | **Rata-rata Suhu Ruangan (X)** | **Jumlah Cacat        (Y)** | **X2** | **Y2** | **XY** |
| 1 | 24 | 10 | 576 | 100 | 240 |
| 2 | 22 | 5 | 484 | 25 | 110 |
| 3 | 21 | 6 | 441 | 36 | 126 |
| 4 | 20 | 3 | 400 | 9 | 60 |
| 5 | 22 | 6 | 484 | 36 | 132 |
| 6 | 19 | 4 | 361 | 16 | 76 |
| 7 | 20 | 5 | 400 | 25 | 100 |
| 8 | 23 | 9 | 529 | 81 | 207 |
| 9 | 24 | 11 | 576 | 121 | 264 |
| 10 | 25 | 13 | 625 | 169 | 325 |
| 11 | 21 | 7 | 441 | 49 | 147 |
| 12 | 20 | 4 | 400 | 16 | 80 |
| 13 | 20 | 6 | 400 | 36 | 120 |
| 14 | 19 | 3 | 361 | 9 | 57 |
| 15 | 25 | 12 | 625 | 144 | 300 |
| 16 | 27 | 13 | 729 | 169 | 351 |
| 17 | 28 | 16 | 784 | 256 | 448 |
| 18 | 25 | 12 | 625 | 144 | 300 |
| 19 | 26 | 14 | 676 | 196 | 364 |
| 20 | 24 | 12 | 576 | 144 | 288 |
| 21 | 27 | 16 | 729 | 256 | 432 |
| 22 | 23 | 9 | 529 | 81 | 207 |
| 23 | 24 | 13 | 576 | 169 | 312 |
| 24 | 23 | 11 | 529 | 121 | 253 |
| 25 | 22 | 7 | 484 | 49 | 154 |
| 26 | 21 | 5 | 441 | 25 | 105 |
| 27 | 26 | 12 | 676 | 144 | 312 |
| 28 | 25 | 11 | 625 | 121 | 275 |
| 29 | 26 | 13 | 676 | 169 | 338 |
| 30 | 27 | 14 | 729 | 196 | 378 |
| Total (Σ) | 699 | 282 | 16487 | 3112 | 6861 |

**Langkah 5 : Hitung a dan b berdasarkan rumus Regresi Linear Sederhana**

Menghitung Konstanta (a) :

a =   (Σy) (Σx²) – (Σx) (Σxy)  
.               n(Σx²) – (Σx)²

a = (282) (16.487) – (699) (6.861)  
                30 (16.487) – (699)²

a = -24,38

 Menghitung Koefisien Regresi (b)

b =   n(Σxy) – (Σx) (Σy)  
.           n(Σx²) – (Σx)²

b = 30 (6.861) – (699) (282)  
.          30 (16.487) – (699)²

b = 1,45

**Langkah 6 : Buat Model Persamaan Regresi**

Y = a + bX  
Y = -24,38 + 1,45X

**Langkah 7 : Lakukan Prediksi atau Peramalan terhadap Variabel Faktor Penyebab atau Variabel Akibat**

I. Prediksikan Jumlah Cacat Produksi jika suhu dalam keadaan tinggi (Variabel X), contohnya : 30°C

Y = -24,38 + 1,45 (30)  
Y = 19,12

Jadi Jika Suhu ruangan mencapai 30°C, maka akan diprediksikan akan terdapat **19,12 unit cacat** yang dihasilkan oleh produksi.

Jika Cacat Produksi (Variabel Y) yang ditargetkan hanya boleh 4 unit, maka berapakah suhu ruangan yang diperlukan untuk mencapai target tersebut ?

