Sistem Pencatatan Digital Untuk Ternak Lele

1st Annisa Elfia Rosa
Prodi S1 Teknik Telekomunikasi
Universitas Telkom Purwokerto
Purwokerto, Indonesia
annisaelfiarosa@student.telkomuniversity.ac.id

2nd Fitri Amalia
Prodi S1 Teknik Telekomunikasi
Universitas Telkom Purwokerto
Purwokerto, Indonesia
fitriamaliafa@student.telkomuniversity.ac.id

3rd Hana Nusaibah
Prodi S1 Teknik Telekomunikasi
Universitas Telkom Purwokerto
Purwokerto, Indonesia
hananusaibah@student.telkomuniversity.ac.id

4th Irmayatul Hikmah
Prodi S1 Teknik Telekomunikasi
Universitas Telkom Purwokerto
Purwokerto, Indonesia
irmayatulhikmah@student.telkomuniversity.ac.id

5th Jafaruddin Gusti Amri Ginting
Prodi S1 Teknik Telekomunikasi
Universitas Telkom Purwokerto
Purwokerto, Indonesia
jafaruddingusti@student.telkomuniversity.ac.id

***Abstrak* — Budidaya ikan lele merupakan salah satu sektor perikanan air tawar yang banyak diminati, namun pencatatan aktivitas budidaya masih sering dilakukan secara manual sehingga rawan menimbulkan kehilangan data, kesalahan pencatatan, dan kesulitan dalam analisis keuntungan usaha. Penelitian ini bertujuan merancang dan mengimplementasikan Sistem Pencatatan Digital untuk Ternak Lele berbasis web yang terintegrasi dengan teknologi Machine Learning untuk prediksi keuntungan budidaya. Sistem dikembangkan menggunakan framework Laravel 9 sebagai backend, MySQL sebagai basis data, serta layanan Machine Learning berbasis Python FastAPI dengan algoritma Ridge Regression untuk menghasilkan prediksi keuntungan berdasarkan data historis budidaya. Fitur utama sistem meliputi pengelolaan data kolam, siklus budidaya, pakan, pertumbuhan, keuangan, dashboard, laporan, serta prediksi keuntungan. Pengujian dilakukan terhadap aspek ekonomi, teknologi, human, durability, scalability, fungsionalitas, dan akurasi model Machine Learning. Hasil pengujian menunjukkan model Ridge Regression dengan 17 fitur input mencapai R^2 sebesar 0,89, MAE Rp663.662, dan RMSE Rp788.896 pada 127 data siklus budidaya, melampaui target spesifikasi sebesar ≥85%. Pengujian usability dengan metode System Usability Scale (SUS) terhadap 30 responden menghasilkan skor 85,58 (kategori Good–Excellent), sedangkan pengujian fungsionalitas terhadap 54 skenario uji menghasilkan tingkat keberhasilan 99,35%. Hasil tersebut menunjukkan bahwa sistem mampu menggantikan pencatatan manual menjadi digital serta menyediakan dukungan pengambilan keputusan berbasis data yang lebih efektif bagi peternak lele.**

***Kata kunci*— budidaya lele, laravel, machine learning, prediksi keuntungan, ridge regression, sistem pencatatan digital**

I. PENDAHULUAN

Budidaya ikan lele merupakan salah satu sektor perikanan air tawar yang memiliki kontribusi penting dalam memenuhi kebutuhan pangan masyarakat Indonesia. Tingginya permintaan pasar terhadap ikan lele mendorong banyak pelaku usaha untuk mengembangkan kegiatan budidaya secara berkelanjutan. Dalam proses budidaya, peternak perlu melakukan berbagai aktivitas pengelolaan seperti pencatatan data kolam, pemberian pakan, pemantauan pertumbuhan ikan, pencatatan biaya operasional, serta pendokumentasian hasil panen [1].

Namun, pada praktiknya masih banyak peternak yang melakukan pencatatan aktivitas budidaya secara manual menggunakan buku catatan sederhana. Kondisi tersebut menyebabkan peternak mengalami kesulitan dalam pengelolaan data, evaluasi hasil budidaya, serta analisis keuntungan usaha. Data yang tidak terdokumentasi dengan baik juga meningkatkan risiko kesalahan pencatatan dan kehilangan informasi penting yang dibutuhkan dalam proses pengambilan keputusan [2].

