

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Recording

Konsep dasar sistem recording adalah pengumpulan, pencatatan, dan penyimpanan data secara sistematis untuk tujuan pengelolaan informasi yang lebih efektif dan efisien. Dalam konteks peternakan, seperti pada ayam petelur, sistem recording bertujuan untuk mencatat data yang berkaitan dengan produksi telur, konsumsi pakan, kondisi kesehatan ayam, serta variabel lainnya yang mempengaruhi operasional peternakan. Sistem ini dapat berupa perangkat lunak yang dirancang untuk mengotomatisasi proses pencatatan, memungkinkan peternak untuk memperoleh data yang lebih akurat dan mudah diakses. Dengan sistem recording yang baik, data dapat dikelola dalam format terstruktur yang memudahkan analisis dan pengambilan keputusan yang berbasis data.

Pada dasarnya, sistem recording harus memastikan bahwa informasi yang dikumpulkan tidak hanya tepat waktu dan akurat, tetapi juga mudah diakses dan diintegrasikan dengan sistem lainnya. Salah satu bentuk pengembangan sistem recording yang efisien adalah penggunaan teknologi berbasis web, yang memungkinkan peternak untuk mengakses dan mengelola data dari berbagai perangkat secara real-time. Dengan adanya sistem ini, proses pencatatan yang biasanya dilakukan secara manual dapat digantikan dengan teknologi yang mengurangi kemungkinan kesalahan, mempercepat pengumpulan data, dan meningkatkan transparansi dalam pengelolaan peternakan. Hal ini pada akhirnya

akan berkontribusi pada peningkatan efisiensi operasional dan produktivitas peternakan.

2.1.1 Pengertian Sistem

Sistem dapat didefinisikan sebagai kumpulan elemen atau komponen yang saling berinteraksi dan bekerja bersama untuk mencapai tujuan tertentu. Dalam konteks informasi, sistem merujuk pada rangkaian prosedur dan alat yang saling terhubung, yang digunakan untuk mengelola, memproses, dan mendistribusikan data atau informasi. Sistem dapat berupa sistem fisik, seperti perangkat keras komputer, maupun sistem abstrak seperti perangkat lunak atau prosedur operasional. Sistem ini juga dapat terdiri dari berbagai sub-sistem yang saling mendukung satu sama lain, seperti sistem informasi yang terdiri dari perangkat keras, perangkat lunak, manusia, dan data.

Selain itu, sistem juga dapat dilihat sebagai suatu entitas yang terbuka atau tertutup, tergantung pada interaksinya dengan lingkungan luar. Sistem terbuka berinteraksi dengan lingkungan eksternal untuk memperoleh input dan menghasilkan output, sedangkan sistem tertutup lebih bersifat otonom dalam operasionalnya. Dalam dunia teknologi dan manajemen, penerapan sistem menjadi penting untuk mempermudah pengelolaan informasi, meningkatkan efisiensi, dan menghasilkan keputusan yang lebih baik berdasarkan data yang terstruktur dan terkelola dengan baik.

2.1.2 Pengertian Recording

Recording, dalam konteks teknologi informasi, merujuk pada proses pencatatan atau penyimpanan data atau informasi untuk tujuan dokumentasi atau pengolahan lebih lanjut. Proses ini dapat dilakukan dengan berbagai cara, mulai dari

pencatatan manual menggunakan buku atau formulir, hingga pencatatan digital yang menggunakan perangkat lunak atau sistem otomatis. Tujuan dari recording adalah untuk memastikan bahwa data yang penting tidak hilang dan dapat diakses kapan saja sesuai kebutuhan. Pencatatan yang tepat dan sistematis akan membantu dalam mengelola informasi yang lebih akurat dan terstruktur.

Dalam konteks operasional, recording tidak hanya berfungsi sebagai cara untuk mengumpulkan data, tetapi juga untuk melacak dan memantau berbagai aktivitas yang terjadi dalam suatu sistem atau proses. Sebagai contoh, dalam dunia bisnis atau industri, recording dapat mencakup pencatatan transaksi, produksi, penggunaan sumber daya, dan aspek lainnya yang penting. Pencatatan yang baik akan memberikan gambaran yang jelas mengenai kinerja dan perkembangan dari sistem atau proses tersebut, serta menjadi dasar yang kuat untuk pengambilan keputusan di masa depan.

