

SISTEM PAKAR DIAGNOSA PENYAKIT HEWAN TERNAK SAPI MENGGUNAKAN METODE FORWARD CHAINING

Mildawati^{*1}, Laode Saidi², Ferdinand Murni Hamundu³

^{1,2,3}Universitas Halu Oleo

Email: ¹mmildawati487@gmail.com, ²Im.saidi@gmail.yahoo.co.id, ³ferdinand@uho.ac.id

^{*} Penulis Korespondensi

Abstrak

Sistem pakar bertujuan agar komputer mampu memberikan solusi cerdas seperti manusia. Salah satu algoritma yang digunakan adalah *forward chaining*. Berdasarkan wawancara dengan dokter hewan, jumlah dokter hewan di Kota Kendari masih sangat kurang, sehingga diagnosa penyakit ternak terkendala. Penelitian ini bertujuan menghasilkan sistem pakar untuk mendiagnosa penyakit sapi menggunakan metode *Forward Chaining*. Metode *forward chaining* merupakan metode runut maju, yang berarti menggunakan kondisi aksi. Metode ini akan menggunakan data dalam menentukan aturan mana yang akan dijalankan, setelahnya aturan tersebut akan dijalankan. *Forward chaining* dapat disebut sebagai metode intervensi yang dimana melakukan penalaran pada suatu masalah kepada solusinya. Metode ini sendiri menggunakan pendekatan *goal-driven*, diawali dengan hipotesis yang dilanjutkan dengan mencari bukti pendukung dari hipotesis tersebut. Hasil uji *blackbox* menunjukkan seluruh fungsi sistem berjalan baik. Uji *confusion matrix* menunjukkan akurasi 87.5%, presisi 100%, dan *recall* 87.5%. Nilai akurasi dari yang kurang begitu baik dari sistem disebabkan oleh faktor keterbatasan intervensi yang dilakukan oleh sistem. Meskipun begitu sistem telah mampu digunakan secara luas dan diharapkan dapat membantu dalam proses diagnosa penyakit pada hewan ternak sapi.

Kata Kunci: Ternak Sapi Potong, Sistem Pakar, Forward Chaining, Diagnosa Penyakit Ternak, Confussion Matrix

Abstract

Expert Systems aim for computers to be able to provide intelligent solutions like a human. One of the algorithms used is forward chaining. According to interviews with a veterinarian, the number of veterinarians in Kendari City is still very limited, so the diagnosis of animal diseases is constrained. The research aims to produce an expert System to diagnose cattle diseases using the Forward Chaining method. The forward chaining method is a forward chaining method, which means using action conditions. This method will use data in determining which rules to run, after which the rules will be executed. Forward chaining can be referred to as an intervention method which reasoning on a problem to its solution. This method itself uses a goal-driven approach, starting with a hypothesis followed by finding supporting evidence for the hypothesis. Blackbox test results show that all system functions run well. The confusion matrix test shows 87.5% accuracy, 100% precision, and 87.5% recall. The accuracy value of the system is not so good due to the limitation factor of the interventions carried out by the system. Even so, the system has been able to be widely used and is expected to help in the process of diagnosing diseases in cattle animals.

Keywords: Beef livestock, Expert system, Forward Chaining, Diagnosa of livestock diseases, Confussion Matrix

1. PENDAHULUAN

Subsektor peternakan menjadi salah satu subsektor yang memberikan kontribusi terhadap tinggi terhadap perekonomian Indonesia serta dapat menyerap tenaga kerja secara signifikan, hal ini dapat diharapkan menjadi upaya perbaikan perekonomian negara. Berdasarkan hasil Sensus Pertanian 2023 (ST2013) bahwa rumah tangga peternakan yang ada di Indonesia mencapai 13,56 Juta rumah tangga. Disisi lain kesediaan produk peternakan secara langsung

dapat meningkatkan status gizi masyarakat, khususnya pemenuhan kalori serta protein hewani. Pemenuhan konsumsi kalori dan protein hewani dapat meningkatkan kualitas sumber daya manusia [1].

Kesehatan hewan ternak mengacu pada kondisi dimana seluruh sel dan cairan tubuh penyusun tubuh hewan berada dalam kondisi fisiologis yang normal. Salah satu aspek penting dalam perawatan kesehatan hewan adalah pengujian penyakit yang dicurigai dan melakukan pemantauan hewan yang sakit. Pengujian dugaan penyakit pada hewan ternak merupakan

proses mendeteksi serta mengamati perubahan yang terjadi pada ternak melalui tanda dan gejala yang terlihat sehingga dapat ditarik kesimpulan dan dapat ditentukan penyebab dari penyakitnya [2].

Penggunaan komputer dalam memproses data membuat pengambilan keputusan modern menjadi lebih cepat, efisien, serta akurat. Teknologi kecerdasan buatan merupakan teknologi yang terkenal dalam pemrosesan komputer. Sistem pakar merupakan sejenis kecerdasan buatan yang menggunakan data, serta strategi penalaran yang sering digunakan oleh manusia dalam menemukan jawaban atas pertanyaan yang memerlukan campur tangan manusia [3].

Secara umum, sistem pakar merupakan bidang ilmu komputer yang menggunakan komputer dengan tujuan bertindak secara cerdas seperti manusia. Sistem ini bertujuan untuk memberikan Solusi yang tepat terhadap masalah yang ada. Penggunaan aplikasi berbasis komputer dalam pemecahan masalah yang melibatkan pemikiran serta keahlian pakar. Nantinya sistem ini akan membantu penyelesaian masalah yang tidak dapat diselesaikan orang biasa, melainkan hanya dapat diselesaikan oleh ahli dari bidang tertentu. Suatu sistem pakar dapat dikatakan berhasil dalam mengambil keputusan jika keputusan yang dihasilkan konsisten dengan keputusan pakarnya.

Forward chaining adalah proses peruntutan yang dimulai dengan menampilkan data atau fakta yang nantinya akan menjadi kesimpulan diakhir. Pada pendekatan ini pelacakan dimulai dengan informasi yang dimasukan yang kemudian akan mencoba memberikan kesimpulan. Sehingga metode ini dapat disebut juga dengan “*data driven*” yang dimulai dengan informasi yang dimasukan terlebih dahulu kemudian menjadi kesimpulan [4]. Penggunaan metode *forward chaining* biasanya digunakan untuk mengatasi masalah dengan memutuskan realita terlebih dahulu, yang kemudian diakhiri dengan jawaban atas masalah tersebut [5]. Penerapan metode ini dapat digunakan dalam proses diagnosa penyakit ternak sapi misalnya dengan memberikan data gejala dari hewan ternak yang kemudian data gejala inilah yang akan diproses untuk mendiagnosa penyakit yang dialami dari hewan ternak tersebut.

