

SISTEM CERDAS BERBASIS MACHINE LEARNING UNTUK DIAGNOSIS PENYAKIT PADA KUCING

**Ade Setiawan¹, Renatan Hosea Silap², Rio Fahrezy³, Syifa Nur Rakhmah⁴, Findi Ayu Sariasih⁵,
Imam Sutoyo⁶**

Program Studi Informatika, Universitas Bina Sarana Informatika
^{1,2,3,4,5,6}Jl. Kramat Raya No.98, RT.2/RW.9, Kwitang, Kec. Senen, Kota Jakarta Pusat, Daerah
Khusus Ibukota Jakarta 10450

Email : ¹*adestwn003@gmail.com*, ²*renatanhoseasilap@gmail.com*, ³*riofahrezy6@gmail.com*,
⁴*syifa.snk@bsi.ac.id*, ⁵*findi.fav@bsi.ac.id*, ⁶*imam.ity@bsi.ac.id*

ABSTRAK

Tantangan dalam mendiagnosis penyakit kucing secara cepat dan akurat, yang disebabkan oleh kecenderungan kucing menyembunyikan rasa sakit serta gejala klinis yang seringkali tidak spesifik, menjadi latar belakang utama penelitian ini. Penelitian ini bertujuan untuk merancang dan membangun sebuah sistem cerdas berbasis *machine learning* yang dapat memberikan rekomendasi diagnosis awal penyakit umum pada kucing berdasarkan data gejala. Metode penelitian yang digunakan mengadopsi kerangka kerja Agile Scrum, dengan algoritma K-Nearest Neighbor (KNN) sebagai inti klasifikasi. Pengujian sistem dilakukan terhadap 200 data uji yang mencakup lima penyakit utama, yaitu Flu Kucing, Cacingan, Infeksi Jamur, Rabies, dan Diare. Hasil pengujian menunjukkan performa yang sangat baik dengan tingkat akurasi rata-rata mencapai 92,50%. Secara spesifik, sistem berhasil mengklasifikasikan 185 data dengan benar dan 15 data salah, dengan Rabies mencatat akurasi tertinggi (96,67%). Meskipun masih terdapat tingkat kesalahan 7,5%, sistem ini terbukti layak digunakan sebagai alat bantu diagnosis awal, namun penggunaannya tetap harus didukung oleh konfirmasi langsung dari dokter hewan profesional.

Keyword : machine learning, agile scrum, K-Nearest Neighbor, klasifikasi, kucing

ABSTRACT

The challenge of diagnosing cat diseases quickly and accurately, caused by the tendency of cats to hide pain and often non-specific clinical symptoms, forms the primary background of this research. This study aims to design and build an intelligent system based on machine learning that can provide initial diagnostic recommendations for common cat diseases based on symptom data. The research method used adopts the Agile Scrum framework, with the K-Nearest Neighbor (KNN) algorithm as the classification core. System testing was conducted on 200 test data points covering five main diseases: Cat Flu, Worms, Fungal Infection, Rabies, and Diarrhea. The test results showed excellent performance with an average accuracy rate of 92.50%. Specifically, the system successfully classified 185 data points correctly and 15 incorrectly, with Rabies recording the highest accuracy (96.67%). Although there is still an error rate of 7.5%, this system is proven feasible for use as an initial diagnostic aid; however, its use must still be supported by direct confirmation from a professional veterinarian.

Keyword : machine learning, agile scrum, K-Nearest Neighbor, classification, cat

1. PENDAHULUAN

Seiring dengan perkembangan Revolusi Industri 4.0, penerapan kecerdasan buatan (AI) dan *machine learning* telah terbukti memberikan dampak transformatif dalam bidang kedokteran. Sistem cerdas mampu menganalisis kumpulan data besar untuk menemukan pola tersembunyi yang dapat membantu dalam penegakan diagnosis. Beberapa penelitian telah berhasil menerapkan metode *machine learning* untuk diagnosis penyakit pada hewan. Sebagai contoh, penelitian oleh menunjukkan bahwa model AI dapat digunakan untuk mendiagnosis penyakit pada hewan ternak dengan akurasi yang tinggi. Di Indonesia, penelitian serupa juga telah dilakukan, seperti pengembangan sistem pakar untuk mendiagnosis penyakit kulit pada kucing menggunakan metode *forward chaining* [1].

