BAB II
TINJAUAN PUSTAKA

2.1.Tinjauan Pustaka

2.1.1. Manajemen Pemeliharaan Ayam Broiler Secara Manual

Manajemen pemeliharaan ayam broiler secara manual masih menjadi pendekatan yang dominan di berbagai peternakan rakyat dan kemitraan. Diah Fadiellah dan Fitri Dian Perwitasari meneliti peternak mitra PT Ciomas Adisatwa di Kabupaten Cirebon, yang masih mengandalkan metode tradisional mulai dari pemberian pakan, vaksinasi, hingga panen. Penelitian ini mencatat bahwa ayam dipelihara selama 29 hari dengan bobot rata-rata 1,6 kg, menggunakan tiga jenis pakan (SB-10, SB-11, SB-12) yang diberikan secara manual sesuai jadwal yang ditentukan perusahaan. Proses pemberian pakan dilakukan oleh tenaga manusia dua kali sehari, dengan frekuensi dan jumlah yang masih rentan terhadap kesalahan penghitungan [3].

Selain itu, pemantauan kesehatan ayam juga dilakukan secara manual melalui inspeksi harian, dengan tindakan isolasi ayam sakit serta pemberian obat dan vitamin. Sistem panen masih mengandalkan tenaga kerja untuk menangkap, menimbang, dan mencatat bobot ayam satu per satu. Meskipun metode ini relatif sederhana dan sesuai dengan kapasitas peternak kecil, kelemahannya adalah tingginya risiko keterlambatan dalam deteksi mortalitas dan kesalahan pencatatan populasi, yang dapat berpengaruh pada akurasi data produksi dan margin keuntungan. Hal ini menunjukkan bahwa metode manual memiliki keterbatasan dalam efisiensi dan akurasi, sehingga menjadi salah satu alasan perlunya inovasi teknologi pada manajemen peternakan ayam broiler [3].

Tantangan dalam manajemen pemeliharaan ayam broiler secara manual juga mencakup aspek pemantauan kondisi lingkungan kandang yang sangat mempengaruhi produktivitas ternak. Verdouw dkk. dalam penelitiannya tentang implementasi ERP di sektor hortikultura Belanda menunjukkan bahwa sistem manajemen manual menghadapi kendala serupa dalam hal pencatatan data

lingkungan dan operasional yang tidak real-time, menyebabkan keterlambatan dalam pengambilan keputusan [4]. Dalam konteks peternakan ayam broiler, pemantauan parameter seperti suhu, kelembaban, dan kualitas udara secara manual seringkali tidak konsisten dan rentan terhadap human error. Nalendra dan Waspada mengembangkan sistem IoT berbasis mobile untuk pemantauan real-time kondisi kandang broiler yang mencakup monitoring suhu, kelembaban, level pakan dan air, serta manajemen inventaris [5]. Sistem ini menunjukkan bahwa integrasi sensor IoT dengan aplikasi mobile dapat meningkatkan efisiensi operasional hingga 50,53% dalam hal waktu pemrosesan data dibandingkan metode manual. Keterlambatan dalam mendeteksi kondisi lingkungan yang tidak optimal dapat menyebabkan heat stress atau cold stress yang berdampak pada penurunan Feed Conversion Ratio (FCR) dan peningkatan mortalitas. Hal ini memperkuat argumen bahwa sistem pemantauan otomatis berbasis IoT yang terintegrasi dengan sistem manajemen peternakan diperlukan untuk meningkatkan responsivitas dan efisiensi operasional.

Kondisi serupa juga ditemukan oleh Sudarko dan Herminingsih pada peternak plasma di Pamijahan, Bogor, yang menunjukkan bahwa kapasitas agribisnis peternak rakyat ayam broiler pada sistem kemitraan inti plasma masih berada dalam kategori rendah dengan ketergantungan tinggi pada perusahaan inti. Penelitian tersebut mengidentifikasi bahwa faktor yang berpengaruh signifikan terhadap kapasitas peternak adalah pendapatan, akses informasi pemasaran, dan materi penyuluhan, mengindikasikan bahwa peningkatan kapasitas tidak hanya bergantung pada modal tetapi juga pada akses informasi yang akurat dan tepat waktu [6]. McCarthy dalam artikelnya tentang kecerdasan buatan menjelaskan bahwa AI dapat membantu mengatasi keterbatasan kognitif manusia dalam memproses data kompleks secara real-time, yang sangat relevan dengan kebutuhan pemantauan peternakan modern [7]. Hal ini memperkuat alasan bahwa inovasi teknologi pada manajemen peternakan perlu dirancang tidak hanya untuk memberdayakan peternak melalui penyediaan informasi real-time yang dapat meningkatkan kemandirian dalam pengambilan keputusan.

