# **BAB II LANDASAN TEORI**

# **Tinjauan Studi**

Ada beberapa penelitian tentang penerapan Algortima *K-Nearest Neighbor* (K-NN) untuk klasifikasi. Diantaranya sebagai berikut :

1. Penelitian yang dilakukan oleh Rio Setyo Nugroho dan Katon Wijana dengan judul “Program bantu prediksi penjualan barang menggunakan metode K-NN” yang diterbitkan dalam jurnal EKSIS Vol 08 No 02 November 2015. Pada penelitian ini penulis menerapkan metode K-NN yang dapat membantu perencanaan dalam setiap transaksi penjualan sehingga dengan adanya aplikasi ini toko bahan bangunan ini dapat meminimalkan biaya investasi. Penelitian ini menghasilkan sebuah aplikasi program bantu untuk memprediksi penjualan barang. Aplikasi ini bertujuan untuk membantu karyawan khususnya admin untuk memprediksi jumlah penyediaan barang yang akan diambil dari suppplier dengan acuan transaksi penjualan pada bulan-bulan sebelumnya.
2. Penelitian yang dilakukan oleh Aditya Nugroho, (2012). Dengan judul “Analisis Pengaruh Kecepatan Kendaraan Terhadap Umur Rencana Jalan Dengan Menggunakan Metode Analitis (Studi Kasus Ruas Jalan Rembang - Bulu), Metode analitis yang digunakan dalam penelitian ini adalah metode Nottingham Design Method dengan menggunakan bantuan program BISAR (Bitumen Analysis in Roads) 3.0, data-data pendukung seperti data lalu lintas harian rata-rata (LHR), data CBR, data temperature tahunan rata-rata, data kecepatan, dan lain sebagainya diperoleh dari Dinas Bina Marga Jawa Tengah. Data-data yang sudah terkumpul kemudian dianalisis untuk mencari nilai yang dibutuhkan sebagai input ke Program BISAR (Bitumen Analysis in Roads) 3.0. Output dari Program BISAR (Bitumen Analysis in Roads) 3.0 yaitu asphalt mix tensile strain untuk kondisi fatigue (ԑt) dan asphalt mix vertikal strain deformasi (ԑz) yang dipakai untuk menghitung besarnya umur rencana perkerasan jalan. Berdasarkan hasil tentang analisis pengaruh kecepatan kendaraan terhadap umur rencana jalan berdasarkan metode analitis (Nottingham Design Method) dengan alat bantu program BISAR (Bitumen Analysis in Roads) 3.0, bahwa kecepatan kendaraan berpengaruh terhadap umur rencana jalan. Dapat ditunjukkan dengan persamaan regresi, untuk kriteria retak lelah pengaruh kecepatan terhadap umur rencana jalan dalam kondisi kritis yaitu y = 0,000x + 0,012 dan untuk kondisi gagal yaitu y = 0,004x + 0,072. Sedangkan untuk kriteria deformasi, pengaruh kecepatan terhadap umur rencana jalan dalam kondisi kritis yaitu y = 0,001x + 0,294 dan kondisi gagal yaitu y = 0,007x + 2,261.
3. Penelitian yang dilakukan oleh Frans Agum Gumelar, Rekyan Regasari Mardi Putri, Indriati, (2018). Dengan judul Implementasi Fuzzy Time Series Pada Prediksi Harga Daging Di Pasar Kabupaten Malang, dilakukan adalah melakukan peramalan kenaikan harga daging sapi. Sehingga pihak DISPERINDAG dapat mempertimbangkan jumlah harga daging sapi di bulan mendatang berdasarkan hasil ramalan. Proses peramalkan didasarkan pada data data historis yang sudah ada. Peramalan ini disebut peramalan data time series. Peramalan data time series lebih menekankan pada relasi antar data-data. Metode yang digunakan untuk peramalan adalah Fuzzy Time Series (FTS). Berdasarkan hasil pengujian menggunakan 21 data harga daging di Kabupaten Malang pada tahun 2016 dan 2017, akurasi yang didapat dari peramalan sebesar 57%. Dengan nilai eror terkecil terletak di bulan juni 2017 sebesar 16,129 dan nilai eror terbesar terletak di bulan maret 2016 sebesar 65,610,000.

# **Tinjauan Pustaka**

## **Jalan**

Definisi jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap, dan perlengkapannya yang diperuntukkan bagi lalu lintas, yang berada permukaan tanah, diatas permukaan tanah, dibawah permukaan tanah dan atau air, serta diatas permukaan air, kecuali jalan kereta api dan jalan kabel (UU No. 38 tahun 2004 tentang Jalan). Jalan umum adalah jalan yang diperuntukkan bagi lalu lintas umum, jalan khusus adalah jalan yang dibangun oleh instansi, badan usaha, perseorangan, atau kelompok masyarakat untuk kepentingan sendiri. Bagian-bagian jalan meliputi ruang manfaat jalan, ruang milik jalan, dan ruang pengawasan jalan :

1. Ruang manfaat jalan meliputi badan jalan, saluran tepi jalan, dan ambang pengamannya.
2. Ruang milik jalan meliputi ruang manfaat jalan dan sejalur tanah tertentu diluar ruang manfaat jalan.
3. Ruang pengawasan jalan merupakan ruang tertentu diluar ruang milik jalan yang ada dibawah pengawasan penyelenggara jalan.