4 = -24,38 + 1,45X  
1,45X = 4 + 24,38  
X = 28,38 / 1,45  
X = 19,57

Jadi Prediksi Suhu Ruangan yang paling sesuai untuk mencapai target Cacat Produksi adalah sekitar **19,57°C**

## **Siklus Pengembangan Sistem**

Pengembangan sistem informasi yang berbasis komputer dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu berbulan-bulan bahkan bertahun tahun untuk menyelesaikannya. Proses pengembangan sistem melewati beberapa tahapan dari mulai sistem itu direncanakan sampai dengan sistem tersebut diterapkan, dioperasikan dan dipelihara. Bila operasi sistem yang sudah dikembangkan masih timbul kembali permasalahan-permasalahan yang kritis serta tidak dapat diatasi dalam tahap pemeliharaan sistem, maka perlu dikembangkan kembali suatu sistem untuk mengatasinya dan proses ini kembali ke tahap yang pertama, yaitu tahap perencanaan sistem. Siklus ini disebut dengan siklus hidup suatu sistem (*systems life cycle*). Siklus hidup pengembangan sistem dengan langkah-langkah utamanya yang akan digunakan adalah sebagai berikut : (Jogiyanto HM, 2005 : 52)

**Kebijakan dan perencanaan sistem**

**Analisis sistem**

**Desain (perancangan) sistem secara umum**

Awal proyek sistem

Pengembangansistem

**Seleksi sistem**

**Implementasi ( penerapan) sistem**

**Perawatan sistem**

Manajemen sistem

**Gambar 2.2**Siklus Hidup Pengembangan Sistem.

### 2.2.6 Analisa Sistem

Jogiyanto HM (2005:129) mendefinisikan analisa sistem sebagai berikut: ”Analisa sistem (*systems analysis*) sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya”.

Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan di dalam tahap ini akan menyebabkan juga kesalahan ditahap selanjutnya. Tahap analisa sistem mencakup studi kelayakan dan analisis kebutuhan.

Di dalam tahap analisis sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analisis sistem, yaitu sebagai berikut :

1. *Identify,* yaitu mengidentifikasi masalah.

Mengidentifikasi (mengenal) masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah (*Problems*) dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan.

1. *Understand,* yaitu memahami kerja dari sistem yang ada.

Langkah kedua dari tahap analisis sistem adalah memahami kerja dari sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang ada beroperasi. Untuk mempelajari operasi dari sistem ini diperlukan data yang dapat diperoleh dengan cara melakukan penelitian.

1. *Analyze,* yaitu menganalisis sistem tanpa *report*

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang telah dilakukan

1. *Report,* yaitu membuat laporan hasil analisis

Tujuan utama dari pembuatan laporan hasil analisis :

1. Pelaporan bahwa analisis telah selesai dilakukan
2. Meluruskan kesalahan-pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analisis sistem tetapi tidak sesuai menurut manajemen.

### 2.2.7 Desain Sistem

Setelah tahap analisis sistem selesai dilakukan, maka analis sistem telah mendapatkan gambaran dengan jelas apa yang harus dikerjakan. Tiba waktunya sekarang bagi analis sistem untuk memikirkan bagaimana membentuk sistem tersebut. Tahap ini disebut dengan desain sistem (*system design*).

Menurut Robert J.Verzello dan John Reuter, dalam Jogiyanto HM (2005 : 196) desain sistem adalah tahap setelah analisis dari siklus pengembangan sistem; pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi menggambarkan bagaimana suatu sistem dibentuk.

Demikian pula Menurut John Burch dan Gary Grudnitski, dalam Jogiyanto HM (2005 : 196) desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi.

Tahap desain sistem mempunyai dua tujuan utama yaitu :

1. Untuk memenuhi kebutuhan kepada pemakai sistem.
2. Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknis lainnya.

Desain sistem dapat dibagi dalam dua bagian yaitu desain sistem secara umum (*general systems design*) dan desain sistem secara terinci (*detailed system design*).

1. Desain sistem secara umum(*General System Design*).

Tujuan dari desain sistem secara umum adalah untuk memberikan gambaran secara umum kepada *user* tentang sistem yang baru, yang mana merupakan persiapan dari desain sistem secara rinci. Desain secara umum dilakukan oleh analis sistem untuk mengidentifikasikan komponen-komponen sistem informasi yang akan didesain secara rinci oleh pemrogram komputer dan ahli teknik lainnya.

Pada tahap ini, komponen-komponen sistem informasi dirancang dengan tujuan untuk dikomunikasikan kepada *user*. Komponen sistem informasi yang didesain adalah model, *input*, *database*, *output*, teknologi dan kontrol.

