**PROPOSAL PENELITIAN**

**PERANCANGAN APLIKASI SISTEM PAKAR BERBASIS WEB UNTUK MENDIAGNOSA PENYAKIT KULIT PADA ANJING MENGGUNAKAN METODE BAYES**

****

**OLEH :**

**EKA PRAJA WIYATA MANDALA (KETUA)**

**MUHAMMAD AFDHAL (ANGGOTA 1)**

**SAPUTRA RAMADHONI (ANGGOTA 2)**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PUTRA INDONESIA “YPTK” PADANG**

**2019**

**RINGKASAN**

Anjing merupakan hewan yang mudah beradaptasi dan dapat menjadi teman baik bagi manusia. Kecerobohan pemilik dalam menjaga dan merawat anjing dapat mengakibatkan kematian bagi anjing. Salah satu penyakit yan sering di jumpai adalah penyakit kulit pada anjing. Pemahaman masyarakat akan penyakit kulit pada anjing masih rendah, sehingga masih banyak masyarakat yang masih mengandalkan keahlian dari pakar secara manual. Selain itu, biaya untuk pengobatan ke dokter hewan sangatlah mahal dan keberadaan dokter hewan masih sangat sedikit. Oleh karena itu, dibuatlah Aplikasi Sistem Pakar Diagnosa Penyakit Kulit Pada Anjing yang dapat digunakan untuk membantu mendeteksi adanya penyakit kuit pada tubuh anjing. Aplikasi ini berbasis web yang dapat diakses setiap saat sehingga mendukung pemeriksaan rutin yang bisa dilakukan oleh diri sendiri. Tugas akhir ini dibuat dengan menggunakan metode *Bayes*, dan diimplementasikan dengan menggunakan bahasa pemrograman PHP, dan database MySQL berbasis web. Dengan adanya aplikasi ini, pengetahuan yang dimiliki oleh pakar mengenai penyakit kulit pada anjing dapat tersusun dalam bentuk basis data, dan diharapkan dapat membantu pasien dalam mendiagnosa penyakit kulit pada anjing secara dini.

**Kata kunci** : Sistem Pakar, Penyakit Kulit Anjing, Bayes

KATA PENGANTAR

Assalamu’alaikum Wr. Wb.

Alhamdulillah, berkat rahmat dan hidayah Allah SWT yang telah memberikan segala karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik dan tepat waktu. Dan tak lupa salawat dan salam kepada Nabi besar Muhammad SAW yang telah berjasa besar dengan membukakan jalan dalam perkembangan ilmu pengetahuan seperti sekarang ini. Dalam penulisan penelitian ini, tidak terlepas dari dukungan bantuan berbagai pihak, pada kesempatan ini penulis mengucapkan rasa terima kasih kepada :

1. Bapak H. Herman Nawas, selaku ketua Yayasan Perguruan Tinggi Komputer “YPTK” Padang.
2. Bapak Prof. Dr. Sarjon Defit, S.Kom, M.Sc selaku Rektor Universitas Putra Indonesia “YPTK” Padang.
3. Bapak Dr. Julius Santony, S.Kom, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Putra Indonesia “YPTK” Padang.
4. Ibu Drh. Rita Mahyona, M.H sebagai pakar yang telah begitu banyak memberikan pengetahuan tentang penyakit kulit anjing dalam penyusunan skripsi ini.

Akhir kata, semoga penelitian ini dapat memberikan konstribusi terhadap masyarakat dan khususnya terhadap Universitas Putra Indonesia “YPTK” Padang.

 Padang, Januari 2019

Penulis

**DAFTAR ISI**

**HALAMAN JUDUL** **i**

**ABSTRAK** **ii**

**KATA PENGANTAR** **iii**

**DAFTAR ISI** **iv**

**BAB I PENDAHULUAN**.................................................................................. **1**

1. Latar Belakang Masalah 1
2. Perumusan Masalah 2
3. Hipotesa 2
4. Batasan Masalah 3
5. Tujuan Penelitian 3
6. Manfaat Penelitian 4
7. Gambaran Umum Pakar 4

**BAB II LANDASAN TEORI**............................................................................ **6**

* 1. Rekayasa Perangkat Lunak 6
		1. Definisi Perangkat Lunak 6
			1. Karakteristik Perangkat Lunak 7
			2. Kategori Utam Perangkat Lunak 7
		2. Rekayasa Perangkat Lunak 9
		3. Pendekatan Pengembangan Perangkat Lunak 11
		4. Proses Rekayasa Perangkat Lunak 14
		5. Model-Model Rekayasa Perangkat Lunak 14
			1. Model *Classic Life Cycle* 14
			2. Model *Prototyping* 15
			3. Model *Spiral* 17
			4. Teknik Generasi Ke Empat (4GT) 19
		6. Tantangan Rekayasa Perangkat Lunak 19
	2. *Unified Modelling Language* (UML) 20
		1. Pengertian UML 20
		2. Sejarah UML 20
		3. Diagram UML 20
			1. *Use Case* 21
			2. *Class Diagram*  25
			3. *Activity Diagram*  27
			4. *Squence Diagram* 28
			5. *State Chart Diagram* 31
			6. *Collaboration Diagram* 32
			7. *Deployment Diagram* 31
	3. Kecerasan Buatan *(Artificial Intelligence)*  34
		1. Defenisi Kecerdasan Buatan 35
		2. Kecerdasan Alami dan Kecerdasan Buatan 38
		3. Ruang Lingkup Kcerdasan Buatan 40
	4. Sistem Pakar 41
		1. Defenisi Sistem Pakar 42
		2. Kelebihan dan Karakteristik Sistem Pakar 44
		3. Kekurangan Sistem Pakar 45
		4. Manfaat Sistem Pakar 45
		5. Ciri-Ciri Sistem Pakar 45
		6. Konsep Dasar Sistem Pakar 46
			1. Kepakaran 46
			2. Pakar 47
			3. *Inferensi* 47
			4. Aturan-Aturan *(Rule)* 48
	5. Metode *Bayes* 48
	6. Bahasa Pemograman PHP 50

 2.6.1 Pengertian PHP 50

 2.6.2 Sejarah PHP 51

 2.6.3 Kelebihan PHP 52

2.7 Database 53

 2.7.1 Definisi Database 50

 2.7.2 DBMS *(Database Management System)* 53

 2.7.3 Database MySQL 54

2.8 Anjing 56

 2.8.1 Mencegah Serangan Penyakit Kulit Pada Anjing 56

 2.8.2 Jenis-Jenis Penyakit Kulit Pada Anjing 57

**BAB III METODOLOGI PENELITIAN**......................................................... **64**

* 1. Kerangka Penelitian 64
	2. Tahapan Penelitian 64
		1. Penelitian Pendahuluan 65
		2. Pengumpulan Data 65
		3. Analisa 69
		4. Perancangan 70
		5. Implementasi 72
		6. Pengujian 72

**BAB IV ANALISA DAN PERANCANGAN** **75**

* 1. Analisa 75
	2. Perancangan 99
		1. Perancangan Model 99
		2. Perancangan Interface 114
	3. Perancangan Sistem 77
		1. Use Case Diagram 78
		2. Class Diagram 82
		3. Sequence Diagram 91
		4. Collaboration Diagram 94
		5. Statechart Diagram 98
		6. Activity Diagram 100
		7. Deployment Diagram 103
		8. Desain Output 104

**BAB V IMPLEMENTASI DAN PENGUJIAN** **122**

* 1. Implementasi 122
		1. Implementasi Sistem 122
		2. Implementasi Program 122
	2. Pengujian 126
		1. Pengujian Online 126
		2. Pengujian LAN 133
		3. Pengujian Aplikasi 142
		4. Pengujian *Interface* 144

**BAB VI PENUTUP**  **153**

* 1. Kesimpulan 153
	2. Keterbatasan Sistem 153
	3. Saran 154

**DAFTAR PUSTAKA** 155

**BAB I**

**PENDAHULUAN**

**1.1 Latar Belakang Masalah**

Berdasarkan penelitian yang dilakukan sebelumnya oleh Arnaz Malikul Hakim, Jusak, dan Erwin Sutomo (2017) dengan judul “*Sistem Pakar Identifikasi Penyakit Kulit Anjing*” dengan metode *Certainty Factor* yang menjelaskan masalah tentang penyebab penyakit kulit pada anjing, jenis penyakit kulit pada anjing, gejala dan cara mengatasinya. Penyakit yang menyerang pada kulit anjing sangat bervariasi sehingga sangat dibutuhkan seorang pakar yang mampu memberikan solusinya. Akan tetapi waktu dan biaya yang menjadi alasan bagi masyarakat untuk tidak melakukan konsultasi kepada pakarnya sehingga sering terjadi kesalahan dalam memberikan solusi penanganannya. Untuk membantu mengatasi permasalahan tersebut, maka dirancang suatu sistem pakar diagnosa penyakit kulit pada anjing yang dapat meniru cara kerja seorang pakar.

Masyarakat yang memelihara anjing pada umunya banyak yang anjing peliharaannya terkena penyakit kulit karena masyarakat kurang memiliki pengetahuan bagaimana gejala awal penyakit kulit pada anjing tersebut, dan masyarakat juga kurang mengetahui solusi untuk mengatasi anjing peliharaan yang menderita penyakit kulit.

Diharapkan penelitian ini dapat membantu permasalahan yang ada dengan pembuatan sistem pakar yang dapat mendiagnosa penyakit kulit pada anjing. Sistem pakar sebagai program yang difungsikan utuk meniru pakar harus bisa melakukan hal-hal yang dapat dikerjakan oleh seorang pakar. Sistem pakar yang akan dirancang menggunakan metode *Bayes* berbasis nilai kepastian atau tingkat keyakinan.

Berdasarkan uraian diatas, maka penulis tertarik untuk mengangkat permasalahan tersebut menjadi judul penelitian yaitu: **“PERANCANGAN APLIKASI SISTEM PAKAR BERBASIS WEB UNTUK MENDIAGNOSA PENYAKIT KULIT PADA ANJING MENGGUNAKAN METODE BAYES”.**

1. **Perumusan Masalah**

Berkaitan dengan latar belakang dari pemilihan judul di atas, maka diperoleh perumusan masalah yang akan dihadapi oleh penulis dalam penelitian ini, adalah sebagai berikut :

1. Bagaimana aplikasi sistem pakar yang dibuat nantinya bermanfaat dan dapat membantu masyarakat dalam mengenali penyakit kulit pada anjing ?
2. Bagaimana aplikasi sistem pakar yang dibuat nantinya dapat membantu memberikan informasi gejala awal penyakit kulit pada anjing ?
3. Bagaimana program aplikasi sistem pakar dengan metode *Bayes* dapat memberikan solusi untuk mengatasi penyakit kulit pada anjing ?

**1.3 Hipotesa**

Berdasarkan perumusan diatas, maka penulis juga dapat menarik hipotesa-hipotesa yang dapat memberikan kesimpulan sementara sebagai acuan penulis dalam penyelesaian penelitian ini, antara lain :

1. Sistem pakar yang dirancang diharapkan dapat bermanfaat dan dapat membantu masyarakat dalam mengenali penyakit kulit pada anjing.
2. Aplikasi sistem pakar diharapkan dapat memberikan informasi gejala awal penyakit kulit pada anjing.
3. Diharapkan aplikasi sistem pakar melakukan penelusuran serta mengeluarkan hasil diagnosa dan memberikan solusi terbaik.
4. **Batasan Masalah**

Agar pembahasan penelitian ini tidak menyimpang dari apa yang telah dirumuskan, maka diperlukan batasan-batasan dalam membantu sistem pakar untuk penelitian ini. Adapun batasan masalahnya adalah mendiagnosa penyakit yang menyerang pada kulit anjing yang mencakup; alergi, *ringworm* (jamur)*, scabies* (kudis), *demodec*, dan *flea and tick* (kutu) dengan metode *Bayes* berbasis *Web* dengan bahasa pemograman *PHP* dan database *MySQL* .

**1.5 Tujuan Penelitian**

Agar terwujudnya dari semua yang telah direncanakan oleh penulis, adapun tujuan dari penelitian ini, yaitu :

1. Membantu masyarakat melakukan identifikasi dalam mengenali penyakit kulit pada anjing menggunakan pengolahan komputer secara online sehingga penangan lebih lanjut terhadap penyakit kulit pada anjing dapat dilakukan dengan cepat dan tepat.
2. Membantu memberikan informasi gejala awal yang timbul pada penyakit kulit pada anjing.
3. Sebagai media informasi untuk membantu masyarakat mengenai akibat yang ditimbulkan oleh penyakit kulit pada anjing serta solusi penangananya.