Adapun ktiteri yang akan digunakan adalah : 1). Jumlah kenderaan yang lewat, 2). Kualitas aspal, dan 3). Beban kenderaan.

**Tabel 2.1.** Data Set

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nama Jalan** | **Tahun** | **Kualitas Aspal** | **Jumlah Kenderaan** | **Beban Kenderaan** | **Output** |
| Jl. Delima | 2015 | 96,55 | 162.000 | 56.911.680 | 10 Thn |
| Jl. Sawit | 2015 | 96,34 | 29.160 | 2.592.000 | 12 Thn |
| Jl. Beringin | 2015 - 2016 | 93,37 | 336.960 | 89.257.680 | 10 Thn |
| Jl. Rambutan | 2015 – 2016 | 96,21 | 372.600 | 32.510.160 | 10 Thn |
| Jl. Benteng Otanaha | 2016 | 94,57 | 65.880 | 6.272.640 | 12 Thn |
| Jl. Tinalgo | 2016 | 96,22 | 56.160 | 39.057.120 | 12 Thn |
| Jl. Jambura | 2016 | 96,84 | 18.360 | 1.477.440 | 12 Thn |
| Jl. Kapuas | 2016 | 95,64 | 28.080 | 2.371.680 | 12 Thn |
| Jl. Mandala | 2016 | 91,15 | 25.920 | 2.067.120 | 12 Thn |
| Jl. Anton Sujarwo | 2016 | 84,35 | 72.360 | 5.346.000 | 10 Thn |
| Jl. Anggur | 2016 | 94,87 | 43.200 | 3.551.040 | 12 Thn |
| Jl. Taman Ria | 2016 | 96,79 | 24.840 | 2.073.600 | 12 Thn |
| Jl. Taman Surya I | 2016 | 96,55 | 41.040 | 3.427.920 | 12 Thn |
| Jl. Mahakam | 2016 | 97,23 | 24.840 | 2.028.240 | 12 Thn |
| Jl. Madura | 2016 | 94,66 | 263.520 | 31.786.560 | 10 Thn |
| Jl. Rusli Datau I | 2016 | 94,41 | 20.520 | 1.691.280 | 10 Thn |
| Jl. Limehu – Bongo | 2016 – 2017 | 52,17 | 22.680 | 1.814.400 | 5 Thn |
| Jl. Kasuari | 2017 | 98,88 | 77.760 | 32.231.520 | 10 Thn |
| Jl. Bandes | 2017 | 98,73 | 27.000 | 2.196.720 | 10 Thn |
| Jl. Makassar | 2017 | 87,32 | 39.960 | 3.434.400 | 10 Thn |
| Jl. Tanjung Kramat | 2017 | 98,67 | 21.600 | 22.229.120 | 5 Thn |

Sumber : Dinas PU dan Dinas Perhubungan Kota Gorontalo, 2018

# ***Data Mining***

*Data mining* (Connolly dan Begg, 2010) adalah suatu proses ekstraksi atau penggalian data yang belum diketahui sebelumnya, namun dapat dipahami dan berguna dari database yang besar serta digunakan untuk membuat suatu keputusan bisnis yang sangat penting. *Data mining* (Segall et.all, 2012) biasa juga disebut dengan “Data atau *knowledge discovery*” atau menemukan pola tersembunyi pada data. Data mining adalah proses dari menganalisa data dari prespektif yang berbeda dan menyimpulkannya ke dalam informasi yang berguna.

Pada prosesnya data mining akan mengekstrak informasi yang berharga dengan cara menganalisis adanya pola-pola ataupun hubungan keterkaitan tertentu dari data-data yang berukuran besar. Data mining berkaitan dengan bidang ilmu-ilmu lain, seperti *Database System, Data Warehousing, Statistic, Machine Learning, Information Retrieval*, dan komputasi tingkat tinggi. Selain itu data mining didukung oleh ilmu lain seperti *Neural Network,* Pengenalan Pola*, Spatial Data Analysis, Image Database, Signal Processing.*

Beberapa survey tentang proses pemodelan dan metodologi menyatakan bahwa, data mining digunakan sebagai penunjuk, dimana data mining menyajikan intisari atas sejarah, deskripsi dan sebagai standar petunjuk mengenai masa depan dari sebuah proses model data mining (Mariscal, Marba’n dan Ferna’ndes, 2010).