1. Desain sistem secara rinci (*detailed system design*).
2. Desain *input* terinci

Masukan merupakan awal dimulainya proses informasi. Bahan mentah dari informasi adalah data yang terjadi dari transaksi-transaksi yang dilakukanoleh organisasi. Data hasil transaksi merupakan masukan untuk sistem informasi. Hasil dari sistem informasi tidak lepas dari data yang dimasukkan. Desain *input* terinci dimulai dari desain dokumen dasar sebagai penangkap *input* yang pertama kali. Jika dokumen dasar tidak didesain dengan baik, kemungkinan *input* yang tercatat dapat salah bahkan kurang.

Fungsi dokumen dasar dalam penanganan arus data :

1). Dapat menunjukkan macam dari data yang harus dikumpulkan.

2). Data dapat dicatat dengan jelas, konsisten dan akurat.

3). Dapat mendorong lengkapnya data disebabkan data yang dibutuhkan

disebutkan satu persatu di dalam dokumen dasarnya.

1. Desain *output* terinci.

Desain *output* terinci dimaksudkan untuk mengetahui bagaimana dan seperti apa bentuk *output*-*output* dari sistem yang baru. Desain *Output* Terinci terbagi atas dua yaitu desain *output* berbentuk laporan dimedia kertas dan desain *output* dalam bentuk dialog pada layar terminal.

1. Desain *Output* Dalam Bentuk laporan

Desain ini dimaksudkan untuk menghasilkan *output* dalam bentuk laporan dimedia kertas. Bentuk laporan yang paling banyak digunakan adalah dalam bentuk tabel dan berbentuk grafik atau bagan.

1. Desain *Output* Dalam Bentuk Dialog Layar Terminal

Desain ini merupakan rancang bangun dari percakapan antara pemakai sistem (*user*) dengan komputer. Percakapan ini dapat terdiri dari proses memasukkan data ke sistem, menampilkan *output* informasi kepada *user* atau keduannya.

Beberapa strategi dalam membuat layar dialog terminal :

1. Dialog pertanyaan / jawaban.
2. Menu.

Menu banyak digunakan karena merupakan jalur pemakai yang mudah dipahami dan mudah digunakan. Menu berisi beberapa alternatif atau option atau pilihan yang disajikan kepada *user*. Pilihan menu akan lebih baik bila dikelompokkan sesuai fungsinya.

1. Desain *database* terinci.

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan disimpanan luar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya. *Database* merupakan salah satu komponen yang penting disistem informasi karena berfungsi sebagai basis penyedia informasi bagi para pemakainya. Penerapan *database* dalam sistem informasi disebut *database system*.

Sistem basis data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi. Dalam sistem basis data, tiap-tiap orang atau bagian dapat memandang *database* dari beberapa sudut pandang yang bebeda.

Pada tahap ini, desain *database* dimaksudkan untuk mendefinisikan isi atau struktur dari tiap-tiap file yang telah diidentifikasikan didesain secara umum.

1. Desain teknologi.

Tahap desain teknologi terbagi atas dua yaitu desain teknologi secara umum dan terinci. Pada tahap ini kita menentukan teknologi yang akan dipergunakan dalam menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi yang dimaksud meliputi :

1. Perangkat Keras (*hardware*), yang terdiri dari alat masukan, alat pemroses, alat *output* dan simpanan luar.
2. Perangkat Lunak (*software*), yang terdiri dari perangkat lunak sistem operasi (*operating system*), perangkat lunak bahasa (*language software*) dan perangkat lunak (*application software*).
3. Sumber Daya Manusia (*brainware*), misalnya operator komputer, pemrogram, spesialis telekomunikasi, sistem analis dan sebagainya.

Desain teknologi sangat diperlukan pada tahap implementasi dan pengujian untuk membuktikan bahwa sistem dapat berjalan secara semestinya.

1. Desain model.

Tahap desain model terbagi menjadi dua yaitu desain model secara umum dan terinci. Tahap desain model secara umum berupa desain sistem secara fisik dan logika. Desain fisik dapat digambarkan dengan bagan alir dokumen.

Desain secara logika digambarkan dengan diagram arus data (DAD). Pada tahap desain model terinci, model akan mendefinisikan secara rinci urutan-urutan langkah dari masing-masing proses yang digambarkan di DAD. Urutan langkah proses ini diwakili oleh suatu program komputer.

