

Development of Web Service for Point of Sale Application of Kedungudi Sky Park Hill Cafe

[Pengembangan Web Service Aplikasi Point of Sale Café Kedungudi Sky Park Hill]

Ari Topan Iqbal Madagaskar¹⁾, Yulian Firdawati²⁾

¹⁾Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia.

²⁾Program Studi Teknik Informatika, Universitas Muhammadiyah Sidoarjo, Indonesia.

*Email Penulis Korespondensi : yulianfindawati@umsida.ac.id

Abstract. *The rapid development of information technology strengthens the need for efficient system integration in sales management at the Point of Sale (POINT OF SALE (POS)) in Sky Park Hill Cafe, Kedungudi. This study explores the application of API (Application Programming Interface) technology to enhance interoperability between the POINT OF SALE (POS) application and other systems. APIs enable more efficient and real-time data exchange between various system components, facilitating accurate transaction recording and timely sales reporting. With the implementation of APIs, it is expected that Sky Park Hill Cafe can improve operational efficiency, increase data transparency, and enhance customer satisfaction through a better shopping experience and more responsive services.*

Keywords - System Design, Information Technology, Point of Sale, Application Programming Interface

Abstrak. *Perkembangan teknologi informasi yang pesat memperkuat kebutuhan akan integrasi sistem yang efisien dalam manajemen penjualan di titik penjualan (Point of Sale) di Cafe Sky Park Hill, Kedungudi. Studi ini mengeksplorasi penerapan teknologi API (Application Programming Interface) untuk meningkatkan interoperabilitas antara aplikasi POINT OF SALE (POS) dengan sistem lainnya. API memungkinkan pertukaran data yang lebih efisien dan real-time antara berbagai komponen sistem, memfasilitasi pencatatan transaksi yang akurat dan laporan penjualan yang tepat waktu. Dengan implementasi API, diharapkan Cafe Sky Park Hill dapat meningkatkan efisiensi operasional, meningkatkan transparansi data, serta meningkatkan kepuasan pelanggan melalui pengalaman berbelanja yang lebih baik dan layanan yang lebih responsif.*

Kata Kunci - Perancangan Sistem, Teknologi Informasi, Point Of Sale, Application Programming Interface

I. PENDAHULUAN

Dalam era digital saat ini, bisnis harus beradaptasi dengan cepat terhadap perkembangan teknologi untuk tetap kompetitif [1]. Salah satu teknologi yang semakin penting adalah aplikasi Point of Sale (POS), yang membantu bisnis dalam mengelola transaksi penjualan secara efektif [2]. Pengembangan web service aplikasi Point of Sale (POS) menawarkan fleksibilitas dan efisiensi yang lebih tinggi dibandingkan dengan sistem Point of Sale (POS) tradisional [3]. Pemanfaatan alat seperti Postman menjadi esensial dalam proses ini, karena memudahkan pengujian dan debugging web service yang dikembangkan [4].

Postman merupakan alat yang sangat berguna dalam pengujian dan debugging *Application Programming Interface (API)* yang digunakan dalam pengembangan web service aplikasi Point of Sale (POS) [5]. Dengan Postman, tim pengembang dapat menguji berbagai skenario permintaan dan respons antara frontend dan backend dengan mudah [6]. Alat ini tidak hanya mempercepat proses pengembangan, tetapi juga meminimalkan risiko kesalahan yang mungkin terjadi selama implementasi [7]. Selain itu, kemampuan Postman dalam dokumentasi *Application Programming Interface (API)* memudahkan tim untuk memahami dan mengintegrasikan berbagai fitur aplikasi dengan lebih baik [8].

Pengembangan web service aplikasi Point of Sale (POS) cafe dengan metodologi Extreme Programming (XP) dapat meningkatkan kualitas dan kecepatan pengembangan [9]. Extreme Programming (XP) menekankan pada komunikasi yang efektif, pengujian berkelanjutan, dan umpan balik cepat. Dengan iterasi pendek, pengembang dapat menyesuaikan dan memperbaiki aplikasi sesuai dengan kebutuhan pengguna. Pendekatan ini tidak hanya meningkatkan kepuasan pengguna tetapi juga memastikan aplikasi yang dikembangkan memiliki ketahanan dan reliabilitas tinggi [10].

Survei literatur singkat menunjukkan bahwa penggunaan aplikasi Point of Sale (POS) berbasis web memberikan banyak keuntungan bagi bisnis kecil seperti cafe. Studi kasus dari berbagai literatur menunjukkan peningkatan efisiensi operasional dan penurunan biaya operasional. Selain itu, integrasi web service memungkinkan pemilik cafe untuk memantau bisnis mereka dari jarak jauh. Penelitian juga mengindikasikan bahwa penggunaan alat seperti Postman dalam pengembangan *Application Programming Interface (API)* dapat mempercepat proses pengembangan dan mengurangi risiko kesalahan [11]–[14].

