

Implementasi Sistem Informasi Layanan Pelanggan Berbasis Web pada Tuku Ramen Menggunakan Model *Waterfall*

Surya Darmawansyah¹, Muhammad Abiet Fatwa Kusumadita^{1*}

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: suryadarmawansyah77@gmail.com, [*abitm107@gmail.com](mailto:abitm107@gmail.com)

(* : coresponding author)

Abstrak—Informasi menu, cabang, jam operasional, paket, pertanyaan umum, dan reservasi Tuku Ramen sebelumnya tersebar pada Instagram, Linktree, WhatsApp, dan komunikasi langsung dengan admin. Kondisi tersebut mengharuskan pelanggan membuka beberapa kanal dan menyebabkan admin menghabiskan sekitar 2–3 jam per hari untuk menjawab pertanyaan berulang. Penelitian ini bertujuan mengimplementasikan sistem informasi layanan pelanggan berbasis web sebagai pusat informasi sekaligus media pengelolaan data operasional. Data diperoleh melalui observasi proses layanan, wawancara dengan General Manager, studi pustaka, dan pemeriksaan repositori aplikasi. Pengembangan menggunakan model Waterfall yang dibatasi pada tahap analisis kebutuhan, perancangan, implementasi, dan pengujian. Sistem dimodelkan menggunakan UML dan basis data relasional, kemudian diimplementasikan dengan Next.js, TypeScript, Supabase PostgreSQL, Supabase Auth, dan Supabase Storage. Halaman publik menyediakan informasi menu per cabang, paket, lokasi, jam operasional, FAQ, serta tautan reservasi WhatsApp. Dasbor admin dan owner mendukung pengelolaan menu, cabang, paket, FAQ, gambar, status ketersediaan, dan hak akses pengguna. Pengujian black-box terhadap 15 skenario menunjukkan seluruh skenario berjalan sesuai hasil yang diharapkan atau mencapai keberhasilan fungsional 100%. Hasil tersebut membuktikan kesesuaian fungsi utama, tetapi belum membuktikan peningkatan efisiensi dan kepuasan pelanggan karena evaluasi sebelum–sesudah serta uji kegunaan belum dilakukan.

Kata Kunci: Sistem Informasi, Layanan Pelanggan, Waterfall, Next.Js, Supabase, Pengujian Black-Box

Abstract—Information on Tuku Ramen menus, branches, operating hours, packages, frequently asked questions, and reservations was previously distributed across Instagram, Linktree, WhatsApp, and direct communication with administrators. This required customers to access several channels and caused administrators to spend approximately 2–3 hours per day answering repetitive questions. This study aimed to implement a web-based customer service information system as a centralized information and operational data management platform. Data were obtained through service-process observation, an interview with the General Manager, literature review, and application repository inspection. Development used the Waterfall model and was limited to requirements analysis, design, implementation, and testing. The system was modeled using UML and a relational database and implemented with Next.js, TypeScript, Supabase PostgreSQL, Supabase Auth, and Supabase Storage. The public pages provide branch-based menus, packages, locations, operating hours, FAQs, and WhatsApp reservation links. The administrator and owner dashboards support the management of menus, branches, packages, FAQs, images, availability status, and user access. Black-box testing of 15 scenarios showed that all scenarios produced the expected results, representing a 100% functional pass rate. These results demonstrate conformity with the main functional requirements; however, they do not yet demonstrate improvements in efficiency or customer satisfaction because pre–post evaluation and usability testing were not conducted.

Keywords: Information System, Customer Service, Waterfall, Next.Js, Supabase, Black-Box Testing

1. PENDAHULUAN

Perkembangan teknologi informasi mendorong pelaku usaha kuliner menyediakan informasi yang cepat, akurat, konsisten, dan mudah diakses. Situs web dapat menjadi kanal terpusat untuk menyajikan informasi sekaligus mendukung pengelolaan data operasional tanpa mengharuskan pelanggan memasang aplikasi tambahan. Pengembangan sistem informasi perlu diawali dengan pemilihan metode yang sesuai dengan karakteristik proyek. Kajian Haniva et al. (2023) menunjukkan bahwa Waterfall sesuai untuk proyek berskala tidak terlalu besar dengan kebutuhan yang relatif jelas, sedangkan Ritonga dan Nasution (2025) menerapkannya secara berurutan melalui analisis, perancangan, implementasi, dan pengujian pada sistem berbasis web.

Tuku Ramen merupakan usaha kuliner ramen halal dengan dua cabang utama, yaitu Ciputat dan Pondok Ranji. Sebelum sistem dikembangkan, informasi menu, paket, lokasi, jam operasional,

dan reservasi tersebar melalui Instagram, Linktree, WhatsApp, serta komunikasi langsung dengan admin. Berdasarkan wawancara dengan General Manager, admin menghabiskan sekitar 2–3 jam per hari untuk menjawab pertanyaan berulang. Perbedaan informasi antar-cabang, terutama jam operasional dan ketersediaan menu, juga berisiko menyebabkan pelanggan menerima informasi yang tidak sesuai dengan outlet tujuan.

Penelitian terdahulu menunjukkan bahwa aplikasi berbasis web dapat membantu sentralisasi informasi dan digitalisasi proses layanan. Basten dan Ardiansyah (2022) mengembangkan sistem informasi desa berbasis web untuk mempermudah pengelolaan dan penyebaran informasi. Riziq et al. (2026) membangun sistem helpdesk ticketing untuk mengelola laporan dan status layanan. Putera et al. (2026) menerapkan Waterfall pada aplikasi presensi berbasis web. Frasetya et al. (2025) mengimplementasikan sistem operasional berbasis web dengan pengelolaan data terpusat. Purnama dan Haryono (2025) memanfaatkan Next.js untuk membangun platform terpusat beserta dasbor administrasi, sedangkan Ritonga dan Nasution (2025) menunjukkan penerapan Waterfall dan pengujian black-box pada sistem web. Meskipun demikian, penelitian tersebut belum membahas kebutuhan usaha kuliner yang memerlukan informasi menu per cabang, status ketersediaan yang cepat diperbarui, reservasi melalui kanal operasional yang sudah digunakan, dan pemisahan kewenangan admin serta owner dalam satu sistem.