Beberapa solusi sejenis yang telah tersedia, seperti eFishery, FishGo, dan Aquaculture Manager, telah menyediakan fitur pendukung budidaya, namun masing-masing masih berfokus pada aspek tertentu seperti otomatisasi pakan atau pemantauan lingkungan, dan belum ada yang secara khusus mengintegrasikan pencatatan kolam, pakan, pertumbuhan, keuangan, dan laporan sekaligus prediksi keuntungan dalam satu sistem yang sederhana bagi peternak skala kecil hingga menengah. Berdasarkan permasalahan tersebut, penelitian ini mengembangkan Sistem Pencatatan Digital untuk Ternak Lele (ExpertLele) berbasis website yang dilengkapi dengan layanan Machine Learning untuk prediksi keuntungan budidaya, sehingga peternak dapat melakukan pencatatan data secara terstruktur sekaligus memperoleh estimasi keuntungan sebagai dasar pengambilan keputusan.

Tujuan dari penelitian ini adalah: (1) mengembangkan sistem pencatatan budidaya lele berbasis website yang mampu mengelola data kolam, pakan, pertumbuhan, keuangan, dan laporan secara terintegrasi; (2) merancang antarmuka pengguna yang sederhana dan mudah digunakan; (3) mengimplementasikan backend yang mampu mengelola penyimpanan dan pengolahan data secara efektif; (4) menyediakan visualisasi data dan dashboard pemantauan budidaya; serta (5) melakukan pengujian terhadap aspek UI/UX, backend, dan akurasi model Machine Learning. Penelitian ini dibatasi pada pengelolaan data kolam, pakan, pertumbuhan ikan, keuangan, dan laporan budidaya, dengan

fokus pengujian pada aspek antarmuka pengguna (UI/UX) dan backend website.

II. KAJIAN TEORI

A. Budidaya Ikan Lele dan Sistem Pencatatan Digital

Budidaya ikan lele merupakan kegiatan pemeliharaan ikan lele yang dilakukan secara terencana mulai dari pembenihan, pembesaran, pemberian pakan, hingga panen. Proses pencatatan yang baik membantu peternak mengetahui perkembangan budidaya dari waktu ke waktu, mengevaluasi penggunaan pakan, tingkat pertumbuhan, dan kondisi keuangan usaha [1]. Sistem informasi berbasis web memungkinkan data budidaya tersimpan secara terpusat, mudah diakses, dan diolah menjadi informasi yang lebih bermanfaat, sehingga dapat meningkatkan efisiensi operasional dan mendukung pengambilan keputusan berbasis data [2].

B. Website

Website dibangun di atas HTML sebagai struktur konten [14], PHP sebagai bahasa pemrograman sisi server [15], dan Bootstrap sebagai kerangka kerja CSS responsif [16]. *User Interface* (UI) mencakup elemen visual (tata letak, warna, tipografi) yang memudahkan interaksi, sedangkan *User Experience* (UX) berfokus pada kenyamanan dan kepuasan pengguna selama berinteraksi dengan sistem [9]. Dashboard menyajikan informasi ringkas dalam bentuk grafik dan tabel agar pengguna dapat memantau kondisi sistem secara cepat [8], dengan Chart.js sebagai *library* JavaScript berbasis HTML5 Canvas untuk grafik interaktif [18].

C. Dashboard dan Visualisasi Data

Dashboard merupakan tampilan visual yang digunakan untuk menyajikan informasi penting secara ringkas dan terstruktur dalam satu halaman, sehingga pengguna dapat memantau kondisi suatu sistem secara cepat tanpa harus menelusuri data secara manual [8]. Pada sistem pencatatan budidaya, dashboard berperan menampilkan ringkasan populasi ikan, jumlah kolam, total biaya, keuntungan, jadwal panen, serta hasil prediksi keuntungan dari model Machine Learning.