Penelitian yang dilakukan oleh Putra, dkk, (2021) dengan judul “Analisis Metode *Forward Chaining* Pada Sistem Pakar Diagnosa Penyakit Hewan Sapi Berbasis Android” bertujuan untuk mengembangkan aplikasi diagnosa penyakit sapi

dengan tujuan memudahkan peternak dalam diagnose awal, identifikasi gejala, dan solusi. Hasilnya adalah sebuah sistem pakar yang dapat diakses melalui *smartphone*, memberikan hasil diagnosa, penyebab, serta solusi dari penyakit sapi.

Berdasarkan hasil wawancara dengan salah satu dokter hewan menjelaskan bahwa dokter hewan khususnya yang ada di Kota Kendari masih kurang sama sekali, sehingga proses diagnosa penyakit hewan ternak mengalami kendala dan kesulitan. Selain itu masih banyak juga pemilik sapi yang belum memiliki pemahaman mengenai penyakit sapi dan juga belum mampu untuk melakukan diagnosa mandiri terhadap hewan ternak milik mereka. Dalam jurnalnya Noorhaliza, dkk, (2022) menjelaskan di Kota Kendari sendiri memiliki fasilitas kesehatan khususnya kesehatan hewan. Namun hanya tersedia dokter hewan yang melakukan praktik mandiri, namun jumlahnya masih terbatas kurang dari 15 praktik yang terbuka sehingga penempatannya masih terpisah – pisah. Jumlah pusat perawatan atau pengobatan hewan yang tidak terhubung atau tidak berada di satu lokasi yang sama masih kurang dari 25. Sebagai upaya dalam penanganan masalah ini maka penulis melakukan penelitian denganjud

“Sistem Pakar Diagnosa Penyakit Hewan Ternak Sapi Menggunakan Metode *Forward Chaining* Di Dinas Peternakan Kota Kendari” dengan harapan sistem yang dikembangkan mampu membantu proses diagnosa penyakit pada hewan ternak secara tepat dan akurat.

Adapun tujuan dari penelitian ini adalah untuk menghasilkan sistem pakar diagnosa penyakit hewan ternak sapi menggunakan metode *Forward Chaining*.

2. METODE

2.1. Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan mulai bulan Agustus 2024 sampai dengan Oktober 2024. Penelitian ini dilakukan di Dinas Peternakan Kota Kendari.

2.2. Prosedur Penelitian

Prosedur penelitian pada dasarnya adalah kerangka hubungan antara konsep – konsep yang ingin diamati atau diukur melalui penelitian yang akan dilakukan. Prosedur penelitian yang digunakan dalam penelitian ini dapat disajikan pada Gambar 1.



Gambar 1. Prosedur Penelitian

2.3. Metode Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data-data di Dinas Peternakan Kota Kendari. Semua tahap pada proses pengumpulan data-data tersebut diperoleh dari studi Pustaka dan hasil wawancara, dimana dalam penelitian ini data itu meliputi data penyakit dan data gejala penyakit sapi

2.3.1. Studi Pustaka

Studi pustaka merupakan langkah awal dalam proses pengumpulan data dengan cara mencari informasi yang relevan dengan topik penelitian melalui dokumen tertulis. Dokumen yang dimaksud dapat berupa buku, jurnal, situs internet, dan media informasi lainnya. Informasi yang diperoleh bisa digunakan sebagai acuan atau referensi selama mengerjakan penelitian.

2.3.2. Wawancara

Wawancara dilakukan dengan melakukan tanya jawab dengan pihak terkait untuk mendapatkan informasi yang dibutuhkan mengenai permasalahan dan hal-hal yang dibutuhkan dalam proses pembuatan dan pengembangan sistem. Setelah melakukan wawancara dengan pakar dari Fakultas Peternakan Universitas Halu Oleo, didapatkan 15 jenis penyakit beserta 53 Gejalanya yang diperoleh dari beberapa sumber literatur. Data kode penyakit yang dapat disajikan pada Tabel 1

Tabel 1 Kode dan Jenis Penyakit

Kode Penyakit	Nama Penyakit
P01	Bovine Ephemeral fever
P02	Bovine Viral Diare
P03	Salmonellosis
P04	Infectious Bovine Rhinotracheitis
P05	Pink eye
P06	Tympany
P07	Scabies

Kode Penyakit	Nama Penyakit
P08	Septichaemia Efizootica
P09	Mastitis
P10	Helminthiasis
P11	Artritis
P12	Myasis
P13	Distokia
P14	Infestasi Kutu
P15	Cystitis

Tabel penyakit menyajikan daftar kode penyakit beserta nama penyakit yang dikumpulkan dari berbagai sumber referensi, termasuk artikel ilmiah, buku, dan jurnal, serta melalui diskusi dengan pakar di bidang kedokteran hewan. Setiap penyakit memiliki kode unik yang memudahkan identifikasi dan pengelompokan dalam sistem pakar. Data ini bertujuan untuk membantu proses diagnosa penyakit pada hewan ternak sapi dengan metode *forward chaining*.