Tabel 1. Perbandingan Penelitian Terdahulu

Penelitian	Objek dan Metode	Hasil Utama	Kesenjangan terhadap Penelitian Ini
Basten & Ardiansyah (2022)	Sistem informasi desa berbasis web; Waterfall	Sentralisasi pengelolaan dan penyebaran informasi desa	Belum membahas menu per cabang, status ketersediaan, reservasi, dan kontrol akses operasional.
Purnama & Haryono (2025)	Platform komunitas berbasis web; Next.js	Platform terpusat dengan dasbor administrasi dan pengelolaan data	Objek komunitas alumni, bukan layanan informasi pelanggan usaha kuliner.
Ritonga & Nasution (2025)	Sistem absensi berbasis web; Waterfall	Tahapan Waterfall dan pengujian black-box mendukung fungsi sistem	Berfokus pada absensi dan tidak membahas informasi publik lintas cabang.
Riziq et al. (2026)	Helpdesk ticketing pelayanan masyarakat; Waterfall	Pengelolaan laporan dan pemantauan status layanan berbasis web	Berfokus pada tiket pengaduan, bukan pengelolaan menu dan informasi usaha kuliner.
Putera et al. (2026)	Presensi karyawan berbasis web; Waterfall	Pencatatan dan pelaporan presensi secara digital	Berfokus pada presensi, bukan layanan informasi publik dan peran admin-owner.
Frasetya et al. (2025)	Sistem pengajuan dan penjadwalan mobil operasional berbasis web	Implementasi pengelolaan data operasional secara terpusat	Berfokus pada mobil operasional, bukan layanan informasi pelanggan usaha kuliner.
Penelitian ini	Layanan pelanggan Tuku Ramen; Waterfall	Pusat informasi publik dan dasbor operasional dengan kontrol akses berbasis peran	Mengintegrasikan menu per cabang, paket, FAQ, status ketersediaan, gambar, dan reservasi WhatsApp.

Berdasarkan kesenjangan tersebut, penelitian ini bertujuan mengimplementasikan sistem informasi layanan pelanggan berbasis web yang mengintegrasikan halaman publik dengan dasbor pengelolaan operasional. Kontribusi penelitian mencakup: (1) pemetaan kebutuhan tiga kelompok pengguna, yaitu pengunjung, admin, dan owner; (2) implementasi struktur basis data untuk mengelola informasi lintas cabang; (3) penerapan autentikasi dan otorisasi berbasis peran; serta (4) evaluasi fungsi utama melalui 15 skenario pengujian black-box. Ruang lingkup sistem tidak mencakup pembayaran daring, integrasi langsung dengan kasir, maupun reservasi otomatis.

2. METODOLOGI PENELITIAN

2.1 Lokasi, Waktu, dan Pengumpulan Data

Penelitian dilaksanakan di Tuku Ramen pada 1 April–10 Juni 2026. Data dikumpulkan melalui empat teknik. Pertama, observasi dilakukan terhadap cara pelanggan memperoleh informasi dan cara admin memperbarui informasi operasional. Kedua, wawancara dengan General Manager digunakan untuk mengidentifikasi masalah, prioritas kebutuhan, pembagian kewenangan pengguna, dan batasan proses bisnis. Ketiga, studi pustaka digunakan untuk menentukan pendekatan pengembangan, pemodelan, basis data, teknologi implementasi, dan metode pengujian. Keempat, pemeriksaan repositori aplikasi digunakan untuk mencocokkan rancangan dengan struktur implementasi. Hasil pengumpulan data dikelompokkan menjadi kebutuhan fungsional, kebutuhan nonfungsional, aturan akses, dan batasan sistem.

2.2 Tahapan Pengembangan

Pengembangan menggunakan model Waterfall karena kebutuhan utama telah dapat diidentifikasi sejak awal dan setiap tahap memerlukan keluaran terdokumentasi sebagai dasar tahap berikutnya. Tahapan penelitian dibatasi pada analisis kebutuhan, perancangan, implementasi, dan pengujian. Tahap pemeliharaan tidak dievaluasi sebagai bagian hasil penelitian karena sistem belum dioperasikan dalam periode jangka panjang. Pemilihan ini sejalan dengan temuan Haniva et al. (2023) mengenai kesesuaian Waterfall untuk proyek dengan ruang lingkup relatif jelas dan dengan penerapan berurutan yang dilaporkan Ritonga dan Nasution (2025). Perbaikan terbatas tetap dilakukan apabila peninjauan pada suatu tahap menemukan ketidaksesuaian.

Tabel 2. Tahapan Pengembangan Sistem

Tahap	Aktivitas	Keluaran
Analisis kebutuhan	Mengidentifikasi aktor, informasi publik, proses pengelolaan data, hak akses, dan batasan sistem.	Daftar kebutuhan fungsional dan nonfungsional.
Perancangan	Menyusun use case, alur proses, struktur basis data, serta rancangan antarmuka.	Model UML, ERD, dan rancangan antarmuka.
Implementasi	Membangun halaman publik dan dasbor menggunakan Next.js, TypeScript, dan Supabase.	Aplikasi web yang dapat dijalankan dan diuji.
Pengujian	Menguji masukan, proses, keluaran, serta pembatasan akses melalui black-box testing.	Hasil 15 skenario pengujian dan status kelulusan.

2.3 Analisis dan Validasi Kebutuhan

Data hasil observasi dan wawancara dianalisis secara deskriptif dengan menelusuri masalah pada proses layanan yang berjalan. Setiap temuan dicatat berdasarkan sumber masalah, pihak yang terdampak, informasi yang dibutuhkan, serta tindakan yang harus tersedia pada sistem. Temuan mengenai informasi yang tersebar diterjemahkan menjadi kebutuhan halaman publik terpusat. Temuan mengenai perbedaan informasi antar-cabang diterjemahkan menjadi kebutuhan relasi menu dan cabang, filter cabang, serta pengelolaan jam operasional secara terpisah. Sementara itu, kebutuhan pembaruan informasi harian diterjemahkan menjadi fungsi dasbor untuk mengubah harga, deskripsi, status ketersediaan, gambar, alamat, dan pertanyaan umum.

Pemetaan kebutuhan menggunakan prinsip keterlacakan antara masalah, aktor, fungsi, dan data. Setiap fungsi diberi kelompok berdasarkan pengguna yang menjalankannya, yaitu pengunjung, admin, owner, atau kontrol akses. Pemetaan tersebut digunakan untuk memastikan bahwa fungsi yang diuji pada tahap akhir memiliki dasar kebutuhan yang jelas. Sebagai contoh, kebutuhan pelanggan untuk memperoleh menu sesuai outlet dihubungkan dengan fungsi filter cabang dan relasi menu_items terhadap branches. Kebutuhan pengelola untuk mencegah perubahan data oleh pengguna yang tidak berwenang dihubungkan dengan autentikasi Supabase, tabel admin_users, pemeriksaan role, dan Row Level Security.