Oleh karena itu, penelitian ini bertujuan untuk Pengembangan Web Service Aplikasi Point of Sale pada Café Kedungudi Sky Park Hill. Café Kedungudi Sky Park Hill merupakan salah satu destinasi kuliner yang berlokasi di

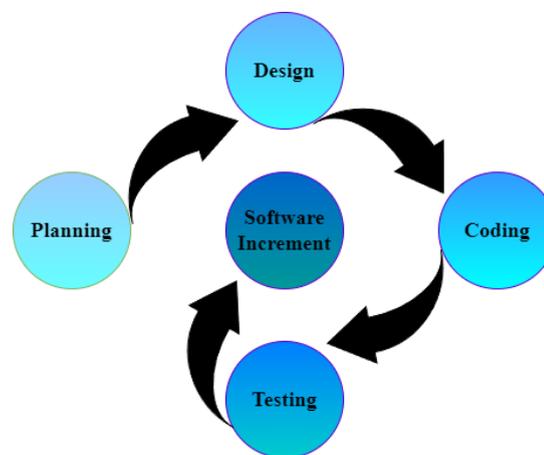
kota Mojokerto, yang selalu berusaha untuk memberikan pengalaman terbaik kepada pelanggan. Dalam upaya untuk meningkatkan efisiensi operasional dan kualitas layanan, pengembangan aplikasi Point of Sale (POS) berbasis web menjadi solusi yang tepat. Aplikasi ini tidak hanya membantu dalam memproses transaksi penjualan dengan cepat dan akurat, tetapi juga memungkinkan manajemen inventaris yang lebih efektif dan analisis data yang mendalam. Penggunaan alat seperti Postman dalam pengembangan web service menjadi krusial untuk memastikan sistem berjalan dengan lancar.

Dengan menerapkan metodologi Extreme Programming (XP) dalam pengembangan web service aplikasi POS untuk Café Kedungudi Sky Park Hill, diharapkan dapat memaksimalkan nilai tambah yang diberikan kepada pelanggan. XP menawarkan pendekatan yang adaptif dan kolaboratif, yang fokus pada pengujian berkelanjutan, umpan balik yang cepat, dan iterasi yang singkat. Dengan demikian, pengembang dapat merespons perubahan kebutuhan pasar dengan lebih fleksibel dan menghasilkan solusi aplikasi yang lebih sesuai dengan harapan pengguna.

II. METODE

Metode penelitian yang digunakan pada penelitian ini adalah Extreme Programming (XP), ditujukan untuk tim berukuran kecil dan menengah yang menghadapi perubahan persyaratan yang sangat cepat di rekayasa perangkat lunak. Extreme Programming (XP) menerapkan praktik-praktik seperti Pair Programming dan Test-Driven Development untuk meningkatkan kualitas perangkat lunak [15]. Pair Programming meningkatkan kolaborasi tim, sementara Test-Driven Development memastikan perubahan kode terus diuji dan memenuhi persyaratan fungsional. Extreme Programming (XP) juga fokus pada umpan balik cepat dan iterasi kecil, memungkinkan peneliti untuk dengan cepat menguji dan memperbaiki hipotesis mereka [16]. Keterlibatan aktif pelanggan membantu memahami preferensi pengguna, krusial dalam mengembangkan solusi yang relevan dan bermanfaat. Dengan demikian, Extreme Programming (XP) efektif dalam menggabungkan praktik-praktik yang meningkatkan kualitas, kecepatan, dan responsivitas dalam pengembangan perangkat lunak.

Kent Beck, pencetus Extreme Programming (XP), dihadapkan pada proyek Chrysler Compensation yang hampir gagal sebelum diintervensi oleh Ron Jeffries untuk menerapkan XP. Beck berhasil memperkenalkan perubahan yang meningkatkan fleksibilitas, efisiensi, dan adaptabilitas proyek sesuai dengan tujuan yang ditetapkan. XP berfokus pada nilai-nilai umpan balik, kesederhanaan, komunikasi, dan keberanian, serta dirancang untuk mengakomodasi perkembangan yang diminta oleh klien. Berikut adalah Alur dari metode Extreme Programming (XP) pada Gambar 2.1.



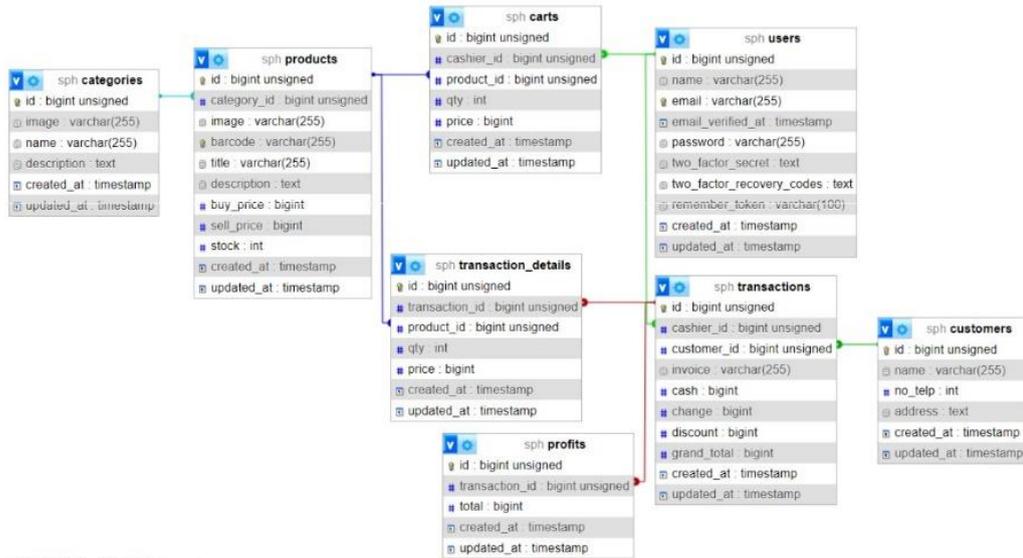
Gambar 2.1 Extreme Programming (XP)

