Contents lists available at JESA

Journal of Engineering and Science Application

journal homepage: https://jesa.aks.or.id/index.php/jesa/index

Sistem Terpadu Pemesanan Online, Kasir, dan Manajemen Inventori Berbasis Web pada D'Celup Chicken Crispy

Maulana Royyan Tsubaisa[a], Ilham Fauzi[b], Abelli Fiqri[c], Wasis Haryono[d]

[a-d]*Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia*

Article Info

Keywords:

Point Of Sale
Pemesanan Online
Manajemen Inventori
Model Prototype
UMKM Kuliner
Black Box Testing

ABSTRACT

Pencatatan transaksi, pendapatan, dan stok pada outlet kemitraan D'Celup Chicken Crispy masih dilakukan dengan tulis tangan oleh satu orang karyawan yang merangkap seluruh pekerjaan, sedangkan pesanan dari berbagai marketplace pesan-antar tersebar pada aplikasi masing-masing. Kondisi ini memicu kesalahan pencatatan hampir setiap hari, menyulitkan rekap pendapatan, dan membuka peluang transaksi yang tidak tercatat sehingga merugikan pemilik. Penelitian ini bertujuan membangun sistem terpadu berbasis web yang menyatukan pemesanan online, kasir (point of sale), kas harian, dan persediaan dalam satu basis data. Data dikumpulkan melalui wawancara, observasi, dan studi pustaka. Sistem dikembangkan dengan model Prototype melalui pengumpulan kebutuhan, perancangan cepat, pembuatan purwarupa, dan evaluasi bersama pemilik serta staf outlet, dimodelkan dengan UML dan entity relationship diagram, kemudian diimplementasikan menggunakan Next.js, PostgreSQL, Prisma, dan Better Auth. Hasilnya adalah aplikasi responsif yang dapat dioperasikan dari telepon genggam dengan hak akses berjenjang, pemesanan melalui website yang menyisihkan stok porsi secara otomatis, pencatatan penjualan langsung dan pesanan marketplace dengan nomor referensi, rekonsiliasi kas harian, pengelolaan stok porsi dan bahan baku, dashboard penjualan, serta jejak audit aktivitas pengguna. Pengujian black box terhadap 47 skenario pada 10 modul menghasilkan tingkat keberhasilan 100%. Sistem memusatkan data penjualan, kas, dan stok sehingga pemilik dapat menelusuri setiap transaksi dan perubahan data tanpa harus berada di outlet. Pengurangan stok bahan baku secara otomatis berdasarkan resep setiap menu dan integrasi langsung dengan marketplace menjadi pengembangan selanjutnya.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Maulana Royyan Tsubaisa
Program Studi Teknik Informatika, Fakultas Ilmu Komputer
Universitas Pamulang
Jl. Raya Puspiptek No. 46, Buaran, Serpong, Kota Tangerang Selatan, Banten 15310, Indonesia
Email: maulanaroyyan33@gmail.com

1. INTRODUCTION

Teknologi informasi kini menjadi bagian dari pengendalian biaya, pengambilan keputusan, dan kualitas layanan, karena sistem informasi mengubah data transaksi harian menjadi informasi yang dapat dipakai untuk mengawasi dan mengelola usaha [1]. Kebutuhan tersebut juga dirasakan Usaha Mikro, Kecil,

dan Menengah (UMKM) [2], termasuk usaha kuliner yang frekuensi transaksinya tinggi, nominalnya kecil, bahan bakunya mudah rusak, dan margin keuntungannya tipis, sehingga selisih kecil akibat transaksi yang tidak tercatat dapat menumpuk menjadi kerugian dalam hitungan hari.

D'Celup Chicken Crispy adalah outlet kemitraan olahan ayam yang mulai beroperasi pada tahun 2025 dan dijaga oleh satu orang karyawan yang merangkap memasak, mengemas, menerima pesanan, dan menjadi kasir. Pesanan datang dari empat *marketplace* pesan-antar (Grab, Gojek, Maxim, dan ShopeeFood), pembeli yang datang langsung, dan WhatsApp. Pesanan *marketplace* terekam pada aplikasi masing-masing, tetapi rekap penjualan, pendapatan, dan stok masih ditulis tangan. Menurut pemilik, lupa mencatat transaksi, salah menghitung, dan lupa mencatat pemakaian stok terjadi hampir setiap hari, dan tidak ada cara untuk mencocokkan penjualan, uang di laci, dan stok yang berkurang.

Sistem *point of sale* (POS) berbasis web terbukti mencatat transaksi secara otomatis, memperbarui stok, dan menghasilkan laporan yang akurat [3], serta meningkatkan akurasi pengelolaan keuangan UMKM [4]. Sistem inventori terkomputerisasi juga meningkatkan akurasi data stok dibandingkan pencatatan konvensional [5]. Ketika pemisahan tugas tidak mungkin dilakukan karena hanya ada satu karyawan, jejak transaksi yang memuat waktu dan pencatatnya menjadi pengendalian pengganti [6]. Penelitian terdahulu membangun aplikasi kasir dengan model *prototype* [7], [8] maupun *waterfall* [9]–[11], sebagian memadukan kasir dengan stok barang [12], sedangkan yang lain menangani pemesanan makanan [13] dan inventori sebagai aplikasi tersendiri.

Penelitian ini membangun sistem terpadu berbasis web untuk D'Celup Chicken Crispy dengan model Prototype, karena kebutuhan pengguna belum terdefinisi sepenuhnya di awal dan pemilik serta staf belum pernah memakai sistem kasir [8], [14]–[16]. Kontribusinya adalah (i) penyatuan pesanan website, penjualan langsung, dan pesanan *marketplace* dalam satu basis data, (ii) pengendalian stok porsi dan kas harian di sisi server beserta jejak auditnya, dan (iii) evaluasi melalui pengujian *black box* dan *user acceptance testing* (UAT).

