

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

Teori yaitu sekumpulan dari sebagian rancangan mencakup maksud serta pemograman untuk memandang peristiwa dengan cara sistem ataupun lewat detail jalinan variabel yang dipakai (Sugiyono 2014). Teori dasar pada bab ini ialah:

2.1.1 Kecerdasan Buatan (Artificial Intelligence)

Kecerdasan buatan atau *artificial intelligence* yakni menunjuk mesin sanggup dan berasumsi, mengukur aksi hendak didapat, serta sanggup mengutip ketetapan semacam yang di coba oleh orang. (T.sutojo, Edy Mulyanto, 2011:1)

(Menurut T.sutojo, Edy Mulyanto, 2011:1-2) Alan Turing, ahli matematika kewarganegaraan inggris yang bergelar bapak komputer modern serta pendobrak sandi Nazi dalam era Perang Dunia II 1950, menetapkan definisi *artificial intelligence*. “Jika komputer tak dapat diperbedakan dengan orang dikala berbicara lewat pusat komputer, maka bisa dibilang prangkat komputer itu pintar, memiliki kepintaran”.

Menurut (T.sutojo, Edy Mulyanto, 2011:13-17) Beberapa sub disiplin ilmu kecerdasan buatan yakni:

1. Sistem pakar ialah sistem yang dirancang sedemikian rupa untuk dapat menyelesaikan masalah seperti seorang pakar dibidang tertentu. Adanya sistem pakar, orang yang bukan ahli bias menjawab pertanyaan dan menyelesaikan masalah seperti yang biasa dilakukan oleh seorang pakar atau ahli.
2. Pengerjaan bahasa alamiah merupakan rancangan yang memiliki kepintaran guna memahami bahasa manusia. Bahasa *natural* merupakan wujud representasi dari sesuatu catatan yang mau dikomunikasikan dampingi manusia. Wujud penting representasi bahasa *natural* ialah berbentuk suara ataupun perkataan, kerap serta diklaim dalam wujud catatan. Dengan pengerjaan bahasa *natural* diharapkan pemakai bisa berbicara dengan komputer lewat bahasa tiap-tiap hari.
3. Pemahaman perkataan ialah pengembangan metode serta sistem yang membuat komputer dapat menyambut masukan berbentuk tutur yang diucapkan.
4. Ilmu robot serta sistem pemeriksaan ialah ilmu pengentahuan serta teknologi rekayasa robot, serta konsep, manufaktur, aplikasi, serta disponsori structural.
5. *Computer vision* ialah ilmu wawasan yang dapat membuat komputer mengidentifikasi subjek yang telah diawasi
6. *Intelligent computer-aided instruction* yaitu *computer* yang mampu dipakai selaku guru yang bisa melatih serta membimbing.

7. *Game playing* yang berarti *game*, ialah sesuatu aktivitas kompleks yang di dalamnya ada peraturan, *play* serta kebiasaan yang bermaksud untuk menata, membatasi serta menentukan *game*.

2.1.2 Sistem Pakar (*Expert System*)

Sistem pakar ialah perwakilan dari *Artificial Intelligence* lumayan lampau sebab kaidah ini sedari di kembangkan tahun 1960. Sistem pakar ada perdana ketika *General-purpose problem solver* (GPS) yang diluaskan dari Newel dan Simon. Hingga dikala telah banyak sistem ahli terbuat, seperti MYCIN guna *diagnosis* penyakit, DENDRAL guna mengidentifikasi struktur elemen gabungan yang tidak diketahui, XCON & XSEL guna menopang bentuk kaidah komputer besar, SHOPIE guna kajian sirkuit elektronik, *prospector* digunakan di bidang ilmu bumi untuk menopang melacak serta menciptakan *deposit*, FOLIO dipakai guna menunjang ketetapan untuk seseorang organisator dalam persediaan serta pemodalan DELTA digunakan buat perawatan induk kereta api listrik diesel. (T.sutojo, Edy Mulyanto, 2011:159-160)

2.1.2.1 Defenisi Sistem Pakar

Menurut (T.sutojo, Edy Mulyanto, 2011:13) sistem pakar kaidah didesain bisa mengikuti kemampuan seseorang ahli dalam menanggapi persoalan serta memecahkan sesuatu permasalahan. Sistem pakar hendak membagikan jalan keluar

sesuatu permasalahan yang diterima dari tanya jawab dengan konsumen. Dengan dukungan sistem pakar seseorang yang bukan ahli atau pakar bisa menanggapi persoalan, menuntaskan, permasalahan dan mengambil keputusan yang bisanya dicoba oleh seseorang ahli.

2.1.2.2 Ciri-ciri Sistem Pakar

Suatu sistem dikatakan sebagai sistem pakar apabila mempunyai ciri sebagai dibawah ini (T.sutojo, Edy Mulyanto, 2011:162) :

1. Tertentu pada daerah kemampuan khusus.
2. Bisa membagikan penalaran buat data-data yang tidak komplit atau pun tidak tentu.
3. Bisa menarangkan bukti menggunakan teknik yang bisa dimengerti.
4. Bertugas bersumber pada kaidah atau rule khusus.
5. Mudah dirubah.
6. Dasar kepintaran serta metode deduksi terbagi.
7. Hasilnya berkarakter saran.
8. Kaidah bisa mengaktifkan kaidah dengan cara pendapat yang cocok, dituntu oleh percakapan bersama pemakai.