1. Tahap Planning memperhatikan kebutuhan sistem secara cermat untuk memahami alur bisnis dan merinci fitur utama yang diinginkan, sebagai langkah awal dalam perancangan aplikasi.
2. Tahap Design meliputi pemodelan sistem berdasarkan analisis dari tahap planning dan pengelolaan basis data untuk menggambarkan hubungan antar data yang diimplementasikan ke MySQL.
3. Tahap Coding mengimplentasikan desain sistem ke dalam kode program menggunakan PHP.
4. Tahap Testing menggunakan Acceptance Testing untuk menganalisis dan menguji aplikasi, fokus pada fitur dan fungsi sistem secara menyeluruh dengan skenario end-to-end, termasuk load testing untuk memastikan kinerja sistem dalam situasi penggunaan yang diantisipasi.

III. HASIL DAN PEMBAHASAN

A. Perancangan Sistem

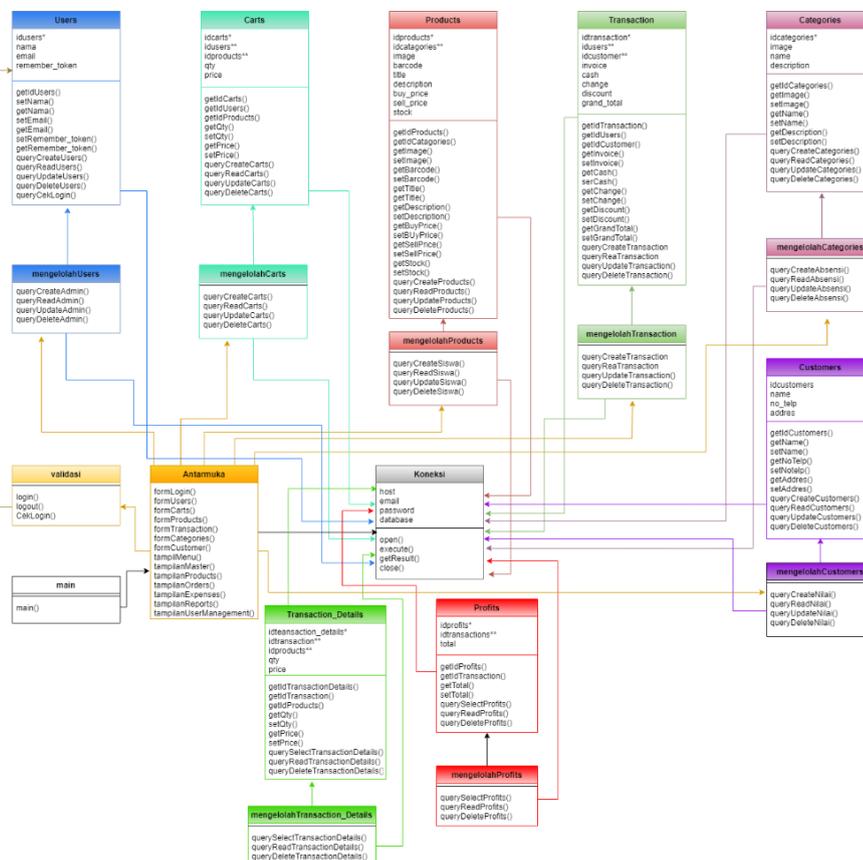
1. Database



Gambar 3.1 Database

Berdasarkan Gambar 3.1 merupakan diagram database yang didesain untuk mendukung berbagai alur kerja dalam Aplikasi Point Of Sale. Admin dapat mengelola informasi pengguna dan pelanggan, produk dan kategori, keranjang belanja, serta transaksi yang meliputi pembayaran dan pengiriman. Selain itu, sistem menyediakan fitur untuk menghasilkan laporan dan analisis berdasarkan data transaksi dan pelanggan, yang dapat membantu dalam mengidentifikasi pola dan tren untuk mendukung pengambilan keputusan bisnis. Database ini juga dapat diperluas dengan tabel tambahan untuk mendukung fungsionalitas yang lebih kompleks, dengan perhatian khusus terhadap keakuratan, konsistensi, dan keamanan data.