**1.6 Manfaat Penelitian**

Adapun manfaat penelitian yang dapat dilakukan untuk melengkapi laporan penelitian antara lain :

1. Masyarakat lebih mengetahui secara umum informasi mengenai penyakit kulit pada anjing.
2. Masyarakat dapat mengidentifikasi gejala awal penyakit kulit pada anjing lebih cepat.
3. Masyarakat mengetahui solusi apa yang harus dilakukan saat anjing peliharaan mereka menderita penyakit kulit.

Penelitian ini bermanfaat sebagai suatu masukan dalam hal memberikan penyuluhan serta edukasi pada masyarakat terhadap pencegahan penyakit kulit pada anjing dan solusi mengatasinya.

* 1. **Gambaran Umum Pakar**

Untuk melakukan penelitian ini sangat di harapkan dukungan dan informasi yang pasti dari seorang pakar yang dipercaya. Dalam penelitian ini yang di jadikan pakar yaitu Drh.Rita Mahyona, M.H. Beliau lahir di Padang, 9 Agustus 1983. Beliau memiliki riwayat pendidikan sekolah dasar sampai menengah atas di SDN 30 cengkeh, SMPN 11 Kota Padang, SMU PGRI 2 Kota Padang. Beliau lulusan sarjana dari Profesi Dokter Hewan Syiah Kuala dan Program Pasca Sarjana Ilmu Hukum di Universitas Eka Sakti. Beliau merupakan Pegawai Negeri Sipil pada Balai Karantina Pertanian Kelas 1 Kota Padang dan menjabat sebagai Medik Veteriner Muda. Selain itu beliau juga membuka praktek dokter hewan di R4 Petshop Dan Klinik yang beralamatkan di jalan Rimba Datar No. 23 Bandar Buat Kota Padang.

Pengalaman pakar dalam mengatasi penyakit kulit pada anjing sudah lumayan banyak karena penyakit kulit adalah penyakit yang rentan di derita oleh anjing. Salah satu contoh penyakit kulit yang pernah di tangani oleh pakar adalah jenis penyakit kulit *scabies* atau yang sering disebut juga kudis. Dalam kasus itu pakar berhasil menangani penyakit tersebut setelah melakukan serangkaian perawatan dan pengobatan.

BAB II

LANDASAN TEORI

* 1. **Rekayasa Perangkat Lunak**
		1. **Defenisi Perangkat Lunak**

Perangkat lunak memiliki beberapa definisi yaitu (Pressman, Roger S, hal: 5, 2012) :

1. Perangkat lunak adalah instruksi-instruksi atau program komputer yang ketika dijalankan menyediakan fitur-fitur, fungsi-fungsi dan kinerja-kinerja yang dikehendaki.
2. Perangkat lunak adalah struktur data yang memungkinkan program-program dalam memanipulasi informasi.
3. Perangkat lunak adalah informasi deskriptif pada salinan tercetak dan bentuk-bentuk maya yang menggambarkan pengoperasian dan penggunaan program-program.

Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*User Manual*). Sebuah program komputer tanpa terasosiasi dengan dokumentasi maka belum dapat disebut perangkat lunak (*Software*) (S, Rosa A dan M.Shalahudin, hal : 2, 2014).

Dari pengertian diatas dapat disimpulkan bahwa perangkat lunak atau *software* merupakan sebuah program komputer, struktur data, dan dokumentasi yang saling berkaitan dan menyediakan metode logika, prosedur atau suatu kontrol yang diminta atau diinginkan.

* + - 1. **Karakteristik Perangkat Lunak**

Perangkat lunak memiliki karakteristik yang berbeda dengan perangkat keras, yaitu (Pressman, Roger S, hal : 6, 2012) :

1. Perangkat lunak dikembangkan atau direkayasa, bukan diproduksi dalam konteks manufaktur.
2. Perangkat lunak tidak mengalami kelelahan.

Pada dasarnya perangkat lunak tidak terpengaruh oleh kondisi lingkungan yang menyebabkan perangkat keras kelelahan. Namun, kinerjanya dapat memburuk.

1. Meskipun industri terus beralih ke konstruksi berbasis komponen, sebagian besar perangkat lunak masih tetap dibuat berdasarkan spesifikasi yang diminta pengguna.

Sebuah komponen perangkat lunak harus dirancang dan diterapkan sedemikian rupa sehingga dapat digunakan kembali dalam program yang berbeda.

Perangkat lunak memiliki beberapa karakter (S, Rosa A dan M.Shalahuddin, hal : 2, 2014) :

1. Perangkat lunak dibangun dengan rekayasa (*software engineering*) bukan diproduksi secara manufaktur atau pabrikan.
2. Perangkat lunak tidak pernah usang (*“wear out”*) karena kecacatan dalam perangkat lunak dapat diperbaiki.
3. Barang produksi pabrikan biasanya komponen barunya akan terus diproduksi, sedangkan perangkat lunak biasanya terus diperbaiki seiring bertambahnya kebutuhan.
	* + 1. **Kategori Utama Perangkat Lunak**

Terdapat 7 kategori utama pada perangkat lunak komputer, yaitu (Pressman, Roger S, hal: 8, 2012) :

1. Perangkat lunak sistem

Merupakan sekumpulan program yang ditulis untuk melayani program-program lain, memproses data yang beragam serta jumlahnya sangat tidak terbatas.

1. Perangkat lunak aplikasi

Merupakan program-program yang menjawab kebutuhan bisnis yang terinci. Aplikasi-aplikasi dalam bidang ini melakukan pemrosesan data bisni atau data teknis yang mendukung berjalannya operasi-operasi bisnis.

1. Perangkat lunak rekayasa/ilmiah

Merupakan perangkat lunak yang telah lama memiliki algoritma yang penuh dengan kalkulasi data numerik, yang dikembangkan untuk kebutuhan ilmiah atau keilmuan.

1. Perangkat lunak yang tertanam

Merupakan pernagkat lunak yang terdapat didalam suatu produk atau sistem dan digunakan untuk menjalankan dan mengendalikan fitur-fitur bagi pengguna akhir dan bagi sistem itu sendiri. Perangkat lunak ini dapat menjalankan fungsi-fungsi terbatas dan seringkali hanya dipahami oleh kalangan-kalangan tertentu.

1. Perangkat lunak lini produk

Merupakan perangkat lunak yang dirancang untuk menyediakan kemampuan khusus untuk digunakan oleh pengguna yang berbeda-beda. Perangkat lunak ini dapat berkonsentrasi pada pasar tertentu dan sifatnya terbatas atau merujuk pada pasar pengguna.

1. Aplikasi web

Disebut WebApps, merupakan kategori perangkat lunak yang berpusat pada jaringan komputer yang menyajikan sederatan luas aplikasi-aplikasi. Seiring dengan munculnya *Web* 2.0, aplikasi *web* berevolusi menjadi komputasi cangguh yang terintegrasi dengan system basis data.

1. Perangkat lunak kecerdasan buatan

Merupakan perangkat lunak yang menggunakan algoritma non-numerik untuk memecahkan permasalahan-permasalahan rumit yang tidak bisa diselesaikan dengan komputasi atau analisis permasalahan secara langsung.

* + 1. **Definisi Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak adalah aplikasi dari suatu pendekatan yang sistematik, disiplin, dan dapat diukur pada pengembangan, operasi, dan perawatan perangkat lunak; yaitu penerapan rekayasa pada perangkat lunak (Pressman, Roger S, hal: 15, 2012).

Rekayasa perangkat lunak (*software enginering)* merupakan pembangunan dengan mengunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien menggunakan mesin (S, Rosa A dan M.Shalahuddin, hal : 4, 2014).

Perangkat lunak banyak di buat dan pada akhirnya sering tidak digunakan karena tidak memenuhi kebutuhan pelanggan atau karena masalah non-teknis seperti keengganan pemakai perangkat lunak (*User*) untuk merubah cara kerja dari manual ke otomatis atau ketidak mampuan user menggunakan komputer.

Terdapat empat tipe perubahan pada fase pendukung yaitu (S, Rosa A dan M.Shalahuddin, hal : 6, 2014) :

1. Koreksi (*correction*)

Walupun dengan jaminan kualitas yang terbaik, akan selalu ada kecacatan atau keinginan pelanggan (*customer*) yang tidak tertangani oleh perangkat lunak. Pemeliharaan dengan melakukan perbaikan terhadap kecacatan perangkat lunak.

1. Adaptasi (*adaptation*)

Pada saat tertentu lingkungan asli (seperti CPU, sistem operasi, aturan bisnis, karakteristik produk luar) dimana perangkat lunak dikembanngkan atau mengalami perubahan. Pemeliharaan adaptasi merupakan tahap untuk memodifikasi perangkat lunak guna mengakomodasi perubahan lingkungan luar dimana perangkat lunak dijalankan.

1. Perbaikan (*enhancement*)

Sejalan dengan digunakannya perangkat lunak, maka pelanggan (*customer*) atau pemakainya (*user*) akan mengenali fungsi tambahan yang dapat mendatangkan manfaat.pemeliharaan perfektif atau penyempurnaan melakukan ekstensi atau penambahan pada kebutuhan fungsional sebelumnya.

1. Pencegahan (*prevention*)

Keadaan perangkat lunak komputer sangat dimungkinkan untuk perubahan. oleh karna itu, pemeliharaan pencegahan (*preventif*) atau sering di sebut juga dengan rekayasa ulang sistem (*software engineering*) harus dikondisikan untuk mampu melayani kebutuhan pemakainya (*user*).

Sesungguhnya rekayasa perangkat lunak merupakan teknologi yang berlapis, merujuk pada gambar 2.1, segala pendekatan rekayasa (termasuk rekayasa perangkat lunak) harus berstandar pada komitmen organisasional pada peningkatan kualitas.

**Sumber: Pressman, Roger S, 2012**

**Gambar 2.1 Lapisan Rekayasa Perangkat Lunak**

* + 1. **Pendekatan Pengembangan Perangkat Lunak**

Ada beberapa pendekatan utama pada industri komputer untuk pengembangan perangkat lunak. Beberapa pendekatan yang ada merupakan pendekatan dasar dan ada juga yang muncul dari lingkungan penelitian. Batasan seperti spesifikasi yang dibutuhkan dan standar sangat perlu untuk menentukan pendekatan yang tepat untuk pengembangan perangkat lunak nantinya. Pendekatan utama pengembangan perangkat lunak adalah sebagai berikut (Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013 ) :

1. *Structured Approach*

*Structured approach* telah diajukan untuk rekayasa perangkat lunak *lifecycle*. Pada tahap analisis, dikenal hubungan hirarki dan fungsi antara objek dan aktivitas. Pada setiap tingkat dekomposisi, komponen sistem dilukiskan sebagai komponen induk, input, output, kontrol, aktivitas, dan mekanisme yang mendukung komponen. Berikut ini adalah beberapa aktivitas yang terdapat pada *structured approach* :

1. *Structured Analysis*

*Structured analysis* didasarkan pada penggunaan dataflow diagram yang diperkenalkan oleh Tom DeMarco. Data flow diagram membuat model proses dengan aliran data dan transformasi, membentuk masukan data jaringan sebagai input, melanjutkan fungsi transformasi dengan maupun tanpa data lain, kemudian menjadi output.

1. *Structured Design*

*Structured design* memetakan aliran data dari pusat masalah sampai ke struktur perangkat lunak. Langkah *structured design* meliputi pengenalan aliran data melalui representasi citra, identifikasi beragam elemen pengubah, mengumpulkan elemen-elemen dalam struktur program hirarki, dan mengoptimalkan elemen.

1. *Structured Programming*

*Structured programming* merupakan pendekatan yang efisien untuk mengembangkan kode program komputer. Pendekatan ini didasarkan pada pembuktian struktur teorema secara matematika yang menyatakan program yang memiliki satu masukan dan satu keluaran adalah sebanding dengan program yang hanya memiliki tiga struktur logika berikut :

1. Sekuensial dari dua atau lebih operasi
2. Cabang kondisional ke satu atau dua operasi dan kembali (*If* A *Then* B *Else* C)
3. Pengulangan operasi saat kondisi true (*Do-While* p)
4. *Object-Oriented Approach*

Pada *object-oriented approach*, model entitas dibentuk sebagai komponen self-contained. Entitas program merujuk pada objek yang lebih dari satu kelas. *Object-oriented design* ditampilkan sebagai metode untuk pemodelan masalah dengan pandangan yang seimbang antara objek dan operasi yang dilakukan seperti yang disarankan oleh Booch berikut ini :

1. Tentukan strategi informal untuk solusi masalah.
2. Identifikasiobjekyang digunakandalamstrategi informal.
3. Identifikasioperasipada objek yang digunakan dalam strategi informal.
4. Tentukanarsitektursistem perangkatlunakdan antarmuka operasi.
5. Ulangiprosesdiatasjika diperlukan.
6. *Entity Relationship Approach*

*Entity relationship approach* menggunakan modelentity relationship untuk mengelompokkan informasi dari dunia nyata. Pendekatan ini mengenali database yang diperlukan pada tingkat logika dan fisik. Informasi ini dibuat dengan menentukan entitas pusat, interrelasi entitas, dan atribut yang dimiliki entitas. Konsep ini harus dipetakan dalam bentuk rencana untuk dapat di implementasikan pada sistem manajemen database.

