



APLIKASI SISTEM PAKAR DIAGNOSA PENYAKIT PADA KUCING MENGGUNAKAN ALGORITMA CERTAINTY FACTOR BERBASIS WEB

(Expert System Application for Diagnosing Diseases in Cats Using a Web-Based Certainty Factor Algorithm)

**Dwipa Handayani¹, Asep Ramdhani Mahbub²,
Achmad Noeman³, Guruh Putra Febian⁴,
Hendarman Lubis^{5*}**

Program Studi Informatika¹, Program Studi Informatika², Program
Studi Informatika³, Program Studi Informatika⁴, Program Studi
Informatika⁵

Fakultas Ilmu Komputer¹, Fakultas Ilmu Komputer², Fakultas Ilmu
Komputer³, Fakultas Ilmu Komputer⁴, Fakultas Ilmu Komputer⁵

Universitas Bhayangkara Jakarta Raya¹²³

*Correspondent Author: hendarman.lubis@dsn.ubharajaya.ac.id

Authors Email: dwipa.handayani@dsn.ubharajaya.ac.id¹,
aseprm@dsn.ubharajaya.ac.id², achmad.noeman@dsn.ubharajaya.ac.id³,
202110715011@mhs.ubharajaya.ac.id⁴,
hendarman.lubis@dsn.ubharajaya.ac.id⁵

Received: August 25, 2025. **Revised:** September 30, 2025. **Accepted:**
October 01, 2025. **Issue Period:** Vol.9 No.2 (2025), Pp. 233-245

Abstrak: Sistem pakar diagnosa penyakit kucing berbasis web ini dirancang untuk memberikan hasil diagnosa secara cepat dan akurat. Pemilik kucing dapat memasukkan gejala yang diamati, kemudian sistem menganalisisnya menggunakan algoritma *Certainty Factor* untuk menghitung tingkat kepastian penyakit. Berdasarkan data pengujian, sistem ini mampu mencapai akurasi hingga 90% untuk beberapa penyakit umum seperti *Feline Panleukopenia Virus* dan *Feline Calicivirus*. *Certainty Factor* digunakan untuk menangani ketidakpastian data dengan menghasilkan nilai antara 0 hingga 1, yang menunjukkan tingkat keyakinan terhadap diagnosis. Sistem ini dibangun dengan *PHP* dan *MySQL* serta dapat diakses secara *real-time* melalui *web* responsif. Dengan adanya sistem ini, proses diagnosa menjadi lebih efisien dan dapat mengurangi ketergantungan terhadap dokter hewan untuk diagnosis awal. Selain itu, sistem ini juga berkontribusi pada pengembangan teknologi sistem pakar di bidang kesehatan hewan melalui penerapan algoritma *certainty factor* yang terbukti efektif.

Kata kunci: Sistem Pakar, Penyakit Kucing, Algoritma *Certainty Factor*, Diagnosa, *Web*.



DOI: 10.52362/jisicom.v9i2.2071

Ciptaan disebarluaskan di bawah [Lisensi Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



Abstract: *This web-based expert system for diagnosing cat diseases is designed to provide fast and accurate diagnoses. Cat owners can input observed symptoms, which are then analyzed using the Certainty Factor algorithm to calculate the likelihood of possible diseases. Based on test data, the system achieves up to 90% accuracy for common diseases such as Feline Panleukopenia Virus and Feline Calicivirus. The algorithm handles uncertainty by generating values between 0 and 1, indicating the confidence level of each diagnosis. Built with PHP and MySQL, the system is accessible in real-time through a responsive web interface. This enhances the efficiency of disease diagnosis and reduces the reliance on veterinarians for initial assessments. Moreover, the system contributes to the advancement of expert system technology in animal health by integrating the proven effectiveness of the Certainty Factor algorithm.*

Keywords: *Expert System, Cat Disease, Certainty Factor Algorithm, Diagnosis, Web*

I. PENDAHULUAN

Perkembangan teknologi informasi saat ini berkembang pesat, terutama dalam pengembangan aplikasi berbasis web yang responsif dan dapat diakses dengan mudah melalui berbagai perangkat. Salah satu teknologi yang banyak digunakan adalah PHP yang terintegrasi dengan MySQL, memungkinkan aplikasi web untuk mengelola interaksi dinamis dan menyimpan data dengan cepat. Penggunaan PHP dan MySQL pada sistem pakar berbasis web dapat memfasilitasi diagnosis penyakit hewan secara real-time, memberikan kemudahan akses bagi pengguna di berbagai perangkat. Kucing, sebagai salah satu hewan peliharaan yang paling populer di Indonesia, sering kali mengalami berbagai penyakit yang dapat terdiagnosis lebih awal melalui teknologi ini, yang membantu pemilik kucing melakukan diagnosa awal sebelum ke dokter hewan[1].