## **Proses Dalam Tahapan *Data Mining***

Proses tahap data mining terdiri dari tiga langkah Utama (Maimon & Rokach, 2010) yaitu :



**Gambar 2.1** Langkah – Langkah Proses Dalam *Data Mining*

(Maimon & Rokach, 2010)

1. *Data Preparation*

Pada langkah ini, data dipilih, dibersihkan, dan dilakukan preprocessed mengikuti pedoman dan knowledge dari ahli domain yang menangkap dan mengintegrasikan data internal dan eksternal ke dalam tinjauan organisasi secara menyeluruh.

1. Algoritma *Data Mining*

Penggunaan algoritma data mining dilakukan pada langkah ini untuk menggali data yang terintegrasi untuk memudahkan identifikasi informasi bernilai.

1. *Fase* Analisa Data

Keluaran dari data mining dievaluasi untuk melihat apakah knowledge domain ditemukan dalam bentuk rule yang telah diekstrak dari jaringan.

## **Karakteristik *Data Mining***

Karakteristik data mining sebagai berikut (Mariscal, Marba’n dan Ferna’ndes, 2010) :

1. Data mining berhubungan dengan penemuan sesuatu yang tersembunyi dan pola data tertentu yang tidak diketahui sebelumnya.
2. Data mining biasa menggunakan data yang sangat besar. Biasanya data yang besar digunakan untuk membuat hasil lebih dapat dipercaya.
3. Data mining berguna untuk membuat keputusan kritis.

Berdasarkan beberapa pengertian tersebut dapat ditarik kesimpulan bahwa Data Mining adalah suatu teknik menggali informasi berharga yang terpendam atau tersembunyi pada suatu koleksi data (database) yang sangat besar sehingga ditemukan suatu pola yang menarik yang sebelumnya tidak diketahui.

## **Fungsi Data Mining**

Teknik data mining telah digunakan untuk menemukan pola yang tersembunyi dan meprediksi tren masa depan. Dan keuntungan kompetitif dari data mining termasuk dengan meningkatnya pendapatan, berkurangnya pengeluaran, dan kemampuan pemasaran yang meningkat. (Pujari et. All, 2012)

Data mining dibagi menjadi dua kategori utama (Pujari et. All, 2012) yaitu:

1. Prediktif

Tujuan dari tugas prediktif adalah untuk memprediksi nilai dari atribut tertentu berdasarkan pada nilai atribut-atribut lain. Atribut yang diprediksi umumnya dikenal sebagai target atau variable tak bebas, sedangkan atribut-atribut yang digunakan untuk membuat prediksi dikenal sebagai explanatory atau variable bebas.

1. Deskriptif

Tujuan dari tugas deskriptif adalah untuk menurunkan pola-pola (korelasi, trend, cluster, teritori, dan anomali) yang meringkas hubungan yang pokok dalam data. Tugas data mining deskriptif sering merupakan penyelidikan dan seringkali memerlukan teknik post-processing untuk validasi dan penjelasan hasil.

Fungsi dari data mining juga ada dalam dunia kesehatan, dimana data mining telah digunakan untuk untuk meningkatkan diagnosis dan pengobatan atau lebih mengerti perilaku dari pasien. (Pujari et. All, 2012).

Data mining juga memiliki beberapa fungsionalitas yaitu *Concept/Class Description: Characterization and Discrimination, Mining Frequent Patterns, Associations, and Correlations, Classification and Prediction, Cluster Analysis, Outlier analysis, dan Evolution analysis*. (Mariscal, Marba’n dan Ferna’ndes, 2010)

## **Tujuan Data Mining**

Tujuan dari data mining (Mariscal, Marba’n dan Ferna’ndes, 2010) adalah:

1. *Explanatory*

Untuk menjelaskan beberapa kondisi penelitian, seperti mengapa penjualan truk pick-up meningkat di Colorado.

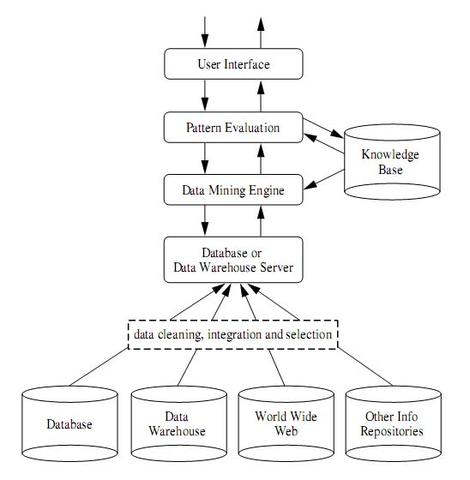
1. *Confirmatory*

Untuk mempertegas hipotesis, seperti halnya dua kali pendapatan keluarga lebih suka dipakai untuk membeli peralatan keluarga dibandingkan dengan satu kali pendapatan keluarga.

1. *Exploratory*

Untuk menganalisa data yang memiliki hubungan yang baru. Misalnya, pola apa yang cocok untuk kasus penggelapan kartu kredit.