2. RESEARCH METHOD

2.1. Pengumpulan Data dan Model Pengembangan

Penelitian dilaksanakan di outlet D'Celup Chicken Crispy, Semper Timur, Cilincing, Jakarta Utara, pada September–Desember 2026. Data dikumpulkan melalui wawancara dengan pemilik, observasi alur kerja dan perangkat di outlet yang hanya memiliki telepon genggam dan Wi-Fi, serta studi pustaka. Sistem dikembangkan dengan model Prototype yang bersifat iteratif: purwarupa dibuat, dievaluasi pengguna, lalu disempurnakan sampai disetujui [14], [17]. Berbeda dengan *rapid application development* dan *waterfall* [9], model ini menempatkan evaluasi pengguna atas tampilan sebagai pusat proses. Tahapannya adalah pengumpulan kebutuhan, perancangan cepat berupa *wireframe*, pembuatan *prototype*, evaluasi dan perbaikan berulang, pembangunan dan pengujian, serta implementasi di outlet.

2.2. Analisis Kebutuhan

Permasalahan, dampak, dan kebutuhan solusi dirangkum pada Tabel 1. Pemilik menilai pencatatan transaksi sebagai kebutuhan paling mendesak. Kebutuhan tersebut diturunkan menjadi fitur login dan hak akses, transaksi kasir, pencatatan asal pesanan, riwayat dan log transaksi, menu, stok, laporan, rekonsiliasi kas harian, dashboard, dan pemesanan daring. Kebutuhan non-fungsionalnya adalah tampilan ramah layar ponsel, akses berbasis peran dengan kata sandi terenkripsi, perhitungan otomatis, dan pencatatan aktivitas pengguna.

Tabel 1. Permasalahan, dampak, dan kebutuhan solusi

No	Permasalahan	Dampak	Kebutuhan solusi
1	Transaksi ada yang tidak tercatat	Data penjualan dan pendapatan tidak akurat	Pencatatan transaksi digital
2	Lupa mencatat pemakaian stok	Stok fisik dan catatan tidak sesuai	Manajemen stok
3	Uang penjualan berpotensi tidak disetor	Kerugian bagi pemilik	Rekonsiliasi kas dan log transaksi
4	Pendapatan dihitung manual	Rekap lama dan rawan salah	Laporan dan dashboard otomatis
5	Catatan tersebar di buku dan banyak aplikasi	Usaha sulit dipantau	Satu basis data terpusat

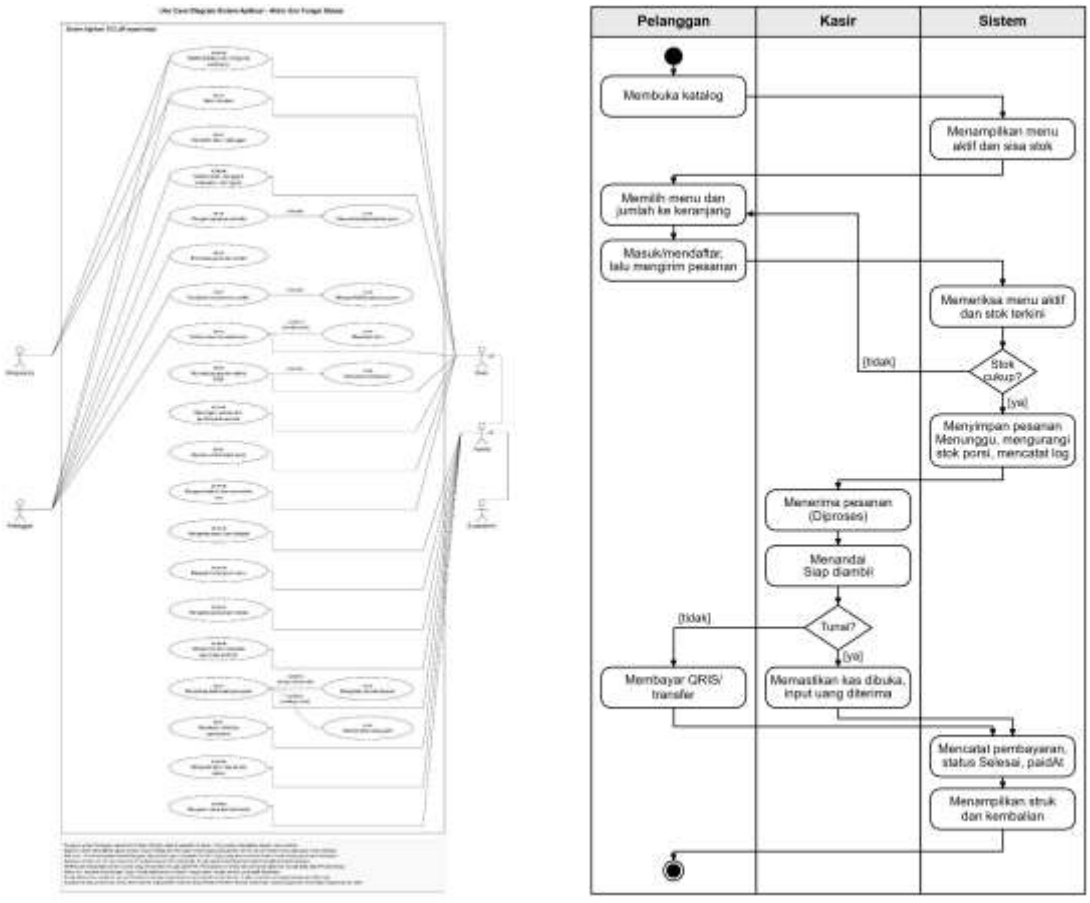
Fitur-fitur tersebut beserta aktor dan prioritasnya dirinci pada Tabel 2. Fitur riwayat dan log transaksi serta rekonsiliasi kas harian merupakan usulan peneliti untuk menjawab risiko kesalahan dan kecurangan pencatatan, dan pengembangan dilakukan bertahap mulai dari fitur berprioritas tinggi.