2.1.2.3 Struktur Sistem Pakar

Unsur yang sangat berharga dari sistem ialah pakar, lingkungan pengembang serta area diskusi (*consultation environment*). Area pengembangan dipakai pencipta sistem pakar guna membuat bagian-bagiannya serta *mempublikasikan* kepintaran ke dalam *knowledge base*. Area diskusi dipakai oleh pemakai untuk bertanya maka pemakai memperoleh pemahaman serta terlihat sistem pakar seperti bertanya. (T.sutojo, Edy Mulyanto, 2011:166)

2.1.2.4 Komponen Sistem Pakar

Menurut pendapat (Hartati and Iswanti 2008:3–7) struktur sistem pakar memiliki 6 komponen sistem pakar sebagai berikut :

1. Antar muka pengguna

Sistem pakar mengambil alih seseorang pakar di sesuatu konteks tertentu, hingga sistem wajib sediakan bukti yang dibutuhkan bagi pengguna yang tidak menguasai permasalahan teknis. Sistem pakar juga sediakan wacana antara sistem serta penggunanya, yang diucap selaku antar muka.

2. Basis pengetahuan

Basis pengetahuan ialah gabungan kepintaran aspek khusus di jenjang pakar dalam bentuk khusus. Wawasan tersebut diproleh dari pengumpulan pemahaman serta asal mula pengetahuan yang lain semacam yang sudah dituturkan tadinya. Dasar pengetahuan bersifat dinamis, dapat bertumbuh dari durasi ke durasi.

3. Mesin inferensi

Mesin inferensi yaitu pikiran sistem pakar, berbentuk piranti lunak yang melaksanakan kewajiban inferensi pemikiran sistem pakar, sering disebut selaku alat pemikir.

4. Memori kerja

Memori kerja ialah aspek dari sistem pakar yang menaruh kenyataan yang didapat saat diuji konsultasi. Kenyataan esoknya diolah mesin inferensi yang berdasarkan wawasan yang dmasukan di *database* untuk membenarkan sesuatu ketetapan jalan keluar persoalan. Konklusinya dapat berbentuk hasil analisis, aksi, dampak.

5. Fasilitas penjelasan

Fasilitas penjelasan yakni cara memastikan ketetapan yang dicoba oleh mesin inferensi sepanjang tahap konsultasi merepresentasikan cara penalaran seseorang ahli. Sebab pengguna kadang kala tidak lah pakar dalam aspek itu hingga dibuatlah sarana uraian, sarana uraian inilah yang bisa memberikan data pada pengguna tentang jalannya penalaran yang berhasil diperoleh sesuatu ketentuan.

6. Fasilitas akuisi pengetahuan

Akuisi pengetahuan merupakan cara akumulasi, perubahan bentuk akan kemampuan atau keahlian penyelesaian permasalahan yang berawal dari sebagian sumber pengetahuan kedalam wujud yang dipahami oleh komputer.

2.1.2.5 Keuntungan dan Kekurangan Sistem Pakar

Sistem pakar menjadi sangat populer karena amat banyak keahlian serta guna yang diberikan, antara lain (T.sutojo, Edy Mulyanto, 2011:160-161) :

1. Meningkatkan produktivitas, karena sistem pakar bisa beraksi lebih cepat dari pada manusia. Membuat seorang yang tidak mengerti berbuat seperti patutnya mereka ahlinya.
2. Meningkatkan kualitas dan memberikan nasehat tidak berubah-ubah serta kurangi kekeliruan.
3. Mampu menangkap wawasan serta keahlian seseorang.
4. Bisa bekerja di area yang berbahaya.
5. Memudahkan akses pengetahuan seorang ahli.
6. Andal, sistem pakar tidak bias jenuh serta kelelahan atau pun sakit.
7. Meningkatkan kapabilitas sistem pada komputer.
8. Dapat bekerja dengan data yang tidak *komplit* atau pun tidak tentu.
9. Bisa digunakan selaku alat penambah dalam pelatihan.
10. Tingkatkan keahlian untuk menuntaskan permasalahan sebab sistem pakar mengutip asal usul wawasan dari banyak pakar.

Sistem pakar juga mempunyai beberapa kekurangan antara lain :

1. Bayaran amat tinggi guna mengolah serta merawatnya.
2. Susah diluaskan sebab memiliki batas kemampuan serta keberadaan pakar
3. Kaidah pakar tak 100% berkualitas betul.

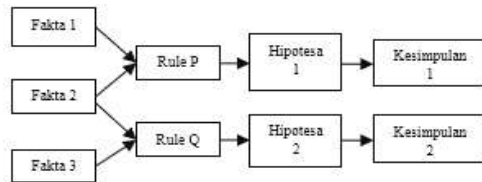
2.1.2.6 Mesin Inferensi Sistem Pakar

Ada 2 Mesin inferensi yang utama pada sistem pakar yaitu pelacakan kedepan dan pelacakan kebelakang yaitu :

1. *Forward chaining* pencarian diawali kenyataan yang dikenal kemudian dicari kesimpulannya. Pelacakan kedepan akan membandingkan fakta dikenal dengan bagian *IF* dari aturan *IF- THEN*.
2. *Backward Chaining* yaitu hipotesis terlebih dahulu kemudian dicoba bukti hipotesis itu serta wajib dicari fakta-fakta yang ada. Proses pencarian diawali dari tujuan, akhirnya merupakan pemecahan yang mau digapai, kemudian dari turan-aturan yang diperoleh, tiap-tiap kesimpulan backward chaining yang memfokus ke kesimpulan ini merupakan akhir yang diinginkan, bila tidak cocok berarti hasil itu bukan solusi yang inginkan.(Sutojo, Edy Mulyanto 2011)