### 2.2.8 Perancangan Konseptual

Perancangan konseptual sering kali disebut dengan perancangan logis. Pada perancangan ini kebutuhan pemakai dan pemecahan masalah yang teridentifikasi selama tahap analisis sistem mulai dibuat untuk di implementasikan. Ada tiga langkah penting yang dilakukan dalam perancangan konseptual, yaitu evaluasi alternatif rancangan, penyiapan spesifikasi rancangan dan penyiapan laporan rancangan sistem secara konseptual.

Menurut Romney, Seinbart dan Cushing, 1997 dalam Abdul Kadir (2003 : 407) evaluasi yang dilakukan mengandung hal-hal berikut :

1. Bagaimana alternatif-alternatif tersebut memenuhi sasaran sistem dan organisasi dengan baik?.
2. Bagaimana alternatif-alternatif tersebut memenuhi kebutuhan pemakai dengan baik?.
3. Apakah alternatif-alternatif tersebut layak secara ekonomi?.
4. Apa saja keuntungan dan kerugian masing-masing?

Setelah alternatif rancangan dipilih, tahap selanjutnya adalah penyiapan spesifikasi rancangan yang elemen-elemen sebagai berikut :

* 1. Keluaran

Rancangan laporan mencakup frekuensi laporan (harian, mingguan, dsb), isi laporan, bentuk laporan dan laporan cukup ditampilkan pada layar atau perlu dicetak.

* 1. Penyimpan Data

Dalam hal ini, semua data yang diperlukan untuk membentuk laporan ditentukan lebih detail, termasuk ukuran data dan letaknya dalam berkas.

* 1. Masukan

Rancangan masukan meliputi data yang perlu dimasukkan kedalam sistem.

* 1. Prosedur Pemrosesan dan Operasi

Rancangan ini menjelaskan bagaimana data masukan diproses dan disimpan dalam rangka untuk menghasilkan laporan.

Langkah berikutnya adalah menyiapkan laporan rancangan sistem konseptual. Berdasarkan laporan inilah, perancangan sistem secara fisik dibuat.

### 2.2.9 Perancangan Fisik

Pada perancangan ini, rancangan yang masih bersifat konsep diterjemahkan dalam bentuk fisik sehingga terbentuk spesifikasi lengkap tentang modul sistem dan antarmuka antar modul serta rancangan basis data secara fisik.

Beberapa hasil akhir setelah tahap perancangan fisik berakhir :

1. Rancangan keluaran

Rancangan keluaran berupa bentuk laporan dan rancangan dokumen.

1. Rancangan masukan

Rancangan masukan berupa rancangan layar untuk pemasukan data.

1. Rancangan antarmuka pemakai dan sistem.

Rancangan ini berupa rancangan interaksi antar pemakai dan sistem, misalnya berupa menu, icon dan lain-lain.

1. Rancangan *platform.*

Rancangan ini berupa rancangan yang menentukan *hardware* dan *software* yang akan digunakan.

1. Rancangan basis data.

Rancangan ini berupa rancangan-rancangan berkas dalam basis data termasuk penentuan kapasitas masing-masing.

1. Rancangan modul.

Rancangan ini berupa rancangan program yang dilengkapi dengan algoritma (cara modul / program kerja).

1. Rancangan kontrol.

Rancangan ini berupa rancangan kontrol-kontrol yang digunakan dalam sistem seperti validasi, otorisasi dan audit data.

1. Dokumentasi.

Berupa hasil dokumentasi hingga tahap perancangan fisik.

1. Rencana pengujian.

Berupa rencana yang dipakai untuk menguji sistem.

1. Rencana konversi.

Berupa rencana untuk menerapkan sistem baru terhadap sistem lama.