Tabel 2. Fitur sistem yang dirancang

No	Fitur	Deskripsi	Aktor	Prioritas
1	Login dan hak akses	Autentikasi pengguna dengan pembagian hak akses	Pemilik, staf	Tinggi
2	Transaksi penjualan (kasir)	Input pesanan, total otomatis, metode pembayaran, penyimpanan transaksi	Staf	Tinggi
3	Pencatatan asal pesanan	Penandaan sumber pesanan agar rekap per saluran dapat dilihat	Staf	Tinggi
4	Riwayat dan log transaksi	Setiap transaksi memiliki waktu dan pencatat; perubahan dapat ditinjau	Pemilik	Tinggi
5	Manajemen produk/menu	Pengelolaan daftar menu, harga, dan status ketersediaan	Pemilik	Sedang
6	Manajemen stok	Stok masuk dan keluar, pengurangan otomatis dari penjualan, penanda stok menipis	Pemilik, staf	Tinggi
7	Laporan penjualan	Rekap penjualan per periode, per saluran, dan per produk	Pemilik	Tinggi
8	Rekonsiliasi kas harian	Pencocokan total transaksi dengan uang kas pada akhir hari	Pemilik, staf	Sedang
9	Dashboard pemantauan	Ringkasan omzet, jumlah transaksi, produk terlaris, dan kondisi stok	Pemilik	Sedang
10	Pemesanan daring	Halaman pemesanan bagi pelanggan yang terhubung dengan sistem	Pemilik, staf	Rendah

2.3. Perancangan Sistem

Sistem dimodelkan dengan *Unified Modeling Language* (UML) [18]. Gambar 1(a) memperlihatkan *use case diagram* dengan lima aktor: pengunjung, pelanggan, kasir, pemilik, dan superadmin; pemilik mewarisi fungsi kasir dan superadmin mewarisi fungsi pemilik. Gambar 1(b) adalah *activity diagram* alur utama sistem, dari pelanggan membuka katalog, sistem memeriksa stok dan menyimpan pesanan, kasir memproses pesanan, sampai pembayaran tunai atau QRIS dicatat dan struk ditampilkan.

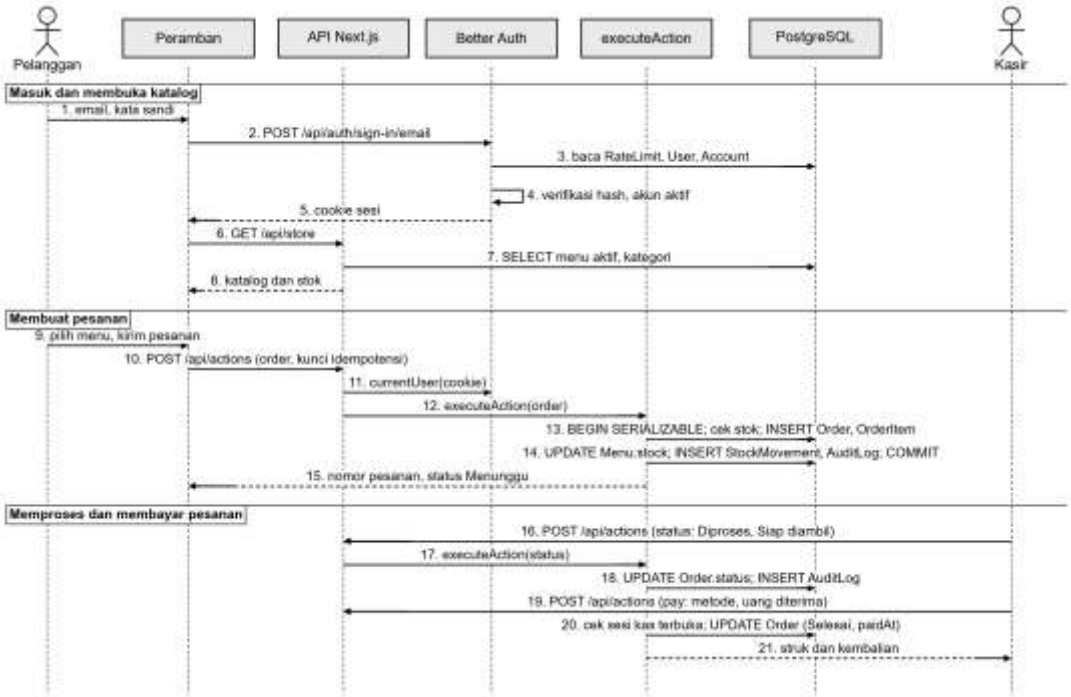


(a) (b)

Gambar 1. (a) *Use case diagram* aktor dan fungsi utama; (b) *activity diagram* pemesanan hingga pembayaran

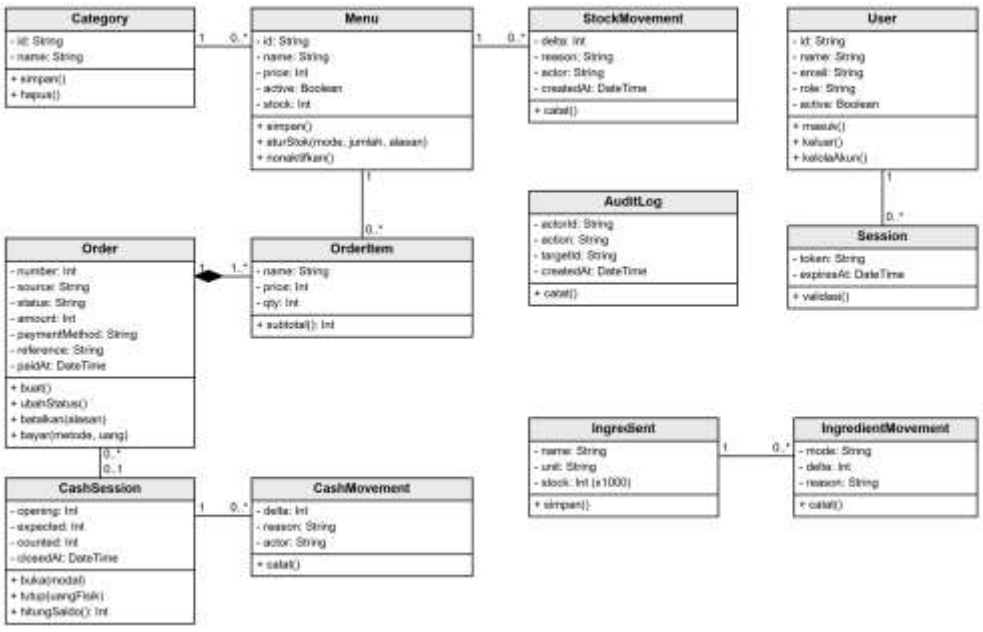
Urutan pesan antarkomponen pada alur yang sama ditunjukkan oleh *sequence diagram* pada Gambar 2. Peramban mengirim permintaan ke API Next.js, Better Auth memeriksa sesi, lalu executeAction menjalankan validasi dan perubahan data dalam satu transaksi basis data. Pada pembuatan pesanan,

pengecekan stok, penyimpanan Order dan OrderItem, pengurangan stok, serta pencatatan mutasi dan log audit terjadi dalam satu transaksi, sehingga gagalnya satu langkah membatalkan seluruh langkah.