1. *Event-Oriented Approach*

*Event-oriented approach* dikenal sebagai konsep respon stimulus, dimana kejadian adalah stimulus bagi sistem, dan respon dibentuk dari aksi yang diambil oleh sistem dan output resultan. Pendekatan ini membangun sistem yang berdasarkan jenis kejadian yang dialami oleh sistem.

1. *Stepwise Refinement Approach*

N. Wirth mengajukan konsep *stepwise refinement*, strategi disaintop-down, yang prosesnya dimulai dari abstraksi tingkat tinggi dan gabungan detil melalui urutan terperinci. Dekomposisi program metode ini paralel dengan proses partisi yang sering digunakan dalam requirements analysis.

* + 1. **Proses Rekayasa Perangkat lunak**

Selama pengembangan suatu perangkat lunak dilakukan proses rekayasa perangkat lunak, yang secara garis besar adalah sebagai berikut :

Mulai

**Sumber: S, Rosa A dan M.Shalahudin, Rekayasa Perangkat Lunak, 2014**

**Gambar 2.2 Tahapan Umum Rekayasa Rerangkat Lunak**

* + 1. **Model-Model Rekayasa Perangkat Lunak**

Rekayasa perangkat lunak tersusun dari sekumpulan langkah-langkah yang menggabungkan metode, alat bantu, dan prosedur. Empat jenis pemodelan yang umum digunakan, yaitu (Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013 ) :

* + - 1. **Model *Classic Life Cycle***

Pendekatan ini sering disebut dengan *Waterfall Model*, yang membutuhkan suatu pendekatan yang sistematis dan berurutan.

Rekayasa Sistem

Analisis

Perancangan

Pembuatan Program

Pengujian

Pemeliharaan

**Sumber: Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013**

**Gambar 2.3 Tahapan Classic Life Cycle**

Berikut penjelasan dari tahapan-tahapan yang ada pada *Classic Life Cycle* :

1. Rekayasa Sistem

Hubungan antara perangkat lunak dengan perangkat keras, manusia dan databasenya.

1. Analisis

Memahami kebutuhan perangkat lunak, fungsi-fungsi, unjuk kerja dan antarmuka yang diperlukan.

1. Perancangan

Tahapan yang berfokus pada empat atribut dari program, yaitu : struktur data, arsitektur perangkat lunak, detil suatu prosedur, dan karakteristik antarmuka.

1. Pembuatan Program

Aktivitas yang mengubah hasil rancangan menjadi bentuk yang dapat dimengerti komputer, biasanya dalam bentuk program.

1. Pengujian

Setelah pengkodean selesai, maka akan dilakukan pengujian program. Pengujian dilakukan untuk menemukan kesalahan serta memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

1. Pemeliharaan

Suatu perangkat lunak akan mengalami perubahan sejalan dengan perubahan waktu. Pembaharuan, penambahan fungsi, dan peningkatkan kinerja perangkat lunak akan terus dilakukan untuk meningkatkan efisiensi dan efektivitas bagi pengguna.

* + - 1. **Model *Prototyping***

Prototyping adalah suatu proses yang memungkinkan penciptaan sebuah model perangkat lunak yang hendak dibangun agar dapat diketahui terlebih dahulu efisiensi suatu algoritma, adaptabilitas sistem operasi atau interaksi manusia dan komputer yang sesuai (Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013 ).

Urutan langkah-langkah umum yang dilakukan pada prototyping ialah:

1. Pengumpulan kebutuhan-kebutuhan
2. Perancangan secara cepat
3. Pembuatan prototype
4. Evaluasiprototype oleh pengguna
5. Penyempurnaan prototype
6. Pembuatan prototype

Model prototipe (*prototyping model*) dimulai dari mengumpulkan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Lalu dibuatlah program prototipe agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program prototipe biasanya merupakan program yang belum jadi. Program ini biasanya menyediakan tampilan dengan simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi. Program prototipe ini dievaluasi oleh pelanggan atau *user* sampai ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user*. Untuk lebih jelasnya model prototipe dapat dilihat pada Gambar 2.3.



**Sumber: S, Rosa A dan M. Shalahuddin ,2014**

**Gambar 2.4 Ilustrasi Model *Prototipe***

* + - 1. **Model *Spiral***

Pemodelan ini dikembangkan dengan menggabungkan kelebihan-kelebihan dari kedua pemodelan sebelumnya, serta menambahkan analisis resiko. Dalam pemodelan ini terdapat empat aktivitas penting, yaitu: perencanaan, analisis resiko, rekayasa produk, dan evaluasi pengguna (Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013 ).

Model spiral dibagi menjadi beberapa kerangka aktifitas atau disebut juga wilayah kerja (*task region*). Banyaknya wilayah kerja biasanya diantara tiga sampai enam eilayah sebagai berikut :

1. Komunikasi dengan pelanggan (*customer communication*).

Aktifitas ini diperlukan untuk membengun komunikasi yang efektif antara pengembang (*developer*) dan pelanggan (*customer*)

1. Perencanaan (*planning*)

Aktifitas ini diperlukan untuk mendefenisikan sumber daya, waktu, dan informasi yang terkait dengan proyek.

1. Analisis risiko (*risk analysis*)

Aktifitas ini diperlukan untuk memperkirakan risiko dari segi teknis maupun manajemen

1. Rekayasa (*engineering*)

Aktifitas ini diperlukan untuk membangun satu atau lebih representasi dari aplikasi perangkat lunak (dapat juga berupa prototipe)

1. Konstruksi dan peluncuran (*contruction and release*)

Aktifias ini dibutuhkan untuk mengonstruksi, menguji, melakukan instalasi, dan menyediakan dukungan terhadap *user.*

1. Evaluasi pelanggan (*customer evaluation*)

Aktifitas ini dibutuhkan untuk mendapatkan umpan balik berdasarkan evaluasi representasi perangkat lunak yang dihasilkan dari proses rekayasa dan diimplementasikan pada tahap instansi. (S, Rosa A dan M.Shalahuddin, hal : 39, 2014)

Berikut adalah gambar model spiral :



**Sumber: S, Rosa A dan M.Shalahudin, 2014**

**Gambar 2.5 Gambar Model *Spiral***

* + - 1. **Teknik Generasi Keempat (4GT)**

Pemodelan ini menggunakan alat bantu perangkat lunak, yakni bahasa pemrograman 4GT, yang memungkinkan pengembang sistem menspesifikasi karakteristik perangkat lunak dimana alat bantu ini akan menghasilkan kode sumber berdasarkan spesifikasi secara otomatis. Empat tahapan dalam pemodelan ini adalah pengumpulan kebutuhan kebutuhan, penyusunan strategi perancangan, implementasi dengan 4GT, dan uji coba (Hermansyah Nur Ahmad dan Nirwan Jember Safa’at, 2013 ).

* + 1. **Tantangan Rekayasa Perangkat Lunak**

Tantangan yang dihadapi dari proses rekayasa perangkat lunak adalah sebagai berikut (S, Rosa A dan M.Shalahudin, hal : 7, 2014) :

1. Tantangan warisan dimana perangkat lunak dikembangkan selama bertahun-tahun oleh orang-orang yang berbeda, hal ini dapat menyebabkan ketidak pahaman atau perubahan tujuan pembuatan perangkat lunak.
2. Tantangan heterogenitas dimana perangkat lunak harus dapat beradaptasi dengan teknologi yang terus berkembang dengan semakin luasnya lingkungan distribusi perangkat lunak.
3. Tantangan pengiriman bahwa perangkat luank dengan sekala besar dan kompleks sekalipun dapat sampai ketangan pelanggan (*customer*) atau *user* dengan cepat dan kualitas tetap terjaga.
	1. ***Unified Modeling Language* (UML)**
		1. **Pengertian UML**

UML adalah bahasa standar untuk penulisan cetak biru perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menentukan, mengonstruksi, dan mendokumentasikan artifak-artifak suatu suatu sistem *software-intensive*. Dengan kata lain, sama seperti arsitek membuat cetak biru untuk digunakan oleh perusahaan konstruksi, arsitek perangkat lunak membuat UML untuk membantu pengembangan perangkat lunak (Pressman, Roger : 2010).

* + 1. **Sejarah UML**

Pada pertengahan 1990-an Grady Booch, Jim Rumbaugh, dan Ivar Jacobson mengembangkan UML dengan banyak umpan balik dari komunitas pengembangan perangkat lunak. UML menggabungkan sejumlah notasi pemodelan yang saling bersaing, yang digunakan oleh industri perangkat lunak saat itu. Pada tahun 1997, UML 1.0 diserahkan kepada ObjectManagement Group, suatu konsorium nirlaba yang aktif dalam pemeliharaan spesifikasi-spesifikasi untuk digunakan oleh industri komputer. UML 1.0 direvisi ke UML 1.1 dan diadopsi pada tahun itu juga. Standar sekarang adalah UML 2.0 dengan 13 diagram berbeda-beda yang digunakan pada pemodelan perangkat lunak (Pressman, Roger S, : 2010).

* + 1. **Diagram UML**

Pada UML terdapat 13 macam diagram, diantaranya (S, Rosa A dan M.Shalahudin : 2014) :

* + - 1. ***Use Case***

*Use Case* atau diagram *use Case* merupakan pemodelan untuk kelakukan (behavior) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sitem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Adadua hal utama pada *use case* yaitu pendefinisian yang di sebut aktor dan *use case*.

1. Aktor merupakan orang proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan di buat di luar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang di sediakan sistem sebagai unit-unit yang saling bertukar pesan antarunit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case:*

#  Tabel 2.1 Tabel Simbol *Use Case Diagram*

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| *Use case* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling betukar pesan antara unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama *use case* |
| \\Aktor / *actor*  nama aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal fase nama aktor. |
| Asosiasi / *association* | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor |
| Ekstensi / *extend*<<extand>> | Relasi *use case* tambahan kesebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan, misal: <<extend>> <<extend>>Arah panah mengarah pada *use case* yang ditambahkan; biasanya *use case* yang menjadi *extend-*nyamerupakan jenis yang sama dengan *use case* yang menjadi induknya. |
| Generalisasi / *generalization* | Hubungan generalisasi dan spesialisasi (umun– khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:arah panah mengarah pada *use case* yang menjadi generalisasinya (umum) |
| Menggunakan / *include / uses* *<<*include*>>* <<uses>> | Relasi *use case* tambahan kesebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat di jalankan *use case* ini.Ada dua sudut pandang yang cukup besar mengenai include *use case:*1. *Include* berarti *use case*yang di tambahkan akan selalu di panggil oleh *use case* tambahan dijalankan, misal pada kasus berikut:

<<include>>1. *Include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *usse case* tambahan dijalankan, misal pada kasus berikut:

 <<include>>Kedua interpretasi diatas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***Class Diagram***

Diagram kelas atau Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun system. Kelas memiliki atribut (variabel-variabel yang dimiliki suatu kelas) dan operasi atau metode (fungsi yang dimiliki suatu kelas).

Kelas yang baik pada diagram kelas memiliki jenis kelas-kelas berikut:

* 1. Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

* 1. Kelas yang menangani tampilan sistem(*view*)

Kelas yang mendefinisikan dan mengatur tampilan kepemakai.

* 1. Kelas yang diambil dari pendefinisian *use case* (*controlkler*)

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

* 1. Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data (S, Rosa A dan M.Shalahudin, hal : 142, 2014).

Berikut adalah simbol-simbol yang ada pada diagram kelas:

**Tabel 2.2 Tabel Simbol *Class Diagram***

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Kelas

|  |
| --- |
| nama\_kelas |
| +atribut |
| +operasi() |

  | Kelas pada struktur sistem |
| Antarmuka / *interface* **nama\_interface** | Sama dengan konsep interface dalam pemograman berorientasi objek |
| Asosiasi / association | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity* |
| Asosiasi berarah / directed association | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity* |
| Generalisasi | Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus) |
| kebergantungan / dependency | Relasi antarkelas dengan makna kebergantungan antarkelas |
| Agregasi / aggregation | Relasi antarkelas dengan makna semua-bagian (whole-part) |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***Activity Diagram***

Diagram activity menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada perangkat lunak. Yang perlu diperhatikan disini adalahbahwa diagram activity menggambarkan aktivitas sistem bukan apa yang dilakukan aktor.