2.1.2.7 Metode Forward Chaining

Forward Chaining adalah pencarian oleh informasi (*data-driven*). Riset pencarian diawali data masukan, serta berikutnya berupaya menjabarkan kesimpulan. Pencarian ke depan mencari kenyataan cocok dengan *IF* dari ketentuan *IF-THEN*. Tata cara pencarian *forward chaining* bisa dipelajari pada gambar dibawah ini:



Gambar 2. 1 Proses *Forward Chaining*

(Sumber: Sutojo, Edy Mulyanto, 2011)

Menurut (Andriani, 2016:15) Pada metode *forward chaining*, pencarian dapat dilakukan dengan dua cara, yaitu :

- a. Memasukan informasi didalam sistem pakar pada tahap diskusi. Metode semacam ini pas serta bermanfaat pada sistem pakar dimana cara di dalamnya spontan mendapat informasi dari *database* ataupun dari sesuatu *set* pemeriksaan.
- b. Memberikan elemen khusus dari data informasi yang didapat sepanjang tahap Tanya jawab dalam sistem pakar. Cara ini kurangi jumlah uang yang dituntut, maka informasi yang dituntut cuma informasi yang betul- betul diperlukan oleh sitem pakar itu yang kelaknnya dipakai untuk mengambil keputusan.

2.2 Variabel (Indikator Masalah)

Variabel penelitian yakni seluruh berupa apa-apa saja yang dipastikan oleh peneliti untuk ditekuni maka didapat data mengenai perihail itu, setelah itu diambil solusinya (Sugiyono, 2014:38). Penelitian ini menjadi variabel adalah penyakit kulit kucing, penyakit ini juga memiliki beberapa jenis, yakni penyakit *ringworm*, penyakit *scabies*, penyakit *pediculosis*, penyakit *alergic dermatitis* dan penyakit *stud tail*.

2.2.1 Penyakit Kulit Pada Kucing

Penyakit kulit ialah penyakit yang biasa dialami oleh kucing, bila tidak tertanggulangi segera serta pas maka bisa segera menyebar sampai bisa mengganggu kegiatan kucing maupun bisa kehilangan nyawa kucing. *Scabies* merupakan satu dari lebih penyakit kucing bisa menimbulkan kematian. Penyakit ini mempunyai pertanda tampaknya rasa gatal berlebih sampai bisa kurangi nafsumakan kucing. Bila kucing yang terkena *scabies* tidak memperoleh penindakan dini kucing itu akan hadapi pengurangan daya tahan tubuh yang *ekstrem* setelah itu mati (Palguna, Putri, Santoso, Marji 2017, hal. 1798). Variabel *Riset* adalah permasalahan kulit kucing yang akan di kelompokkan kedalam beberapa indikator diantaranya sebagaiberikut:

1. Penyakit *Ringworm*

Ringworm atau *dermatophytosis* adalah merupakan penyakit yang diakibatkan oleh jamur yang terkumpul dalam golongan *dermatophytes* yang nyaris seragam

dengan julukan penyakitnya. Tetapi, pada kucing umumnya diakibatkan oleh *Microsporum Canis* ataupun *Trichophyton Mentagrophytes*.



Gambar 2.2 Penyakit *Ringworm*

(Sumber: data olahan 2021)

Gejala yang terdapat pada penyakit *ringworm* pada kucing adalah sebagai berikut :

- a. Terdapat luka atau lesi berbentuk seperti lingkaran
- b. Bulu rontok
- c. Kucing mengalami gatal- gatal
- d. Kulit bersisik
- e. Muncul memar di area kulit yang terinfeksi

2. Penyakit *Scabies*

Scabies ialah penyakit yang kerap melanda pada kucing. Pemicu scabies pada kucing ialah dari kutu tipe scabies ataupun sarcoptes. *Scabies* tidak cuma melanda pada kucing saja, tetapi juga hewan semacam anjing serta kucing pun bisa terserang *scabies*.



Gambar 2.3 Penyakit *Scabies*

(Sumber: data olahan 2021)

Gejala yang terdapat pada penyakit *scabies* pada kucing adalah sebagai berikut:

- a. Kucing mengalami gatal- gatal
- b. Bulu rontok
- c. Timbul luka atau lecet
- d. Kulit memerah
- e. Muncul kerak di sekitar luka

3. Penyakit *Pediculosis*

Pediculosis adalah penyakit kulit pada kucing yang diakibatkan oleh kutu(*lice*).

Pada biasanya, kutu melekat pada bulu serta menimbulkan kerontokan pada rambut sebab mengganggu susunan selaput.



Gambar 2.4 Penyakit *Pediculosis*

(Sumber: data olahan 2021)

Gejala yang terdapat pada penyakit *Pediculosis* pada kucing adalah sebagai berikut :

- a. Terlihat ada serangga kecil di bulu kucing
- b. Terlihat butiran telur dalam bulu kucing
- c. Kucing mengalami gatal-gatal
- d. Menggigiti ekor, lengan, paha, dan bokong terus menerus
- e. Terlihat lemah dan lesu

4. Penyakit *Alergic Dermatitis*

Penyakit kulit ini ialah salah satu penyakit alergi yang mempunyai *respon* kepada produk pemeliharaan, makanan, serta kendala area, semacam serbuk sari ataupun gigitan kutu. Umumnya kucing hendak menggaruk dekat leher ataupun bagian kepala bila hadapi alergi.



