BAB I

PENDAHULUAN

A. Latar Belakang

Produktifitas ternak yang belum optimal menjadi salah satu permasalahan mendasar dalam pengembangan usaha peternakan sapi. Ada beberapa faktor penentu dalam optimalisasi proses produksi ternak sapi, diantaranya adalah *feeding* (pakan), *breeding* (perbibitan), serta aspek management (tatakelola). Beberapa konsep ini juga tentunya mendapat dukungan dari beberapa faktor lainnya seperti pemanfaatan limbah ternak, kesehatan serta pengelolahan ternak. Inovasi-inovasi seperti ini sangatlah perlu disosialisasikan dan diterapkan khususnya bagi usaha peternak sapi. (Said, dkk, 2018)

Manajemen kesehatan ternak merupakan usaha yang diperlukan untuk mencegah berbagai penyakit yang dapat mengganggu ternak melalui berbagai tindakan, termasuk menjaga higienitas, sanitasi, manajemen pakan, serta pemberian suplemen dan obat-obatan yang dapat meningkatkan imun ternak (Lestari et al., 2020). Penyakit yang menyerang ternak diketahui dapat menurunkan pembentukan daging serta produktivitas ternak karena gangguan penyerapan nutrisi. Gangguan kesehatan hewan tentunya dapat merugikan peternak yang disebabkan oleh kematian ternak, biaya yang dikeluarkan untuk pengobatan, penurunan produksi, serta turunnya efisiensi pakan. Kerugian tersebut menunjukkan bahwa tata laksana kesehatan ternak penting diterapkan dalam usaha peternakan.

Salah satu desa yang sebagian besar penduduknya bermata pencaharian sebagai peternak sapi adalah Desa Pakeng, Kecamatan Lembang, Kabupaten Pinrang. Sektor peternakan sapi di desa ini memiliki potensi besar untuk dikembangkan karena kondisi geografis dan iklim yang mendukung.

Metode tradisional yang masih banyak digunakan oleh peternak menyebabkan keterbatasan informasi tentang teknik beternak yang tepat, serta penanganan kesehatan sapi. Selain itu, peternak sering kali kesulitan dalam pemasaran hasil ternak karena minimnya akses ke informasi harga pasar yang kompetitif.

Salah satu upaya untuk mengatasi masalah tersebut, diperlukan sebuah sistem informasi yang komprehensif dengan unggulan, dapat menyediakan sebuah informasi yang lebih relevan pada sebuah lingkup masalah tertentu, seperti informasi tentang teknik beternak, manajemen pakan, monitoring kesehatan ternak, dan pemasaran produk (Chaniago, dkk, 2019). Dengan sistem ini, diharapkan peternak dapat mengelola ternak mereka lebih efektif dan efisien.

B. Rumusan Masalah

Bagaimana cara memberikan kemudahan akses informasi terkait kesehatan ternak, pakan, perawatan, serta pengelolaan data peternakan sapi yang terintegrasi, guna mendukung keputusan yang lebih tepat dan meningkatkan kesejahteraan peternak di Desa Pakeng?

C. Tujuan Penelitian

Untuk memberikan kemudahan akses informasi terkait kesehatan ternak, pakan, perawatan, serta pengelolaan data peternakan sapi yang terintegrasi, guna mendukung keputusan yang lebih tepat dan meningkatkan kesejahteraan peternak di Desa Pakeng.

D. Batasan Masalah

Agar penelitian lebih terfokus dan tidak meluas dari pembahasan dimaksudkan, maka dengan ini membataskan ruang lingkup penelitian Sistem Informasi Ternak Sapi di Desa Pakeng, Kecamatan Lembang, Kabupaten Pinrang:

1. Penelitian ini hanya akan dilakukan di Desa Pakeng, Kecamatan Lembang, dan tidak mencakup wilayah lainnya.
2. Fokus penelitian ini adalah pada ternak sapi saja, sehingga tidak akan membahas ternak lain.
3. Sistem informasi yang dirancang akan berbasis *web*.

E. Manfaat penelitian

Manfaat dari penelitian ini adalah, peternak di Desa Pakeng, Kecamatan Lembang dapat dengan mudah mengakses informasi mengenai pakan, kesehatan ternak sapi yang baik tanpa harus terus menerus berpatok pada ajaran tradisional saja. Selain itu, dengan menyediakan basis data untuk analisis yang mendalam, sistem informasi mendukung pengambilan keputusan yang lebih baik, serta berfungsi sebagai alat edukasi untuk meningkatkan pengetahuan dan keterampilan peternak.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

1. Pengertian sistem informasi

a. Sistem

Menurut Kamus Besar Bahasa Indonesia (KBBI), sistem adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas. Sistem juga diartikan sebagai susunan yang teratur dari pandangan, teori, asas, dan sebagainya.

Suatu sistem terdiri dari sekumpulan elemen – elemen yang saling bekerja sama untuk mencapai tujuan sistem. Suatu elemen yang telah ada merupakan bagian dari sistem lain yang lebih besar. (Arifin,dkk,2022)

b. Informasi

informasi adalah inti dari komunikasi yang menghubungkan pengirim dan penerima melalui berbagai media. Informasi dipandang sebagai elemen penting dalam transformasi sosial dan pemahaman lintas budaya. (Abisono, dkk,2023)

Informasi adalah data yang telah diproses sehingga memiliki arti atau nilai yang dapat digunakan oleh penerima untuk mendukung pemahaman, pengambilan keputusan, atau tindakan. Informasi memiliki karakteristik seperti relevansi, keakuratan, dan kejelasan yang membuatnya bermanfaat dalam berbagai konteks, baik untuk kebutuhan pribadi, organisasi, maupun masyarakat luas.

c. Sistem Infomasi

Sistem informasi dapat diartikan sebagai suatu sistem yang dibuat oleh manusia yang meliputi berbagai macam komponen – komponen dalam organisasi untuk mencapai temuan yaitu menghasilkan informasi (Arifin,dkk,2022).

Sistem informasi adalah sebuah kombinasi yang terdiri dari perangkat, elemen, dan proses yang saling terorganisir dan bekerja bersama untuk mengolah data menjadi informasi yang bermakna. Sistem informasi ini memiliki tujuan utama untuk mendukung pemahaman, pengambilan keputusan, serta operasional organisasi secara efisien. Dengan kata lain, sistem informasi berfungsi sebagai alat yang menghubungkan komponen dalam suatu sistem dengan informasi yang relevan, sehingga dapat digunakan untuk berbagai kebutuhan individu maupun organisasi.

2. *Website*

Website adalah kumpulan halaman yang digunakan untuk menampilkan informasi dalam berbagai format, seperti teks, gambar, video, dan animasi, baik statis maupun dinamis, yang saling terkait melalui jaringan internet dan dapat diakses melalui perangkat lunak seperti *browser* (Bekti, 2020).

Menurut Abdullah, (2021), *website* adalah ruang informasi yang terdiri atas berbagai dokumen digital yang dapat mencakup teks, multimedia, dan program aplikasi yang dirancang untuk menyediakan layanan berbasis internet, baik untuk kebutuhan pribadi, bisnis, maupun organisasi.

Sriasih, (2022), mencatat bahwa *website* adalah *platform* digital yang memungkinkan interaksi antara penyedia informasi dan pengguna melalui

teknologi berbasis *hyperlink* dan protokol seperti HTTP, yang terus berkembang menjadi lebih dinamis dan interaktif.

3. Basis data *(Database)*

Basis data adalah kumpulan data terkait yang disimpan bersama dengan redundansi terkontrol untuk melayani satu atau lebih aplikasi secara optimal. Data disimpan sedemikian rupa sehingga terlepas dari program yang digunakan orang untuk mengakses data. Akses untuk menambahkan data baru, mengedit, dan mengambil data yang ada dari database adalah umum dan terkontrol. Basis data juga didefinisikan sebagai kumpulan informasi yang terkait secara logis, disimpan bersama, dirancang untuk memenuhi kebutuhan informasi suatu organisasi. Kami juga dapat mendefinisikannya sebagai sistem entri file elektronik (M. Riyan,dkk,2023).

4. Mysql

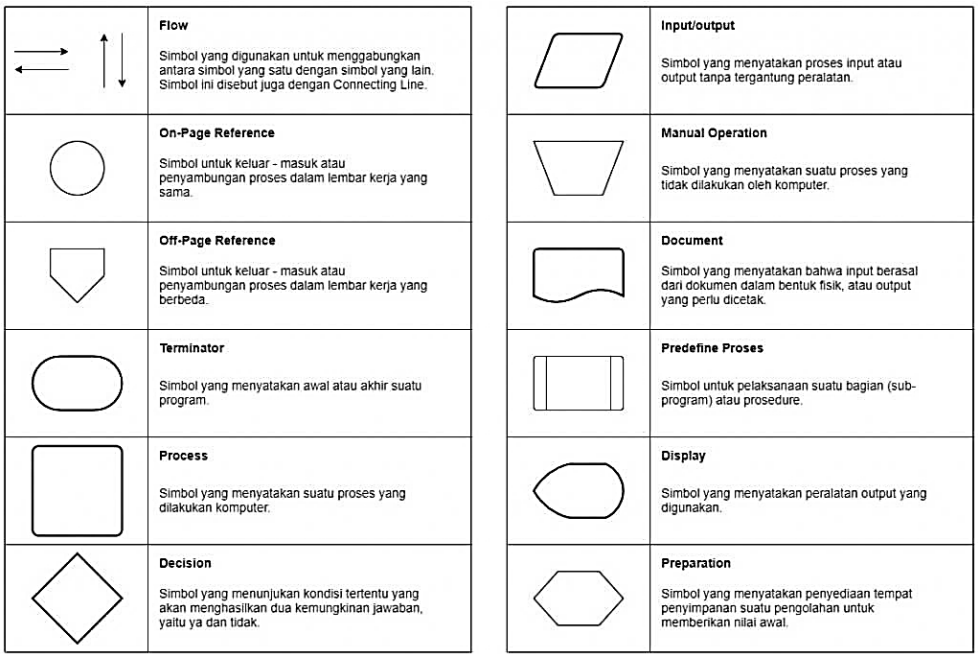
Adaa *MySQ*L adalah sistem manajemen basis data relasional (RDBMS) yang menggunakan bahasa *SQL (Structured Query Language*) sebagai bahasa utama untuk pengelolaan datanya. *MySQL* bersifat *open-source* dan banyak digunakan dalam pengembangan aplikasi berbasis web karena kemampuannya mengelola data dengan cepat, aman, dan stabil.

Menurut Kurniawan (2019), *MySQL* memungkinkan pengembang aplikasi untuk melakukan operasi dasar seperti, *CRUD (Create, Read, Update, Delete)* terhadap data, Pengaturan hak akses pengguna Pengelolaan transaksi data dan integritas basis data.

Dalam pengembangan sistem informasi peningkatan kualitas ternak ini, *MySQL* digunakan sebagai media penyimpanan data peternak, penyuluh, konsultasi, rekomendasi, serta informasi manajemen lainnya, yang kemudian diolah dan ditampilkan melalui aplikasi berbasis *web.*

5. Flowchart

Menurut Amira (2024), *Flowchart* adalah diagram yang menggambarkan aliran dokumen dalam suatu sistem informasi. Implementasi *Flowchart* lebih berguna daripada deskripsi tertulis untuk mengilustrasikan sistem. Menurut Amira (2024), Nugraha et al (2023), fungsi *Flowchart* antara Memperjelas alur logika atau alur proses suatu sistem, memudahkan dalam memahami alur proses atau alur logika suatu sistem yang mungkin terlihat rumit atau sulit dipahami, serta dapat mengidentifikasi kesalahan atau kekurangan dalam sistem.



Gambar 2. 1 Simbol *Flowchart*

6. Unified Modeling Language (UML)

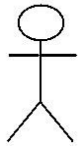
Unified Modelling Language (UML) merupakan sebuah bahasa yang divisualisasikan dalam bentuk gambar atau grafikyang berfungsi untuk memberikan gambaran dan spesifikasi dalam pembangunan dan dokumentasi dari sebuah pengembangan sistem berorientasi objek (*object oriented*) (Saputra,dkk , 2023).

UML terdiri dari sekelompok diagram atau bagan sistem. Diagram atau bagan tersebut menggambarkan permasalahan dan solusinya. Terdapat sembilan diagram pada UML, namun dalam penelitian ini hanya akan digambarkan tiga diagram sebagai berikut:

a. *Use Case diagram*

Use Case diagram merupakan visualisasi dari beberapa komponen, seperti *actor, Use Case,* dan *relasianta*r komponen (Hasanah & Untari, 2020). Beberapa simbol atau notasi digunakan dalam penggambaran fungsionalitas sebuah sistem dalam *use case diagram*. Melalui *use case diagram*, dapat membantu analis dalam penyusunan kebutuhan (*requirement*) pengembangan sistem.

Tabel 2. 1 Simbol *Use Case* Diagram

Simbol	Nama	Keterangan
	*Aktor*	Merepresentasikan pengguna atau peran yang berinteraksi dengan *Use Case*
	*Usecase*	Deskripsi tentang urutan aksi atau interaksi antara aktor dengan sistem yang menghasilkan output tertentu bagi aktor
	Association	Hubungan yang menghubungkan antara aktor dengan *Use Case* untuk menggambarkan interaksi di antara keduanya.

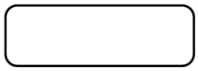
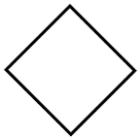
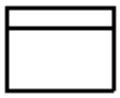
Lanjutan Tabel 2. 2 Simbol *Use Case* Diagram

Simbol	Nama	Keterangan
	*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk *(ncestor)*
	*System*	Gambaran batasan sistem yang mencakup elemen-elemen yang ada di dalam sistem
	*Dependency*	Hubungan antara perubahan yang terjadi pada suatu elemen dapat memengaruhi elemen lainnya, biasanya menunjukkan ketergantungan
	*Include*	Menunjukkan bahwa sebuah *Use Case* selalu menyertakan *Use Case* lain dalam alur eksekusinya.
	*Extend*	Menunjukkan bahwa sebuah *Use Case* dapat menambahkan alur tambahan pada *Use Case* lain secara opsional.

b. Activity diagram

Activity diagram merepresentasikan aliran proses atau aktivitas dalam sebuah sistem yang akan dibangun, mulai dari proses awal, keputusan-keputusan yang terjadi di dalam sistem, hingga bagaimana sebuah proses berakhir (Hasanah & Untari, 2020). *Activity* diagram juga memvisualisasikan proses-proses paralel yang terjadi ketika sistem dieksekusi.

Tabel 2. 3 Simbol *Activity* Diagram

Simbol	Nama	Keterangan
	*(initial Node)* *Status Awal*	Pada diagram aktivitas, alur proses dimulai dari status awal yang menunjukkan titik pertama sistem beroperasi.
	*Activity state*	Aktivitas dalam sistem menggambarkan aksi tertentu dan biasanya diawali dengan kata kerja.
	Control Flow	Menggambarkan perpindahan kendali dari satu aktivitas ke aktivitas lainnya.
	*Marge/Decision*	Situasi di mana terdapat lebih dari satu alternatif aktivitas dalam percabangan.
	*Final Node* (Status Akhir)	Status terakhir yang dicapai oleh sistem, di mana diagram aktivitas menunjukkan satu titik akhir.
	*Fork Node/Join Node*	Penggabungan beberapa aktivitas menjadi satu kesatuan.
	*Swimlane*	Sebuah organisasi bisnis yang bertanggung terhadap aktivitas yang terjadi.

7. Pakan

Pakan merupakan salah satu faktor yang sangat penting dalam pengembangan usaha peternakan. Keberhasilan usaha peternakan tergantung pada manajemen pakan. Biaya pakan merupakan biaya terbesar (60-70%) dari keseluruhan biaya operasional usaha peternakan (Mashur, 2021).

Pakan adalah segala sesuatu yang kita berikan pada hewan/ternak untuk dimakan. Pakan berbeda dengan pangan. Istilah pakan (*feed*) digunakan untuk makanan hewan/ternak, sedangkan istilah pangan (*food*) digunakan untuk makanan manusia. Pakan hewan/ternak merupakan campuran dari beberapa jenis bahan baku pakan baik yang sudah lengkap maupun yang masih akan dilengkapi, yang disusun secara khusus dan mengandung zat-zat gizi yang mencukupi kebutuhan ternak untuk hidup pokok, produksi, dan reproduksi. Pakan yang diberikan pada ternak dapat disesuaikan dengan jenis ternak, umur, bobot badan, dan tujuan pemeliharaan ternak (Mashur,dkk,2023).

B. Kajian Hasil Penelitian Terdahulu

Penelitian terkait sistem informasi untuk meningkatkan kualitas ternak sapi telah dilakukan oleh beberapa peneliti serupa sebelumnya, yang memberikan kontribusi signifikan dalam pengelolaan data dan layanan informasi untuk para Peternak Sapi. Berikut adalah beberapa penelitian terdahulu yang relevan:

1. **Muhammad Irfan Said, Farida Nur Yuliati dan Muhammad Hatta (2018)**

   Penelitian ini berjudul *" Aplikasi Teknologi Pakan Dan Pengolahan Limbah Ternak Pada Kelompok Peternak Sapi Pedaging Di Kabupaten Bantaeng"*. Sistem yang dikembangkan memberikan layanan informasi yang mencakup informasi tentang bagaimana pemberian pakan yang baik dan benar.Tujuan utama penelitian ini adalah menyediakan layanan informasi yang memudahkan masyarakat khususnya Para Peternak dalam meningkatkan kualitas ternak mereka.

2. **Muhammad Heryan Chaniago, Faizatul Amalia, dan Agus Wahyu Widodo (2019)**

   Dalam penelitian berjudul *"Sistem Informasi Manajemen Ternak Sapi Perah di AKA Milk Jakarta"*, sistem yang dirancang bertujuan untuk meningkatkan efisiensi dalam proses manajemen Sapi perah. Fitur utama sistem ini meliputi pengelolaan data, serta informasi manajemen untuk membantu peningkatan kualitas sapi perah. Tujuan utama dari penelitian ini adalah untuk membuat suatu perangkat lunak yang dapat mendokumentasikan kegiatan perawatan dan pemerahan sapi, mencatat riwayat penjualan susu, penentuan pakan dan perhitungan keuntungan..

3. **Sriyanto, Darminto Pujotomo dan Yogi Octavian MP (2010)**

   Penelitian berjudul *"pengembangan sistem informasi berbasis komputer untuk efisiensi dan efektivitas pakan pada usaha penggemukan sapi potong"* berfokus pada pengembangan aplikasi yang mampu mengelola data ternak sapi secara terintegrasi. Sistem ini bertujuan untuk merancang dan mendesain suatu sistem informasi yang disesuaikan dengan kebutuhan sistem untuk mengatasi permasalahan di peternakan.

Dari hasil kajian penelitian terdahulu, dapat disimpulkan bahwa setiap penelitian berfokus pada peningkatan layanan informasi dan efisiensi manajemen Peternak Sapi. Memuat informasi yang membantu para peternak dalam proses peningkatan kualitas produksi dari ternak mereka.

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian ini menggunakan jenis penelitian deskriptif dengan pendekatan kualitatif. Prosedur penelitian yang menghasilkan data deskriptif berupa kata-kata tertulis atau lisan dari orang-orang atau perilaku yang diamati. Penelitian kualitatif, "menggambarkan" berarti mendeskripsikan dan menjelaskan peristiwa, fenomena, atau situasi sosial yang diteliti. Analisis adalah tindakan menafsirkan, menafsirkan, dan membandingkan data penelitian (Waruwu, M. 2023).

Menurut Harahap (2021); Pendekatan penelitian ini adalah pendekatan penelitian lapangan, yaitu. penelitian yang dilakukan untuk mempelajari secara intensif pengaruh lingkungan, kedudukan dan kondisi lapangan suatu satuan penelitian (seperti satuan sosial dan satuan pendidikan). Objek penelitian ini dapat berupa individu, masyarakat atau intuisi. Pada dasarnya objek penelitiannya relatif kecil. Namun fokus dan variabel yang diteliti cukup luas.

B. Lokasi dan waktu

Lokasi penelitian dilaksanakan di Desa Pakeng, Kecamatan Lembang, Kabupaten Pinrang, dengan kurun waktu penelitian dilakukan kurang lebih selama

5(Lima) bulan, dimulai pada awal bulan Januari sampai dengan bulan April 2025. Jadwal pelaksanaan penelitian ditunjukkan pada Tabel 3.1.

Tabel 3. 1 Jadwal Pelaksanaan Penelitian

NO	Uraian Kegiatan	Tahun 2025				
		Januari	**Februari**	**Maret**	**April**	**Mei**
1.	Studi Literatur					
2.	Tahap Desain					
3.	Tahap perancangan dan pembuatan aplikasi					
4.	Tahap pengujian					
5.	Analisis hasil pengujian					

C. Alat dan Bahan

Dalam pelaksanaan penelian ini digunakan beberapa alat dan bahan yang terdiri dari Perangkat Keras (hardware) dan Perangkat Lunak (Software). Perangkat keras yang digunakan terdiri dari laptop asus dengan prosesor Core i3-1005G1 1.20GHz, kapasitas RAM 4GB. Adapun perangkat lunak yang digunakan dalam penelitan ini adalah windows 11.