D. Framework Laravel, Basis Data MySQL, dan REST API

Laravel merupakan framework PHP berbasis arsitektur Model-View-Controller (MVC) yang banyak digunakan dalam pengembangan sistem informasi berbasis web karena menyediakan struktur kode yang terorganisasi serta berbagai fitur bawaan untuk autentikasi, validasi data, dan pengelolaan basis data [6]. Operasi dasar pengelolaan data pada sistem menggunakan prinsip CRUD (Create, Read, Update, Delete) sebagai fondasi fitur manajemen data [7]. Basis data MySQL digunakan sebagai media penyimpanan terpusat untuk seluruh data budidaya. Komunikasi antara backend dan layanan eksternal dilakukan melalui REST API dengan format pertukaran data JSON, yang bersifat ringan dan mudah diurai baik oleh manusia maupun mesin.

E. *Machine Learning* dan *Ridge Regression*

Machine Learning merupakan salah satu cabang kecerdasan buatan yang memungkinkan komputer mempelajari pola dari data dan menghasilkan prediksi tanpa harus diprogram secara eksplisit untuk setiap kondisi. Melalui proses pembelajaran dari data historis, model Machine Learning dapat mengenali hubungan antar variabel dan menggunakannya untuk memprediksi data baru [5]. Ridge Regression merupakan pengembangan dari Linear Regression dengan menambahkan regularisasi L2 untuk mengurangi risiko overfitting. Regularisasi dilakukan dengan memberi penalti terhadap nilai koefisien sehingga model menghasilkan prediksi yang lebih stabil ketika menangani data dengan banyak variabel atau hubungan antar-variabel yang kuat [5]. Algoritma ini dipilih karena fitur-fitur pada dataset penelitian ini secara alami memiliki korelasi tinggi—misalnya feed_cost dengan feed_cost_per_seed, serta mortality_rate dengan survival_rate—sehingga Linear Regression biasa berpotensi menghasilkan koefisien yang tidak stabil. Penggunaan varian RidgeCV memungkinkan sistem memilih nilai parameter regularisasi (alpha) secara otomatis melalui validasi silang tanpa penyesuaian manual.
Evaluasi performa model regresi pada penelitian ini menggunakan tiga metrik utama [5], [25]: MAE (Mean Absolute Error) yang mengukur rata-rata selisih absolut antara nilai prediksi dan aktual; RMSE (Root Mean Squared Error) yang merupakan akar dari rata-rata kuadrat selisih sehingga lebih sensitif terhadap kesalahan besar; serta R^2 (Coefficient of Determination) yang mengukur proporsi variansi target yang dapat dijelaskan oleh model, dengan nilai mendekati 1 menunjukkan performa yang lebih baik.

F. Pengujian Sistem

Pengujian sistem mencakup functional testing menggunakan metode black-box [11], usability testing dengan metode System Usability Scale (SUS) [12], dan performance testing untuk mengukur waktu respons dan kapasitas layanan di bawah beban kerja tertentu [11].

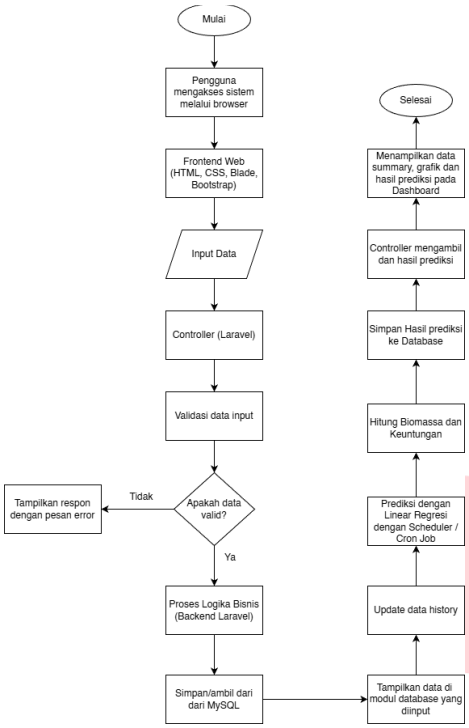
III. METODE

Penelitian ini menggunakan metode pengembangan perangkat lunak dengan tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Sumber data diperoleh dari data primer hasil siklus budidaya lele yang dicatat melalui sistem, meliputi data kolam, pakan, pertumbuhan, dan keuangan.