Kucing merupakan salah satu hewan peliharaan yang paling banyak dipelihara oleh manusia[2]. Kucing sangat populer dikalangan masyarakat Indonesia, karena fisik dan tingkah yang sangat mengemaskan menjadi salah satu alasan menyukai hewan peliharaan tersebut, akan tetapi tidak sebanding dengan ketersediaan dokter yang mencukupi dan pemilik juga tidak mengetahui cara pemeliharaan yang benar[3]. Salah satu tantangan utama yang dihadapi oleh pemilik hewan dan praktisi veteriner adalah proses diagnosis penyakit yang cepat dan akurat. Kucing memiliki insting untuk menyembunyikan rasa sakit sebagai mekanisme pertahanan diri, yang menyebabkan gejala klinis seringkali baru terdeteksi pada stadium lanjut [4]. Kondisi ini secara signifikan dapat mempersulit penanganan dan menurunkan prognosis kesembuhan pasien.

Diagnosis konvensional yang dilakukan oleh dokter hewan, meskipun merupakan standar emas, seringkali bergantung pada serangkaian pemeriksaan penunjang yang mahal dan memakan waktu, seperti tes darah, urinalisis, dan pencitraan diagnostik. Selain itu, banyak penyakit pada kucing menunjukkan gejala awal yang serupa dan tidak spesifik, seperti anoreksia, letargi, dan demam, sehingga diagnosis diferensial menjadi sangat luas [5].

Berdasarkan tinjauan latar belakang dan studi literatur, masalah-masalah yang dapat diidentifikasi adalah sebagai berikut:

1. Proses diagnosis penyakit pada kucing seringkali berjalan lambat dan rumit. Hal ini disebabkan oleh sifat alami kucing yang menyembunyikan gejala serta banyaknya tanda klinis awal yang tidak spesifik, sehingga memerlukan proses eliminasi yang panjang dan berpotensi mahal untuk menentukan penyakit secara akurat.
2. Proses diagnosis konvensional dapat terkendala oleh faktor biaya, waktu, dan ketersediaan fasilitas diagnostik canggih di semua daerah.

Berdasarkan identifikasi masalah tersebut, maka pertanyaan penelitian ini dirumuskan sebagai berikut:

1. Bagaimana merancang dan mengimplementasikan sebuah model klasifikasi *machine learning* yang dapat mendiagnosis penyakit pada kucing dengan tingkat akurasi yang dapat diterima secara ilmiah, berdasarkan dataset gejala klinis?
2. Seberapa efektif sistem yang dikembangkan dalam memberikan rekomendasi diagnosis yang akurat jika dibandingkan dengan kebutuhan waktu dan biaya pada proses diagnosis konvensional?

Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun sebuah sistem cerdas berbasis *machine learning* yang dapat memberikan rekomendasi diagnosis awal penyakit umum pada kucing secara cepat berdasarkan data gejala, sebagai alat bantu bagi pemilik hewan dan dokter hewan.

Ada beberapa penelitian sebelumnya yang dapat dirujuk. Misalnya, sebuah studi dari [6] menjelaskan bahwa KNN dapat mendiagnosis difteri dengan memeriksa data rekam medis 148 pasien, menghasilkan akurasi 95,17% dari 44 kasus data yang diuji. Ada juga studi untuk mendiagnosis penyakit pada kucing menggunakan sistem Case Based Reasoning (CBR) berbasis K Nearest Neighbor [7] dan juga studi yang menerapkan Case Based Reasoning untuk mendiagnosis penyakit dan hama pada tanaman mangga [8].

2. TINJAUAN PUSTAKA

2.1 Sistem Cerdas dan Sistem Pakar

Sistem cerdas, sering diwujudkan dalam bentuk sistem pakar, adalah program komputer yang dirancang untuk meniru kemampuan pengambilan keputusan seorang pakar dalam domain pengetahuan yang spesifik. Dalam konteks kedokteran hewan, sistem ini bertujuan untuk membantu pemilik hewan atau dokter hewan dalam mengidentifikasi penyakit berdasarkan gejala yang ada.

Sistem pakar yang baik harus memiliki ciri-ciri sebagai berikut[9]:

- a. Memiliki fasilitas informasi yang handal.
- b. Mudah dimodifikasi.
- c. Dapat digunakan dalam berbagai komputer.
- d. Memiliki kemampuan untuk belajar beradaptasi.