Gambar 2.5 Penyakit *Alergic Dermatitis*

(Sumber: data olahan 2021)

Gejala yang terdapat pada penyakit *Alergic Dermatitis* pada kucing adalah sebagai berikut :

- a. Kulit memerah
- b. Bersin terus menerus
- c. Gangguan pencernaan
- d. Bulu rontok
- e. Kucing mengalami gatal- gatal

5. Penyakit *Stud Tail*

Stud Tail adalah situasi yang berlangsung dampak kenaikan ekskresi kelenjar minyak di area pangkal buntut kucing jantan yang tengah hadapi ketidak seimbangan hormone testosterone.



Gambar 2.6 Penyakit *Stud Tail*

(Sumber: data olahan 2021)

Gejala yang terdapat pada penyakit *Stud Tail* pada kucing adalah sebagai berikut :

- a. Bulu ekor rontok
- b. Bulu tipis disekitar ekor
- c. Ekor berminyak dan berbau
- d. Terdapat cairan hitam kecoklatan di ekor

2.3 Software Pendukung

Aplikasi pendukung ialah fitur lunak komputer yang dipakai untuk mengonsep sesuatu sistem. Ada pula aplikasi ataupun fitur lunak yang dipakai di riset ini yakni:

2.3.1 Unified Modeling Language (UML)

Dikala pembentukan fitur lunak muncullah suatu standar metode rancangan mengarah subjek, ialah UML. UML menggambarkan bahasa visual buat pemodelan serta komunikasi dengan memanfaatkan diagram serta pendukung (A.S and Shalahuddin 2011).

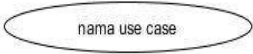
2.3.1.1 Use Case Diagram

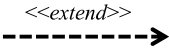
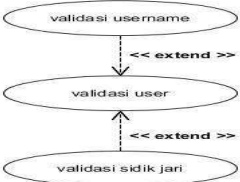
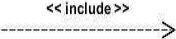
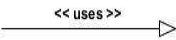
Use case yakni metode langkahyang hendak dicoba sistem data. *Use case* ialah cerminan suatu interaksi antara satu ataupun lebih pemakai kaidah data yang hendak terbuat. Dengan kata lain, *use case* dipakai buat mengenali guna yang terdapat di dalam suatu sistem data serta siapa pemakai yang dapat melaksanakan fungsi itu (A.S and Shalahuddin 2011).

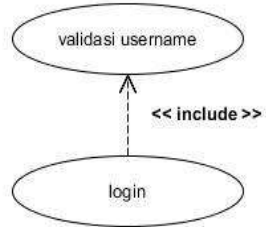
Menurut(A.S and Shalahuddin 2011) Terdapat 2 perihal penting pada *use case* ialah pengertian yang diucap pemakai serta *use case*.

1. Aktor ialah pelaku, teknik ataupun sistem lain berhubungan dengan data yang hendak terbuat pada sistem data sendiri. Dalam perihal ini meski umumnya dilambangkan ilustrasi pemeran, tetapi pemeran belum pasti ialah orang.
2. *Use Case* ialah fungsionalitas diadakan selaku unit-unit silih berganti catatan serta data. Selanjutnya merupakan simbol-simbol yang terdapat bagan *use case*(A.S and Shalahuddin 2011):

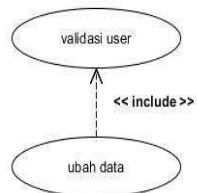
Tabel 2.1 Simbol *Use Case Diagram*

Simbol	Deskripsi
Use Case 	Fungsionalitas diadakan sistem selaku bagian silih beralih catatan, bisa diklaim dengan memakai kata kegiatan di permulaan nama <i>use case</i> .
Aktor / <i>Actor</i> 	Pemakai, cara ataupun struktur berbeda yang berhubungan dengan sistem data yang hendak terbuat didalam struktur data. Dalam perihal tersebut, meski lambang pada pemakai ialah ilustrasi orang, tetapi aktor belum pasti ialah orang, umumnya diklaim memakai istilah barang di mula frase gelar pelaku.
Asosiasi / <i>association</i> 	hubungan antar anggota ataupun pemakai serta <i>use case</i> yang sama-sama mempunyai hubungan.

Ekstensi / <i>extend</i> 	<i>Use Case</i> tambahan mempunyai sebutan awal yang serupa dengan <i>use case</i> dimasukan. 
Generalisasi / <i>generalization</i> 	kaitan biasa–spesial antara 2 pokok <i>use case</i> dimana tugas satu merupakan tugas lebih biasa antara lainnya.
Menggunakan / <i>include</i> / <i>uses</i>  	Kedekatan <i>use case</i> ekstra ke suatu <i>use case</i> dimana <i>use case</i> yang dimasukan lagi. Terdapat 2 ujung penglihatan yang lumayan banyak tentang <i>include</i> di <i>use case</i>: 1. <i>Include</i> berarti <i>use case</i> yang dimasukan hendak di sebut saat <i>use case</i> baru dijalani.



2. *Include* berarti *use case* yang baru hendak senantiasa melaksanakan pemeriksaan apakah *use case* yang baru dimasukan sudah dijalani saat setelah *use case* yang baru dijalani.