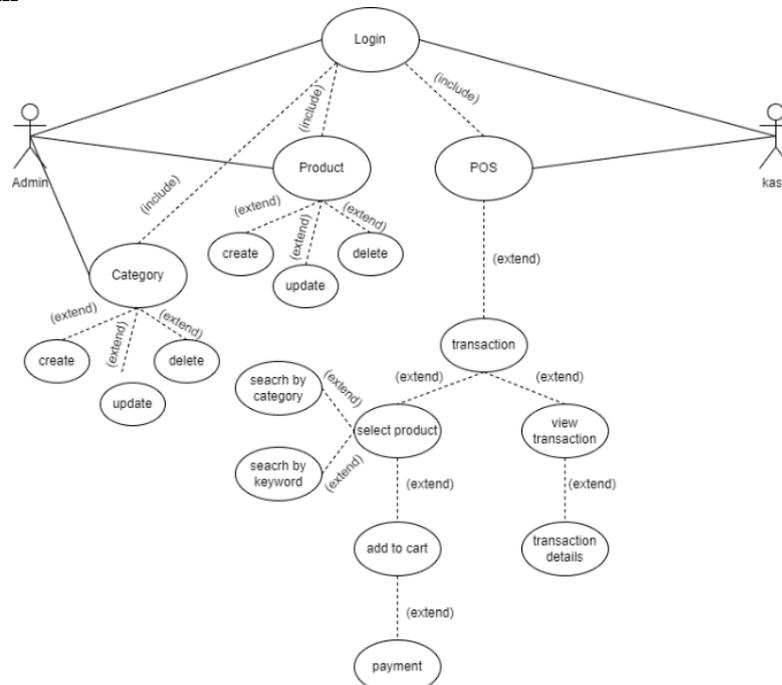
2. Class Diagram



Gambar 3.2 Class Diagram

Bedasarkan Gambar 3.2 merupakan class diagram yang terdapat kelas-kelas seperti Users, Carts, Products, Transaction, Categories, Customers, transaction_Details, Profits serta antarmuka seperti formLogin(), formUsers(), formCarts(), dan lain-lain. Setiap kelas memiliki atribut dan method yang berbeda-beda, dan terdapat hubungan antar kelas seperti association antara kelas Transaction dengan kelas Users dan Customers,

3. Use Case Diagram

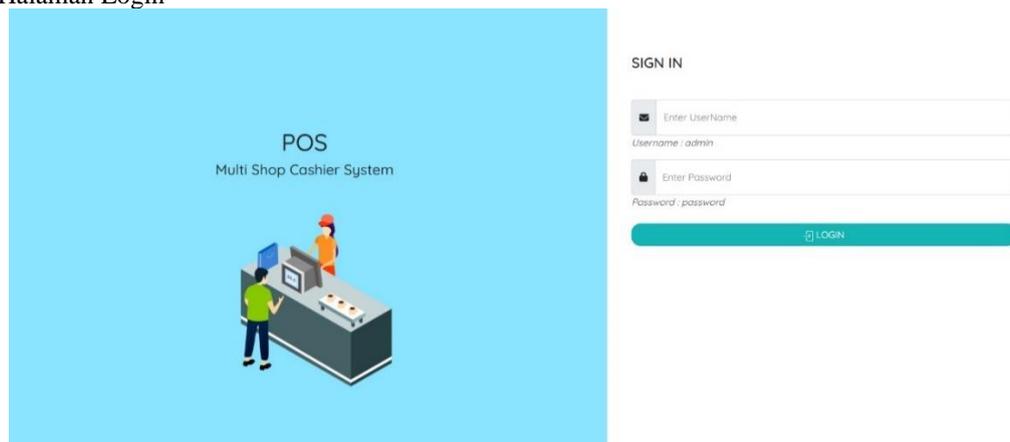


Gambar 3.3 Use Case Diagram

Berdasarkan Gambar 3.3 merupakan Diagram use case pada Aplikasi Point of Sale menggambarkan sistem pengelolaan produk dan transaksi, yang mungkin digunakan untuk sistem penjualan atau e-commerce. Admin memiliki akses penuh untuk mengelola produk dan kategori, sementara kasir memiliki akses terbatas untuk melakukan transaksi penjualan. Use case mencakup proses login untuk kedua aktor, pengelolaan produk dan kategori, serta transaksi penjualan yang melibatkan pemilihan produk, penambahan ke keranjang, dan pembayaran. Diagram ini memberikan gambaran umum tentang fungsionalitas sistem tanpa detail implementasi yang mendalam.

B. Design Interface

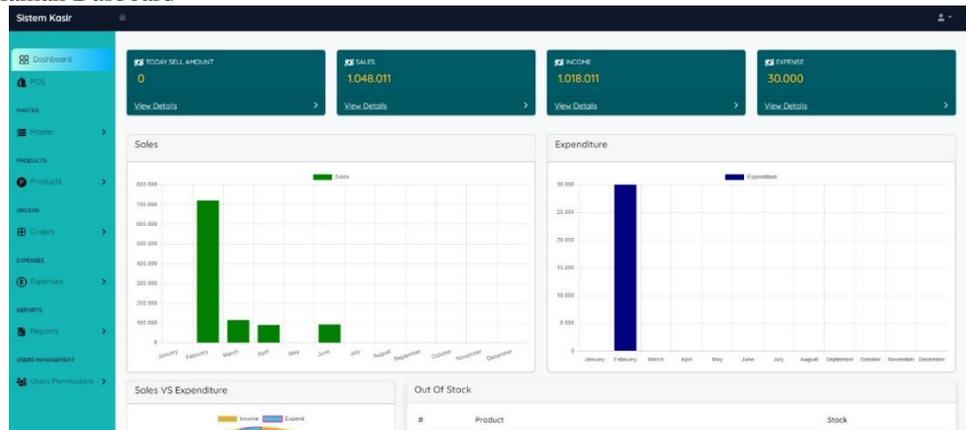
1. Halaman Login



Gambar 2.1 Halaman Login

Pada Gambar 2.1 pada halaman ini digunakan untuk Admin login supaya bisa mengakses website.