Diagram aktifitas juga digunakan untuk mendefinisikan hal berikut (S, Rosa A dan M.Shalahudin, hal, 2014) :

1. Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan peroses bisnis sistem yang di definisikan.
2. Urutan atuu pengelompokan tampilan dari sistem / *user intreface*dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian diamana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut adalah simbol- simbol yang ada pada activity diagram:

**Tabel 2.3 Tabel Simbol *Activity Diagram***

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Status awal | status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| Aktivitasaktivitas | aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kerja keras |
| Percabangan / decision | asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| Penggabungan / join  | asosiasi penggabungan dimana lebih dari satu akvitas digabungkan menjadi satu |
| Status akhir | status akhir yang dilakukan sistem, sebuah diagram aktivitas memiki sebuah status akhir |
| Swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap yang terjadi |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***Sequence Diagram***

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objekdan message yang dikirimkan dan dirima antarobjek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skanario yang pada *use case* (S, Rosa A dan M.Shalahudin, 2014).

Berikut adalah simbol-simbol yang ada pada diagram sekuen:

 **Tabel 2.4 Tabel Simbol *Sequence Diagram***

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Aktor nama aktoratau nama aktornama aktorntanpa waktu aktif | Orang,proses,atau sistem lainyang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang,tapi aktor belum tentu merupakan orang;biasanya dinyatakan menggunakan kata benda diawal frase nama aktor |
| Garis hidup / lifeline | Menyatakan kehidupansuatu objek |
| Objeknama objek:nama kelas | Menyatakan objek yang berinteraksi pesan |
| Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya misalnaya1: login() 2: cekStatusLogin() 3: open()Maka cek status login() dan open() dilakukan di dalam metode login () Aktor tidak memiliki waktu aktif |
| Pesan tipe *create* <<create>> | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat |
| Pesan tipe *call*1 : nama\_metode() | Menyataka suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri, 1 : nama\_metode()Arah panah mengarah pada objek yang memiliki operasi / metode, karna ini memanggil operasi / metode maka operasi / metode yang di panggil harus harus ada pada diagram kelas sesuai dengan kelas pbjek yang berorioentasi. |
| Pesan tipe send1: masukan | Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi objek lainnya, arah panah mengarah pada objek yang di kirim |
| Pesan tipe return1: keluaran | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian  |
| Pesan tipe destroy<<destroy>>  | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah menagarah pada objek yang diakhiri. |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***State Chart Diagram***

*Statechart* atau *state machine diagram* disebut sebuah diagram mesin status atau sering juga disebut diagram status yang digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek. Jika diagram sekuen digunakan untuk interaksi antar objek maka diagram status digunakan untuk interaksi didalam sebuah objek. Perubahan tersebut digambarkan dalam sebuah graf berarah. *Satate machine diagram* merupakan pengembangan dari lingkungan  *finite state automata* dengan penambahan beberapa fitur dan konsep baru. (S, Rosa A dan M. Shalahuddin, hal : 163, 2014).

*State machine diagram* cocok digunakan untuk menggambarkan alur interaksi pengguna dengan sistem. Berikut adalah contoh gambar diagram mesin status :

Status awal

Status 1

Status 3

Status 2

Status akhir

event triger

aksi

event triger

aksi

event triger

 aksi transaksi kediri sendiri

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

**Gambar 2.6 Contoh *State Machine Diagram***

Berikut ini komponen –komponen dasar yang ada dalam *state machine diagram* :

#  Table 2.5 Tabel Simbol *State Machine Diagram*

|  |  |
| --- | --- |
| **Simbol**  | **Keterangan** |
| start / status awal (*initial state*) | *Star* atau *initial state* adalah *state* atau keadaan awal pada saat sistem mulai hidup |
| End / status akhir (*final state*) | *End* atau *final state* keadaan akhir dari daur hidupsuatu sistem |
| Event  Event | *Event* kegiatan yang menyebabkan perubahan setatus mesin |
| State state | Sistem pada waktu tertentu. *State* dapat berubah jika ada *event* tertentu yang memicu perubahan tesebut |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***Collaboration Diagram***

*Collaboration diagram* adalah bentuk kompleks dari diagram komponen yang merupakan diagram yang menggambarkan interaksi antar objek / bagian dalam bentuk urutan pengiriman pesan.

Berikut simbol-simbol yang terdapat pada diagram kolaborasi :

**Table 2.6 Tabel Simbol *Collaboration Diagram***

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Objeknama objek:nama kelas | Objek yang melakukan interaksi pesan |
| Link | Relasi antar-objek yang menghubungkan objek satu dengan lainnya atau dengan dirinya sendirinama objek:nama kelas |
| Arah pesan / *stimulus* | Arah pesan yang terjadi, jika pada suatu link ada dua arah pesan yang berbeda maka arah juga digambarkan dua arah pada sisi *link* |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* + - 1. ***Deployment Diagram***

Diagram deployment menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram deployment juga dapat digunakan untuk memodelkan hal-hal berikut :

1. Sistem tambahan (embedded system) yang menggambarkan rancangan device, node dan hardware.
2. Sistem client/ server

Berikut adalah simbol-simbol pada deployment diagram :

 **Table 2.7 Tabel Simbol *Deployment Diagram***

|  |  |
| --- | --- |
| **Simbol** | **Deskripsi** |
| Package package | Package merupakan sebuah bungkusan dari satu atau lebih node |
| Node **nama\_node** | Biasanya mengacu pada perangkat keras (hardware), perangkat lunak yang tidak dibuat sendiri, jika di dalam node disertakan komponen mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen |
| Kebergantungan / dependency | Kebergantungan antar node, arah panah mengarah pada node yang dipakai |
| Link | Relasi antar node |

**Sumber: S, Rosa A dan M.Shalahudin, 2014**

* 1. **Kecerdasan Buatan *(Artificial Inteligence)***

Kecerdasan Buatan adalah salah satu cabang Ilmu pengetahuan berhubungan dengan pemanfaatan mesin untuk memecahkan persoalan yang rumit dengan cara yang lebih manusiawi. Hal Ini biasanya dilakukan dengan mengikuti/mencontoh karakteristik dan analogi berpikir dari kecerdasan/Inteligensia manusia, dan menerapkannya sebagai algoritma yang dikenal oleh komputer. Dengan suatu pendekatan yang kurang lebih fleksibel dan efisien dapat diambil tergantung dari keperluan, yang mempengaruhi bagaimana wujud dari perilaku kecerdasan buatan. AI biasanya dihubungkan dengan Ilmu Komputer, akan tetapi juga terkait erat dengan bidang lain seperti Matematika, Psikologi, Pengamatan, Biologi, Filosofi, dan yang lainnya. Kemampuan untuk mengkombinasikan pengetahuan dari semua bidang ini pada akhirnya akan bermanfaat bagi kemajuan dalam upaya menciptakan suatu kecerdasan buatan (Nugraha, Dedi dan Sri Winiarti, 2014).

* + 1. **Defenisi Kecerdasan Buatan**

Kecerdasan buatan berasal dari bahasa inggris “*Artificial Intelligence*” atau disingkat AI, yaitu *intelligence* adalah kata sifat dari cerdas, sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud disini merujuk pada mesin yang mampu berfikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011).

Berikut defenisi kecerdasan buatan yang telah didefenisikan oleh beberapa ahli :

1. Alan Turing

Menetapkan bahwa defenisi kecerdasan buatan yaitu jika komputer tidak dapat dibedakan dengan manusia saat berbincang melalui terminal komputer, maka bisa dikatakan komputer itu cerdas, mempunyai kecerdasan.

1. John McCarthy

Kecerdasan sebagai kemempuan untuk mencapai sukses dalam menyelesaikan suatu permasalahan.

1. Herbert Alexander Simon

Kecerdasan buatan merupakan kawasan penelitian, aplikasi, dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang dalam pndangn manusia adalah cerdas.

1. Rich and Knight

Kecerdasan buatan merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang ada saat ini dapat dilakukan lebih baik oleh manusia.

1. Encuclopedi Britannica

Kecerdasan buatan merupakan cabang dari ilmu komputer yang dalam merepresentasikan pengetahuan lebih banyak menggukan bentuk simbol-simbol dari pada bilangan dan memproses informasi berdasarkan metode heuristis atau dengan berdasarkan sejumlah aturan.

 Kecerdasan buatan merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia. Pada awal diciptakannya, computer hanya difungsikan sebagai alat hitung saja. Namun seiring dengan perkembangan zaman, maka peranan komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu komputer diharapkan dapat diberdayakan untuk mengerjakan segala sesuatu yang bias dikerjakan manusia (Wijaya, Edi, 2013).

 Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki seseorang tentu saja diharapkan akan lebih mampu dalam menyelesaikan permasalahan. Namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang mereka miliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidakakan dapat menyelesaikan masalah denganbaik. Demikian pula, dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik. Agar komputer bisa bertindak seperti dan sebaik manusia, maka komputer juga harusdiberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Untuk itu pada AI akan mencoba untuk memberikan beberapa metoda untuk membekali computer dengan kedua komponen tersebut agar komputer bias menjadi mesin yang pintar.

 Pengertian kecerdasan buatan dapat dipandang dari berbagai sudut pandang, antara lain:

1. Sudut pandang kecerdasan

Kecerdasan buatan akan membuat mesin menjadi ‘cerdas’ (mampu berbuat seperti apa yang dilakukan oleh manusia).

1. Sudut pandang penelitian

Kecerdasan buatan adalah suatu studi bagaimana membuat agar komputer dapat melakukan sesuatu sebaik yang dikerjakan oleh manusia.

1. Sudut pandang bisnis

Kecerdasan buatan adalah kumpulan peralatan yang sangat *powerful* dan metodologis dalam menyelesaikan masalahmasalah bisnis.

1. Sudut pandang pemrograman

Kecerdasan buatan meliputi studi tentangpemrograman simbolik, penyelesaian masalah (*problem solving*) dan pencarian (*searching*).

 Kecerdasan buatan merupakan bidang ilmu komputer yang mempunyai peran penting di era kini dan masa akan datang. Kata *intelligence* adalah kemampuan memahami dan melakukan aksi (Budiharto, Widodo dan Derwin Suhartono, hal : 2, 2014).

 Artificial Intelligence (AI) atau kecerdasan buatan merupakan cabang dari ilmu komputer yang menaruh perhatian pada pengotomatisasi tingkah laku cerdas. Pernyataan tersebut juga dapat dijadikan defenisi dari AI. Defenisi ini menunjukkan bahwa AI adalah bagian dari komputer sehingga harus didasarkan pada sound theoretical (teori suara) dan prinsip - prinsip aplikasi dari bidangnya (Rahman, Fakhrul, Eka Praja Wiyata Mandala, Teri Ade Putra, 2016).

* + 1. **Kecerdasan Alami dan Kecerdasan Buatan**

Kecerdasan alamiah adalah kecerdasan yang dimiliki oleh manusia. Jika dibandingkan dengan kecerdasan buatan, ada beberapa keuntungan kecerdasan buatan dibanding kecerdasan alamiah, yaitu (Wijaya, Edi, 2013) :

1. Lebih permanen

Kecerdasan alamiah akan cepat mengalami perubahan. Hal ini dimungkinkan karena sifat manusia yang pelupa. Kecerdasan buatan tidak akan berubah sepanjang system komputer dan program tidak diubah.

1. Memberikan kemudahan dalam duplikasi dan penyebaran.

Mentransfer pengetahuan manusia dari satu orang ke orang lain membutuhkan proses yang sangat lama, dan keahlian itu juga tidak akanpernah dapat diduplikasi dengan lengkap. Oleh karena itu, jika pengetahuan terletak pada suatu sistem komputer, pengetahuan tersebut dapat disalin dari komputer tersebut dan dapat dipindahkan dengan mudah ke komputer yang lain.

1. Relatif lebih murah dan kecerdasan alamiah.

Menyediakan layanan komputer akan lebih mudah dan lebih murah dibandingkan dengan harus mendatangkan seseorang untuk mengerjakan sejumlah pekerjaan dalam jangka waktu yang sangat lama.

1. Konsisten dan teliti.

Hal ini disebabkan karena kecerdasan buatan adalah bagian dari teknologi komputer. Sedangkan kecerdasan alami akansenantiasa berubah-ubah.

1. Dapat didokumentasi.

Keputusan yang dibuat oleh komputer dapat didokumentasi dengan mudah dengan cara melacak setiap aktivitas dari sistem tersebut. Kecerdasan alami sangat sulit untuk direproduksi.

1. Dapat mengerjakan beberapa *task* dengan lebih cepat dan lebih baik disbanding manusia.

 Sedangkah, keuntungan kecerdasan alamiah dibanding kecerdasan buatan:

1. Bersifat lebih kreatif.

Kemampuan untuk menambah ataupun memenuhi pengetahuan itu sangat melekat pada jiwa manusia. Pada kecerdasan buatan, untuk menambah pengetahuan harus dilakukan melalui sistem yang dibangun.