Sumber : (A.S & Shalahuddin,2011)

2.3.1.2 Activity Diagram

Diagram aktivitas melukiskan alur kerja ataupun kegiatan antara suatu *system* yang hendak terbuat. Serta tutur lain bagan kegiatan cuma menerangkan kegiatan *system* tidak apa yang dicoba oleh pengguna.(A.S and Shalahuddin 2011).

kegiatan dipakai untuk mendeskripsikan keadaan selanjutnya(A.S and Shalahuddin 2011):

1. Rancangan metode aspek upaya dimana masing-masing antrean kegiatan yang ditafsirkan..
2. Antrean atau pengelompokan wujud dari sistem ataupun *use interface* mempunyai sesuatu rancangan antar muka wujud.
3. Skema pengujian dimana masing-masing gerakan diprediksi memerlukan uji coba demi diartikan permasalahannya.
4. Rancangan menu yang diperlihatkan pada *software*.

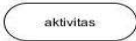
Selanjutnya merupakan simbol yang terdapat di kegiatan (A.S and Shalahuddin 2011):

Tabel 2.2 Simbol *Activity Diagram*

Simbol	Deksripsi
Status awal 	Status dini kegiatan <i>system</i> ialah posisi dini sistem dikala dijalani.

Sumber: Rosa A.S dan M. Shalahuddin(2011: 134-135)

Tabel 2.2 Lanjutan

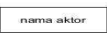
Simbol	Deksripsi
Aktivitas 	Kegiatan yang diuji oleh sistem bisa dimulai dengan tutur kegiatan.
Percabangan / <i>decision</i> 	Perhimpunan percabangan yang mempunyai opsi kegiatan lebih dari satu alternatif.
Penggabungan / <i>join</i> 	Asosiasi penyatuan dimana dicoba penyatuan kegiatan jadi satu
Status akhir 	Keadaan puncak yang dicoba sistem kala berakhir menjalankan
<i>Swimlane</i>  Atau 	Membedakan badan bidang usaha yang memiliki tugas kepada kegiatan terjalin.

2.3.1.3 Sequence Diagram

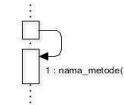
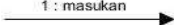
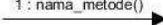
Diagram sekuen secara umum berfungsi buat melukiskan aksi subjek pada use case dengan merujuk pada durasi hidup subjek serta catatan yang dikirimkan serta didapat objek. Kuantitas diagram sekuen dilukiskan dalam sesuatu bentuk merupakan sebesar banyak arti *use case*, maka terus menjadi banyak *use case* yang diartikan hingga bagan sekuen yang mesti terbuat pula terus menjadi besar (A.S and Shalahuddin 2011).

Selanjutnya karakter simbol yang terdapat di diagram sekuen(A.S and Shalahuddin 2011):

Tabel 2.3 Simbol *Sequence Diagram*

Simbol	Deksripsi
 Aktor Atau  Tanpa waktu aktif	Orang, cara, ataupun sistem lain yang berhubungan sistem data, meski lambang pengguna merupakan lukisan aktor tidak mesti ialah orang.
Garis hidup / <i>lifeline</i> 	Melaporkan gerakan ataupun aksi sesuatu subjek dikala sistem dijalani.
Objek 	Melaporkan subjek yang berhubungan agar beralih catatan serta data.
Waktu aktif 	Melaporkan subjek dalam kondisi hidup melaksanakan interaksi dengan beralih catatan.

Tabel 2.3 Lanjutan

Simbol	Deksripsi
Pesan tipe <i>create</i> 	Melaporkan sesuatu subjek membuat subjek lain membidik pada subjek terkini.
Pesan tipe <i>call</i> 	Melaporkan subjek menyebut tata cara pada subjek lain ataupun dirinya sendiri, Arah panah membidik pada subjek pengguna pembedahan atau tata cara.
Pesan tipe <i>send</i> 	Melaporkan jika sesuatu subjek bisa mengirimkan dan masukan data ke subjek yang lain, dengan acuan arah panah membidik pada subjek yang menyambut kembalian
Pesan tipe <i>return</i> 	Melaporkan sesuatu subjek yang sudah melaksanakan sesuatu subjek tata cara bisa mengirim catatan kembali ke subjek khusus, arah panah membidik pada subjek yang dikirim
Pesan tipe <i>destroy</i> 	Melaporkan sesuatu subjek bisa memusnahkan subjek lain, arah panah subjek membidik pada subjek yang diakhiri, hendaknya bila terdapat <i>create</i> hingga terdapat <i>destroy</i>

Sumber : Rosa A.S dan M. Shalahuddin (2011:138-139)

2.3.1.4 Class Diagram

Kelas diagram menjelaskan bentuk aturan dari bidang pengartian kelas kategori yang hendak dikerjakan menjadi sistem. Kategori kelas mempunyai apa yang diucap ciri serta tata cara ataupun operasi.

1. Ciri ialah variabel yang dipunyai oleh sesuatu kategori kelas.
2. Pembedahan ataupun tata cara merupakan kegunaan yang dipunyai oleh sesuatu kategori kelas.

Kategori yang terdapat pada bentuk sistem mesti bisa menjalankan kegunaan yang searah dengan keperluan sistem. Urutan format kategori yang bagus pada bagan kelas hendaknya mempunyai tipe-tipe kategori seperti dibawah ini(A.S and Shalahuddin 2011):

1. Kelas *main*

Kategori yang mempunyai guna dini dieksekusi kala sistem dijalankan.