2.2 Algoritma KNN

Algoritma *K-Nearest Neighbor* (K-NN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. K-Nearest-Neighbours (K-NN) adalah metode klasifikasi non-parametrik yang sederhana namun efektif dalam banyak kasus[10]. Sebagaimana clustering, merupakan pengelompokkan suatu data baru dengan cara pengukuran jarak data baru tersebut dengan data-data lain atau dengan tetangga terdekat yang ada. Sebelum melakukan pengukuran jarak dengan tetangga, yang perlu dilakukan adalah menentukan nilai K tetangga (neighbor). Adapun dalam mengukur jarak dua titik yang berbeda, yaitu data training dengan data testing, maka digunakan jarak Euclidean dengan rumus sebagai berikut:

$$\text{dist}(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \dots\dots\dots(1)$$

di mana: d(x,y) adalah jarak antara data x ke data y
 x_i adalah data testing ke-i
 y_i adalah data training ke-i
 n = jumlah fitur

3. METODE PENELITIAN

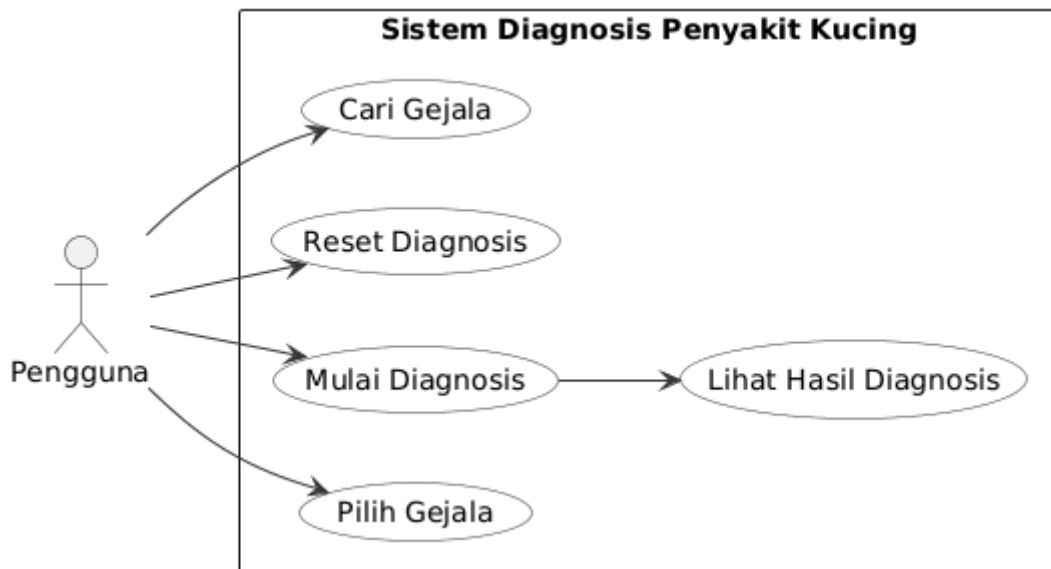
Untuk menjawab rumusan masalah dan mencapai tujuan penelitian secara sistematis, penelitian ini mengadopsi kerangka kerja Agile Scrum. Metodologi ini memecah keseluruhan proses menjadi siklus-siklus kerja pendek yang disebut Sprint, memungkinkan pengembangan yang adaptif dan evaluasi berkelanjutan terhadap model *machine learning* yang dibangun. Alur kerja penelitian akan mengikuti tahapan berikut:

1. Pembuatan *product backlog*: Tahap awal adalah menyusun daftar induk (*master list*) dari semua kebutuhan penelitian yang diurutkan berdasarkan prioritas. Item dalam backlog ini secara langsung diturunkan dari rumusan masalah, mencakup tugas-tugas seperti: pengumpulan data gejala klinis, implementasi dan perbandingan beberapa algoritma klasifikasi, serta perancangan purwarupa sistem.
2. *Sprint Planning*: Sebelum setiap siklus dimulai, diadakan sesi perencanaan untuk memilih serangkaian tugas dari Product Backlog. Tujuannya adalah untuk membentuk **Sprint Backlog** yang fokus pada pencapaian tujuan spesifik.
3. *Sprint*: Merupakan periode kerja dengan durasi tetap di mana peneliti fokus mengeksekusi tugas dari Sprint Backlog. Setiap Sprint dirancang untuk menghasilkan *Increment* atau hasil nyata, seperti dataset yang sudah bersih, model yang terlatih, atau fitur antarmuka yang fungsional.
4. *Daily Stand up*: Pertemuan harian singkat untuk melaporkan kemajuan, menyelaraskan pekerjaan selanjutnya, dan mengidentifikasi hambatan yang dapat memperlambat proses penelitian dalam menjawab rumusan masalah.
5. *Sprint Review*: Di akhir Sprint, hasil kerja (*Increment*) akan didemonstrasikan untuk dievaluasi dan mendapatkan umpan balik, khususnya dari dosen pembimbing atau pakar domain. Sesi ini krusial untuk memvalidasi apakah model yang dihasilkan menunjukkan potensi akurasi yang baik dan sesuai dengan tujuan.
6. *Sprint Retrospective*: Setelah Sprint Review, dilakukan evaluasi internal terhadap proses kerja. Tujuannya adalah untuk mengidentifikasi perbaikan yang dapat meningkatkan efisiensi dan kualitas hasil penelitian di Sprint berikutnya.
7. *Product Backlog Refinement*: Merupakan aktivitas berkelanjutan untuk memperbarui daftar induk tugas berdasarkan umpan balik dari Sprint Review dan temuan tak terduga dari hasil eksperimen. Hal ini memastikan penelitian tetap relevan dan fokus pada pengembangan solusi yang paling optimal.