Hasil analisis dikonfirmasi kembali melalui peninjauan rancangan dan demonstrasi aplikasi kepada pihak operasional. Masukan yang berkaitan dengan susunan informasi, kemudahan pembaruan data, pembagian kewenangan, dan alur reservasi digunakan untuk menyempurnakan rancangan sebelum pengujian akhir. Validasi ini tidak diposisikan sebagai uji kegunaan kuantitatif karena tidak menggunakan instrumen skala atau jumlah responden tertentu. Tujuannya adalah memastikan kesesuaian kebutuhan bisnis dan mencegah fungsi yang dibangun menyimpang dari proses layanan Tuku Ramen.

Kebutuhan nonfungsional ditetapkan sebagai batas kualitas implementasi. Sistem harus dapat diakses melalui peramban desktop dan perangkat bergerak, menyajikan antarmuka yang responsif, menjaga konsistensi informasi per cabang, memisahkan akses baca publik dan akses tulis terautentikasi, serta menyimpan gambar pada layanan penyimpanan berkas. Struktur komponen Next.js dan penggunaan tipe data TypeScript juga dipilih agar perubahan kode lebih mudah dipelihara. Kriteria tersebut menjadi acuan desain, tetapi belum seluruhnya diukur menggunakan pengujian performa, keamanan, atau kompatibilitas lintas-perangkat secara formal.

2.4 Teknologi dan Implementasi Sistem

Pemodelan sistem menggunakan Unified Modeling Language (UML) untuk menggambarkan aktor, fungsi, dan alur interaksi. Penggunaan UML dan rancangan antarmuka membantu menerjemahkan kebutuhan pengguna menjadi spesifikasi sistem yang terdokumentasi, sebagaimana diterapkan Agustin et al. (2025) pada perancangan sistem informasi berbasis web. Basis data disusun secara relasional untuk mengurangi duplikasi, menjaga keterhubungan antarentitas, dan memisahkan data berdasarkan tanggung jawab pengelolaannya.

Aplikasi dikembangkan menggunakan Next.js App Router dan TypeScript. Next.js digunakan untuk pengaturan rute, rendering, pengambilan data, dan penyusunan halaman aplikasi; penerapannya pada platform web terpusat juga dilaporkan oleh Purnama dan Haryono (2025). Dokumentasi Vercel (n.d.) digunakan sebagai acuan teknis Next.js. Supabase digunakan sebagai layanan backend yang menyediakan PostgreSQL, autentikasi, penyimpanan gambar, dan Row Level Security (Supabase, n.d.). Sistem dibatasi pada penyajian informasi dan pengelolaan data; pembayaran daring, integrasi langsung dengan kasir, serta reservasi otomatis belum termasuk dalam ruang lingkup.

2.5 Metode Pengujian

Pengujian menggunakan pendekatan black-box untuk memeriksa kesesuaian fungsi berdasarkan masukan dan keluaran tanpa menilai struktur kode internal. Pendekatan berbasis skenario dan dokumentasi hasil digunakan pada berbagai pengujian sistem web, antara lain oleh Rachmadi et al. (2025), Masdyah et al. (2025), dan Maemunah et al. (2025). Setiap skenario pada penelitian ini memuat aktor, kondisi atau tindakan pengujian, hasil yang diharapkan, hasil aktual, dan status. Skenario dinyatakan lulus apabila hasil aktual sesuai dengan hasil yang diharapkan. Lima belas skenario disusun untuk mewakili layanan publik, fungsi admin, fungsi owner, dan kontrol akses. Tingkat keberhasilan dihitung menggunakan Persamaan (1).

$$\text{Tingkat keberhasilan} = (\text{jumlah skenario lulus} / \text{jumlah skenario diuji}) \times 100\% \quad (1)$$

2.6 Kriteria Keberhasilan dan Batasan Evaluasi

Implementasi dinyatakan memenuhi kebutuhan fungsional apabila seluruh skenario prioritas menghasilkan keluaran yang sesuai dengan hasil yang diharapkan. Kriteria ini mencakup keberhasilan akses halaman publik, penyaringan data, autentikasi, pengelolaan informasi, pengelolaan data master, dan penolakan akses bagi akun yang tidak memiliki kewenangan.

Pengujian dilakukan pada fungsi yang telah tersedia dalam ruang lingkup penelitian dan tidak mencakup pembayaran daring, sinkronisasi dengan kasir, reservasi otomatis, atau rekap transaksi.

Evaluasi penelitian dibatasi pada validitas fungsi utama. Persentase kelulusan tidak digunakan sebagai bukti bahwa waktu kerja admin berkurang, pelanggan lebih puas, atau sistem aman terhadap seluruh jenis serangan. Pembuktian manfaat tersebut memerlukan rancangan evaluasi terpisah, misalnya pengukuran waktu sebelum dan sesudah implementasi, kuesioner System Usability Scale, pengujian beban, pengujian keamanan, serta pengamatan penggunaan dalam periode operasional yang lebih panjang.

3. ANALISA DAN PEMBAHASAN

3.1 Analisis Kebutuhan Sistem

Sistem memiliki tiga kelompok pengguna. Pengunjung dapat mengakses informasi tanpa autentikasi. Admin harus masuk melalui Supabase Auth dan hanya dapat menjalankan fungsi operasional. Owner memperoleh kewenangan tambahan untuk mengelola data master dan akun admin. Pemisahan hak akses diterapkan agar perubahan harian dapat dilakukan oleh admin tanpa memberikan kewenangan penuh terhadap penghapusan data dan pengelolaan pengguna.

Tabel 3. Kebutuhan Fungsional Sistem

Aktor	ID Fungsi	Kebutuhan Fungsional
Pengunjung	P-01 s.d. P-05	Melihat beranda, menu, paket, cabang, jam operasional, dan FAQ; memfilter menu berdasarkan cabang dan kategori; membuka tautan reservasi WhatsApp.
Admin	A-01 s.d. A-07	Login, melihat ringkasan dasbor, mengubah data menu dan status ketersediaan, mengunggah gambar, memperbarui informasi cabang, serta mengelola FAQ.
Owner	O-01 s.d. O-02	Menambah atau menghapus data master serta mengelola akun dan peran admin.
Kontrol akses	K-01	Menolak pengguna yang tidak terdaftar pada tabel <code>admin_users</code> atau tidak memiliki peran yang sesuai.