A. Rancangan Penelitian dan Arsitektur Sistem

Penelitian dilaksanakan melalui tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Sistem terdiri atas empat komponen: frontend (Blade template, antarmuka interaksi pengguna), backend (Laravel 9, mengelola logika bisnis, autentikasi, validasi, serta komunikasi dengan basis data dan layanan ML), database (MySQL, menyimpan data pengguna, kolam, siklus, pakan, pertumbuhan, keuangan), dan layanan machine learning (Python FastAPI, memproses data historis untuk prediksi keuntungan).

Basis data terdiri atas enam model utama — User, Pond, Cycle, Pakan, Pertumbuhan, dan Keuangan — yang saling berelasi melalui Eloquent ORM. Model User memiliki relasi hasMany ke lima model turunan, sedangkan model Cycle (merepresentasikan satu periode budidaya tebar–panen) memiliki relasi belongsTo ke User dan Pond, serta hasMany ke Pakan, Pertumbuhan, dan Keuangan.



GAMBAR 1
(Arsitektur Utama Sistem)

B. Model Machine Learning

Model prediksi dibangun sebagai layanan API terpisah menggunakan algoritma RidgeCV (Ridge Regression dengan pemilihan parameter regularisasi *alpha* otomatis melalui validasi silang), dengan pertimbangan bahwa dataset mengandung fitur-fitur yang secara alami berkorelasi tinggi (misalnya *feed_cost* dengan *feed_cost_per_seed*). Model dilatih menggunakan 18 fitur input yang terbagi menjadi 9 fitur langsung dari basis data (*duration_days, seed_count, feed_cost, feed_qty_raw, other_cost, last_avg_weight, total_death_count, last_remaining_count, pond_size*) dan 9 fitur turunan hasil rekayasa fitur (*mortality_rate, survival_rate, feed_cost_per_seed, other_cost_per_seed, feed_qty_per_seed, seed_density, fcr, adg, estimated_price_per_kg*), dengan target prediksi berupa nilai *profit*. Tingkat kematian dan kelangsungan hidup dihitung dengan persamaan 1 dan persamaan 2:

$$mortality_rate = \frac{total_death_count}{seed_count} \quad (1)$$

$$survival_rate = \frac{last_remaining_count}{seed_count} \quad (2)$$

Data dibagi menjadi 80% data latih dan 20% data uji. Pipeline pelatihan dibangun menggunakan Pipeline dan ColumnTransformer dari Scikit-learn yang menggabungkan SimpleImputer (penanganan nilai kosong), StandardScaler (normalisasi fitur), dan RidgeCV (estimator regresi), sehingga preprocessing dan training konsisten tanpa risiko *data leakage*. Model dievaluasi menggunakan metrik R^2 (koefisien determinasi), MAE (*Mean Absolute Error*), dan RMSE (*Root Mean Squared Error*):

$$error\ (\%) = \frac{Aktual - Prediksi}{Aktual}\ x\ 100\% \quad (3)$$

C. Model Pengujian

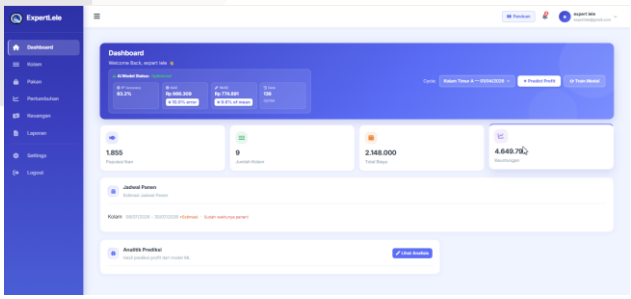
Pengujian dilaksanakan secara daring pada domain produksi selama April–Mei 2026, mencakup: (1) *black-box testing* terhadap 54 skenario fungsional pada 11 kategori modul; (2) *performance testing* menggunakan skrip pengiriman *request* berurutan dan Apache Bench untuk simulasi beban bersamaan; (3) *usability testing* menggunakan kuesioner SUS terhadap 30 responden; (4) *security testing* untuk verifikasi *ownership check* data antar pengguna; (5) *compatibility testing* pada resolusi desktop (1920×1080 px) dan mobile (375×812 px); serta (6) pengujian akurasi model ML secara bertahap dengan menambah volume data pelatihan.