2.1.2. Pemanfaatan *Computer Vision* dalam Peternakan Ayam

Untuk menjawab kelemahan dalam perhitungan manual, teknologi *computer vision* telah banyak dikembangkan untuk pemantauan populasi ayam secara otomatis. Cruz dkk. berhasil merancang *Automated Counting and Monitoring System for Poultry Tracking* (ACMSPT) yang menggunakan model YOLOv10 untuk menghitung populasi ayam secara *real-time* di lingkungan peternakan komersial. Sistem yang diimplementasikan pada perangkat *edge computing* ini mampu mencapai presisi hingga 93,1% membuktikan tingkat efektivitas yang tinggi dalam mengatasi tantangan seperti variasi pencahayaan dan kepadatan kandang yang tinggi [8]. Sementara itu Pramudya dkk. menunjukkan keberhasilan penggunaan model YOLOv4-tiny untuk melakukan deteksi dan pengukuran ayam broiler, yang menjadi dasar perhitungan populasi dan bobot ayam broiler [9].

Namun, sebagian besar penelitian ini berfokus pada pengembangan dan validasi model deteksi sebagai sistem yang berdiri sendiri (*standalone*). Sistem-sistem tersebut menghasilkan data operasional penting seperti jumlah populasi, tetapi tidak terintegrasi secara langsung dengan sistem manajemen bisnis yang lebih luas seperti sistem informasi manajemen. Akibatnya, data hasil deteksi belum dapat dimanfaatkan secara otomatis untuk memperbarui modul lain, misalnya modul keuangan untuk menghitung *Feed Conversion Ratio* (FCR) aktual atau modul penjualan untuk merencanakan jadwal panen.

Perkembangan teknologi *computer vision* dalam peternakan ayam juga mencakup penggunaan berbagai arsitektur deep learning untuk meningkatkan akurasi deteksi dalam kondisi lingkungan yang kompleks. Elmessery dkk. mengembangkan model berbasis YOLOv8 yang diintegrasikan dengan citra termal dan visual untuk deteksi fenomena patologis pada ayam broiler di kandang intensif [10]. Model ini mampu mendeteksi kondisi seperti ayam lemas (lethargic), tendon tergelincir (slipped tendons), mata sakit, stres (paruh terbuka), dan pendulous crop dengan akurasi mAP50 mencapai 98,8% dan F1-score 97,2% pada dataset citra termal. Keunggulan penggunaan citra termal adalah kemampuannya untuk memantau ayam dalam kondisi pencahayaan rendah atau malam hari, yang

merupakan tantangan utama dalam sistem berbasis kamera visual konvensional. Namun, meskipun model ini menunjukkan performa deteksi yang sangat tinggi, implementasinya masih berfokus pada validasi akurasi model tanpa integrasi langsung dengan sistem manajemen operasional peternakan, sehingga data deteksi fenomena patologis belum dapat secara otomatis memicu tindakan korektif atau pembaruan data kesehatan dalam sistem informasi manajemen peternakan.

2.1.3. Deteksi Mortalitas Ayam Menggunakan *Computer Vision*

Selain perhitungan populasi, deteksi dini mortalitas menjadi aspek krusial untuk mencegah penyebaran penyakit dan menekan kerugian. Khanal dkk. mengembangkan sebuah sistem deteksi bangkai ayam otomatis menggunakan pendekatan dua tahap. Tahap pertama adalah mengidentifikasi objek stasioner (tidak bergerak) dalam rekaman CCTV, yang kemudian diverifikasi pada tahap kedua menggunakan *classifier* berbasis *Vision Transformer* (DeiT-Tiny) untuk memastikan objek tersebut adalah ayam. Metode yang divalidasi dengan akurasi 98,7% ini sangat efektif untuk lingkungan peternakan yang padat dan dinamis. Pendekatan serupa yang memanfaatkan *deep learning* untuk mendeteksi ayam mati juga telah diimplementasikan menggunakan arsitektur YOLO, yang menunjukkan tingkat presisi yang tinggi dalam identifikasi bangkai [11].