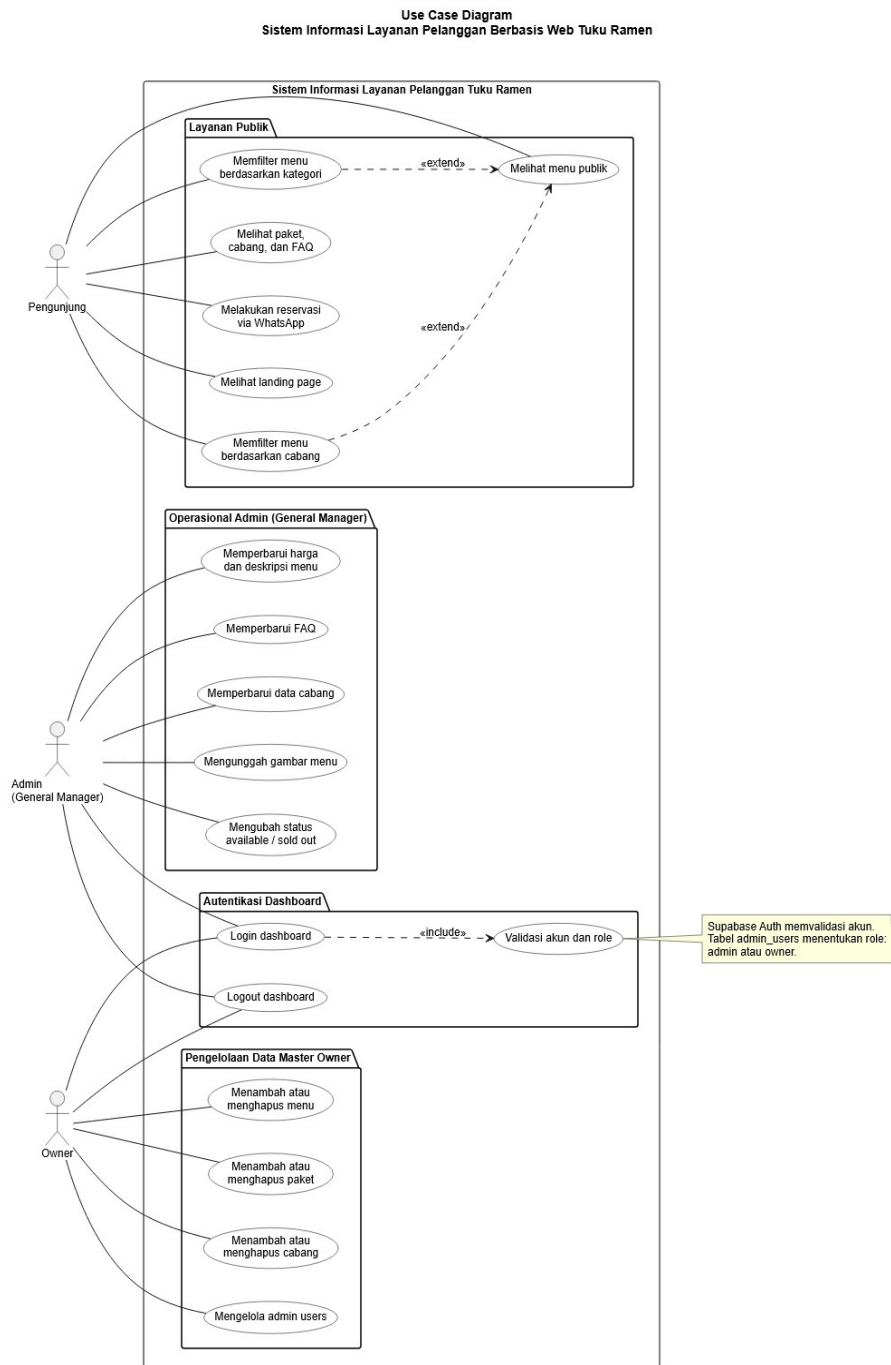
Kebutuhan nonfungsional meliputi akses melalui peramban desktop dan perangkat bergerak, antarmuka responsif, pemisahan hak baca dan tulis, penyimpanan gambar di luar tabel utama, serta konsistensi informasi per cabang. Sistem mempertahankan WhatsApp sebagai kanal reservasi agar tidak mengubah prosedur operasional yang sudah berjalan.

3.2 Interaksi Pengguna dan Hak Akses

Use case diagram pada Gambar 1 memetakan hubungan antara pengunjung, admin, dan owner. Pengunjung berinteraksi dengan layanan informasi publik dan reservasi. Admin mengelola data operasional setelah melalui autentikasi dan pemeriksaan peran. Owner memiliki seluruh fungsi admin serta kewenangan terhadap data master dan pengguna. Akun yang berhasil diautentikasi tetapi tidak tercatat pada tabel `admin_users` tetap ditolak untuk mencegah akses dasbor oleh pengguna yang tidak berwenang.

Tabel 4. Matriks Hak Akses Pengguna

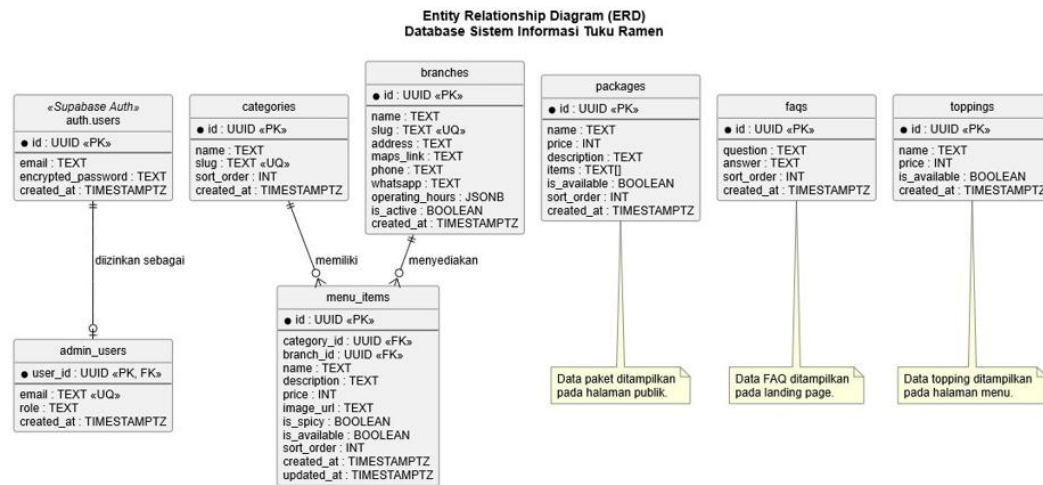
Fungsi	Pengunjung	Admin	Owner
Melihat informasi publik	Ya	Ya	Ya
Memperbarui data operasional	Tidak	Ya	Ya
Menambah/menghapus data master	Tidak	Tidak	Ya
Mengelola akun admin	Tidak	Tidak	Ya