IV. HASIL DAN PEMBAHASAN

A. Implementasi Sistem

Sistem diimplementasikan dengan alur kerja: pengguna mengirim permintaan melalui frontend, diteruskan ke backend Laravel untuk divalidasi dan diproses, lalu backend membaca/menulis data ke MySQL. Pada fitur prediksi, backend mengambil data historis dari database view v_profit_dataset (agregasi otomatis dari tabel cycles, pakan, pertumbuhan, dan keuangan menjadi satu baris per siklus), mengirimkannya ke layanan ML melalui endpoint POST /predict, dan model Ridge Regression yang telah dilatih mengembalikan nilai predicted_profit beserta daftar fitur yang digunakan sebagai bentuk transparansi kepada pengguna. Endpoint /predict hanya aktif setelah model dilatih minimal satu kali melalui POST /train, yang mensyaratkan minimal 10 siklus berstatus selesai (DONE) dengan data profit lengkap.

Sistem dapat diakses oleh peternak untuk mengelola data budidaya lele secara terstruktur. Halaman dashboard merupakan halaman utama sistem yang menampilkan ringkasan informasi budidaya secara visual, meliputi jumlah populasi ikan, jumlah kolam, total biaya, keuntungan, jadwal panen, serta status dan hasil analitik prediksi dari model Machine Learning, seperti ditunjukkan pada Gambar 2. Tata letak dashboard dirancang dengan struktur navigasi sederhana berupa sidebar yang menghubungkan pengguna ke modul kolam, pakan, pertumbuhan, keuangan, dan laporan, sehingga pengguna dapat mengakses informasi utama secara lebih efisien.



GAMBAR 2
(Tampilan Dashboard Sistem)

Selain dashboard, sistem menyediakan modul pengelolaan kolam dan siklus budidaya, modul pencatatan pakan dengan penghitungan otomatis FCR (Feed Conversion Ratio), modul pencatatan pertumbuhan ikan, modul keuangan untuk mencatat pemasukan dan pengeluaran

(termasuk kategori obat dan infrastruktur), serta modul laporan yang merekapitulasi data budidaya per siklus.

B. Hasil Pengujian Fungsionalitas Sistem

Pengujian black-box terhadap 54 skenario pada 11 kategori modul (autentikasi, manajemen kolam, manajemen siklus, pencatatan pakan/pertumbuhan/keuangan, dashboard, machine learning, laporan & ekspor Excel, keamanan sistem, dan responsivitas UI) menghasilkan tingkat keberhasilan 99,35%, dengan rincian pada TABEL 1.

TABEL 1
(Pengujian Fungsionalitas)

Akses Menu	**Test Case**	**Hasil yang Diharapkan**	**Keberhasilan**	
			Ya	**Tidak**
User				
Login	01	*Website* dapat menampilkan halaman login dan menerima kredensial pengguna yang valid	✓	
Registrasi	02	*Website* dapat membuat akun baru dengan data valid dan mengarahkan pengguna ke *dashboard*	✓	
Dashboard	03	*Website* menampilkan ringkasan data kolam, populasi ikan, biaya, dan estimasi keuntungan	✓	
Kolam	04	*Website* dapat menampilkan, menambah, mengubah, dan menghapus data kolam milik pengguna	✓	
Siklus	05	*Website* dapat membuat siklus baru, mengubah status siklus, dan menutup siklus dengan data panen	✓	
Pakan	06	*Website* dapat mencatat data pemberian pakan dan mengkalkulasi total biaya pakan secara otomatis	✓	
Pertumbuhan	07	*Website* menampilkan data pertumbuhan ikan dan menghitung indikator ADG, FCR, dan Survival Rate secara otomatis	✓	
Keuangan	08	*Website* dapat mencatat transaksi pemasukan dan pengeluaran serta menghitung saldo akhir dengan benar	✓	
Laporan	09	*Website* menampilkan rekapitulasi laporan siklus selesai dan dapat mengekspor laporan ke format Excel	✓	
Prediksi Keuntungan	10	*Website* menampilkan prediksi keuntungan dari model *Machine Learning Ridge Regression* pada siklus aktif	✓	
Settings	11	*Website* menampilkan halaman pengaturan profil dan penggantian password yang sesuai dengan jenis akun (email/Google)	✓	
Keamanan Data	12	*Website* tidak mengizinkan pengguna mengakses data milik pengguna lain (mengembalikan HTTP 403 Forbidden)	✓	