Ridho Petcare, sebagai klinik hewan di Bekasi, menjadi tempat penelitian ini karena banyaknya kasus penyakit kucing yang sulit didiagnosis dengan cepat oleh pemiliknya. Beberapa kendala yang dihadapi oleh pemilik kucing adalah kurangnya pengetahuan mengenai gejala penyakit dan terbatasnya akses ke fasilitas medis untuk hewan, terutama di daerah yang jauh dari klinik. Sistem pakar berbasis web yang menggunakan algoritma Certainty Factor diharapkan dapat membantu pemilik kucing mendiagnosis penyakit secara mandiri melalui gejala yang terlihat, mempercepat pengobatan[2], dan mengurangi ketergantungan pada klinik hewan. Penelitian ini bertujuan mengembangkan aplikasi sistem pakar yang responsif untuk membantu pemilik kucing melakukan diagnosis dini[3].

II. METODE PENELITIAN

Metode *Agile* merupakan pendekatan pengembangan perangkat lunak yang bersifat iteratif dan adaptif, di mana proses pengembangan dilakukan melalui tahapan berulang (*sprint*) yang memungkinkan penyesuaian secara fleksibel berdasarkan umpan balik pengguna[4]. Metode ini sangat sesuai untuk pengembangan sistem pakar diagnosis penyakit pada kucing berbasis web, karena dapat mengakomodasi perubahan kebutuhan pengguna dan memastikan kualitas sistem tetap terjaga selama proses pengembangan berlangsung[5].





Gambar 1. Metode Agile

1. *Plan* (Perencanaan)

Pada tahap ini, dilakukan identifikasi kebutuhan pengguna terhadap sistem pakar, termasuk fitur utama seperti input gejala dan hasil diagnosa. Kebutuhan disusun dalam *backlog*, diprioritaskan, dan direncanakan dalam *sprint-sprint* pendek. Tim dan *stakeholder* menetapkan tujuan iterasi serta tugas pengembangan.

2. *Design* (Perancangan)

Desain dilakukan di awal tiap *sprint* dengan fokus pada fitur yang akan dikembangkan, seperti antarmuka input gejala dan struktur basis pengetahuan. Desain juga mencakup skema *database MySQL* dan logika algoritma *Certainty Factor* untuk diagnosa.

3. *Develop* (Pengembangan)

Pengembangan dilakukan bertahap dalam setiap *sprint*, mulai dari membangun antarmuka web menggunakan *PHP*, menghubungkan *database*, hingga implementasi algoritma *Certainty Factor* untuk memproses gejala dan menghasilkan diagnosa.

4. *Test* (Pengujian)

Fitur diuji langsung dalam *sprint* yang sama, mulai dari *unit testing*, *integration testing*, hingga *user acceptance testing* untuk memastikan sistem berjalan sesuai harapan dan menghasilkan diagnosa yang akurat.

5. *Deploy* (Penerapan)

Setelah diuji, sistem di-*deploy* ke server uji atau lingkungan staging untuk diakses pengguna dan mendapatkan umpan balik awal, bukan hanya di akhir pengembangan.

6. *Review* (Tinjauan)

Setiap akhir *sprint*, dilakukan *review* terhadap hasil kerja tim, dengan umpan balik dari *stakeholder* dan pengguna mengenai akurasi diagnosa dan antarmuka sistem. Masukan digunakan untuk perbaikan pada *sprint* berikutnya.