2. Halaman Dashboard



Gambar 2.2 Halaman Dashboard

Gambar 2.2 merupakan tampilan dari dashboard website yang menampilkan grafik penjualan.

3. Halaman Products

The products page displays a list of products with the following details:

No	Name	Image	Buy Price	Sell Price	Actions
1	Kopi Susu Gula Aren		30000	20000	Action +
2	Kopi Matcha		10000	15000	Action +
3	Kopi Susu Gula Aren		30000	20000	Action +
4	Kopi Matcha		10000	15000	Action +
5	Kopi Susu Gula Aren		30000	20000	Action +

Gambar 2.3 Halaman Product

Pada Gambar 2.3 merupakan Halaman Product yang dimana akan menampilkan Product yang akan dijual di cafe.

4. Halaman Carts

The cart page displays a grid of products with their prices and a summary panel on the right:

Product	Price
Kopi Susu Gula Aren	Rp. 20.000
Kopi Matcha	Rp. 15.000
Kopi Susu Gula Aren	Rp. 20.000
Kopi Matcha	Rp. 15.000
Kopi Susu Gula Aren	Rp. 20.000
Kopi Matcha	Rp. 15.000

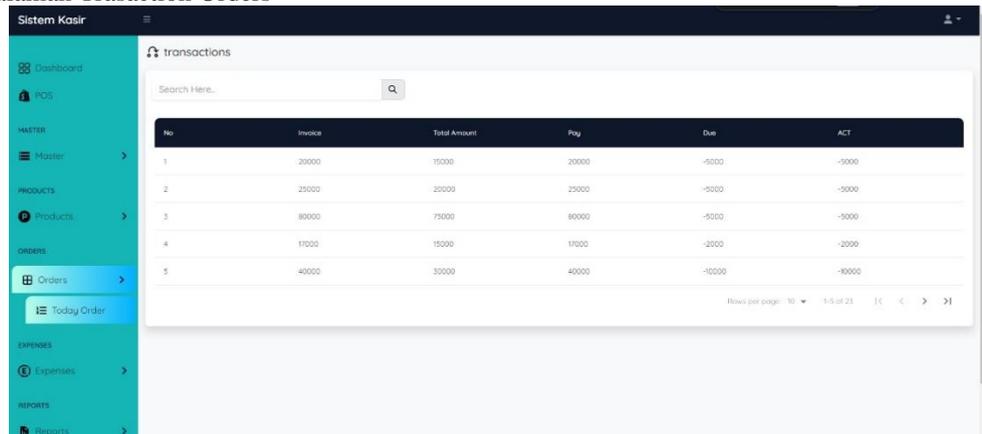
Summary Panel:

- Total Bayar:** NaN
- Sub Total:** Rp. 0
- Disc:** 0
- Ppn:** 0
- Customer:** Pilih Customers
- Bayar:** 0
- Kembalian:** 0

Gambar 2.4 Halaman Carts

Gambar 2.4 merupakan tampilan halaman Carts yang digunakan untuk mengimput pesanan dari customer.

5. Halaman Trasaction Orders

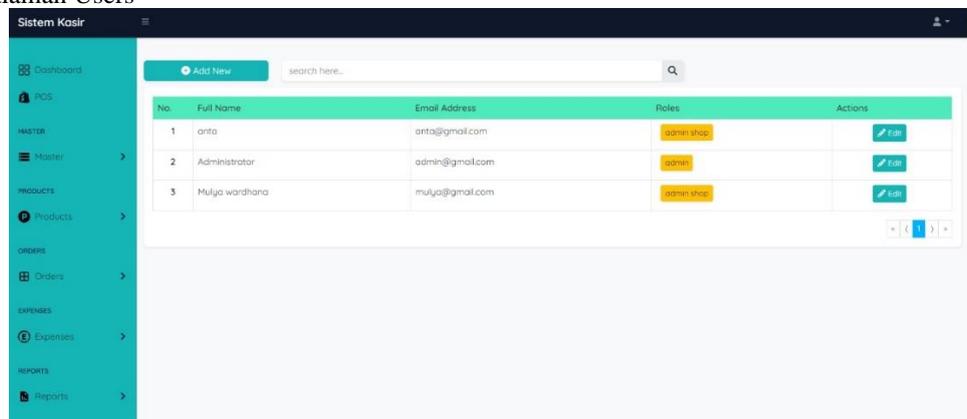


No	Invoice	Total Amount	Pay	Due	ACT
1	20000	15000	20000	-5000	-5000
2	25000	20000	25000	-5000	-5000
3	80000	75000	80000	-5000	-5000
4	17000	15000	17000	-2000	-2000
5	40000	30000	40000	-10000	-10000

Gambar 2.5 Halaman Trasaction Oreders

Gambar 2.5 merupakan tampilan dari halaman transaksi dari orderan dalam satu hari atau bisa juga disebut riwayat pemesanan dalam satu hari.

6. Halaman Users



No	Full Name	Email Address	Roles	Actions
1	anta	anta@gmail.com	admin.shop	Edit
2	Administrator	admin@gmail.com	admin	Edit
3	Mulya wardhana	mulya@gmail.com	admin.shop	Edit

Gambar 2.6 Halaman User

Gambar 2.6 merupakan halaman yang dimana akan menampilkan User yang dimana user tersebut adalah user untuk menjadi kasir.