D. Analisis Kebutuhan dan Rancangan

Dalam proses pembuatan sistem informasi berbasis web ini memanfaatkan alat bantu perancangan UML (Unified Modelling Language) dan beberapa tools bahasa pemrograman seperti bahasa pemrograman PHP dan Basis data MySql.

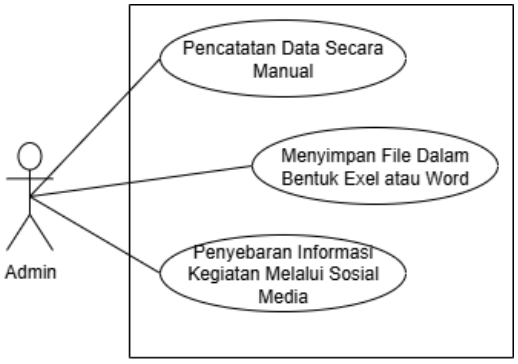
UML (Unifield Modeling Language) merupakan alat bantu pemodelan visual yang digunakan untuk perancangan dan membangun sebuah software atau perangkat lunak dengan berorientasikan pada objek. UML juga sebuah standar

penulisan atau semacam blue print yang mana didalamnya memiliki bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang detail.

1. *Use Case*

Use Case adalah gambaran dari fungsionalitas yang diharapkan dari sebuah sistem, dan lebih merepresentasikan sebuah interaksi antara aktor dan sistem yang terlibat. *Use Case* terdapat aktor yang merupakan sebuah gambaran entitas dari manuasia atau sebuah sistem yang melakukan pekerjaan di sistem.

***a. Use Case* Berjalan**



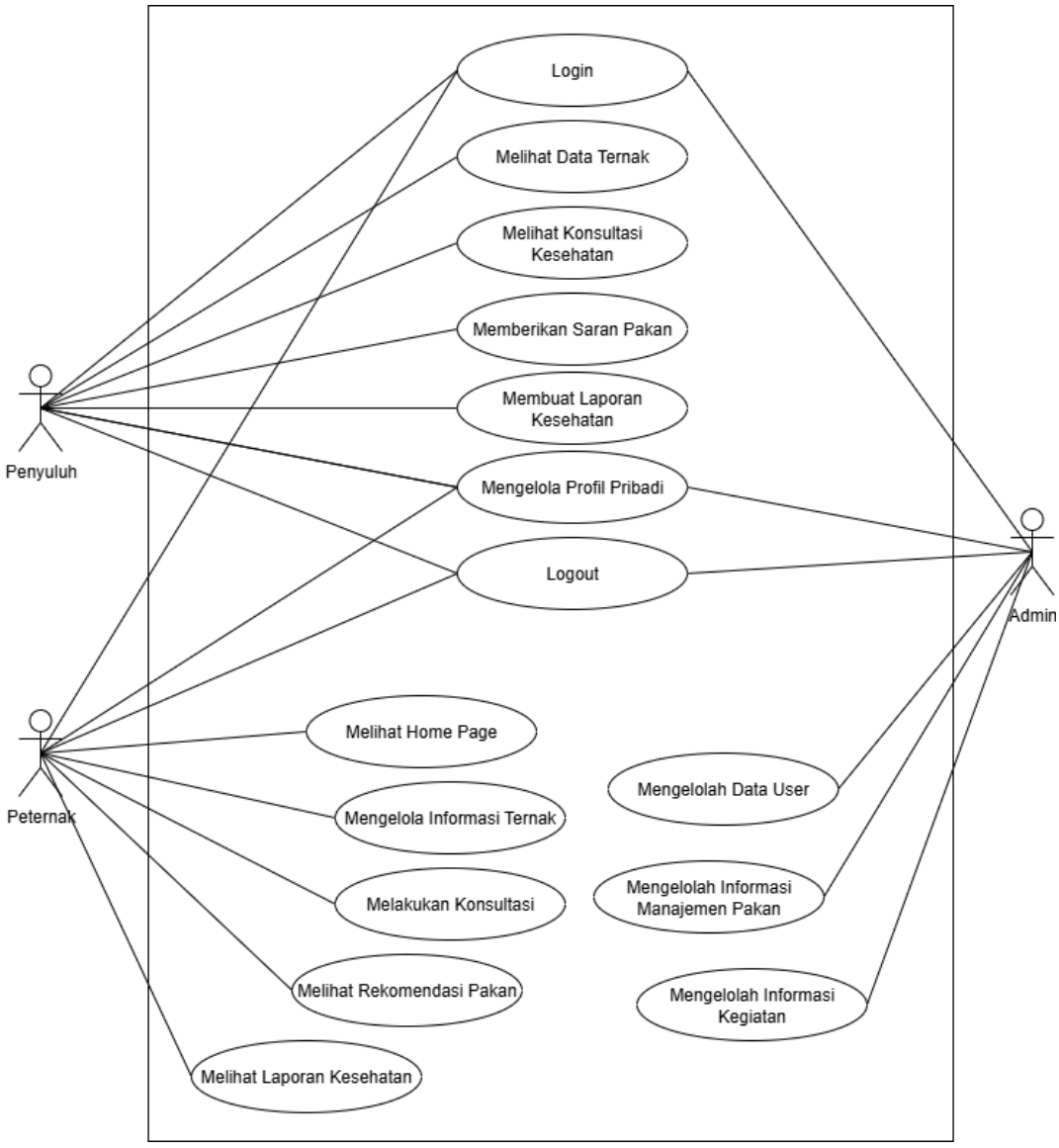
Gambar diatas memberikan penjelasan tentang bagaimana proses pengumpulan data yang dilakukan pemerintah desa setempat, dimana mereka masih melakukan cara manual untuk mengetaui data peternak yang ada di desa dan penyimpanan data serta penyebaran informasi yang masih manual, hal ini menjelaskan bagaimana ketidak efesiennya cara sistem kerja yang tersedia.

Tabel 3. 2 *Use Case* Diagram Berjalan

***Use Case* Diagram**	**Deskripsi**
Pencatatan data Secara manual	Pengurus desa melakukan pencatatan data masih dengan cara manual.
Menyimpan File Dalam Bentuk Exel & Word	*Use Case* ini menjelaskan bahwa hasil pencatatan data yang telah dilakukan oleh admin desa akan disimpan dalam bentuk file exel atau word.

Penyebaran informasi Kegiatan Melalui Sosial Media	*Use Case* ini menjelaskan bahwa pengerus desa akan menyebarkan informasi kegiatan melalui media sosial seperti WA.

b. Use Case Diusulkan



Gambar 3. 1 *Use Case* diagram

Diagram *Use Case* di atas menggambarkan sistem informasi untuk meningkatkan kualitas ternak sapi dengan melibatkan tiga aktor utama, yaitu

Peternak, Penyuluh Pertanian, dan Admin Sistem. Setiap aktor memiliki peran dan akses ke berbagai fitur dalam sistem sesuai dengan tugasnya.

Berikut adalah penjelasan tabel *Use Case berdasarkan gambar* diagram *Use Case*:

Tabel 3. 3 *Use Case* Diagram yang Diusulkan

Nama *Use Case*	**Deskripsi**
Login	Admin, Penyuluh dan Peternak dapat masuk ke dalam sistem dengan kredensial yang valid.
Logout	Admin, Penyuluh dan Peternak dapat keluar dari sistem.
Melihat Dashboard	Aktor dapat melihat informasi umum yang ditampilkan di halaman utama
Melihat Data Ternak	Penyuluh dapat melihat informasi data ternak.
Memberikan Konsultasi Kesehatan	Penyuluh akan melakukan sesi konsultasi, sesuai keluhan Peternak.
Memberikan Saran Pakan	Penyuluh dapat memberikan saran pakan bagi peternak, sesuai dengan kondisi ternak.
Membuat Laporan Kesehatan	Penyuluh akan membuat laporan Kesehatan setalah melakukan analisis Kesehatan ternak.
Mengelola Profil	Aktor dapat mengubah informasi profil milik mereka.
Mengelolah Data *User*	Admin akan memantau dan mengelolah data *User*.
Mengelola Informasi Pakan	Admin dapat menambahkan informasi pakan secara umum sebagai informasi utama.
Mengelola Informasi Kegiatan	Admin dapat melakukan pengeditan terhadap informasi kegiatan.
Mengelola Informasi Ternak	Peternak dapat menambah, mengedit informasi ternak milik mereka.
Melakukan Konsultasi	Peternak dapat mengirimkan konsultasi ternak kepada penyuluh.
Melihat Rekomendasi Pakan	Peternak dapat melihat rekomendasi pakan sesui dengan kebutuhan ternak
Melihat Laporan Kesehatan	Peternak daoat mengakses informasi tentang Kesehatan ternak mereka.

2. Rancangan *Interface* yang diusulkan

Rancangan antarmuka (*Interface*) dalam sistem informasi ternak sapi dirancang untuk memberikan kemudahan bagi pengguna dalam mengakses berbagai fitur utama yang mendukung pengelolaan peternakan. Antarmuka ini memiliki tampilan yang intuitif dan sederhana agar dapat digunakan oleh peternak, penyuluh, dan admin dengan mudah.



Gambar 3. 2 Rancangan *Interface* beranda

Rancangan antarmuka ini menampilkan halaman utama dari Sistem Informasi Ternak Sapi, yang dirancang untuk memberikan kemudahan bagi peternak dalam mengakses informasi terkait manajemen ternak, konsultasi, dan laporan kesehatan.



Gambar 3. 3 Rancangan *Interface* halaman peternak

Rancangan antarmuka ini merupakan halaman utama (dashboard) untuk pengguna dengan peran "Peternak" dalam Sistem Informasi Ternak Sapi. Memungkinkan peternak untuk melakukan pengelolaan data ternak mereka, melakukan konsultasi, melihat saran pakan, serta melihat laporan Kesehatan.



Gambar 3. 4 Rancangan *Interface* halaman penyuluh

Rancangan antarmuka ini merupakan halaman utama (dashboard) untuk pengguna dengan peran "Penyuluh" dalam Sistem Informasi Ternak Sapi, memungkinkan penyuluh Dallam melakukan perannyam, melakukan konsultasi dan memberikan saran pakan sesuai kebutuhan kondisi ternak secara khusus.



Gambar 3. 5 Rancangan *Interface* halaman Admin

Rancangan antarmuka ini merupakan halaman dashboard untuk pengguna dengan peran sebagai Admin dalam Sistem Informasi Ternak Sapi, yang memungkinkan admin dalam pengelolaan data *User*, mengelola informasi pakan secara umum, serta melakukan pengelolalan informasi kegiatan penyuluhan.

Itulah beberapa tampilan *Interface* dari sistem ini, yang tidak menutup kemungkinan akan mengalami perubahan dengan menyesuaikan kebutuhan dari rancangan sistem ini kedepannya.

E. Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini yaitu :

1. Survei

Melakukan proses survei untuk untuk mendapatkan informasi pengguna yang menjadi sasaran.

2. Wawancara

Melakukan kegiatan tanya jawab dan wawancara pada beberapa peternak yang menjadi sasaran, untuk mendapatkan informasi mengenai keluhan, kelebihan apapun yang tengah dialami oleh ternak sapi mereka.

3. Observasi

mengamati bagaimana proses peternak dalam menjaga kesahatan ternak melalui proses pemberian pakan dan pemeliharaan lingkungan kandang ternak.

4. Studi Pustaka

Merupakan bahan-bahan yang diterbitkan secara rutin atau berkala. Pengumpulan data dilakukan dengan cara mengumpulkan literatur yang berkaitan dengan judul penelitian. Bahan –bahan yang penulis gunakan sebagai literatur diperoleh dari beberapa buku dan jurnal terkait. Tujuan dari Membaca dan mempelajari buku, jurnal dan literatur adalah untuk mendapatkan informasi terkait informasi yang lebih relevan dan bermanfaat bagi para peternak di Desa Pakeng, Kecamatan Lembang, Kabupaten Pinrang.

F. Metode Pengujian

Pengujian sistem adalah proses penting dalam siklus hidup perangkat lunak yang bertujuan untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Pengujian sistem dilakukan setelah fase pengembangan dan sebelum sistem diluncurkan atau digunakan secara resmi. Proses pengujian menggunakan dua teknik, yaitu dengan menggunakan pengujian *Balack Box* .

Pengujian *Balack Box* berfokus pada persyaratan fungsional perangkat lunak, Proses *Balack Box Testing* dengan cara mencoba program yang telah dibuat dengan mencoba memasukkan data pada setiap formnya. Pengujian ini diperlukan untuk mengetahui program tersebut berjalan sesuai dengan yang dibutuhkan oleh perusahaan (Wahyudi, dkk, 2016).

BAB IV

HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Berdasarkan penelitian yang dilakukan di Desa Pakeng, Kecamatan Lembang, sistem informasi ternak sapi yang dikembangkan ditujukan untuk memberikan kemudahan kepada peternak dalam mengakses informasi mengenai perawatan, pakan, kesehatan, dan layanan konsultasi dari penyuluh.

Penelitian ini juga melibatkan identifikasi kebutuhan pengguna dan kondisi lapangan, di mana diketahui bahwa sebagian besar peternak di Desa Pakeng masih mengandalkan metode tradisional dalam pengelolaan ternak, dengan akses informasi yang terbatas.

Sistem yang dikembangkan terdiri dari beberapa fitur utama, di antaranya:

Tabel 4. 1 Sistem yang Dikembangkan

Fitur Utama	Deskripsi
Dashboard Informasi	Sebagai pusat informasi utama
Manajemen Data Ternak	Memungkinkan peternak mencatat dan mengelola data ternak miliknya.
Konsultasi Kesehatan	Fitur interaktif antara peternak dan penyuluh.
Rekomendasi Pakan	Informasi pakan yang diberikan oleh penyuluh dan admin.
Manajemen Pengumuman dan Informasi Kegiatan	Sebagai media informasi yang dikelola admin.

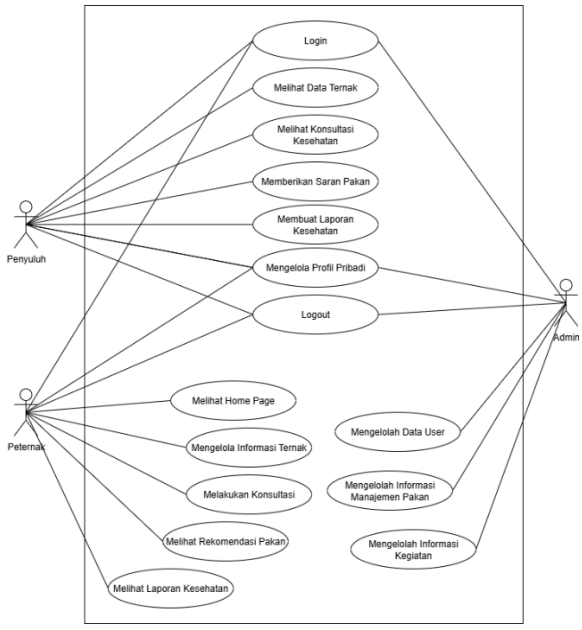
Sistem ini telah diuji oleh beberapa pengguna yang terdiri dari peternak dan penyuluh, yang kemudian memberikan umpan balik melalui angket evaluasi.

B. Analisis Aliran Data UML

1. *Use Case* Diagram

Unified Modelling Language (UML) merupakan sebuah bahasa yang divisualisasikan dalam bentuk gambar atau grafikyang berfungsi untuk memberikan gambaran dan spesifikasi dalam pembangunan dan dokumentasi dari sebuah pengembangan sistem berorientasi objek (*object oriented*).

a. Use Case



Gambar 4. 1 Diagram *Use Case* yang Diusulkan

Diagram *Use Case* di atas menunjukkan interaksi antara Peternak, Penyuluh, dan Admin dengan sistem, di mana masing-masing aktor menjalankan tugas sesuai perannya, seperti pengelolaan data, konsultasi, rekomendasi pakan, dan laporan, sedangkan admin mengelola data pengguna dan informasi sistem.

Tabel 4. 2 Diagram *Use Case* yang Diusulkan

Nama *Use Case*	Deskripsi
Login	Admin, Penyuluh dan Peternak dapat masuk ke dalam sistem dengan kredensial yang valid.
Logout	Admin, Penyuluh dan Peternak dapat keluar dari sistem.
Melihat Home Page	Aktor dapat melihat informasi umum yang ditampilkan di halaman utama
Melihat Data Ternak	Penyuluh dapat melihat informasi data ternak.
Memberikan Konsultasi Kesehatan	Penyuluh akan melakukan sesi konsultasi, sesuai keluhan Peternak.
Memberikan Saran Pakan	Penyuluh dapat memberikan saran pakan bagi peternak, sesuai dengan kondisi ternak.
Membuat Laporan Kesehatan	Penyuluh akan membuat laporan Kesehatan setalah melakukan analisis Kesehatan ternak.
Mengelola Profil	Aktor dapat mengubah informasi profil milik mereka.
Mengelolah Data *User*	Admin akan memantau dan mengelolah data *User*.
Mengelola Informasi Pakan	Admin dapat menambahkan informasi pakan secara umum sebagai informasi utama.
Mengelola Informasi Kegiatan	Admin dapat melakukan pengeditan terhadap informasi kegiatan.
Mengelola Informasi Ternak	Peternak dapat menambah, mengedit informasi ternak milik mereka.
Melakukan Konsultasi	Peternak dapat mengirimkan konsultasi ternak kepada penyuluh.
Melihat Rekomendasi Pakan	Peternak dapat melihat rekomendasi pakan sesui dengan kebutuhan ternak
Melihat Laporan Kesehatan	Peternak daoat mengakses informasi tentang Kesehatan ternak mereka.

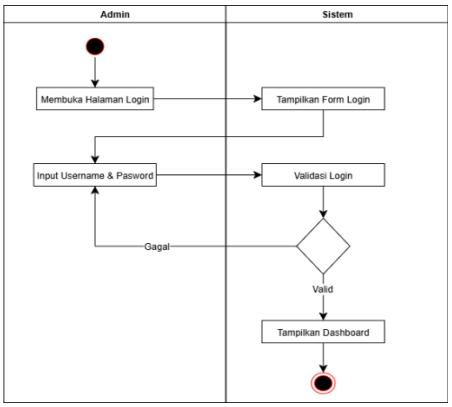
2. Activity Diagram

Activity Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan alur kerja (*workflow*) atau

aktivitas dalam sistem. Diagram ini menggambarkan urutan aktivitas, keputusan, percabangan, dan aliran data dari satu proses ke proses lainnya.