Namun, sama halnya dengan sistem pemantauan populasi, sistem deteksi mortalitas ini umumnya dirancang sebagai alat peringatan (*alert system*) yang terisolasi. Ketika bangkai terdeteksi, sistem hanya memberikan notifikasi tanpa adanya alur kerja yang terintegrasi. Informasi kematian tidak secara otomatis tercatat dalam modul produksi sistem informasi manajemen, tidak memperbarui data mortalitas harian, dan tidak memicu penyesuaian otomatis pada perhitungan biaya pakan yang terbuang. Keterbatasan ini membuat data mortalitas masih memerlukan input manual ke dalam sistem manajemen yang terpisah, sehingga mengurangi efisiensi dan responsivitas operasional.

Pengembangan sistem deteksi mortalitas juga telah mengeksplorasi penggunaan citra termal untuk mengatasi keterbatasan sistem berbasis kamera visual dalam kondisi pencahayaan rendah. Elmessery dkk. menunjukkan bahwa

integrasi citra termal dan visual dalam model YOLOv8 tidak hanya efektif untuk deteksi mortalitas, tetapi juga untuk identifikasi berbagai fenomena patologis yang merupakan indikator awal kematian ayam [10]. Sistem ini menggunakan teknik augmentasi data Mosaic yang terbukti meningkatkan performa model secara signifikan, dengan mAP50 mencapai 98,8% pada dataset citra termal dibandingkan 82,9% pada dataset visual. Kemampuan sistem untuk beroperasi dalam kondisi pencahayaan yang bervariasi termasuk malam hari memberikan keunggulan operasional yang signifikan. Namun, sama seperti sistem deteksi mortalitas lainnya, implementasi sistem ini masih bersifat standalone dan belum terintegrasi dengan alur kerja manajemen peternakan. Data deteksi tidak secara otomatis tercatat dalam modul produksi sistem informasi manajemen, tidak memperbarui data mortalitas harian, dan tidak memicu penyesuaian otomatis pada perhitungan biaya pakan yang terbuang, sehingga masih memerlukan input manual ke dalam sistem manajemen yang terpisah.

2.1.4. Pengembangan Aplikasi Manajemen Peternakan Ayam Broiler

Untuk melakukan pencatatan secara digital dan meminimalisir kesalahan dalam pencatatan manual, berbagai aplikasi smart farming telah dikembangkan. Muzakki dkk. merancang sebuah aplikasi berbasis Android yang memungkinkan petugas dan pemilik peternakan untuk melakukan pencatatan data ternak, jadwal pakan, serta membuat laporan produksi secara sistematis [12]. Aplikasi ini terbukti dapat meningkatkan efisiensi operasional dan akurasi pencatatan jika dibandingkan dengan metode manual. Fitur-fitur seperti pengelolaan inventaris pakan dan vitamin, serta pelaporan kematian dan hasil panen, membantu sentralisasi data manajemen peternakan dalam satu platform.

Namun, aplikasi manajemen seperti ini masih sangat bergantung pada input data manual dari petugas kandang. Petugas harus secara berkala menghitung dan memasukkan data populasi, jumlah kematian, dan stok pakan ke dalam aplikasi. Ketergantungan pada proses manual ini masih menyisakan ruang untuk kesalahan manusia (*human error*), inkonsistensi data, dan keterlambatan pembaruan informasi. Aplikasi ini telah menyediakan kerangka kerja digital untuk manajemen,

tetapi belum mengotomatisasi proses akuisisi data di tingkat operasional, yang merupakan sumber dari inefisiensi utama.

2.2. Celah Penelitian

Dari tinjauan pustaka di atas, dapat diidentifikasi bahwa:

1) Pada aspek manajemen pemeliharaan manual, proses perhitungan populasi, mortalitas, hingga pencatatan produksi masih dilakukan secara manual, sehingga rentan kesalahan, inkonsistensi, dan keterlambatan dalam pengambilan keputusan [3] [12].
2) Pada aspek teknologi *computer vision*, penelitian yang ada telah berhasil menghitung populasi dan mendeteksi ayam mati dengan akurasi tinggi, namun sistem yang dikembangkan masih berdiri sendiri (*standalone*) dan tidak terintegrasi dengan sistem manajemen bisnis yang lebih luas seperti sistem informasi manajemen. [8] [11] [10].
3) Pada aspek integrasi sistem belum ada penelitian yang mengintegrasikan *computer vision* untuk deteksi populasi dan mortalitas secara *real time* dengan sistem informasi manajemen khusus peternakan ayam broiler [4] [5].