Adapun data kode gejala pada hewan dapat disajikan pada Tabel 2

Tabel 2 Kode dan Jenis Gejala

Kode Gejala	Gejala
G01	Demam
G02	Adanya Leleran Hidung
G03	Kehilangan Nafsu Makan
G04	Sesak Nafas
G05	Gerakan Rumen Menurun
G06	Terjadi Konjungtivitas
G07	Air Mata dan Hidung Selalu Keluar Leleran Air
G08	Diare
G09	Keguguran
G10	Susu Berhenti
G11	Radang Hidung
G12	Lesu
G13	Mata Lembab
G14	Adanya Sedikit Konstriksi pada Pupil
G15	Photophobi
G16	Kembung
G17	Bulu Kusam dan Kasar
G18	Glambir Membengkak
G19	Mengalami Kesulitan dalam lahiran
G20	Berat Badan Menurun dan kurus
G21	Produksi Air Mata Meningkat
G22	Gatal
G23	Terdapat Darah pada Urin

G24	Leleran di Hidung
G25	Ngorok
G26	Air Susu Encer Bercampur nanah
G27	Kelenjar Air susu membengkak
G28	Kurus
G29	Anorexia
G30	Kaki Pincang
G31	Luka Berdarah
G32	Muncul Belatung pada luka
G33	Radang Mata
G34	Keluarnya Air Ketuban Tanpa Adanya Janin
G35	Konstraksi Tidak Efektif
G36	Kehilangan Bulu
G37	Urin yang Dikeluarkan Selalu Sedikit
G38	Mengalami Iritasi pada Kulit
G39	Janin Terlihat Tapi Tidak Bisa Keluar
G40	Infeksi Sekunder
G41	Bulu Rontok
G42	Kesulitan Bergerak
G43	Air Liur Berlebihan
G44	Kesulitan Berjalan atau Pincang

Kode Gejala	Gejala
G45	Penurunan Produksi Susu
G46	Kulit Merah dan Meradang
G47	Pembengkak pada Sendi
G48	Hidung dan Daerah Wajah Membengkak
G49	Membran Mukosa Pucat atau Kebiruan
G50	Batuk
G51	Kulit Berkerak
G52	Mata Berair
G53	Diare berdarah

Adapun untuk menjadi acuan dari penelitian ini diperlukan sebuah aturan atau *rule* yang digunakan dalam proses diagnosa penyakit. Adapun aturannya sebagai berikut

1. Kaidah 1: *IF G01 AND G04 AND G09 AND G11 AND G14 AND G18 AND G20 AND G21 AND G43 THEN P01*
2. Kaidah 2 : *IF G05 AND G06 AND G07 AND G08 AND G02 AND G42 AND G53 THEN P02*
3. Kaidah 3 : *IF G01 AND G09 AND G08 AND G10 THEN P03*
4. Kaidah 4 : *IF G11 AND G01 AND G09 AND G12 AND G04 AND G48 AND G50 THEN P04*
5. Kaidah 5 : *IF G13 AND G14 AND G15 AND*

G33 AND G52 THEN P05

6. Kaidah 6 : *IF G12 AND G17 AND G02 AND G16 THEN P06*
7. Kaidah 7 : *IF G20 AND G22 AND G41 AND G46 AND G51 THEN P07*
8. Kaidah 8 : *IF G01 AND G24 AND G25 AND G20 AND G04 AND G49 THEN P08*
9. Kaidah 9 : *IF G26 AND G45 AND G27 THEN P09*
10. Kaidah 10 : *IF G08 AND G17 AND G28 AND G29 THEN P10*
11. Kaidah 11 : *IF G12 AND G30 AND G44 AND G47 THEN P11*
12. Kaidah 12 : *IF G31 AND G32 THEN P12*
13. Kaidah 13 : *IF G19 AND G34 AND G35 AND G39 THEN P13*
14. Kaidah 14 : *IF G22 AND G36 AND G38 AND G40 THEN P14*
15. Kaidah 15 : *IF G23 AND G03 AND G01 AND G37 THEN P15*

2.4. Perancangan Sistem

Perancangan sistem adalah merancang atau mendesain suatu sistem yang baik, yang isinya adalah langkah-langkah operasi dalam proses pengelolaan data dan proses prosedur-prosedur untuk mendukung operasi sistem. Proses perancangan sistem ini dilakukan dengan proses pemodelan sistem menggunakan UML dan *Entity*

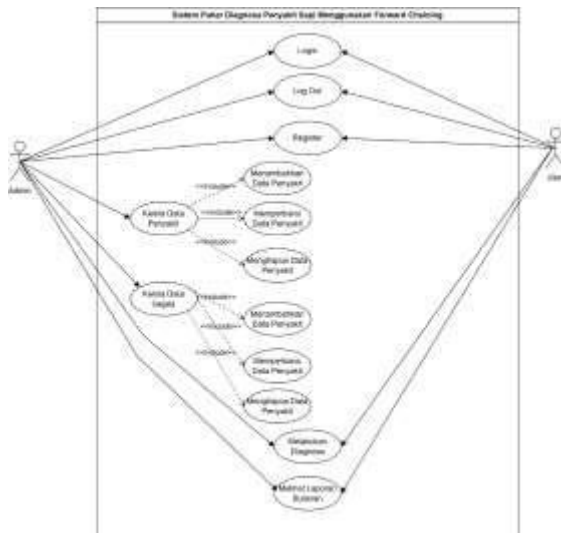
relational diagram dengan tujuan memberikan gambaran rancangan sistem yang dibuat.

2.4.1. Pemodelan Sistem

Pada tahap proses pemodelan sistem yang akan dibangun menggunakan UML yang dirancang menggunakan Use Case Diagram dan *Activity Diagram*. Use Case Diagram digunakan untuk menggambarkan interaksi antara sistem dengan pengguna atau aktor lainnya, sedangkan *Activity Diagram* digunakan untuk menggambarkan alur kerja atau proses dalam sistem tersebut. Dengan kedua diagram ini, akan memudahkan pemahaman terhadap fungsi dan proses yang terjadi dalam sistem yang akan dibangun.

1. Use Case Diagram

Use case diagram merupakan diagram yang mempresentasi interaksi yang dilakukan oleh aktor terhadap sistem dan penjelasan setiap fungsi yang tersedia untuk setiap aktor. Pada penjelasannya menggambarkan mengenai apa saja yang dapat dilakukan aktor terhadap sistem, dalam hal ini *admin* dan *user*, *use case diagram* pada penelitian ini dapat disajikan pada Gambar 2



Gambar 2 Use Case Diagram

Dalam sistem yang dikembangkan terdapat satu aktor yang nantinya akan berinteraksi dengan sistem. interaksi yang akan dilakukan oleh admin dalam sistem ini meliputi proses mengelola data penyakit, mengelola data gejala, melihat laporan tentang penggunaan sistem, statistik diagnosa, atau informasi lainnya yang relevan, serta melakukan diagnosa pada hewan ternak.

