# BAB II

# LANDASAN TEORI



## Tinjauan Studi

Ada beberapa penelitian terkait yang pernah dilakukan yaitu :

1. Penelitian yang dilakukan oleh Fathur Rahman dan Muhammad Iqbal Firdaus (2016) yang berjudul Penerapan Data Mining Metode *Naïve Bayes* Untuk Prediksi *Hasil Belajar Siswa Sekolah Menengah Pertama (SMP).* Penelitian ini bertujuan untuk mengetahui tingkat akurasi serta hasil presisi dan Recall dari metode klasifikasi data mining yaitu *Naïve Bayes* untuk prediksi hasil belajar siswa menengah pertama (SMP). Penentuan hasil belajar dari siswa didik merupakan hal penting dalam dunia pendidikan, Hal ini dianggap serius karena sulitnya menentukan faktor atau variabel yang mempengaruhi hasil belajar siswa. Penerapan data mining yang dapat mengenali dan mengekstrak pola pengetahuan menawarkan solusi untuk masalah pendidikan ini. Ada beberapa model klasifikasi dalam data mining yaitu algoritma ID3 dan C4.5 juga Naïve Bayes yang dapat digunakan untuk memprediksi hasil belajar siswa khususnya siswa menengah pertama. Pada penelitian ini akan digunakan metode klasifikasi naïve bayes untuk memprediksi hasil belajar siswa SMP dengan harapan mendapat akurasi yang lebih baik dari metode sebelumnya yang pernah diuji.
2. Penelitian yang dilakukan oleh Riszki Wijayatun Pratiwi dan Yusuf Sulistyo Nugroho (2016) yang berjudul Prediksi *Rating Film* Menggunakan Metode *Naïve Bayes*. Penelitian ini berisi tentang Para penikmat film juga membutuhkan film-film yang mempunyai kualitas gambar, suara, alur cerita dan nilai positif yang baik dalam sebuah film, agar mereka tetap antusias dalam mengikuti film-film yang terbaru. Namun film-film yang ada tidak semuanya dapat dinikmati dan tidak semua kalangan menyukai semua film. Agar suatu film dapat terus berkembang, tentunya membutuhkan penilaian-penilaian dari para penikmat film, untuk mengetahui selera film yang sesuai dengan para penikmat film. Untuk itu dibutuhkan analisis agar dapat mengetahui bagaimana minat penikmat film yaitu dengan membuat penilaian-penilaian yang nantinya digunakan untuk mengetahui *rating* suatu film menggunakan metode *naïve bayes* yaitu metode yang melakukan pendekatan statistika yang fundamental dalam pengenalan pola (*pattern recognition*). Pendekatan ini didasarkan pada kuantifikasi *trade-off* antara berbagai keputusan klasifikasi dengan menggunakan probabilitas dan resiko yang ditimbulkan dalam keputusan-keputusan tersebut. Metode tersebut merupakan salah satu metode dari *data mining*, dengan atribut yang sudah ditentukan, yaitu meliputi *genre* film, aktor film, bahasa, warna, durasi film, negara, dan lainnya yang dapat digunakan sebagai tolak ukur sutradara untuk membuat film.
3. Penelitian yang dilakukan oleh Nanang Budi Kurniawan, Pulung Nurtantio Andono (2016) yang berjudul Prediksi *Kemenangan Bot Dota 2* Menggunakan Metode *Naive Bayes* Dota 2 Bot Victory Prediction Using Naive Bayes. Penelitian ini membahas tentang permainan game dota 2*,* pemain bisa dibantu oleh hero dota sekutu yang dijalankan oleh AI game. Masing-masing team memiliki markas utama yang harus dilindungi, markas ini disebut Ancient. Ketika ancient salah satu team hancur, maka pertandingan selesai. Naive bayes adalah algoritma data mining klasifikasi yang digunakan untuk prediksi kemenangan bot dalam pertandingan dota 2. Dengan pembobotan statistik pada masing-masing bot akan dilakukan prediksi kemenangan bot pada game dota 2 sehingga akan diketahui team radiant atau dire yang akan memenangkan pertandingan dalam game dota 2. Akurasi yang dihasilkan naive bayes untuk memprediksi kemenangan team dota 2 adalah sebesar 54.17%.

## Tinjauan Pustaka



### Pajak Kendaraan Bermotor

Pajak Kendaraan bermotor (PKB) adalah semua kendaraan beroda beserta gandengannya yang digunakan di semua jenis jalan darat, dan digerakkan oleh peralatan teknik berupa motor atau peralatan lainnya yang berfungsi untuk mengubah suatu sumber daya energi tertentu menjadi energi tertentu menjadi tenaga gerak kendaraan bermotor yang bersangkutan, termasuk alat berat dan alat besar yang dalam operasinya menggunakan roda dan motor yang tidak melekat secara permanen serta kendaraan bermotor yang dioperasikan diair. Pajak kendaraan bermotor, dipungut pajak atas kepemilikan dan/atau pengusaan kendaraan bermotor. (Peraturan Daerah Nomor 2 Tahun 2015 tentang perubahan Peraturan Daerah Nomor 8 Tahun 2010 tentang Pajak Kendaraan Bermotor)

Pajak Kendaraan Bermotor dikenakan untuk Masa Pajak 12 bulan berturut-turut terhitung mulai saat pendaftaran Kendaraan Bermotor. Pajak Kendaraan Bermotor dibayar sekaligus di muka. Untuk Pajak Kendaraan Bermotor yang karena keadaan kahar (force majeure) Masa Pajaknya tidak sampai 12 bulan, dapat dilakukan restitusi atas pajak yang sudah dibayar untuk porsi Masa Pajak yang belum dilalui. Ketentuan lebih lanjut mengenai tata cara pelaksanaan restitusi diatur dengan Peraturan Gubernur.