2. Kelas mengatasi bentuk sistem

Kelas mengartikan, mengendalikan bentuk pengguna.

3. Kelas dimuat penjabaran *use case*

Kelas mengatasi kegunaan yang wajib dimuat penjabaran *use case*, umumnya diucap kategori cara mengatasi cara bidang usaha pada *software*.

4. Kelas yang dimuat dari penjabaran fakta

Kategori yang dipakai untuk menggenggam ataupun membalut informasi jadi suatu kesatuan yang didapat ataupun hendak ditaruh ke dasar informasi.

Berikut adalah lambang pada diagram kelas(A.S and Shalahuddin 2011):

Tabel 2.4 Simbol *Class Diagram*

Simbol	Deskripsi
Kelas 	Kategori bentuk system
Antarmuka / <i>interface</i> 	Serupa rancangan antarmuka di rancangan berpusat subjek
Asosiasi / <i>association</i> 	Kedekatan antar kelas dengan arti biasa, federasi umumnya pula diiringi dengan multiplicity
Asosiasi Berarah / <i>directed association</i> 	Kedekatan antar kelas dengan arti biasa, federasi umumnya pula diiringi dengan multiplicity
Generalisasi 	kedekatan antar kelas memiliki arti generalisasi=spesialisasi (umum-khusus)
Kebergantungan / <i>dependency</i> 	kedekatan antarkelas dengan maksud generalisasi-spesialisasi (umum-khusus)
Agregasi / <i>aggregation</i> 	Kedekatan antarkelas memiliki arti seluruh elemen (<i>whole-part</i>).

Sumber : Rosa A.S dan M.Shalahuddin

2.3.2 XAMPP (*X*apache *M*ySQL *P*HP)

Menurut(Safitri 2017). *XAMPP* ialah bagian *PHP* dan *MYSQL* berbasis *open source* berlaku seperti alat menolong meluaskan pelaksanaan berplatform *PHP*, *XAMPP* memadukan sebagian fitur lunak di dalam suatu bagian.

XAMPP ialah suatu *software* yang dipakai selaku server local yang di gunakan untuk melaksanakan suatu program seperti *PHP*, *HTML* dan pembantunya. Agar memperoleh aplikasi *XAMPP* ini dapat segera diunduh di *web* aslinya.



Gambar 2.7 Logo *XAMPP*

(Sumber: Ariyanti Safitri, (2017:2)

2.3.2.1 *PHP: Hypertext Preprocessor (PHP)*

Menurut(Palit, Rindengan, and Lumenta 2015) *PHP* yakni rancangan yang dipakai dengan cara meluas untuk tindakan penciptaan dan perluasan suatu *web* dan dapat dipakai di *HTML*. *PHP* kependekan “*PHP :Hypertext Preprocessor*”, menggambarkan rancangan dalam *HTML*, juga bekerja di bagian *server* (*server-sideHTML-embedded scripting*). Maksudnya tugas yang diserahkan hendak seluruhnya dijalani di server namun diikutkan pada lembaran *HTML*, maka *script*-nya tidak terlihat disamping *pelanggan*.

PHP (PHP :Hypertext Preprocessor) menggambarkan bahasa pemrograman untuk pembangunan sesuatu rencana aplikasi ibarat *web*, *PHP* mampu diikuti berbarengan bersama *HTML* berisi *ssintaknya* dengan perluasan data.



Gambar 2.8 Logo *PHP*

(Sumber: Randi Palit, 2015: 3)

2.3.2.2 HTML

HTML merlambangkan pendekatan tutur kata sangat bawah serta berarti dipakai guna menunjukkan serta mengatur bentuk pada laman *web*. Berdasarkan asal yang penyusun ambil pada *Wikipedia*, *HTML* dipakai guna menunjukkan bermacam data suatu penjajah *web* internet serta *formatting hypertext* biasa yang dicatat ke dalam berkah bentuk ASCII supaya bisa menciptakan bentuk bentuk yang berintegrasi (Saputra 2012).

HTML (Hyper Text Markup Language) dimanfaatkan demi menciptakan beranda situs serta menunjukkan sebagian data serupa kepentingan.



Gambar 2.9 Logo *HTML*

(Sumber: Agus Saputra, 2012:1)

2.3.2.3 *CSS (Cascading Style Sheet)*

CSS yakni bahasa perancangan situs di gambarkan spesial guna mengatur serta menciptakan bermacam bagian dalam website lebih tertata serta serupa. *CSS* ialah suatu perancangan harus *html* serta mesti di mengerti oleh tiap pelaku pemrograman *web*,



Gambar 2.10 Logo *CSS*

(Sumber: Agus Saputra, 2012:27-29)

Maksud pokok *CSS* ialah untuk membedakan tema penting dri bentuk arsip yang lain. situs yang memakai *CSS* hendak bertambah mudah serta ringan untuk

aktifkan dari pada situs tidak memakai *CSS*. Dengan menggunakan *CSS*, begitu besar manfaat yang bisa di dapat, yaitu (Saputra 2012):

1. menguraikan penyusunan dokumen (*CSS dan HTML*).
2. Meringankan serta mempercepat penciptaan serta perawatan dokumen *web*.
3. Jalan situs sangat pesat saat proses (memesatkan artikulasi *HTML*).
4. Mudah, interaktif, bentuk lebih menarik serta aman ditatap.
5. Sangat sedikit dimensi data alhasil kapasitas yang dipakai juga langsung menjadi sangan kecil.
6. Bisa dipakai ke seluruh situs pencarian.