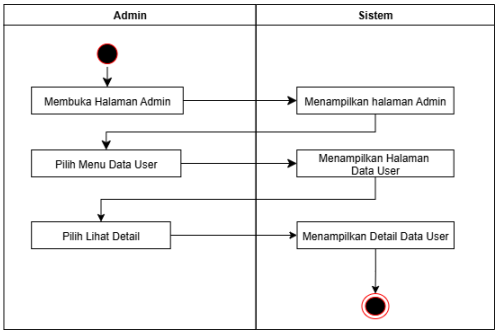
a. *Activity* Diagram Admin

1) *Activity* diagram *Login*



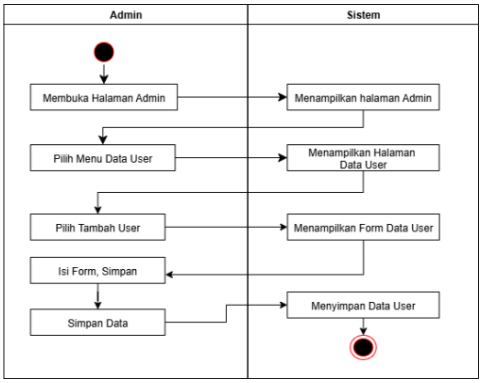
Gambar 4. 2 *Activity* Diagram *Login*

2) *Activity* Diagram Data *User*



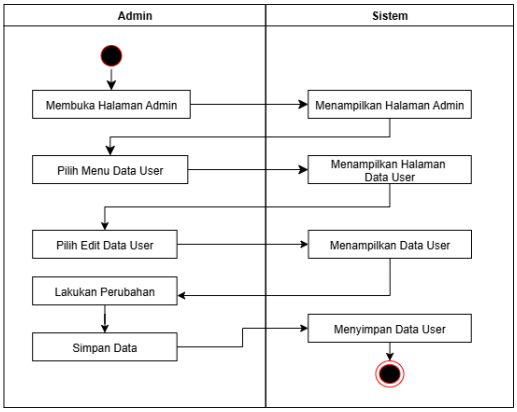
Gambar 4. 3 *Activity* Diagram Data *User*

3) *Activity* Diagram Tambah Data



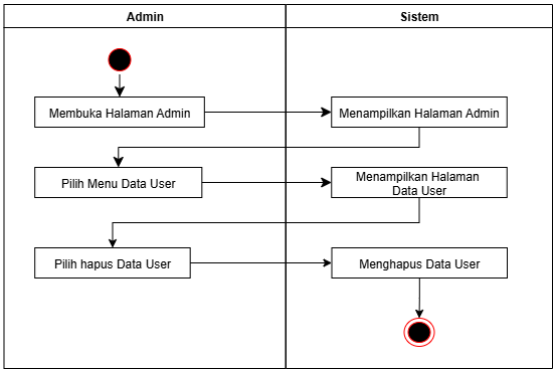
Gambar 4. 4 *Activity* Diagram Tambah Data *User*

4) *Activity* Diagram *Edit* Data *User*



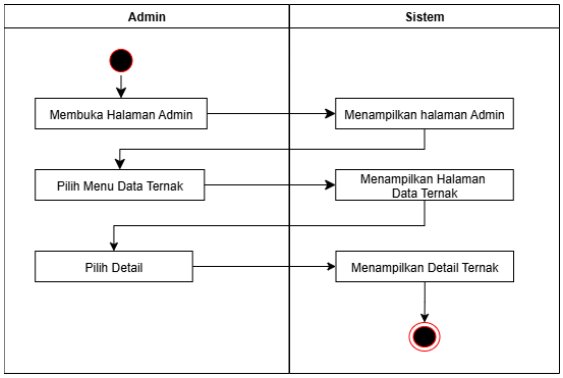
Gambar 4. 5 *Activity* Diagram Edit Data *User*

5) *Activity* Diagram Hapus Data *User*



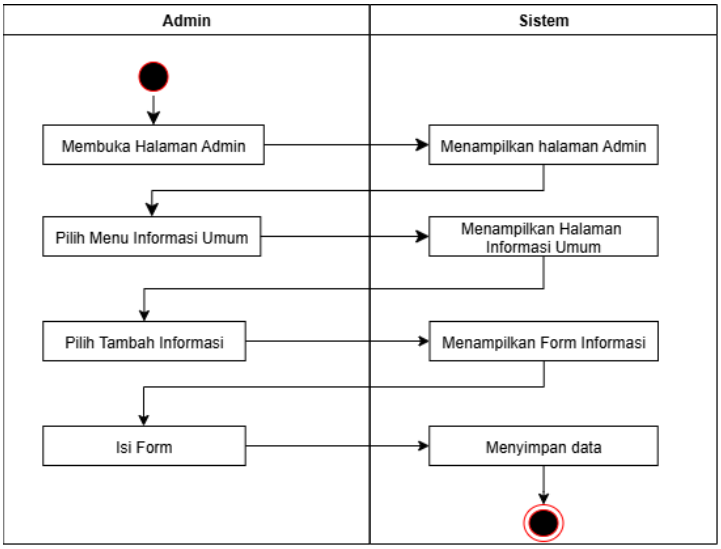
Gambar 4. 6 *Activity* Diagram Hapus Data *User*

6) *Activity* Diagram Data Ternak



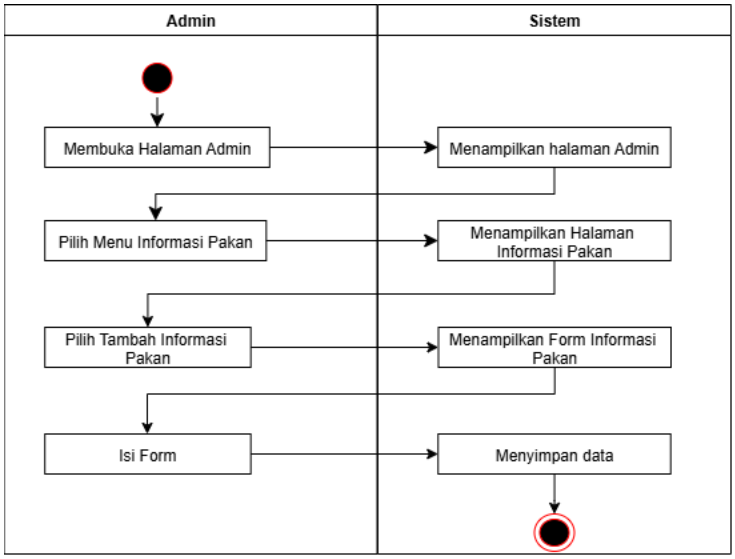
Gambar 4. 7 *Activity* Diagram Data Ternak

7) *Activity* Diagram Tambah Informasi Umum



Gambar 4. 8 *Activity* Diagram Tambah Informasi Umum

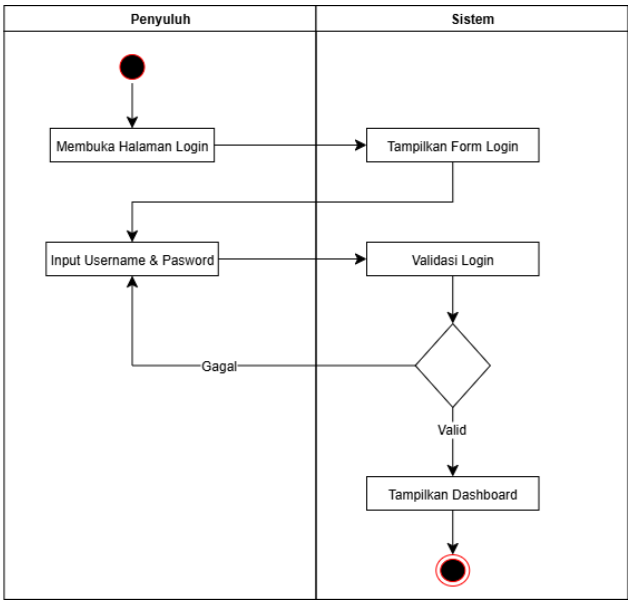
8) *Activity* Diagram Tambah Pakan Ternak



Gambar 4. 9 *Activity* Diagram Tambah Pakan Ternak

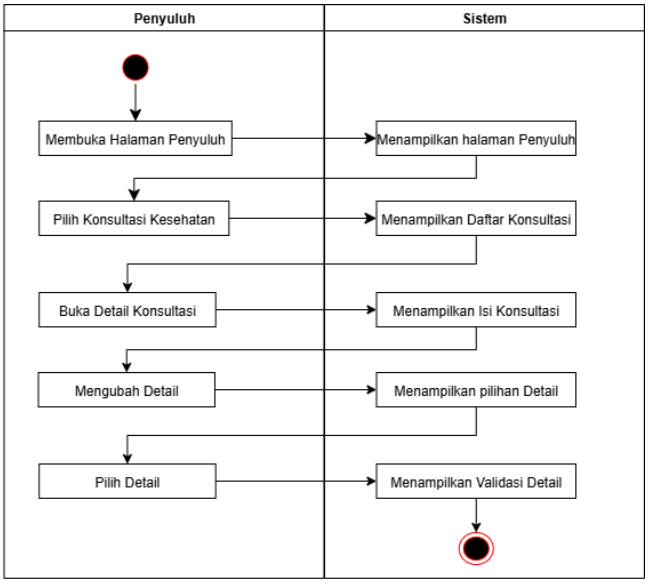
b. Activity Diagram Penyuluh

1) *Activity* Diagram *Login*



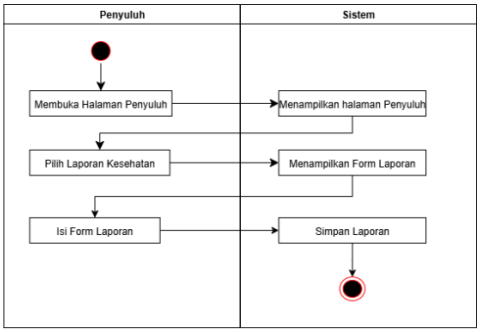
Gambar 4. 10 *Activity* Diagram *Login* Penyuluh

2) *Activity* Diagram Konsultasi Kesehatan



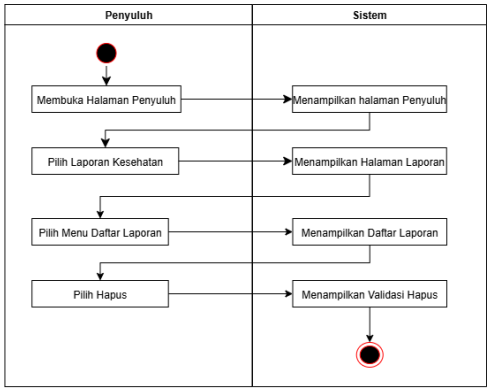
Gambar 4. 11 *Activity* Diagram Konsultasi Kesehatan

3) *Activity* Diagram Laporan Kesehatan



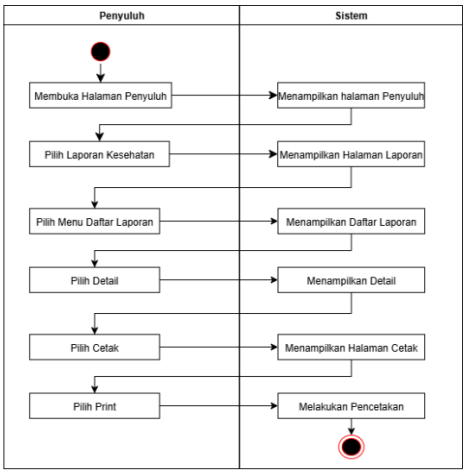
Gambar 4. 12 *Activity* Diagram Laporan Kesehatan

4) *Activity* Diagram Hapus Laporan



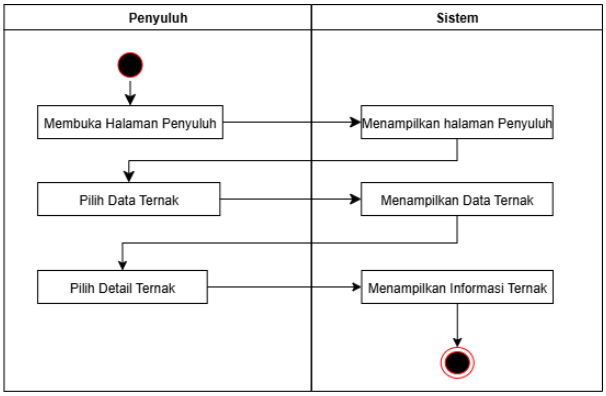
Gambar 4. 13 *Activity* Diagram Hapus Laporan

5) *Activity* Diagram Cetak Laporan



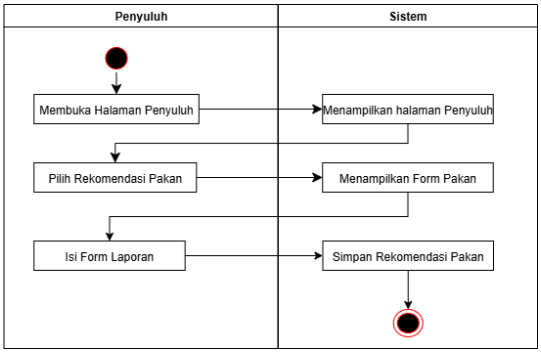
Gambar 4. 14 *Activity* Diagram Cetak Laporan

6) *Activity* Diagram Data Ternak



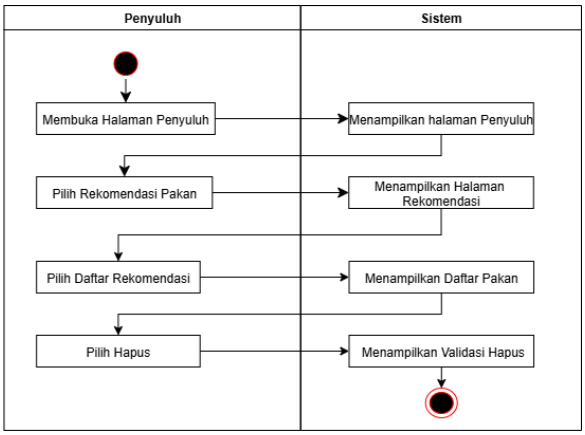
Gambar 4. 15 *Activity* Diagram Penyuluh Data ternak

7) *Activity* Diagram Rekomendasi Pakan



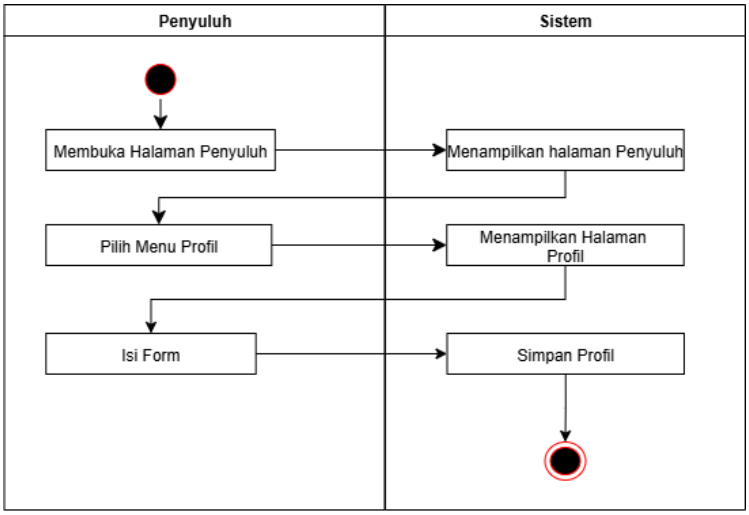
Gambar 4. 16 *Activity* Diagram Rekomendasi Pakan

8) *Activity* Diagram Hapus Rekomendasi Pakan



Gambar 4. 17 *Activity* Diagram Hapus Rekomendasi

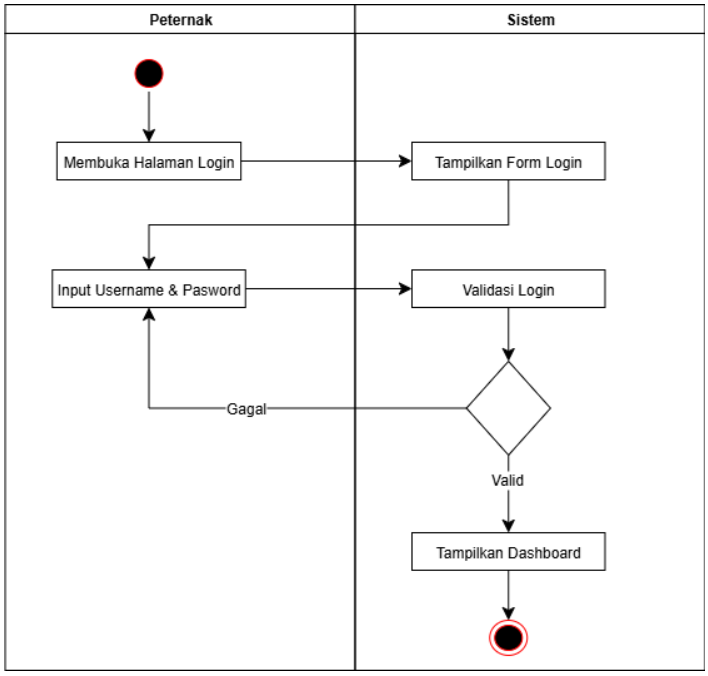
9) *Activity* Diagram Mengubah Profil



Gambar 4. 18 *Activity* Diagram Mengubah Profil

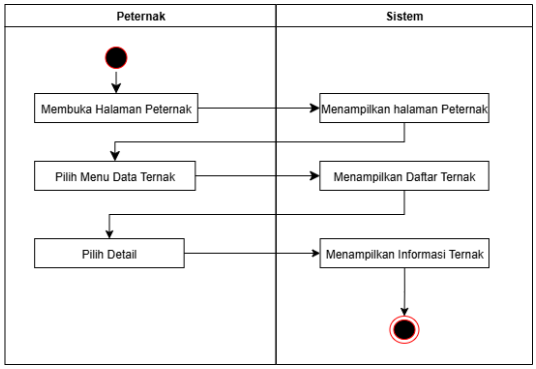
c. Activity Diagram Peternak

1) *Activity* Diagram *Login*



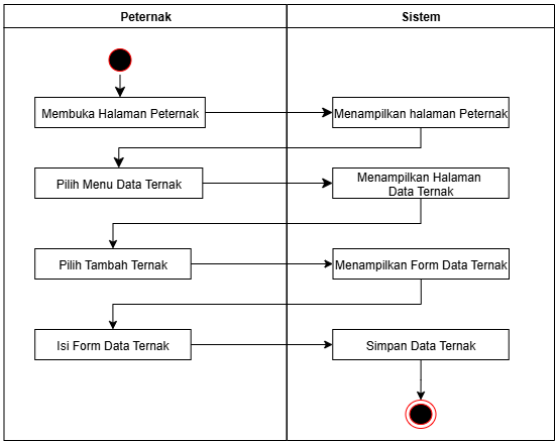
Gambar 4. 19 *Activity* Diagram *Login* Peternak

2) *Activity* Diagram Menampilkan Data Ternak



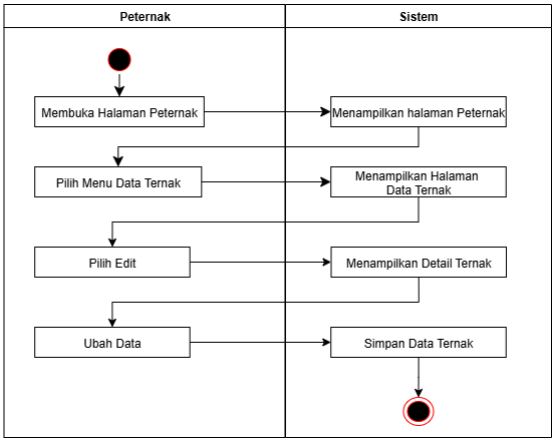
Gambar 4. 20 *Activity* Diagram Data Ternak

3) *Activity* Diagram Menambahkan Data Ternak



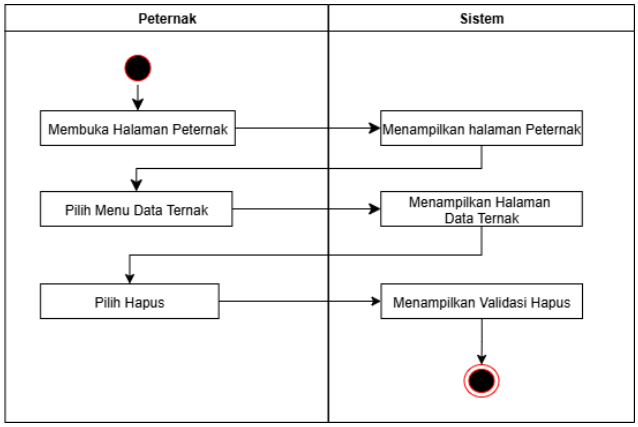
Gambar 4. 21 *Activity* Diagram Tambah Data Ternak

4) *Activity* Diagram Edit Data Ternak



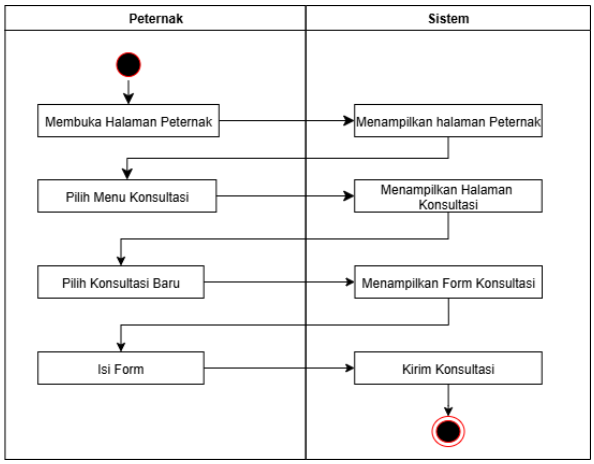
Gambar 4. 22 *Activity* Diagram Edit Data Ternak

5) *Activity* Diagram Menghapus Data Ternak



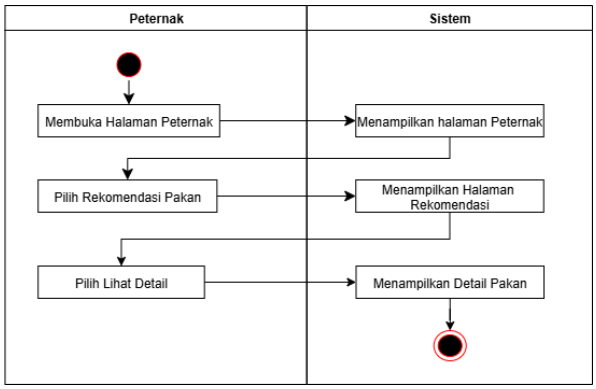
Gambar 4. 23 *Activity* Diagram Hapus Data Ternak