2. Activity Diagram

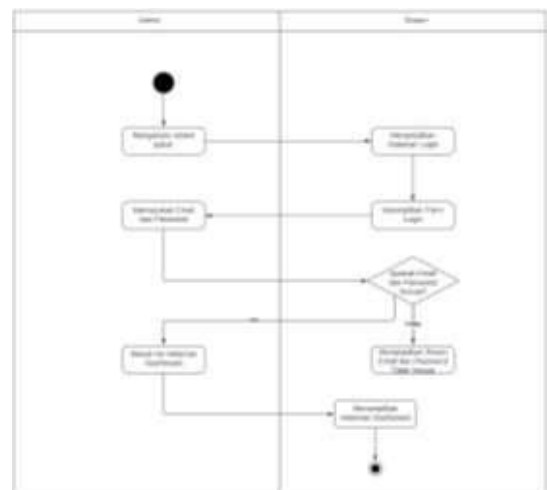
Pada sistem ini terdapat beberapa *activity diagram* yang digunakan untuk menggambarkan alur kerja dari suatu proses atau sistem. *Activity diagram* membantu dalam memahami interaksi antara aktivitas-aktivitas yang terjadi dalam sistem

tersebut. *Activity diagram* juga dapat digunakan untuk mengidentifikasi proses-proses yang memerlukan perbaikan atau optimisasi. Dengan menggunakan *activity diagram*, dapat ditemukan potensi bottleneck atau redundansi dalam alur kerja suatu sistem.

Dengan demikian, perbaikan dapat dilakukan untuk meningkatkan efisiensi dan produktivitas sistem. Selain itu, *activity diagram* juga memudahkan dalam komunikasi antara tim pengembang dan pemangku kepentingan terkait proses yang sedang dianalisis. Adapun *activity diagram* pada sistem ini dijelaskan sebagai berikut.

a. Activity Diagram Login

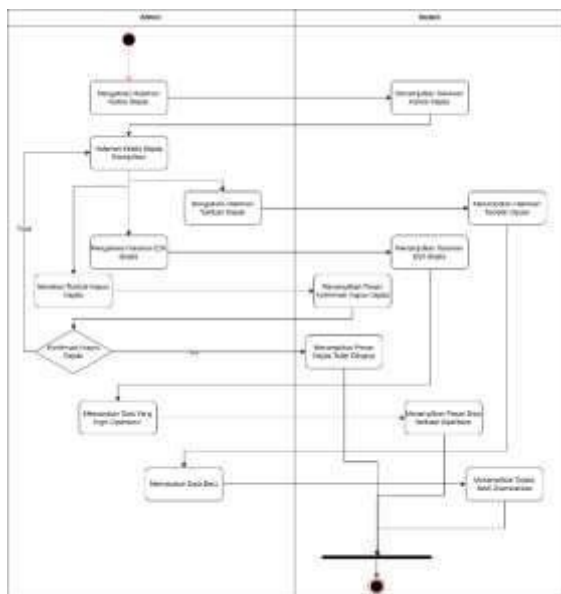
Pada Struktur *Login Page*, Admin akan memasukkan *username* dan *password* yang telah diinput penulis pada *database* sistem. Setelah berhasil *login* sistem akan menampilkan menu utama *website* tapi jika *username* dan *password* salah maka akan Kembali ke menu halaman *login page*. *Activity Diagram Login Page* ditunjukkan pada Gambar 3



Gambar 3 Activity Diagram Login

b. Activity Diagram Kelola Gejala

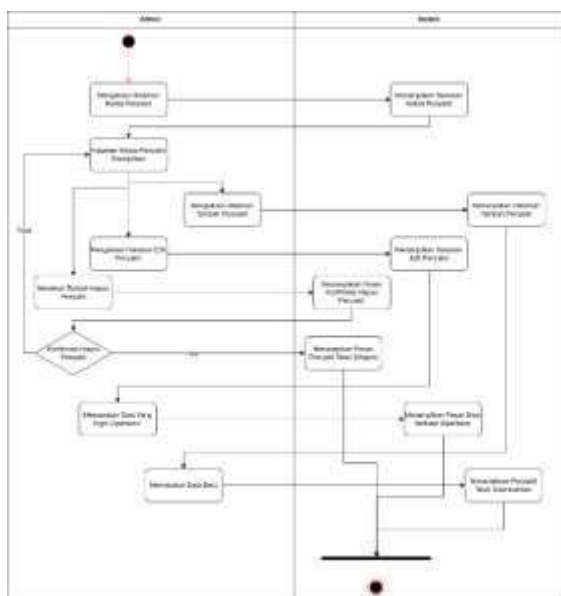
Pada Struktur *Kelola Gejala*, Admin akan menginput gejala - gejala pada *database* sistem. Setelah berhasil *meninput* maka akan muncul tabel data gejala. *Activity Diagram Kelola Gejala* pada Gambar 4



Gambar 4 Activity Diagram Kelola Gejala

c. Activity Diagram Kelola Penyakit

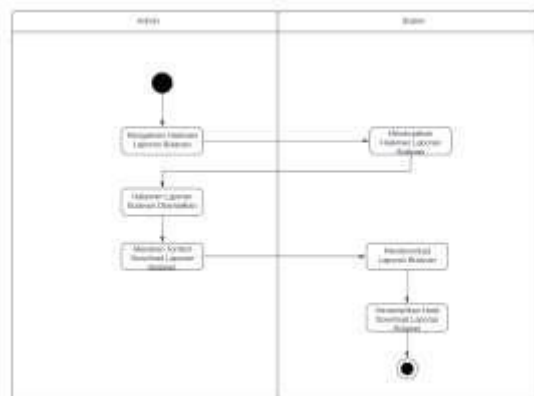
Pada Struktur Kelola Penyakit, *Admin* akan menginput penyakit beserta gejala pada *database* sistem. Setelah berhasil *menginput* maka akan muncul tabel data penyakit beserta gejala. *Activity Diagram* Kelola Penyakit ditunjukkan pada Gambar 5



Gambar 5 Activity Diagram Kelola Penyakit

d. Activity Diagram Lihat Laporan

Pada Struktur Lihat Laporan, *Admin* akan masuk pada menu Lihat Laporan. Setelah itu *Admin* mendownload laporan bulanan. *Activity Diagram* Lihat Laporan ditunjukkan pada Gambar 6