1. Dapat melakukan proses pembelajaran secara langsung, sementara AI harus mendapatkan masukan berupa simbol dan representasi.
2. Fokus yang luas sebagai referensi untuk pengambilan keputusan, sebaiknya AI menggunakan fokus yang sempit.

Komputer dapat digunakan untuk mengumpulkan informasi tentang obyek, kegiatan (*events*), proses dan dapat memproses sejumlah besar informasi dengan lebih efisien dari yang dapat dikerjakan manusia. Namun di sisi lain, manusia dengan menggunakan insting dapat melakukan hal yang sulit diprogram pada komputer, yaitu kemampuan mengenali (*recognize*) hubungan antara hal-hal tersebut, menilai kualitas dan menemukan pola yang menjelaskan hubungan tersebut.

* + 1. **Ruang Lingkup Kecerdasan Buatan**

Menurut Wijaya, Edi (2013), dalam jurnalnya Adapun ruang Lingkup utama dalam kecerdasan buatan adalah:

1. Sistem Pakar (*Expert System*)

Disini, komputer digunakan sebagai sarana untuk menyimpan pengetahuan parapakar. Dengan demikian, komputer akan memiliki keahlian untuk menyelesaikan permasalahan dengan meniru keahlian yang dimiliki oleh pakar.

1. Pengolahan Bahasa Alami (*NaturalLanguage Processing*).

Dengan pengolahan bahasa alami ini diharapkan *user* dapat berkomunikasi dengan komputer dengan menggunakan bahasa sehari-hari.

1. Pengenalan Ucapan (*Speech Recognition*).

Melalui pengenalan ucapan diharapkan manusia dapat berkomunikasi dengan computer menggunakan suara.

1. Robotika & Sistem Sensor (*Robotics &Sensory Systems*).
2. *Computer Vision*, mencoba untuk dapat menginterpretasikan gambar atau objekobjek tampak melalui komputer.
3. *Intelligent Computer-aided Instruction*.

Komputer dapat digunakan sebagai tutor yang dapat melatih dan mengajar.

1. *Game Playing*.
	1. **Sistem Pakar**

 Secara umum, sistem pakar (expert system) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari pakar (Sunarya, Rahayu dan Dini Destiani, 2016).

 Istilah sistem pakar berasal dari kata knowledge-based expert system. Istilah ini muncul karena untuk memecahkan suatu masalah, sistem pakar menggunakan pengetahuan yang dimasukkan ke dalam komputer. Seseorang yang bukan pakar menggunakan sistem pakar untuk meningkatkan kemampuan pemecahan masalah, sedangkan seorang pakar menggunakan sistem pakar untuk knowledge assistant (Rahman, Fakhrul, Eka Praja Wiyata Mandala, Teri Ade Putra, 2016).

* + 1. **Definisi Sistem Pakar**

 Sistem pakar adalah suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna. Dengan bantuan sistem pakar seseorang yang bukan pakar dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011).

 Sistem Pakar adalah program komputer yang menyimulasikan penilaian san prilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bisang tertentu. Biasanya, sistem seperti ini berisi pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan penambahan basis pengetahuan atau set aturan. Di antaranya banyak sistem pakar yang ada, yang terkenal adalah aplikasi bermain catur dan sistem diagnosis medis (Budiharto, Widodo dan Derwin Suharto, (hal : 132, 2014)).

 Definisi sistem pakar yang paling dikenal adalah:

1. Sebuah model dan prosedur terkait yang memeaparkan, dalam satu domain tertentu, derajat keahlian dalam pemecahan masalah yang sebanding dengan seorang pakar manusia(lgnizio).
2. Sistem pakar adalah sistem komputer yang mengemulasi kemamapuan pengambilan keputusan seorang manusia ahli. (Giarratano & Riley).

 Pemrosesan yang dilakukan oleh sistem pakar merupakan pemprosesan pengetahuan, bukan pemrosesan data pada sistem komputer konversional. Pengetahuan (*knowledge*) adalah pemahaman secara praktis maupun teoritis terhadap suatu objek atau domaintertentu. Pengetahuan yang digunakan pada sistem pakar merupakan serangkaian informasi mengenai gejala-diagnosa, sebab-akibat, aksi-reaksi tentang suatu domain tertentu (misalnya, domain diagnosa medis). Beberapa model representasi pengetahuan yang penting adalah :

1. Jaringan semantic (*Semantik Nets*)
2. Bingkai (*Frame*)
3. Kaidah Prosuksi (*Production Rule*)
4. Logika Predikat (*Predicate Logic*)

 Dengan sistem pakar, permasalahan yang seharusnya hanya dapat diselesaikan oleh para pakar/ahli, dapat diselesaikan oleh orang biasa/ awam. Sedangkan untuk para ahli, sistem pakar membantu aktivitas mereka sebagai asisten yang seolah-olah sudah mempunyai banyakpengalaman. Tabel 2.8 dibawah ini menggambarkan presentase sistem pakar yang dikembangkan pada area tertentu (Budiharto, Widodo dan Derwin Suharto, (hal : 134, 2014)):

**Tabel 2.8 Persentase Sistem Pakar**

|  |  |
| --- | --- |
| **Area** | **Presentase** |
| **Produksi/Manajemen Operasi** | 48% |
| **Keuangan** | 17% |
| **Sistem Informasi** | 12% |
| **Pemasaran** | 10% |
| **Akutansi** | 5% |
| **Bisnis internasional** | 3% |
| **Sumber** | 2% |
| **Lainnya** | 2% |

**Sumber : (Budiharto, Widodo dan erwin Suharto, 2014)**

* + 1. **Kelebihan dan Karakteristik Sistem Pakar**

Sistem pakar banyak digunakan pada aplikasi terkini dan kompleks karena (Budiharto, Widodo dan erwin Suharto, (hal : 134, 2014)):

1. Sistem pakar dapat bertindak sebagai konsultan, instruktur, atau pasangan/rekan
2. Meningkatkan *availability*atau kepakaran tersedia pada semua perangkat komputer
3. Mengurangi bahaya
4. Permanen
5. Pengetahuan dapat tidak lengkap, namun keahlian dapat diperluaskan sesuai kebutuhan. Program konvensional harus “lengkap” sebelum mereka dapat digunakan
6. Database yang cerdas, sistem pakar dapat digunakan untuk mengakses database secara cerdas, misalnya data mining

Sistem pakar biasanya didesain untuk memiliki karakteristik sebagai berikut (Budiharto, Widodo dan erwin Suharto, 2014) :

1. *Hign performance*
2. *Adaquate respone time*
3. *Good realiability*
4. *Understandable*
	* 1. **Kekurangan Sistem Pakar**

Ada beberapa kekurangan yang ada pada sistem pakar, diantaranya (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011) :

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar.
	* 1. **Manfaat Sistem Pakar**

Sistem pakar menjadi sangat populer karena sangat banyak kemampuan dan manfaat yang diberikan, diantaranya (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011) :

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat dari pada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi dilingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Andal, sistem pakar tidak pernak menjadi bosan dan kelelahan.
8. Meningkatkan kapabilitas sistem komputer.
9. Mampu bekerja dengan informasi yang tidak lengkap.
10. Bisa digunakan sebagai media pelengkapdalam pelatihan.
11. Meningkatkan kemampuan untuk menyelesaikanmasalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.
	* 1. **Ciri-ciri Sistem Pakar**

Ciri-ciri dari sistem pakar adalah sebagai berikut (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011) :

1. Terbatas pada domain keahlian tertentu.
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.
3. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami.
4. Bekerja berdasarka n kaidah/*rule* tertentu.
5. Mudah dimodifikasi.
6. Basis pengetahuan dan mekanisme inferensi terpisah.
7. Keluarannya bersifat anjuran.
8. Sistem dapat mengaktifkan *rule* secara searah yang sesuai, dituntun oleh dialog dengan pengguna.
	* 1. **Konsep Dasar Sistem Pakar**

**2.4.6.1. Kepakaran**

Kepakaran merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dan lebih baik dari pada seorang yang bukan pakar (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011).

Kepakaran itu sendiri meliputi pengetahuan tentang :

1. Fakta-fakta tentang bidang permasalahan tertentu.
2. Teori-teori tentang bidang permasalahan tertentu.
3. Aturan-aturan dan prosedur-prosedur menurut bidang permasalhan umumnya.
4. Aturan *heuristic*  yang harus dikerjakan dalam suatu situasi tertentu.
5. Strategi global untuk pemecahan permasalahan.
6. Pengetahuan tentang pengetahuan (*meta knowlwdge*).

**2.4.6.2. Pakar**

Pakar adalah seorang yang mempunyai pengetahuan, pengalaman, dan metode khusus, serta mampu menerapkannya untuk memecahkan masalah atau memberi nasehat. Seorang pakar mampu menjelaskan dan mempelajari hal-hal baru yang berkaitan dengan topik permasalahan, jika perlu harus mampu menyusun kembali pengetahuan-pengetahuan yang didapatkan, dan dapat memecahkan aturan-aturan serta menentukan relevansi kapakarannya. Jadi seorang pakar harus mampu melakukan kegiatan-kegiatan sebagai berikut (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011) :

1. Mengenali dan memformulasikan permasalahan.
2. Memecahkan permasalahan secara cepat dan tepat.
3. Menerangkan pemecahannya.
4. Belajar dari pengalaman.
5. Merestrukturisasi pengetahuan.
6. Memecahkan aturan-aturan.
7. Menentukan relevansi.

**2.4.6.3. *Inferensi***

Inferensi adalah sebuah prosedur (program) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Semua pengetahuan yang dimiliki oleh seorang pakar disimpan pada basis pengetahuan oleh sistem pakar. Tugas mesin inferensi adalah mengambil keputusan berdasarkan basis pengetahuan yang dimilikinya (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011).

Inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau yang diasumsikan. Inferensi adalah konklusi logis atau implikasi berdasarkan informasi yang tersedia. Dalam sistem pakar proses inferensi dilakukan dalam suatu modul yang disebut mesin inferensi (Kusrini, 2008).

**2.4.6.4. Aturan-aturan (*Rule*)**

Kebanyakan *software* sistem pakar komersial adalah sistem yang berbasis *rule* (*rule-based-systems*), yaitu pengetahuan disimpan terutama dalam bentuk *rule,* sebagai proedur-prosedur pemecahan masalah (Sutujo, T , Edy Mulyanto, Vincent Suhartono, 2011).

**2.5 Metode *Bayes***

Teorema Bayes ditemukan oleh Reverend Thomas Bayes (1970-1761). Pada umumnya, teori *bayes* digunakan untuk menghitung nilai kebenaran probabilitas dari suatu evidence. Di dalam teori probabilits dan statistika, teorema *bayes* juga dikenal sebagai sebuah teori dengan dua penafsiran yang berbeda. Dalam penafsiran *bayes*, teori ini menyatakan jauh derajat kepercayaan subjektif harus berubah secara rasional ketika ada petunjuk baru. Teori ini merupakan dasar dari statistika *bayes* dan memiliki penerapan dalam *sains*, rekayasa, ilmu ekonomi, teori *games*, kedokteran, dan hukum.Teori probabilitas *bayes* digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang di dapat dari pengujian. Probabilitas *bayes* menerangkan hubungan antara probabilitas terjadinya hipotesis Hi dengan terdapat fakta (*evidence*) E telah terjadi dan probabilitas terjadinya *evidence* E dengan syarat hipotesis Hi telah terjadi (Budiharto, Widodo dan Derwin Suharto, (hal : 86, 2014)).

Secara garis besar model naive bayes classifier ditunjukkan pada Persamaan (Sutojo,2011) .

𝑝(𝐻|𝐸) = 𝑝(𝐸|𝐻) 𝑥 𝑝(𝐻)𝑝(𝐸)

 𝑝(𝐸)

Dengan:

1. p(H|E) = probabilitas hipotesis H terjadi jikaevidence E terjadi

2. p(E|H) = probabilitas munculnya evidence E jika hipotesis H terjadi

3. p(H) = probabilitas hipotesis H tanpa memandang evidence apapun

4. p(E) = probabilitas evidence E tanpa memandang apapun

Atau dengan kata lain persamaan diatas dapat digambarkan sebagai Persamaan .

𝑃𝑜𝑠𝑡𝑒𝑟𝑖𝑜𝑟 = 𝑃𝑟𝑖𝑜𝑟 𝑥 𝐿𝑖𝑘𝑒𝑙𝑖ℎ𝑜𝑜𝑑

 𝐸𝑣𝑖𝑑𝑒𝑛𝑐𝑒

Perhitungan naive bayes dapat dilakukan dengan langkah berikut ini:

1. Mencari nilai prior untuk tiap-tiap kelas dengan menghitung rata-rata tiap kelas

dengan menggunakan persamaan.