Gambar 1. Use Case Diagram Sistem

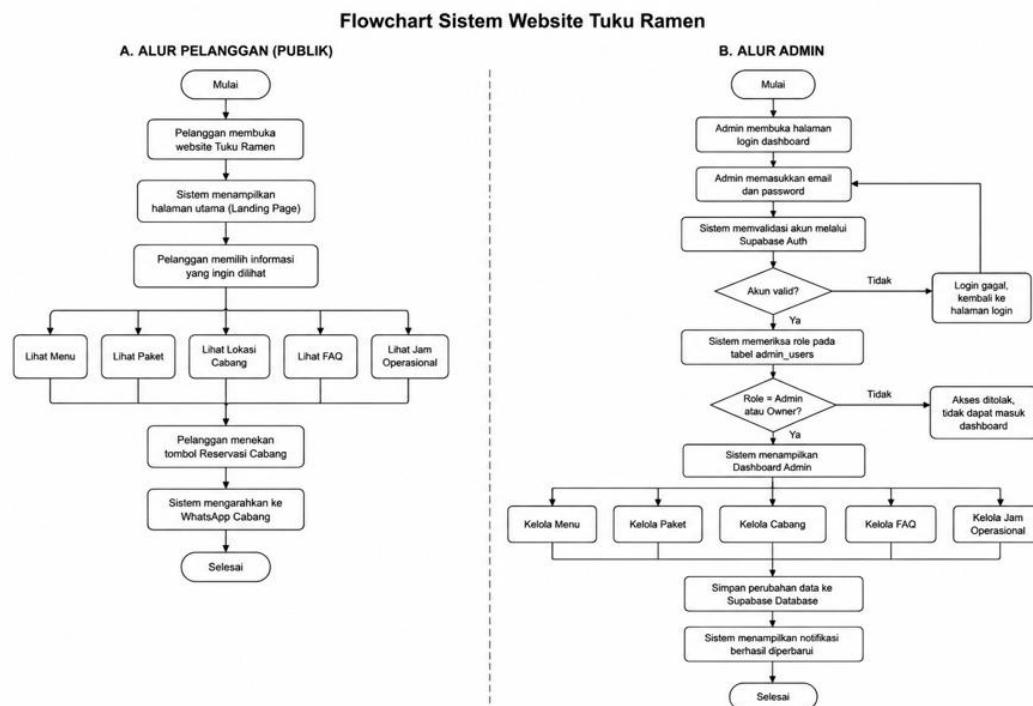
3.3 Struktur Basis Data dan Alur Sistem

Basis data aplikasi terdiri atas tujuh tabel utama, yaitu categories, branches, menu_items, packages, faqs, toppings, dan admin_users. Tabel menu_items berelasi dengan categories dan branches sehingga informasi menu dapat difilter berdasarkan kategori dan cabang tanpa menduplikasi data master. Tabel auth.users dikelola oleh Supabase Auth, sedangkan admin_users menyimpan penghubung user_id, email, dan role untuk menentukan kewenangan pengguna. Gambar menu disimpan pada Supabase Storage dan alamat berkasnya dicatat pada menu_items sehingga data biner tidak disimpan di tabel utama.



Gambar 2. Entity Relationship Diagram Sistem

Alur publik dimulai ketika pelanggan membuka situs, memilih cabang atau kategori, membaca informasi, dan menekan tombol reservasi untuk diarahkan ke WhatsApp cabang yang sesuai. Alur admin dimulai dari login, validasi akun dan peran, akses dasbor, perubahan data, penyimpanan ke basis data, dan pemberian notifikasi keberhasilan. Apabila akun tidak valid atau tidak memiliki izin, sistem menolak akses dan mengarahkan pengguna kembali ke halaman login.



Gambar 3. Flowchart Sistem

3.4 Arsitektur Sistem dan Pengamanan Data

Arsitektur aplikasi dibagi menjadi lapisan antarmuka, logika aplikasi, dan layanan data. Lapisan antarmuka terdiri atas halaman publik serta dasbor admin dan owner yang dibangun menggunakan komponen Next.js. Lapisan logika aplikasi menangani navigasi, validasi masukan, pemilihan cabang, penyaringan kategori, pengiriman perubahan data, dan pemeriksaan sesi pengguna. Lapisan layanan data menggunakan Supabase PostgreSQL untuk data terstruktur,

Supabase Auth untuk identitas pengguna, dan Supabase Storage untuk gambar. Pemisahan ini membuat halaman publik dan dasbor menggunakan sumber data yang sama tanpa mencampurkan berkas gambar berukuran besar ke dalam tabel utama.

Akses baca terhadap informasi yang memang ditujukan kepada pelanggan dipisahkan dari akses tulis. Pengguna yang membuka dasbor harus memiliki sesi autentikasi yang valid dan tercatat pada `admin_users`. Setelah itu, nilai role menentukan fungsi yang dapat dijalankan. Admin menangani pembaruan operasional, sedangkan owner memperoleh fungsi tambahan untuk menambah atau menghapus data master serta mengelola akun admin. Row Level Security digunakan sebagai pengamanan pada tingkat basis data agar pembatasan tidak hanya bergantung pada tampilan antarmuka. Dengan demikian, penyembunyian tombol pada halaman bukan satu-satunya mekanisme kontrol akses.

Konsistensi informasi cabang dijaga melalui foreign key antara `menu_items`, `categories`, dan `branches`. Data menu tidak perlu disalin menjadi tabel terpisah untuk setiap cabang karena hubungan tersebut memungkinkan sistem menampilkan data berdasarkan pilihan pelanggan. Alamat gambar yang tersimpan pada Storage dicatat sebagai URL pada `menu_items`. Apabila gambar diperbarui, sistem mengganti referensi berkas tanpa mengubah struktur data menu lainnya. Pemetaan data dan tanggung jawab pengelola diringkas pada Tabel 5.