Bagan Alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.Bagan alir sistem digambarkan dengan simbol-simbol sebagai berikut :

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabel 2.4**Bagan Alir Sistem | | | |
| **NO** | **NAMA SIMBOL** | **SIMBOL** | **KETERANGAN** |
| 1. | Simbol Dokumen |  | Menunjukkan dokumen *input* dan *output* baik itu proses manual, mekanik, atau komputer |
| 2. | Simbol Kegiatan Manual |  | Menunjukan pekerjaan manual |
| 3. | Simbol Simpanan Offline |  | Menunjukkan file non-komputer yang diarsip urut angka (*numerical*), huruf (*alphabetical*), atau tanggal (*chronological*) |
| 4. | Simbol Kartu Plong |  | Menunjukkan *input* dan *output* yang menggunakan kartu plong (*punched card*). |
| 5. | Simbol Proses |  | Menunjukkankegiatan proses dari operasi program komputer |
| 6 | Simbol Operasi Luar |  | Menunjukkan operasi yang dilakukan di luar proses operasi komputer |
| 7. | Simbol Pengurutan Offline |  | Menunjukkan proses urut data di luar proses komputer. operasi luar, menunjukkan operasi yang dilakukan di luar proses operasi komputer |
| 8. | Simbol Pita Magnetik |  | Menunjukkan *input* dan *output* menggunakan pita *magnetic*. |
| 9. | Simbol Hard Disk |  | Menunjukkan *input* dan *output*  menggunakan *harddisk* |
| 10. | Simbol Diskette |  | Menunjukkan *input* dan *output* menggunakan *diskette* |
| 11. | Simbol Drum Magnetik |  | Menunjukkan *input* dan *output* menggunakan drum magnetic |
| 12. | Simbol Pita Kertas Berlubang |  | Menunjukkan *input* dan *output* menggunakan pita kertas berlubang. |
| 13. | Simbol Keyboard |  | Menunjukkan *input* yang menggunakan *on-line keyboard* |
| 14. | Simbol Display |  | Menunjukkan *output* yang ditampilkan di monitor. |
| 15. | Simbol Pita Kontrol |  | Menunjukkan penggunaan pita kontrol (*control tape*) dalam *batch control* total untuk pencocokan di proses *batch processing*. |
| 16 | Simbol Hubungan Komunikasi |  | Menunjukkan proses transmisi data melalui *channel* komunikasi. |
| 17. | Simbol Garis Alir |  | Menunjukkan arus dari proses |
| 18. | Simbol Penjelasan |  | Menunjukkan penjelasan dari suatu proses |
| 19. | Simbol Penghubung |  | Menunjukkan penghubung ke halaman yang masih sama atau ke halaman yang lain |

Sumber : (Jogiyanto HM, 2005 : 796-799)

Untuk mempermudah penggambaran suatu sistem yang ada atau sistem yang baru yang akan dikembangkan secara logika tanpa memperhatikan lingkungan fisik di mana data tersebut mengalir atau lingkungan fisik di mana data tersebut akan disimpan, maka digunakan Diagram Arus Data (DAD) atau *Data Flow Diagram* (DFD). Dalam menggambarkan sistem perlu dilakukan pembentukan simbol, berikut ini simbol-simbol yang sering digunakan dalam DAD :

1. *External entity* (kesatuan luar) atau *boundary* (batas sistem).

Setiap sistem pasti mempunyai batas sistem *(boundary)* yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Kesatuan luar *(external entity)* merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lain yang berada di lingkungan luarnya yang akan memberikan *input* serta menerima *output* dari sistem. (Jogiyanto HM, 2005 ;701)

**Gambar 2.3**Contoh Notasi kesatuan luar

1. *Data flow* (arus data)

Arus data ini menunjukkan arus atau aliran data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem. (Jogiyanto HM, 2005 ;702)

Nama Arus Data

**Gambar 2.4** Contoh Notasi arus data

1. *Process* (proses)

Suatu proses adalah kegiatan atau kerja yang dilakukan orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses. (Jogiyanto HM, 2005 ;705)



**Gambar 2.5** Contoh Notasi proses

1. *Data store* (simpanan data).

Simpanan data pada DFD dapat disimbolkan dengan sepasang garis horisontal paralel yang tertutup disalah satu ujungnya. (Jogiyanto HM, 2005 ;707)

Media Nama Data store

**Gambar 2.6** Contoh Notasi simpanan data

## **2.2.10 Implementasi Sistem**

Sistem telah dianalisa dan didesain secara rinci dan teknologi telah diseleksi dan dipilih. Tiba saatnya sekarang sistem untuk diimplementasikan (diterapkan). Tahap implementasi sistem merupakan tahap meletakkan sistem supaya siap untuk dioperasikan. Tahap implementasi sistem dapat terdiri dari langkah-langkah sebagai berikut:

1. Menerapkan rencana implementasi

Rencana implementasi merupakan kegiatan awal dari tahap implementasi sistem. Rencana implementasi dimaksudkan terutama untuk mengatur biaya dan waktu yang dibutuhkan selama tahap implementasi.