Gambar 2. *Sequence diagram* masuk, pemesanan website, dan pembayaran

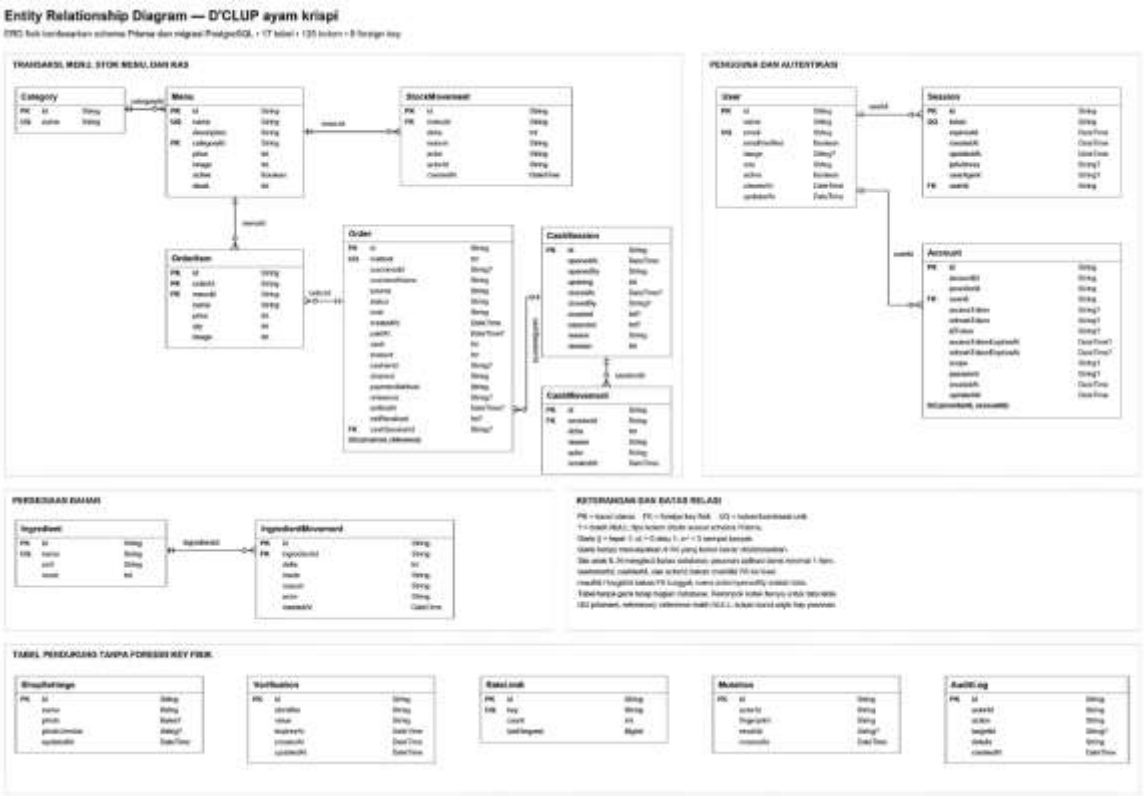
Aplikasi ditulis dengan TypeScript dan model data Prisma, sehingga *class diagram* pada Gambar 3 menggambarkan model domain: atribut berasal dari skema basis data, sedangkan operasi mewakili aksi server yang mengubah kelas tersebut, misalnya Order.bayar() dan CashSession.tutup(). Order memiliki OrderItem secara komposisi dan berelasi opsional dengan CashSession, karena hanya pembayaran tunai yang terikat pada sesi kas. Kelas pendukung Account, Verification, RateLimit, Mutation, dan ShopSettings tidak digambarkan agar diagram tetap terbaca.



Gambar 3. *Class diagram* model domain sistem

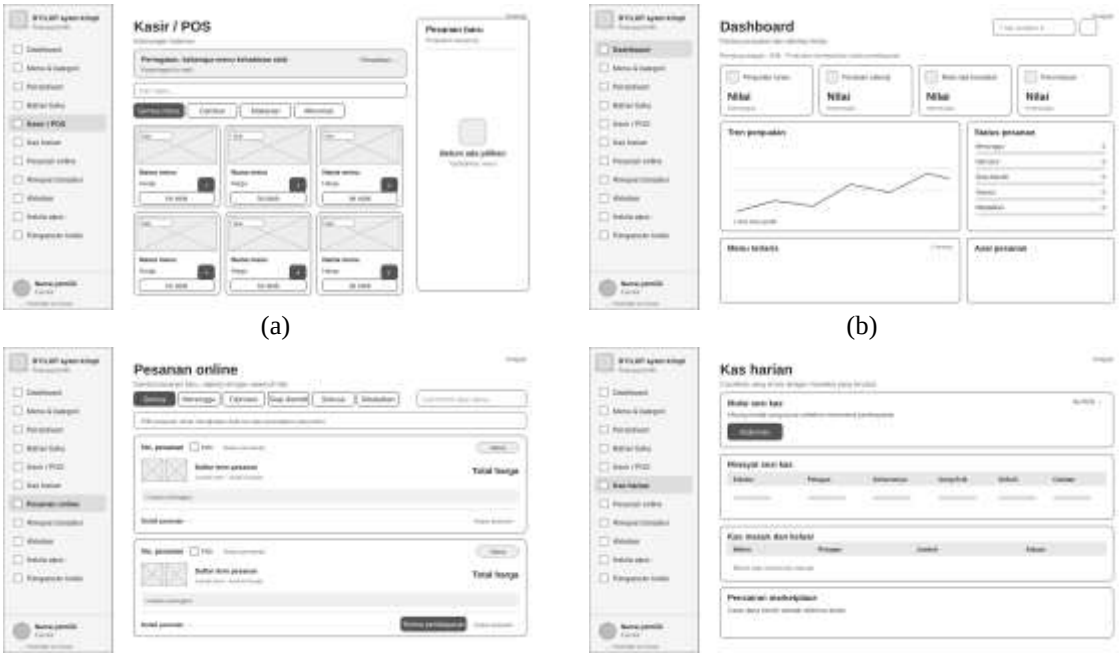
Basis data dirancang dengan *entity relationship diagram* (ERD) [19], [20] pada Gambar 4. Entitas Order menjadi pusat rancangan karena terhubung dengan rincian menu dan sesi kas. Basis data terdiri atas 18