6) *Activity* Diagram Melakukan Konsultasi



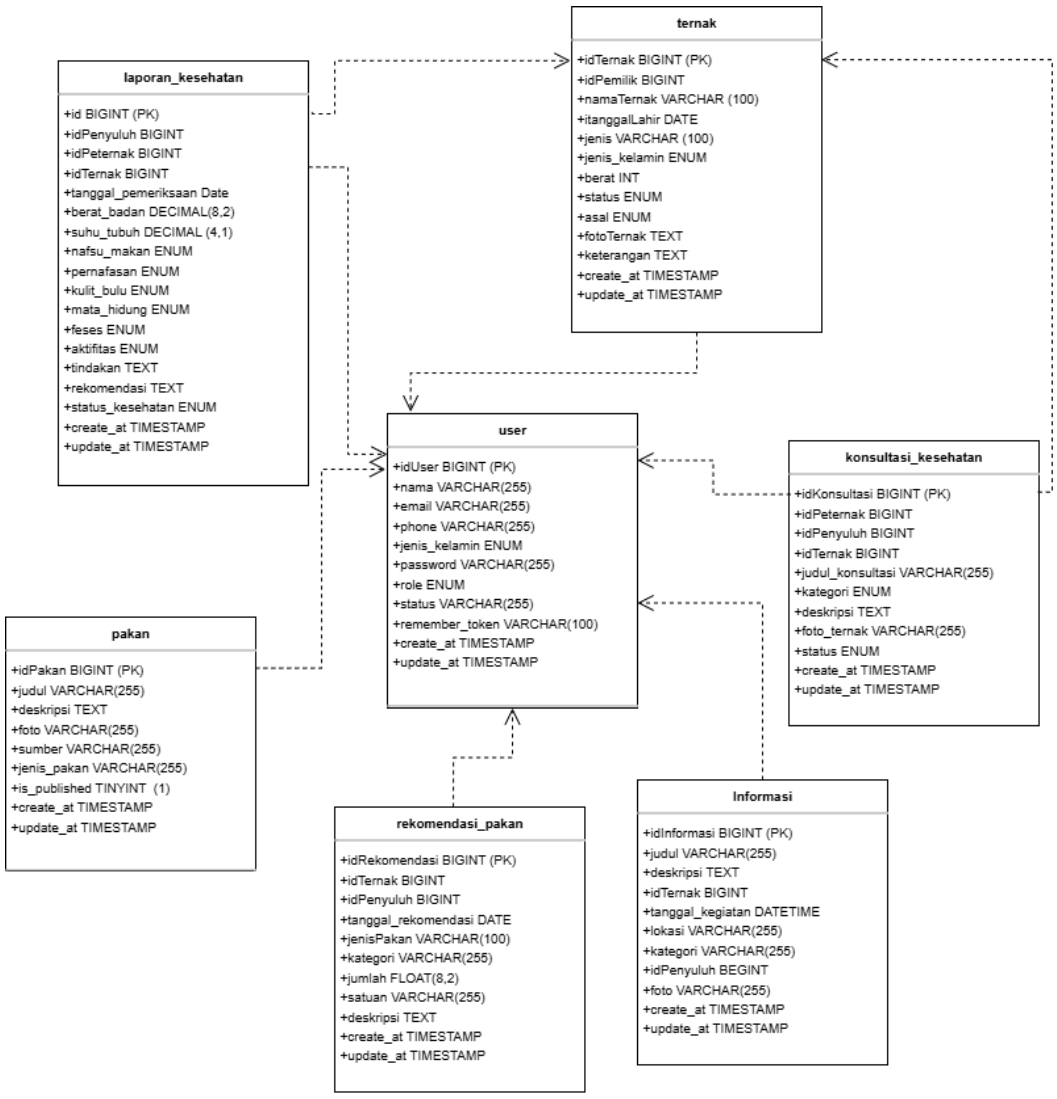
Gambar 4. 24 *Activity* Diagram Konsultasi

7) *Activity* Diagram Melihat Rekomendasi Pakan



Gambar 4. 25 *Activity* Diagram Melihat Rekomendasi Pakan

C. Diagram ERD



Gambar 4. 26 Diagram *ERD*

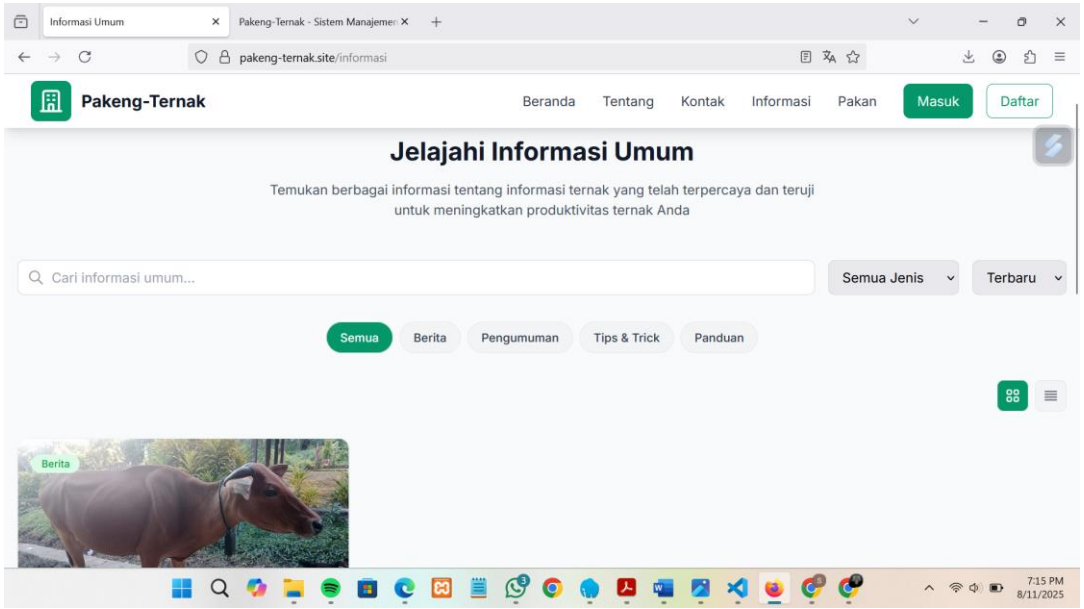
Pada Gambar ERD dijelaskan bagaimana relasi antar tabel dalam sistem manajemen peternakan. Tabel *User*s menyimpan data pengguna seperti peternak, penyuluh, dan admin, termasuk nama, email, password, dan peran. Tabel ternak berisi data hewan ternak yang dimiliki oleh peternak, seperti nama ternak, tanggal lahir, jenis, dan asal. Tabel laporan_kesehatan mencatat hasil pemeriksaan ternak oleh penyuluh, yang terhubung dengan peternak dan ternaknya. Tabel konsultasi_kesehatan mencatat permintaan konsultasi dari peternak kepada

penyuluh terkait kesehatan ternak. Tabel rekomendasi_pakan digunakan penyuluh untuk memberikan saran pakan kepada peternak berdasarkan data ternak. Tabel pakan berisi data jenis-jenis pakan yang dapat dijadikan referensi. Sementara itu, tabel informasi digunakan oleh penyuluh untuk membagikan pengumuman atau kegiatan yang berkaitan dengan peternakan. Setiap tabel memiliki relasi melalui foreign key untuk memastikan keterhubungan data antar entitas dalam sistem.

D. Implementasi Sistem

1. Halaman Informasi Keigatan

Tampilan ini merupakan layar awal yang muncul ketika pengguna pertama kali membuka Website, menampilkan logo sebagai identitas utama.



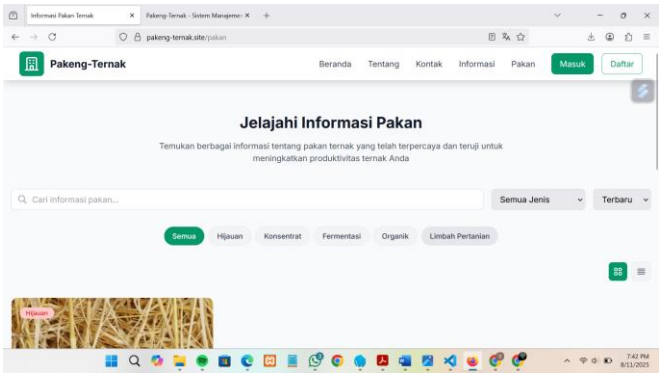
Gambar 4. 27 Halaman Informasi Umum

Berikut script halaman utama:

```
 <?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
class Informasi extends Model
{
    protected $primaryKey = 'idInformasi';
    protected $fillable = [
        'judul',
        'deskripsi',
        'tanggal_kegiatan',
        'lokasi',
        'kategori',
        'idPenyuluh',
        'foto'
    ];
```

Model Informasi pada kode tersebut merupakan representasi dari tabel pada database MySQL yang menyimpan data informasi kegiatan. Model ini menggunakan Eloquent ORM dari Laravel untuk memudahkan proses pengelolaan data, seperti menambah, mengubah, atau menghapus data tanpa harus menulis query SQL secara manual. Primary key yang digunakan adalah idInformasi, yang berfungsi sebagai identitas unik setiap baris data. Properti $fillable berisi daftar kolom yang dapat diisi secara massal, yaitu judul, deskripsi, tanggal_kegiatan, lokasi, kategori, idPenyuluh, dan foto. Dengan menggunakan model ini, alur penyimpanan dan pengambilan data antara aplikasi dan database menjadi lebih terstruktur, aman, dan efisien.

1) Halaman Informasi Pakan



Gambar 4. 28 Halaman Informasi Pakan

Berikut script halaman Informasi Pakan:

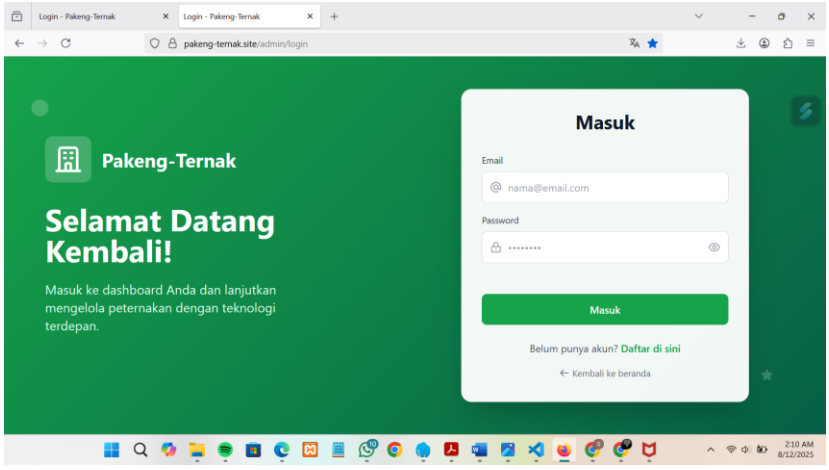
```
 <?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
class InformasiPakan extends Model
{
    protected $table = 'pakan';
    protected $primaryKey = 'idPakan';
    protected $fillable = [
        'idPakan',
        'judul',
        'deskripsi',
        'foto',
        'sumber',
        'jenis_pakan',
        'is_published',
        'created_at',
        'updated_at'
    ];
```

Kelas InformasiPakan pada kode di atas adalah representasi dari tabel pakan di database MySQL menggunakan Eloquent ORM Laravel. Setiap objek

InformasiPakan berkaitan langsung dengan satu baris data pada tabel pakan, di mana primary key-nya adalah idPakan, bukan id default Laravel. Field $fillable menentukan kolom-kolom yang dapat di-mass assign, seperti judul, deskripsi, foto, sumber, jenis_pakan, status publikasi (is_published), serta timestamp created_at dan updated_at. Dengan pendekatan ini, Laravel mempermudah manipulasi data melalui model ini—seperti menambah data baru, mengedit, atau menghapus—tanpa perlu menulis query SQL secara langsung. Alur database-nya bekerja dengan cara ORM menerjemahkan operasi model menjadi query MySQL, sehingga memastikan sinkronisasi antara kode aplikasi dan struktur tabel secara efisien.

2. Halaman *Login*

Tampilan ini merupakan tampilan selanjutnya setelah fokus utama ke home page, *Login* akan memberikan akses kepada *User* untuk masuk kedalam dashboard sesuai dengan peranan masing-masing, setelah tentunya melakukan proses pembuatan akun jika tidak mempunyai akun untuk *Login*.



Gambar 4. 29 Halaman *Login*

Berikut script halaman *Login*:

```
  public function login(Request $request)
    {$request->validate([
            'email' => 'required|email',
            'password' => 'required|min:6',
        ], ['email.required' => 'Silahkan Masukkan Email Anda',
             'email.email' => 'Format email yang Anda masukkan tidak valid',
             'password.required' => 'Silahkan Masukkan Password Anda','password.min' => 'Password minimal terdiri dari 6 karakter',
        ]);
        $infologin = [
            'email' => $request->email,
            'password' => $request->password,
        ];
                    if (Auth::attempt($infologin, $request->has('remember'))) {
            $user = Auth::user();

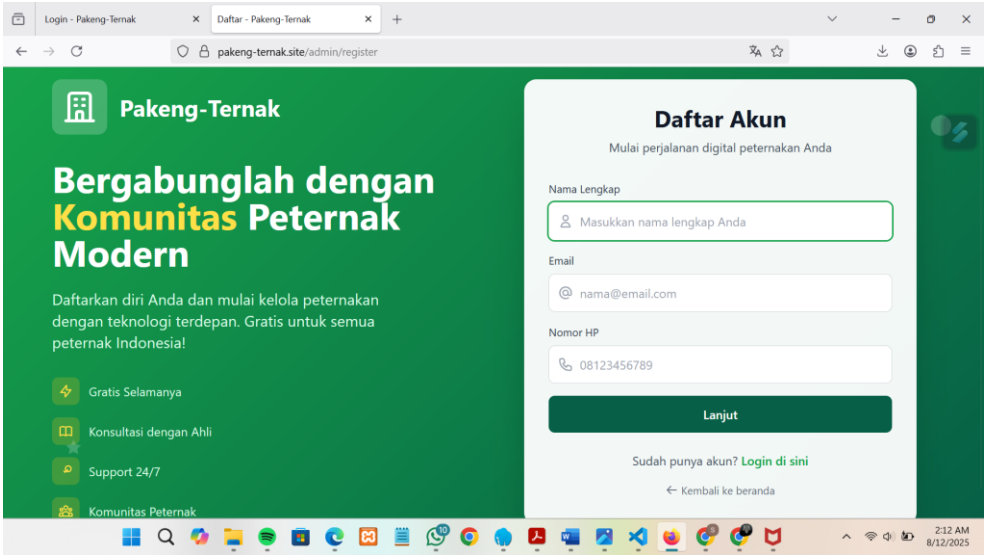
            if ($user->role == "Peternak") {
                return redirect('/peternak/dashboard');
            } elseif ($user->role == "Penyuluh") {
                return redirect('/penyuluh/dashboard');
            } elseif ($user->role == 'Admin') {
                return redirect('/admin/dashboard');
            }
        }
        return redirect('/admin/login')->withErrors(['login' => 'Login Gagal, Email atau Password tidak sesuai!'])->withInput();
```

Kelas *User* pada kode di atas adalah representasi dari tabel *User*s dalam database MySQL, menggunakan Eloquent ORM milik Laravel. Dalam alur database-nya, setiap kali Anda berinteraksi dengan objek *User* di Laravel (seperti

membuat, membaca, memperbarui, atau menghapus data), Eloquent akan menerjemahkannya menjadi query SQL yang dijalankan ke tabel *User*s. Kolom primary key-nya adalah id*User*, bukan id seperti default Laravel. Field yang bisa diisi ($fillable) meliputi nama, email, password, phone, dan role, yang artinya hanya kolom-kolom ini yang dapat dimass-assigned saat membuat atau mengupdate data. Password disembunyikan dalam respons JSON ($hidden) dan secara otomatis di-hash melalui cast ke hashed. Selain itu, kolom email_verified_at secara otomatis akan dikonversi menjadi objek datetime, sehingga memudahkan manipulasi tanggal di Laravel. Semua ini memastikan integrasi yang aman dan efisien antara aplikasi dan database MySQL.

3. Halaman Daftar Akun

Tampilan ini merupakan proses pembuatan akun pengguna, baik sebagai penyuluh ataupun peternak.



Gambar 4. 30 Halaman Daftar Akun

Berikut script daftar akun:

```
'nama.required' => 'Nama Wajib Diisi',
'email.required' => 'Silahkan Masukkan Email Anda',
'email.email' => 'Format email yang Anda masukkan tidak valid',
'email.unique' => 'Email sudah terdaftar',
'password.required' => 'Silahkan Masukkan Password Anda',
'password.min' => 'Password minimal terdiri dari 6 karakter',
'password.confirmed' => 'Konfirmasi password tidak sesuai',
'phone.required' => 'Nomor telepon wajib diisi',
'role.required' => 'Silahkan Pilih Role Anda',
]);
$user = User::create([
'nama' => $request->nama,
'email' => $request->email,
'password' => Hash::make($request->password),
'phone' => $request->phone,
'role' => $request->role
]);
return   redirect('/admin/login')->with('success',   'Registrasi
berhasil! Silahkan Login Ke Akun Anda.');
}
```

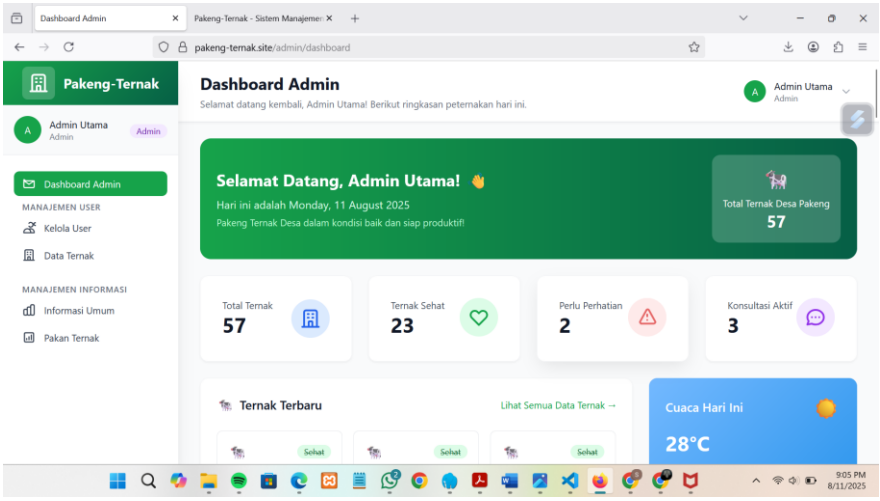
Fungsi register() merupakan bagian dari proses registrasi pengguna dalam aplikasi, yang bertugas memvalidasi input data dari pengguna dan menyimpannya ke dalam database menggunakan model *User*. Validasi dilakukan melalui $request->validate() untuk memastikan bahwa setiap data yang diterima memenuhi kriteria tertentu. Misalnya, field nama wajib diisi dan berupa teks, email harus dalam format email yang benar dan belum terdaftar sebelumnya pada tabel *User*s, password harus terdiri dari minimal 6 karakter dan cocok dengan kolom konfirmasi password, phone wajib berupa angka, dan role harus dipilih antara "Peternak" atau

"Penyuluh". Pesan kesalahan (custom error messages) disediakan untuk memberikan umpan balik yang lebih informatif kepada pengguna saat input tidak valid. Jika semua validasi berhasil, maka data akan disimpan ke database menggunakan *User*::create(), dengan kata sandi yang telah di-hash melalui Hash::make() untuk menjaga keamanan. Setelah penyimpanan berhasil, pengguna akan diarahkan ke halaman *Login* (/admin/*Login*) dengan notifikasi bahwa registrasi telah berhasil.

4. Halaman Admin

1) Halaman Utama Admin

Tampilan ini merupakan halaman utama untuk admin, berisikan informasi informasi umum mengenai ternak, penyuluh, dan peternak itu sendiri. Dashboard ini berfungsi sebagai pusat kendali (control center) visual dan informasi bagi admin peternakan—menyajikan statistik penting secara ringkas, akses cepat ke data detail, serta kondisi lingkungan saat ini seperti cuaca. Semua ini ditampilkan secara *User*-friendly dengan layout modern dan informatif.



Gambar 4. 31 *Dashboard* Admin

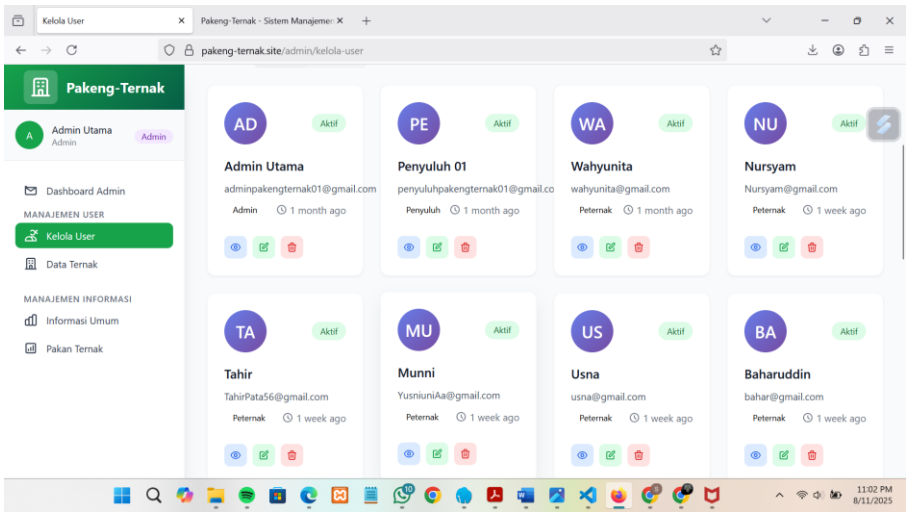
Berikut script halaman utama admin:

```
class AdminController extends Controller
{
    public function index()
    {
        $totalTernak = Ternak::all()->count();
        $ternaks = Ternak::with('pemilik')->get();
          return view('admin.dashboard', compact('totalTernak',
'ternaks'));
    }
    public function userView()
    {
        $userList = User::all();
        $totalUser = User::all()->count();
        $userAktif = User::where('status', 'aktif')->count();
           $userNonAktif = User::where('status', 'nonaktif')-
>count();
       $userSuspend = User::where('status', 'suspend')->count();
               return view('admin.user', compact('userList',
'totalUser', 'userAktif', 'userNonAktif', 'userSuspend'));
    }
```

Data yang diolah dalam metode index() dan *User*View() inilah yang digunakan untuk menampilkan statistik dan informasi di dashboard admin serta tampilan manajemen *User*, seperti terlihat pada screenshot "Pakeng-Ternak". Jika kamu ingin menambahkan statistik seperti "Ternak Sehat", "Perlu Perhatian", atau "Konsultasi Aktif", kamu bisa memperluas fungsi index() agar menghitung dan menyediakan data tersebut secara efisien.