2.1.3 Pengertian Sistem Recording

Sistem recording merujuk pada suatu sistem yang dirancang untuk mengotomatisasi, mengelola, dan memproses pencatatan atau penyimpanan data dalam suatu organisasi atau proses. Sistem ini menggabungkan perangkat keras dan perangkat lunak yang bekerja secara bersama-sama untuk mencatat dan menyimpan data dengan cara yang efisien dan terstruktur. Dalam konteks peternakan ayam petelur, sistem recording dapat digunakan untuk mencatat data mengenai produksi telur, konsumsi pakan, kesehatan ayam, serta faktor-faktor lainnya yang mempengaruhi kinerja peternakan. Sistem ini memungkinkan data untuk diproses secara real-time dan diakses kapan saja oleh pemangku kepentingan yang

membutuhkan informasi tersebut. Recording data merupakan komponen kunci dalam sistem informasi manajemen, karena proses ini memastikan bahwa informasi yang dicatat dan disimpan dapat diakses dan digunakan secara efektif (Cahyono & Sutarti, 2021).

Sistem recording yang efektif tidak hanya mencakup proses pencatatan data, tetapi juga mencakup kemampuan untuk menganalisis, menyajikan, dan melaporkan informasi yang relevan untuk pengambilan keputusan. Misalnya, dengan adanya sistem recording berbasis web, peternak dapat memperoleh informasi terkait produktivitas ayam secara langsung dan melakukan analisis terhadap pola-pola yang muncul. Sistem seperti ini memungkinkan integrasi data dari berbagai sumber, sehingga memudahkan peternak dalam melakukan perencanaan jangka panjang dan meningkatkan efisiensi operasional. Dengan demikian, sistem recording tidak hanya berfungsi sebagai alat pencatat data, tetapi juga sebagai alat bantu dalam proses pengelolaan dan perencanaan yang lebih baik.

2.2 Sistem Recording Ayam Petelur Nasa Poultry

Peternakan ini masih menggunakan metode manual dalam pencatatan data produksi harian, seperti jumlah ayam dan jumlah telur yang diproduksi, pencatatan manual yang dimaksud adalah dengan menggunakan buku. Pencatatan dalam buku meliputi beberapa kategori yaitu kandang, jumlah ayam, jumlah telur, persentase, ayam afkir/mati dan umur ayam. Pencatatan manual ini berisiko menyebabkan kesalahan dalam data, keterlambatan dalam pemantauan hasil produksi, serta kesulitan dalam melakukan analisis tren secara efisien. Selain itu, pengelolaan data

yang terbatas pada metode manual membuat perencanaan dan pengambilan keputusan jangka panjang menjadi kurang optimal.

2.3 Sistem Recording Ayam Petelur Berbasis Web

Sistem recording ayam petelur berbasis web adalah solusi teknologi yang dirancang untuk menggantikan metode manual dalam pencatatan dan pengelolaan data operasional peternakan ayam petelur. Sistem ini memungkinkan peternak untuk mencatat data produksi telur, konsumsi pakan, kesehatan ayam, serta parameter lainnya secara digital dan terstruktur. Dengan menggunakan platform berbasis web, data dapat diakses dan diperbarui secara real-time dari berbagai perangkat yang terhubung ke internet, sehingga memudahkan peternak dalam memantau dan mengelola operasional peternakan tanpa terbatas oleh lokasi dan waktu. Sistem ini juga mendukung integrasi data dari berbagai aspek peternakan, sehingga memudahkan analisis dan pengambilan keputusan yang lebih cepat dan tepat. (Orolean, C. Y., Irawan, J. D., & Rudhistiar, D. 2021).

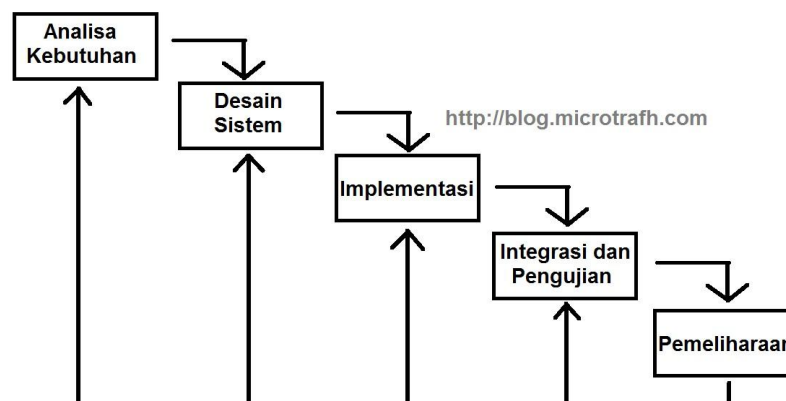
Selain meningkatkan efisiensi dalam pencatatan dan pengelolaan data, sistem berbasis web juga memberikan keuntungan dalam hal akurasi data dan pengelolaan informasi yang lebih terorganisir. Penggunaan sistem ini mengurangi ketergantungan pada pencatatan manual yang rentan terhadap kesalahan dan kehilangan data. Dengan adanya fitur otomatisasi seperti pelaporan harian, pengingat jadwal pemberian pakan dan vaksinasi, serta analisis data historis, peternak dapat dengan mudah mendapatkan informasi yang dibutuhkan untuk meningkatkan produktivitas dan kualitas hasil ternak. Sistem recording ayam petelur berbasis web juga dapat menjadi alat yang efektif dalam merencanakan