2.3.2.4 *JavaScript dan jQuery*

Menurut(B 2014)*JavaScript* merupakan tutur kata scripting kecil, mudah, mengarah subjek yang diletakan pada kode *HTML* dan di olah dibagian pelanggan. *JavaScript* dipakai dalam pembangunan web supaya lebih aktif dengan membagikan keahlian baru kepada *HTML* melintasi arahan di bagian pencarian. *JavaScript* mampu menjawab arahan pengguna dengan sangat gesit serta mambuat lembaran web menjadi sangat tanggap. *JavaScript* mempunyai susunan *simpel*, simbolnya bisa diselipkan di dalam catatan *HTML* maupun bertumpu menjadi satu program.

JavaScript dan *jQuery* yakni kode- kode program yang mempunyai bahasa pemrograman tingkatan atas serta mempunyai bentuk bahasa yang *simpel*, *JavaScript*

serta menginstruksikan serta melaksanakan perintah dibagian peramban dan diolah dibagian pengguna. Yang dimasukan dalam kode *HTML*.



Gambar 2.11 Logo *JavaScript*

(Sumber: Indra Yatini, 2014: 2)

2.3.2.5 *MySQL*

Menurut(Palit et al. 2015), *MySQL* adalah data *baseserver* sumber terbuka yang lumayan dikenal kehadirannya. Begitu bermacam kelebihan dipunyai, memakai aplikasi basisdata dipakai para pegiat guna membuat sesuatu proyek. Menggunakan layanan API dipunyai oleh *Mysql*, membolehkan beragam penerapan pc dicatat dengan bermacam penerapan perancangan bisa membuka basis laporan *MySQL*.



Gambar 2.12 Logo *MySQL*

(Sumber: Randi Palit, 2015: 4)

MySQL ialah salah penyimpanan golongan dunia yang amat sesuai apabila dipadukan dengan bahasa perancangan *PHP*. *MySQL* bertugas memakai *SQL* (*Structure Query Language*) yang ialah bahasa umum yang dipakai guna memanfaatkan basis data. Biasanya arahan yang sangat kerap dipakai di *MySQL* adalah mengambil, menambah, mengubah, serta menghapus (Saputra 2012). *MySQL* ialah *software* guna mengelola basis data, *MySQL* dipakai guna mengadaptasi ataupun tipu daya basis data.

2.3.2.6 Notepad++

Menurut (Palevi 2013) *Notepad++* yakni satu buah aplikasi *text* pengedit yang bertabiat *free*. *Notepad* menitik beratkan manfaat penerapan mengubah tulisan di durasi yang kilat serta efisien. *Notepad++* *mensupport* bentuk kaidah perancangan semacam *PHP*, *HTML*, *JavaScript* dan *CSS*. Implementasi ini bisa diunduh dengan cara leluasa di “*notepad-plusplus.org*”.

Notepad++ ialah fitur lunak pengedit tulisan yang dipakai guna menjalankan satu buah rancangan implememtasi memakai tanda-tanda khusus serta perancangan ini *mensupport* bentuk kaidah perancangan semacam *PHP*, *HTML*, *JavaScript* dan *CSS*.



Gambar 2.13 Logo *Notepad++*

(Sumber: Krisnawati, 2013: 3)

2.4 Penelitian Terdahulu

Beberapa lampiran penelitian terdahulu mengenai sistem pakar didalam jurnal menjadi referensi bagi penulis dalam laporan penelitian sebagai berikut:

1. (Petrus, Sukma, and Gafrun 2019) Sistem Pakar Diagnosa Penyakit Kucing Berbasis Web Menggunakan Metode Forward Chaining, Volume 10 No 2. Sedikitnya informasi perihwal penyakit kucing serta pula sedikitnya pemahaman mengenai berartinya menjaga kesehatan kucing piaraan menyebabkan ragam kucing yang bukan terpelihara kesehatannya. Memakai penerapan berbasis situs, oleh dari sesuatu ahli hendak ringan diterima bagi pemakai, tak mesti sampai pada seseorang pakar atau orang yang berpengalaman pada bidangnya, maka sebab itu guna menanggulangi kasus itu hingga butuh diciptakan sistem ahli yang sanggup melaksanakan analisis penyakit dari kucing dengan memperhitungkan tanda-tanda terdapat dari kucing yang tengah tidak sehat.
2. (Rezza Ramadhan, Indah Fitri Astuti 2014) Sistem Pakar *Diagnosis* Penyakit Kulit Pada Kucing Menggunakan Metode *Certainty Factor David*, Vol2, No.1.

Maksud dari *riset* yakni mewujudkan aplikasi sistem ahli buat analisis penyakit kucing Persia. Sistem ini mendiagnosis penyakit secara *forward chaining* serta membagi angka mungkin dengan *Certainty Factor*. Teknik meyakinkan apakah sesuatu kenyataan itu jelas ataupun tak tentu berupa *metric* yang dasarnya dipakai di kaidah, amat sesuai untuk sistem ahli yang menelaah suatu tidak mementu. Aspek kejelasan melaporkan kepercayaan dalam suatu peristiwa ataupun kenyataan dalam hipotesa bersumber pada fakta atau pun evaluasi ahli. Hasil riset berbentuk aplikasi sistem ahli yang bisa mendiagnosis penyakit kucing persia dengan 99% teliti menolong, menanggulangi kasus *problem* kucing persia.