## **Arsitektur *Data Mining***



**Gambar 2.2** Arsitektur Sistem *Data Mining*

Data mining merupakan proses pencarian pengetahuan yang menarik dari data berukuran besar yang disimpan dalam basis data, data warehouse atau tempat penyimpanan informasi lainnya. Dengan demikian arsitektur sistem data mining memiliki komponen-komponen utama (Mariscal, Marba’n dan Ferna’ndes, 2010) yaitu:

1. Database, data warehouse, World Wide Web, atau tempat penyimpanan informasi lainnya: bisa berbentuk satu atau banyak database, data warehouse, spreadsheet, ataupun tempat penyimpanan informasi lainnya. Data Cleaning, Data Integration dan Data Selection dapat dijalankan pada data tersebut.
2. Database dan data warehouse server. Komponen ini bertanggung jawab dalam pengambilan data yang relevan, berdasarkan permintaan pengguna.
3. Knowledge Based. Komponen ini merupakan domain knowledge yang digunakan untuk memandu pencarian atau mengevaluasi pola-pola yang dihasilkan. Pengetahuan tersebut meliput hirarki konsep yang digunakan untuk mengorganisasikan atribut atau nilai atribut kedalam level abstraksi yang berbeda. Pengetahuan tersebut juga dapat berupa kepercayaan pengguna (user belief), yang dapat digunakan untuk menentukan kemenarikan pola yang diperoleh.
4. Data mining engine. Bagian ini merupakan komponen penting dalam arsitektur sistem data mining. Komponen ini terdiri dari modul-modul fungsional seperti karakterisasi, asosiasi, klasifikasi, dan analisis cluster.
5. Ghrapical user interface (GUI). Modul ini berkomunikasi dengan pengguna dan data mining. Melalui komponen ini, pengguna berinteraksi dengan sistem menggunakan query.
   * 1. **Prediksi (*Forecasting*)**

*Forecasting* adalah prediksi (perkiraan) mengenai sesuatu yang belum terjadi. Forecasting atau peramalan adalah memperkirakan sesuatu pada waktu-waktu yang akan datang berdasarkan data masa lampau yang dianalisis secara ilmiah. Prediksi merupakan kegiatan untuk memperkirakan apa yang akan terjadi pada masa yang akan datang. Proses perkiraan pengukuran berdasarkan data yang relevan dengan masa lalu dan dianalisis secara ilmiah menggunakan metode statistika yang bertujuan untuk memperbaiki peristiwa yang akan terjadi pada masa yang akan datang. Dengan kata lain prediksi bertujuan untuk mendapatkan perkiraan yang bisa meminimumkan kesalahan prediksi (*forecast error*) yang biasanya diukur dengan *Standard Error Estimate* (SEE), *Mean Absolute PercentError* (MAPE) dan sebagainya

# **Algoritma *K-Nearest Neighbors (K-NN)***

Algoritma *K*-*NearestNeighbor* (K-NN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan data uji. *K-Nearest Neighbor* termasuk kelompok *instance*-*based learning*. Algoritma ini juga merupakan salah satu tehnik *lazylearning* dikarenakan hanya menyimpan sebagian atau seluruh data latih, kemudian menggunakan data latih tersebut ketika proses prediksi. *K-Nearest Neighbor* dilakukan dengan mencari kelompok *k* objek dalam data *training* yang paling dekat (mirip) dengan objek pada data baru atau data *testing* (Wu, 2009). Adapun K-Nearest Neighbor termasuk kategori *memory based method*, yaitu seluruhnya atau sebagian dari *trainingset* tetap disimpan dan dipakai dalam proses klasifikasi.

Pada proses pengklasifikasian, algoritma ini tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Algoritma K-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi sampel dari uji yang baru. Pengukuran jarak ketetanggan yang digunakan seperti :*Euclidean Distance*, *Mikwoski Distance*, dan *ManhattamDistance*. Dan yang paling sering digunakan adalah *Euclidean Distance*.

Pada fase *training*, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi data *trainingsample*. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk *testing* data (klasifikasinya belum diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor *trainingsample* dihitung, dan sejumlah *k* buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut.

Nilai *k* yang terbaik untuk algoritma ini tergantung pada data, secara umumnya, nilai *k* yang lebih tinggi akan mengurangi efek *noise* pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Secara umum nilai *k* optimal yang sering digunakan berkisar diantara 3-10 atau dimana *n* merupakan jumlah data latih. Itu akan menghasilkan hasil yang lebih baik dibandingkan dengan 1NN.

Adapun penerapan algoritma K-Nearest Neighbor terdiri dari empat langkah, yaitu :

1. Menentukan parameter *k* (jumlah ketetanggaan yang paling dekat).
2. Menghitung jarak ketetanggaan (*similaritymeasure*) masing-masing objek terhadap data sampel yang diberikan.
3. Mengurutkan objek-objek tersebut kedalam kelompok yang mempunyai jarak terkecil sampai terbesar.
4. Mengumpulkan kategori Y (Klasifikasi *NearestNeighbor*).
5. Dengan menggunakan kategori *Nearest Neighbor* yang paling mayoritas maka dapat diprediksi nilai *queryinstance* yang telah dihitung.