7. Tampilan Nota



Permissions - KASIR - Go...
localhost:5173/admin/transac...

Logo
SKY PARK HILL
Alamat: Kedungudi, Trawas Mojokerto
Telp: 082141938309

TANGGAL : 08-Jun-2024 17:14:33
FAKTUR : SPHTRX0003
KASIR : Administrator
PEMBELI : tester

PRODUK	QTY	HARGA
Kopi Matcha	1	15.000
SUB TOTAL :		15.000
DISKON :		
TUNAI :		20.000
KEMBALI :		- 5.000

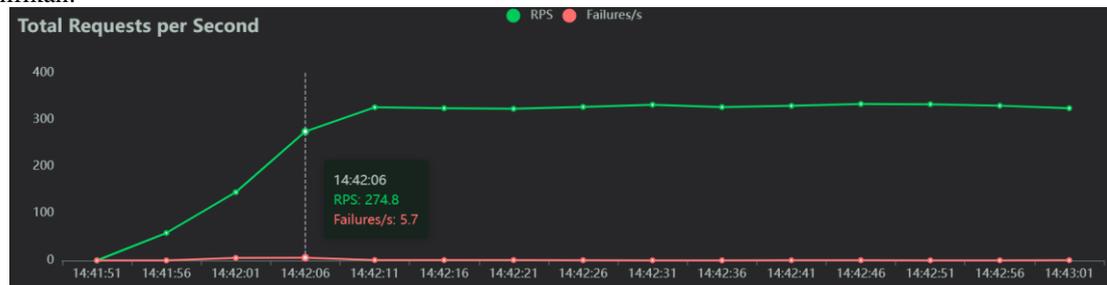
TERIMA KASIH
ATAS KUNJUNGAN ANDA

Gambar 2.7 Tampilan Nota

Gambar 2.7 merupakan tampilan dari nota pemesanan customer yang menampilkan data pesanan yang dipesan oleh customer.

C. Hasil dari Acceptance Testing

Peneliti menggunakan pengujian *Acceptance* testing yang biasanya melibatkan skenario "end-to-end" untuk menguji berbagai fitur dan fungsi yang relevan dari sistem yang telah dikembangkan. Selain itu, peneliti dapat melakukan Load testing menggunakan locust.io untuk memastikan bahwa sistem atau aplikasi dapat menangani jumlah pengguna dan transaksi yang diharapkan tanpa mengalami kegagalan atau penurunan kinerja yang signifikan.



Gambar 3.8 Grafik RPS

Berdasarkan Gambar 3.8 merupakan Grafik yang menunjukkan RPS meningkat hingga 300 pada menit ke-14:42:06 dan stabil. Sistem menangani 300 permintaan per detik tanpa penurunan kinerja signifikan, dengan rata-rata 5-6 kegagalan per detik. Persentase kegagalan kecil, menunjukkan ketersediaan yang baik. Sistem mampu menangani 300 RPS dengan kegagalan yang dapat diterima, namun analisis lanjut diperlukan untuk memastikan kinerja memenuhi persyaratan. Berikut adalah hasil Grafik RPS Bisa dilihat di table 3.1 *Request Statistics*.

Nama Permintaan	Tipe #	Median	Rata-rata	Min	Maks	Rata-rata	RPS		
		Gagal (ms)	rata (ms)	(ms)	(ms)	Ukuran (byte)	Saat Ini	Gagal/s	
/api/admin/carts	GET 5753	11	30	227.83	2	6347	933.21	83.3	0.1
/api/admin/login	GET 5607	25	29	217.11	2	6369	930.83	80.3	0
/api/admin/products	GET 5696	20	29	206.59	2	6332	931.72	78.3	0.1
/api/admin/transaction-orders	GET 5783	20	29	231.46	2	7107	931.77	82.3	0.1
Aggregated	- 22839	76	29	220.82	2	7107	931.89	324.1	0.3

Tabel 3.1 Request Statistics

Tabel 3.1 *Request Statistics* menampilkan hasil tes beban yang dijalankan pada server menggunakan Locust.io. Tes ini berhasil Semua permintaan adalah GET, dengan 4 endpoint diuji: */api/admin/carts*, */api/admin/login*, */api/admin/products*, dan */api/admin/transaction-orders*.

Jumlah permintaan:

- */api/admin/carts* terbanyak dengan 5.753 permintaan.
- */api/admin/transaction-orders* terkecil dengan 5.783 permintaan.

Performa permintaan:

- Median respons tercepat */api/admin/login* 25 ms, terlama */api/admin/transaction-orders* 30 ms.
- Rata-rata respons tercepat */api/admin/login* 29 ms, terlama */api/admin/transaction-orders* 30 ms.
- Min respons tercepat */api/admin/login* dan */api/admin/products* 2 ms, terlama */api/admin/transaction-orders* 2 ms.
- Max respons terlama */api/admin/carts* 6.347 ms, tercepat */api/admin/login* dan */api/admin/products* 2 ms.
- Rata-rata ukuran permintaan terbesar */api/admin/transaction-orders* 7.107 bytes, terkecil */api/admin/login* 6.332 bytes.

RPS (Requests Per Second):

- Rata-rata 324.1 RPS.
- */api/admin/carts* tertinggi 83.3 RPS.
- */api/admin/transaction-orders* terendah 82.2 RPS.