1. Melakukan Kegiatan Implementasi

Kegiatan implementasi dilakukan dengan dasar kegiatan yang telah direncanakan dalam rencana implementasi. Kegiatan-kegiatan yang dapat dilakukan dalam tahap implementasi ini adalah sebagai berikut :

* 1. Persiapan tempat dan instalasi perangkat keras dan perangkat lunak

Jika peralatan baru akan dimiliki, maka tempat atau ruangan untuk peralatan ini perlu dipersiapkan terlebih dahulu. Keamanan fisik dari tempat ini perlu juga dipertimbangkan. Sistem komputer yang besar membutuhkan tempat dengan lingkungan yang lebih harus diperhitungkan. Langkah selanjutnya setelah persiapan fisik tempat adalah menginstalasi perangkat keras yang sudah dikirim dan menginstalasi perangkat lunak yang sudah ada.

* 1. Pemrograman dan pengetesan sistem

Pemrograman merupakan kegiatan menulis kode program yang akan dieksekusi oleh komputer. Kode program yang ditulis oleh pemrogram harus berdasarkan dokumentasi yang disediakan oleh analis sistem hasil dari desain sistem secara rinci. Sebelum program diterapkan, maka program harus terlebih dahulu bebas dari kesalahan-kesalahan. Oleh sebab itu, program harus diuji untuk menemukan kesalahan-kesalahan yang mungkin dapat terjadi. Program dites untuk tiap-tiap modul dan dilanjutkan dengan pengetesan untuk semua modul yang telah dirangkai.

* 1. Pengetesan sistem.

Pengetesan sistem biasanya dilakukan setelah pengetesan program. Pengetesan sistem dilakukan untuk memeriksa kekompakan antar komponen sistem yang diimplementasikan. Tujuan utama dari pengetesan sistem ini adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan.

## **2.2.11 Operasi dan Pemeliharaan**

Setelah masa sistem berjalan sepenuhnya menggantikan sistem lama, sistem memasuki pada tahapan operasi dan pemeliharaan. Zwass (1999) membagi pemeliharaan perangkat lunak menjadi tiga macam,yaitu :

1. Pemeliharaan perfektif.

Pemeliharaan Perfektif ditujukan untuk memperbarui sistem lama sebagai tanggapan atas perubahan kebutuhan pemakai dan kebutuhan organisasi, meningkatkan efesiensi sistem, dan memperbaiki dokumentasi.

1. Pemeliharaan adaptif.

Pemeliharaan Adaptif berupa perubahan aplikasi untuk menyesuaikan diri terhadap lingkungan perangkat keras dan perangkat lunak baru. Sebagai contoh pemeliharaan ini dapat berupa perubahan aplikasi dari *mainframe* ke lingkungan *client/ server* atau mengkonversi dari sistem berbasis berkas ke lingkungan basis data.

1. Pemeliharaan korektif.

Pemeliharaan korektif berpa pembetulan atas kesalahan-kesalahan yang ditemukan pada saat sistem berjalan.

## **2.2.12 Teknik Pengujian Sistem**

### 2.2.12.1 White Box

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, desain dan pengkodean.

Pengujian sistem / perangkat lunak memiliki sejumlah aturan yang berfungsi sebagai sasaran pengujian, diantaranya adalah sebagai berikut :

1. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
2. *Test case* yang baik adalah *test case* yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
3. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

Pengujian *White Box*, adalah metode pengujian yang menggunakan struktur kontrol desain prosedur untuk memperoleh *test case*. Dengan menggunakan metode *white box*, perekayasa sistem dapat melakukan *test case* yang memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali, menggunakan semua keputusan logis pada sisi *true* dan *false*, mengeksekusi semua *loop* pada batasan mereka dan pada batas operasional mereka, dan menggunakan stuktur data internal untuk menjamin validitasnya. Pengujian *Basis Path* adalah teknik pengujian *white box* yang diusulkan pertama kali oleh Tom McCabe. Metode *basis path* ini memungkinkan desainer *test case* mengukur kompleksitas logis dari desain prosedural dan menggunakannya sebagai pedoman untuk menetapkan basis set dari jalur eksekusi. (Roger S. Pressman, 2002 : 536).