strategi jangka panjang, memantau tren produktivitas, serta mengoptimalkan manajemen sumber daya yang ada di peternakan.

2.4 Metode Yang Digunakan

2.4.1 Waterfall

Model pengembangan sistem yang diterapkan dalam penelitian ini adalah metode Waterfall, yang juga dikenal sebagai metode air terjun. Pendekatan ini digunakan untuk menggambarkan proses pengembangan perangkat lunak secara sistematis dan terstruktur, di mana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya (Badrul, M. 2021). Dengan pendekatan berurutan seperti ini, metode Waterfall memastikan bahwa semua kebutuhan sistem dapat terpenuhi dengan baik. Hasil akhirnya adalah sebuah sistem yang dirancang untuk berjalan secara optimal sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Metode ini sering dipilih karena kemampuannya untuk memberikan kerangka kerja yang jelas dan terorganisasi selama proses pengembangan perangkat lunak.



Gambar 2.1 Metode Waterfall

Tahap – tahap pengembangan waterfall model adalah:

1. Analisis dan definisi persyaratan Pelayanan, batasan, dan tujuan sistem ditentukan melalui konsultasi dengan user.
2. Perancangan sistem dan perangkat lunak Kegiatan ini menentukan arsitektur sistem secara keseluruhan.
3. Implementasi dan pengujian unit Perancangan perangkat lunak direalisasikan sebagai serangkaian program.
4. Integrasi dan pengujian sistem Unit program diintegrasikan atau diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah terpenuhi
5. Operasi dan pemeliharaan Merupakan fase siklus yang paling lama. Sistem diinstall dan dipakai. Perbaikan mencakup koreksi dari berbagai error, perbaikan dan implementasi unit sistem dan pelayanan sistem.

2.5 Alat Bantu Perancangan

2.5.1 Pengertian UML (Unified Modelling Language)

Unified Modeling Language (UML) adalah bahasa pemodelan visual yang digunakan untuk merancang dan mendokumentasikan sistem perangkat lunak. UML menyediakan cara standar untuk memvisualisasikan desain suatu sistem, termasuk

interaksi antar komponen, alur kerja, serta struktur internalnya. Dikembangkan oleh Object Management Group (OMG), UML telah menjadi standar industri dalam pengembangan perangkat lunak berbasis objek. Dengan menggunakan UML, pengembang dapat memodelkan berbagai aspek sistem secara terstruktur dan konsisten, sehingga memudahkan komunikasi dan kolaborasi antar anggota tim proyek.

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan visual yang berfungsi untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak melalui berbagai diagram yang jelas dan mudah dipahami. Sebagai standar industri, UML digunakan dalam pengembangan perangkat lunak berbasis objek untuk merepresentasikan beragam aspek sistem, baik dari segi struktur, fungsi, maupun dinamika. Selain itu, UML menyatukan berbagai pendekatan pemodelan yang telah ada sebelumnya ke dalam satu bahasa terpadu yang dapat diterapkan di seluruh tahap siklus hidup pengembangan perangkat lunak.

2.5.2 Use Case Diagram

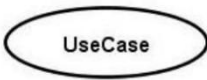
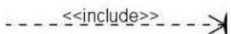
Use Case Diagram secara umum merupakan representasi visual dari proses sistem secara menyeluruh yang melibatkan aktor sebagai pengguna. Use case berfungsi untuk menggambarkan bagaimana para pemangku kepentingan sistem berinteraksi dengan sistem tersebut. Pengembangan use case berguna untuk mendalami dan memahami kebutuhan sistem secara lebih rinci.

Berdasarkan kutipan tersebut, *Use Case Diagram* menekankan representasi fungsi atau layanan yang ditawarkan oleh sistem dari sudut pandang pengguna. Dalam diagram ini, aktor direpresentasikan sebagai entitas eksternal yang

berhubungan dengan sistem untuk mencapai tujuan tertentu, sementara sistem digambarkan sebagai sebuah kotak yang mencakup berbagai fungsi yang disediakan.