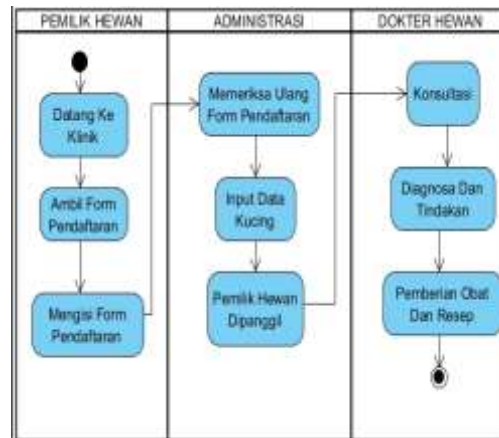
7. *Launch* (Peluncuran)

Setelah fitur utama selesai dan disetujui, sistem diluncurkan secara resmi ke produksi. Sistem pakar dapat diakses oleh pemilik kucing untuk diagnosa mandiri sebelum konsultasi ke dokter hewan, dengan dokumentasi dan pelatihan jika diperlukan.

III. PEMBAHASAN DAN HASIL

3.1. Analisis Sistem Berjalan

Proses diagnosis penyakit pada kucing saat ini masih dilakukan secara langsung di klinik hewan, di mana pemilik harus membawa kucingnya untuk diperiksa oleh dokter. Dokter mendiagnosis penyakit berdasarkan gejala dan pemeriksaan fisik. Proses ini memerlukan waktu, biaya, dan menyulitkan pemilik dengan keterbatasan akses atau waktu.

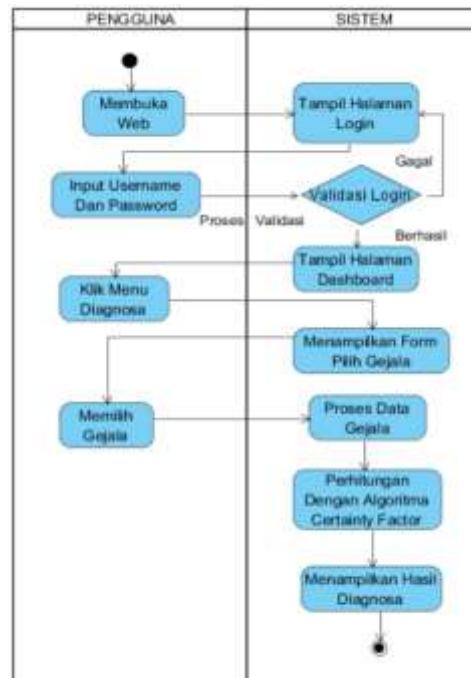


Gambar 2. Analisis Sistem Berjalan

Pemilik hewan datang ke klinik dan mengambil formulir pendaftaran untuk diisi. Setelah mengisi formulir, pemilik hewan menyerahkannya kepada administrasi. Administrasi kemudian memeriksa kelengkapan dan keakuratan formulir yang telah diisi. Setelah formulir diperiksa, administrasi memasukkan data tentang hewan peliharaan ke dalam sistem dan memanggil pemilik hewan untuk melanjutkan ke tahap berikutnya. Selanjutnya, dokter hewan melakukan konsultasi dengan pemilik hewan mengenai kondisi kesehatan hewan peliharaan, mendiagnosa kondisi hewan, dan menentukan tindakan atau perawatan yang diperlukan. Dokter hewan kemudian memberikan obat atau resep yang diperlukan untuk pengobatan hewan.

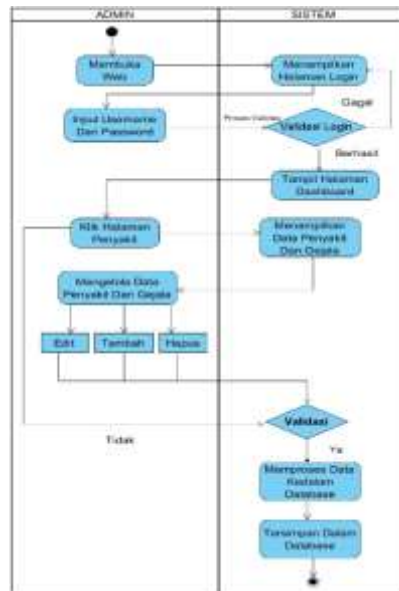
3.2. Analisis Sistem Usulan

Penulis mengusulkan solusi dengan mengembangkan aplikasi sistem pakar berbasis web untuk mendiagnosis penyakit pada kucing menggunakan algoritma *certainty factor*. Aplikasi ini memungkinkan pengguna untuk mengakses informasi tentang penyakit kucing dengan mudah. Berikut adalah ilustrasi alur sistem yang akan diimplementasikan dalam aplikasi diagnosa penyakit pada kucing, yang ditampilkan pada gambar di bawah ini:



Gambar 3. Analisis Sistem Usulan Pengguna

Deskripsi sistem usulan pengguna dimulai dengan pengguna membuka aplikasi *web*, yang kemudian menampilkan halaman *login* untuk memasukkan *username* dan *password*. Setelah pengguna memasukkan informasi *login*, sistem akan memvalidasi data yang dimasukkan; jika validasi berhasil, pengguna langsung diarahkan ke halaman *dashboard*, namun jika gagal, sistem akan kembali ke halaman *login* untuk mencoba input ulang. Setelah berhasil *login*, sistem menampilkan halaman *dashboard*. Pengguna kemudian mengklik menu "Diagnosa", yang akan menampilkan formulir untuk memilih gejala. Pengguna memilih gejala yang dialami, dan sistem akan memproses data gejala tersebut. Selanjutnya, sistem menghitung diagnosis menggunakan algoritma *Certainty Factor* untuk memberikan hasil diagnosa.



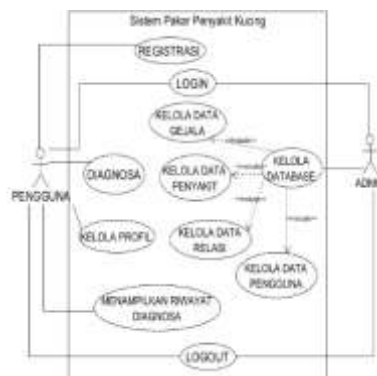
Gambar 4. Analisis Sistem Usulan Admin

Deskripsi sistem unggulan admin dimulai dengan admin membuka *web*, yang kemudian menampilkan halaman *login*. Admin menginput *username* dan *password* untuk login, dan sistem akan memvalidasi data tersebut. Setelah berhasil login, sistem menampilkan halaman *dashboard*. Admin kemudian mengakses halaman penyakit di dalam sistem, di mana sistem menampilkan data penyakit dan gejala yang sudah ada. Admin dapat memilih untuk mengedit, menambah, atau menghapus data penyakit dan gejala. Sistem akan melakukan validasi untuk memastikan bahwa data yang dimasukkan sesuai dan valid. Setelah proses validasi, sistem memproses data dan menyimpannya dalam *database*, yang kemudian dapat digunakan nanti.

3.3. Desain Sistem

Desain sistem untuk pengembangan sistem informasi ini menggunakan *Unified Modeling Language (UML)*[6]. Desain sistem penting untuk memahami alur dari sistem yang akan dibangun, mencakup desain antarmuka (tampilan) dan desain sistem secara keseluruhan. UML meliputi *use case diagram*, *activity diagram*, *class diagram*, serta *sequence diagram*.

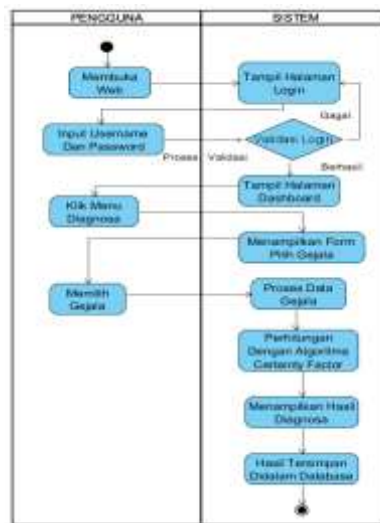
Use Case Diagram



Gambar 5. Use Case Diagram

Use Case Diagram ini melibatkan dua aktor, pengguna dan admin, dengan 10 *use case* yang menggambarkan interaksi mereka. Pengguna registrasi untuk membuat akun, lalu login untuk mengakses fitur diagnosis, kelola profil, dan riwayat diagnosa. Admin mengelola data *master*, termasuk gejala, penyakit, relasi gejala-penyakit, dan akun pengguna. Pengguna dapat memasukkan gejala kucing untuk diagnosis, memperbarui profil, dan melihat riwayat diagnosa, sementara admin dapat menambah, mengedit, atau menghapus data. Keduanya dapat *logout* dari sistem.