2) Halaman Kelola *User*

Tampilan ini menampilkan semua daftar *User* yang telah melakukan proses pendaftaran akun, admin memiliki kehendak untuk melakukan proses pengeditan terhadap semua data *User*.



Gambar 4. 32 Halaman Kelola *User*

Berikut script Kelola *User*:

```
public function storeUser(Request $request)
    {
        try {
            $validated = $request->validate([
                'nama' => 'required|string|max:255',
                'email' => 'required|email|unique:users,email',
                'password' => 'required|string|min:8|confirmed',
                'role' => 'required|in:admin,peternak,user',
                'phone' => 'nullable|string|max:20',
                'status' => 'required|in:aktif,nonaktif,suspend'
            ]);
            $user = new User();
```

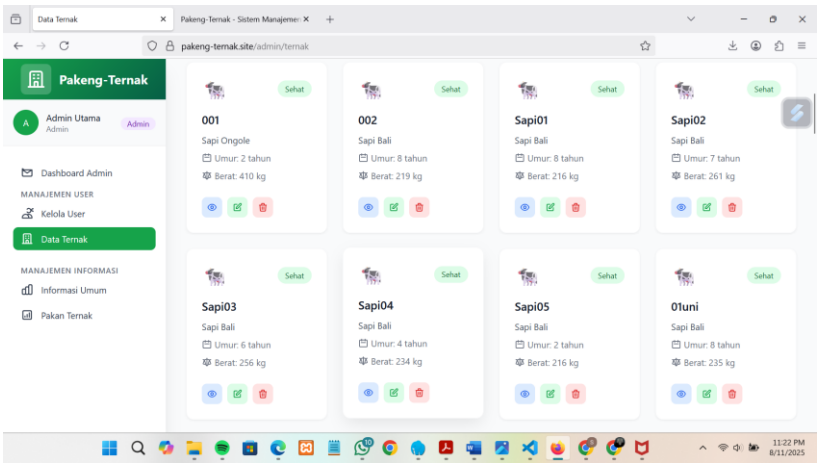
```
        $user->nama = $validated['nama'];
        $user->email = $validated['email'];
        $user->password = Hash::make($validated['password']);
        $user->role = $validated['role'];
        $user->phone = $validated['phone'];
        $user->status = $validated['status'];
        $user->save();

        if ($request->ajax() || $request->wantsJson()) {
            return response()->json([
                'success' => true,
                'message' => 'User berhasil ditambahkan!',
                'data' => $user
            ]);
        }
```

Fungsi store*User*(Request $request) dimulai dengan memvalidasi data input—seperti nama, email, password, role, phone, dan status—dengan aturan yang ketat (misalnya email harus unik di tabel *User*s, password harus dikonfirmasi, dll.). Kalau validasi berhasil, dibuat objek *User* baru, data dimasukkan ke atribut model (termasuk password yang di-hash), lalu disimpan ke database. Setelah itu, Laravel memeriksa jenis permintaan menggunakan ajax() (untuk permintaan AJAX) atau wantsJson() (jika klien mengharapkan respons JSON). Jika salah satu kondisi terpenuhi, fungsi langsung mengirim respons dalam format JSON berisi status kesuksesan, pesan, dan data pengguna baru—buat mendukung penggunaan front-end berbasis JavaScript atau API.

3) Tampilan Halaman Data Ternak

Tampilan ini menampilkan semua daftar data ternak yang telah melakukan disi langsung oleh para peternak.



Gambar 4. 33 Halaman Data Ternak

Berikut script tampilan data Ternak

```
public function ternak()

{$ternakList = Ternak::where('idPemilik', Auth::user()->idUser)-
>get();

$totalTernak = $ternakList->count();

$ternakSehat = Ternak::where('idPemilik', Auth::user()->idUser)-
>where('status', 'sehat')->count();

$ternakSakit = Ternak::where('idPemilik', Auth::user()->idUser)-
>where('status', 'sakit')->count();

$ternakPerawatan = Ternak::where('idPemilik', Auth::user()->idUser)-
>where('status', 'perawatan')->count();

$konsultasiSaya = Konsultasi::where('idPeternak', Auth::user()-
>idUser)->count();

return view('admin.ternak', compact(

'ternakList',

'totalTernak',

'ternakSehat',

'ternakSakit',

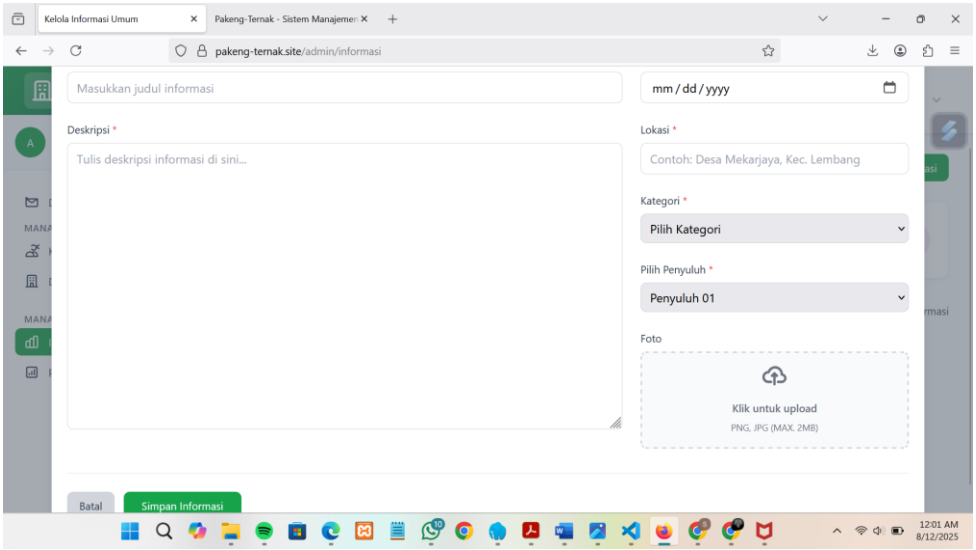
'ternakPerawatan',

'konsultasiSaya'
```

Pada metode ternak(), aplikasi mengambil data ternak milik pengguna saat ini (ditentukan lewat Auth::*User*()->id*User*) menggunakan Ternak::where ('idPemilik', …) -> get() dan menghitung total ternak melalui count(). Selanjutnya, aplikasi menghitung jumlah ternak berdasar status: sehat, sakit, dan perawatan, masing-masing dengan query count() terpisah. Selain itu, aplikasi menghitung jumlah konsultasi yang dilakukan pengguna melalui Konsultasi::where('idPeternak', …)->count(). Data-data ini, daftar ternak serta statistik, kemudian dikirim ke view pengguna agar bisa ditampilkan di antarmuka pengguna.

4) Halaman Tambah Informasi Umum

Tampilan ini menampilkan form pengisian yang mengahkan admin untuk melakukan proses penambahan data tentang informasi umum.



Gambar 4. 34 Tambah Informasi Umum

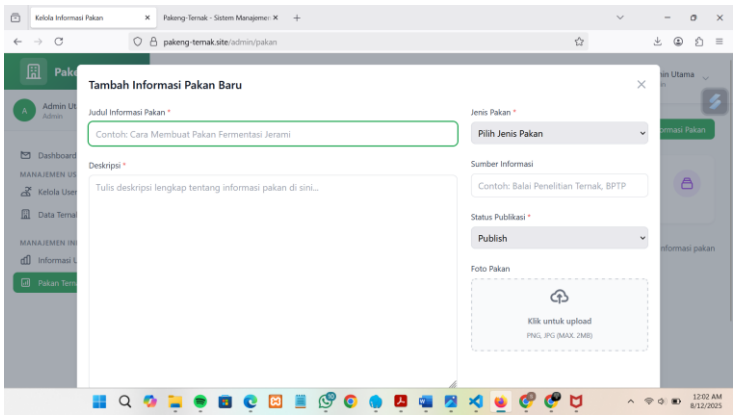
Berikut scrip tambah informasi umum:

```
$informasi = Informasi::create([
                'judul' => $request->input('judul'),
                'deskripsi' => $request->input('deskripsi'),
                'tanggal_kegiatan' => $request-
>input('tanggal_kegiatan'),
                'lokasi' => $request->input('lokasi'),
                'kategori' => $request->input('kategori'),
                'idPenyuluh' => $request->input('idPenyuluh'),
                'foto' => $fotoPath,
            ]);
            if ($request->ajax() || $request->wantsJson()) {
                return response()->json([
                    'success' => true,
                    'message' => 'Informasi berhasil disimpan',
                    'data' => $informasi
                ]);
            }
            return redirect()->route('admin.informasi.index')-
>with('success', 'Informasi berhasil disimpan.');
```

Dengan sintaks tersebut, Laravel menggunakan metode create() dari Eloquent untuk langsung memasukkan data baru ke database dengan sekali baris kode. Anda hanya perlu memastikan model Informasi sudah mengizinkan "mass assignment" untuk kolom-kolom tersebut melalui $fillable atau $guarded agar Laravel tidak menolak data—ini demi mencegah risiko keamanan seperti kolom tak terduga diubah saat pengguna mengirim permintaan HTTP

5) Halaman Tambah Informasi Pakan

Tampilan ini menampilkan form pengisian daftar informasi pakan, guna untuk melakukan pengisian dalam proses penambahan informasi pakan.



Gambar 4. 35 Halaman Tambah Informasi Pakan

Berikut tampilan scrip tambah informasi pakan:

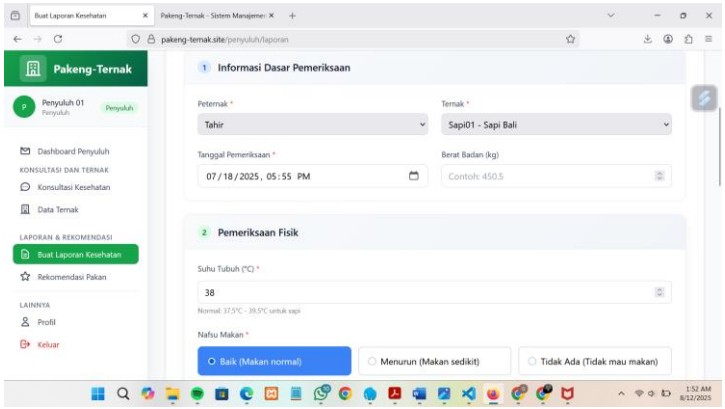
```
            $pakan = InformasiPakan::create([
                'judul' => $request->input('judul'),
                'deskripsi' => $request->input('deskripsi'),
                'foto' => $fotoPath,
                'sumber' => $request->input('sumber'),
                'jenis_pakan' => $request->input('jenis_pakan'),
                'is_published' => $request->input('is_published'),
            ]);
            if ($request->ajax() || $request->wantsJson()) {
                return response()->json([
                    'success' => true,
                    'message' => 'Informasi berhasil disimpan',
                    'data' => $pakan
                ]);
            }
              return redirect()->route('admin.pakan')->with('success',
'Pakan berhasil disimpan.');
```

Kode tersebut menggunakan InformasiPakan::create([...]) untuk menyimpan data baru sekaligus ke database dalam satu baris—ini adalah fitur mass assignment di Laravel yang memuat data instan ke atribut-atribut model. Pastikan model mengizinkan kolom-kolom tersebut melalui $fillable atau $guarded, agar operasi ini aman dan berhasil LaravelDEV Community. Setelah penyimpanan berhasil, kode memeriksa apakah permintaan berasal dari AJAX ($request->ajax()) atau jika klien mengharapkan respons JSON ($request->wantsJson()). Jika ya, maka fungsi langsung mengembalikan respons dalam format JSON dengan status sukses, pesan, dan data entitas yang baru dibuat. Kalau tidak, alih-alih JSON, pengguna diarahkan ke rute 'admin.pakan' dengan pesan sukses melalui sistem flash message.

5. Halaman Penyuluh

1) Halaman Tambah Laporan Kesehatan

Tampilan ini menampilkan halaman laporan, sistem akan menediakan form kepada penyuluh untuk melengkapi proses penambahan laporan kesehatan yang baru.



Gambar 4. 36 Halaman Laporan Kesehatan

Berikut script membuat laporan kesehatan:

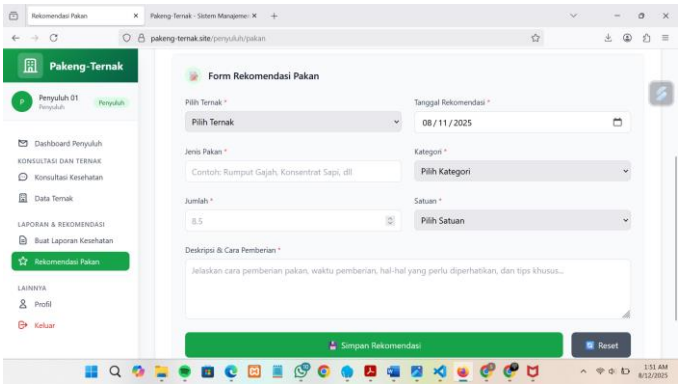
```
$laporan = Laporan::create([
                'idPenyuluh' => $currentUserId,
                'idPeternak' => $validated['idPeternak'],
                'idTernak' => $validated['idTernak'],
                                        'tanggal_pemeriksaan'     =>
Carbon::parse($validated['tanggal_pemeriksaan']),
                'berat_badan' => $validated['berat_badan'],
                'suhu_tubuh' => $validated['suhu_tubuh'],
                'nafsu_makan' => $validated['nafsu_makan'],
                'pernafasan' => $validated['pernafasan'],
                'kulit_bulu' => $validated['kulit_bulu'],
                'mata_hidung' => $validated['mata_hidung'],
                'feses' => $validated['feses'],
                'aktivitas' => $validated['aktivitas'],
                'tindakan' => $validated['tindakan'],
                'rekomendasi' => $validated['rekomendasi'],
                'status_kesehatan' => $kondisiKesehatan,
            ]);
            Log::info('Laporan created successfully', [
                'laporan_id' => $laporan->id,
                'laporan' => $laporan->toArray()
            ]);
            // Log aktivitas
            $this->logActivity('create_health_report', [
                'laporan_id' => $laporan->id,
                'ternak_id' => $validated['idTernak'],
                'peternak_id' => $validated['idPeternak'],
                'status_kesehatan' => $kondisiKesehatan,
            ]);
            // Redirect dengan pesan sukses
            return redirect()->route('penyuluh.laporan')
              ->with('success', 'Laporan kesehatan berhasil disimpan!')
                ->with('laporan_id', $laporan->id);
        } catch (\Illuminate\Validation\ValidationException $e) {
```

```
            Log::error('Validation failed', [
                'errors' => $e->errors(),
                'request_data' => $request->all()
            ]);
```

Kode ini digunakan untuk menyimpan data laporan pemeriksaan kesehatan ternak. Pertama, data divalidasi dan kemudian disimpan ke database menggunakan Laporan::create([...]), yang memuat berbagai informasi seperti ID penyuluh, ID peternak, ID ternak, tanggal pemeriksaan (yang diformat dengan Carbon::parse()), serta kondisi fisik dan rekomendasi medis lainnya. Setelah berhasil disimpan, Laravel mencatat log informasi tentang laporan yang dibuat menggunakan Log::info(), yang berguna untuk keperluan debugging atau audit. Kemudian fungsi log*Activity*() dipanggil untuk mencatat aktivitas pengguna terkait pembuatan laporan ini. Jika semua berjalan lancar, pengguna akan diarahkan ke halaman laporan penyuluh dengan pesan sukses. Namun, jika validasi gagal, sistem akan menangkap pengecualian (ValidationException) dan mencatat kesalahan serta data permintaan ke dalam log menggunakan Log::error()

2) Tampilan Halaman Tambah Rekomendasi Pakan

Tampilan ini merupakan tampilan halaman rekomendasi pakan, penyuluh bisa memberikan rekomendasi pada ternak yang telah ia pilih yang nantinya data akan terhubung dengan halaman rekomendasi pakan pada peternak.



Gambar 4. 37 Halaman Rekomendasi Pakan

Berikut script rekomendasi pakan:

```
        try {
            $pakan = Pakan::create([
                'idTernak' => $request->idTernak,
                'idPenyuluh' => Auth::id(),
                'tanggalRekomendasi' => $request->tanggalRekomendasi,
                'jenisPakan' => $request->jenisPakan,
                'kategori' => $request->kategori,
                'jumlah' => $request->jumlah,
                'satuan' => $request->satuan,
                'deskripsi' => $request->deskripsi,
            ]);
            return redirect()->route('penyuluh.pakan')
                        ->with('success', 'Rekomendasi pakan berhasil
disimpan.');
        } catch (\Exception $e) {
            return redirect()->back()
                ->withInput()
                   ->with('error', 'Gagal menyimpan rekomendasi pakan.
Silakan coba lagi.');
        }
    }
```

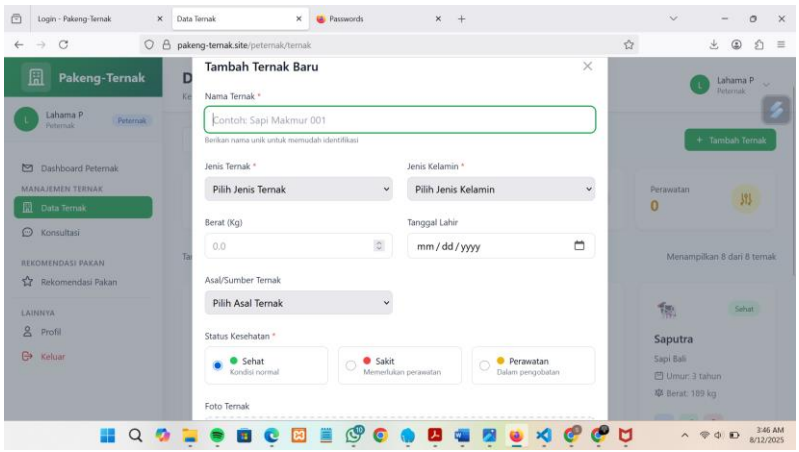
Dalam fungsi tersebut, kode berusaha menyimpan data rekomendasi pakan baru menggunakan Pakan::create([...]) dengan data seperti idTernak, jenisPakan,

dan lainnya, lalu langsung diarahkan (redirect) kembali ke halaman 'penyuluh.pakan' dengan pesan sukses jika penyimpanan berhasil. Namun, jika terjadi kesalahan atau pengecualian (seperti duplikat, error database, dll.), blok catch akan menangkapnya dan mengarahkan ulang pengguna ke halaman sebelumnya (back()), menjaga input pengguna (withInput()) dan menyertakan pesan error agar bisa dicoba ulang.

6. Peternak

1) Tampilan Halaman Tambah Data Ternak

Tampilan ini merupakan tampilan fom pengisian data data pribadi



Gambar 4. 38 Halaman Tambah Data Ternak

```
            'asal'           => $validated['asal'],
            'keterangan'     => $validated['keterangan'],
            'fotoTernak'     => $fotoPath,
        ]);
        if ($request->ajax()) {
            return response()->json([
                'success' => true,
                'message' => 'Data ternak berhasil disimpan.',
                'data' => $ternak
            ]);
        }
          return  redirect()->route('peternak.ternak')->with('success',
'Data ternak berhasil disimpan.');
    }
```

Berikut script tampilan tambah data ternak:

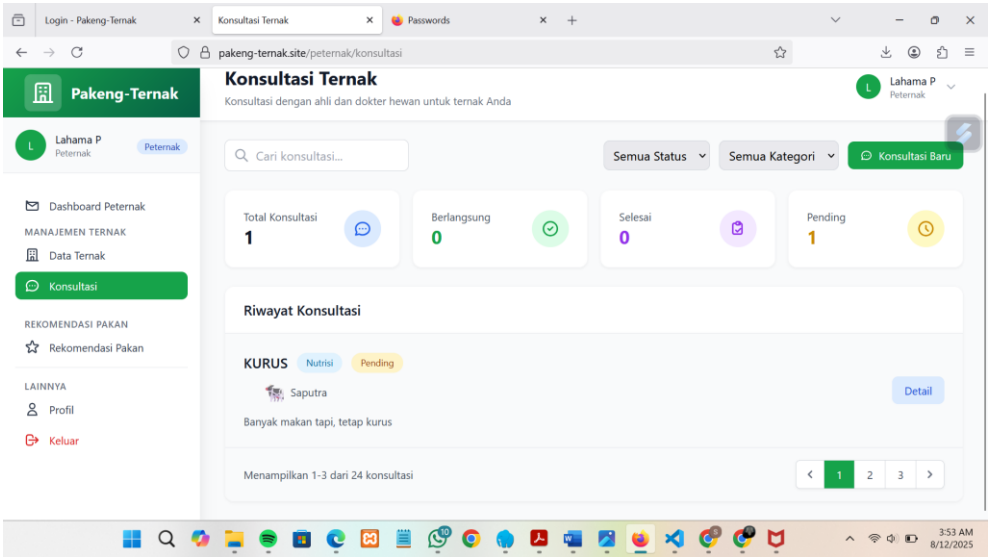
Dalam bagian kode ini, aplikasi menyimpan data Ternak baru ke dalam database menggunakan metode create() dari Eloquent, yang secara otomatis membuat dan menyimpan objek model sekaligus. Data yang disimpan mencakup berbagai informasi seperti pemilik, nama, jenis kelamin, tanggal lahir, berat, status, asal, keterangan, dan path foto ternak. Setelah berhasil, pemeriksaan dilakukan—jika permintaan dilakukan via AJAX ($request->ajax()), maka fungsi langsung mengembalikan respons JSON berisi status sukses, pesan, dan data ternak; jika

```
        $ternak = Ternak::create([
            'idPemilik'      => $user->idUser,
            'namaTernak'     => $validated['namaTernak'],
            'jenis'          => $validated['jenis'],
            'jenis_kelamin'  => $validated['jenis_kelamin'],
            'tanggalLahir'   => $validated['tanggalLahir'],
            'berat'          => $validated['berat'],
            'status'         => $validated['status'],
```

tidak, aplikasi mengarahkan pengguna kembali ke rute 'peternak.ternak' dengan pesan sukses menggunakan flash notification.