Berikut adalah simbol-simbol yang ada pada *use case diagram* dapat dilihat pada tabel berikut ini :

Tabel 2.1 Use Case Diagram

Simbol	Nama	Deskripsi
	Use case	fungsi yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja
	Aktor / actor	Abstraksi dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan Use Case, tetapi tidak memiliki kontrol terhadap use case
	Asosiasi / association	Asosiasi antara aktor dan use case, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data
	Asosiasi / association	Asosiasi antara aktor dengan use case yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
	Include	Include, merupakan di dalam use case lain (required) atau pemanggilan use case oleh use case contohnya adalah pemanggilan sebuah fungsi program
	Extend	Extend, merupakan perluasan dari use case lain jika kondisi atau syarat terpenuhi

2.5.3 Activity Diagram

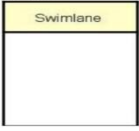
Activity Diagram adalah representasi alur aktivitas atau alur kerja dalam sebuah sistem yang akan dijalankan. Diagram ini memungkinkan perancang untuk menentukan bagaimana sistem aplikasi mencapai tujuannya serta menggambarkan urutan aktivitas dalam proses pengoperasian aplikasi.

Diagram ini menggambarkan cara suatu aktivitas atau proses berlangsung dalam sistem, menunjukkan urutan langkah-langkah yang harus dilakukan untuk mencapai tujuan tertentu. Activity Diagram sangat bermanfaat dalam memvisualisasikan alur kontrol dan pengambilan keputusan yang terjadi, serta menjelaskan proses atau alur kerja dari awal hingga akhir, termasuk aktivitas yang perlu dilakukan, keputusan yang harus diambil, dan kondisi yang harus dipenuhi.

Berikut adalah simbol-simbol yang ada pada *activity diagram* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Activity Diagram

Simbol	Nama	Deskripsi
	Start Point	Start Point, diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	End Point	End Point, akhir aktivitas
	Activities	Activities, menggambarkan suatu proses atau kegiatan bisnis
	Fork atau Percabangan	Fork atau percabangan, digunakan untuk menunjukan kegiatan yang dilakukan secara paralel untuk menggabungkan dua kegiatan paralel menjadi satu
	Join atau Penggabungan	Join (penggabungan) atau rake, digunakan untuk menunjukan adanya dekomposisi

	Decision Points	Decision points, menggambarkan pilihan untuk pengambilan keputusan, true atau false
	Swimlane	Swimlane, pembagian activity diagram untuk menunjukkan siapa melakukan apa

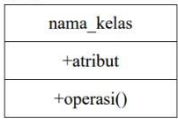
2.5.4 Class Diagram

Class Diagram adalah representasi visual yang menunjukkan struktur sistem melalui definisi kelas-kelas yang akan digunakan dalam pengembangan sistem. Kelas (class) merupakan sekumpulan objek yang memiliki struktur, perilaku, hubungan, dan makna yang serupa. Penentuan atau identifikasi kelas dilakukan dengan menganalisis objek-objek yang terdapat dalam Sequence Diagram dan Collaboration Diagram. Sebuah kelas biasanya digambarkan dalam bentuk persegi panjang yang terbagi menjadi tiga bagian.

Class Diagram digunakan untuk memvisualisasikan struktur statis suatu sistem, dengan menekankan pada kelas-kelas, atribut, metode (operasi), dan hubungan antar kelas dalam sistem tersebut. Class Diagram adalah alat yang sangat krusial dalam perancangan sistem berbasis objek karena memberikan gambaran yang jelas tentang cara data dan fungsionalitas diorganisasikan dalam sistem.

Berikut adalah simbol-simbol yang ada pada *class diagram* dapat dilihat pada tabel berikut ini:

Tabel 2.3 Class Diagram

Simbol	Nama	Deskripsi
	Kelas	Kelas pada struktur sistem

	Antar muka / interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
	Asosiasi / association	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
	Asosiasi berarah / directed association	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
	Generalisasi	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)
	Kebergantungan / dependency	Relasi antarkelas dengan makna kebergantungan antarkelas
	Agregasi / aggregation	Relasi antar kelas dengan makna semua – bagian (whole-part)

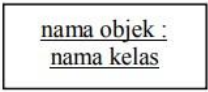
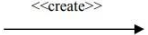
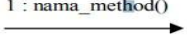
2.5.5 Sequence Diagram

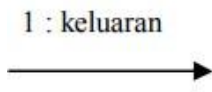
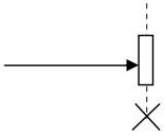
Sequence Diagram adalah diagram yang menunjukkan interaksi antar objek yang disusun berdasarkan urutan waktu. Diagram ini menggambarkan objek dan kelas yang terlibat dalam sebuah skenario, serta pesan-pesan yang dipertukarkan antara objek-objek tersebut untuk melaksanakan fungsi dalam skenario tersebut.