Tabel 5. Pemetaan Data dan Tanggung Jawab Pengelolaan

Entitas	Data Utama	Pengelola	Fungsi dalam Sistem
<code>categories</code>	Nama dan urutan kategori	Owner	Mengelompokkan menu agar dapat difilter berdasarkan kategori.
<code>branches</code>	Nama, alamat, jam, peta, WhatsApp, status	Admin/Owner	Menyimpan informasi khusus setiap cabang.
<code>menu_items</code>	Menu, harga, deskripsi, gambar, status, relasi	Admin/Owner	Menjadi sumber daftar menu per kategori dan cabang.
<code>packages</code>	Nama paket, harga, deskripsi, ketersediaan	Admin/Owner	Menyajikan dan mengelola informasi paket.
<code>faqs</code>	Pertanyaan, jawaban, urutan tampil	Admin/Owner	Menyediakan jawaban untuk pertanyaan pelanggan yang berulang.
<code>toppings</code>	Nama, harga, dan status ketersediaan	Owner	Menyimpan data tambahan menu yang dikelola sebagai data master.
<code>admin_users</code>	<code>user_id</code> , email, dan role	Owner	Menghubungkan akun autentikasi dengan kewenangan pengguna.

Pemetaan tersebut mendukung satu sumber data operasional untuk seluruh kanal aplikasi. Perubahan pada data cabang, menu, atau FAQ disimpan pada entitas terkait dan dapat ditampilkan kembali pada halaman publik tanpa memasukkan informasi yang sama secara berulang. Namun, konsistensi data tetap bergantung pada ketepatan pengguna saat memperbarui informasi. Sistem belum menerapkan alur persetujuan berjenjang, riwayat perubahan yang rinci, atau mekanisme pemulihan versi untuk setiap operasi.

4. IMPLEMENTASI

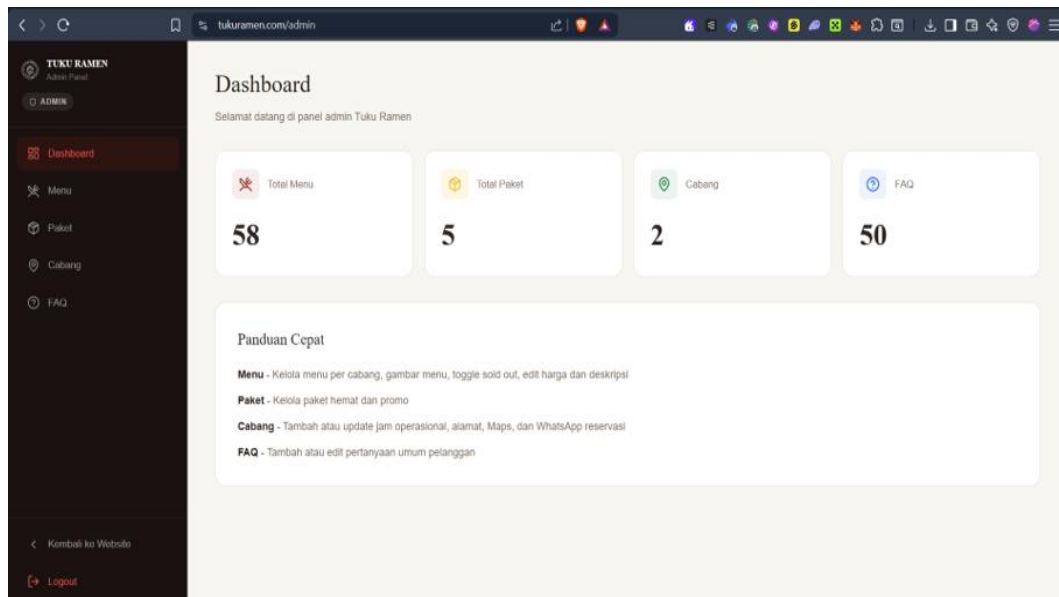
4.1 Implementasi Halaman Publik dan *Dashboard*

Halaman publik menyajikan identitas Tuku Ramen, menu, paket, lokasi cabang, jam operasional, FAQ, dan tombol reservasi. Pemilihan cabang dan kategori digunakan untuk menyaring menu agar pelanggan memperoleh informasi yang sesuai dengan outlet tujuan. Tombol reservasi membuka percakapan WhatsApp pada cabang yang dipilih sehingga proses komunikasi tetap mengikuti prosedur operasional yang berjalan.



Gambar 4. Implementasi Halaman Beranda

Dashboard menampilkan ringkasan data dan navigasi menuju pengelolaan menu, paket, cabang, dan FAQ. Pada data yang digunakan saat pengujian terdapat 58 menu, 5 paket, 2 cabang, dan 50 FAQ. Admin dapat mengubah harga dan deskripsi, menandai menu tersedia atau habis, mengunggah gambar, serta memperbarui alamat, jam operasional, peta, tautan WhatsApp, dan FAQ. Owner memperoleh fungsi tambahan untuk menambah atau menghapus data master dan mengelola pengguna admin.



Gambar 5. Implementasi *Dashboard Admin*

Implementasi telah memusatkan informasi yang sebelumnya tersebar ke dalam satu situs dan menyediakan pemisahan kewenangan antara pengguna operasional dan pemilik. Pemusatan data operasional berbasis web juga dilaporkan membantu mengurangi risiko kesalahan pencatatan dan memperbaiki keterlacakan informasi oleh Frasetya et al. (2025). Namun, reservasi pada sistem Tuku Ramen masih diteruskan ke WhatsApp sehingga pencatatan reservasi belum terotomasi dan belum dapat digunakan untuk menghasilkan rekap analitik secara langsung.

4.2 Implementasi Pengelolaan Data dan Kontrol Akses

Pengelolaan data dilakukan melalui formulir pada dasbor. Ketika admin mengubah harga, deskripsi, status ketersediaan, alamat, jam operasional, tautan peta, atau FAQ, aplikasi memvalidasi data yang diwajibkan sebelum mengirim perubahan ke Supabase. Setelah penyimpanan berhasil, informasi dapat dimuat ulang pada halaman publik dari sumber data yang sama. Mekanisme ini mengurangi kebutuhan untuk memperbarui informasi secara terpisah pada beberapa halaman dan membantu menjaga kesesuaian informasi antara dasbor dan tampilan pelanggan.