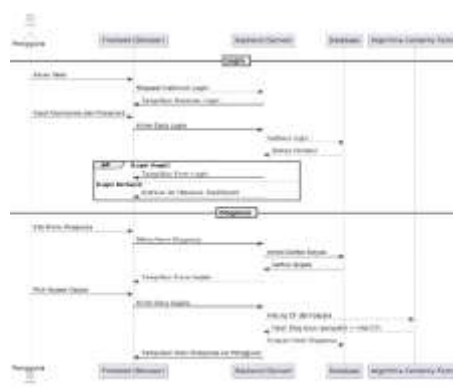
Activity Diagram



Gambar 6. Activity Diagram

Activity Diagram Diagnosa menggambarkan alur dari pengguna membuka web hingga hasil diagnosa tersimpan. Pengguna mengakses aplikasi, memasukkan *username* dan *password* untuk *login*. Setelah berhasil, pengguna mengklik menu "Diagnosa" dan memilih gejala. Sistem memproses data dan menghitung diagnosis menggunakan algoritma *Certainty Factor*. Hasil diagnosa ditampilkan dan disimpan dalam *database*. Proses selesai setelah hasil tersimpan[7].

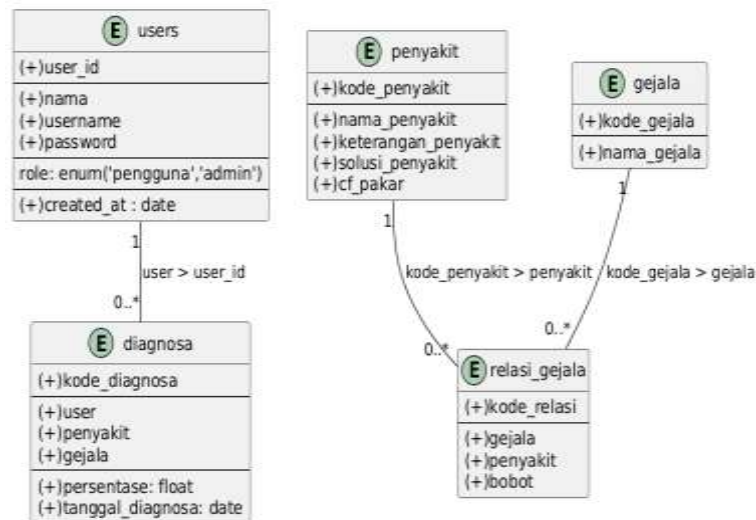
Sequence Diagram



Gambar 7. Sequence Diagram

Gambar *Sequence Diagram* Diagnosa menggambarkan interaksi antara Pengguna, *Frontend*, *Backend*, *Database*, dan Algoritma *Certainty Factor* dalam dua fase: *Login* dan *Diagnosa*. Pada fase *Login*, pengguna mengakses *web*, memasukkan *username* dan *password*, yang divalidasi oleh *Backend* dengan *Database*. Jika berhasil, *Backend* mengarahkan *Frontend* ke *Dashboard*. Pada fase *Diagnosa*, pengguna mengklik menu *Diagnosa*, memilih gejala, yang kemudian diproses oleh *Backend* menggunakan Algoritma *Certainty Factor*. Hasil diagnosa disimpan di *Database* dan ditampilkan oleh *Frontend*[8].

Class Diagram



Gambar 8. Class Diagram

Class diagram merupakan representasi visual dari struktur statis sistem, menunjukkan kelas-kelas, atribut-atributnya, operasi-operasinya, dan hubungan antar kelas. Dalam konteks sistem ini, *class diagram* digunakan untuk memodelkan entitas data dan interaksi logis antara mereka. Diagram ini mencerminkan struktur *database* yang akan dibangun[9].