Sequence Diagram menggambarkan urutan interaksi antara objek-objek dalam sistem, dari awal hingga akhir suatu proses. Diagram ini memperlihatkan objek-objek yang terlibat serta urutan pesan yang dikirim antar objek untuk melaksanakan fungsi atau aktivitas tertentu. Setiap objek dalam diagram ini diwakili oleh sebuah lifeline yang digambarkan sebagai garis vertikal, sementara interaksi antar objek ditunjukkan dengan panah horizontal yang menggambarkan pesan yang dikirim.

Berikut adalah simbol-simbol yang ada pada *Sequence diagram* dapat dilihat pada tabel berikut ini:

Tabel 2.4 *Sequence Diagram*

Simbol	Nama	Deskripsi
	Aktor	poses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang
	Garis hidup / lifeline	Menyatakan kehidupan suatu objek
	Objek	Menyatakan objek yang berinteraksi pesan
	Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
	Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
	Pesan tipe call	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
	Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim

	Pesan tipe return	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
	Pesan tipe destroy	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy

2.5.6 Website

Web, atau lebih dikenal dengan World Wide Web (WWW), adalah sistem informasi yang memungkinkan pengguna untuk mengakses dan berbagi data melalui internet. Web terdiri dari berbagai situs dan aplikasi yang dapat diakses menggunakan browser, yang memungkinkan pengguna untuk melihat informasi dalam bentuk teks, gambar, video, dan multimedia lainnya. Web menghubungkan berbagai jenis sumber daya digital di seluruh dunia, yang dapat diakses secara realtime. Teknologi web telah berkembang pesat, tidak hanya untuk tujuan informasi, tetapi juga untuk bisnis, pendidikan, komunikasi, dan berbagai sektor lainnya, memungkinkan interaksi antara pengguna dan perangkat lunak yang dapat diakses secara online dari mana saja.

Pengembangan aplikasi berbasis web telah menjadi pilihan utama dalam banyak industri karena kemudahan aksesnya yang dapat dilakukan melalui berbagai perangkat, seperti komputer, tablet, atau ponsel pintar. Aplikasi berbasis web memungkinkan pengelolaan data secara terpusat, di mana data dapat disimpan di server dan diakses kapan saja tanpa perlu instalasi perangkat lunak khusus pada

setiap perangkat pengguna. Dengan fitur seperti responsif, integrasi dengan berbagai platform lain, serta pembaruan sistem yang dapat dilakukan secara langsung, aplikasi berbasis web memberikan fleksibilitas dan efisiensi yang lebih besar. Hal ini menjadikannya sangat berguna dalam berbagai konteks, termasuk pengelolaan sistem informasi di bidang pertanian, seperti dalam penelitian pengembangan sistem recording ayam petelur berbasis web.

2.6 Alat Bantu Pemrograman

2.6.1 PHP Native

PHP Native merujuk pada penggunaan PHP tanpa framework tambahan, yang memungkinkan pengembang untuk menulis kode dengan cara yang lebih fleksibel dan kontrol penuh atas struktur aplikasi. Dengan PHP Native, pengembang dapat membangun aplikasi web secara langsung dengan memanfaatkan fitur-fitur dasar PHP seperti pengolahan formulir, pengelolaan sesi, koneksi database, dan manipulasi file, tanpa adanya dependensi pada framework eksternal. Hal ini memberikan keuntungan berupa pengembangan yang lebih cepat untuk proyek kecil atau untuk pengembang yang ingin memahami dan menguasai PHP secara mendalam, karena mereka dapat menyesuaikan dan mengoptimalkan setiap bagian kode sesuai kebutuhan spesifik.