Pengelolaan gambar menggunakan alur yang berbeda dari data teks. Berkas dipilih melalui formulir, diunggah ke Supabase Storage, kemudian URL hasil unggahan disimpan bersama data menu. Pemisahan ini membuat tabel menu hanya menyimpan referensi berkas dan metadata yang dibutuhkan aplikasi. Apabila unggahan gagal, perubahan URL tidak diterapkan sehingga data menu sebelumnya tetap dapat digunakan. Ukuran, format, dan kualitas gambar tetap perlu dikendalikan oleh pengelola karena penelitian belum menguji kompresi otomatis maupun batas kapasitas penyimpanan dalam penggunaan jangka panjang.

Kontrol akses diterapkan pada saat pengguna memasuki dasbor dan ketika operasi data dijalankan. Akun yang telah login tetapi tidak tercatat pada `admin_users` tidak langsung memperoleh akses. Admin dapat memperbarui informasi operasional, tetapi tindakan tambah atau hapus data master dan pengelolaan akun dibatasi untuk owner. Pemisahan ini mengurangi risiko perubahan struktur data oleh pengguna operasional dan memastikan pengelolaan identitas dilakukan oleh peran dengan kewenangan lebih tinggi.

Implementasi notifikasi keberhasilan membantu pengguna mengetahui bahwa perubahan telah tersimpan. Sebaliknya, kegagalan autentikasi atau ketidaksesuaian peran menghasilkan penolakan akses dan pengalihan ke halaman login. Mekanisme tersebut memenuhi kebutuhan dasar pengamanan fungsi, tetapi belum menggantikan audit keamanan menyeluruh. Penelitian belum menguji serangan injeksi, pengambilalihan sesi, pembatasan percobaan login, maupun ketahanan terhadap akses serentak dalam jumlah besar.

4.3 Hasil Pengujian *Black-Box*

Pengujian dilakukan terhadap 15 skenario. Tabel 6 menunjukkan bahwa seluruh hasil aktual sesuai dengan hasil yang diharapkan. Data uji mencakup akses informasi publik, filter data, autentikasi, pengelolaan informasi, pengelolaan data master, dan penolakan akses.

Tabel 6. Hasil Pengujian *Black-Box*

ID/Aktor	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
P-01 Pengunjung	Membuka halaman beranda.	Identitas usaha, navigasi, dan informasi utama ditampilkan.	Beranda dan informasi utama tampil.	Lulus
P-02 Pengunjung	Membuka daftar menu.	Data menu ditampilkan dari basis data.	Daftar menu tampil sesuai data.	Lulus
P-03 Pengunjung	Memilih cabang.	Menu tersaring berdasarkan cabang.	Filter cabang bekerja sesuai pilihan.	Lulus
P-04 Pengunjung	Memilih kategori menu.	Menu tersaring berdasarkan kategori.	Filter kategori bekerja sesuai pilihan.	Lulus
P-05 Pengunjung	Menekan tombol reservasi.	Sistem membuka WhatsApp cabang yang dipilih.	Tautan WhatsApp terbuka dengan tujuan yang sesuai.	Lulus
A-01 Admin	Login dengan akun admin yang valid.	Pengguna diarahkan ke dasbor.	Login berhasil dan dasbor tampil.	Lulus
A-02 Admin	Membuka ringkasan dasbor.	Jumlah data menu, paket, cabang, dan FAQ ditampilkan.	Ringkasan data tampil sesuai basis data.	Lulus

ID/Aktor	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
A-03 Admin	Mengubah harga atau deskripsi menu.	Perubahan tersimpan dan tampil pada halaman publik.	Data menu berhasil diperbarui.	Lulus
A-04 Admin	Mengubah status tersedia/habis.	Status menu diperbarui.	Status ketersediaan berubah sesuai input.	Lulus
A-05 Admin	Mengunggah gambar menu.	Gambar tersimpan pada Storage dan tampil pada menu.	Gambar berhasil disimpan dan ditampilkan.	Lulus
A-06 Admin	Memperbarui alamat, jam, peta, atau WhatsApp cabang.	Informasi cabang tersimpan dan tampil pada halaman publik.	Informasi cabang berhasil diperbarui.	Lulus
A-07 Admin	Menambah atau mengubah FAQ.	Data FAQ tersimpan dan ditampilkan.	FAQ berhasil dikelola.	Lulus
O-01 Owner	Menambah atau menghapus data master.	Perubahan data master tersimpan sesuai kewenangan owner.	Operasi tambah/hapus berhasil.	Lulus
O-02 Owner	Mengelola akun dan peran admin.	Data pengguna dan peran tersimpan.	Akun admin berhasil dikelola.	Lulus
K-01 Akses	Akun tidak terdaftar mencoba membuka dasbor.	Sistem menolak akses atau mengarahkan ke login.	Akses ditolak sesuai aturan.	Lulus

Berdasarkan Persamaan (1), tingkat keberhasilan fungsional adalah $(15/15) \times 100\% = 100\%$. Hasil tersebut menunjukkan bahwa implementasi memenuhi kebutuhan fungsional yang diuji, termasuk pembatasan akses. Nilai 100% tidak boleh ditafsirkan sebagai bukti peningkatan efisiensi, kepuasan pelanggan, keamanan menyeluruh, atau kemampuan menangani beban tinggi. Penelitian belum melakukan pengukuran waktu sebelum–sesudah, System Usability Scale, pengujian beban, penetration testing, maupun evaluasi penggunaan dalam periode operasional yang panjang. Dengan demikian, kesimpulan pengujian dibatasi pada kesesuaian fungsi utama dengan skenario yang ditetapkan.

4.4 Pembahasan Hasil dan Keterbatasan

Hasil implementasi menunjukkan bahwa kebutuhan informasi publik dan pengelolaan operasional dapat disatukan dalam satu aplikasi. Temuan ini sejalan dengan Basten dan Ardiansyah (2022) yang menempatkan situs web sebagai media pengelolaan dan penyebaran informasi terpusat, serta Purnama dan Haryono (2025) yang menggunakan Next.js untuk membangun platform dengan dasbor administrasi. Perbedaan utama penelitian ini terletak pada pengelolaan menu berdasarkan cabang, status ketersediaan, pemisahan peran admin-owner, dan pemanfaatan WhatsApp sebagai penghubung menuju proses reservasi yang sudah digunakan oleh usaha.