### 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 database besar. Istilah data mining memiliki hakikat sebagai disiplin ilmu yang tujuan utamanya adalah untuk menemukan, menggali, atau menambang pengetahuan dari data atau informasi yang kita miliki. Data mining, sering juga disebut sebagai *Knowledge Discovery in Database* (KDD). KDD adalah kegiatan yang meliputi pengumpulan, pemakaian data, historis untuk menemukan keteraturan, pola atau hubungan dalam set data berukuran besar (Mujib Ridwan, dkk. 2013).

1. Metode Pelatihan

Secara garis besar metode pelatihan yang digunakan dalam teknik-teknik *data mining* dibedakan ke dalam dua pendekatan, yaitu :

* *Unsupervised learning,* metode ini diterapkan tanpa adanya latihan *(training)* dan tanpa ada guru *(teacher).* Guru di sini adalah label dari data.
* *Supervised learning,* yaitu metode belajar dengan adanya latihan dan pelatih. Dalam pendekatan ini, untuk menemukan fungsi keputusan, fungsi pemisah atau fungsi regresi, digunakan beberapa contoh data yang mempunyai output atau label selama proses *training*.

2. Pengelompokan Data Mining

Ada beberapa teknik yang dimiliki *data mining* berdasarkan tugas yang bisa dilakukan, yaitu :

* Deskripsi

Para peneliti biasanya mencoba menemukan cara untuk mendeskripsikan pola dan trend yang tersembunyi dalam data.

* Estimasi

Estimasi mirip dengan klasifikasi, kecuali variabel tujuan yang lebih kearah numerik dari pada kategori.

* Prediksi

Prediksi memiliki kemiripan dengan estimasi dan klasifikasi. Hanya saja, prediksi hasilnya menunjukkan sesuatu yang belum terjadi (mungkin terjadi dimasa depan).

* Klasifikasi

Dalam klasifikasi variabel, tujuan bersifat kategorik. Misalnya, kita akan mengklasifikasikan pendapatan dalam tiga kelas, yaitu pendapatan tinggi, pendapatan sedang, dan pendapatan rendah.

* *Clustering*

*Clustering* lebih ke arah pengelompokan *record*, pengamatan, atau kasus dalam kelas yang memiliki kemiripan.

* Asosiasi

Mengidentifikasi hubungan antara berbagai peristiwa yang terjadi pada satu waktu

3. Tahap-Tahap Data Mining

Sebagai suatu rangkaian proses, *data mining* dapat dibagi menjadi beberapa tahap proses tersebut bersifat interaktif, pemakai terlibat langsung atau dengan perantaraan *knowledgebase*. Tahap-tahap *data mining* adalah sebagai berikut:

* Pembersihan data *(data cleaning)*

Pembersihan data merupakan proses menghilang-kan *noise* dan data yang tidak konsisten atau data tidak relevan.

* Integrasi data *(data integration)*

Integrasi data merupakan penggabungan data dari berbagai *database* ke dalam satu *database* baru.

* Seleksi data *(data selection)*

Data yang ada pada database sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari *database*.

* Transformasi data *(data transformation)*

Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam *data mining*.

* Proses *mining*

Merupakan suatu proses utama saat metode diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

* Evaluasi pola *(pattern evaluation)*

Untuk mengidentifikasi pola-pola menarik kedalam *knowledge based* yang ditemukan.

* Presentasi pengetahuan *(knowledgepresentation)*

Merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna.



### Naïve Bayes

Naïve Bayes merupakan salah satu metode *data mining* yang menggunakan konsep klasifikasi dan prediksi. Metode ini melakukan prediksi dengan menghitung nilai probabilitas (x│y), yaitu probbilitas kelas x jika diketahui suatu y, berdasarkan theorema Bayes. Prediksi dapat dilakukan untuk menentukan suatu kelas x € X dari suatu dokumen y € Y dengan X = {x1, x2, ...xn) dan Y = (y1, y2, …yn). Penentuan kelas dalam prediksi dokumen tersebut dilakukan dengan cara memilih nilai maksimum dari p(x│y) dari distribusi probabilitas P= {p (x│y│x € X dan y € Y}. Suatu dokumen y ke i dapat direpresentasikan sebagai vektor dari nilai-nilai fitur yang ada pada dokumen tersebut sehingga y = {fi1, fi2……., fin}. Nilai dari elemen tiap vektor merupakan nilai untuk fitur fj pada himpunan fitur F = {f1, f2, …, f3} dengan fij adalah nilai dari fitur ke j pada dokumen y ke i. Berdasarkan metode Naïve Bayes perhitungan nilai probabilitas p(x│y) dapat dilakukan dengan menggunakan persamaan (Mitchell, 2005).

p(x│y) = ......................... (2.1)

Dengan p(x│y) merupakan nilai probabilitas dari kemunculan dokumen y pada kelas x, p(x) merupakan nilai probabilitas kemunculan kelas x dan p(y) merupakan nilai probabilitas kemunculan dokumen y.

Umumnya, Bayes mudah dihitung untuk fitur bertipe kategoris seperti pada kasus klasifikasi. Namun untuk fitur dengan tipe numerik (kontinu) ada perlakuan khusus sebelum dimasukkan dalam Naïve Bayes yaitu dengan mengasumsikan bentuk tertentu dari distribusi probabilitas untuk fitur kontinu dan memperkirakan parameter distribusi dengan data pelatihan. Distribusi Gaussian biasanya dipilih untuk mempresentasikan probabilitas bersyarat dari fitur kontinu pada sebuah kelas *p*, sedangkan distribusi Gaussian dikarakteristikkan dengan dua parameter: mean, , dan varian, . Untuk setiap kelas yj probabilitas bersyarat kelas yj untuk fitur Xi adalah:

*p*

Parameter bisa didapat dari mean sampel Xi dari semua data latih yang menjadi milik kelas yj sedangkan dapat diperkirakan dari varian sampel (s2) dari data latih.