Namun, meskipun menawarkan fleksibilitas, penggunaan PHP Native juga memiliki beberapa kelemahan, terutama pada proyek dengan skala besar atau yang membutuhkan pengelolaan kode yang lebih terstruktur. Tanpa framework, pengembang mungkin menghadapi tantangan dalam hal pengorganisasian kode, pengelolaan keamanan, dan skalabilitas aplikasi. Oleh karena itu, meskipun PHP

Native cocok untuk aplikasi sederhana atau bagi pengembang yang ingin membangun solusi yang lebih ringan, framework PHP seperti Laravel atau CodeIgniter lebih sering digunakan untuk aplikasi yang lebih kompleks, karena menawarkan berbagai fitur seperti routing, otentikasi, dan ORM yang dapat mempercepat pengembangan dan meningkatkan keamanan aplikasi.

2.6.2 HTML

HTML (HyperText Markup Language) adalah bahasa dasar yang digunakan untuk membuat dan menyusun halaman web di internet. HTML berfungsi untuk menentukan struktur konten dalam sebuah halaman web dengan menggunakan elemen-elemen atau tag yang mengatur teks, gambar, tautan, tabel, formulir, dan elemen multimedia lainnya. Tag HTML terdiri dari pasangan tag pembuka dan penutup, yang mengelilingi konten yang ingin ditampilkan di halaman. Sebagai contoh, tag digunakan untuk mendefinisikan paragraf, tag untuk menampilkan gambar, dan untuk membuat tautan. HTML juga memungkinkan pengaturan atribut untuk elemen-elemen tertentu, seperti ukuran gambar atau warna teks, yang meningkatkan fleksibilitas desain web.

HTML merupakan fondasi dari pengembangan situs web, namun untuk membuatnya lebih dinamis dan interaktif, HTML biasanya digunakan bersama CSS (Cascading Style Sheets) untuk penataan dan JavaScript untuk interaktivitas. HTML5, versi terbaru dari HTML, membawa berbagai peningkatan dan fitur baru, seperti dukungan untuk video dan audio, elemen formulir yang lebih canggih, dan API yang memungkinkan pengembangan aplikasi web yang lebih kompleks. Meskipun HTML sendiri tidak dapat digunakan untuk membuat halaman web yang

dinamis tanpa tambahan teknologi lainnya, HTML tetap menjadi bahasa inti yang tidak bisa dipisahkan dalam pembangunan dan pengembangan situs web modern.

2.6.3 CSS

CSS (Cascading Style Sheets) adalah bahasa pemrograman yang digunakan untuk mendesain dan mengatur tampilan elemen-elemen pada halaman web, seperti warna, ukuran, tata letak, dan font. CSS bekerja dengan memisahkan konten dari presentasi, memungkinkan pengembang web untuk mengubah desain suatu halaman tanpa mempengaruhi struktur HTML-nya. Dengan menggunakan CSS, pengembang dapat mengontrol berbagai aspek visual seperti background, margin, padding, dan font untuk menciptakan antarmuka pengguna yang lebih menarik dan mudah dinavigasi. CSS juga mendukung responsivitas desain, sehingga tampilan halaman web dapat menyesuaikan dengan berbagai ukuran layar perangkat, seperti desktop, tablet, dan ponsel.

CSS memiliki berbagai cara untuk menyusun gaya pada elemen-elemen HTML, seperti melalui penggunaan selektor, properti, dan nilai. Selektor digunakan untuk memilih elemen yang akan diberi gaya, sedangkan properti menentukan atribut yang ingin diubah, dan nilai memberikan penjelasan tentang bagaimana elemen tersebut akan tampil. CSS dapat diterapkan langsung dalam file HTML menggunakan tag `<style>`, atau dengan merujuk ke file eksternal menggunakan tag `<link>`. Penggunaan file CSS eksternal memungkinkan pengembang untuk mengelola desain secara terpusat dan lebih efisien, karena satu file CSS dapat digunakan untuk banyak halaman web. CSS adalah salah satu keterampilan dasar yang sangat penting dalam pengembangan web modern, memungkinkan pembuatan halaman yang estetis, fungsional, dan mudah diakses.

2.6.4 Java Script

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web lebih interaktif dan dinamis. Bahasa ini dijalankan di sisi klien (browser), memungkinkan pengembang untuk menambahkan berbagai fitur interaktif seperti animasi, validasi formulir, dan pengelolaan konten secara realtime tanpa perlu memuat ulang halaman. JavaScript mendukung pengolahan data, manipulasi DOM (Document Object Model), serta berkomunikasi dengan server melalui teknologi seperti AJAX, yang memungkinkan pengembangan aplikasi web yang responsif dan efisien. Keunggulan utama JavaScript adalah kemampuannya untuk berfungsi di berbagai platform, sehingga memungkinkan pengembang untuk membuat aplikasi web yang dapat diakses di berbagai perangkat dan sistem operasi.