3. (Bela, Putri, and Santoso 2017) Diagnosis Penyakit Kulit Pada Kucing Memakai Metode *Modified K-Nearest Neighbor*, Vol.1, No.12. Pertanda dini dari penyakit kulit sering-kali tak kelihatan serta tidak begitu mengganggu, oleh sebab itu sering-kali kucing juga kelihatan baik saja akibatnya *owner* tak mengetahui. Amat sedikitnya wawasan *owner* mengenai penyakit kulit dirasakan kucing dan kesamaan tanda-tanda dari bermacam penyakit kulit yang susah dikenali oleh orang biasa jadi sebab penting penulis untuk melaksanakan riset mengenai analisa penyakit kulit kucing memakai teknik *Modified K-Nearest Neighbor* dipakai guna pengaturan fakta baru yang pengelompokannya tak diketahui beralaskan nilai *k* terdekat.
4. (Astono et al. 2019) Sistem Pakar Diagnosa Penyakit Kucing Feline Virus Menggunakan Metode *Certainty Factor* Berbasis *Web*, Volume VI Nomor 2.

Sistem pakar ini dibuat menerapkan kaidah perancangan PHP dengan metode *Certainty Factor* serta selaku penerima informasinya hendak memakai *Database MySQL*. Sistem Pakar yang dibangun menciptakan output analisis penyakit yang bersumber pada pengguna yang memasukan Sistem ahli pula bermanfaat selaku perlengkapan menaruh wawasan seseorang ahli spesialisnya ahli penyakit kucing. Dengan terdapatnya sistem ahli ini diharapkan warga bisa mendapatkan data hal penyakit yang dirasakan kucing.

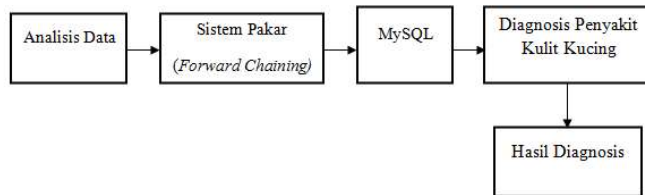
5. (Kurniati et al. 2017) Diagnosa Penyakit Parasit Pada Kucing Menggunakan Metode *Certainty Factor* (Studi Kasus : Puskewan Cibadak Kabupaten Sukabumi, Vol.4, No.2. Kucing yang terserang parasit bisa alami penurunan tubuh kendala dari parasit bisa menyebabkan terdapatnya darah pada kulit dampak luka cedera serta kulit jadi kasar benalu merupakan, hewan yang hidup dengan metode menempel pada insan lainnya, meresap nutrisi, serta tak berikan khasiat padanya tentu perihal itu hendak memberatkan yang ditemplei, ilustrasi benalu misalnya kutu kucing. Beralasan hambatan di temui di buat sistem ahli menggunakan tata cara *Certainty factor* ialah tata cara inferensi yang melaksanakan kajian dari sesuatu permasalahan pada jalan keluarnya.
6. (Munthe et al. 2019) *Expert System for Diagnosing Diseases in Cats that Can Contagious to Humans Using the Certainty Factor Method*. Vol.3 No.4, Sebagian penyakit pada kucing apalagi dapat menular ke orang. Untuk itu dicoba riset untuk mengonsep sistem pakar buat mendiagnosis penyakit pada kucing yang bisa menular ke orang dengan memakai metode *Certainty Factor*, alhasil

owner kucing bisa mengidentifikasi pertanda penyakit pada kucing peliharaanannya. Hasil pengujian *black box* membuktikan kalau fitur-fitur aplikasi sudah berjalan dengan bagus. Sebaliknya percobaan keabsahan membuktikan kalau informasi cocok dengan buku referensi.

7. (Aulia et al. 2019) *Android Application to Detect Cat Disease Using an Expert System*. Vol.4, No.5. Penelitian ini adalah untuk mengembangkan sebuah aplikasi android bernama “TANYA MEOW” yang bisa dipakai oleh *owner* kucing buat mengetahui penyakit yang dialami kucingnya bersumber pada pertanda yang nampak. *Owner* kucing butuh menanggapi kuesioner untuk mengenali penyakit apa yang dialami kucingnya. Dampak dari penyakit itu pula diiringi dengan penyembuhan awal yang dapat dicoba *owner* kucing untuk menjaga kucingnya.

2.5 Kerangka Pemikiran

Kerangka pemikiran ialah suatu rancangan ataupun jalan kegiatan dalam memecahkan kasus *riset*. Kerangka pemikiran diawali dari kasus hingga pendapatan tujuan. Bersumber pada ulasan hingga pengarang membuat suatu kerangka pemikiran yakni:



Gambar 2.14 Kerangka Berpikir

(Sumber: data olahan 2021)

Informasi diperlukan berhubungan penyakit kulit pada kucing dianalisis dulu supaya lebih simpel serta gampang dicoba cara pengerjaan informasinya. Data-datanya itu setelah itu diolah memakai sistem pakar memakai metode *forward chaining*. Sistem pakar yang memanfaatkan perancangan pelacakan kedepan ini memakai basis data *MySQL* yang mampu difungsikan untuk mendiagnossis penyakit kulit pada kucing dan mewujudkan keluaran analisis.