*K-Nearest Neighbor* memiliki beberapa kelebihan yaitu tangguh terhadap data *template* yang *noisy* dan efektif apabila jumlah data template besar. Sedangkan kelemahan dari metode ini adalah perlunya menentukan nilai parameter *k* (jumlah ketetanggaan terdekat), pembelajaran berdasarkan jarak tidak jelas mengenai jenis jarak apa yang harus digunakan dan atribut mana yang harus digunakan untuk mendapatkan hasil yang terbaik, dan biaya komputasi yang cukup tinggi karena diperlukan perhitungan jarak dari tiap *queryinstance* pada keseluruhan *trainingsample*.

**Contoh Penerapan K-NN :**

Terdiri dari 2 attribut dengan skala kuantitatif sebagai data latih yaitu x1 dan x2 serta Y yaitu kelas baik dan buruk seperti berikut :

Tabe 2.2 Data Latih

|  |  |  |
| --- | --- | --- |
| **X1** | **X2** | **Y** |
| 7 | 7 | Buruk |
| 7 | 4 | Buruk |
| 3 | 4 | Baik |
| 1 | 4 | Baik |

Terdapat data uji yaitu X1=3 dan X2=7, tentukan nilai Y !

Langkah penyelesaian :

1. Tentukan parameter *k* jumlah tetangga terdekat (misal *k*=3)
2. Menghitung jarak antara data baru dengan semua data latih

Tabel 2.3 Perhitungan Kuadrat Jarak Data Latih Dengan Data Uji

|  |  |  |
| --- | --- | --- |
| **XI** | **X2** | **Kuadrat jarak dengan data baru (3,7)** |
| 7 | 7 | (7-3)2 + (7-7)2 = 16 |
| 7 | 4 | (7-3)2 + (4-7)2 = 25 |
| 3 | 4 | (3-3)2 + (4-7)2 = 9 |
| 1 | 4 | (1-3)2 + (4-7)2 = 13 |

1. Urutkan hasil kuadrat jarak secara ascending dan tetapkan tetangga terdekat berdasarkan nilai *k* (*k*=3)

Tabel 2.4 Penentuan 3 Tetangga Terdekat Dari Data Uji

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **X1** | **X2** | **Euclidean** | **Peringkat jarak** | **Tetangga terdekat** |
| 7 | 7 |  | 3 | Ya |
| 7 | 4 |  | 4 | Tidak |
| 3 | 4 |  | 1 | Ya |
| 1 | 4 |  | 2 | Ya |

Tabel 2.5 Klasifikasi Kelas Tetangga Terdekat

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **X1** | **X2** | **Euclidean** | **Peringkat jarak** | **Tetangga terdekat** | **Y** |
| 7 | 7 |  | 3 | Ya | Buruk |
| 7 | 4 |  | 4 | Tidak | - |
| 3 | 4 |  | 1 | Ya | Baik |
| 1 | 4 |  | 2 | Ya | Baik |

1. Berdasarkan hasil dari tabel diatas dimana nilai *k*=3, mayoritas hasil Y yang diperoleh yaitu “baik”. Jadi prediksi data uji dengan nilai X1=3 dan X2=7 adalah Y = kelas baik.

# **Euclidean Distance**

*Eunclidean distance* adalah metrika yang paling sering digunakan untuk menghitung kesamaan dua vektor. Jarak *Euclidean* menghitung akar dari kuadrat perbedaan 2 vektor (*root of squere differences between 2 vector*). Persamaan dari jarak *Euclidean* adalah (Putra, 2009) :

Dengan D adalah jarak antara titik pada *trainingx* dan titik data *testing y* yang akan diklasifikasi, dimana x = x1,x2,x3,..,xi dan y = y1,y2,y3,..,yi dan k merepresentasikan nilai atribut dan k merupakan dimensi atribut.

**Contoh** : jika terdapat 2 vektor : dan , maka jarak *Euclidean* dari 2 vektor tersebut adalah :

## **Keuntungan dan Kelemahan Algoritma K-Nearset Neigbors (k-NN)**

Kelebihan dari Algoritma *k-Nearset Neigbor* (k-NN) :

1. Lebih efektif di data yang besar.
2. Dapat menghasilkan data yang lebih akurat.

Kekurangan dari Algoritma *k-Nearset Neigbor* (k-NN) :

1. Perlu untuk menentukan nilai k yang optimal sehingga untuk menyatakan jumlah tatangga terdekatnya lebih mudah.
2. Biaya komputasi yang cukup tinggi karena perhitungan jarak harus dilakukan pada setiap query instance.

# **Siklus Pemgembangan Sistem**

Proses pengembangan sistem melewati beberapa tahapan dari mulai sistem itu direncanakan sampai dengan sistem tersebut diterapkan, dioperasikan dan dipelihara. Bila operasi yang sudah dikembangkan masih timbul kembali permasalahan-permasalahan yang tidak dapat diatasi dalam tahap pemeliharaan, 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).* Daur atau siklus hidup dari pengembangan sistem merupakan suatu bentuk yang digunakan untuk menggambarkan tahapan utama dan langkah-langkah didalam tahapan tersebut dalam proses pengembangannya.