 𝑃 = X

 A

Keterangan :

P = Nilai prior

X = Jumlah data tiap kelas

A = jumlah data seluruh kelas

2. Mencari nilai Likehood untuk tiap-tiap kelas dengan menggunakan persamaan.

 L= 𝐹

 𝐵

Keterangan:

L = Nilai likelihood

F = jumlah data feature tiap kelas

B = jumlah seluruh fitur tiap kelas

3. Mencari nilai posterior dari tiap kelas yang ada dengan menggunakan persamaan.

 P(c|a) = P(c) x P(a|c) (5)

Keterangan:

P(c) = Nilai prior tiap kelas

P(a|c) = Nilai likelihood

Hasil klasifikasi dengan menggunakan Metode *Bayes* dilakukan dengan membandingkan nilai posterior dari kelas-kelas yang ada. Nilai posterior yang paling tinggi yang terpilih sebagai hasil klasifikasi.

**2.6 Bahasa Pemrograman PHP**

**2.6.1 Pegertian PHP**

Hypertext Prepocesor (PHP) adalah bahasa skrip yang dapat ditanamkan atau disisipkan kedalam HTML. PHP banyak dipakai untuk membuat situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS (conten Management System). PHP merupakan bahasa yang banyak digunakan yang bersifat open source (Mandala, Eka Praja Wiyata, 2015).

PHP adalah kependekan dari *PHP : Hyper Text Preprocessor* (rekursif, mengikuti gaya menamaan di \*nix), merupakan bahasa utama *script server-side* yang disisipkan pada HTML yang dijalankan di server, dan juga bisa digunakan untuk pembuatan aplikasi desktop (Sidik, Betha, 2012). Integarsi PHP dengan server web dilakukan dengan teknik CGI *(Common Gateway Interface)*, FastCGI, dan modul server web. Teknik CGI dan FastCGI memisahkan antara server web dan PHP, sedangkan modul server web menjadi PHP sebagai bagian dari server web (Sidik, Betha, 2012).

FastCGI adalah standar baru yang menambahkan dan meningkatkan kemampuan dari program CGI. FastCGI merupakan standar terbuka yang telah dimulai banyak pengguna oleh serverweb komersial dan *opensource*. PHP telah dikembangkan juga menjadi bahasa pemograman *script* yang dapat di jalankan di atas *platform* sistem operasi secara langsung atau bahasa pemograman shell jika di lingkungan \*nix/Linux. PHP dapat digunakan juga untuk membuat program desktop. Kini, pemograman tidak harus mempelajari bahasa pemograman lain, untuk membuat aplikasi desktop, karena PHP juga telah dapat digunakan dan dieksekusi dari desktop. PHP CLI (*Command Line Interface*) adalah istilah untuk PHP yang dapat dieksekusidari sistem operasi (Sidik, Betha, 2012).

**2.6.2 Sejarah PHP**

Rasmus Lerdorf merasa kurang puas dengan sistem yang ada pada saat itu sehingga di menciptakan suatu model interface (antarmuka) yang dapat digunakan untuk menampung informasi tentang para pengunjung situsnya. Pertama kali Rasmus membuat interface dengan menggunakan PERL dan selanjutnya dia mengembangkan dengan menggunakan bahasa C untuk memberikan fleksibilitas pada interface/parser tersebut (Sidik, Betha, 2012).

Pada mulanya, interface tersebut diberi nama *Personal Home Page,* yang memiliki kemampuan untuk mencatat seluruh informasi dari pengunjung situs online-nya. Kemudian, interface atau perser tersebut dimodifikasi dengan mendukung database mSQL atau Mini Structure Query Language dengan menggunakan perser SQL, pengembangan ini diberi nama FI (*Form Interface*). Kemudian PHP FI versi 2.0 diluncurkan dan merupakan awal kelahiran dari PHP yang saat ini sudah mencapai versi 5.x.x. Selanjutnya PHP ini dikembangkan oleh tim untuk memberikan kemampuan yang seimbang dengan aplikasi lainnya (Sidik, Betha, 2012).

Saat ini, Zend menjadi pengembang utama dan telah mendistribusikan Zend Engine-nya untuk perkembangan PHP. Sampai sekarang, pengguna PHP sudah sangat banyak karena kemudahan dan keandalannya di dalam proses pemrograma (Sidik, Betha, 2012).

**2.6.3 Kelebihan PHP**

Kelebihan PHP adalah sebagai berikut (Sidik, Betha, 2012) :

1. PHP dapat di gunakan di berbagai sistem operasi, mulai dari linux, windows, macos dan os yang lain.
2. PHP telah mendukung banyak database, ini mengapa banyak developer webmenggunakan PHP Adabas D Adabas D, dBase dBase, Empress Empress, FilePro (read-only) FilePro (read-only) Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase mSQL, Direct MS-SQL, MySQL, ODBC, Oracle (OCI7 and OCI8), Ovrimos, PostgreSQL SQLite, Solid, Sybase, Velocis, Unix dbm.
3. PHP sudah mendukung on the fly, artinya dengan php anda dapat membuat document text, Word, Excel, PDF, menciptakan image dan flash, juga menciptakan file-fileseperti zip, XML, dan banyak lagi.

**2.7 Database**

**2.7.1 Definisi Database**

Sistem basis data (database) adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data (database) adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat (S, Rosa A dan M.Shalahuddin, 2014).

Sistem informasi tidak dapat dipisahkan dengan kebutuhan akan basis data apa pun bentuknya, entah berupa file teks ataupun Database Management System (DBMS).

Kebutuhan basis data dalam sistem informasi meliputi :

1**.** Memasukkan, menyimpan, dan mengambil data.

2. Membuat laporan berdasarkan data yang disimpan.

**2.7.2 DBMS *(Database Management System)***

DBMS *(Database Management System)* atau dalam bahasa Indonesia sering disebut sebagai Sistem Manajemen Basis Data adalah suatu sistem aplikasi yang digunakan untuk menyimpan, mengelola, dan menampilkan data. Suatu sistem aplikasi disebut DBMS jika memenuhi persyaratan minimal sebagai berikut :

1. Menyediakan fasilitas untuk mengelola akses data.

2. Mampu menangani integritas data.

3. Mampu menangani akses data yang dilakukan secara cepat.

4. Mampu menangani backup data.

Karena pentingnya data bagi suatu organisasi/perusahaan, maka hampir sebagian besar perusahaan memanfaatkan DBMS dalam mengelola data yang mereka miliki. Pengelolaan DBMS sendiri biasanya ditangani oleh tenaga ahli yang spesialis menangani DBMS yang disebut sebagai DBA (Database Administrator).

Berikut ini adalah 4 macam DBMS versi komersial yang paling banyak digunakan di dunia saat ini, yaitu :

1. Oracle.

2. Microsoft SQL Server.

3. IBM DB2.

4. Microsoft Access.

Sedangkan DBMS versi open source yang cukup berkembang dan paling banyak digunakan saat ini adalah sebagai berikut :

1. MySQL.

2. PostgreSQL.

3. Firebird.

4. SQLite.

Hampir semua DBMS mengadopsi SQL sebagai bahasa untuk mengelola data pada DBMS (S, Rosa A dan M.Shalahuddin, 2014).

**2.7.3 Database MySQL**

SQL (Structured Query Language) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus (S, Rosa A dan M.Shalahuddin, 2014).

SQL mulai berkembang pada tahun 1970an. SQL mulai digunakan sebagai standar yang resmi pada tahun 1986 oleh ANSI (American National Standars Institute) dan pada tahun 1987 oleh ISO (International Organization for Standardization) dan disebut sebagai SQL-86. Pada perkembangannya SQL beberapa kali dilakukan revisi. Berikut sejarah perkembangan SQL sampai saat ini :

|  |  |  |
| --- | --- | --- |
| No | Tahun | Nama |
| 1 | 1986 | SQL-86 |
| 2 | 1989 | SQL-89 |
| 3 | 1992 | SQL-92 |
| 4 | 1999 | SQL : 1999 |
| 5 | 2003 | SQL : 2003 |
| 6 | 2006 | SQL : 2006 |
| 7 | 2008 | SQL : 2008 |

Meskipun SQL diadopsi dan diacu sebagai bahasa standar oleh hampir sebagian besar RDBMS yang beredar saat ini, tetapi tidak semua standar yang tercantum dalam SQL diimplementasikan oleh seluruh DBMS tersebut. Sehingga kadang-kadang ada perbedaan perilaku (hasil yang ditampilkan) oleh DBMS yang berbeda padahal query yang dimasukkan sama.

Berikut ini beberapa kelebihan MySQL sebagai database server antara lain (http://hastomo.net/php/pengertian-dan-kelebihan-database-mysql/) :

1. Source MySQL dapat diperoleh dengan mudah dan gratis.

2. Sintaksnya lebih mudah dipahami dan tidak rumit.

3. Pengaksesan database dapat dilakukan dengan mudah.

4. MySQL merupakan program yang multithreaded, sehingga dapat dipasang pada server yang memiliki multiCPU.

5. Didukung program program umum seperti C, C++, Java, Perl, PHP, Python, dsb.

6. Bekerja pada berbagai platform (tersedia berbagai versi untuk berbagai sistem operasi).

7. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem database.

8. Memiliki sistem sekuriti yang cukup baik dengan verifikasi host.

9. Mendukung ODBC untuk sistem operasi Windows.

10. Mendukung record yang memiliki kolom dengan panjang tetap atau panjang bervariasi.

**2.8 Anjing**

 Anjing adalah hewan peliharaan mamalia, bukan hewan liar alami. Mereka awalnya dibiakkan dari serigala. Mereka telah lama dikembangbiakkan manusia, dan merupakan hewan pertama yang harus dijinakkan. Saat ini, beberapa anjing digunakan sebagai hewan peliharaan, yang lainnya digunakan untuk membantu manusia melakukan pekerjaan mereka. Mereka adalah hewan peliharaan yang populer karena mereka biasanya main-main, ramah, setia dan mendengarkan manusia. Anjing sering memiliki pekerjaan, termasuk sebagai anjing polisi, anjing tentara, anjing bantuan, anjing pemadam kebakaran, anjing pembawa pesan, anjing pemburu, anjing penggembala, atau anjing penyelamatan (Alfian Himawan, Nurul Hidayat, Mahardeka Tri Ananta 2018).

**2.8.1 Mencegah Serangan Penyakit Kulit Pada Anjing**

Mencegah penyakit kulit pada anjing merupakan upaya yang bijak. Langkah ini jelas lebih praktis dan ekonomis di bandingkan dengan mengobati anjing yang sudah terserang penyakit kulit. Pengobatan membutuhkan biaya tinggi, waktu, dan juga tenaga. Hasilnya pun belum tentu sempurna. Bahkan ada anjing yang mati meski sudah menghabiskan biaya.

Cara pencegahan yang ampuh adalah dengan menyamak kulit anjing ras sehingga menjadi lebih tahan terhadap serangan penyakit kulit. Upaya ini dilakukan dengan memandikan anjing dengan bermacam-macam formula larutan penyamak kulit. Salah satu di antaranya adalah menggunakan larutan tebakau.

Untuk membuat formula tersebut, tembakau dimasukkan ke dalam panci kemudian diberi air secukupnya, kemudian direbus sampai mendidih 2-3 kali. Setelah didinginkan, larutan tersebut siap digunakan. Mandikan anjing dengan larutan tersebut kemudian diamkan 5-10 menit agar penyamakan lebih sempurna. Setelah itu baru bilas dengan air bersih. Keringkan badan anjing agar tidak pilek. Larutan ini akan melindungi anjing dari gatal-gatal karena serangan kutu da tungau. Pemandian ini dapat dilakukan setiap minggu atau minimal sebulan sekali. Dengan cara ini maka anjing akan terbebas dari serangan penyakit kulit (HS, Wheindrata , 2012).

**2.8.2 Jenis Jenis Penyakit Kulit Pada Anjing**

Kita perlu belajar mendiagnosa tipe penyakit kulit yang menyerang anjing agar dapat mengobatinya secara tepat sasaran. Menurut penyebabnya, penyakit kulit ada berbagai macam, antara lain (HS, Wheindrata , 2012) :

 1. Penyakit Kulit *Demodexcosis*

Penyakit kulit yang di sebabkan oleh demodex sulit di berantas karena tungau ini lebih senang ngendong dan hidup pada pangkal akar rambut anjing dan jarang hidup pada permukaan kulitnya.

 A. Penyebab

Penyebab utamanya adalah serangan *Demodectic Mange* (Tungau *Demodec folliculorum*) atau disebut *Demodex* yang hanya dapat dilihat melalui mikroskop saja karena sangat kecil.

 Faktor faktor yang mempengaruhi :

a. Sanitasi kandang yang kurang bagus.

b. Kotoran dan air kencing kurang cepat dibersihkan.

c. Kandang terlalu lembab, kurang sinar matahari yang masuk, ventilasi kurang baik, dll.