3.4. Implementasi Algoritma *Certainty Factor*

Implementasi algoritma *Certainty Factor* pada aplikasi sistem pakar diagnosa penyakit pada kucing bertujuan untuk memberikan diagnosis yang akurat berdasarkan gejala yang dipilih pengguna. Algoritma ini menggabungkan bobot keyakinan setiap gejala yang dimasukkan, di mana nilai *CF* berkisar antara -1 hingga 1, yang menunjukkan tingkat dukungan gejala terhadap kemungkinan penyakit. Selain itu, algoritma ini juga mempertimbangkan bobot pakar atau tingkat keahlian dokter hewan untuk menghasilkan nilai kepercayaan akhir. Fungsi *calculate CF()* menghitung tingkat keyakinan terhadap setiap penyakit berdasarkan gejala dan bobot pakar yang relevan[10]. Hasilnya adalah daftar penyakit yang terurut berdasarkan tingkat kemungkinannya, membantu pengguna untuk mendapatkan gambaran awal kondisi kucing mereka. Dengan demikian, algoritma *certainty factor* memungkinkan aplikasi ini memberikan diagnosis berbasis data yang lebih akurat dan transparan, serta memberi kemudahan bagi pemilik kucing dalam mengambil keputusan terkait perawatan hewan peliharaan mereka.

3.5 Perhitungan Metode *Certainty Factor*

Untuk menentukan tingkat keyakinan terhadap diagnosa penyakit berdasarkan gejala yang muncul pada kucing, sistem ini menggunakan metode *Certainty Factor* (CF). CF merupakan metode representasi pengetahuan yang digunakan untuk menangani ketidakpastian dalam sistem pakar. Metode ini mengukur seberapa yakinnya seorang pakar terhadap hubungan antara suatu gejala dan suatu penyakit. *Certainty Factor* dihitung dengan rumus:

$$CF = W \times M \quad (1)$$

Keterangan :

W = Bobot kepentingan gejala terhadap penyakit (*Weight*)

M = Tingkat Kepastian (Measure of belief), yaitu seberapa besar kepercayaan user/pengguna bahwa gejala tersebut memang benar dialami

CF = Hasil nilai keyakinan yang digunakan dalam proses inferensi

Tabel berikut menggambarkan nilai CF dari masing-masing gejala (G01–G20) yang telah dihitung berdasarkan bobot dan tingkat kepastian yang diberikan. Nilai-nilai ini digunakan oleh sistem pakar untuk melakukan proses diagnosa dan menyimpulkan kemungkinan penyakit yang diderita oleh kucing.

Tabel 1. Perhitungan Metode *Certainty Factor*

Gejala	Penyakit	Bobot (W)	Tingkat Kepastian (M)	CF (<i>Certainty Factor</i>)
(G01) Muntah	(P01) <i>Feline Panleukopenia</i>	100	1	1
(G02) Diare	(P01) <i>Feline Panleukopenia</i>	80	0,8	0,64
(G03) Lemas	(P01) <i>Feline Panleukopenia</i>	60	0,6	0,36
(G04) Demam	(P01) <i>Feline Panleukopenia</i>	60	0,7	0,42
(G05) Tidak Napsu Makan	(P01) <i>Feline Panleukopenia</i>	10	0,5	0,05
(G06) Pucat	(P01) <i>Feline Panleukopenia</i>	10	0,3	0,03
(G07) Dehidrasi	(P01) <i>Feline Panleukopenia</i>	80	0,9	0,72
(G08) Terlalu Sering Buang Air Kecil	(P01) <i>Feline Panleukopenia</i>	10	0,2	0,02
(G09) Sulit Bernafas	(P02) <i>Feline Calicivirus</i>	50	0,8	0,4