Celah penelitian yang muncul adalah perlunya pengembangan sistem informasi manajemen khusus peternakan ayam broiler yang terintegrasi dengan teknologi *computer vision* untuk mengotomatisasi perhitungan populasi dan deteksi mortalitas secara *real-time*. Integrasi ini memungkinkan data hasil deteksi langsung memperbarui modul produksi, keuangan, dan penjualan tanpa input manual, sehingga meningkatkan efisiensi operasional, akurasi data, serta profitabilitas peternak.

Tabel 2.1 Tabel Perbandingan Penelitian

Penulis	Bahasa Pemrograman	Framework	Platform Aplikasi	Kelebihan	Kekurangan
Fadiellah & Perwitasari **[3]**	-	-	Manual	Gambaran nyata manajemen tradisional	Rentan kesalahan, lambat
Muzakki dkk. **[12]**	Kotlin	Firebase, IoT	*Smartphone*	Memudahkan pencatatan digital	Masih Input Manual
Khanal dkk. **[11]**	Python	PyTorch, DeiT	Server CCTV	Akurasi tinggi (98.7%)	Fokus hanya mortalitas, *standalone*
Pramudya dkk. **[9]**	Python	Darknet (YOLO)	Raspberry Pi & Kamera	Mendeteksi & ukur ayam	Fokus hanya pada ukuran ayam, tidak integrasi dengan sistem manajemen
Cruz dkk. **[8]**	Python	YOLOv10	*Edge Computing Device* & CCTV	Presisi tinggi (93.1%)	*Standalone*, tidak integrasi dengan sistem manajemen
Elmessery dkk. [10]	Python	YOLOv8	Server & Kamera Termal and Visual	Akurasi deteksi patologis tinggi (mAP50 98,8%)	Fokus validasi model tanpa integrasi sistem informasi manajemen.
Nalendra & Waspada **[5]**	Swift, C++	IoT & Mobile App	*Smartphone*	Monitoring real time suhu, kelembapan, level pakan & air	Terbatas aplikasi monitoring belum integrasi sistem informasi

					manajemen.
Sudarko & Herminingsih **[6]**			Peternakan plasma konvensional	Studi kapasitas agribisnis, akses informasi dan pemasaran	Kapasitas peternak rendah, bergantung pada perusahaan inti
Penulis	Dart & Golang	Flutter & Golang	Desktop & Smartphone (Multi-Plaform), CCTV + Server Inferensi Lokal + Integrasi Cloud	Integrasi penuh sistem informasi manajemen dengan implementasi *Computer Vision*, multi-modul (SCM, *Finance*, *Sales*, *Production Monitoring*), *real-time*	Membutuhkan resource tinggi (server + cctv)

Ringkasan dari 2.1 Perbedaan penelitian ini terhadap penelitian terdahulu

Penelitian terdahulu terkait manajemen pemeliharaan ayam broiler dan pemanfaatan teknologi mencakup berbagai pendekatan mulai dari metode manual hingga penggunaan teknologi komputer vision dan aplikasi smart farming. Fadiellah & Perwitasari [3] meneliti metode tradisional yang masih sangat bergantung pada aktivitas manual tanpa integrasi sistem komputer, sehingga rentan kesalahan dan keterlambatan. Muzakki dkk. [12] mengembangkan aplikasi Android berbasis Kotlin dengan framework Firebase dan IoT yang membantu pencatatan digital namun masih memerlukan input manual dan terbatas pada platform Android tanpa integrasi sistem informasi manajemen yang lengkap. Khanal dkk. [11] dan Pramudya dkk. [9] menggunakan bahasa Python dengan framework PyTorch dan Darknet (YOLO) untuk mendeteksi mortalitas dan ukuran ayam, masing-masing beroperasi secara standalone dengan akurasi tinggi (98,7% dan efektif dalam