**2.2.3.1 Contoh Penerapan Metode Naive Bayes**

Perhitungan Naïve Bayes untuk prediksi pembelian tiket :

Tabel 2.1 Contoh Data Untuk Penerapan Naïve Bayes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID |  |  |  | Class |
|
| 1 | Garuda Indonesia | Jerman | Internasional | Vip |
| 2 | Batik Air | Amerika Serikat | Domestik | Vip |
| 3 | Garuda Indonesia | Amerika Serikat | Domestik | Vip |
| 4 | Batik Air | Amerika Serikat | Domestik | Ekonomi |
| 5 | Japan Airline | Jepang | Internasional | Vip |
| 6 | Batik Air | Amerika Serikat | Domestik | Vip |
| 7 | Japan Airline | Jerman | Domestik | Vip |
| 8 | Garuda Indonesia | Amerika Serikat | Domestik | Ekonomi |
| 9 | Korea Air | Jerman | Domestik | Ekonomi |
| 10 | Garuda Indonesia | Jerman | Internasional | Vip |
| 11 | Korea Air | Jerman | Domestik | Ekonomi |
| 12 | Garuda Indonesia | Amerika Serikat | Domestik | Vip |
| 13 | Batik Air | Amerika Serikat | Domestik | Vip |
| 14 | Japan Air | Amerika Serikat | Internasional | Ekonomi |
| 15 | Korea Air | Jerman | Domestik | Ekonomi |
| 16 | Garuda Indonesia | Amerika Serikat | Domestik | Ekonomi |
| 17 | Garuda Indonesia | Amerika Serikat | Domestik | Ekonomi |
| 18 | Garuda Indonesia | Jerman | Internasional | Vip |
|  | | | | |
| ID |  |  |  | Class |
| 19 | Garuda Indonesia | Jerman | Internasional | Vip |
| 20 | Batik Air | Amerika Serikat | Domestik | Ekonomi |
| 21 | Korea Air | Jepang | Internasional | Vip |
| 22 | Garuda Indonesia | Amerika Serikat | Domestik | Ekonomi |
| 23 | Garuda Indonesia | Amerika Serikat | Domestik | Ekonomi |
| 24 | Batik Air | Amerika Serikat | Domestik | Ekonomi |
| 25 | Japan Airline | Jepang | Internasional | Ekonomi |
| 26 | Japan Airline | Jepang | Internasional | Ekonomi |
| 27 | Korea Air | Jepang | Internasional | Vip |
| 28 | Japan Airline | Jerman | Domestik | Vip |
| 29 | Batik Air | Amerika Serikat | Domestik | Vip |
| 30 | Garuda Indonesia | Jerman | Internasional | Vip |

Jika Diketahui Tabel Distrubisi yang ada di atas. Tentukan pernyataan berikut mengunakan *naïve bayes :*

Korea Airline | Jerman | Internasional | ?

Vip = 16/30 = 0.533333

Ekonomi = 14/30 = 0,46666667

Korea Airline | Jerman | Internasional | ?

P ( = Korea Airline | Class = Vip) =

P ( = Jerman | Class = Vip) =

P ( = Internasional | Class = Vip) =

Vip = 0,53333333 (0,125 x 0,43799 x 0,5) = 0,01458333

P ( = Korea Airline | Class = Ekonomi) =

P ( = Jerman | Class = Ekonomi) =

P ( = Internasional | Class = Ekonomi) =

Ekonomi =( = 0.712646

Sehingga Keputusannya adalah **Ekonomi.**

### 2.2.4 Confusion Matrix

Confusion Matrix adalah suatu metode yang biasanya digunakan untuk melakukan perhitungan akurasi pada konsep data mining. Confusion matrix memberikan keputusan yang diperoleh penilaian performance klasifikasi berdasarkan objek dengan benar atau salah (gurunescu). Confusion matrix berisi informasi actual (actual) dan prediksi (predicted) pada sistem klasifikasi.

Tabel 2.2 Tabel Confusion Matrix

|  |  |  |
| --- | --- | --- |
|  | Actual | Actual |
|  | positive | Negative |
| predicted positive | *TP* | *FP* |
| predicted negative | *FN* | *TN* |

* *Recall* adalah proposisi kasus positif yang diidentifikasi dengan benar. Rumus dari*Recall =* d/(c+d)
* *Precision* adalah proporsi kasus dengan hasil positif yang benar. Rumus dari *Precision = d*/(b+d)
* *Accuracy* adalah perbandingan kasus yang diidentifikasi dengan benar dengan jumlah semua kasus. Rumus dari *Accuracy* = (a+d)(a+b+c+d)
* *Erorr Rate* adalah jumlah kasus yang diidentifikasi salah dengan jumlah semua kasus. Rumus dari *Error Rate* = (b+c)(a=b+c+d)

Keterangan :

a = jika hasil prediksi negative dan nilai sebenarnya positif

b = jika hasil prediksi postitif sedangkan nilai sebenarnya negative

c = jika hasil prediksi negative sedangkan nlai sebenarnya positif

d = jika hasil prediksi positif sedangkan nilai sebenarnya positif



## Siklus Hidup Pengembangan Sistem

Dalam membangun sebuah sistem (dalam hal ini lebih mengacu kepada pengertian aplikasi perangkat lunak) digunakan metode siklus hidup pengembangan sistem (System Development *Life Sycle* atau SDLC). SDLC terdiri dari sejumlah tahapan yang dilaksanakan secara berurutan. *System Development Life Sycle* atau siklus hidup pengembangan sistem (SDLC), merupakan metode alternatif. Metode SDLC mempunyai beberapa kelebihan dan kekurangan.