B. Gejala

Gejala yang menjadi ciri dari penyakit ini adalah :

a. Radang dan pembekakan dari folikel rambut dan kelenjar lemak di bawah kulitnya.

b. Penebalan kulit berjalan secara kronis, bulu rontok, kulit kemerahan dan gatal-gatal, sering terlihat menggaruk-garuk dan mengoso-gosokkan badan ke dinding kandang. Bau badannya tidak enak.

c. Terlihat gelisah, lesu, stress, dan kadang demam. Nafsu makan menurun

d. Pertumbuhan terganggu, berat badan merosot dan terlihat kurus.

e. Tidak dapat tidur nyenyak karena gatal.

C. Pengobatan

Pengobatan radikal untuk anjing yang menderita penyakit Demodexcosis yang parah, yang sudah menyerang seluruh tubuh, agar tidak kambuh lagi, gunakan salep insektisid berkadar 20%. Caranya, tambahkan cairan insektisida ke dalam salep atau emulsi. Obat ini akan mengobati secara tuntas semua penyakit gudig pada anjing dengan biaya murah. Semakin tinggi konsentrasi insektisida, semakin cepat penyembuhannya.

2. Penyakit Kulit *Scabies* (Kudis)

 Penyakit *scabies* sering berjangkit pada jenis anjing berbulu pendek. Sebetulnya penyakit ini bersifat kronis dan tidak cepat menular, namun penyembuhannya memakan waktu lama karena harus melunakkan kulit yang telah mengalami penebalan.

 A. Penyebab

Penyebab utamanya adalah tungau atau kutu golongan Sarcoptes Scabiei Canis yang merupakan parasit yang sangat kecil, sulit dilihat dengan mata telanjang.

 Faktor faktor yang mempengaruhi :

a. Sanitasi kandang yang kurang bagus.

b. Air kencing yang menggenang di lantai kandang dapat membasahi tubuh anjing sehingga membuat anjing terserah penyakit ini.

c. Kandang terlalu lembab, kurang sinar matahari yang masuk, ventilasi kurang baik, dll.

d. Banyak kutu, caplak, dan tungau.

e. Luka-luka bukas perkelahian.

B. Gejala

Gejala yang menjadi ciri dari penyakit ini adalah :

a. Rasa gatal yang dilanjutkan dengan terjadinya radang.

b. Penyakit kulit ini sangat cepat menular, mula-mula menyerang bagian kulit yang tipis, terutama daerah ketiak, lipatan bawah perut, dan daun telinga.

c. Terlihat jelas penebalan dan warna kemerah-merahan pada kulit di telinga dan belakang telinga, yang kemudian menjalar ke kaki, belakang tengkuk, dan lpatan paha dan punggung.

d. Bulu langsung rontok, kadang-kadang sampai hilang sama sekali.

e. Nafsu makan menurun sehingga anjing menjadi kurus dan akhirnya akan mati.

C. Pengobatan

 Pengobatan penyakit kulit ini tidak sulit asal dilakuka secara *kontinu*. Caranya ada beberapa pilihan, misalnya menggunakan *Insecticidal Shampoos* atau salep atau krim yang mengandung organo phospat. Formula kedua dengan bahan aktif *Benzoas benzilicus* yang dapat dibuat menjadi salep atau emulsi.

3. Invasi Jamur

Penyebab utamanya adalah jamur *Ringworm*, yaitu penyakit infeksi yang ganas. Penularanya melalui spora yang terbawa oleh papan kayu dan alat-alat peternakan lainnya. Bentuk luka penyakit kulit ini selalu bulat, biasanya di sertai rasa gatal.

 A. Penyebab

Penyebab utamanya adalah serangan dari jamur *Microsporum spp* dan *Tricophyton spp*.

 Faktor faktor yang mempengaruhi :

a. Sanitasi kandang yang kurang bagus.

b. Kandang terlalu lembab dengan temperatur tinggi.

c. Kekurangan vitamin dan mineral sehingga kulit rentan terserang jamur dan gatal-gatal.

d. Anjing yang terserang kutu dan caplak lebih rentan terhadap serangan jamur.

B. Gejala

Gejala yang menjadi ciri dari penyakit ini adalah :

a. Mula-mula terlihat ada bulu rontok pada satu area, berbentuk membulat.

b. Rasa gatal muncul di daerah yang terinfeksi.

c. Area terinfeksi terlihat kemerah-merahan dan kemudian menebal. Warnanya berubah menjadi kelam

d. Bulatan yang terinfeksi mulai bersisik dengan bau yang tidak enak.

e. Nafsu makan menurun dan tidak menentu, kadang tidak mau makan karena sibuk menggaruk. Kadang rasa gatal ini disertai demam.

C. Pengobatan

a. Sebelum di obati sebaiknya anjing di mandikan lebih dahulu dengan sabun *sulfur.* .

b. Oles dengan *Jodium Tincture* area yang terinfeksi jamur.

c. Berikan vitamin dan mineral yang cukup untuk memperbaiki kondisi kulit.

d. Cepat berikan vitamin dan mineral khusus untuk pertumbuhan bulu agar aman dan tidak mudah terserang penyakit jamur.

e. Obat yang paling tepat adalah *antibiotika* khusus untuk jamur yang disebut *Griseofulvin*.

4. Alergi

 Alergi pada kulit anjing terjadi ketika sistem kekebalan tubuh anjing mengalami reaksi berlebihan terhadap zat asing atau partikel berukuran kecil (disebut *alergen*). Sistem kekebalan tubuh yang terlalu aktif ini dapat menyebabkan reaksi berupa alergi. Pengobatan alergi salah satunya adalah dengan melakukan *pest control.*

5. Infeksi bakteri dan kutu

 Kutu termasuk parasit di luar tubuh *(ektoparasit)* yang perlu dikontrol rutin oleh pemilik hewan. Karena kutu akan mengisap darah hewan yang ditumpangi, dengan cara merusak pembuluh darah terkecil *(kapiler).* Bagi hewan yang sensitif, rusaknya *kapiler* darah dan sekresi ludah kutu (yang mengandung *neurotoksin*) akan menimbulkan reaksi alergi, sehingga hewan bersangkutan akan memperlihatkan gejala kegatalan kulit. Hewan yang memiliki cukup banyak kutu akan menggigit-gigit, menggaruk-garuk serta menggosok kulitnya sehingga merusak lapisan luar kulit *(epidermis)* yang kemudian menimbulkan kerusakan (iritasi) pada kulit. Iritasi ini dapat meluas menjadi bentuk peradangan kulit yang bila tidak segera diobati akan menghasilkan bentuk klasik berupa rontoknya rambut yang secara nyata diikuti kebotakan dan penebalan kulit *(hiperkeratosis)*. Akibatnya, selain kulit hewan kesayangan menjadi buruk dan rusak, iritasi lokal kulit akan menimbulkan infeksi yang mempermudah masuknya kuman penyakit seperti virus, jamur, parasit, dan infeksi bakteri.

**BAB III**

**METODOLOGI PENELITIAN**

**3.1 Kerangka Penelitian**

 Makna penelitian secara sederhana adalah bagaimana mengetahui sesuatu yang dilakukan melalui cara tertentu dengan prosedur yang sistematis. Maka penulis membentuk kerangka penelitian sebagai berikut :

**Penelitian** **pendahuluan**

**Pengumpulan** **data**

**Analisa**

**Perancangan**

**Implementasi**

**Pengujian**

**Gambar 3.1 Kerangka Penelitian**

**3.2 Tahapan Penelitian**

Tahapan penelitian ini menjelaskan langkah-langkah dalam melakukan pencatatan data serta mengumpulkan beberapa laporan yang di perlukan untuk dapat dijadikan pedoman dalam pembuatan penelitian ini, yaitu:

**3.2.1 Penelitian Pendahuluan**

Penelitian pendahuluan ini merupakan langkah pertama dalam melakukan suatu penelitian. Penelitian dilaksanakan di R4 Petshop Dan Klinik Hewan. Bertujuan untuk membantu pemilik anjing dalam mengenali penyakit kulit anjing serta solusi penanganannya secara dini, sehingga para pemilik anjing ini dapat mengantisipasi sedini mungkin jika anjing peliharaan mereka terserang penyakit kulit. Serta membantu mengatasi keterbatasan jumlah pakar/dokter yang ada, karena banyak sekali masyarakat yang masih mengandalkan keahlian dari pakar/dokter secara manual. Sehingga biaya yang ditanggung cukup mahal dan dilihat dari waktu juga kurang efesien. Penelitian pendahuluan ini dilakukan dengan cara mendatangi langsung objek yang menjadi pokok dari penelitian.

* + 1. **Pengumpulan Data**

Dalam melakukan proses pengumpulan data, penulis melakukan wawancara secara langsung ke pada Ibu Drh.Rita Mahyona, M.H yang bekerja sebagai Pegawai Negeri Sipil pada Balai Karantina Pertanian Kelas 1 Kota Padang dan dokter hewan di R4 Petshop Dan Klinik Hewan. Penulis mendapatkan beberapa informasi penting yang berhubungan dengan penelitian yang sedang dilakukan.

A. Waktu Penelitian

Penelitian dilakukan dengan memproses data-data yang telah didapat oleh peneliti, pengambilan data dilakukan dari bulan September 2018 sampai dengan selesai. Untuk lebih rincinya dapat dilihat pada tabel dibawah ini :

**Tabel 3.1 Jadwal Penelitian**

|  |  |  |
| --- | --- | --- |
| **No** | **Kegiatan** | **Bulan** |
| **1** | **2** | **3** | **4** |
| **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| **1** | **Perencanaan Penelitian** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2** | **Pengumpulan Data** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3** | **Analisa Data dan Sistem** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | **Perancangan Aplikasi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **Pembuatan Aplikasi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **6** | **Uji Coba** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **7** | **Implementasi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **8** | **Bimbingan Skripsi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | **Penyusunan Skripsi** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

B. Tempat Penelitian

Adapun tempat melakukan penelitian ini dilakukan di R4 Petshop Dan Klinik Hewan yang beralamatkan di Jalan Rimba Datar No. 23 Bandar Buat, Kota Padang, Sumatera Barat.

C. Metode Penelitian

Adapun metode penelitian yang dilakukan sebagai berikut:

* + - 1. Penelitian Lapangan

Untuk menindak lanjuti penelitian dilakukan penelitian lapangan untuk mempelajari penyakit kulit pada anjing dengan cara sebagai berikut :

1. Wawancara

Wawancara diperlukan untuk memperoleh data atau informasi mengenai penyakit kulit pada anjing serta cara menanggulanginya. Wawancara dilakukan dengan Ibu Drh.Rita Mahyona, M.H selaku dokter hewan di R4 Petshop Dan Klinik Hewan, Bandar Buat, Kota Padang, Sumatera Barat.

1. Pengamatan

Pada tahapan ini dilakukan pengamatan lapangan dengan cara melihat serta mempelajari permasalahan tentang gangguan penyakit kulit pada anjing yang diteliti dan mengumpulkan data-data yang bersangkutan dengan penyakit kulit anjing. Penelitian dilaksanakan di R4 Petshop Dan Klinik Hewan, Bandar Buat, Kota Padang, Sumatera Barat.

2. Penelitian Pustaka

Metode yang dilakukan adalah dengan cara mencari bahan tentang penyakit kulit pada anjing yang mendukung dalam pendefinisian masalah, yaitu dengan membaca buku atau browsing internet. Sehingga data-data yang diperoleh dapat digunakan sebagai landasan pada tahap penelitian selanjutnya.

3. Penelitian Laboratorium

Merupakan tahap penelitian yang dilakukan dengan cara *research* labor komputer guna mempraktekkan langsung hasil dari analisa yang bertujuan untuk menguji kebenaran sistem yang dirancang:

**Tabel 3.2 Perangkat Keras Yang Digunakan**

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat Keras** | **Keterangan** |
| 1 | *Processor* | Intel(R) Core (TM) i5-4210U CPU 1.70 GHz 2.40 GHz |
| 2 | *RAM* | 4,00 GB |
| 3 | *Harddisk* | 500 GB |
| 4 | *Laptop* | ASUS X455LN |
| 5 | Media *Input* | *Mouse, Flashdisk* dan Kabel *Usb* |

Perangkat lunak yang digunakan dalam pembangunan Aplikasi ini adalah sebagai berikut:

**Tabel 3.3 Perangkat Lunak Yang Digunakan**

|  |  |  |
| --- | --- | --- |
| **No** | **Perangkat Lunak** | **Keterangan** |
| 1 | Sistem Operasi | Windows 7 Ultimate 64-bit |
| 2 | Aplikasi Browser | Mozila Firefox dan Google Chrome |
| 3 | Microsoft Office | Microsoft Word 2010 |
| 4 | Software Program  | Adobe Dreamweaver CS6, Rational Rose, Astah |
| 5 | Software Pendukung | Mowes dan XAMPP 2.5 |

**3.2.3 Analisa**

1. Analisa Data

Analisa dilakukan untuk mendapatkan suatu fakta berdasarkan metode penelitian yang dilakukan. Data-data yang diperoleh dari metode penelitian yang dilakukan kemudian akan dianalisis sehingga diperoleh seperangkat *aturan (rule base)* yang nantinya akan di asosiasikan dengan data input menggunakan metode *Bayes.*

Data tersebut berisi jenis-jenis penyakit kulit pada anjing, gejala-gejala dari penyakit tersebut serta cara penangulanginya.