2) Tampilan Halaman Tambah Konsultasi Baru

Tampilan ini merupakan tampilan selanjutnya setelah memilih *button* konsultasi baru, sostem akan menampilkan form pengisian untuk melengkapi data data konsultasi yang kemudian nantinya akan diterima oleh penyuluh.



Gambar 4. 39 Halaman Konsultasi Baru

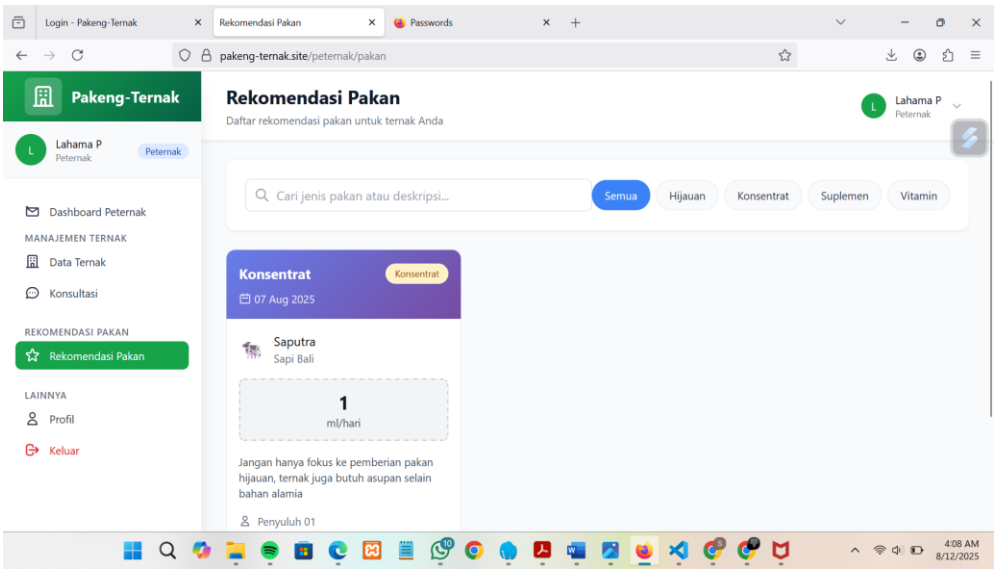
Berikut script tampilan Konsultasi Baru

Pada kode tersebut, aplikasi menyimpan data konsultasi baru menggunakan Konsultasi::create([...]), mengambil nilai dari validasi request seperti idPeternak, idPenyuluh, judul_konsultasi, dan lainnya, lalu menyimpannya ke database sekaligus. Setelah itu, jika permintaan datang via AJAX ($request->ajax()), maka fungsi akan langsung mengembalikan respons dalam format JSON yang berisi status sukses, pesan, dan objek konsultasi yang baru dibuat; bila tidak, aplikasi akan mengarahkan ulang ke halaman peternak.konsultasi dengan pesan sukses lewat flash notification.

3) Tampilan Halaman Rekomendasi Pakan

```
        $konsultasi = Konsultasi::create([
            'idPeternak' => $validated['idPeternak'],
            'idPenyuluh' => $validated['idPenyuluh'],
            'idTernak' => $validated['idTernak'],
            'judul_konsultasi' => $validated['judul_konsultasi'],
            'kategori' => $validated['kategori'],
            'status' => $validated['status'],
            'deskripsi' => $validated['deskripsi'],
            'foto_ternak' => $fotoPath,
        ]);
        if ($request->ajax()) {
            return response()->json([
                'success' => true,
                'message' => 'Data Konsultasi berhasil disimpan.',
                'data' => $konsultasi
            ]);
             }        return  redirect()->route('peternak.konsultasi')->with('success', 'Data konsultasi berhasil disimpan.');
    }
```

Tampilan ini merupakan tampilan yang memuat data rekomendasi kepada tenak terpilih yang telah ditentukan oleh penyuluh



Gambar 4. 40 Halaman Rekomendasi Pakan

Berikut script tampilan rekomendasi pakan:

```
    public function pakan(Request $request)
    {
        $query = Pakan::with(['ternak', 'penyuluh']);
        if ($request->filled('kategori')) {
            $query->where('kategori', $request->kategori);
        }
        if ($request->filled('search')) {
            $search = $request->search;
            $query->where(function ($q) use ($search) {
                $q->where('jenisPakan', 'like', '%' . $search .
'%')
                    ->orWhere('deskripsi', 'like', '%' . $search
. '%')
                    ->orWhereHas('ternak', function ($query) use
($search) {
                        $query->where('nama_ternak', 'like', '%'
. $search . '%');
                    });
            });
        }
        $pakans = $query->orderBy('tanggalRekomendasi', 'desc')-
>paginate(12);
        return view('peternak.pakan', compact('pakans'));
    }
}
```

Metode pakan(Request $request) memungkinkan pengguna (peternak) untuk melihat daftar rekomendasi pakan dengan fitur pencarian dan penyaringan (filter). Pertama, query dibangun menggunakan model Pakan dan memuat relasi ternak serta penyuluh. Jika permintaan berisi parameter kategori, query akan difilter sesuai kategori yang dipilih. Jika terdapat parameter search, maka query akan mencari kecocokan pada kolom jenisPakan, deskripsi, atau nama ternak melalui

relasi ternak menggunakan whereHas(). Hasil query kemudian diurutkan berdasarkan tanggalRekomendasi secara menurun, dan hasilnya ditampilkan dengan pagination—menampilkan 12 item per halaman. Data ini kemudian dikirim ke view peternak.pakan agar bisa ditampilkan dengan navigasi halaman yang sesuai.

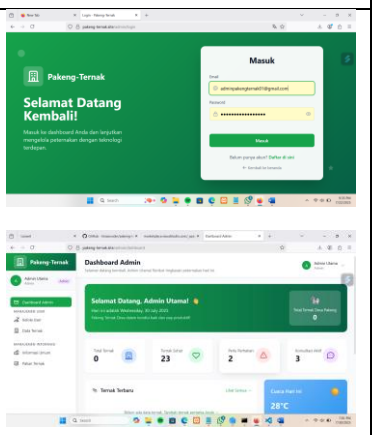
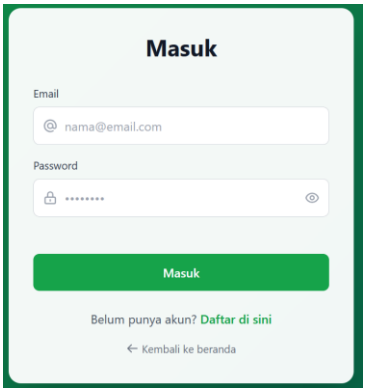
E. Pengujian Sistem

1. *Balack Box Testing*

a. Admin

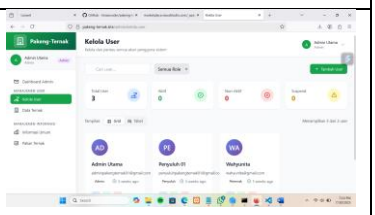
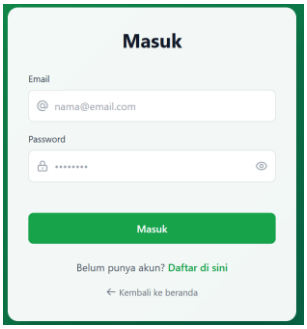
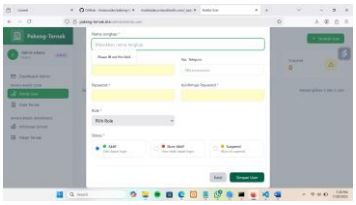
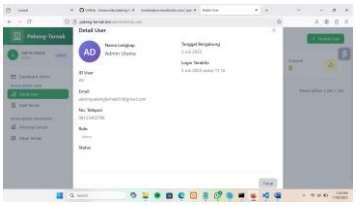
1) *Balack Box Testing Login*

Tabel 4. 3 *Balack Box TestingLogin*

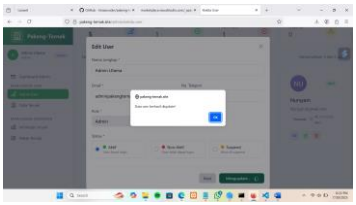
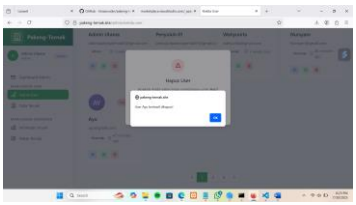
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
admin memasukkan *email* dan *password* yang sesuai	✓	Ketika alamat *email* dan *password* sesui maka berhasil melakukan proses *Login*, kemudian sistem akan menampilkan laman *admin*	
Admin memasukkan *email* atau *password* yang salah	✓	Berhasil, sistem akan mengulang proses dan meminta untuk pengisian kembali	

2) *Balack Box Testing* Kelola *User*

Tabel 4. 4 *Balack Box Testing* Kelola *User*

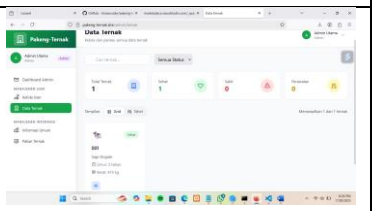
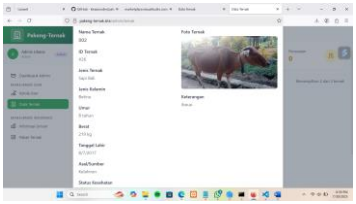
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
admin memilih menu kelola *User*	✓	Berhasil, sistem akan menampilkan halaman daftar *User*	
Admin menambahkan *User* baru	✓	Berhasil, sistem akan menampilkan pesan *" User Berhasil Ditambahkan"*	
Admin tidak mengisi form	✓	Berhasil, sistem akan menampilkan pesan peringatan untuk mengisi form yang kosong	
Admin memilih liat detail	✓	Berhasil, sistem akan menampilkan detail data *User* yang dipilih	

Lanjutan Tabel 4.4

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Admin memilih ubah data	✓	Berhasil, sistem akan menampilkan pesan *"Data User Berhasil Diubdate "*	
Admin memilih hapus data *User*	✓	Berhasil, sistem akan menampilkan pesan *" User Berhasil Dihapus "*	

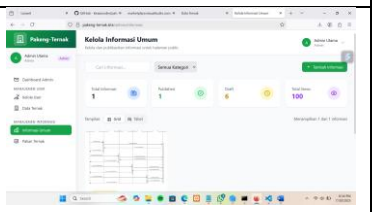
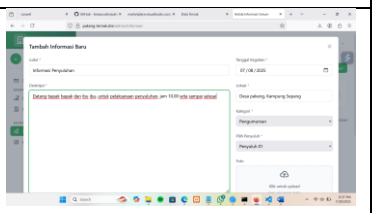
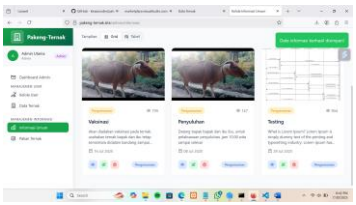
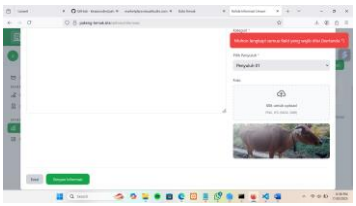
3) *Balack Box Testing* Data Ternak

Tabel 4. 5 *Balack Box Testing* Data Ternak

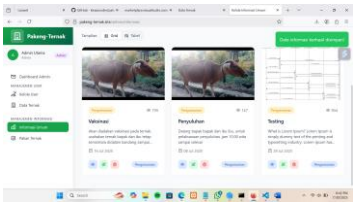
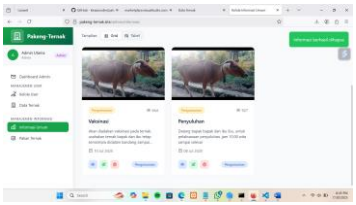
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Admin Memilih menu data ternak	✓	Berhasil, sistem menampilkan halaman data ternak	
Admin memilih detail pada data ternak	✓	Berhasil, sistem akan menampilkan detail data ternak	

4) *Balack Box Testing* Informasi Umum

Tabel 4. 6 *Balack Box Testing* Informasi Umum

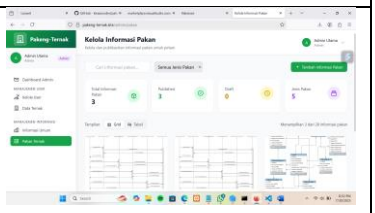
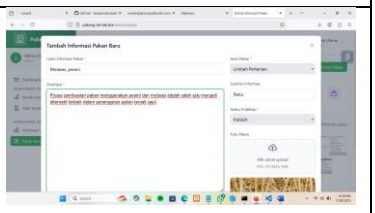
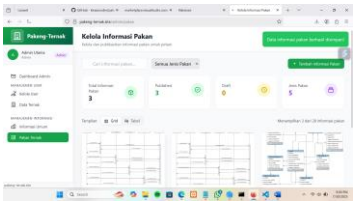
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
admin memilih menu informasi umum	✓	Berhasil, sistem akan menampilkan halaman informasi umum	
Admin memilih tambah informasi	✓	Berhasil, sistem akan menampilkan pesan *" Data Informasi Berhasil Disimpan"*	 
Admin tidak mengisi form	✓	Berhasil, sistem akan menampilkan pesan peringatan untuk mengisi form yang kosong	
Admin memilih liat detail	✓	Berhasil, sistem akan menampilkan detail informasi	

Lanjutan Tabel 4.6

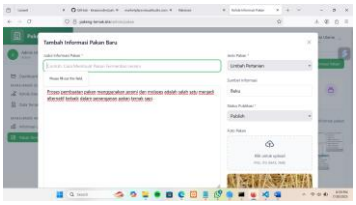
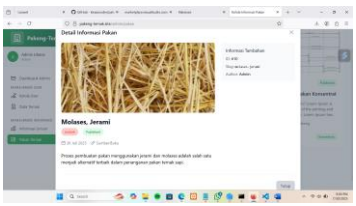
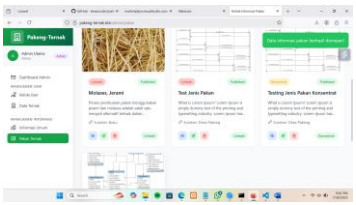
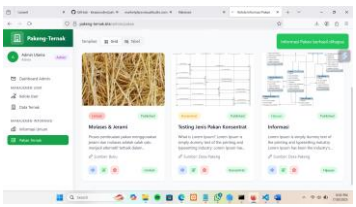
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Admin memilih ubah data	✓	Berhasil, sistem akan menampilkan pesan *"Data Informasi Berhasil Disimpan "*	
Admin memilih hapus data *User*	✓	Berhasil, sistem akan menampilkan pesan *" Informasi Berhasil Dihapus"*	

5) *Balack Box Testing* Pakan Ternak

Tabel 4. 7 *Balack Box Testing* Pakan Ternak

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
admin memilih menu informasi umum	✓	Berhasil, sistem akan menampilkan halaman Pakan Ternak	
Admin memilih tambah informasi	✓	Berhasil, sistem akan menampilkan pesan *"Data Pakan Berhasil Disimpan"*	 

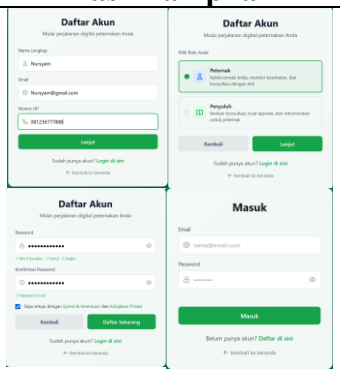
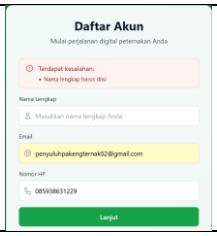
Lanjutan Tabel 4.7

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Admin tidak mengisi form	✓	Berhasil, sistem akan menampilkan pesan peringatan untuk mengisi form yang kosong	
Admin memilih liat detail	✓	Berhasil, sistem akan menampilkan detail informasi	
Admin memilih ubah data	✓	Berhasil, sistem akan menampilkan pesan *"Data pakan Berhasil Disimpan "*	
Admin memilih hapus data *User*	✓	Berhasil, sistem akan menampilkan pesan *" Informasi Berhasil Dihapus"*	

b. Penyuluh

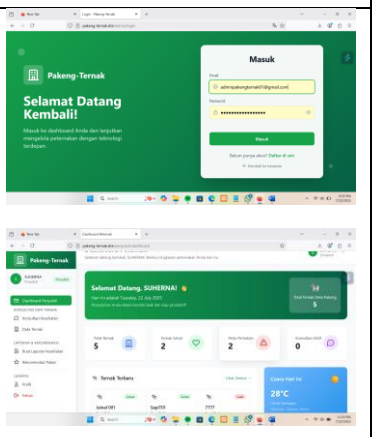
1) *Balack Box Testing* Daftar Akun

Tabel 4. 8 *Balack Box Testing* Daftar Akun

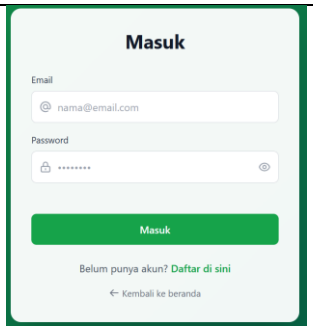
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh Melakukan Pendaftaran	✓	Berhasil, sistem menampilkan pesan *"Sukses! Pendaftaran berhasil ditambahkan."*	
Penyuluh tidak mengisi salah satu pertanyaan	✓	Berhasil, sistem akan menampil-kan pesan Kesalahan untuk field yang wajib diisi	

2) *Balack Box Testing Login*

Tabel 4. 9 *Balack Box Testing Login*

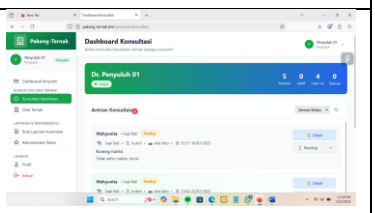
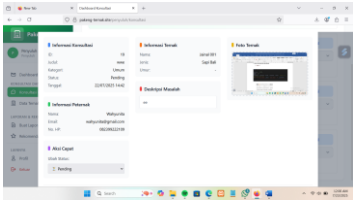
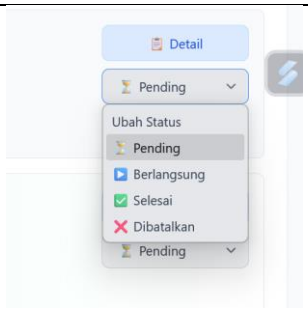
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh memasukkan *email* dan *password* yang sesuai	✓	Ketika alamat *email* dan *password* sesui maka berhasil melakukan proses *Login*, kemudian sistem akan menampilkan laman Penyuluh	

Lanjutan Tabel 4.9

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh memasukkan *email* atau *password* yang salah	✓	Berhasil, sistem akan mengulang proses dan meminta untuk pengisian kembali	

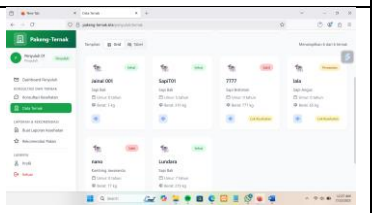
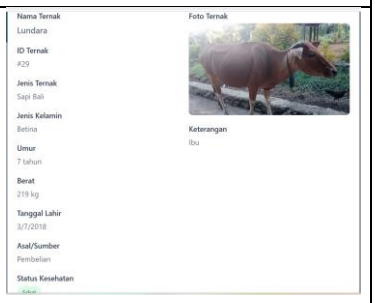
3) *Balack Box Testing* Menu Konsultasi

Tabel 4. 10 *Balack Box Testing* Menu Konsultasi

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh mengklik menu konsultasi	✓	Berhasil, sistem akan menampilkan daftar konsultasi	
Penyuluh mengklik detail pada salah satu daftar konsultasi	✓	Berhasil, sistem akan menampilkan detail konsultasi	
Penyuluh mengubah status konsultasi	✓	Berhasil, sistem akan mengubah status konsultasi	