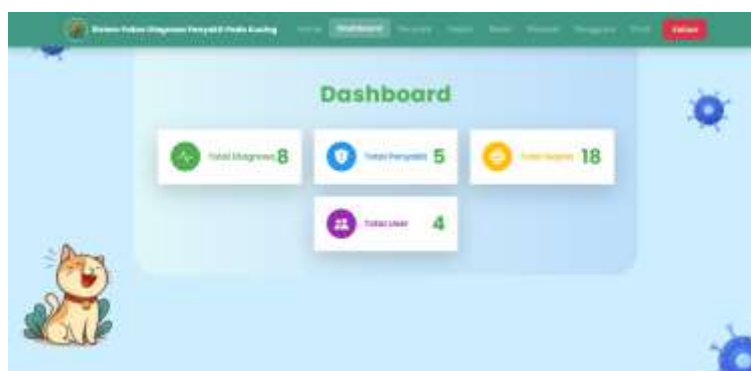
(G10) Radang Gusi	<i>(P02) Feline Calicivirus</i>	70	0,85	0,595
(G11) Perut Membesar	<i>(P02) Feline Calicivirus</i>	80	0,8	0,64
(G12) Mata Berair	<i>(P02) Feline Calicivirus</i>	80	0,9	0,72
(G13) Tidak Mau Minum	<i>(P03) Feline Infectious Peritonitis</i>	10	0,3	0,03
(G14) Luka-Luka	<i>(P03) Feline Infectious Peritonitis</i>	50	0,7	0,35
(G15) Batuk	<i>(P04) Feline Viral Rhinotracheitis</i>	60	0,75	0,45
(G16) Pilek	<i>(P04) Feline Viral Rhinotracheitis</i>	100	0,9	0,9
(G17) Radang Mata	<i>(P04) Feline Viral Rhinotracheitis</i>	80	0,8	0,64
(G18) Badan Kuning	<i>(P05) Feline Immunodeficiency Virus</i>	70	0,85	0,595

Sumber: Hasil Penelitian (2025)

3.6 Implementasi Tampilan Halaman Antarmuka

Tampilan Halaman Dashboard

Dashboard pengguna yang menampilkan berbagai menu utama, seperti diagnosa, riwayat, dan pengaturan profil, yang dapat diakses setelah login berhasil [11].



Gambar 9. Halaman Dashboard

Tampilan Halaman Diagnosa

Halaman di mana pengguna memilih gejala yang dialami oleh kucing mereka, yang kemudian diproses oleh sistem untuk memberikan diagnosis penyakit.



DOI: 10.52362/jisicom.v9i2.2071

Ciptaan disebarluaskan di bawah [Lisensi Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



9.	Admin menambah data relasi baru dengan semua <i>field</i> terisi valid.	Sistem berhasil menyimpan data relasi baru dan menampilkannya di daftar.	Sesuai	Berhasil
10.	Pengguna/Admin mengklik tombol "Keluar" / "Logout" dari halaman <i>dashboard</i> atau halaman lain yang sudah <i>login</i> .	Sistem berhasil mengeluarkan pengguna dari sesi saat ini dan mengarahkan kembali ke halaman <i>login</i> .	Sesuai	Berhasil

Sumber: Hasil Penelitian (2025)

IV. KESIMPULAN

Sistem pakar berbasis web yang dikembangkan dalam penelitian ini terbukti mampu memberikan hasil diagnosis yang cukup akurat dalam mengidentifikasi penyakit pada kucing berdasarkan input gejala dari pengguna, dengan tingkat akurasi mencapai 90% pada lima jenis penyakit utama yang diuji, yaitu *Feline Panleukopenia Virus*, *Feline Calicivirus*, *Feline Infectious Peritonitis*, *Feline Viral Rhinotracheitis*, dan *Feline Immunodeficiency Virus*. Sistem ini menunjukkan performa yang konsisten, dengan keluaran sistem yang sebagian besar sejalan dengan hasil diagnosis dari pakar. Selain akurat, sistem ini juga mendapat respons positif dari para pengguna karena memberikan kemudahan akses, kecepatan dalam menampilkan hasil, serta tampilan antarmuka yang sederhana dan mudah dipahami. Kemampuan sistem dalam membantu pemilik kucing untuk mengenali penyakit sejak dini, terutama bagi mereka yang berada di daerah dengan keterbatasan layanan dokter hewan, menjadikan aplikasi ini sebagai solusi efektif dan aplikatif. Pengembangan selanjutnya dapat diarahkan pada penambahan cakupan jenis penyakit, optimalisasi algoritma diagnosa, serta pengembangan aplikasi versi mobile untuk meningkatkan fleksibilitas akses pengguna.