Mekanisme *ownership check* pada setiap controller mencegah pengguna mengakses data milik pengguna lain (mengembalikan HTTP 403 Forbidden), yang berhasil pada seluruh skenario keamanan. Pengujian kompatibilitas pada enam skenario resolusi (desktop dan mobile) untuk halaman *Landing Page*, Dashboard, Pakan, dan Laporan juga mencapai keberhasilan 100%, menunjukkan konsistensi desain responsif berbasis *media query* di seluruh modul sistem.

C. Hasil Pengujian Perfoma

Pengujian *processing time* menggunakan skrip pengiriman 1.000 *request* berurutan ke endpoint /predict menghasilkan total waktu 7,119 detik atau rata-rata 7,12 ms/request, jauh di bawah target spesifikasi ≤5 detik/request. Pengujian beban bersamaan menggunakan Apache Bench dengan 10 *concurrent users* dan 100 total *request* menghasilkan 0 failed requests, *throughput* 49,38 request/detik, waktu respons rata-rata 202,5 ms (median 196 ms, maksimum 373 ms) sebagaimana dirangkum pada TABEL 2.

TABEL 2
(Hasil Pengujian Performa)

Metrik	**Nilai**
Concurrency Level	10 users
Total Request	100 request
Failed Requests	0 (0%)
Time taken for tests	2,025 detik
Requests per second	49,38 req/s
Time per request (mean)	202,504 ms
Time per request (concurrent)	20,250 ms
Median response time (50th)	196 ms
Max response time (100th)	373 ms

Latensi jaringan tidak menjadi hambatan utama (rata-rata waktu koneksi ≈0 ms); bottleneck berada pada komputasi prediksi model, yang tetap dalam batas wajar untuk layanan ML *real-time*.

D. Hasil Pengujian Usability

Pengujian SUS terhadap 30 responden dengan 10 pernyataan berskala 1–5 menghasilkan total skor kontribusi (ΣR) sebesar 1.027, sehingga:

$$U = \frac{\Sigma R \; x \; 2{,}5}{n} = \frac{1027 \; x \; 2{,}5}{30} = 85{,}58 \qquad (4)$$

Skor 85,58 termasuk kategori *Excellent* (>80,3) pada skala interpretasi SUS, berada pada persentil ke-80 dibandingkan basis data sistem yang pernah diuji secara global dengan metode yang sama, menunjukkan sistem dinilai mudah digunakan dan memuaskan oleh mayoritas responden.

E. Hasil Pengujian Akurasi Model *Machine Learning*

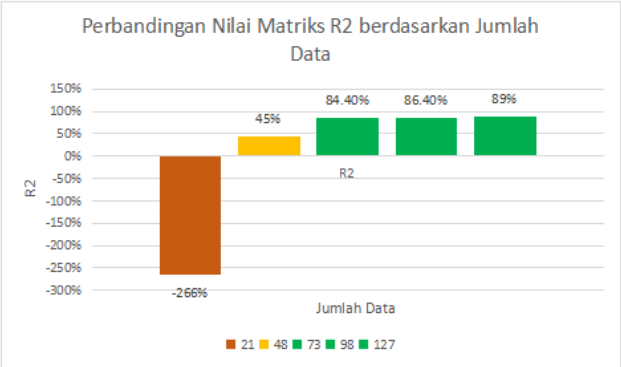
Evaluasi model dilakukan bertahap dengan menambah volume data pelatihan, sebagaimana disajikan pada TABEL 3.