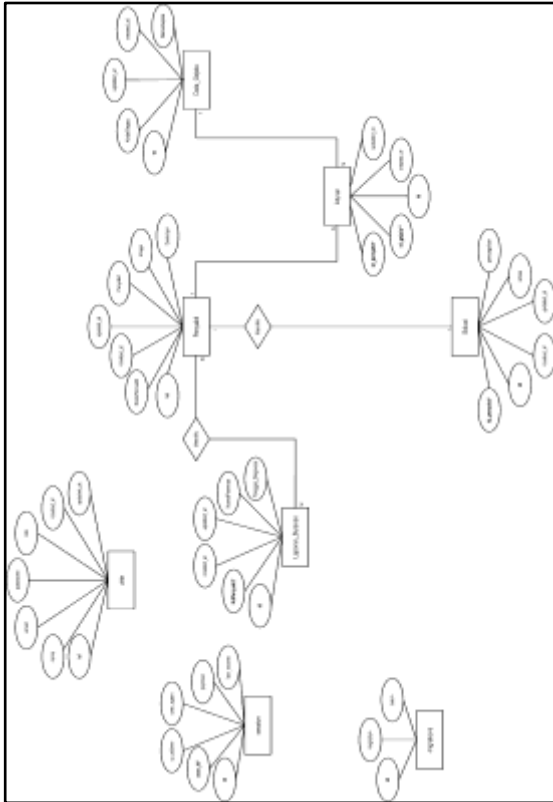


Gambar 6 Activity Diagram Lihat Laporan

3. Entity Relational Diagram

Entity Relationship Diagram (ERD) untuk sistem ini terdiri dari 8 tabel utama. Tabel pertama adalah tabel Penyakit yang terdiri dari atribut id, KodePenyakit, Penyakit, *image*, dan Deskripsi. Selanjutnya, tabel Data_Gejala yang terdiri atas atribut id, KodeGejala, dan NamaGejala. Tabel berikutnya adalah tabel Solusi, yang berisi id, id_penyakit (merujuk ke tabel Penyakit), solusi, dan pencegahan. Berikutnya adalah tabel Aturan, yang terdiri atas atribut id, id_penyakit (relasi dengan Penyakit), dan id_gejala (relasi dengan Data_Gejala). Tabel lain yang terdapat pada sistem adalah Laporan_Bulanan, yang berisi id, KdPenyakit (kode penyakit), Tanggal_Diagnosa, dan NamaPeternak (nama peternak).

Tambahan entitas lainnya adalah *User*, yang terdiri dari atribut id, email, name, password, dan role. Ada juga tabel *Session*, yang menyimpan data sesi pengguna dengan atribut id, user_id (relasi ke tabel User), ip_address, user agent, payload, dan last activity. Terakhir, tabel Migrations yang digunakan untuk mencatat informasi migrasi sistem, dengan atribut id, migration, dan batch. Diagram ini menggambarkan bagaimana hubungan antar entitas dapat dikelola dalam sistem untuk pengelolaan data yang efisien. ERD dari sistem telah disajikan pada Gambar 7



Gambar 7 Entity Relational Diagram

3. HASIL DAN PEMBAHASAN

3.1. Implementasi

Implementasi sistem mencakup pengembangan interface sistem serta penerapan metode forward chaining untuk menghasilkan diagnosa penyakit hewan ternak sapi. Metode forward chaining digunakan untuk melakukan pengecekan gejala-gejala yang muncul pada hewan ternak sapi dan menghasilkan diagnosa penyakit yang sesuai berdasarkan aturan yang telah ditentukan.

3.1.1. Halaman Diagnosa Penyakit

Halaman diagnosa penyakit digunakan oleh admin untuk melakukan proses diagnosa dari penyakit ternak sapi berdasarkan gejala yang ada pada hewan ternak tersebut. Pada proses ini admin akan memasukkan beberapa inputan mulai dari nama peternak, tanggal diagnosa serta memilih gejala – gejala yang sekiranya nampak pada hewan ternak. Setelah mengisi semua *form input* yang disediakan, selanjutnya admin dapat menekan tombol “Diagnosa” untuk memulai proses diagnosis. Tahap selanjutnya sistem akan melakukan diagnosa menggunakan metode *forward chaining*, yang dimana proses diagnosa penyakit pada ternak sapi ini akan mengacu pada data gejala yang telah diberikan oleh admin. Data ini kemudian akan diproses sehingga didapatkannya hasil diagnosa penyakit