4. PEMBAHASAN

4.1. Rancangan Aplikasi

Aplikasi sistem pakar dibuat berdasarkan penggambaran dalam bentuk diagram Use Case. Diagram yang dimaksud terdapat pada Gambar 1.

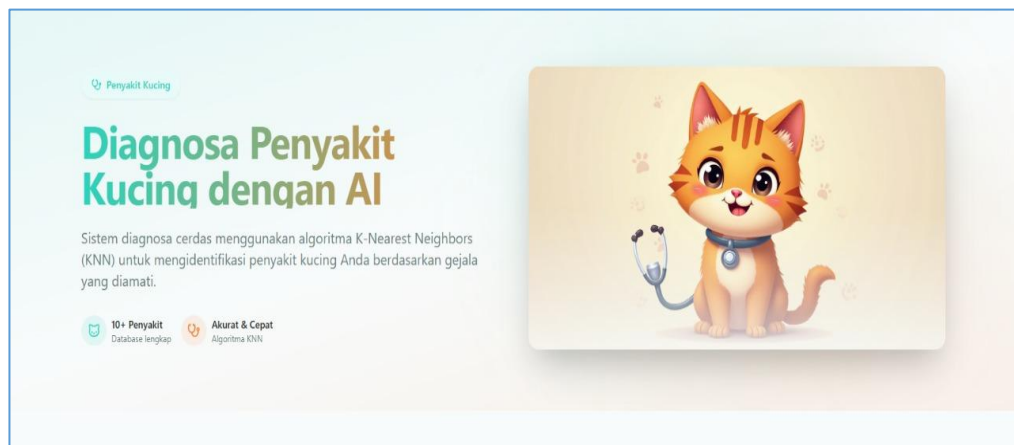


Gambar 1. Usecase Diagram

Diagram use case di atas menggambarkan hubungan antara pengguna dan sistem diagnosis penyakit kucing yang berfungsi untuk membantu mendeteksi penyakit berdasarkan gejala yang diamati. Dalam sistem ini, pengguna menjadi aktor utama yang berinteraksi langsung dengan antarmuka aplikasi untuk memberikan input berupa gejala yang terlihat pada kucing. Sistem kemudian memanfaatkan input tersebut untuk memproses dan memberikan hasil diagnosis secara otomatis.

Sebelum memulai proses diagnosis, pengguna dapat menggunakan fitur Cari Gejala untuk menemukan gejala tertentu dari daftar yang tersedia, kemudian melakukan pemilihan gejala yang sesuai dengan kondisi kucing. Setelah semua gejala dipilih, pengguna menekan tombol Mulai Diagnosis, yang akan memicu sistem menjalankan algoritma diagnosis untuk menentukan kemungkinan penyakit yang dialami. Hasil analisis tersebut kemudian dapat dilihat melalui fitur Lihat Hasil Diagnosis, yang menampilkan informasi penyakit beserta tingkat keyakinannya.

Selain itu, sistem juga menyediakan fitur Reset Diagnosis yang memungkinkan pengguna menghapus seluruh pilihan gejala maupun hasil diagnosis sebelumnya. Fitur ini memudahkan pengguna untuk memulai proses baru tanpa harus keluar dari sistem. Secara keseluruhan, diagram ini menunjukkan alur interaksi yang sederhana namun efektif antara pengguna dan sistem, di mana setiap fitur dirancang untuk mendukung proses diagnosis penyakit kucing secara cepat, akurat, dan mudah digunakan.