1. Analisa Proses

Pada tahap analisa proses ini, penulis menggunakan metode *Bayes.* Dimana sistem ini dapat membantu pakar dalam menyelesaikan masalah penyakit. Hasil yang tidak pasti disebabkan oleh dua faktor, yaitu aturan yang tidak pasti dan jawaban pengguna yang tidak pasti atas suatu pertanyaan yang diajukan oleh sistem pakar.

1. Analisa Sistem

Analisa sistem dapat di definisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan, serta hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang di harapkan sehingga dapat di usulkan perbaikan. Pada tahap analisis sistem ini akan dirancang sebuah sistem pakar yang dibuat berbasis android yang berfungsi untuk memudahkan *user* dalam menggunakan sistem tersebut.

**3.2.4 Perancangan**

1. Perancangan Model

Dalam tahap perancangan ini, Penulis menggunakan metode UML (*Unified Modelling language)*dalam melakukan perancangan model pada sistem pakar ini.

UML (*Unified Modelling language)* yang akan digunakan sebagai berikut :

1. *Use Case Diagram*

Diagram ini menggambarkan fungsionalitas yang di harapkan dari sebuah sistem. Use Case merepresentasikan sebuah interaksi antara aktor dengan sistem pakar. Aktor disini terdiri dari pengguna sistem pakar dan orang-orang lain yang terlibat. Interaksi antara aktor dan sistem dalam bentuk naratif, yang terdiri dari input user dan respon-respon sistem. Selain itu, use case mendeskripsikan kelakuan sistem dari sudut pandang pengguna yang berguna untuk membantu memahami kebutuhan.

1. *Class diagram*

Diagram ini menjelaskan bagai mana hubungan antar *class* pada aplikasi sistem pakar tersebut terjadi, yang terdiri dari *nama class*, *attribute,* dan *operation.*

1. *Sequence Diagram*

Diagram ini menjelaskan urutan-urutan kejadian yang akan terjadi seiring dengan waktu pada saat user menggunakan aplikasi ini, juga bertujuan untuk menggambarkan interaksi antara keseluruhan objek.

1. *Activity Diagram*

Diagram ini menjelaskan bagaimana masing-masing aliran bekerja mulai dari decision yang akan terjadi maupun solusi yang akan didapatkan oleh user.

1. *Deployment Diagram*

Diagram *deployment* menunjukan tata letak sebuah sistem secara fisik. Menggambarkan detail bagai mana komponen di *deployment* dalam infrastruktur sistem pakar, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi.

1. *State Chart Diagram*

Diagram *state chart* digunakan untuk memodelkan perubahan keadaan *(state)* yang dipengaruhi oleh suatu kejadian (*event*) dan waktu, tetapi hanya fokus pada suatu objek.

1. *Communication Diagram* atau lebih dikenal *Collaborasi Diagram*

Diagram ini menjelaskan bagaimana *class* yang ada pada sistem berkolaborasi atau bekerjasama. Diagram ini menampilkan informasi yang sama dengan diagram sequence (*sequence* *diagram*), hanya saja lebih memfokuskan pada kegiatan objek dari waktu informasi itu dikirim.

1. Perancangan Interface

Perancangan *interface* adalah bentuk rancangan tampilan sementara dari pembuatan aplikasi sistem pakar ini. Perancangan ini dibuat untuk memberikan penjelasan tentang tampilan yang dihadapi oleh *user* pada saat menggunakan sistem, sehingga dapat mempermudah dalam mengimplementasikan aplikasi serta akan mempermudah pembangunan aplikasi yang memenuhi prinsip antarmuka yang baik.

**3.2.5 Implementasi**

 Implementasi sistem merupakan tahap memetakan sistem sehingga siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan, sehingga pengguna dapat memberi masukan kepada pengembangan sistem. Pada tahap ini peneliti akan membahas mengenai bahasa pemograman yang akan dipakai. Penelitian ini menggunakan bahasa pemrograman PHP sebagai alat bantu dalam melakukan proses konsultasi dan pengolahan data yang ada. PHP merupakan bahasa pemograman yang sesuai untuk membuat aplikasi *website* dinamis, karena memiliki performa yang tinggi, mudah dipelajari, *multiplatform*, aman, *open source*, serta mudah dikoneksikan dengan berbagai macam *database*.

PHP banyak dipakai untuk pemograman situs *web,* walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. PHP merupakan perangkat lunak *open soure.* Penulisan kode program PHP menyatu dengan HTML yang berjalan pada sisi *server*. Hal ini berarti semua *sintaks* yang telah ditulis akan sepenuhnya dijalankan pada *server*, hanya hasilnya saja yang dikirmkan ke sisi *browser*.

**3.2.6 Pengujian**

Pengujian ini dilakukan dengan membandingkan keluaran atau *Output* dari sistem dengan hasil perhitungan manual terhadap sejumlah rumus yang ada dalam metode *Bayes.*Pengujian aplikasi dilakukan dengan melihat kesesuaian antara *output* yang diberikan sebagai hasil analisis dari aplikasi dengan kondisi yang sebenarnya. Dan setelah pengkodean selesai maka akan dilakukan proses pengujian terhadap aplikasi yang dihasilkan untuk mengetahui apakah aplikasi yang dirancang sudah berjalan dengan benar dan sesuai dengan perancangan yang dilakukan. Pengujian yang akan di lakukan meliputi sebagai berikut :

1. Pengujian Online

Pembuatan program ini nantinya dapat di *online*kan dengan menggunakan fasilitas situs penyedia *subdomain* dan *hosting* gratis. Selanjutnya membuat sebuah *domain* maupun *subdomain* dengan nama *domain* yang diinginkan, lalu *upload* kan *website* dan *database* tersebut, sehingga *website* tersebut dapat di*online*kan. Nantinya setiap pengguna yang ingin memakai aplikasi ini cukup dengan membuka dengan *web browser* mereka dan menuliskan nama *domain* yang telah ditentukan agar dapat menjalankan aplikasi *web* ini. Dengan adanya koneksi *internet,* mereka (pengguna layanan *internet*) dapat mengakses aplikasi *web* ini kapanpun dan dimanapun mereka berada.

1. LAN

Penelitian hanya memonitor melalui *PC* dengan tampilan *web* yang telah dirancang, dilakukan melalui akses jaringan komputer dengan sistem *Local Area Network* (LAN), dengan cara menghubungkan komputer *server* ke komputer *clien* dengan media kabel, yang terlebih dahulu dilakukan *setting* *ip address* pada komputer *server* dan komputer *clien,* setelah terkoneksi*.* Selajutnya seting pada *v-host* *server* dan *v-host* di *system32* pada *windows,* lakukan hal yang sama pada komputer *clien*. Sehingga *clien* dapat mengakses data pada komputer *server*, sedangkan pada komputer *server* menyediakan informasi yang dibutuhkan oleh *clien.*

1. Pengujian Aplikasi

Pengujian ini dilakukan dengan membandingkan keluaran atau *Output* dari sistem dengan hasil perhitungan manual terhadap sejumlah rumus yang ada dalam metode Bayes.

Pengujian aplikasi dilakukan dengan melihat kesesuaian antara *output* yang diberikan sebagai hasil analisis dari aplikasi dengan kondisi yang sebenarnya. Pengujian ini difokuskan pada fungsionalitas dari sistem yang meliputi sebagai berikut:

1. Kesalahan fungsi.
2. Kesalahan struktur data atau pengaksesan *database.*
3. Kesalahan kinerja.
4. Pengujian *Interface*

Pengujian sistem merupakan tahap melakukan *testing* untuk mengetahui kesalahan dalam sistem. Pengujian ini difokuskan pada fungsionalitas dari sistem yang meliputi Kesalahan *interface.*

**BAB IV**

**BIAYA DAN JADWAL KEGIATAN**

Penelitian dilakukan dalam jangka waktu 1 (satu) tahun dengan rincian kegiatan sebagaimana dijelaskan melalui matriks tabel berikut ini

**Tabel 4.1. Rencana Anggaran Biaya Penelitian**

|  |  |  |
| --- | --- | --- |
| **No** | **Jenis Pengeluaran** | **Biaya yang diusulkan (Rp)** |
| 1 | Gaji dan upah | Rp. 1.500.000,- |
| 2 | Peralatan Penunjang | Rp. 1.000.000,- |
| 2 | Bahan habis pakai | Rp. 2.500.000,- |
| 3 | Perjalanan | Rp. 1.000.000,- |
| 4 | Lain – lain (publikasi, seminar) | Rp. 2.000.000,- |

**4.2 Jadwal Penelitian**

Jadwal penelitian dapat dilihat pada *bar chart* dibawah ini :

**Tabel 4.2. Jadwal Penelitian**

|  |  |  |
| --- | --- | --- |
| **No** | **Jenis Kegiatan** | **Bulan** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **I. Persiapan**  |  |
| 1  | Studi Pendahuluan  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2  | Mempelajari Literatur  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3  | Pengumpulan Data  |  |  |  |  |  |  |  |  |  |  |  |  |
| **II. Analisa dan Perancangan** |  |
| 4.  | Analisa Data  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5.  | Analisa Sistem  |  |  |  |  |  |  |  |  |  |  |  |  |

**DAFTAR PUSTAKA**

Ahmad, Hermansyah Nur dan Nirwan Jember Syaf’at. 2013. Rancang Bangun Perangkat Lunak Untuk Pembelajaran Bahasa Sunda.Program Studi Teknik Informatika, Fakultas Ilmu Komputer, STMIK Subang.

Budiharto, Widodo dan Derwin Suhartono. 2014. Artificial Intelligence Konsep dan Penerapan.Andi : Yogyakarta.

Hakim, Arnaz Malikul, Jusak dan Erwin Sutomo, 2015. Sistem Pakar Identifikasi Penyakit Kulit Anjing Menggunakan Metode Certainty Factor. Teknik Informatika, Universitas Brawijaya.

Himawan, Alfian, Nurul Hidayat dan Mahardeka Tri Ananta, 2016. Sistem Diagnosis Penyakit Hewan Pada Anjing Dengan Menggunakan Metode Naive Bayes. Teknik Informatika, Universitas Brawijaya.

HS, Wheindrata. 2012. Buku Pintar Kesehatan Anjing Ras.Andi : Yogyakarta.

Kusrini. 2008. Aplikasi Sistem Pakar Menentukan Faktor Kepastian Pengguna dengan Metode Kuantifikasi Pernyataan. Andi: Yogyakarta.

Mandala, Eka Praja Wiyata. Web Programing Project 1. Andi: Yogyakarta, 2015.

Nugraha, Dedi dan Sri Winiarti, 2014. Pengembangan Media Pembelajaran Sistem Pelacakan Pada mata Kuliah Kecerdasan Buatan Berbasis Multimedia. Program Studi Teknik Informatika, Universitas Ahmad Dahlan.

Pressman, Roger S, Ph.D. 2012. Rekayasa Perangkat Lunak. Andi : Yogyakarta

Rahman, Fakhrul, Eka Praja Wiyata Mandal dan Teri Ade Putra, 2016. Perncangan Aplikasi Sistem Pakar Dengan Menggunakan Metode Certainty Factor Untuk Menentukan Jenis Gangguan Disleksia Berbasis Web. Teknik Informatika, Universitas Putra Indonesia “YPTK”, Padang.

S, Rosa A dan M. Shalahuddin. 2014. Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Bandung: Informatika.

Sidiq, Betha, 2012, Pemograman Web Dengan Php. Bandung : Informatika.

Sunarya, Rahayu dan Dini Destiani, 2016. Pengembangan Sistem Pakar Diagnosis Hama dan Penyakit Pada Tanaman Bawang Merah Berbasis Android. Jurnal Algoritma, Sekolah Tinggi Teknologi Garut: Jayaraga Garut.

Sutojo, T, Edy Mulyanto dan Vincent Suhartono. 2011. Kecardasan Buatan.Andi : Yogyakarta.

Wijaya, Edi. 2013. Analisa Penggunaan algoritma Breadth First Seaerch Dalam Konsep Artificial Intelegencia. STMIK Time: Medan.