yang sesuai. Kode dari metode *forward chaining* dapat disajikan pada Gambar 8

```

1 // Mendapatkan data input berdasarkan grafik yang dipilih
2 peternak = input.peternak; // peternak; // peternak; // peternak;
3 // Mendapatkan data input berdasarkan grafik yang dipilih
4 gejala = input.gejala; // gejala; // gejala; // gejala;
5 // Mendapatkan data input berdasarkan grafik yang dipilih
6 diagnosa = input.diagnosa; // diagnosa; // diagnosa; // diagnosa;
7 // Mendapatkan data input berdasarkan grafik yang dipilih
8 sapi = input.sapi; // sapi; // sapi; // sapi;
9 // Mendapatkan data input berdasarkan grafik yang dipilih
10 // Mendapatkan data input berdasarkan grafik yang dipilih
11 // Mendapatkan data input berdasarkan grafik yang dipilih
12 // Mendapatkan data input berdasarkan grafik yang dipilih
13 // Mendapatkan data input berdasarkan grafik yang dipilih
14 // Mendapatkan data input berdasarkan grafik yang dipilih
15 // Mendapatkan data input berdasarkan grafik yang dipilih
16 // Mendapatkan data input berdasarkan grafik yang dipilih
17 // Mendapatkan data input berdasarkan grafik yang dipilih
18 // Mendapatkan data input berdasarkan grafik yang dipilih
19 // Mendapatkan data input berdasarkan grafik yang dipilih
20 // Mendapatkan data input berdasarkan grafik yang dipilih
21 // Mendapatkan data input berdasarkan grafik yang dipilih
22 // Mendapatkan data input berdasarkan grafik yang dipilih
23 // Mendapatkan data input berdasarkan grafik yang dipilih
24 // Mendapatkan data input berdasarkan grafik yang dipilih
25 // Mendapatkan data input berdasarkan grafik yang dipilih
26 // Mendapatkan data input berdasarkan grafik yang dipilih
27 // Mendapatkan data input berdasarkan grafik yang dipilih
28 // Mendapatkan data input berdasarkan grafik yang dipilih
29 // Mendapatkan data input berdasarkan grafik yang dipilih
30 // Mendapatkan data input berdasarkan grafik yang dipilih
31 // Mendapatkan data input berdasarkan grafik yang dipilih
32 // Mendapatkan data input berdasarkan grafik yang dipilih
33 // Mendapatkan data input berdasarkan grafik yang dipilih
34 // Mendapatkan data input berdasarkan grafik yang dipilih
35 // Mendapatkan data input berdasarkan grafik yang dipilih
36 // Mendapatkan data input berdasarkan grafik yang dipilih
37 // Mendapatkan data input berdasarkan grafik yang dipilih
38 // Mendapatkan data input berdasarkan grafik yang dipilih
39 // Mendapatkan data input berdasarkan grafik yang dipilih
40 // Mendapatkan data input berdasarkan grafik yang dipilih
41 // Mendapatkan data input berdasarkan grafik yang dipilih
42 // Mendapatkan data input berdasarkan grafik yang dipilih
43 // Mendapatkan data input berdasarkan grafik yang dipilih
44 // Mendapatkan data input berdasarkan grafik yang dipilih
45 // Mendapatkan data input berdasarkan grafik yang dipilih
46 // Mendapatkan data input berdasarkan grafik yang dipilih
47 // Mendapatkan data input berdasarkan grafik yang dipilih
48 // Mendapatkan data input berdasarkan grafik yang dipilih
49 // Mendapatkan data input berdasarkan grafik yang dipilih
50 // Mendapatkan data input berdasarkan grafik yang dipilih
51 // Mendapatkan data input berdasarkan grafik yang dipilih
52 // Mendapatkan data input berdasarkan grafik yang dipilih
53 // Mendapatkan data input berdasarkan grafik yang dipilih
54 // Mendapatkan data input berdasarkan grafik yang dipilih
55 // Mendapatkan data input berdasarkan grafik yang dipilih
56 // Mendapatkan data input berdasarkan grafik yang dipilih
57 // Mendapatkan data input berdasarkan grafik yang dipilih
58 // Mendapatkan data input berdasarkan grafik yang dipilih
59 // Mendapatkan data input berdasarkan grafik yang dipilih
60 // Mendapatkan data input berdasarkan grafik yang dipilih
61 // Mendapatkan data input berdasarkan grafik yang dipilih
62 // Mendapatkan data input berdasarkan grafik yang dipilih
63 // Mendapatkan data input berdasarkan grafik yang dipilih
64 // Mendapatkan data input berdasarkan grafik yang dipilih
65 // Mendapatkan data input berdasarkan grafik yang dipilih
66 // Mendapatkan data input berdasarkan grafik yang dipilih
67 // Mendapatkan data input berdasarkan grafik yang dipilih
68 // Mendapatkan data input berdasarkan grafik yang dipilih
69 // Mendapatkan data input berdasarkan grafik yang dipilih
70 // Mendapatkan data input berdasarkan grafik yang dipilih
71 // Mendapatkan data input berdasarkan grafik yang dipilih
72 // Mendapatkan data input berdasarkan grafik yang dipilih
73 // Mendapatkan data input berdasarkan grafik yang dipilih
74 // Mendapatkan data input berdasarkan grafik yang dipilih
75 // Mendapatkan data input berdasarkan grafik yang dipilih
76 // Mendapatkan data input berdasarkan grafik yang dipilih
77 // Mendapatkan data input berdasarkan grafik yang dipilih
78 // Mendapatkan data input berdasarkan grafik yang dipilih
79 // Mendapatkan data input berdasarkan grafik yang dipilih
80 // Mendapatkan data input berdasarkan grafik yang dipilih
81 // Mendapatkan data input berdasarkan grafik yang dipilih
82 // Mendapatkan data input berdasarkan grafik yang dipilih
83 // Mendapatkan data input berdasarkan grafik yang dipilih
84 // Mendapatkan data input berdasarkan grafik yang dipilih
85 // Mendapatkan data input berdasarkan grafik yang dipilih
86 // Mendapatkan data input berdasarkan grafik yang dipilih
87 // Mendapatkan data input berdasarkan grafik yang dipilih
88 // Mendapatkan data input berdasarkan grafik yang dipilih
89 // Mendapatkan data input berdasarkan grafik yang dipilih
90 // Mendapatkan data input berdasarkan grafik yang dipilih
91 // Mendapatkan data input berdasarkan grafik yang dipilih
92 // Mendapatkan data input berdasarkan grafik yang dipilih
93 // Mendapatkan data input berdasarkan grafik yang dipilih
94 // Mendapatkan data input berdasarkan grafik yang dipilih
95 // Mendapatkan data input berdasarkan grafik yang dipilih
96 // Mendapatkan data input berdasarkan grafik yang dipilih
97 // Mendapatkan data input berdasarkan grafik yang dipilih
98 // Mendapatkan data input berdasarkan grafik yang dipilih
99 // Mendapatkan data input berdasarkan grafik yang dipilih
100 // Mendapatkan data input berdasarkan grafik yang dipilih

```

Gambar 8 Program Forward Chaining

Seperti yang dijelaskan sebelumnya, pada proses diagnosa admin akan memasukkan nama peternak, tanggal diagnosa dan memilih gejala – gejala yang sekiranya ada pada hewan ternak. Tampilan halaman untuk melakukan diagnosa penyakit pada hewan ternak dapat disajikan pada Gambar 9.



Gambar 9 Halaman Diagnosa Penyakit

3.1.2. Halaman Laporan Bulanan

Halaman laporan bulanan merupakan halaman yang akan menampilkan data hasil diagnosa yang telah dilakukan oleh admin. Pada halaman ini akan menampilkan data nama peternak, tanggal diagnosa serta hasil diagnosa yang telah dilakukan pada hewan ternak. Data yang ada pada laporan bulanan ini nantinya akan dicocokkan dengan hasil diagnosa secara manual oleh dokter hewan, kemudian hasil dari pencocokan tersebut akan dijadikan pembandingan untuk melihat tingkat akurasi dari sistem pakar yang telah dibuat. Tampilan halaman laporan bulanan dapat disajikan pada Gambar 10.