**11**

**9**

**10**

**5**

**4**

**1**

**2**

**3**

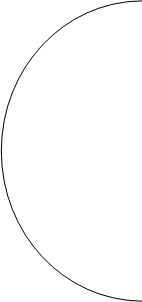
**6**

**7**

**8**

**Gambar 2.7** Bagan Alir

Bagan alir digunakan untuk menggambarkan struktur kontrol program. Dan untuk menggambarkan grafik alir, harus memperhatikan representasi desain prosedural pada bagan alir. Pada gambar dibawah ini, grafik alir memetakan bagan alir tersebut ke dalam grafik alir yang sesuai (dengan mengasumsikan bahwa tidak ada kondisi senyawa yang diisikan di dalam diamond keputusan dari bagan alir tersebut). Masing-masing lingkaran, yang disebut *simpul* grafik alir, merepresentasikan satu atau lebih statemen prosedural. Urutan kotak proses dan permata keputusan dapat memetakan simpul tunggal. Anak panah tersebut yang disebut *edges* atau *links*, merepresentasikan aliran kontrol dan analog dengan anak panah bagan alir. *Edge* harus berhenti pada suatu simpul, meskipun bila simpul tersebut tidak merepresentasikan statemen prosedural. (Roger S. Pressman, 2002 : 536).

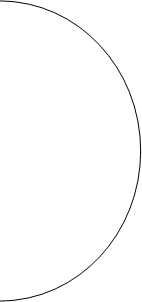


*Edge*

Simpul

Predikat

*Node*



Region

**R1**

**R4**

**R2**

**R3**

**Gambar 2.8** Grafik Alir

* *Node* adalah lingkaran yang merepresentasikan satu atau lebih statemen prosedural.
* *Edge* adalah anak panah pada grafik alir.
* *Region* adalah area yang membatasi *edge* dan *node*
* Simpul Predikat adalah simpul atau *node* yang berisi kondisi yang ditandai dengan dua atau lebih *edge* yang berasal darinya.

Dari gambar *flowgraph* di atas didapat :

*Path* 1 =1– 11

*Path* 2 =1– 2 – 3 – 4 – 5 – 10– 1–11

*Path* 3 =1– 2 – 3 – 6 – 8 – 9 – 10– 1 – 11

*Path* 4 =1– 2 – 3 – 6 – 7 – 9–10–1–11

*Path* 1,2,3,4 yang telah didefinisikan diatas merupakan *basis set* untuk diagram alir.

*Cyclomatic complexity* digunakan untuk mencari jumlah *path* dalam satu *flowgraph*. Dapat dipergunakan rumusan sebagaiberikut :

1. Jumlah region grafik alir sesuai dengan *cyclomatic complexity*.

2. *Cyclomatix complexity*V(G) untuk grafik alir dihitung dengan rumus:

***V(G) =E– N +2*** …………………. (2.9)

Dimana:

E= jumlah *edge* pada grafik alir

N= jumlah *node* pada grafik alir

Jalur 3 : 1 – 2 – 3 – 6 – 8 – 9 – 10 – 1 – 11

Jalur 4 : 1 – 2 – 3 – 6 – 7 – 9 – 10 – 1 – 11

Jalur 1, 2, 3, dan 4 yang ditentukan di atas terdiri dari sebuah basis set untuk grafik alir pada gambar 2.9 Bagaimana kita tahu banyaknya jalur yang dicari? Komputasi kompleksitas siklomatis memberikan jawaban. Fondasi kompleksitas siklomatis adalah teori grafik, dan memberi kita metriks perangkat lunak yang sangat berguna. Kompleksitas dihitung dalam salah satu dari tiga cara berikut :

1. Jumlah region grafik alir sesuai dengan kompleksitas siklomatis.
2. Kompleksitas siklomatis, *V(G),* untuk grafik alir *G* ditentukan sebagai

*V(G)* = *E – N* + 2 di mana *E* adalah jumlah *edge* grafik alir dan *N* adalah jumlah simpul grafik alir.