tabel, 133 kolom, dan 9 *foreign key*; delapan *foreign key* memakai ON DELETE RESTRICT sehingga data induk yang masih dirujuk tidak dapat dihapus, dan nama serta harga menu disalin ke rincian pesanan agar riwayat transaksi tetap utuh.



Gambar 4. *Entity relationship diagram* sistem

Perancangan antarmuka dibuat sebagai *wireframe* 13 halaman pada tahap perancangan cepat. Semua halaman memakai kerangka yang sama: menu navigasi di kiri sesuai hak akses, judul halaman di atas, dan isi berupa kartu atau tabel. Gambar 5 memperlihatkan empat di antaranya, yaitu halaman Kasir/POS dengan daftar menu dan panel pesanan baru, dashboard dengan kartu ringkasan, grafik tren, dan status pesanan, halaman Pesanan online yang dikelompokkan menurut status, serta halaman Kas harian yang memuat pembukaan sesi, riwayat sesi, mutasi kas, dan pencairan *marketplace*.



(a) (b)

(c) (d)

Gambar 5. Rancangan antarmuka (*wireframe*): (a) Kasir/POS; (b) Dashboard; (c) Pesanan online; (d) Kas harian

2.4. Pengujian

Pengujian fungsional memakai *black box testing*, yaitu pemeriksaan keluaran terhadap spesifikasi tanpa melihat kode [21], seperti pada sistem pemesanan sejenis [13]. Sebanyak 47 skenario pada 10 modul disusun dengan *equivalence partitioning*, *boundary value analysis*, *state transition*, dan *decision table*; keberhasilannya dihitung dengan (1), dengan V jumlah skenario valid dan N jumlah seluruh skenario. Penerimaan pengguna diukur dengan UAT [14] berupa kuesioner sepuluh pernyataan berskala Likert 1–5 (sangat tidak setuju sampai sangat setuju) yang diisi oleh pemilik, karyawan, dan pelanggan; pelanggan hanya menilai pernyataan yang berkaitan dengan perannya. Persentase tiap pernyataan dihitung dengan (2), dengan ΣS jumlah skor dan R jumlah responden yang menilai pernyataan tersebut; nilai 81–100% berarti sangat baik, 61–80% baik, 41–60% cukup, dan di bawah 41% kurang.

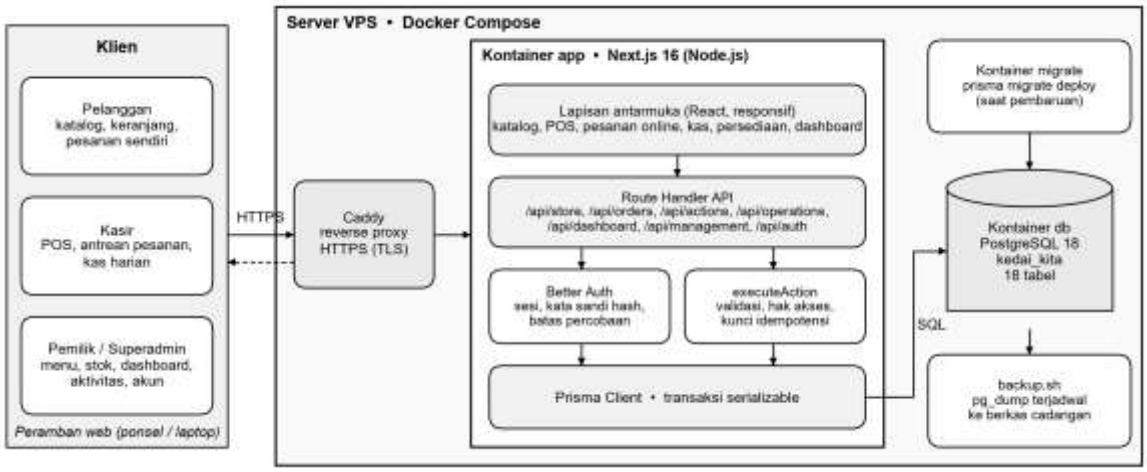
$$P = (V / N) \times 100\% \quad (1)$$

$$U = \Sigma S / (5 \times R) \times 100\% \quad (2)$$

3. RESULTS AND DISCUSSION

3.1. Arsitektur Sistem

Gambar 6 memperlihatkan arsitektur sistem. Pengguna mengakses aplikasi dari peramban ponsel atau laptop melalui HTTPS. Di server VPS, Docker Compose menjalankan Caddy sebagai *reverse proxy*, kontainer aplikasi Next.js 16 [22], dan kontainer PostgreSQL 18 [23]. Di dalam aplikasi, lapisan antarmuka React memanggil Route Handler API; Better Auth menangani sesi dan batas percobaan login, executeAction memeriksa hak akses dan kunci idempotensi, dan Prisma [24] menjalankan kueri dalam transaksi berisolasi *serializable* [25]. Kontainer migrate menerapkan perubahan skema saat pembaruan, dan skrip cadangan menyalin basis data secara berkala.