REFERENSI

- [1] H. Syahputra and D. Monica Syafindy, "Sistem Pakar Diagnosa Penyakit Hepatitis Dengan Menggunakan Metode Certainty Factor," *J. Sains Inform. Terap.*, vol. 2, no. 1, pp. 45–50, 2022, doi: 10.62357/jsit.v2i1.186.
- [2] M. A. A. Fawwaz, K. N. Ramadhani, and F. Sthevani, "Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional NeuralNetwork (CNN)," *e-Proceeding Eng.*, vol. 8, no. 1, pp. 715–730, 2020.
- [3] A. Yudhana, "Diagnosa dan Observasi Terapi Infestasi Ektoparasit Notoedres cati Penyebab Penyakit Scabiosis Pada Kucing Peliharaan," *Media Kedokt. Hewan*, vol. 32, no. 2, p. 70, 2021, doi: 10.20473/mkh.v32i2.2021.70-78.
- [4] I. Larasati, A. N. Yusril, and P. Al Zukri, "Systematic Literature Review Analisis Metode Agile Dalam Pengembangan Aplikasi Mobile," *Sistemasi*, vol. 10, no. 2, p. 369, 2021, doi: 10.32520/stmsi.v10i2.1237.
- [5] A. Ariesta, Y. N. Dewi, F. A. Sariasih, and F. W. Fibriany, "Penerapan Metode Agile Dalam Pengembangan Application Programming Interface System Pada Pt Xyz," *J. CoreIT J. Has. Penelit. Ilmu Komput. dan Teknol. Inf.*, vol. 7, no. 1, p. 38, 2021, doi: 10.24014/coreit.v7i1.12635.
- [6] T. Arianti, A. Fa'izi, S. Adam, and M. Wulandari, "Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (Unified Modelling Language)," *J. Ilm. Komput. Tera[an dan Inf.*, vol. 1, no. 1, pp. 19–25, 2022, [Online]. Available: <https://journal.polita.ac.id/index.php/politati/article/view/110/88>



DOI: 10.52362/jisicom.v9i2.2071

Ciptaan disebarluaskan di bawah [Lisensi Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



e-ISSN : 2597-3673 (Online) , p-ISSN : 2579-5201 (Printed)

Vol.9 No.2 (December 2025)

Journal of Information System, Informatics and Computing

Website/URL: <http://journal.stmikjayakarta.ac.id/index.php/jisicom>

Email: jisicom@stmikjayakarta.ac.id , jisicom2017@gmail.com

- [7] V. Yoga and P. Ardhana, “Pemodelan Activity Diagram Untuk Perancangan Sistem Informasi Rekam Medis Pada Klinik,” *J. Kesehat. Qamarul Huda*, vol. 9, pp. 106–109, 2021.
- [8] S. Nabila, A. R. Putri, A. Hafizhah, F. H. Rahmah, and R. Muslikhah, “Pemodelan Diagram UML Pada Perancangan Sistem Aplikasi Konsultasi Hewan Peliharaan Berbasis Android (Studi Kasus: Alopel),” *J. Ilmu Komput. dan Bisnis*, vol. 12, no. 2, pp. 130–139, 2021, doi: 10.47927/jikb.v12i2.150.
- [9] S. Ramdany, “Penerapan UML Class Diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web,” *J. Ind. Eng. Syst.*, vol. 5, no. 1, 2024, doi: 10.31599/2e9afp31.
- [10] Y. S. Harahap, “Pendidikan Teknologi dalam Al-Qur’an,” *J. Pendidik. dan Konseling*, vol. 4, pp. 1349–1358, 2022.
- [11] F. D. Cahyo, S. D. Sancoko, and S. Diwandari, “Implementasi Layanan Payment Gateway Pada Aplikasi Klinik Hewan Peliharaan Berbasis Mobile,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 1, pp. 84–94, 2024, doi: 10.47233/jteksis.v6i1.1098.
- [12] J. Shadiq, A. Safei, and R. W. R. Loly, “Pengujian Aplikasi Peminjaman Kendaraan Operasional Kantor Menggunakan BlackBox Testing,” *Inf. Manag. Educ. Prof. J. Inf. Manag.*, vol. 5, no. 2, p. 97, 2021, doi: 10.51211/imbi.v5i2.1561.



DOI: 10.52362/jisicom.v9i2.2071

Ciptaan disebarluaskan di bawah [Lisensi Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).