Kelebihan-kelebihan dari metode ini adalah:

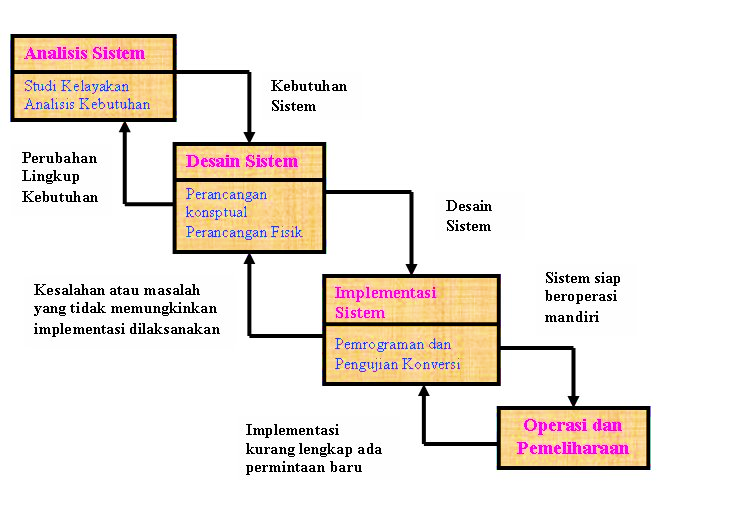
1. Menyediakan tahapan yang dapat digunakan sebagai pedoman mengembangkan sistem.
2. Memberikan hasil sistem yang lebih baik karena sistem dianalisis dan dirancang secara keseluruhan sebelum di implementasikan.

Disamping kelebihan-kelebihan tersebut, SDLC juga mempunyai kekurangan, yang diantaranya adalah:

1. Hasil dari SDLC tergantung dari hasil tahap analisis, sehingga jika terdapat kesalahan analisis, akan terbawa terus.
2. Dibutuhkan waktu yang lama untuk mengembangkannya karena sistem harus dikembangkan sampai selesai semua terlebih dahulu.

Tahapan-tahapan dalam metode SDLC adalah sebagai berikut :

1. Analisis sistem
2. Perancangan sistem
3. Implementasi sistem
4. Operasi dan perawatan sistem

SDLC tampak jika sistem yang sudah dikembangkan dan dioperasikan tidak dapat dirawat lagi, sehingga dibutuhkan pengembangan sistem kembali. Siklus hidup pengembangan sistem dengan langkah-langkah utama adalah sebagai berikut:

Gambar 2.1 Siklus Hidup Pengembangan Sistem



### Analisa Sistem

Analisa sistem (*System Analisa*) dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Analisa sistem adalah spesialis yang mempelajari masalah dan kebutuhan sebuah organisasi untuk menentukan bagaimana orang, data, proses dan teknologi informasi dapat mencapai kemajuan terbaik untuk bisnis.

Analisis sistem adalah *Stakeholder* yang berperan sebagai fasilitator atau pelatih, menjembatani jurang komunikasi yang dapat secara alamiah berkembang antara pemilik dan pengguna *system nonteknis* atau desainer dan perkembangan sistem teknis.

Whitten, et al (2004 :33) mengungkapkan “*System analysis* adalah study domain masalah bisnis untuk merekomendasikan perbaikan dan menspesifikasi persyaratan dan prioritas bisnis untuk solusi”.

Sebagai tambahan keahlian analisis dan desain sistem formal, seorang analis harus mengembangkan atau memilki keahlian lain, pengetahuan, dan karakter untuk menyelesaikan pekerjaan. Hal ini termasuk:

1. Pengalaman dan keahlian pemrograman komputer.

Sulit untuk membayangkan bagaimana para analisis sistem dapat dengan cukup mempersiapkan bisnis dan spesifikasi teknis untuk programer jika mereka tidak memilki pengalaman programan. Kebanyakan analis sistem harus menguasai satu atau lebih bahasa pemrograman tingkat tinggi.

1. Pengetahuan umum proses dan teknologi bisnis.

Analis sistem harus mampu berkomunikasi dengan para ahli bisnis untuk memperoleh pemahaman masalah dan kebutuhan mereka. Untuk analis paling tidak sebagian dari pengetahuan ini datang hanya dari pengalaman. Pada saat yang sama analis yang terinspirasi harus mengambil manfaat dari setiap kesempatan untuk menyelesaikan mata kuliah teori bisnis dasar.

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

1. Studi Kelayakan.

Studi kelayakan digunakan untuk menentukan kemungkinan keberhasilan solusi yang diusulkan. Tahapan berguna untuk memastikan bahwa solusi yang diusulkan tersebut benar-benar dapat dicapai dengan sumber daya dan dengan memperhatikan kendala yang terdapat pada perusahaan serta dampak terhadap lingkungan sekeliling. Tugas-tugas yang tercakup dalam studi kelayakan meliputi:

1. Penentuan masalah dan peluang yang dituju sistem.
2. Pembentukan sasaran sistem baru secara keseluruhan.
3. Pengidentifikasian para pemakai sistem.
4. Pembentukan lingkup sistem.

Selain itu, selama dalam tahapan studi kelayakan sistem analisis juga melakukan tugas-tugas sebagai berikut :

1. Pengusulan perangkat lunak dan perangkat keras untuk sistem baru.
2. Pembuatan analisis untuk membuat atau membeli aplikasi.
3. Pembuatan analisis biaya/manfaat.
4. Pengkajian terhadap resiko proyek.

Studi kelayakan diukur dengan memperhatikan aspek teknologi, ekonomi, faktor organisasi dan kendala hukum, etika, dan yang lain (Turban, et al 1999 dalam Abdul Kadir, 2003:403).

1. Analisis kebutuhan.

Analisis kebutuhan dilakukan untuk menghasilkan spesifikasi kebutuhan (disebut juga spesifikasi fungsional). Spesifikasi kebutuhan adalah spesifikasi yang rinci tentang hal-hal yang akan dilakukan sistem ketika diimplementasikan. Spesifikasi ini sekaligus dipakai untuk membuat kesepakatan antara pengembang sistem, pemakai yang kelak akan menggunakan sistem, manajemen, dan mitra kerja yang lain (misalnya auditor internal).

Analisis kebutuhan ini diperlukan untuk menentukan keluaran yang akan dihasilkan sistem, masukan yang diperlukan sistem, lingkup proses yang digunakan untuk mengolah masukan menjadi keluaran, volume data yang akan ditangani sistem, jumlah pemakai dan kategori pemakai, serta kontrol terhadap sistem.

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

1. *Identify,* yaitu mengidentifikasi masalah

Mengidentifikasi (mengenai) masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah (*problems*) dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Tahap identifikasi masalah sangat penting karena akan menentukan keberhasilan pada langkah-langkah selanjutnya.