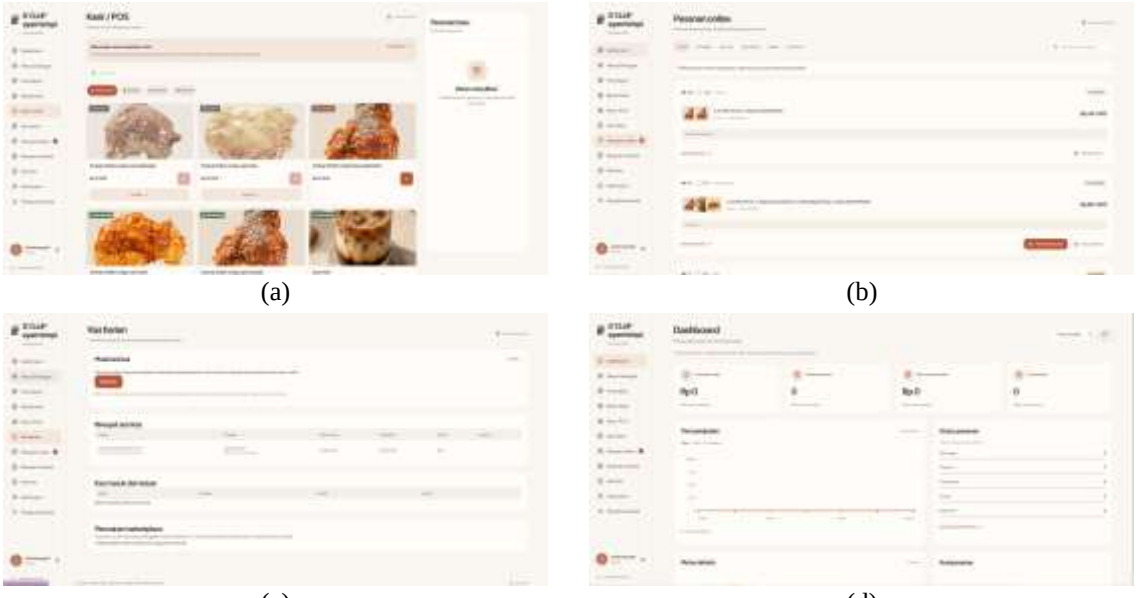
Gambar 6. Arsitektur sistem

3.2. Implementasi Antarmuka

Gambar 7 menampilkan fitur utama yang telah diimplementasikan. Pada Kasir/POS (a), penjualan langsung tersimpan berstatus Selesai, menu yang habis tidak dapat ditambahkan, dan pesanan Grab, Gojek, Maxim, atau ShopeeFood wajib membawa nomor referensi yang belum pernah dipakai. Pada Pesanan online (b), pesanan pelanggan tersimpan berstatus Menunggu dengan stok porsi langsung disisihkan, lalu maju satu langkah ke Diproses, Siap diambil, dan Selesai; pembatalan hanya dapat dilakukan saat Menunggu dan stoknya dikembalikan. Total pesanan dihitung dengan (3), dengan h harga dan q jumlah porsi menu ke-i, sedangkan kembalian tunai dihitung dengan (4) dan hanya diterima jika uang U tidak kurang dari total T.

$$T = \Sigma\,(h_i \times q_i),\ \ i = 1, 2, \ldots, n \tag{3}$$

$$K = U - T,\ \ U \geq T \tag{4}$$



(a) (b)

(c) (d)

Gambar 7. Tampilan aplikasi: (a) Kasir/POS; (b) Pesanan online; (c) Kas harian; (d) Dashboard

Kas harian (c) menjawab kekhawatiran pemilik tentang uang yang tidak masuk ke laci. Saldo seharusnya S dihitung dengan (5) dari modal M, penjualan tunai P, kas masuk I, dan kas keluar O, lalu selisih D terhadap uang fisik F dihitung dengan (6) dan wajib diberi keterangan bila tidak nol. Pada uji coba, modal Rp100.000, penjualan tunai Rp47.000, dan kas keluar Rp20.000 menghasilkan saldo Rp127.000, dan uang fisik Rp122.000 tercatat sebagai selisih −Rp5.000. Dashboard (d) menjumlahkan transaksi lunas per periode beserta tren, menu terlaris, dan asal pesanan, sedangkan halaman Aktivitas menampilkan log audit sehingga pemilik dapat menelusuri perubahan data tanpa berada di outlet.

$$S = M + \Sigma P + \Sigma I - \Sigma O \tag{5}$$

$$D = F - S \quad (6)$$

Stok dikelola pada dua tingkat. Stok porsi menu berkurang otomatis saat pesanan dibuat dan kembali saat pesanan dibatalkan; penambahan atau koreksi oleh pemilik wajib disertai alasan dan tersimpan sebagai mutasi yang memuat jumlah, alasan, pelaku, dan waktu. Halaman Persediaan menandai kondisi stok sebagai Habis, Menipis, atau Aman. Modul bahan baku mencatat bahan masuk, terpakai, terbuang, dan hasil hitung fisik dengan ketelitian tiga desimal; jumlahnya disimpan sebagai bilangan bulat per seribu agar tidak terjadi galat pembulatan. Setiap tindakan yang mengubah data membawa kunci idempotensi, sehingga tombol yang tertekan dua kali atau koneksi yang terputus tidak menghasilkan penjualan ganda.

3.3. Hasil Pengujian Black Box

Pengujian dilaksanakan pada 10 November 2026 dengan Playwright 1.63 [26] pada Chromium 153, memakai basis data uji terpisah dan akun *dummy* untuk keempat peran. Skenarionya antara lain penolakan pesanan yang melebihi stok, pengembalian stok saat pembatalan, penolakan pembayaran tunai ketika kas belum dibuka, penolakan nomor referensi duplikat, dan penolakan akses pelanggan ke halaman pemilik. Tabel 3 memuat contoh skenario yang mewakili aturan bisnis utama, dan Tabel 4 menunjukkan seluruh 47 skenario valid, sehingga menurut (1) tingkat keberhasilannya 100%.