Kegagalan permintaan:

- Tingkat kegagalan keseluruhan 0.3%.

- */api/admin/login* tidak ada kegagalan.
- */api/admin/transaction-orders* tingkat kegagalan tertinggi 0.1%.

Kesimpulan :

Berdasarkan data yang tersedia, tabel Request Static menunjukkan bahwa sistem secara keseluruhan dapat menangani beban hingga sekitar 324.1 RPS dengan tingkat kegagalan yang relatif rendah. Namun, terdapat beberapa endpoint yang memiliki performa yang lebih lambat dan tingkat kegagalan yang lebih tinggi dibandingkan dengan endpoint lainnya.

# Failures	Method	Name	Message
3	GET	<i>/api/admin/carts</i>	ConnectionResetError(10054, 'An existing connection was forcibly closed by the remote host', None, 10054, None)
5	GET	<i>/api/admin/transaction-orders</i>	ConnectionResetError(10054, 'An existing connection was forcibly closed by the remote host', None, 10054, None)
6	GET	<i>/api/admin/login</i>	ConnectionResetError(10054, 'An existing connection was forcibly closed by the remote host', None, 10054, None)
6	GET	<i>/api/admin/products</i>	ConnectionResetError(10054, 'An existing connection was forcibly closed by the remote host', None, 10054, None)
8	GET	<i>/api/admin/carts</i>	ConnectionAbortedError(10053, 'An established connection was aborted by the software in your host machine', None, 10053, None)
14	GET	<i>/api/admin/products</i>	ConnectionAbortedError(10053, 'An established connection was aborted by the software in your host machine', None, 10053, None)
15	GET	<i>/api/admin/transaction-orders</i>	ConnectionAbortedError(10053, 'An established connection was aborted by the software in your host machine', None, 10053, None)
19	GET	<i>/api/admin/login</i>	ConnectionAbortedError(10053, 'An established connection was aborted by the software in your host machine', None, 10053, None)

Tabel 3.2 Failures Statistics

Berdasarkan Tabel 3.2 merupakan Tabel Failures Statistics merangkum kejadian-kejadian error yang terjadi selama load testing. Tabel ini memberikan informasi tentang jumlah kegagalan, metode HTTP, endpoint, dan pesan error spesifik untuk setiap permintaan yang gagal. Berdasarkan Tabel 3.2, Bisa disimpulkan sebagai berikut:

- Masalah Koneksi: Semua kegagalan terkait dengan masalah koneksi, khususnya:
 - *ConnectionResetError(10054)*: Artinya host jarak jauh menutup koneksi secara paksa.
 - *ConnectionAbortedError(10053)*: Artinya koneksi dihentikan oleh perangkat lunak pada host lokal.
- Dampak pada Endpoint: Semua endpoint mengalami kegagalan, dengan */api/admin/login* dan

- `/api/admin/products` memiliki jumlah kegagalan tertinggi.
- Frekuensi: Frekuensi kegagalan berbeda-beda untuk setiap endpoint, dengan `/api/admin/login` memiliki frekuensi tertinggi untuk `ConnectionAbortedError`.

IV. KESIMPULAN

Perancangan sistem Point of Sale mencakup desain database yang mendukung pengelolaan informasi pengguna, pelanggan, produk, kategori, keranjang belanja, serta transaksi. Database ini menghasilkan laporan dan analisis data transaksi untuk mendukung keputusan bisnis dengan akurasi, konsistensi, dan keamanan data. Class diagram mencakup kelas seperti Users, Carts, Products, Transaction, Categories, Customers, dan transaction_Details serta antarmuka seperti formLogin() dan formUsers(). Use case diagram menggambarkan sistem pengelolaan produk dan transaksi oleh admin dan kasir, di mana admin memiliki akses penuh dan kasir memiliki akses terbatas. Desain antarmuka meliputi halaman login, dashboard, products, carts, transaction orders, users, dan tampilan nota. Pengujian penerimaan menunjukkan bahwa sistem mampu menangani 300 permintaan per detik dengan kegagalan yang dapat diterima, rata-rata kegagalan per detik adalah 5-6, dan persentase kegagalan kecil menunjukkan ketersediaan yang baik. Pengujian performa menunjukkan variasi dalam waktu respons dan ukuran permintaan dengan tingkat kegagalan keseluruhan sebesar 0.3%. Secara keseluruhan, sistem Point of Sale ini menunjukkan performa baik dalam menangani beban tinggi dengan tingkat kegagalan rendah.

UCAPAN TERIMA KASIH

Semoga tulisan ini memberikan manfaat dan pemahaman yang mendalam bagi para pembaca. Terima kasih kepada semua pihak yang telah berperan serta dalam proses penulisan dan penyusunan tulisan ini. Dukungan serta kontribusi yang diberikan oleh berbagai pihak selama tahap penelitian dan penulisan sangat berarti. Tanpa bantuan mereka, tulisan ini tidak akan pernah terwujud.