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 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 dilakukan ;

1. Pelaporan bahwa analisi telah selesai dlakukan
2. Meluruskan kesalah pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analis sistem tetapi tidak sesuai menurut manajemen.

### Desain Sistem

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

Whitten, et al (2004:34) mengungkapkan: ”*System design* adalah spesifikasi atau intruksi solusi yang teknis dan berbasis komputer untuk persyaratan bisnis yang diidentifikasikan dalam analisis sistem.”

Desain sistem adalah spesifikasi atau intruksi solusi yang teknis dan berbasis komputer untuk persyaratan bisnis yang di identifikasikan dalam analisis sistem.

Driver teknologi sekarang (dan dimasa depan) paling berimpak pada proses dan keputusan desain sistem. Banyak organisasi mengidentifikasikan arsitektur teknologi informasi umum yang didasarkan pada driver-driver teknologi ini.

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 pemogram komputer dan ahli-ahli teknik lainnya.

Perancangan sistem adalah suatu keinginan membuat desain teknis yang berdasarkan evaluasi yang dilakukan pada kegiatan analisis. Perancangan disini dimaksudkan sebagai proses pemahaman dan perancangan suatu sistem berbasis komputer yang akan menghasilkan komputerisasi.

Dengan demikian, suatu kegiatan perancangan sistem bertujuan untuk menghasilkan suatu sistem komputerisasi. Komputerisasi adalah suatu kegiatan atau sistem pengolahan data dengan menggunakan komputer sebagai alat bantu. Perancangan sistem dilakukan setelah tahap analisis sistem selesai dilaksanakan yang kemudian akan menghasilkan *output* berupa kebutuhan yang akan dijadikan dasar untuk merancang sistem tersebut. Perancangan sistem terbagi dua yaitu :

1. 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 diimplementasikan. 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, et al*.* 1997 dalam abdul kadir (2003:407) evaluasi yang dilakukan mengandung hal-hal berikut :

1. Bagaiamana 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 masing-masing?

Setelah alternatif rancangan dipilih tahap selanjutnya adalah penyiapan spesikasi rancangan yang mencakup elemen- elemen sebagai berikut:

1. Keluaran

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

1. Penyiapan 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 dimasukan kedalam sistem.

1. Prosedur pemrosesan dan operasi

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

1. 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 dengan sistem

Rancangan ini berupa rancangan interaksi antara pemakai dan sistem, Misalnya: berupa menu, ikon, dan lain-lain.

1. Rancangan *platform*

Rancangan ini berupa rancangan yang menentukan *hardware* (perangkat keras) dan *software* (perangkat lunak) yang akan digunakan. 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 bekerja).

1. Rancangan control

Rancangan ini berupa rancangan kontrol-kontrol yang dugunakan dalam sistem seperti validasi, otorisasi, 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. Dalam perancangan sistem yang baik melalui tahapan-tahapan sebagai berikut:

1. Identifikasi masalah yaitu mengidentifikasi masalah yang ada secara rinci agar tidak timbul masalah lain selain masalah utama.
2. Menentukan input, proses dan output yang diinginkan yaitu menginginkan hasil dari perancangan sistem yang dibuat sesuai dengan prosedur.
3. Menentukan algoritma
4. Mengimplementasikan dengan bahasa pemograman tertentu

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

1. Desain sistem secara umum

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 analisis sistem untuk mengidentifikasikan

Komponen-komponen sistem informasi yang akan didesain secara rinci oleh pemograman komputer dan ahli teknik lainya.

1. Desain Sistem Terinci (*Detailed system design*)
2. *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 di media kertas dan desain output dalam bentuk dialog di layar terminal. Desain Output dalam bentuk laporan.

Desain ini dimaksudkan untuk menghasilkan output dalam bentuk laporan di media 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 rancangan bangun dari percakapan antara pemakai sistem (*user*) dengan komputer. Percakapan ini dapat terdiri dari proses memasukkan data ke system menampilkan output informasi kepada *user*, atau keduanya.

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 di sajikan kepada user. Pilihan menu akan lebih baik bila dikelompokkan fungsinya.

1. *Desain input* Terinci

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

Fungsi dokumen dasar dalam penanganan arus data:

1. Dapat menunjukan macam dari data yang harus dikumpulkan dan ditangkap.
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.
4. *Desain Database* Terinci

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan dan disimpan diluar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya. *Databse* merupakan salah satu komponen yang penting di sistem informasi, karena berfungsi sebagai 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 dengan yang lainya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam didalam suatu organisasi. Dengan sistem basis data ini tiap-tiap orang atau bagian dapat memandang *database* dari beberapa sudut pandang yang berbeda. Bagian kredit dapat memandangnya sebagai data penjualan, bagian personalia dapat memandangnya sebagai data karyawan, bagian gudang data yang dapat memandangnya sebagai data persediaan, semuanya terintegrasi dalam sebuah data yang umum.

1. Desain Teknologi

Tahap desain terbagi atas dua yaitu desain teknologi secara umum di rinci. Pada tahap ini kita menentukan teknologi yang akan di pergunakan dalam menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi yang di maksud 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 lain sebagainya. Desain teknologi sangat diperlukan pada tahap implementasi dan pengujian untuk membuktikan bahwa sistem dapat berjalan secara semestinya.

Tahap desain 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 di gambarkan dengan bagan alir sistem bagan alir dokumen, dan desain secara logika digambarkan dengan diagram dengan arus data (DAD), pada tahap desain model terinci, mode didefinisikan secara terinci urut-urutan langkah proses ini diwakili oleh suatu program komputer.