Selain itu, JavaScript terus berkembang dengan munculnya berbagai pustaka dan framework yang mempermudah pengembangan aplikasi web. Beberapa pustaka dan framework terkenal seperti React, Angular, dan Vue.js membantu pengembang dalam membangun aplikasi web modern dengan lebih cepat dan efisien. JavaScript juga berintegrasi dengan berbagai teknologi lain, seperti HTML dan CSS, untuk menciptakan pengalaman pengguna yang lebih baik. Dengan dukungan yang luas dari komunitas pengembang, JavaScript telah menjadi salah satu bahasa pemrograman paling populer di dunia, baik untuk pengembangan web front-end maupun back-end (menggunakan Node.js). Sebagai bahasa yang sangat fleksibel dan memiliki ekosistem yang kaya, JavaScript memainkan peran penting dalam perkembangan teknologi web modern.

2.6.5 Database MySQL

MySQL adalah salah satu sistem manajemen basis data relasional (RDBMS) yang paling populer di dunia, digunakan untuk mengelola dan menyimpan data dalam bentuk tabel yang saling terhubung. MySQL bersifat opensource dan mendukung berbagai platform, seperti Windows, Linux, dan macOS, yang membuatnya sangat fleksibel dan mudah diintegrasikan dengan aplikasi berbasis web. Dalam MySQL, data disimpan dalam database yang terdiri dari tabel-tabel dengan baris dan kolom, di mana setiap kolom memiliki tipe data tertentu sesuai dengan kebutuhan, seperti angka, teks, atau tanggal. MySQL juga menawarkan bahasa query SQL (Structured Query Language) untuk melakukan operasi pada data, seperti penyimpanan, pengambilan, pembaruan, dan penghapusan data, serta memungkinkan pembuatan relasi antar tabel menggunakan kunci utama (primary key) dan kunci asing (foreign key).

Keunggulan utama MySQL adalah kemampuannya dalam menangani volume data yang besar dan performa yang tinggi, sehingga sangat cocok untuk aplikasi yang memerlukan pemrosesan data secara cepat, seperti pada aplikasi berbasis web dan e-commerce. MySQL juga menawarkan keamanan yang kuat dengan fitur autentikasi, kontrol akses berbasis peran (role-based access control), dan enkripsi data, memastikan bahwa data yang disimpan tetap aman. Selain itu, MySQL mendukung replikasi dan clustering, yang memungkinkan pengelolaan data secara terdistribusi dan memastikan ketersediaan data yang tinggi. Dengan dukungan komunitas yang besar dan dokumentasi yang lengkap, MySQL menjadi

pilihan utama untuk pengembangan aplikasi yang memerlukan basis data yang handal dan skalabel.

2.6.6 XAMPP

XAMPP adalah sebuah paket perangkat lunak yang menyediakan berbagai komponen untuk mengembangkan aplikasi berbasis web secara lokal, termasuk server Apache, database MySQL/MariaDB, serta bahasa pemrograman PHP dan Perl. XAMPP memungkinkan pengguna untuk menginstal dan mengonfigurasi lingkungan pengembangan web secara cepat dan mudah di komputer lokal tanpa memerlukan server fisik yang terpisah. Salah satu fitur utamanya adalah kemudahan dalam mengelola database menggunakan phpMyAdmin, yang merupakan antarmuka berbasis web untuk mengelola MySQL atau MariaDB. Dengan XAMPP, pengembang dapat menguji dan mengembangkan aplikasi web secara offline sebelum mempublikasikannya ke server nyata, yang sangat bermanfaat untuk aplikasi yang membutuhkan pengujian atau pengembangan berkelanjutan.

Database MySQL atau MariaDB dalam XAMPP digunakan untuk menyimpan data dalam format tabel yang saling berhubungan, memungkinkan penyimpanan dan pengambilan data dengan cepat dan efisien. Sistem manajemen database ini menggunakan bahasa query SQL untuk mengelola data yang ada, memungkinkan pengguna untuk membuat, membaca, memperbarui, dan menghapus informasi (CRUD). Dalam konteks pengembangan sistem berbasis web, seperti yang digunakan untuk manajemen peternakan ayam petelur, database ini sangat penting untuk menyimpan berbagai informasi seperti data produksi telur, konsumsi pakan, kesehatan ayam, serta data historis lainnya. Penggunaan XAMPP

untuk mengelola database mempermudah pengembangan dan pemeliharaan aplikasi, serta memungkinkan integrasi data secara lebih efektif dalam pengelolaan peternakan yang berbasis teknologi.