Berikut langkah-langkah yang digunakan :

***Desain Sistem***

Perancangan Konseptual

Perancangan Fisik

***Operasi dan Pemeliharaan***

***Analisis Sistem***

Studi Kelayakan

Analisis Kebutuhan

***Implementasi Sistem***

Pemilihan dan Pelatihan Personil Instalasi Perangkat Keras dan RPL Pemrograman dan Pengujian

Kebutuhan Sistem

Perubahan

Desain Sistem

Lingkup/ Kebutuhan

Sistem Siap Beroperasi Mandiri

beroperasi mandiri

Kesalahan atau masalah

yang tidak memungkinkan

implementasi dilaksanakan

Implementasi kurang lengkap / ada permintaan baru

**Gambar 2.3.** Siklus Hidup Pengembangan Sistem

* + - 1. **Perencanaan Sistem**

Kebijakanuntuk mengembangkan sistem informasi dilakukan oleh manajemen puncak karena menginginkan untuk meraih kesempatan-kesempatan yang ada yang tidak dapat diraih oleh sistem lama atau sistem yang lama mempunyai banyak kelemahan-kelemahan yang perlu diperbaiki. Setelah manajemen puncak menetapkan kebijakan untuk mengembangkan sistem informasi, Sebelum sistem ini sendiri dikembangkan, maka perlu direncanakan terlebih dahulu dengan cermat. Perencanaan sistem ini menyangkut estimasi dari kebutuhan-kebutuhan fisik, tenaga kerja, dan dana yang dibutuhkan untuk mendukung pengembangan sistem ini serta untuk mendukung operasinya setelah diterapkan.

Selama fase perencanaan sistem, hal yang perlu dipertimbangkan adalah :

1. Faktor-Faktor Kelayakan (*Feasibility Factors*)yang berkaitan dengan kemungkinan berhasilnya sistem informasi yang dikembangkan dan digunakan.
2. Faktor-Faktor Strategis (*Strategic Factors*)yang berkaitan dengan pendukung sistem informasi dari sasaran bisnis dipertimbangkan untuk setiap proyek yang diusulkan. Nilai-nilai yang dihasilkan dievaluasi untuk menentukan proyek sistem mana yang akan menerima prioritas yang tertinggi.
   * + 1. **Analisis Sistem**

Tahapan analisis sistem dimulai karena adanya permintaan terhadap sistem baru. Permintaan bisa datang dari seorang Pimpinan/Manajer di luar departemen sistem informasi yang melihat adanya masalah atau menemukan adanya peluang baru. Namun, adakalanya inisiatif pengembangan sistem baru berasal dari bagian yang bertanggung jawab terhadap pengembangan sistem informasi. Tujuan utama dari analisis sistem adalah menentukan hal-hal secara detail yang akan dikerjakan oleh sistem yang diusulkan.

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

1. *Identify,* mengidentifikasi (mengenal) masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah dapat di definisikan sebagai suatu pertanyaan yang di inginkan untuk dipecahkan. Tahap identifikasi masalah sangat penting karena akan menentukan keberhasilan pada langkah-langkah selanjutnya.
2. *Understand,* 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.
3. *Analyze,* menganalisis sistem tanpa report.
4. *Report,* yaitu membuat laporan hasil analisis. Tujuan utama dari pembuatan laporan hasil analisis yaitu pelaporan bahwa analisis telah selesai dilakukan.
   * + 1. **Desain Sistem**

Dalam desain sistem, dibutuhkan alat bantu desain. Dalam tahapan ini, pengembang sistem bisa menentukan arsitektur sistemnya, merancang gambaran konseptual sistem, merancang database, perancangan interface, hingga membuat flowchart program. Salah satu alat bantu yang bisa digunakan dalam pembuatan sistem bantu keputusan adalah *Data Flow Diagram* (DFD). DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan asal data dan tujuan data yang keluar dari sistem, tempat penyimpanan data, proses apa yang menghasilkan data tersebut, serta interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

Menurut John Burch dan Gary Grudnitski, desain sistem dapat didefinisikan sebagai penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

Tahap desain sistem mempunyai dua tujuan utama :

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

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

1. **Desain Sistem Secara Umum (*general systems design***)

Pada tahap desain secara umum, komponen-komponen sistem informasi yang dirancang dengan tujuan dikomunikasikan kepada user bukan untuk pemrograman. Komponen sistem informasi yang di desain adalah model, output, input, database, teknologi, dan kontrol.

**a. Desain Model Secara Umum**

Analisis sistem dapat mendesain model dari sistem informasi yang di usulkan dalam bentuk *physical* sistem dan *logical* model. Bagan alir sistem merupakan alat yang tepat digunakan untuk menggambarkan *physical systems,* logical model dapat digambar dengan diagram arus data.