pengukuran), namun tanpa integrasi dengan suatu sistem informasi manajemen. Cruz dkk. [8] juga menggunakan Python dan YOLOv10 dalam perangkat edge computing untuk hitung populasi ayam secara presisi (93,1%) namun masih belum terintegrasi dengan sistem manajemen lebih luas. Elmessery dkk. [10] melangkah lebih jauh dengan model YOLOv8 yang mengolah citra termal dan visual untuk deteksi patologis ayam dengan akurasi sangat tinggi (mAP50 98,8%) namun fokus pada validasi model tanpa integrasi sistem informasi manajemen operasional. Nalendra & Waspada [5] menggunakan teknologi IoT dan aplikasi mobile untuk monitoring suhu dan kelembaban secara real-time, namun tanpa integrasi *computer vision*. Sudarko & Herminingsih mengkaji kapasitas agribisnis peternak plasma yang masih rendah dan bergantung pada perusahaan inti tanpa membahas sistem digitalisasi secara spesifik [6]. Berbeda dengan seluruh penelitian tersebut, sistem yang dikembangkan oleh penulis menggunakan kombinasi bahasa Dart, Golang, dan Flutter untuk membangun aplikasi lintas platform (*Desktop* dan *Smartphone*) dengan integrasi penuh sistem informasi manajemen multi-modul yang meliputi manajemen manajemen gudang, keuangan, penjualan, dan monitoring produksi secara *real-time*. Sistem ini juga memadukan penggunaan CCTV dengan server inferensi lokal serta integrasi cloud, sehingga mampu otomatisasi deteksi populasi dan mortalitas ayam broiler yang sebelumnya belum terwujud dalam penelitian lain, walaupun membutuhkan resource komputasi yang lebih tinggi.

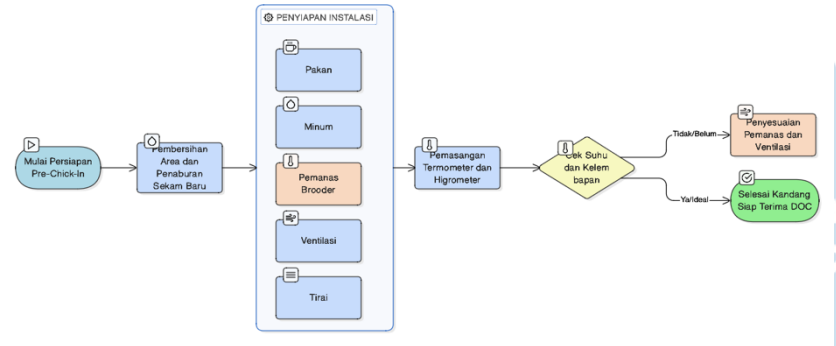
2.3. Landasan Teori

2.3.1. Proses Bisnis dan Manajemen Peternakan Ayam Broiler

Manajemen pemeliharaan ayam broiler merupakan serangkaian proses sistematis yang bertujuan untuk mengoptimalkan pertumbuhan ayam, menekan angka kematian (mortalitas), dan mencapai target *Feed Conversion Ratio* (FCR) yang efisien. Secara umum, alur proses bisnis operasional peternakan ayam broiler terbagi menjadi empat fase utama, yaitu:

2.3.1.1. Fase Persiapan Kandang (Pre-Chick-In)

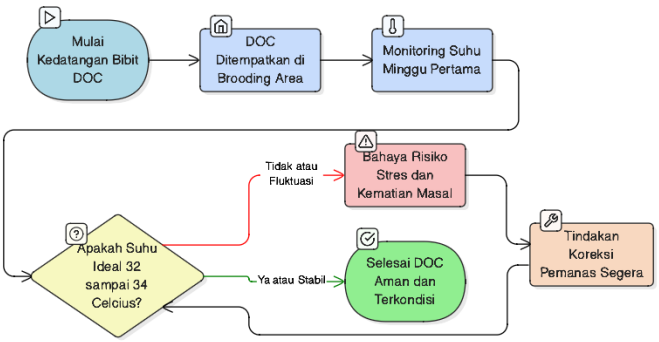
Sebelum bibit ayam atau *Day Old Chick* (DOC) tiba di lokasi, kandang harus dipersiapkan secara menyeluruh. Proses ini meliputi pembersihan area, penaburan sekam baru, serta penyiapan instalasi tempat pakan, tempat minum, mesin pemanas (*brooder*), kipas ventilasi, dan tirai kandang. Selain itu, pemasangan instrumen pengukur seperti termometer dan higrometer dilakukan untuk memastikan suhu dan kelembapan berada pada standar ideal [13].