2.6.7 Aplikasi Visual Studio Code

Visual Studio Code (VS Code) adalah editor kode sumber yang ringan dan kuat yang dikembangkan oleh Microsoft, yang mendukung berbagai bahasa pemrograman seperti JavaScript, Python, C++, dan banyak lagi. Salah satu keunggulan utama dari VS Code adalah kemampuannya untuk disesuaikan melalui berbagai ekstensi yang memungkinkan penggunanya menambah fitur sesuai kebutuhan, seperti dukungan untuk bahasa pemrograman tambahan, integrasi sistem kontrol versi, hingga tools debugging. Dengan antarmuka yang sederhana dan intuitif, VS Code memungkinkan pengembang untuk menulis dan mengelola kode dengan lebih efisien. Fitur seperti auto-completion, syntax highlighting, dan linting sangat membantu dalam meningkatkan produktivitas pengembang.

Selain itu, Visual Studio Code juga menawarkan berbagai fitur kolaborasi dan integrasi dengan platform pengembangan lain, seperti GitHub, yang memudahkan pengelolaan proyek bersama tim. Kemampuan debugging langsung dalam editor dan integrasi dengan terminal membuatnya menjadi pilihan populer bagi banyak pengembang perangkat lunak. Dukungan terhadap berbagai sistem operasi, seperti Windows, macOS, dan Linux, menjadikan VS Code sebagai alat yang sangat fleksibel untuk berbagai jenis proyek, dari pengembangan aplikasi web hingga perangkat lunak berbasis server. Karena sifatnya yang ringan, VS Code

sangat cocok untuk digunakan dalam proyek kecil maupun besar, memberikan keseimbangan antara kemudahan penggunaan dan kemampuan pengembangan tingkat lanjut.

2.7 Penelitian Terdahulu

Tabel 2.5. Penelitian Terdahulu

No	Penulis dan tahun	Judul Penelitian	Metode	Hasil Penelitian	Perbedaan
1.	Nelfira, N., Suryani, A. I., Elizamiharti, E., & Anggraini, I. (2024)	Sistem Informasi Pengolahan Data Peternakan Ayam Merah Petelur pada Astipel Farm Berbasis Web	Studi pustaka, observasi, wawancara, implementasi sistem	Sistem informasi berbasis web berhasil mengurangi kesalahan pencatatan manual dan meningkatkan efisiensi pengelolaan data produksi telur, data ayam, stok pakan, dan penjualan.	Penelitian ini lebih fokus pada pencatatan data produksi ayam petelur secara terintegrasi dengan fitur analisis yang lebih mendalam
2.	Werdiningsih, P., Isnanto, R. R., & Prasetyo, A. B. (2023)	Pengembangan Aplikasi Recording Telur Ayam Berbasis Web pada Kelompok Wanita Tani (KWT) Kembang Wono	Pengembangan aplikasi berbasis web	Aplikasi berbasis web berhasil mengatasi masalah pencatatan manual, seperti kesalahan data, kesulitan penyimpanan, dan pencarian data yang lambat.	Penelitian ini menggunakan pendekatan berbasis web yang lebih terstruktur dan mencakup pencatatan produksi telur, kondisi ayam, serta laporan keuangan

3.	Febryana, R. F. H., Brata, A. H., & Widodo, A. W. (2020)	Pengembangan Sistem Informasi Manajemen Produksi Telur PT. Vega Nusa Argita berbasis Web (Studi Kasus : Desa Watukebo Kecamatan Rogojampi Banyuwangi)	Studi pustaka, analisis kebutuhan, implementasi sistem	Sistem informasi berbasis web berhasil mengatasi masalah pencatatan manual, seperti biaya pencatatan ulang, pencarian data yang lama, dan penyimpanan data yang rentan hilang atau rusak. Telur pada Vega Farm dilakukan masih manual, sehingga dalam pengerjaannya dapat memakan waktu yang lama	Penelitian ini memiliki cakupan yang lebih luas dengan mencakup pencatatan ayam, telur, kandang, serta laporan operasional peternakan
4.	Ikhwan Fauzy Djuhu, Drs. Muhammad Rifai Katili, Sitti Suhada (2020)	Web Portal Pemasaran Telur Ayam	Studi pustaka, analisis kebutuhan, perancangan, dan pengujian	Aplikasi pemasaran berbasis web mempermudah proses jual beli telur ayam	Penelitian ini lebih berfokus pada pemasaran telur ayam secara online, sedangkan penelitian ini fokus pada manajemen produksi dan pencatatan operasional peternakan