TABEL 3
(Hasil Pengujian Akurasi Model Machine Learning)

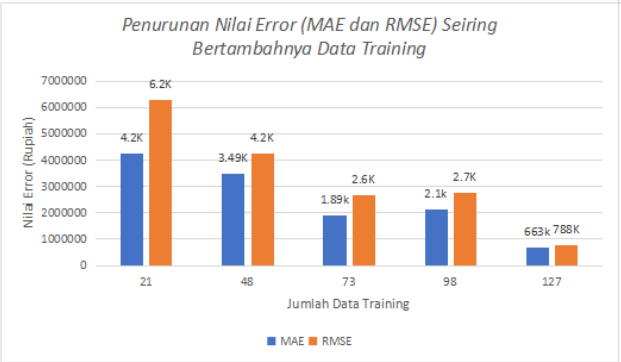
No	Jumlah Data	R^2 Score	MAE (Rp)	MAPE (%)	RMSE (Rp)	NRMSE(%)	Ket
1	21	-2,660	4.257.878	83.3%	6.289.329	123.1%	Tidak Layak
2	48	0,450	3.492.478	68,4%	4.254.776	83.3%	Cukup
3	73	0,844	1.899.752	37,2%	2.649.639	51.9%	Baik
4	98	0,864	2.117.700	31.5%	2.773.684	54,3%	Baik
5	127	0,890	663.662	13.0%	788.896	15,4%	Baik

Pada 21 siklus data, nilai R^2 negatif menandakan model belum mampu mengestimasi 17–18 koefisien fitur secara andal akibat data yang belum mencukupi. Titik infleksi terjadi pada kisaran 70–75 siklus (R^2 melonjak dari 0,450 pada 48 data menjadi 0,844 pada 73 data), mengindikasikan model mulai menangkap pola representatif dari data budidaya. Pada 127 siklus data (102 latih, 25 uji), model mencapai R^2 = 0,89, melampaui target spesifikasi ≥85%, dengan MAE turun 84,4% dan MAPE rata-rata 13,3% terhadap data aktual pada 10 sampel pengujian langsung. Pengujian latensi model tunggal menunjukkan waktu respons 1,897 ms per *request*, membuktikan model Ridge Regression yang di-*deploy* pada FastAPI memiliki performa inferensi ringan dan layak digunakan secara *real-time*.

Dari hasil pengujian Matriks R^2 dapat dilihat perbandingan dari setiap data yang diinput melalui diagram berikut :



GAMBAR 3
(Grafik Perbandingan Nilai R^2)



GAMBAR 4
(Grafik Penurunan Error (MAE/RMSE))

V. KESIMPULAN

Sistem Pencatatan Digital untuk Ternak Lele berhasil dikembangkan sebagai aplikasi web berbasis Laravel 9 dengan arsitektur MVC, basis data MySQL, dan layanan cloud hosting, yang mengintegrasikan pengelolaan data kolam, siklus budidaya, pakan, pertumbuhan, keuangan, dan laporan ke dalam satu platform terstruktur sehingga berhasil menggantikan pencatatan manual menjadi digital. Implementasi *machine learning* dengan algoritma Ridge Regression berhasil memprediksi keuntungan budidaya berdasarkan 18 fitur operasional, mencapai R^2 sebesar 0,89, MAE Rp 663.662, dan RMSE Rp 788.896 pada 127 siklus data, melampaui target spesifikasi yang ditetapkan. Hasil pengujian menunjukkan sistem memenuhi seluruh spesifikasi yang ditetapkan: tingkat keberhasilan fungsional 99,35% dari 54 skenario uji, skor *usability* SUS 85,58 (kategori *Excellent*), serta performa layanan prediksi yang responsif dengan 0 kegagalan pada pengujian beban bersamaan. Secara keseluruhan, sistem terbukti mampu mendukung efisiensi dan keberlanjutan usaha budidaya lele di Indonesia melalui pendekatan pencatatan yang terintegrasi dan berbasis data.

REFERENSI

[1] R. Kurniawan, A. Nugroho, dan D. A. Prasetya, "Sistem Informasi Pencatatan Budidaya Ikan Lele Berbasis Web," *Jurnal Teknologi dan Sistem Informasi*, vol. 6, no. 2, pp. 85-94, 2020.

[2] L. L. Atamina, Sudjiono, dan D. Riningsih, "Model Akuntansi Sederhana untuk Penyelenggaraan Pembukuan Bagi Peternak Lele di Kelurahan Ngronggo Kota Kediri," *Jurnal Pengabdian Kepada Masyarakat*, vol. 2, no. 3, pp. 366-371, Sep. 2021.