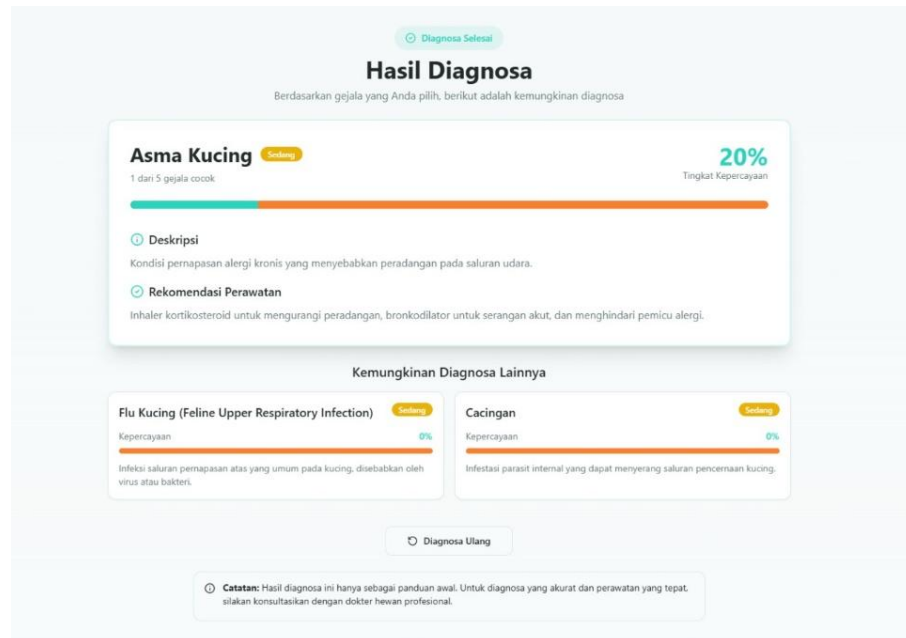
Gambar 2. Tampilan Awal

Pada gambar 2 ditampilkan halaman utama dari sistem AI Diagnosa Penyakit Kucing. Halaman ini berfungsi sebagai pengenalan sistem kepada pengguna. Di bagian kiri, terdapat judul besar “Diagnosa Penyakit Kucing dengan AI” yang menjelaskan tujuan utama aplikasi ini, yaitu membantu mendiagnosis penyakit kucing secara cerdas. Penjelasan di bawahnya menyebutkan bahwa sistem menggunakan algoritma K-Nearest Neighbors (KNN) untuk mengidentifikasi penyakit berdasarkan gejala yang diamati. Dua ikon informasi di bagian bawah memberikan gambaran mengenai keunggulan sistem, yakni mampu mengenali lebih dari 10 jenis penyakit dan memberikan hasil yang akurat serta cepat. Di sisi kanan, terdapat ilustrasi kucing mengenakan stetoskop yang memperkuat kesan bahwa aplikasi ini dirancang untuk mendukung pemilik kucing dalam mengenali kondisi kesehatan hewan peliharaannya. Tampilan ini memberikan kesan awal yang profesional, ramah, dan mudah dipahami oleh pengguna.

Gambar 3. Output

Gambar 3 menampilkan halaman pemilihan gejala, yaitu tahap di mana pengguna memasukkan data observasi terkait kondisi kucingnya. Pada bagian atas terdapat judul “Pilih Gejala yang Dialami” serta instruksi untuk mencentang semua gejala yang diamati. Di bawahnya terdapat kolom pencarian agar pengguna dapat menemukan gejala tertentu dengan mudah. Daftar gejala ditampilkan secara lengkap dan

rapi dalam bentuk pilihan, seperti bersin-bersin, demam, muntah, bulu kusam, lesu, kulit kemerahan, hingga diare parah. Pengguna dapat memilih satu atau beberapa gejala sesuai kondisi kucing. Di bagian bawah terdapat keterangan jumlah gejala yang telah dipilih serta tombol “Diagnosa Sekarang” untuk memulai proses analisis. Pesan pengingat di bawah tombol tersebut menegaskan bahwa pengguna harus memilih minimal satu gejala agar diagnosis dapat dilakukan. Tampilan ini merepresentasikan tahap input data yang sangat penting karena hasil diagnosis sistem nantinya bergantung pada gejala yang dipilih oleh pengguna.