Bagan alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan alir sistem di gambar dengan simbol-simbol berikut:

Tabel 2.3 Bagan Alir Sistem

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **NAMA SIMBOL** | **SIMBOL** | **KETERANGAN** |
| 1 | Simbol Terminal |  | Menunjukkan untuk memulai dan mengakhiri Suatu proses. |
| 2 | Simbol Dokumen |  | Menunjukkan dokumen input dan output baik itu proses manual, mekanik, atau komputer. |
| 3 | Simbol Kegiatan Manual |  | Menunjukan pekerjaan manual |
| 4 | Simbol Simpanan Offline |  | Menunjukkan file non-komputer yang diarsip urut angka (*numerical*), huruf (*alphabetical*), atau tanggal (*chronological*) |
| 5 | Simbol Kartu Plong |  | Menunjukkan input dan output yang menggunakan kartu plong (*punched card*). |
| 6 | Simbol Proses |  | Menunjukkan kegiatan proses dari operasi program komputer |
| 7 | Simbol Operasi Luar |  | Menunjukkan operasi yang dilakukan di luar proses operasi komputer |
| 8 | Simbol Pengurutan Offline |  | Menunjukkan proses urut data di luar proses komputer. operasi luar, menunjukkan operasi yang dilakukan di luar proses operasi komputer |
| **NO** | **NAMA SIMBOL** | **SIMBOL** | **KETERANGAN** |
| 9 | Simbol Pita Magnetik |  | Menunjukkan input dan output menggunakan pita *magnetic*. |
| 10 | Simbol Hard Disk |  | Menunjukkan *input* dan *output* menggunakan *harddisk* |
| 11 | Simbol Diskette |  | Menunjukkan *input* dan *output* menggunakan *diskette* |
| 12 | Simbol Drum Magnetik |  | Menunjukkan *input* dan *output* menggunakan drum magnetic |
| 13 | Simbol Pita Kertas Berlubang |  | Menunjukkan *input* dan *output* menggunakan pita kertas berlubang. |
| 14 | Simbol Keyboard |  | Menunjukkan *input* yang menggunakan *on-line keyboard* |
| 15 | Simbol Display |  | Menunjukkan *output* yang ditampilkan di monitor. |
| 16 | Simbol Pita Kontrol |  | Menunjukkan penggunaan pita kontrol (*control tape*) dalam *batch control* total untuk pencocokan di proses *batch processing*. |
| 17 | Simbol Hubungan Komunikasi |  | Menunjukkan proses transmisi data melalui channel komunikasi. |
| 18 | Simbol Garis Alir |  | Menunjukkan arus dari proses |
| 19 | Simbol Penjelasan |  | Menunjukkan penjelasan dari suatu proses |
| **NO** | **NAMA SIMBOL** | **SIMBOL** | **KETERANGAN** |
| 20. | Simbol Penghubung |  | Menunjukkan penghubung ke halaman yang masih sama atau ke halaman yang lain |

Sumber : Jogiyanto, 2005 : 802

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 2.2 Notasi kesatuan luar di DAD

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 : 701)

Gambar 2.3 Nama Arus Data di DAD

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 2.4 Notasi Proses di DAD

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 2.5 Notasi Simpanan Data di DAD



### Implementasi Sistem

*Whitten*, *et al*. (2004 : 34) mengungkapkan: ” *System Implementation* adalah konstruksi, instalasi, pengujian, dan pengiriman sistem ke dalam produksi (artinya operasi sehari-hari)”.

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. Pemilihan dan Pelatihan Personil.

Telah diketahui bahwa manusia merupakan faktor yang perlu dipertimbangkan dalam sistem informasi. Jika sistem informasi ingin sukses, maka personil-personil yang terlibat harus diberi pengertian dan pengetahuan yang cukup tentang sistem informasi dan posisi serta tugas mereka.

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.

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



### Pemeliharaan Sistem

Tujuan dasar Pemeliharaan sistem

1. Membuat perubahan yang dapat diperkirakan pada program yang sudah ada untuk memperbaiki yang telah dibuat selama desain atau implementasi sistem.
2. Mempertahankan aspek-aspek program-program yang sudah benar dan menghindari kemungkinan bahwa “perbaikan-perbaikan pada program menyebabkan aspek lain dari program bertingkah laku dengan cara yang berbeda”
3. Sedapat mungkin menghindari terjadinya degradasi performasi sistem. Pemeliharaan sistem yang buruk dapat mengurangi *throughput* dan waktu proses.
4. Untuk menyelesaikan tugas secepat mungkin tanpa mengorbankan kualitas dan keandalan.

Untuk mencapai tujuan-tujuan tersebut, perlu memahami dengan tepat program yang sedang diperbaiki dan memahami aplikasi dimana program tersebut terlibat, Kurangnya pemahaman akan menyebabkan gagalnya perawatan sistem.

Tugas utama dalam pemeliharaan sistem adalah membuat perubahan yang diperlukan pada suatu program. Tugas ini dilakukan oleh programmer aplikasi. Pada dasarnya programmer merespon persyaratan yang menetapkan harapan untuk memperbaiki masalah tersebut. Programmer “*men-debug” (*mengedit) salinan program yang bermasalah. Tidak diadakan suatu perubahan pada program produksi. Hasilnya adalah versi perbaikan dari sebuah program. Kandidat yang artinya kandidat untuk menjadi versi produksi selanjutnya dari program tersebut.