[5] E. Sakina dan A. H. Mirza, "Prediksi Hasil Produksi Ikan Lele Menggunakan Machine Learning," *Informatics of*

Science and Technology Journal, vol. 7, no. 1, pp. 22-30, 2024.
[6] D. P. Sari dan A. F. Rochim, "Pengembangan Sistem Informasi Manajemen Perikanan Berbasis Web Menggunakan Framework Laravel," *J-PTIIK*, vol. 6, no. 4, pp. 1821-1829, 2022.
[7] F. A. Ramadhan dan N. Hidayah, "Implementasi Sistem CRUD pada Aplikasi Manajemen Data Berbasis Web," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 5, no. 3, pp. 211-219, 2019.
[8] I. Wahyudi dan R. Lestari, "Perancangan Dashboard Monitoring Data Menggunakan Bootstrap pada Sistem Informasi," *Jurnal Media Informatika*, vol. 8, no. 1, pp. 1-9, 2022.
[9] P. W. Fu'adah dan A. C. Padmasari, "Perancangan Desain UI pada Aplikasi Tailon," *Jurnal Desain Komunikasi Visual*, vol. 10, no. 1, pp. 1-10, 2023.
[10] H. Riyadi, "Pengertian MySQL Beserta Fungsi dan Sejarah Terbentuknya MySQL," 2019.
[11] A. W. Nugraha, B. Santoso, dan H. T. Sukmana, "Analisis Kinerja Aplikasi Web Menggunakan Black Box Testing dan Apache JMeter," *Jurnal Informatika dan Sistem Informasi*, vol. 7, no. 2, pp. 101-110, 2021.
[12] A. P. Wicaksono, D. K. Sari, dan M. A. Hakim, "Pengujian Usability Aplikasi Berbasis Web Menggunakan Metode System Usability Scale (SUS)," *Jurnal Teknologi Informasi dan Komunikasi*, vol. 10, no. 2, pp. 134-142, 2021.
[13] S. Mulyani, R. P. Pratama, dan A. R. Hakim, "Penerapan Data Flow Diagram dan Entity Relationship Diagram dalam Perancangan Sistem Informasi," *Jurnal Sistem Informasi dan Komputer*, vol. 9, no. 1, pp. 27-35, 2020.
[14] M. Husni dan N. Fitri, "Pengantar Pengembangan Web: HTML, CSS, dan JavaScript untuk Pemula," *Jurnal Informatika Upgris*, vol. 7, no. 1, pp. 45-53, 2021.
[15] A. Pranata, R. Fauzi, dan D. Kurniawan, "Implementasi PHP dan MySQL dalam Pengembangan Aplikasi Sistem Informasi Berbasis Web," *Jurnal Ilmu Komputer dan Informatika*, vol. 3, no. 2, pp. 98-107, 2020.
[16] R. Navissa, A. Syahputra, dan B. Hermawan, "Pemanfaatan Framework Bootstrap dalam Pengembangan Antarmuka Sistem Informasi Responsif," *Jurnal Teknik Komputer AMIK BSI*, vol. 8, no. 1, pp. 56-63, 2022.
[18] M. A. Fauzi dan R. D. Septiana, "Implementasi Chart.js untuk Visualisasi Data pada Sistem Informasi Berbasis Web," *Jurnal Ilmiah Teknologi Informasi Terapan*, vol. 9, no. 2, pp. 112-120, 2023.
[19] B. Otwell. *Laravel: The PHP Framework for Web Artisans — Documentation*. (2023). [Online]. Available: https://laravel.com/docs
[20] Google LLC. *Google Identity: OAuth 2.0 for Web Server Applications*. (2023). [Online]. Available: https://developers.google.com/identity/protocols/oauth2
[21] Maatwebsite. *Laravel Excel — Supercharged Excel Exports and Imports in Laravel*, versi 3.x. (2023). [Online]. Available: https://laravel-excel.com
[22] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," Ph.D. dissertation, University of California, Irvine, 2000.
[23] G. Van Rossum dan F. L. Drake, *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009.
[24] S. Ramírez. *FastAPI Documentation*, versi 0.100+. (2023). [Online]. Available: https://fastapi.tiangolo.com
[25] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.