Gambar 10 Halaman Laporan Bulanan

3.2. Pengujian Sistem

Pengujian merupakan tahap yang utama dalam pembuatan suatu aplikasi. Pengujian dari sistem pakar yang telah dibuat ini akan menggunakan dua pengujian yaitu pengujian *confusion matrix* untuk menguji akurasi dari sistem serta pengujian *blackbox testing* untuk menguji fungsionalitas dari sistem. Hasil pengujian yang didapat akan dijadikan sebagai tolak ukur dalam proses pengembangan selanjutnya. Pengujian ini dilakukan untuk mengetahui hasil yang didapat dari aplikasi yang telah dibuat. Pengujian dilakukan untuk mengidentifikasi sistem yang dikembangkan sesuai dengan analisis dan perancangan sistem.

3.2.1. Blackbox Testing

Penerapan *black box testing* pada penelitian ini bertujuan untuk menguji fungsionalitas dari sistem yang telah dibuat. Tahapan dari pengujian ini akan berfokus terhadap fungsionalitas dari halaman yang telah dibuat tanpa memperhatikan *source code* dari sistem. Adapun bagian dari pengujian ini adalah pemberian skenario tertentu dengan tujuan untuk hasil yang dicapai oleh sistem telah sesuai dengan hasil yang diharapkan.

Pengujian Diagnosa penyakit bertujuan untuk menguji fungsionalitas dari halaman diagnosa penyakit. Proses pengujian ini akan memberikan skenario – skenario tertentu dengan harapan hasil dari pengujian ini sesuai dengan hasil yang telah diperkirakan sebelumnya. Pengujian ini juga diharapkan dapat membantu dalam mengidentifikasi serta memperbaiki potensi kesalahan fungsionalitas yang terjadi pada halaman diagnosa penyakit. Adapun skenario pengujian halaman diagnosa penyakit dapat disajikan pada Tabel 3

Tabel 3 Pengujian Diagnosa Penyakit

Skenario	Hasil Diharapkan	Hasil Didapatkan
Mengakses halaman diagnosa penyakit	Menampilkan form untuk melakukan diagnosa penyakit.	Form untuk melakukan diagnosa penyakit ditampilkan.

Skenario	Hasil Diharapkan	Hasil Didapatkan
Memasukan data nama peternak, tanggal diagnosa dan memilih gejala yang ada pada hewan	Jika gejala sesuai aturan antara penyakit pada hewan sesuai akan menampilkan nama penyakit, informasi singkat serta solusi dari penyakit yang ditampilkan. Jika gejala tidak sesuai dengan aturan maka akan memunculkan “Penyakit Tidak Ditemukan”	Hasil diagnosa berhasil ditampilkan.

Setelah seluruh skenario pengujian dilakukan, didapatkan hasil telah selesai dengan hasil yang diharapkan sebelum awal pengujian. Jadi berdasarkan hasil pengujian dapat ditarik kesimpulan bahwa seluruh fungsionalitas dari sistem telah berjalan dengan baik tanpa adanya kendala sama sekali.

3.2.2. Confusion Matrix

Confusion Matrix adalah tabel yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah. Untuk melakukan pengujian *confusion matrix* terlebih dahulu harus diketahui hasil diagnosa penyakit sapi dari Dinas Peternakan Kota Kendari. Pada penelitian ini penulis menggunakan Data Laporan Pelayanan Kesehatan Hewan Dinas Peternakan Kota Kendari Bulan April hingga Bulan September tahun 2023. Perbandingan data yang digunakan dapat disajikan pada Tabel 4.

Tabel 4 Hasil Perbandingan Data Pengujian

no	Nama	Diagnosa Awal	Hasil Diagnosa Sistem	Hasil	Ket
1	S	Helminthiasis	Helminthiasis	Sesuai	TP
2	L.A	Artritis	Artritis	Sesuai	TP
3	L.A	Helminthiasis	Helminthiasis	Sesuai	TP
4	A.T	Helminthiasis	Helminthiasis	Sesuai	TP
5	U.D	Helminthiasis	Helminthiasis	Sesuai	TP
6	M.T	Helminthiasis	Helminthiasis	Sesuai	TP
7	H.A	Myasis	Myasis	Sesuai	TP

no	Nama	Diagnosa Awal	Hasil Diagnosa Sistem	Hasil	Ket
8	H.I	Helminthi asis	Helminthi asis	Sesuai	TP
9	W.S	Helminthi asis	Helminthi asis	Sesuai	TP
10	T.H	Bovine Ephemera l Fever	Infectious Bovine Rhinotracheitis	Tidak Sesuai	FN
11	D.W	Pink eye	Pink eye	Sesuai	TP
12	N.B	Helminthi asis	Helminthi asis	Sesuai	TP
13	R.M	Pink eye	Pink eye	Sesuai	TP
14	R.M	Helminthi asis	Helminthi asis	Sesuai	TP
15	I.R	Helminthi asis	Helminthi asis	Sesuai	TP
16	S.K	Helminthi asis	Helminthi asis	Sesuai	TP
dan seterusnya...					
49	S.H	Helminthi asis	Helminthi asis	Sesuai	TP
50	A.F	Scabies	Scabies	Sesuai	TP
51	B.N	Helminthi asis	Helminthi asis	Sesuai	TP
52	H.A	Bovine Ephemera l Fever	Bovine Ephemera l Fever	Sesuai	TP
53	K.M	Helminthi asis	Helminthi asis	Sesuai	TP
54	J.N	Bovine Ephemera l Fever	Infectious Bovine Rhinotracheitis	Tidak Sesuai	FN
55	T.A	Helminthi asis	Helminthi asis	Sesuai	TP
56	LA G	Helminthi asis	Helminthi asis	Sesuai	TP

Mengacu pada Tabel 4 hasil perbandingan data di temukanlah hasil perbandingan antara data diagnosa awal dari laporan Puskesmas dan hasil sistem. Terdapat 56 data yang digunakan yang merupakan laporan Pelkeswan dari April hingga September 2023, dari data tersebut terdapat 49 data yang sesuai dengan hasil diagnosa sistem dan 7 data yang tidak sesuai dengan hasil diagnosa sistem. Tahapan selanjutnya adalah dilakukannya perhitungan tingkat akurasi dari sistem diantaranya adalah *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) dengan aturan sebagai berikut

1. Jika status pada data awal sama dengan status dari sistem maka *True Positive* (TP).
2. Jika status pada data awal tidak sesuai dengan hasil sistem maka *False Negative* (FN).
3. Jika status pada hasil sistem tidak sesuai dengan data awal maka *True Negative* (TN).
4. Jika status pada hasil sistem sesuai dengan data awal maka *False Positive* (FP).