Tabel 3. Contoh skenario dan hasil pengujian *black box*

No	Modul	Skenario	Hasil aktual	Status
1	Login dan hak akses	Login dengan kata sandi salah	Login ditolak dengan pesan “Email atau kata sandi salah, atau akun tidak aktif.”	Valid
2	Login dan hak akses	Pelanggan membuka halaman khusus pemilik	Halaman menampilkan “Akses terbatas” dan API menolak dengan status 403	Valid
3	Pemesanan online	Jumlah pesanan melebihi stok	Tombol tambah nonaktif setelah 3 porsi; permintaan 4 porsi ditolak server (409)	Valid
4	Pemesanan online	Pelanggan membatalkan pesanan berstatus Menunggu	Status menjadi Dibatalkan dan stok kembali dari 19 menjadi 20	Valid
5	Proses pesanan	Pembayaran tunai saat kas harian belum dibuka	Ditolak dengan pesan “Buka kas terlebih dahulu melalui menu Kas harian.”	Valid
6	Pembayaran	Uang tunai Rp50.000 untuk total Rp30.000	Pembayaran berhasil, struk menampilkan kembalian Rp20.000	Valid
7	Kasir/POS	Nomor referensi Grab duplikat	Transaksi ditolak walaupun berbeda huruf besar dan kecil	Valid
8	Kas harian	Tutup kas dengan selisih tanpa keterangan	Kas tidak ditutup; muncul pesan alasan minimal 3 huruf	Valid
9	Menu dan persediaan	Tambah menu dengan harga Rp999	Menu tidak disimpan; muncul pesan harga minimal Rp1.000	Valid
10	Laporan dan riwayat	Dashboard menampilkan penjualan hari ini	Kartu penjualan menampilkan Rp80.000, sama dengan jumlah transaksi lunas	Valid

Tabel 4. Rekapitulasi hasil pengujian *black box*

No	Modul	Jumlah skenario	Valid	Tidak valid	Persentase
1	Login dan hak akses	9	9	0	100%
2	Pemesanan online	5	5	0	100%
3	Proses pesanan (kasir)	4	4	0	100%
4	Kas harian	5	5	0	100%
5	Pembayaran	2	2	0	100%
6	Kasir/POS	6	6	0	100%
7	Menu dan persediaan	6	6	0	100%
8	Bahan baku	2	2	0	100%
9	Akun dan pengaturan	5	5	0	100%
10	Laporan dan riwayat	3	3	0	100%
	Total	47	47	0	100%

3.4. Hasil User Acceptance Testing

UAT diikuti empat responden, yaitu pemilik, karyawan outlet, dan dua pelanggan, pada 20 November 2026 setelah masing-masing mencoba sistem sesuai perannya. Pernyataan 3 dan 6–9 hanya dinilai oleh pemilik dan karyawan (R = 2), sedangkan pernyataan lainnya dinilai oleh seluruh responden (R = 4). Hasilnya disajikan pada Tabel 5. Rata-rata penerimaan sebesar 92,5% termasuk kategori sangat baik. Skor tertinggi (100%) diperoleh pernyataan tentang kemudahan pemakaian dari telepon genggam dan pencatatan pesanan *marketplace* beserta nomor referensinya, sedangkan skor terendah (80%) diperoleh pernyataan tentang kas harian, yang menunjukkan bahwa pencocokan uang di laci masih perlu dibiasakan dalam operasional harian.

Tabel 5. Hasil *user acceptance testing*

No	Pernyataan	R	Jumlah skor	Persentase
1	Tampilan aplikasi mudah dipahami	4	19	95,0%
2	Aplikasi mudah dioperasikan dari telepon genggam	4	20	100,0%
3	Mencatat penjualan di POS lebih cepat daripada tulis tangan	2	9	90,0%
4	Pesanan melalui website mudah dibuat dan dipantau	4	19	95,0%
5	Stok porsi berkurang dan kembali sesuai pesanan	4	19	95,0%
6	Kas harian membantu mencocokkan uang di laci	2	8	80,0%
7	Pesanan marketplace mudah dicatat beserta nomor referensinya	2	10	100,0%
8	Dashboard membantu memantau penjualan	2	9	90,0%
9	Log aktivitas membantu menelusuri perubahan data	2	9	90,0%
10	Sistem sesuai dengan kebutuhan operasional outlet	4	18	90,0%
	Rata-rata			92,5%

3.5. Pembahasan

Hasil ini sejalan dengan penelitian yang menunjukkan bahwa kasir berbasis web mengurangi kesalahan pencatatan dan mempercepat rekap UMKM [3], [4], dan bahwa model Prototype sesuai bagi pengguna yang kebutuhannya baru dapat dirumuskan setelah melihat rancangan [7], [8], [15]. Perbedaannya, aplikasi kasir terdahulu umumnya berhenti pada transaksi, stok, dan laporan [9], [10], sedangkan sistem ini menyatukan pesanan website, penjualan langsung, pesanan *marketplace*, kas harian, dan persediaan, dengan rekonsiliasi kas dan log audit sebagai pengganti pemisahan tugas [6]. Keterbatasannya, stok bahan baku belum berkurang otomatis karena resep per menu belum tersedia, pesanan *marketplace* masih dimasukkan kasir karena integrasi API memerlukan persetujuan penyedia platform, dan dampak sistem terhadap selisih kas dan stok dalam operasional jangka panjang belum diukur.

4. CONCLUSION

Sistem terpadu berbasis web untuk D'Celup Chicken Crispy berhasil dibangun dengan model Prototype dan menyatukan pemesanan website, kasir/POS, pesanan *marketplace*, kas harian, persediaan, dashboard, dan jejak audit dalam satu basis data dengan hak akses berjenjang. Rancangannya didokumentasikan melalui *use case*, *activity*, *sequence*, dan *class diagram*, ERD, arsitektur, serta *wireframe*. Pengujian *black box* atas 47 skenario menghasilkan keberhasilan 100%, dan UAT menghasilkan tingkat penerimaan 92,5% dengan kategori sangat baik. Pengembangan selanjutnya adalah resep per menu agar bahan baku berkurang otomatis, integrasi API *marketplace*, laporan periodik yang dapat diunduh, dan evaluasi pada operasional yang lebih panjang.

REFERENCES

[1] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 16th ed. Harlow, U.K.: Pearson, 2020.

[2] Republik Indonesia, *Undang-Undang Nomor 20 Tahun 2008 tentang Usaha Mikro, Kecil, dan Menengah*. Jakarta, Indonesia, 2008.