Gambar 2.1 Alur Fase Persiapan Kandang

2.3.1.2. Fase Penerimaan Bibit (Chick-In)

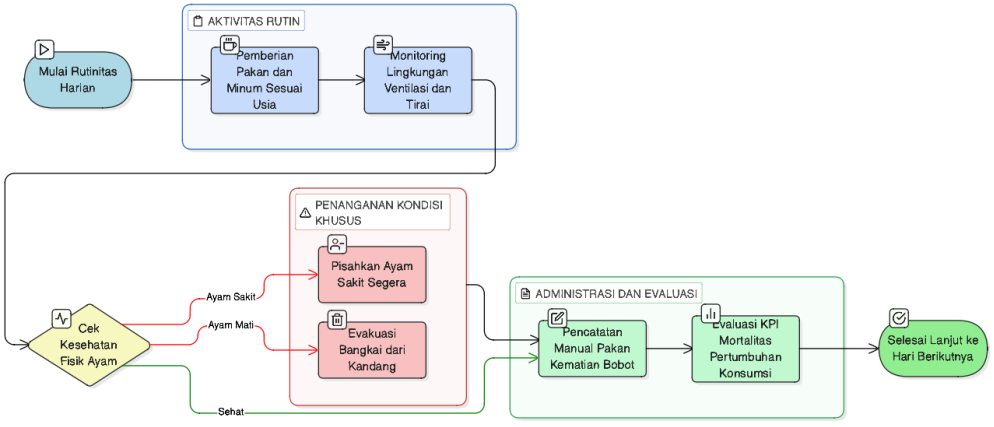
Fase ini merupakan fase paling kritis di mana DOC yang baru tiba sangat rentan terhadap perubahan suhu lingkungan. DOC langsung ditempatkan di dalam area pemanas (*brooding area*) dengan suhu ideal antara 32–34°C pada minggu pertama. Sedikit kesalahan dalam manajemen suhu dapat menyebabkan stres atau bahkan kematian masal pada anak ayam [13].



Gambar 2.2 Alur Fase Penerimaan Bibit *(Chick-In)*

2.3.1.3. Fase Pemeliharaan Harian

Pada fase ini, aktivitas operasional rutin dilakukan setiap hari yang meliputi pemberian pakan dan minum yang disesuaikan dengan takaran usia dan kebutuhan harian ayam. Selain itu, dilakukan pemantauan kesehatan dan lingkungan melalui pengecekan sirkulasi udara atau ventilasi, pengaturan tirai, serta pengecekan kesehatan fisik ayam secara berkala. Penanganan mortalitas juga menjadi fokus krusial, di mana ayam yang sakit akan segera dipisahkan, dan bangkai ayam yang mati langsung dievakuasi dari dalam kandang guna mencegah penyebaran penyakit. Dari sisi pencatatan administratif pada peternakan konvensional, pendataan seperti jumlah pakan yang terpakai, jumlah ayam mati, dan penimbangan bobot masih dilakukan secara manual. Sebagai tolak ukur keberhasilan, terdapat tiga indikator utama yang dievaluasi secara terus-menerus dalam fase ini, yaitu angka mortalitas, pertumbuhan bobot tubuh, dan tingkat konsumsi pakan [13].