Dari sisi operasional, satu sumber data memungkinkan perubahan harga, status menu, jam operasional, atau FAQ diterapkan melalui dasbor dan ditampilkan kembali kepada pelanggan. Pendekatan ini berpotensi mengurangi ketidakkonsistenan informasi, sebagaimana pemusatan data juga dilaporkan mendukung keterlacakan proses pada penelitian Frassetta et al. (2025). Namun, penelitian ini tidak mengukur besarnya pengurangan waktu kerja admin. Pernyataan manfaat operasional karena itu harus dipahami sebagai potensi yang didukung oleh rancangan fungsi, bukan sebagai hasil kausal yang telah diuji.

Penggunaan WhatsApp sebagai tujuan reservasi memberikan keuntungan berupa kesinambungan dengan prosedur yang sudah dikenal pelanggan dan pengelola. Di sisi lain, pilihan tersebut membuat data reservasi tidak masuk secara otomatis ke basis data aplikasi. Sistem belum

dapat menampilkan ketersediaan meja, mencegah pemesanan waktu yang sama, menghasilkan rekap reservasi, atau menghubungkan reservasi dengan transaksi. Pengembangan berikutnya perlu menentukan apakah otomatisasi reservasi akan menggantikan atau tetap berjalan berdampingan dengan WhatsApp.

Keterbatasan lain terdapat pada cakupan pengujian. Lima belas skenario membuktikan fungsi yang dipilih berjalan sesuai ekspektasi, tetapi belum mewakili seluruh variasi perangkat, jaringan, kesalahan input, dan kondisi beban. Evaluasi lanjutan dapat menggunakan pengujian kompatibilitas pada beberapa peramban, pengujian performa dengan jumlah akses bertahap, audit kebijakan Row Level Security, serta uji kegunaan yang melibatkan pelanggan, admin, dan owner. Pengukuran sebelum dan sesudah implementasi juga diperlukan untuk membandingkan waktu pencarian informasi, jumlah pertanyaan berulang, dan beban komunikasi admin.

5. KESIMPULAN

Sistem informasi layanan pelanggan Tuku Ramen berhasil diimplementasikan sebagai pusat informasi menu, cabang, jam operasional, paket, FAQ, dan reservasi WhatsApp. Sistem mengintegrasikan halaman publik dengan dasbor admin dan owner, tujuh tabel aplikasi, autentikasi Supabase, penyimpanan gambar, serta otorisasi berbasis peran. Pengujian black-box terhadap 15 skenario menghasilkan seluruh skenario lulus atau tingkat keberhasilan fungsional 100%. Hasil tersebut menunjukkan bahwa fungsi utama dan pembatasan akses berjalan sesuai kebutuhan yang diuji.

Penelitian belum membuktikan pengurangan waktu kerja admin, peningkatan kepuasan pelanggan, keamanan menyeluruh, maupun kemampuan sistem melayani beban tinggi. Pengembangan berikutnya disarankan mencakup reservasi terintegrasi, pencatatan analitik, riwayat perubahan, backup berkala, pengujian keamanan dan beban, serta evaluasi kegunaan dengan responden pelanggan dan pengelola. Pengukuran sebelum–sesudah perlu dilakukan agar manfaat sistem dapat dinilai secara kuantitatif.

REFERENCES

- Agustin, A. P., Doifa, A. W., Pramuditha, M., & Supriyatna, S. (2025). Analisis perancangan sistem informasi reimbursement transport berbasis web dengan metode Scrum di Kampus Al Azhar Serpong. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 881–889. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5807>
- Basten, I., & Ardiansyah, M. (2022). Perancangan sistem informasi desa berbasis web menggunakan model Waterfall (Studi kasus Desa Banjarsari Kabupaten Lebak). *Scientia Sacra: Jurnal Sains, Teknologi dan Masyarakat*, 2(1), 147–156.
- Frasetya, M. A., Prinandio, G. D., Ramadhan, K., & Ardiansyah, M. (2025). Implementasi sistem pengajuan dan penjadwalan mobil operasional berbasis web untuk SPPD pada PT Setya Kuliner Mandiri. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 954–964. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5889>
- Haniva, D. T., Ramadhan, J. A., & Suharso, A. (2023). Systematic literature review penggunaan metodologi pengembangan sistem informasi Waterfall, Agile, dan Hybrid. *Journal of Information Engineering and Educational Technology*, 7(1), 36–42. <https://doi.org/10.26740/jieet.v7n1.p36-42>
- Maemunah, S., Roh, L. C., & Anwar, C. (2025). Pengujian fungsional sistem informasi inventori berbasis web menggunakan standar ISO/IEC 29119. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 939–944. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5869>
- Masdyah, D., Lianti, L., & Anwar, C. (2025). Pengujian fungsional website sistem informasi persediaan barang Toko Jaya Makmur menggunakan standar ISO/IEC 29119. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 927–938. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5866>
- Purnama, R., & Haryono, W. (2025). Pengembangan platform digital komunitas alumni Universitas Al-Azhar berbasis web menggunakan Next.js dan Prisma ORM. *OKTAL: Jurnal Ilmu*

- Komputer dan Sains*, 4(12), 840–845. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5801>
- Putera, A. I. P., Ilham, F., Khambali, A. R., & Amrulloh, D. A. (2026). Rancang bangun aplikasi presensi karyawan berbasis web untuk meningkatkan efisiensi di PT Nyonya Lin Jaya. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 5(06), 437–445. <https://journal.mediapublikasi.id/index.php/oktal/article/view/6178>
- Rachmadi, N. G., Febrian, A., & Anwar, C. (2025). Pengujian perangkat lunak berstandar ISO/IEC 29119 pada website literasi digital SukaBaca. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 846–860. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5852>
- Ritonga, R. A. K. H., & Nasution, M. I. P. (2025). Perancangan sistem informasi absensi karyawan pada CV Azzara Gemilang menggunakan metode Waterfall. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 4(12), 913–921. <https://journal.mediapublikasi.id/index.php/oktal/article/view/5832>
- Riziq, A. B., Ilham, F., Firmansyah, F., & Prayogi, G. A. (2026). Rancang bangun sistem informasi helpdesk ticketing pelayanan masyarakat berbasis web. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 5(06), 501–509. <https://journal.mediapublikasi.id/index.php/oktal/article/view/6198>
- Supabase. (n.d.). (2026). *Supabase documentation*. Retrieved June 19, from <https://supabase.com/docs>
- Vercel. (n.d.). *Next.js documentation*. Retrieved June 19, 2026, from <https://nextjs.org/docs>