[3] I. K. Kumaini, Mutamassikin, and A. Triadi, "Rancang bangun sistem informasi point of sales berbasis web pada Toko Sembako Putri," *J. Komput. Teknol. Inf. dan Sist. Inf. (JUKTISI)*, vol. 4, no. 2, pp. 1104–1112, 2025, doi: 10.62712/juktisi.v4i2.596.

[4] E. Orlando and M. Riastuti, "Penerapan sistem kasir berbasis web pada UMKM untuk meningkatkan akurasi dan efisiensi pengelolaan keuangan," *J. Ilm. SIKOMTEK*, vol. 15, no. 2, pp. 82–90, 2025. [Online]. Available: https://sikomtek.jakstik.ac.id/index.php/sikomtek/article/view/12

[5] H. Christanto and R. Somya, "Sistem informasi berbasis web menggunakan framework Laravel pada rancangan sistem inventory gudang," *Jutisi: J. Ilm. Tek. Inform. dan Sist. Inf.*, vol. 12, no. 3, pp. 1204–1214, 2023, doi: 10.35889/jutisi.v12i3.1588.

[6] M. B. Romney and P. J. Steinbart, *Accounting Information Systems*, 14th ed. Harlow, U.K.: Pearson, 2018.

[7] I. Isnaurokhmah, K. A. Anggraeni, L. Rochmah, and I. Albana, "Perancangan aplikasi kasir dengan metode prototype: Studi kasus UMKM Artasari," *Merkurius: J. Ris. Sist. Inf. dan Tek. Inform.*, vol. 2, no. 4, pp. 135–148, 2024, doi: 10.61132/merkurius.v2i4.142.

[8] S. A. Putra, M. A. Zikri, J. A. S. Murad, and W. Haryono, "Pengembangan sistem kasir berbasis web menggunakan model prototype pada usaha Laundry Indimo Azza untuk meningkatkan efisiensi operasional dan pelayanan," *Switch: J. Sains dan Teknol. Inf.*, vol. 3, no. 4, pp. 43–52, 2025, doi: 10.62951/switch.v3i4.532.

[9] Yulianti, N. J. Pangihutan, R. Bashari, S. Ariadi, and S. Mulyati, "Pengembangan aplikasi kasir pada Rumah Makan Bahari menggunakan model waterfall," *J. Teknol. Sist. Inf. dan Apl.*, vol. 5, no. 4, pp. 255–260, Oct. 2022, doi: 10.32493/jtsi.v5i4.21249.

[10] R. Rifai and F. A. Nugroho, "Perancangan sistem informasi kasir berbasis web (studi kasus: Toko Sari)," *OKTAL: J. Ilmu Komput. dan Sains*, vol. 3, no. 4, pp. 792–801, Apr. 2024.

[11] R. Maulana and Y. Cahyono, "Rancang bangun aplikasi penjualan toko berbasis web dengan metode waterfall (studi kasus: Toko Usaha Baru)," *LOGIC: J. Ilmu Komput. dan Pendidik.*, vol. 1, no. 6, pp. 1495–1506, 2023.

[12] A. Kurniawan, M. Zacky, A. Tohirin, and W. Haryono, "Rancang bangun sistem informasi stok barang dan penjualan berbasis web untuk efisiensi operasional Toko Percetakan Zacky Karya Abadi," *SABER: J. Tek. Inform., Sains dan Ilmu Komun.*, vol. 3, no. 3, pp. 165–181, 2025.

[13] A. P. Sulaiman, D. R. Setiawan, K. N. A. Aripin, and W. Haryono, "Sistem informasi layanan catering berbasis web pada CV. Serayu," *J. Manaj. Teknol. Inform.*, vol. 3, no. 2, 2025, doi: 10.70038/jentik.v3i2.165.

[14] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY, USA: McGraw-Hill Education, 2020.

[15] A. G. Subekti, A. D. Setiadi, R. W. Pradipta, and W. Haryono, "Pengembangan sistem administrasi berbasis web pada Optik Inty menggunakan model prototype," *BINER: J. Ilmu Komput. Tek. dan Multimed.*, vol. 3, no. 2, pp. 125–134, Jun. 2025.

[16] S. Awaliyah, Nuraina, S. A. Purnamawati, and W. Haryono, "Pengembangan aplikasi berbasis web untuk penerimaan donasi secara online di Yayasan Pundi Amal Mulia," *Appl. Inf. Technol. Comput. Sci. (AICOMS)*, vol. 4, no. 1, pp. 43–50, Jun. 2025, doi: 10.58466/aicoms.v4i1.1859.
[17] I. Sommerville, *Software Engineering*, 10th ed. Boston, MA, USA: Pearson, 2016.
[18] J. Rumbaugh, I. Jacobson, and G. Booch, *The Unified Modeling Language Reference Manual*, 2nd ed. Boston, MA, USA: Addison-Wesley, 2004.
[19] P. P.-S. Chen, "The entity-relationship model—Toward a unified view of data," *ACM Trans. Database Syst.*, vol. 1, no. 1, pp. 9–36, Mar. 1976, doi: 10.1145/320434.320440.
[20] T. Connolly and C. Begg, *Database Systems: A Practical Approach to Design, Implementation, and Management*, 6th ed. Harlow, U.K.: Pearson, 2015.
[21] G. J. Myers, C. Sandler, and T. Badgett, *The Art of Software Testing*, 3rd ed. Hoboken, NJ, USA: Wiley, 2011.
[22] Vercel, "Next.js documentation." Accessed: Nov. 15, 2026. [Online]. Available: https://nextjs.org/docs
[23] The PostgreSQL Global Development Group, "PostgreSQL documentation: Constraints." Accessed: Nov. 15, 2026. [Online]. Available: https://www.postgresql.org/docs/current/ddl-constraints.html
[24] Prisma, "Prisma documentation." Accessed: Nov. 15, 2026. [Online]. Available: https://www.prisma.io/docs
[25] H. Berenson, P. Bernstein, J. Gray, J. Melton, E. O'Neil, and P. O'Neil, "A critique of ANSI SQL isolation levels," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 1995, pp. 1–10, doi: 10.1145/223784.223785.
[26] Microsoft, "Playwright documentation." Accessed: Nov. 15, 2026. [Online]. Available: https://playwright.dev/docs/intro