Hasil dari perbandingan diagnosa awal dan hasil dari sistem dapat disajikan pada Tabel 5.

Tabel 5 Hasil *Confusion Matrix*

Kriteria Pengujian	Jumlah
TP (True Positive)	49
TN (True Negative)	0
FP (False Positive)	0
FN (False Negative)	7

Setelah data hasil perbandingan didapatkan maka tahap selanjutnya adalah melakukan perhitungan untuk mengukur tingkat keakuratan dari sistem yang telah dibuat, yang dimana tingkat keakuratan ini akan menggambarkan seberapa akurat hasil dari sistem yang telah dibuat. Adapun perhitungan tingkat akurasi sistem dapat disajikan sebagai berikut

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \\
 &= \frac{49+0}{49+0+0+7} \times 100\% \\
 &= \frac{49}{56} \times 100\% \\
 &= 87.5\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Precision} &= \frac{TP}{TP+FP} \times 100\% \\
 &= \frac{49}{49+0} \times 100\% \\
 &= 100\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{TP}{TP+FN} \times 100\% \\
 &= \frac{49}{49+7} \times 100\% \\
 &= 87.5\%
 \end{aligned}$$

Dari hasil pengujian yang telah dilakukan didapatkanlah hasil bahwa nilai *accuracy* dari sistem yaitu sebesar 87.5%, nilai *precision* sebesar 100% dan nilai *recall* sebesar 87.5%.

4. KESIMPULAN

Berdasarkan hasil pembahasan yang telah dilakukan maka dapat ditarik kesimpulan sistem pakar diagnosa penyakit hewan ternak sapi menggunakan metode *Forward Chaining* telah berjalan dengan semestinya dan diharapkan dapat membantu proses diagnosa penyakit hewan ternak sapi. Berdasarkan hasil pengujian fungsionalitas menggunakan metode *blackbox testing* telah menunjukkan fungsionalitas dari seluruh bagian sistem telah sesuai. Berdasarkan hasil pengujian *confusion matrix* menunjukkan tingkat akurasi sebesar 87.5%, tingkat presisi 100% dan tingkat recall sebesar 87.5%. Nilai akurasi dari yang kurang begitu baik dari sistem disebabkan oleh beberapa faktor diantaranya keterbatasan dari

metode yang digunakan, ketidakpastian pada sistem dan

keterbatasan interfensi yang dilakukan oleh sistem. Meskipun begitu sistem telah mampu digunakan secara luas dan diharapkan dapat membantu dalam proses diagnosa penyakit pada hewan ternak sapi.

Penelitian selanjutnya disarankan untuk menggunakan algoritma lain atau membandingkan 2 algoritma dalam proses diagnosa penyakit pada sapi ternak sehingga dapat menghasilkan hasil yang lebih akurat nantinya. Selanjutnya diharapkan dapat menggunakan bahasa pemrograman lain sehingga dapat melihat serta membandingkan kelebihan maupun kekurangan bahasa pemrograman yang digunakan dalam proses pembuatan sistem pakar, penelitian selanjutnya juga diharapkan dapat mengembangkan sistem pakar berbasis mobile dengan tujuan agar lebih mudah digunakan oleh calon pengguna.

5. DAFTAR PUSTAKA

- [1] Direktorat Statistik Peternakan Perikanan dan Kehutanan, *Peternakan Dalam Angka 2023*, vol. 8, no. 1. Jakarta: Badan Pusat Statistik, 2023.
- [2] M. K. Junersi, Y. T. R. M. R. Simarmata, dan M. U. E. Sanam, "Laporan Kasus Scours Pada Anak Babi Di Tanah Merah, Desa Noelbaki, Kecamatan Kupang Tengah, Kabupaten Kupang," *Jurnal Veteriner Nusantara*, vol. 4, no. 1, hlm. 1–9, 2021.
- [3] A. A. Laksono, M. Syahlanisyiam, dan P. Rosyani, "Sistem Pakar Diagnosa Penyakit Kulit Dengan Menggunakan Metode Forward Chaining," *Jurnal Manajemen, Ekonomi, Hukum, Kewirausahaan, Kesehatan, Pendidikan Dan Informatika (MANEKIN)*, vol. 1, no. 4, hlm. 152–157, 2023, [Daring]. Tersedia pada: <https://journal.mediapublikasi.id/index.php/manekin>
- [4] R. Firnando, Asnawati, dan J. Fredicka, "Sistem Pakar Diagnosis Penyakit Pada Tanaman Kelapa Sawit Menggunakan Metode Forward Chaining Berbasis Android," *Jurnal Komputer, Informasi dan Teknologi*, vol. 2, no. 2, hlm. 643–652, 2022, doi: 10.53697/jkomitek.v2i2.
- [5] F. Arzalega, Rosalinda, R. Kamil, dan A. Sani, "Analisa Penerapan Metode Forward Chaining Untuk Mendeteksi Kerusakan Komputer Pada PT. JakPro SBU Pasar Muara Karang," *Jurnal Bidang Penelitian Informatika*, vol. 1, no. 2, hlm. 119–128, 2023, [Daring]. Tersedia pada: <https://ejournal.kreatifcemerlang.id/index.php/jbpi>
- [6] F. R. B. Putra, A. Fadlil, dan R. Umar,

"Analisis Metode Forward Chaining Pada Sistem Pakar Diagnosa Penyakit Hewan Sapi Berbasis Android," *Jurnal Sains Komputer & Informatika (J-SAKTI)*, vol. 5, no. 2, hlm. 1034–1044, 2021.

- [7] A. Noorhaliza, Ilham, dan L. O. A. Syukur, "Pusat Kesehatan Dan Pemeliharaan Hewan Dengan Pendekatan Arsitektur Ekologis Di Kota Kendari," *Jurnal Mahasiswa Jurusan Arsitektur*, vol. 7, no. 1, hlm. 66–73, 2022.