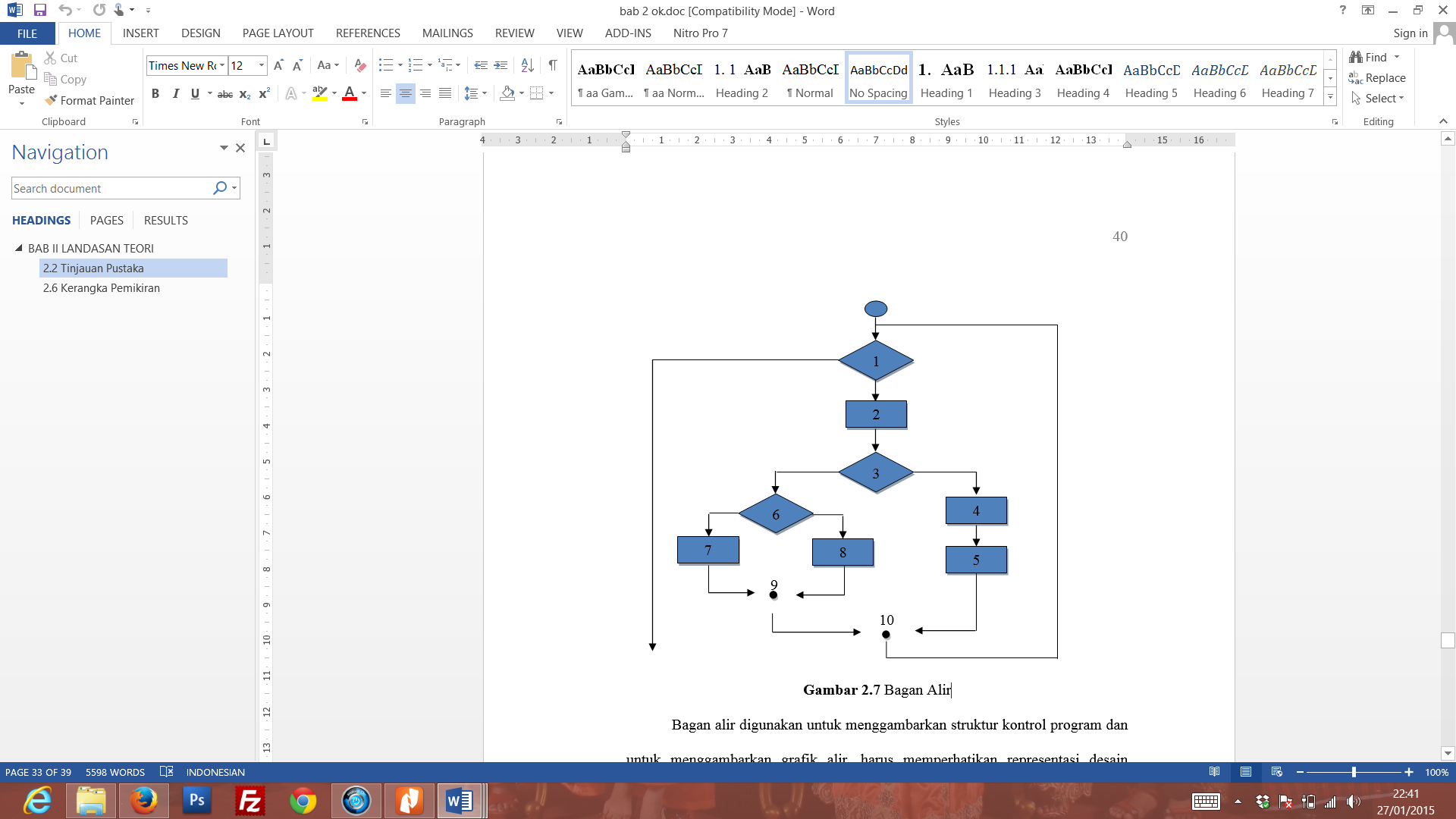


## Teknik Pengujian Sistem



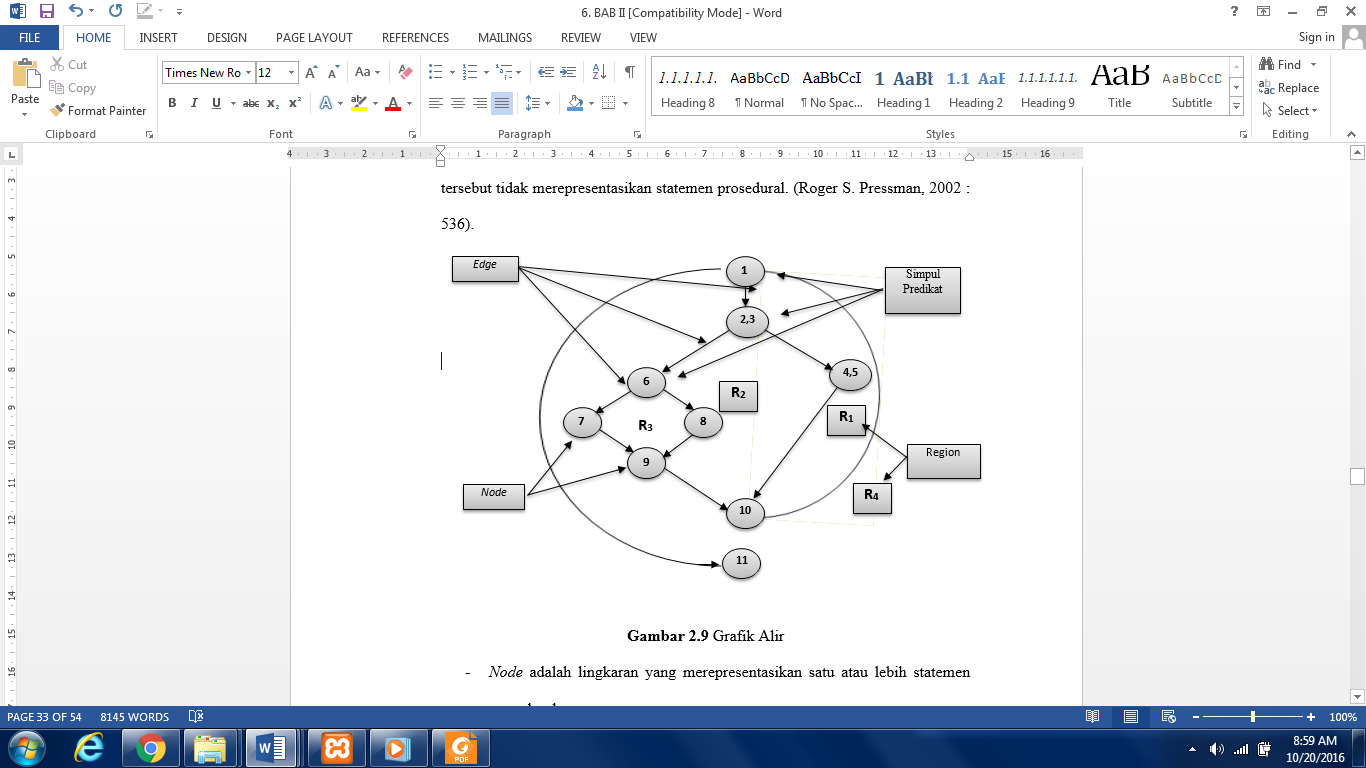
### *White Box*

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 struktur 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 procedural dan menggunakannya sebagai pedoman untuk menentukan basis set dari jalur eksekusi (Roger S. Pressman, 2002:536).