Gambar 4. Hasil Diagnosa

Pada gambar 4 terlihat halaman hasil diagnosa, yang merupakan output utama dari sistem. Setelah pengguna memilih gejala dan menekan tombol diagnosa, sistem menampilkan hasil analisis menggunakan algoritma KNN. Di bagian atas terdapat tulisan “Hasil Diagnosa” dan status “Diagnosa Selesai” yang menunjukkan bahwa proses telah selesai. Hasil utama ditampilkan dalam bentuk kartu diagnosis, misalnya penyakit Asma Kucing, disertai dengan tingkat kepercayaan 20% yang menunjukkan seberapa besar kemungkinan penyakit tersebut sesuai dengan gejala yang dimasukkan. Di bawahnya terdapat bagian Deskripsi yang menjelaskan kondisi penyakit, serta Rekomendasi Perawatan yang memberikan saran tindakan awal, seperti penggunaan inhaler kortikosteroid untuk meredakan peradangan dan mencegah pemicu alergi. Selain hasil utama, sistem juga menampilkan kemungkinan diagnosa lain seperti *Flu Kucing* dan *Cacingan*, lengkap dengan tingkat kepercayaan masing-masing. Di bagian bawah, terdapat tombol “Diagnosa Ulang” yang memungkinkan pengguna melakukan analisis baru dengan gejala berbeda, serta catatan penting bahwa hasil diagnosis ini bersifat sementara dan tetap harus dikonfirmasi oleh dokter hewan profesional. Tampilan ini memberikan hasil yang informatif, terstruktur, dan mudah dipahami oleh pengguna.

Tabel 1. Tabel Sampel Data

No.	Nama Penyakit	Gejala	Deskripsi	Perawatan	Keparahan
1	Flu Kucing	bersin, mata_berair, hidung_meler, demam, nafsu_makan_turun	Infeksi saluran pernapasan atas yang umum pada kucing, biasanya disebabkan oleh virus herpes atau calicivirus.	Istirahat, perawatan suportif, antibiotik jika ada infeksi sekunder	medium
2	Cacingan	muntah, diare, perut_buncit, bulu_kusam, nafsu_makan_turun	Infestasi parasit internal yang dapat menyerang sistem pencernaan kucing.	Obat cacing sesuai jenis parasit, perawatan rutin	medium

3	Scabies (Kudis)	gatal_berlebihan, kulit_kemerahan, kerontokan_bulu, luka_garukan	Penyakit kulit yang disebabkan oleh tungau parasit yang hidup di lapisan kulit.	Obat anti-parasit topikal atau oral, mandi dengan antiseptik	medium
4	Panleukopenia	muntah, diare_parah, demam_tinggi, lesu, dehidrasi	Penyakit virus yang sangat menular dan berbahaya pada kucing, menyerang sistem kekebalan tubuh	Perawatan intensif di klinik, cairan IV, antibiotik, vaksinasi pencegahan	high
5	Infeksi Saluran Kemih	sering_buang_air_kecil, kesulitan_buang_air_kecil, darah_pada_urine, menjilat_area_genital	Infeksi atau peradangan pada saluran kemih yang menyebabkan rasa sakit dan sering buang air kecil	Antibiotik, peningkatan asupan air, diet khusus	high

Tabel 2. Tabel Tingkat Akurasi

No.	Nama Penyakit	Jumlah Data Uji	Prediksi Benar	Prediksi Salah	Akurasi (%)
1	Flu Kucing	50	46	4	92.00
2	Cacingan	40	37	3	92.50
3	Infeksi Jamur	35	32	3	91.43
4	Rabies	30	29	1	96.67
5	Diare	45	41	4	91.11
	Total/Rata-rata	200	185	15	92.50

Akurasi dihitung dengan rumus:

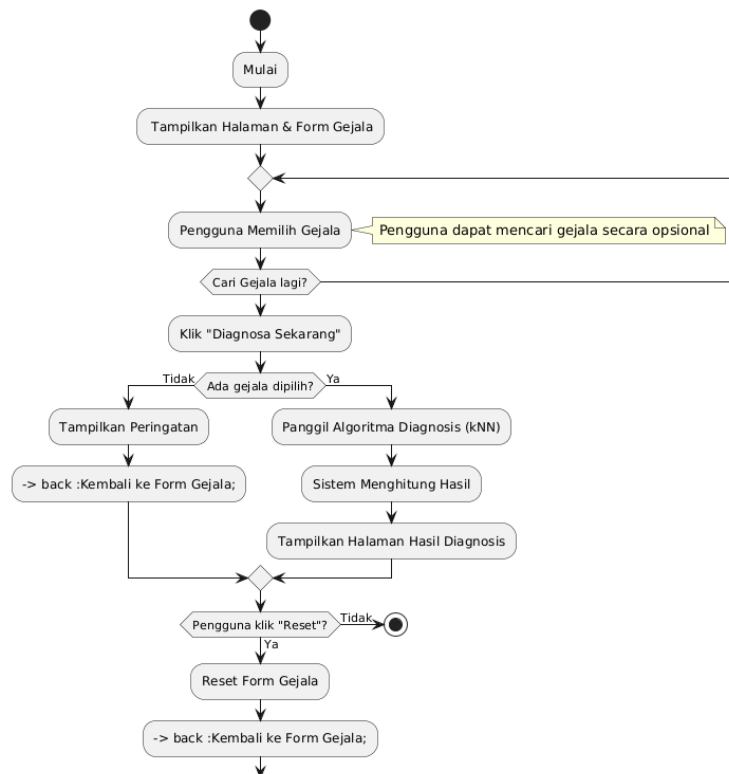
$$\text{Akurasi} = \frac{\text{Jumlah Prediksi Benar}}{\text{Jumlah Data Uji}} \times 100\%$$

Dari tabel pengujian:

$$\text{Akurasi} = \frac{185}{200} \times 100\% = 92.5\%$$

Hasil pengujian sistem diagnosis penyakit kucing berbasis kecerdasan buatan menunjukkan performa yang sangat baik dengan rata-rata akurasi sebesar 92,50% dari total 200 data uji, di mana 185 data berhasil diklasifikasikan dengan benar. Sistem mampu mengenali lima jenis penyakit utama — flu kucing, cacingan, infeksi jamur, rabies, dan diare — dengan tingkat akurasi yang tinggi, dan rabies menjadi penyakit dengan hasil terbaik sebesar 96,67%. Meskipun masih terdapat tingkat kesalahan sekitar 7,5% akibat kemiripan gejala antarpenyakit dan keterbatasan data pelatihan, performa sistem secara keseluruhan dinilai cukup andal. Untuk meningkatkan akurasi, disarankan penambahan variasi data latih serta optimasi algoritma. Dengan demikian, sistem ini layak digunakan sebagai alat bantu diagnosis awal, namun tetap perlu didukung oleh pemeriksaan langsung dari dokter hewan untuk memastikan hasil yang lebih akurat.

4.2. Perancangan Sistem



Gambar 5. Flowchart

Flowchart di atas menjelaskan alur proses pada sistem Diagnosis Penyakit Kucing yang dimulai dengan menampilkan halaman serta formulir gejala. Pengguna kemudian memilih gejala yang sesuai dengan kondisi kucing dan dapat mencari gejala tertentu secara opsional. Setelah selesai memilih, pengguna menekan tombol “Diagnosa Sekarang” untuk memulai proses diagnosis. Jika pengguna belum memilih gejala, sistem akan menampilkan peringatan dan mengarahkan kembali ke halaman form gejala.

Apabila gejala telah dipilih, sistem akan menjalankan algoritma KNN untuk menghitung dan menentukan hasil diagnosis berdasarkan data yang dimasukkan. Hasilnya kemudian ditampilkan pada halaman hasil diagnosis agar pengguna dapat mengetahui kemungkinan penyakit yang dialami kucing. Setelah melihat hasil, pengguna dapat menekan **reset** untuk menghapus data dan kembali ke form gejala, atau mengakhiri proses jika tidak ingin mengulang diagnosis.

5. KESIMPULAN

Penelitian ini telah berhasil merancang dan membangun sebuah sistem cerdas untuk diagnosis penyakit pada kucing menggunakan algoritma K-Nearest Neighbor (KNN). Berdasarkan hasil pengujian yang dilakukan terhadap 200 data uji yang mencakup lima penyakit berbeda, sistem menunjukkan performa yang sangat baik dengan mencapai tingkat akurasi rata-rata sebesar 92,50%. Secara spesifik, sistem berhasil mengklasifikasikan 185 data dengan benar, sementara 15 data salah terklasifikasi. Diagnosis Rabies menunjukkan akurasi tertinggi (96,67%) , sedangkan Diare mencatat akurasi terendah (91,11%). Meskipun masih terdapat tingkat kesalahan sekitar 7,5%, yang kemungkinan disebabkan oleh kemiripan gejala antar penyakit, sistem ini terbukti layak digunakan sebagai alat bantu diagnosis awal. Namun, penggunaannya harus tetap didukung oleh pemeriksaan langsung dari dokter hewan untuk konfirmasi yang akurat. Untuk pengembangan selanjutnya, disarankan agar dilakukan penambahan variasi data latih serta optimasi algoritma guna meningkatkan akurasi sistem, di masa depan program ini dapat dikembangkan menjadi aplikasi mobile untuk memudahkan user mengaksesnya.