Bagan alir sistem merupakan bagan yang menunjukan arus pekerjaan secara keseluruhan dari sistem. Bagan alir sistem digambar dengan simbol-simbol sebagai berikut :

**Tabel 2.6** 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 |
| 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 magnetik |
| 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 |
| 20. | Simbol Penghubung |  | Menunjukkan penghubung ke halaman yang masih sama atau ke halaman yang lain |

Sumber: Jogiyanto HM, 2014.

Untuk mempermudah penggambaran suatu sistem yang ada atau sistem yang baru yang akan dikembangkan secara logika dan tanpa memperhatikan lingkungan fisik data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan, maka digunakan Diagram Arus Data (DAD) atau *Data Flow Diagram* (DFD).

**Tabel 2.7** Daftar Simbol Diagram Alir Dokumen

Simbol proses, menunjukan transformasi dari masukan menjadi keluaran

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

Aliran atau arus data, menggambarkan gerakan paket data atau informasi dari suatu bagian ke bagian yang lain, dimana penyimpanan mewakili lokasi penyimpanan data.

Penyimpanan, digunakan untuk memodelkan kumpulan data atau paket data

(Sumber : Jogiyanto : 2014)

**b. Desain *Output* Secara Umum**

Output adalah produk dari sistem informasi yang dapat dilihat. Output terdiri dari macam-macam jenis seperti hasil di media kertas, dan hasil di media lunak. Disamping itu output dapat berupa hasil dari suatu proses yang akan digunakan oleh proses lain dan tersimpan di suatu media seperti tape, disk, atau kartu. Yang dimaksud dengan output pada tahap desain ini adalah output yang berupa tampilan di media kertas atau di layar video.

**c.** **Desain*Input* Secara Umum**

Alat input dapat digolongkan ke dalam 2 golongan, yaitu alat input langsung *(online input device)* dan alat input tidak langsung *(offline input device).* Alat input langsung merupakan alat input yang langsung dihubungkan dengan CPU, sedangkan alat input tidak langsung adalah alat input yang tidak langsung dihubungkan dengan CPU.

**d. Desain *Database* Secara Umum**

Basis data (database) adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diluar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya. Sistem basis data 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.

**2. DesainSistem Secara Rinci (*Detailed systems design***)

**a. 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.

1. Desain output dalam bentuk laporan : dimaksudkan untuk menghasilkan output dalam bentuk laporan dimedia kertas. Bentuk laporan yang paling banyak digunakan adalah dalam bentuk tabel dan berbentuk grafik atau bagan.
2. Desain output dalam bentuk dialog layar terminal : merupakan rancang bangun dari percakapan antara pemakai sistem atau user dengan komputer. Percakapan ini dapat terdiri dari proses memasukkan data ke sistem, menampilkan output informasi kepada user, atau keduanya.

**b. Desain *Input* Terinci**

Masukan merupakan awal dimulainya proses informasi. Bahan mentah dari informasi adalah data yang terjadi dari transaksi-transaksi yang dilakukan oleh organisasi. Data hasil dari transaksi merupakan masukan untuk sistem informasi. Hasil dari sistem informasi tidak lepas dari data yang dimasukan. Desain input terinci dimulai dari desain dokumen dasar sebagai penangkap input yang pertama kali. Jika dokumen dasar tidak di desain 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 dan ditangkap.
2. Dapat dicatat dengan jelas, konsisten, dan akurat.
3. Dapat mendorong lengkapnya data, disebabkan data yang dibutuhkan disebutkan satu persatu di dalam dokumen dasarnya.

**c. Desain *Database* Terinci**

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

* + - 1. **Seleksi Sistem**

Tahap ini merupakan tahap untuk memilih perangkat yang akan digunakan untuk sistem informasi. Pengetahuan dibutuhkan oleh pemilih sistem diantaranya adalah pengetahuan tentang siapa yang menyediakan teknologi ini, cara pemilikannya, dan sebagainya. Pemilihan sistem yang harus paham dengan teknik-teknik evaluasi untuk menyelesaikan sistem.

* + - 1. **Implementasi Sistem**

Implementasi sistem merupakan tahapan untuk meletakkan sistem supaya siap untuk dioperasikan. Pada tahapan ini terdapat banyak aktifitas yang dilakukan, yaitu :

1. Pemrograman dan pengetesan program

Pemrograman merupakan kegiatan menulis program yang akan dieksekusi oleh komputer. Kode program harus berdasarkan dokumentasi yang disediakan oleh analis sistem hasil dari desain sistem.

1. Instalasi perangkat keras dan lunak

Proses pemasangan perangkat keras dan instalasi perangkat lunak yang sudah ada.

1. Pelatihan kepada pemakai

Manusia merupakan faktor yang diperlukan dalam sistem informasi. Jika ingin sukses dalam sistem informasi, maka personil-personil yang terlibat harus diberi pengertian dan pengetahuan tentang sistem informasi dan posisi serta tugas mereka.