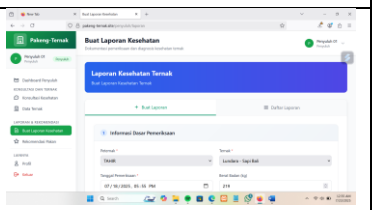
4) *Balack Box Testing* Menu Data Ternak

Tabel 4. 11 *Balack Box Testing* Data Ternak

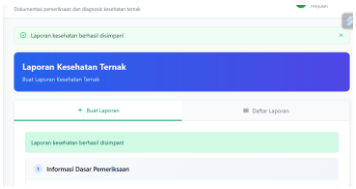
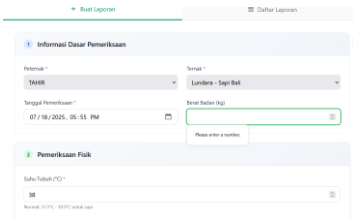
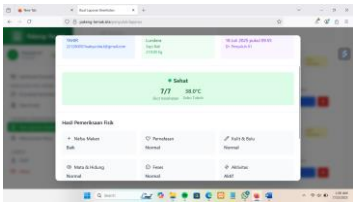
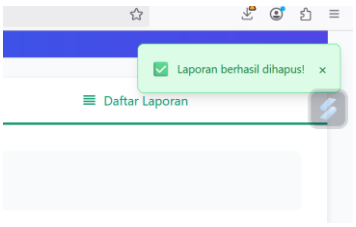
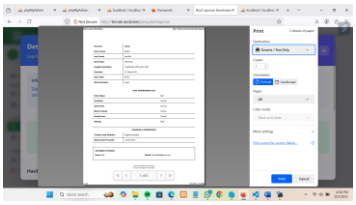
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh mengklik menu Data Ternak	✓	Berhasil, sistem akan menampilkan Daftar Ternak	
Penyuluh mengklik detail pada salah satu data ternak	✓	Berhasil, sistem akan menampilkan detail data ternak	

5) *Balack Box Testing* Menu Laporan Kesehatan

Tabel 4. 12 *Balack Box Testing* Laporan Kesehatan

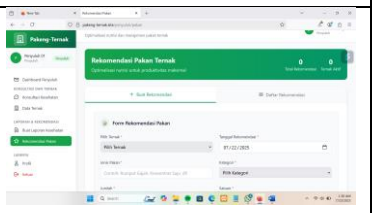
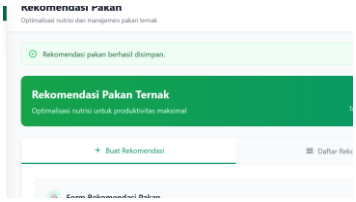
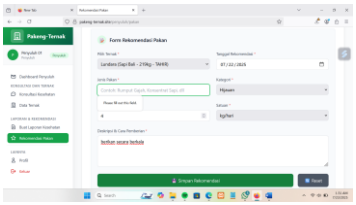
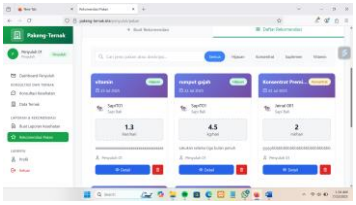
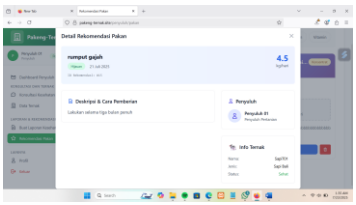
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh mengklik laporan kesehatan	✓	Berhasil, sistem akan menampilkan form laporan kesehatan	

Lanjutan Tabel 4.12

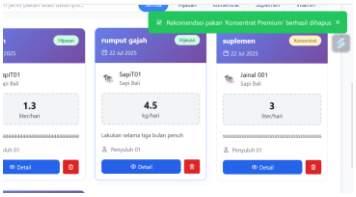
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh Penyimpan Laporan	✓	Berhasil, sistem akan menampilkan pesan *" Laporan kesehatan berhasil disimpan"*	
Penyuluh tidak mengisi *field*	✓	Berhasil, sistem akan menampilkan pesan untuk mengisi form yang kosong	
Penyuluh Memilih detail	✓	Berhasil, sistem berhasil menampilkan detail laporan	
Penyuluh Menghapus data laporan	✓	Berhasil, laporan terhapus dengan pesan *"Laporan berhasil dihapus"*	
Penyuluh memilih cetak	✓	Berhasil, sistem menampilkan laman pencetakan	

6) *Balack Box Testing* Rekomendasi Pakan

Tabel 4. 13 *Balack Box Testing* Rekomendasi Pakan

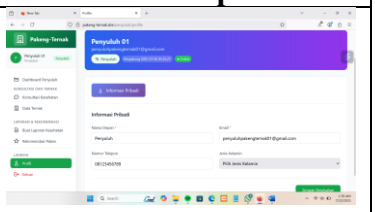
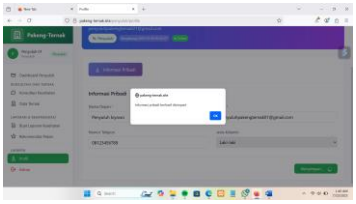
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh mengklik menu rekomendasi pakan	✓	Berhasil, sistem akan menampilkan form rekomendasi pakan	
Penyuluh mengklik detail pada salah satu data ternak	✓	Berhasil, sistem akan menampilkan pesan *" Rekomendasi pakan berhasil disimpan"*	
Penyuluh tidak mengisi *field*	✓	Berhasil, sistem akan menampilkan pesan untuk mengisi form yang kosong	
Penyuluh Mengklik daftar rekomendasi pakan	✓	Berhasil, sistem berhasil menampilkan daftar rekomendasi pakan	
Penyuluh Memilih detail	✓	Berhasil, sistem berhasil menampilkan detail pakan	

Lanjutan Tabel 4.13

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh Menghapus data laporan	✓	Berhasil, laporan terhapus dengan pesan *"Rekomendasi pakan 'nama' berhasil dihapus"*	

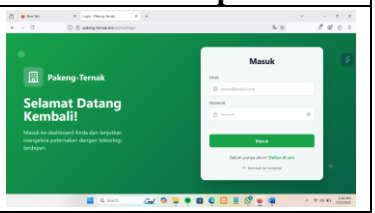
7) *Balack Box Testing* Menu Profil

Tabel 4. 14 *Balack Box Testing* Menu Profil

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh Memilih menu profil	✓	Berhasil, sistem menampilkan halaman profil pribadi	
Penyuluh menyimpan perubahan	✓	Berhasil, sistem akan menampil-kan pesan *" informasi pribadi berhasil disimpan"*	

8) *Balack Box Testing* Menu Logout

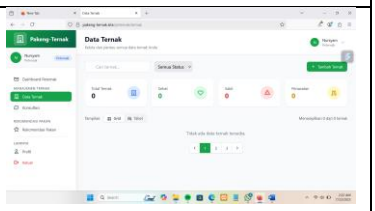
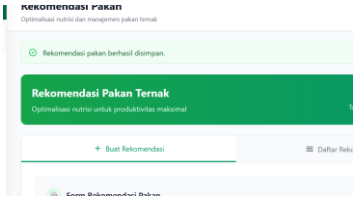
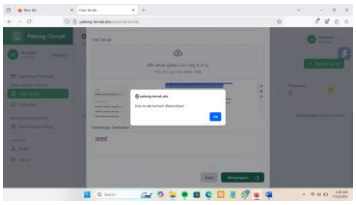
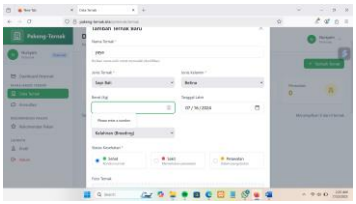
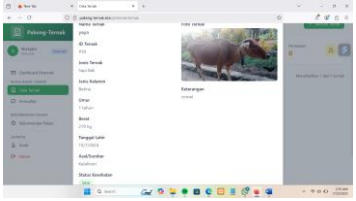
Tabel 4. 15 *Balack Box Testing* Logout

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Penyuluh memilih logout	✓	Berhasil, sistem akan menampilkan halaman *Login*	

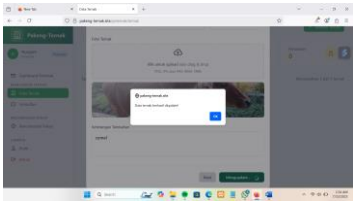
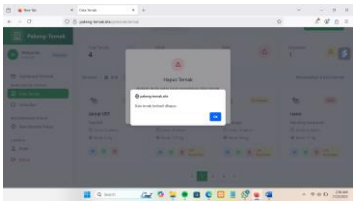
c. Peternak

1) *Balack Box Testing* Menu Data Ternak

Tabel 4. 16 *Balack Box Testing* Data Ternak

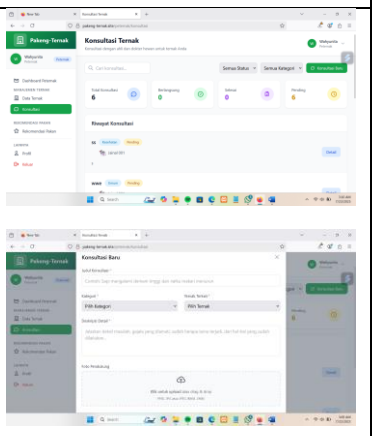
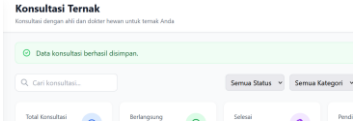
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak mengklik menu data ternak	✓	Berhasil, sistem akan menampilkan laman data ternak	
Peternak menambah data ternak	✓	Berhasil, sistem akan menampilkan pesan *" Rekomendasi pakan berhasil disimpan"*	
Peternak menyimpan data ternak	✓	Berhasil, sistem akan menampilkan pesan *"Data ternak berhasil ditambahkan!"*	
Peternak tidak mengisi *field*	✓	Berhasil, sistem akan menampilkan pesan untuk mengisi form yang kosong	
Peternak Memilih detail	✓	Berhasil, sistem berhasil menampilkan detail ternak	

Lanjutan Tabel 4.16

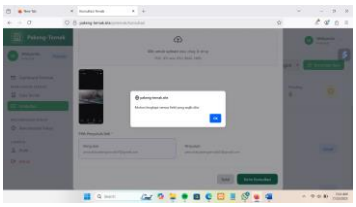
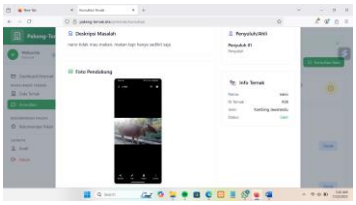
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak Mengedit data ternak	✓	Berhasil, sistem melakukan perubahan dengan pesan *"Data ternak berhasil diupdate"*	
Peternak Menghapus data data ternak	✓	Berhasil, data ternak terhapus dengan pesan *"Data ternak berhasil dihapus"*	

2) *Balack Box Testing* Menu Konsultasi

Tabel 4. 17 *Balack Box Testing* Konsultasi

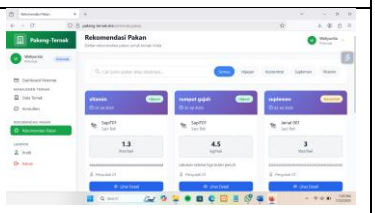
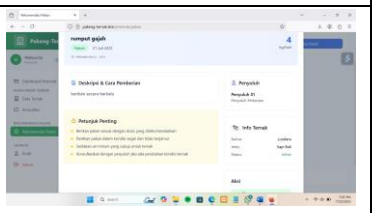
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak mengklik menu konsultasi dan konsultasi baru	✓	Berhasil, sistem akan menampilkan laman konsultasi dan form konsultasi	
Peternak mengirimkan konsultasi	✓	Berhasil, data terkirim dengan pesan "Data konsultasi berhasil disimpan"	

Lanjutan Tabel 4.17

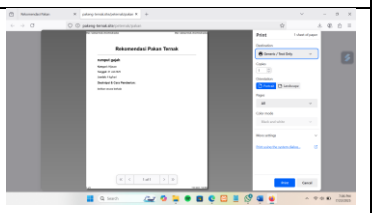
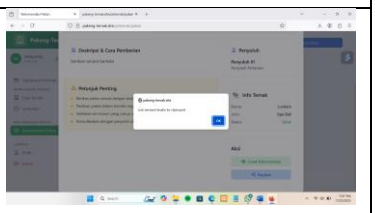
Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak mengubah status konsultasi	✓	Berhasil, sistem akan mengubah status konsultasi	
Peternak memilih detail	✓	Berhasil, sistem menampilkan detail konsultasi	

3) *Balack Box Testing* Menu Rekomendasi Pakan

Tabel 4. 18 *Balack Box Testing* Rekomendasi Pakan

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak mengklik menu rekomendasi pakan	✓	Berhasil, sistem akan menampilkan rekomendasi pakan	
Peternak memilih detail	✓	Berhasil, sistem menampilkan detail pakan	

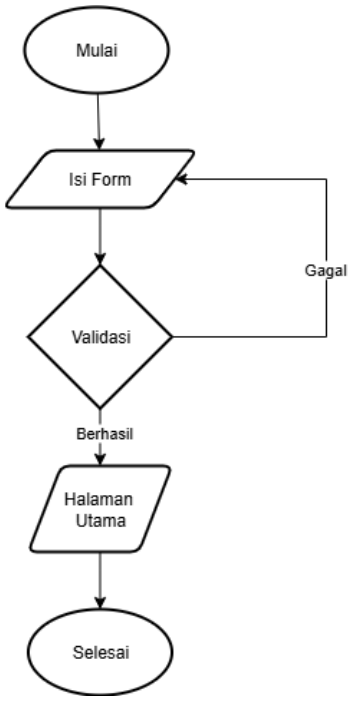
Lanjutan Tabel 4.18

Tes Faktor	Hasil	Keterangan	Hasil Tampilan
Peternak memilih cetak rekomendasi	✓	Berhasil, sistem akan menampilkan halaman pencetakan	
Peternak memilih bagikan	✓	Berhasil, sistem menampilkan pesan *"Link berhasil disalin ke clipboard!"*	

2. *White Box Testing*

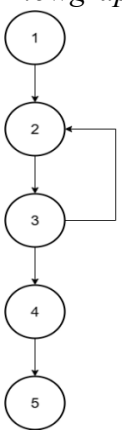
a. *White Box Testing Login* admin

1) *Flowchart*



Gambar 4. 41 *Whitebox Testing Login*

2) *Flowgraph*



Gambar 4. 42 *Whitebox Testing Login*

Berdasarkan gambar diatas, dilakukan perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* V (G) = E – N + 2

N (node) = 5

E (edge) = 5

P (predikat node) = 1

Penyelesaian : V (G) = E – N + 2

= 5-5+2

= 2

Predikat Node (N) = P + 1

= 1 + 1 = 2

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* diatas memiliki *Region* = 2

c) *Independent path* pada flowgraph diatas adalah :

Path 1 = 1-2-3-4-5

Path 2 = 1-2-3-2-3-4-5

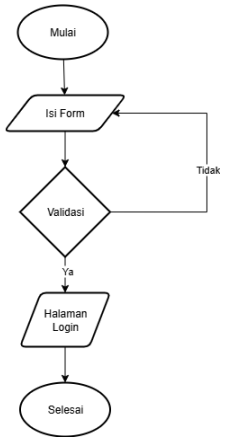
d) Grafik Matriks

Tabel 4. 19 Grafik matriks *Login*

	1	2	3	4	5	E - 1
1		1				1-1=0
2			1			1-1=0
3		1		1		2-1=1
4					1	1-1=0
5						0
Sum (E+1)						1+1=2

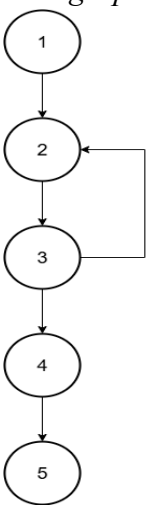
b. *White Box Testing* Daftar

1) *Flowchart*



Gambar 4. 43 *Whitebox Testing* Daftar

2) *Flowgraph*



Gambar 4. 44 *Whitebox Testing* Daftar

Berdasarkan tampilan gambar diatas, dapat dilakukan perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* V (G) = E – N + 2

   N (node) = 5

   E (edge) = 5

   P (predikat node) = 1

   Penyelesaian : V (G) = E – N + 2

   = 5-5+2

   = 2

   Predikat Node (N) = P + 1 = 1 + 1 = 2

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* diatas memiliki *Region* = 2

c) *Independent path* pada flowgraph diatas adalah :

   Path 1 = 1-2-3-4-5

   Path 2 = 1-2-3-2-3-4-5

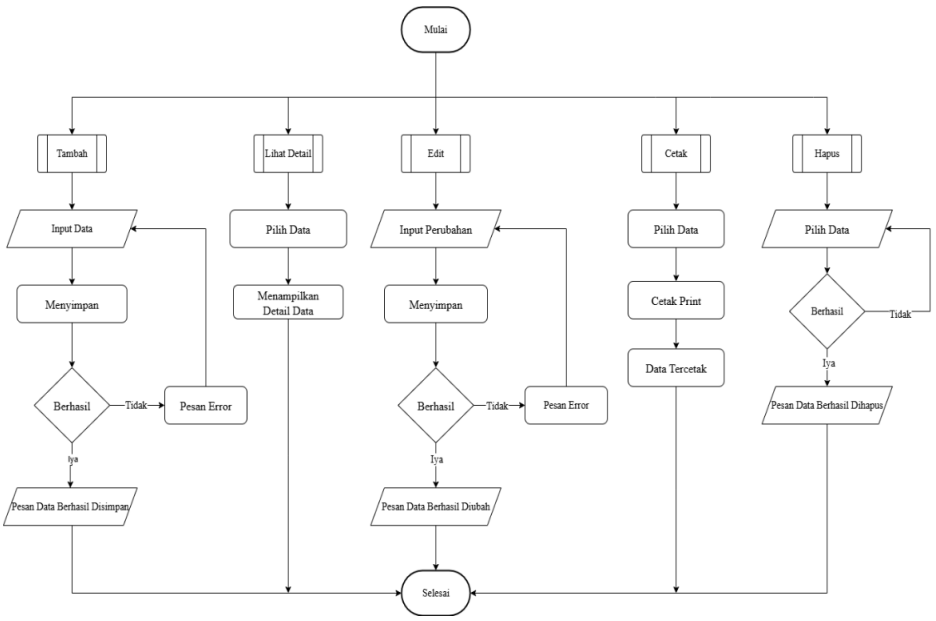
d) Grafik Matriks

Tabel 4. 20 Grafik Matriks Daftar

	1	2	3	4	5	E - 1
1		1				1-1=0
2			1			1-1=0
3		1		1		2-1=1
4					1	1-1=0
5						0
Sum (E+1)						1+1=2

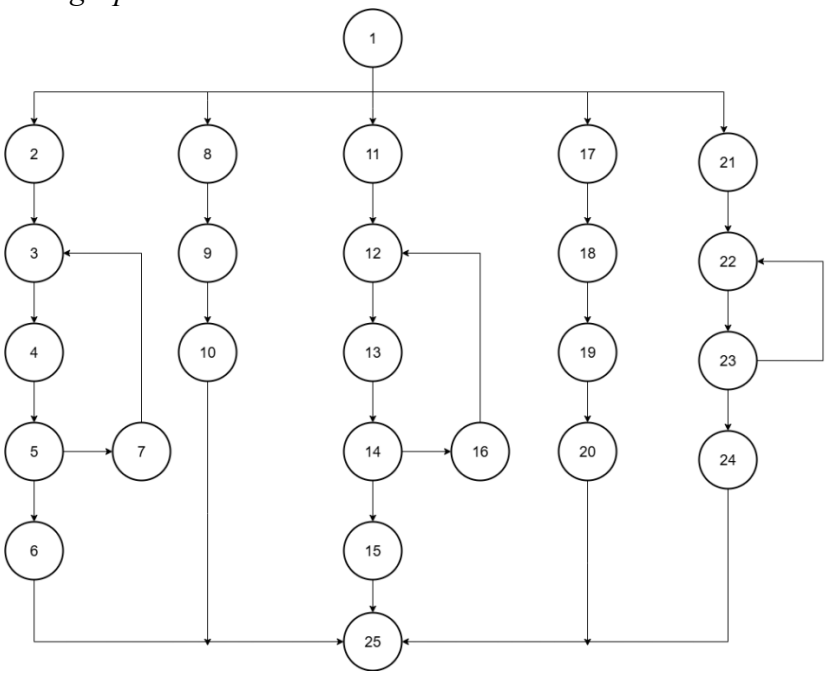
c. *White Box Testing* Tambah, Lihat Detail, Edit, Cetak dan Hapus

1) *Flowchart*



Gambar 4. *45* *Whitebox Testing* Tambah, Lihat Detail, Edit, Cetak, dan Hapus

2) *Flowgraph*



Gambar 4. 46 *Whitebox Testing* Tambah, Lihat Detail, Edit, Cetak, dan Hapus

Berdasarkan gambar diatas, dilakukan perhitungan sebagai berikut :

a) Menghitung *Cyclomatic Complexity* V (G) = E – N + 2

   N (node) = 25

   E (edge) = 31

   P (predikat node) = 3

   Penyelesaian : V (G) = E – N + 2

   = 31-25+2

   = 8

b) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* diatas memiliki *Region* = 8

c) *Independent path* pada flowgraph diatas adalah :

   Path 1 = 1-2-3-4-5-6-25

   Path 2 = 1-2-3-4-5-7-3

   Path 3 = 1-8-9-10-25

   Path 4 = 1-11-12-13-14-15-25

   Path 5 = 1-11-12-13-14-12

   Path 6 = 1-17-18-19-20-25

   Path 7 = 1-21-22-23-24-25

   Path 8 = 1-21-22-23-22

d) Grafik MatrikS.