REFERENSI

- [1] R. S. Halim, T. Y. Chandra, and H. A. Mau, "Volume 2 Nomor 9 September 2023 KELALAIAN MEDIS DI INDONESIA," *J. Multidisiplin Indones.*, vol. 2, no. September, pp. 3048–3074, 2023, [Online]. Available: <https://jmi.rivierapublishing.id/index.php/rp/article/view/580/666>
- [2] R. Laksono and J. R. Gultom, "Penggunaan Digital Marketing Dan Poin of Sales (Pos) System Sebagai Strategi Pengembangan Usaha Pada Umkm Warung Tegal Kharisma Bahari Di Jakarta," *Mediastima*, vol. 28, no. 1, pp. 1–10, 2022, doi: 10.55122/mediastima.v28i1.383.
- [3] O. A. I. Susilowaty and M. Rukismono, "Faktor-Faktor Yang Dipertimbangkan Dalam Memilih Moka POS Sebagai Sistem Point Of Sale di Surabaya," *Semin. Nas. Ilmu Terap. V 2021*, pp. 1–7, 2021.
- [4] L. G. N. Iksyam, *Implementasi bisnis aplikasi mecha: layanan perbaikan kendaraan berbasis android menggunakan mvp sebagai design pattern*. 2023. [Online]. Available: dspace.uui.ac.id/123456789/48037
- [5] V. Y. P. Ardhana, M. T. Hidayat, M. Jannah, S. Sumiati, P. Rini, and N. Sari, "Implementasi RESTful API Pada Laravel dan Simulator IoT Wokwi Untuk Pengukuran Suhu dan Kelembaban Menggunakan Metode Waterfall," *Arcitech J. Comput. Sci. Artif. Intell.*, vol. 3, no. 2, p. 93, 2023, doi: 10.29240/arcitech.v3i2.9334.
- [6] I. Setiawan, R. A. Nughroho, and H. Supratikta, "Penerapan arsitektur microservices pada migrasi sistem informasi dwidaya tour dengan pendekatan pengembangan aplikasi cepat (rad)," pp. 1167–1174, 2024, doi: 10.62567/micjo.v1i3.138.
- [7] T. Bratakusuma, I. U. Azmi, and S. Ayuningtiyas, "Pengembangan Back End Pada Aplikasi Alat Tulis Kantor Bank Indonesia Perwakilan Purwokerto Menggunakan Nodejs," *Semin. Nas. Inov. dan Pengemb. Teknol. Terap. Cilacap*, pp. 119–127, 2022, [Online]. Available: <https://ejournal.pnc.ac.id/index.php/senovtek>
- [8] M. I. Syahbana, *Rancang Bangun Aplikasi Monitoring Produksi Harian Mandor Unit Tebenan Pt. Perkebunan Nusantara Vii Berbasis Android*. 2023.
- [9] H. Setiawan, S. Dwi, and Y. Kusuma, "OKTAL : Jurnal Ilmu Komputer dan Science PENERAPAN METODE EXTREME PROGRAMMING PADA PERANCANGAN APLIKASI POINT OF SALES BERBASIS ANDROID (STUDI KASUS KEDAI KOPI MOKARA)," vol. 2, no. 12, pp. 3268–3282, 2023.
- [10] D. Oscar, Hendri, M. chwanul Muslim, and M. Fahmi, "Jurnal Vocational Teknik Elektronika dan Informatika Rancang Bangun Sistem Informasi Point Of Sales (POS)," *J. Vocat. Tek. Elektron. dan Inform.*, vol. 11, no. 3, 2023, [Online]. Available: <http://ejournal.unp.ac.id/index.php/voteknika/>
- [11] R. Nandang Pratama and Y. A. Susetyo, "Implementasi Python API dengan Framework Flask sebagai Cloud Run Service Untuk Proses Update di PT. XYZ," vol. 5, no. 2, pp. 669–676, 2024.
- [12] A. Simangunsong and A. A. Nababan, "Pkm Pemanfaatan Aplikasi Point Of Sale (Pos) Pada Pos Coffee Berbasis Digital," *J. Pengabd. Kpd. Masy. Nusant.*, vol. 4, no. 3, pp. 2113–2117, 2023.
- [13] R. Afriansyah, M. Sholeh, and D. Andayati, "Perancangan Aplikasi Pemrograman Antarmuka Berbasis Web Menggunakan Gaya Arsitektur Representasi Untuk Sistem Presensi Sekolah," *J. Scr.*, vol. 9, no. 1, pp. 84–93, 2021.
- [14] I. G. Wiryawan, M. D. Agustiningsih, M. Yusuf, V. A. Pratama, and L. D. Wahyuningsih, "Z-Waste: Aplikasi Ramah Lingkungan Berbasis Mobile," *J. Resist. (Rekayasa Sist. Komputer)*, vol. 4, no. 1, pp. 26–32, 2021, doi: 10.31598/jurnalresistor.v4i1.664.
- [15] R. Nuari, "Sistem Informasi Pendaftaran Siswa Baru Menggunakan Extreme Programming," *J. Data Sci. Inf. Syst.*, vol. 1, no. 4, pp. 166–174, 2023, [Online]. Available: <https://doi.org/10.58602/dimis.v1i4.79>

- [16] F. Aisyah Mutia Dawis, Yusuf Wahyu Setiya Putra, N. R. F. Dini Hamidin, Syifa Nurgaida Yutia, Maniah, and F. N. Dea Wemona Rahma, *REKAYASA PERANGKAT LUNAK PANDUAN PRAKTIS UNTUK PENGEMBANGAN APLIKASI BERKUALITAS*. 2023.

Conflict of Interest Statement:

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.