1. Pembuatan dokumentasi

Dokumentasi adalah melakukan pencatatan terhadap setiap langkah pekerjaan pembuatan sebuah program yang dilakukan dari awal sampai selesai.

* + - 1. **Perawatan Sistem**

Perawatan sistem informasi adalah suatu upaya untuk memperbaiki, menjaga, menanggulangi, mengembangkan sistem yang ada. Perawatan ini di perlukan untuk meningkatkan efisiensi dan efektivitas kinerja sistem yang ada agar dalam penggunaannya dapat optimal. Beberapa alasan mengapa kita perlu memelihara sistem yang ada yaitu: agar dapat meningkatkan sistem / kinerja sistem, dan menyesuaikan dengan perkembangan, agar sistem yang ada tidak tertinggal.

Aplikasi yang professional dalam SDLC dan teknik maupun perangkat modeling yang mendukungnya adalah hal-hal keseluruhan yang terbaik yang dapat seseorang lakukan untuk meningkatkan maintainabilitas sistem.

Jenis – jenis perawatan sistem meliputi :

* 1. Perawatan korektif : adalah pemeliharaan yang mengkoreksi kesalahan-kesalahan yang ditemukan pada sistem, pada saat sistem di jalankan/berjalan.
  2. Pemeliharaan adaptif : yaitu pemelihaaan yang bertujuan untuk menyesuaikan perubahan yang terjadi.
  3. Pemeliharaan perfektif :pemeliharaan ini bertujuan untuk meningkatkan cara kerja suatu sistem.
  4. Pemeliharaan preventif : pemeliharaan ini bertujuan untuk menangani masalah-masalah yang ada.

# **Pengujian Sistem**

## **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.

**9**

**10**

**6**

**7**

**8**

**1**

**2**

**3**

**5**

**4**

**11**

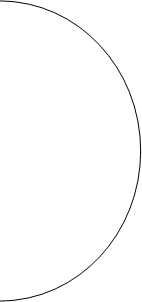
**Gambar 2.4.** Bagan Alir

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.

Edge

Simpul Predikat

Node



Region

R1

R4

R2

**Gambar 2.5**Grafik Alir

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.
3. 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. Grafikalirmempunyai 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 2.3 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.

## **Black Box**

Pengujian *Blackbox* aproach adalah Suatu sistem dimana input dan outputnya dapat didefinisikan tetapiprosesnya 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 uji coba *black box* memfokuskan pada keperluan fungsional dari software. Karena itu ujicoba *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 ujicoba *whitebox*, 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, ujicoba *black box* diaplikasikan dibeberapa tahapan berikutnya. Karena uji coba *black box* dengan sengaja mengabaikan struktur kontrol, sehingga perhatiannya difokuskan pada informasi *domain*. Ujicoba didesain untuk dapat menjawab pertanyaan-pertanyaan berikut :

1. Bagaimana validitas fungsionalnya diuji?
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 ujicoba *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 ujicoba yang spesifik.

# ***Software* Pendukung**

Perancangan Perangkat lunak yang digunakan pada system ini :

**Table 2.8**Perangkat Pendukung Sistem

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Perangkat Lunak** | **Kegunaan** |
| 1 | PHP | Untuk membuat halaman web yang dinamis, yang hasilnya dikirimkan ke client tempat pemakai menggunakan browser |
| 2 | My SQL | Dapat menampung kapasitas yang besar, sehingga MySQL menjadi database yang popular hingga saat ini |

# **Kerangka Pikir**

1. Bagaimana hasil penerapan metode *K-Nearest Neighbor* (K-NN) untuk prediksi usia jalan di Kota Gorontalo?
2. Apakah sistem yang direkayasa dapat di implementasikan sebagai sebuah sistem pada Kantor Dinas Pekerjaan Umum Kota Gorontalo?

**MASALAH**

Dinas Pekerjaan Umum Kota Gorontalo membutuhkan sistem yang dapat memprediksi usia jalan di Kota Gorontalo

**PELUANG**

Diagram Alir Dokumen

**ANALISIS SISTEM**

1. White Box
2. Black Box

**PENGUJIAN SISTEM**

Dinas Pekerjaan Umum Kota Gorontalo

**IMPLEMENTASI SISTEM**

Implementasi Data Mining Untuk Prediksi Usia Jalan Di Kota Gorontalo Menggunakan Metode *K-Nearest Neighbor* (K-NN)

**SOLUSI**

1. PHP
2. MySQL

**PEMBANGUNAN SISTEM**

1. Untuk mengetahui hasil penerapan metode *K-Nearest Neighbor* (K-NN) untuk Prediksi usia jalan di Kota Gorontalo.
2. Agar sistem yang direkayasa dapat di implementasikan sebagai sistem pada Kantor Dinas Pekerjaan Umum Kota Gorontalo

**TUJUAN**

1. Desain Model
2. Desain Output
3. Desain Input
4. Desain Basis Data
5. Desain Teknologi

**DESAIN SISTEM**

**Gambar : 2.6** Kerangka Pikir