1. Kompleksitas siklomatis, *V(G),* untuk grafik alir G juga ditentukan sebagai *V(G)* = P + 1, dimana P adalah jumlah simpul predikat yang diisikan dalam grafik alir *G*.

Pada gambar 2.8 grafik alir, kompleksitas siklomatis dapat dihitung dengan menggunakan masing-masing dari algoritma yang ditulis di atas :

1. Grafik alir mempunyai 4 region
2. *V(G)* = 11 *edge* – 9 simpul + 2 = 4
3. *V(G)* = 3 simpul yang diperkirakan + 1 =4

Dengan demikian, kompleksitas siklomatis dari grafik alir pada gambar 2.9 adalah 4. Yang lebih penting, nilai untuk *V(G)* memberi kita batas atas untuk jumlah jalur independen yang membentuk basis set, dan implikasinya, batas atas jumlah pengujian yang harus didesain dan dieksekusi untuk menjamin semua statemen program.

### 2.2.12.2 Black Box

Pengujian *Black-Box* berusaha menemukan kesalahan dalam kategori :

* pada struktur data (pengaksesan basis data)
* Kesalahan Fungsi tidak benar atau hilang
* Kesalahan antar muka
* Kesalahan inisialisasi dan akhir program
* Kesalahan performasi.

Pengujian ini berfokus pada persyaratan fungsional perangkat lunak dan merupakan komplemen dari pengujian *White-Box*. Hal tersebut dapat dicapai melalui :

1. Pengujian *Graph-based* : dimulai dengan membuat grafik sekumpulan *node* yang mempresentasikan objek (misal *New File*, Layar baru dengan atributnya), link (hubungan antar objek), *node-weight* (misal nilai data tertentu seperti atribut layar, perilaku), dan link-weight (karakteristik suatu link, misal menu select).
2. *Equivalence Partitioning* : membagi domain *input* untuk pengujian agar diperoleh kelas-kelas kesalahan (misal kelompok data karakter, atau atribut yang lain).
3. Analisis Nilai Batas : pengujian berdasarkan nilai batas domain *input*.
4. Pengujian Perbandingan : disebut juga pengujian *back-to-back* yang diterapkan pada pada suatu versi perangkat lunak atau perangkat lunak redundan untuk memastikan konsistensinya.

## **2.2.13 Tools Pendukung**

**Tabel 2.5** Tools Pendukung

|  |  |  |
| --- | --- | --- |
| No | Tools | Keterangan |
| 1 | PHP | PHP merupakan kependekan dari *Personal Home Page* (Situs personal),sebagai bahasa pemrograman yang digunakan utnuk pembutan sistem |
| 2 | MySql | MySQL adalah sistem manajemen database SQL yangmer bersifat Open Source dan paling populer saat ini. |

## **Kerangka Pemikiran**

1. Untuk mengetahui hasil penerapan metode regresi linear untuk prediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo
2. Metode Regresi Linier mampu memprediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo

**Peluang**

1. Bagaimana hasil penerapan Metode Regresi Linear untuk prediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo?
2. Seberapa besar tingkat akurasi untuk prediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo dengan Metode Regresi Linear?

**Masalah**

**Analisis Sistem**

1.Sistem berjalan.

2.Sistem yang diusulkan.

**Pembangunan Sistem**

1. PHP
2. MySQL.
3. Desain Sistem
4. Desain Model
5. Desain User Interface
   1. Desain Output
   2. Desain Input
   3. Desain Database
6. Desain Teknologi

**Desain Sistem**

**Pengujian Sistem**

1.*White* Box.

2.Black*Box*.

**Implementasi Sistem**

\

Kantor Imigrasi Kelas I TPI Gorontalo

1. Untuk mengetahui hasil penerapan Metode Regresi Liniear untuk prediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo.
2. Untuk memperoleh akurasi yang baik untuk prediksi jumlah permintaan blangko paspor di Kantor Imigrasi Kelas I TPI Gorontalo dengan menggunakan Metode Regresi Linear*.*

**Tujuan**

Gambar 2.9 Kerangka Pemikiran

**Gambar 2.7**  Bagan Kerangka berpikir