Gambar 2.6 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)



Gambar 2.7 Grafik Alir

*Kompleksitas siklomatis* adalah metriks perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program. Bila metriks ini digunakan dalam konteks metode pengujian *basis path*, maka nilai yang terhitung untuk kompleksitas siklomatis menentukan jumlah *jalur independen*. Jalur independen adalah jalur yang melalui program yang mengintroduksi sedikitnya satu rangkaian statemen proses baru atau suatu kondisi baru. Bila dinyatakan dengan terminologi grafik alir, jalur independen harus bergerak sepanjang paling tidak satu *edge* yang tidak dilewatkan sebelum jalur tersebut ditentukan. Sebagai contoh, serangkaian jalur independen untuk grafik alir yang ditunjukkan pada gambar 2.7 adalah :

Jalur 1 : 1 – 11

Jalur 2 : 1 – 2 – 3 – 4 – 5 – 10 – 1 – 11

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 2.7 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 2.7 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 grafikalir pada gambar 2.7 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.



### *Black Box*

*Black box approach* adalah suatu sistem dimana *input dan output-*nya dapat didefinisikan tetapi prosesnya tidak diketahui atau tidak terdefinisi. Metode ini hanya dapat dimengerti oleh pihak dalam (yang menangani sedangkan pihak luar hanya mengetahui masukan dan hasilnya). Sistem ini terdapat pada subsistem tingkat terendah.

Metode ujicoba *black box* memfokuskan pada keperluan fungsional dari *software*. Karena itu uji coba *black box* memungkinkan pengembang *software* untuk membuat himpunan kondisi *input* yang akan melatih seluruh syarat-syarat fungsional suatu program. Ujicoba *black box* bukan merupakan alternatif dari uji coba *white box*, tetapi merupakan pendekatan yang melengkapi untuk menemukan kesalahan lainnya, selain menggunakan metode *white box*.

Ujicoba *black box* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses *database eksternal*
4. Kesalahan performa
5. Kesalahan inisialisasi dan terminasi

Tidak seperti metode *white box* yang dilaksanakan diawal proses, uji coba *black box* diaplikasikan dibeberapa tahapan berikutnya. Karena uji coba *black box* dengan sengaja mengabaikan struktur kontrol, sehingga perhatiannya difokuskan pada informasi *domain*. Uji coba didesain untuk dapat menjawab pertanyaan-pertanyaan berikut :

1. Bagaimana validitas fungsionalnya di uji ?
2. Jenis *input* seperti apa yang akan menghasilkan kasus uji yang baik ?
3. Apakah sistem secara khusus sensitif terhadap nilai input tertentu ?
4. Bagaimana batasan-batasan kelas data diisolasi ?
5. Berapa rasio data dan jumlah data yang dapat ditoleransi oleh sistem ?
6. Apa akibat yang akan timbul dari kombinasi spesifik data pada operasi sistem?

Dengan mengaplikasikan uji coba *black box*, diharapkan dapat menghasilkan sekumpulan kasus uji yang memenuhi kriteria berikut :

1. Kasus uji yang berkurang, jika jumlahnya lebih dari 1, maka jumlah dari uji kasus tambahan harus didesain untuk mencapai ujicoba yang cukup beralasan.
2. Kasus uji yang memberitahukan sesuatu tentang keberadaan atau tidaknya suatu jenis kesalahan, dari pada kesalahan yang terhubung hanya dengan suatu uji coba yang spesifik.

### 2.4.3 Perangkat Lunak Pendukung

Perangkat lunak pendukung yang digunakan penulis dalam membangun sistem ini yaitu PHP dam MySQL seperti pada tabel di bawah ini :

**Tabel 2.4** Perangkat Lunak Pendukung

|  |  |  |
| --- | --- | --- |
| **NO** | **TOOLS** | **KEGUNAAN** |
| 1 | PHP | Sebuah bahasa *scripting* yang terpasang pada HTML. Yang bertujun untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat. |
| 2 | MySQL | Salah satu pengolah database yang menggunakan SQL (*Strukture Query Language*) sebagai bahan dasar untuk mengakses databasenya. Yang memiliki keuntungan seperti *open source* dan memiliki kemampuan menampung kapasitas yang besar. |



## Kerangka Pemikiran

1. Desain Model
2. Desain User Interface

* Desain Output
* Desain Input
* Desain Menu

1. Desain Database
2. Desain Teknologi

SOLUSI

ANALISIS SISTEM

1. Sistem Berjalan
2. Sistem Diusulkan

DESAIN SISTEM

PEMBANGUNAN SISTEM

IMPLEMENTASI SISTEM

1. White Box Testing
2. Black Box Testing

PENGUJIAN SISTEM

TUJUAN

1. Untuk mengetahui hasil penerapan metode *Naive Bayes* untuk prediksi pendapatan pajak kendaraan bermotor.
2. Untuk memperoleh akurasi yang tepat dalam prediksi pendapatan pajak kendaraan bermotor dengan menggunakan metode *Naive Bayes.*
3. Bagaimana hasil penerapan metode *Naive Bayes* untuk prediksi pendapatan pajak kendaraan bermotor?
4. Seberapa besar tingkat akurasi dalam prediksi pendapatan pajak kendaraan bermotor dengan menggunakan metode *Naive Bayes*?

MASALAH

PELUANG

Adanya kebutuhan untuk prediksi pajak kendaraan bermotor pada Kantor UPTD Samsat Kota Gorontalo

1. PHP
2. MySQL

Kantor UPTD Samsat Kota Gorontalo

Prediksi Pendapatan Pajak Kendaraan Bermotor Menggunakan Metode *Naive Bayes*



**Gambar 2.8** Bagan Kerangka Pemikiran