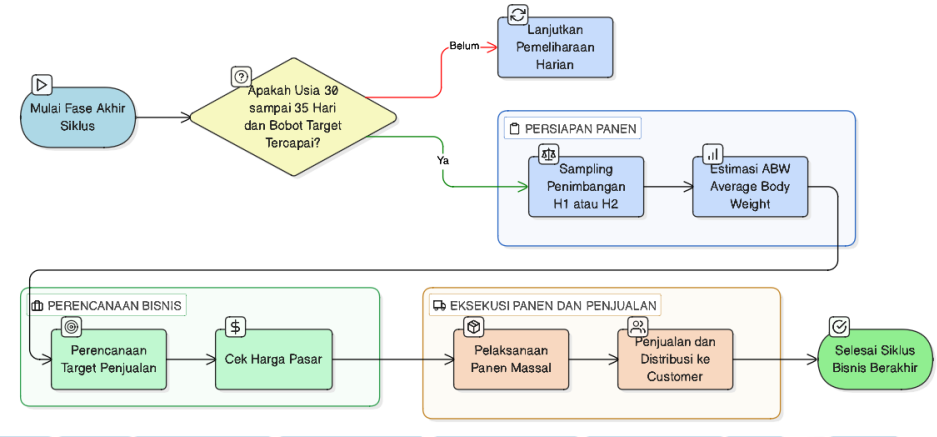


Gambar 2.3 Alur Fase Pemeliharaan Harian

2.3.1.4. Fase Panen dan Distribusi

Panen ayam broiler umumnya dilakukan ketika usia pemeliharaan mencapai 30 hingga 35 hari, tergantung pada target bobot yang diminta oleh pasar. Sekitar satu atau dua hari sebelum proses panen massal, peternak akan melakukan pengambilan sampel penimbangan (*Average Body Weight* / ABW) untuk mengestimasi bobot rata-rata ayam dalam satu kandang. Data *sampling* ini kemudian digunakan sebagai dasar perencanaan target penjualan harian dan

penyesuaian dengan harga pasar. Untuk proses penjualan dan pendistribusian dilakukan secara offline kepada customer dengan pertemuan secara langsung [13].



Gambar 2.4 Fase Panen dan Distribusi

2.3.2. Sistem Informasi Manajemen (SIM)

Sistem Informasi Manajemen (SIM) merupakan sistem informasi berbasis komputer yang dirancang untuk mendukung proses pengambilan keputusan, koordinasi, pengendalian, serta visualisasi informasi dalam suatu organisasi. SIM berfungsi mengintegrasikan data dari berbagai aktivitas operasional sehingga manajemen dapat memperoleh informasi yang relevan, akurat, dan tepat waktu. Sistem informasi manajemen memiliki peran penting dalam mendukung tiga tingkatan keputusan organisasi, yaitu *operational control*, *management control*, dan *strategic planning*. Dengan adanya SIM, organisasi mampu mengelola aliran data operasional, sumber daya, serta informasi strategis secara terpusat sehingga meningkatkan efisiensi dan efektivitas proses bisnis [14].

Dalam konteks penelitian ini, penerapan sistem informasi manajemen difokuskan pada industri peternakan ayam broiler untuk mengelola siklus kandang, logistik pakan, pencatatan keuangan, serta distribusi hasil produksi. Implementasi SIM diharapkan mampu memberikan dukungan keputusan berbasis data secara real-time sehingga operasional bisnis menjadi lebih terstruktur, transparan, dan optimal.

2.3.3. *Computer Vision*

Computer Vision merupakan salah jenis dari kecerdasan buatan yang bertujuan untuk melatih komputer agar dapat menafsirkan dan memahami dunia visual. Dengan menggunakan gambar digital dari kamera, video, dan model *deep learning* sehingga komputer dapat secara akurat mengidentifikasi dan mengklasifikasikan objek, lalu memberikan reaksi terhadap apa yang dilihat [15]. Tahapan yang harus dikerjakan dalam proses pembuatan model *computer vision* antara lain pengambilan gambar, pemrosesan, analisis, dan pengenalan objek [15]. Dalam penelitian ini, teknologi *computer vision* diterapkan untuk menganalisis rekaman dari kamera di dalam kandang ayam, dengan tujuan utama melakukan deteksi objek untuk melakukan perhitungan populasi dan deteksi indikasi kematian pada ayam.

2.3.4. Metode Waterfall

Metode Waterfall adalah salah satu metode klasik dalam *Software Development Life Cycle (SDLC)* yang mengadopsi pendekatan sekuensial dari analisis kebutuhan, desain sistem, implementasi, pengujian, hingga pemeliharaan. Meskipun banyak model pengembangan modern bermunculan, waterfall tetap relevan karena memberikan struktur sistematis dan prediktabilitas yang tinggi [16].

Alur metode ini dimulai dari tahap analisis kebutuhan untuk mendokumentasikan spesifikasi sistem secara rinci, yang kemudian diterjemahkan ke dalam cetak biru teknis pada tahap desain sistem. Rancangan tersebut selanjutnya direalisasikan menjadi kode program pada tahap implementasi, lalu divalidasi melalui pengujian untuk memastikan sistem bebas dari kesalahan. Siklus ini diakhiri dengan tahap pemeliharaan yang mencakup perbaikan atau penyesuaian sistem setelah digunakan oleh pengguna.