Tabel 4. 21 Grafik Matriks Tambah, Lihat Detail, Edit, Cetak, dan Hapus

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	E-1
1		1						1			1						1				1					5-1=4
2			1																							1-0=0
3				1																						1-1=0
4					1																					1-1=0
5						1	1																			2-1=1
6																									1	1-1=0
7			1																							1-1=0
8									1																	1-1=0
9										1															1	1-1=0
10																										1-1=0
11												1														1-1=0
12													1													1-1=0
13														1												1-1=0
14															1	1										2-1=1
15																									1	1-1=0
16											1															1-1=0
17																		1								1-1=0
18																			1							1-1=0
19																				1						1-1=0
20																									1	1-1=0
21																						1				1-1=0
22																							1			1-1=0
23																							1	1		2-1=1
24																									1	1-1=0
25																										0
																										Sum(E+1) 7+1=8

3. Pengujian Kegunaan

Pengujian kegunaan dilakukan untuk menilai sejauh mana sistem informasi yang telah dikembangkan dapat digunakan dengan mudah, memberikan manfaat, dan memenuhi kebutuhan pengguna, khususnya peternak di Desa Pakeng, Kecamatan Lembang.

Metode yang digunakan adalah penyebaran kuesioner berbasis skala Likert (1–5), dengan 15 pernyataan tertutup dan 4 pertanyaan terbuka. Kuesioner disebarkan kepada peternak yang telah mencoba sistem, baik secara langsung maupun melalui simulasi penggunaan. Skala nilai 1 hingga 5 diartikan sebagai berikut:

a. 1 = Sangat Tidak Setuju
b. 2 = Tidak Setuju
c. 3 = Netral
d. 4 = Setuju
e. 5 = Sangat Setuju

Aspek yang dinilai dalam pengujian ini meliputi:

a. Kemudahan penggunaan sistem
b. Kejelasan tampilan dan informasi
c. Kecepatan dan kemudahan dalam konsultasi
d. Manfaat sistem terhadap aktivitas beternak
e. Tingkat kepuasan pengguna

Tabel 4. 22 Tabel Responden

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
R1	5	4	5	4	5	4	4	4	4	4	4	4	4	4	5
R2	4	4	5	4	5	4	4	4	5	4	4	4	4	4	4
R3	4	4	4	4	5	4	4	4	4	4	4	4	4	4	5
R4	4	3	4	4	4	4	4	4	5	4	4	4	3	4	4
R5	3	4	5	4	4	3	4	4	5	4	4	4	4	3	4
R6	4	4	4	4	4	4	4	4	4	4	4	4	4	4	5
R7	4	4	4	4	4	4	5	4	5	5	4	4	4	4	4
R8	4	3	4	5	4	5	4	4	4	4	4	4	4	5	4
R9	4	4	5	4	4	4	4	4	5	4	5	3	4	4	4
R10	4	4	5	5	3	4	4	3	4	5	4	4	4	4	5
R11	4	4	4	4	4	4	3	4	4	4	5	4	5	4	4
R12	4	4	4	4	4	4	4	4	4	4	5	4	4	3	4
R13	5	5	5	5	5	4	4	4	4	5	4	4	4	4	5
R14	4	4	4	4	5	4	4	4	4	4	4	5	4	4	4
R15	4	4	4	4	5	4	4	4	4	4	4	5	4	4	4

Berdasarkan hasil kuesioner yang diisi oleh 15 responden, dapat dilakukan analisis terhadap tingkat kepuasan pengguna dan efektivitas sistem dalam mendukung peternak. Kuesioner ini mencakup berbagai aspek terkait kemudahan penggunaan, manfaat sistem, serta kecepatan respon dari penyuluh. Berikut adalah hasil analisis secara rinci:

a. Kemudahan Penggunaan (P1):

   Mayoritas responden memberikan nilai 4 dan 5 pada pernyataan "Sistem mudah digunakan oleh saya sebagai peternak." Hal ini menunjukkan bahwa sistem dianggap cukup mudah diakses dan digunakan oleh peternak tanpa

banyak kesulitan. Nilai ini mencerminkan antarmuka sistem yang ramah pengguna dan mudah dipahami.

b. Tampilan Sistem yang Jelas (P2): Sebagian besar responden juga memberikan nilai 4 dan 5 pada pernyataan ini, yang berarti tampilan sistem dianggap cukup jelas dan tidak membingungkan. Pengguna merasa nyaman dengan navigasi dan desain antarmuka yang disediakan, yang mempermudah mereka dalam menggunakan sistem.

c. Pemahaman Tanpa Bantuan Orang Lain (P3): Penilaian pada dimensi ini menunjukkan bahwa sebagian besar responden merasa dapat memahami cara penggunaan sistem tanpa bantuan eksternal. Sebagian besar memberikan nilai 4, yang menandakan bahwa meskipun sistem intuitif, ada beberapa pengguna yang merasa perlu sedikit bantuan awal.

d. Bantuan dalam Pemberian Pakan yang Benar (P4): Berdasarkan hasil evaluasi, responden menilai bahwa sistem cukup efektif dalam memberikan informasi tentang cara pemberian pakan yang benar. Dengan skor rata-rata 4 dan 5, ini menunjukkan bahwa peternak merasa informasi yang diberikan sangat membantu dalam meningkatkan pengelolaan pemberian pakan mereka.

e. Informasi Kesehatan Ternak yang Mudah Dipahami (P5): Sistem ini berhasil memberikan informasi tentang kesehatan ternak yang mudah dipahami oleh peternak. Mayoritas responden merasa bahwa informasi yang diberikan cukup jelas dan mudah diterima. Penilaian di sekitar nilai 4

menunjukkan bahwa sistem ini cukup baik dalam menyampaikan informasi teknis dengan cara yang mudah dipahami.

f. Kemudahan Berkonsultasi dengan Penyuluh (P6): Aspek ini mendapatkan nilai yang cukup tinggi dari sebagian besar responden. Mereka merasa bahwa sistem mempermudah konsultasi dengan penyuluh, meskipun ada sedikit kritik terhadap kecepatan balasan yang diterima.

g. Kecepatan Balasan Konsultasi (P7): Meskipun sebagian besar responden memberi nilai 4 dan 5, beberapa mengungkapkan bahwa balasan konsultasi terkadang tidak cukup cepat. Hal ini menunjukkan adanya ruang untuk meningkatkan kecepatan respons penyuluh dalam sistem.

h. Manfaat Informasi untuk Ternak (P8): Sistem ini dinilai bermanfaat oleh mayoritas responden. Mereka merasa bahwa informasi yang diberikan dapat membantu dalam perawatan ternak mereka. Penilaian yang cenderung tinggi pada aspek ini menunjukkan tingkat relevansi yang baik dari informasi yang diberikan oleh sistem.

i. Keputusan dalam Merawat Ternak (P9): Sebagian besar responden menilai sistem ini sangat membantu dalam pengambilan keputusan yang berkaitan dengan merawat ternak. Hal ini menunjukkan bahwa peternak merasa lebih percaya diri dengan saran atau informasi yang diberikan oleh sistem.

j. Perbandingan dengan Cara Langsung Bertanya (P10): Sistem ini mendapatkan penilaian positif dibandingkan dengan bertanya langsung kepada petugas.

Responden merasa bahwa sistem ini lebih efisien dan praktis, yang tercermin dalam skor rata-rata yang lebih tinggi pada aspek ini.

k. Efisiensi Waktu dalam Penggunaan Sistem (P11): Sebagian besar responden merasa bahwa sistem ini membantu mereka menghemat waktu dalam perawatan ternak. Mereka tidak perlu lagi menghabiskan waktu untuk berkonsultasi secara langsung, dan bisa mendapatkan informasi dengan lebih cepat melalui sistem.

l. Kesesuaian Fitur dengan Kebutuhan Peternak (P12): Mayoritas responden memberikan skor positif pada pernyataan ini, yang menunjukkan bahwa fitur yang ada dalam sistem sesuai dengan kebutuhan mereka sebagai peternak. Namun, beberapa responden menyatakan bahwa masih ada beberapa fitur yang perlu ditambahkan atau ditingkatkan.

m. Meningkatkan Kepercayaan Diri dalam Merawat Ternak (P13): Sistem ini dirasa membantu peternak merasa lebih percaya diri dalam merawat ternak. Skor yang diberikan cenderung tinggi, mencerminkan bahwa sistem ini memberikan rasa aman bagi pengguna untuk melakukan perawatan yang lebih baik.

n. Manfaat bagi Peternak di Desa (P14): Sebagian besar responden merasa bahwa sistem ini sangat bermanfaat bagi peternak yang tinggal di desa, karena dapat mempermudah akses mereka terhadap informasi dan layanan yang sebelumnya lebih sulit dijangkau.

o. Kepuasan Umum terhadap Sistem (P15): Secara keseluruhan, mayoritas responden merasa puas dengan sistem yang digunakan. Mereka memberikan penilaian positif terkait kecepatan, kemudahan, kenyamanan, dan manfaat sistem secara keseluruhan. Hal ini menunjukkan bahwa sistem anjungan mandiri sangat membantu dalam meningkatkan efisiensi layanan bagi peternak.

Dari hasil kuesioner yang telah diisi oleh para pengguna, didapatkan data berikut:

Tabel 4. 23 Aspek Penujian Responden

Aspek Pengujian	Rata-Rata (Skala 1 -5)
Kemudahan penggunaan sistem	4.13
Kejelasan informasi yang disediakan	4.23
Kesesuaian fitur dengan kebutuhan pengguna	4.02
Kecepatan & efektivitas layanan	4.15
Manfaat sistem terhadap keputusan & kepercayaan diri	4.06
Dampak umum sistem terhadap peternak	4.33

4. Hasil Diperoleh

Perhitungan Tingkat Kepuasan Pengguna (S) dilakukan berdasarkan ratarata skor dari 15 indikator kepuasan :

Tabel 4. 24 Rata-rata Indikator Pernyataan

Indikator Pernyataan	Rata-rata
Sistem mudah digunakan oleh saya sebagai peternak	4.06
Tampilan sistem jelas dan tidak membingungkan.	3.93
Saya memahami cara menggunakan sistem tanpa bantuan orang lain.	4.4
Sistem membantu saya memahami cara pemberian pakan yang benar.	4.2
Sistem memberi informasi kesehatan ternak secara mudah dipahami.	4.26
Saya bisa berkonsultasi dengan penyuluh melalui sistem dengan mudah.	4.0
Saya mendapatkan balasan konsultasi dengan cepat dan jelas.	4.06
Informasi dari sistem bermanfaat untuk ternak saya.	3.86
Sistem ini membantu saya mengambil keputusan dalam merawat ternak.	4.33
Saya merasa sistem ini lebih baik daripada hanya bertanya secara langsung.	4.2
Sistem ini membantu saya menghemat waktu.	4.2
Fitur-fitur sistem sesuai dengan kebutuhan saya sebagai peternak.	4.0
Sistem ini membuat saya lebih percaya diri dalam merawat ternak.	4.0
Sistem sangat bermanfaat bagi peternak di desa.	4.33
Saya puas menggunakan sistem ini secara keseluruhan.	4.33

$$\frac{\sum_{i=1}^{n} Si}{n} S$$

Gambar 4. 47 Rumus Tingkat Kepuasan Pengguna (s)

Keterangan :

S = Tingkat Kepuasan Pengguna

n = Jumlah indikator yang di

ukur

Si = Skor kepuasan pada indikator ke-i

S = (4.06 + 3.93 + 4.4 + 4.2 + 4.26 + 4.0 + 4.06 + 3.86 + 4.33 + 4.2 + 4.2 + 4.0 + 4.0 + 4.33 + 4.33) / 15

S = 62.16 / 15

S = 4.14

Berdasarkan hasil penyebaran angket kepada 15 responden yang terdiri dari peternak di Desa Pakeng, Kecamatan Lembang, diperoleh nilai rata-rata keseluruhan dari lima belas pernyataan sebesar 4.14 pada skala Likert 1–5. Nilai ini menunjukkan bahwa responden secara umum memberikan tanggapan positif terhadap sistem informasi peningkatan kualitas ternak sapi yang dikembangkan.

Secara lebih rinci, pernyataan dengan nilai tertinggi terdapat pada indikator P9 (Sistem ini membantu saya mengambil keputusan dalam merawat ternak) dan P15 (Saya puas menggunakan sistem ini secara keseluruhan), masing-masing

dengan rata-rata 4.33. Hal ini menunjukkan bahwa sistem memberikan pengaruh signifikan terhadap proses pengambilan keputusan peternak serta menimbulkan tingkat kepuasan yang tinggi secara keseluruhan.

BAB V

PENUTUP

A. Kesimpulan

1. **Sistem Informasi yang Efektif**: Sistem informasi yang dikembangkan terbukti cukup efektif dalam memberikan manfaat kemudahan akses informasi terkait perawatan, pakan, kesehatan ternak, dan layanan konsultasi. Sistem ini memberikan solusi yang lebih efisien dibandingkan dengan metode tradisional yang selama ini digunakan oleh peternak.
2. Berdasarkan hasil angket: dengan rata-rata 4,14, sistem informasi peningkatan kualitas ternak sapi mendapat tanggapan positif dari peternak. Nilai tertinggi pada indikator pengambilan keputusan (P9) dan kepuasan penggunaan (P15) menunjukkan bahwa sistem ini tidak hanya bermanfaat dalam mendukung keputusan perawatan ternak, tetapi juga memberikan kepuasan secara keseluruhan, sehingga layak digunakan dalam praktik lapangan.
3. **Pengujian Sistem yang Memadai:** Pengujian sistem menggunakan metode *Balack Box* dan *White Box* menunjukkan bahwa sistem berfungsi sesuai

dengan tujuan yang telah ditetapkan, memberikan hasil yang optimal dan sesuai dengan harapan pengguna.

B. Saran

Sebagai saran, Penulis berharap sistem informasi yang dikembangkan dapat lebih diperluas dengan menambahkan fitur baru yang lebih canggih, seperti pemantauan kesehatan ternak otomatis menggunakan sensor atau perangkat IoT. Selain itu, penting untuk memberikan pelatihan lebih lanjut kepada peternak dan penyuluh agar mereka dapat memanfaatkan sistem ini dengan lebih efektif.

DAFTAR PUSTAKA

Abisono, F. Q., et al. (2023). Informasi sebagai Elemen Utama Transformasi Sosial dalam Komunikasi Digital. *Informasi: Jurnal Komunikasi Digital*, 10(3), 123-135. https://doi.org/[DOI/jurnal terkait]

Arifin, NY, Kom, S, Kom, M, Tyas, SS, Sulistiani, H, Kom, M, & ... (2022). *Analisa Perancangan Sistem Informasi*., books.google.com, (Diakses pada 22 desember 2024)

Abdullah, M. S. (2021). "Pengembangan Website Dinamis untuk Layanan Publik." Jurnal Sistem Informasi dan Teknologi Informasi, 10(3), 67-80.

Amira. (2024). Apa Itu *Flowchart*: Pengertian Menurut Ahli, Fungsi, dan Jenisnya. Gramedia. Dikutip tanggal 27 Mei 2025 pada link berikut https://www.gramedia.com/literasi/*Flowchart*/

Bekti, A. (2020). "Website sebagai Media Informasi Interaktif: Kajian pada Pengguna Internet di Indonesia." Jurnal Teknologi dan Informasi, 15(2), 45-55.

Chaniago, M. H., Amalia, F., & Widodo, A. W. (2019). Sistem Informasi Manajemen Ternak Sapi Perah di AKA Milk Jakarta. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(7), 6920-6929.

Dirgantara, MR, Syahputri, S, & ... (2023). Pengenalan Database Management System (DBMS). *Madani: Jurnal ...*, jurnal.penerbitdaarulhuda.my.id,

Fintri Indriyani, Yunita, Muthia, D. A., Surniandari, A., & Sriyadi. (2019). Analisa Perancangan Sistem Informasi. Bina Sarana Informatika Jakarta.

Hidayat, C. (2021). Pengertian Metode Waterfall dan Tahap-Tahapnya. https://ranahresearch.com/metodewaterfall/ (Diakse pada 22 juli 2024)

Hasanah, F. N., & Untari, R. S. (2020). *Rekayasa Perangkat Lunak*. UMSIDA Press.

Narulita, S, Nugroho, A, & Abdillah, MZ (2024). Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen

Penelitian dan Pengabdian Masyarakat (SIMLITABMAS). *Bridge: Jurnal publikasi …*, journal.aptii.or.id,

Supriyanta, Supriyanta, Deddy Supriadi, dan Beni Susanto. Perancangan Sistem Informasi Penggajian Karyawan Dengan Metode Waterfall. *Indonesian Journal Computer Science*, vol. 1, no. 1, 2022, Universitas Bina Sarana Informatika,2(6), ISSN: 2829-3819.

Said, M. I., Yuliati, F. N., & Hatta, M. (2018). Aplikasi teknologi pakan dan pengolahan limbah ternak pada kelompok peternak sapi pedaging di Kabupaten Bantaeng. *Jati Emas (Jurnal Aplikasi Teknik dan Pengabdian Masyarakat)*, 2(1), 59-65.

Siregar, T. N., Wahyuni, S., Thasmi, C. N., & Husnurrizal. (2020). *Gangguan Reproduksi Repeat Breeding pada Sapi Aceh*. Syiah Kuala University Press.

Sriasih, N. W. (2022). "Website sebagai Alat Komunikasi Digital dalam Era Transformasi Teknologi." Jurnal Komunikasi dan Teknologi, 9(1), 12-23.

Saputra, D., Dharmawan, W. S., Syarif, M., & Risdiansyah, D. (2023). *Analisis & Perancangan Sistem Informasi*. Penerbit Insan Cendekia Mandiri.

Sriyanto, S, Pujotomo, D, & MP, YO (2010). Pengembangan Sistem Informasi Berbasis Komputer untuk Efisiensi dan Efektivitas Pakan pada Usaha Penggemukan Sapi Potong. *J@ ti Undip: Jurnal Teknik …*, ejournal.undip.ac.id,

Triakoso, N. (2009). *Aspek klinik dan penularan pada pengendalian penyakit ternak.* Surabaya: Universitas Airlangga.

Tombeng, MT, Tambanua, S, Ambat, B, & ... (2023). Perancangan UML dan UI untuk Sistem Pemberian Pakan Ikan Otomatis Berbasis Aplikasi Mobile. *Prosiding CORISINDO …*, stmikpontianak.org,

Utomo, R., & Widiastuti, R. (2020). Ilmu Nutrisi Ternak: Konsep Dasar dan Aplikasi. Universitas Brawijaya Press.

Komaruzzaman, A., Sundari, S. S., & Hidayat, C. R. (2024). Sistem informasi perencanaan dan penganggaran dana kegiatan pada satuan pendidikan

(study kasus: SMAN 7 Tasikmalaya*). Informatech: Jurnal Ilmiah Informatika dan Komputer*,1(1),41-46. (Diakse pada 26 juli 2024)

Mufizar, T., Hidayat, C. R., & Kamaludin, W. (2018). Sistem informasi terintegrasi untuk pengelolaan rusunawa STMIK Tasikmalaya. Dalam *Prosiding Seminar Nasional Sistem Informasi dan Teknologi Informasi 2018 (SENSITEK 2018)* (hlm. 632-637). STMIK Pontianak.

Mashur, I, & Agustin, DN (2023). *Ilmu Pakan dan Nutrisi Hewan: Teori dan Praktik*.,books.google.com,https://books.google.com/books?hl=en&lr=&id=qPmyEAAAQBAJ&oi=fnd&pg=PP1&dq=pengertiant+pakan+sapi&ots=Roony2w9sZ&sig=3hNoEN2Uh20ZlQjnT46aSo9EHyg

Narulita, S, Nugroho, A, & Abdillah, MZ (2024). Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS). *Bridge: Jurnal publikasi…*,journal.aptii.or.id,

Nuraini, D. M., Pramono, A., Prastowo, S., & Widyas, N. (2022). Penyuluhan Manajemen Kesehatan Sapi Potong dan Penyakit Zoonosis di Kelompok Tani Kenteng Makmur, Ngargoyoso, Karanganyar. *AgriHealth: Journal of Agri-food, Nutrition and Public Health, 3(1), 10–18.*

Lestari, V. S., Sirajuddin, S. N., Saleh, I. M., & Indah, K. P. (2020). Perilaku peternak sapi potong terhadap pelaksanaan biosekuriti. *Prosiding Seminar Nasional Teknologi Peternakan dan Veteriner*.

Wekke, I. S., dkk. (2019). *Metode Penelitian Sosial*. Yogyakarta: Penerbit Gawe Buku.

Wahyudi, R., Utami, E., & Arief, M. R. (2016). Sistem pakar e-tourism pada Dinas Pariwisata DIY menggunakan metode Forward Chaining. *Data Manajemen dan Teknologi Informasi (DASI)*, *17*(2), 6