DAFTAR PUSTAKA

- [1] Nurajizah, S., & Saputra, M. (2018). Sistem Pakar Berbasis Android Untuk Diagnosa Penyakit Kulit Kucing Dengan Metode Forward Chaining. *Jurnal Pilar Nusa Mandiri*, 14(1).
<https://ejournal.nusamandiri.ac.id/index.php/pilar/article/view/81>
- [2] Purnomo, Dwi, Beni Irawan, and Yulrio Brianorman. "Sistem Pakar Diagnosa Penyakit Pada Kucing Menggunakan Metode Dempster-Shafer Berbasis Android." *Coding: Jurnal Komputer dan Aplikasi* 5.1 (2017). <https://jurnal.untan.ac.id/index.php/jcskommipa/article/view/19177>
- [3] Saputra, Deby, Uning Lestari, and Edhy Sutanta. "Sistem Pakar Untuk Diagnosa Penyakit Kucing Berbasis Web Menggunakan Frameworkcodeigniter Web Based Expert System For Diagnosing Cat Disease Using Codeigniter Framework." *Jurnal SCRIPT* Vol 3.1 (2015).
https://www.academia.edu/download/45761192/2015_Deby_S-Uning_L-Edhy_S-Jurnal_SCRIPT-Sistem_Pakar_untuk_Diagnosa_Penyakit_Kucing.pdf
- [4] Little, Susan E. *The cat: clinical medicine and management*. Elsevier Health Sciences, 2011.
https://books.google.com/books?hl=id&lr=&id=rnyakraHGdcC&oi=fnd&pg=PP1&dq=The+cat:+clinical+medicine+and+management&ots=cjj5VPntlg&sig=fyshaT_pqeQAMN7qJ-icVJoKlOA
- [5] Norsworthy, G. D. (2018). *The feline patient*. John Wiley & Sons.
https://books.google.com/books?hl=id&lr=&id=1PRYDwAAQBAJ&oi=fnd&pg=PR1&dq=The+feline+patient&ots=HBiVuaTwaT&sig=9MmlMu8m1kQIFpKST7_P1hV71DE
- [6] Fatoni, Chavid Syukri, and Friandy Dwi Noviandha. "Case Based Reasoning Diagnosis Penyakit Difteri dengan Algoritma K-Nearest Neighbor." *Creative Information Technology Journal* 4.3 (2018): 220-232.
<http://citec.amikom.ac.id/main/index.php/citec/article/view/112>
- [7] Darmawan, Arief Soma, Devi Sugianti, and Muhammad Faisal Halim. "Case Based Reasoning Untuk Mendeteksi Penyakit Kucing Dengan Metode KNN." *Smart Comp: Jurnalnya Orang Pintar Komputer* 11.2 (2022): 109-115.
<http://ejournal.poltekharber.ac.id/index.php/smartcomp/article/view/3452>
- [8] Setiawan, Anton, and Setyawan Wibisono. "Case Based Reasoning Untuk Mendiagnosa Penyakit dan Hama Pada Tanaman Mangga Menggunakan Algoritma Similaritas Sorgenfrei." *Dinamik* 23.1 (2018): 1-10.
<http://download.garuda.kemdikbud.go.id/article.php?article=1307943&val=544&title=CASE%20BASED%20REASONING%20UNTUK%20MENDIAGNOSA%20PENYAKIT%20DAN%20HAMA%20PADA%20TANAMAN%20MANGGA%20MENGUNAKAN%20ALGORITMA%20SIMILARITAS%20SORGENFREI>
- [9] Azhar, Samsilul, Herlina Latipa Sari, and Leni Natalia Zulita. "Sistem pakar penyakit ginjal pada manusia menggunakan metode forward chaining." *Jurnal Media Infotama* 10.1 (2014).
<https://jurnal.unived.ac.id/index.php/jmi/article/view/228>
- [10] Hand, David J. "Principles of data mining." *Drug safety* 30.7 (2007): 621-622.
<https://link.springer.com/article/10.2165/00002018-200730070-00010>