2.3.5. Flutter

Flutter merupakan framework cross-platform yang dikembangkan oleh Google Corporation dan dirilis pada tahun 2017. Framework ini memungkinkan pengembang untuk membuat aplikasi Web, mobile (Android dan iOS), serta

aplikasi desktop dan embedded menggunakan satu basis kode (code base). Flutter menggunakan Dart sebagai bahasa pemrogramannya, dimana Dart merupakan bahasa yang dioptimalkan untuk pembuatan antarmuka pengguna (user interface). Keunggulan utama dari teknologi cross-platform seperti Flutter adalah kemampuannya untuk mengurangi waktu yang dibutuhkan dalam mengantarkan produk akhir ke semua platform, yang secara langsung berdampak pada pengurangan sumber daya dan biaya pengembangan proyek. Selain itu, pengembang perangkat lunak multi-platform tidak perlu menguasai semua teknologi native untuk setiap platform target [17].

Dalam konteks arsitektur aplikasi, Flutter memiliki karakteristik unik dalam pengelolaan thread untuk operasi antarmuka pengguna. Aplikasi Flutter menggunakan empat thread yang bertanggung jawab terhadap user interface, yaitu main application thread (UI thread), Render thread, Flutter's UI thread, dan Raster thread. Hal ini berbeda dengan aplikasi native Android yang hanya menggunakan dua thread untuk keperluan yang sama. Meskipun Flutter menawarkan kemudahan pengembangan cross-platform, hasil penelitian menunjukkan bahwa aplikasi native Android masih lebih efisien secara waktu dibandingkan aplikasi Flutter, terutama dalam hal waktu kerja UI threads dan waktu startup aplikasi. Perbedaan performa ini terutama terlihat signifikan pada saat inisialisasi aplikasi, dimana Flutter harus melakukan tugas-tugas berat yang menyebabkan peningkatan waktu startup aplikasi [17].

2.3.6. Dart

Pengembangan aplikasi *mobile* dapat dilakukan dengan dua cara, yaitu natif (seperti Java untuk satu platform spesifik) dan multi-platform (cross-platform), seperti Dart dengan framework Flutter yang memungkinkan satu basis kode untuk beroperasi di berbagai sistem, seperti Android dan iOS. Analisis perbandingan antara Java dan Dart menjadi penting untuk menentukan teknologi yang paling sesuai untuk sebuah aplikasi, dengan mempertimbangkan metrik performa seperti beban CPU, konsumsi RAM, konsumsi baterai, dan waktu eksekusi , serta aspek kualitatif seperti struktur kode dan dukungan komunitas. Penelitian dalam jurnal ini

menyimpulkan bahwa sulit untuk menentukan satu bahasa yang lebih unggul secara mutlak; Java menunjukkan keunggulan dalam efisiensi RAM , sementara Dart (Flutter) menawarkan efisiensi pengembangan lintas platform yang signifikan. Dalam konteks penelitian ini, perbandingan teknologi tersebut relevan untuk memilih arsitektur aplikasi mobile yang paling optimal untuk sistem informasi manajemen peternakan [18].

2.3.7. Golang

Golang adalah bahasa pemrograman modern yang dikembangkan oleh Google dengan fokus pada kesederhanaan, efisiensi, dan dukungan *concurrency*. Go telah menjadi bahasa utama dalam pengembangan sistem mikroservis di skala industri karena memiliki performa tinggi, sintaks sederhana, dan mendukung *goroutines* untuk pemrograman paralel [19]. Hal ini menjadikan Golang sangat sesuai untuk implementasi backend pada aplikasi smart farming yang membutuhkan pemrosesan data *real-time* dari perangkat *edge* dan *server cloud*.

2.3.8. PostgreSQL

PostgreSQL adalah sistem manajemen basis data relasional berorientasi objek (*object-relational database management system/ORDBMS*). PostgreSQL dirancang untuk mendukung tipe data kompleks, ekstensi pengguna, trigger, serta pemrosesan data aktif [20]. Dengan fitur skalabilitas tinggi, PostgreSQL sangat cocok digunakan dalam sistem smart farming untuk menyimpan data deteksi ayam secara real-time baik di server lokal maupun cloud.