

INTEGRASI PAYMENT GATEWAY PADA APLIKASI POINT OF SALES BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK REACTJS (STUDI KASUS: TOKO ADIDA PRATAMA)

Muhammad Idzhar Surya Pratama^{1*}, Budi Arif Dermawan², Carudin³

1,2,3 Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang;

Jl. H.S. Ronggowaluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361

Keywords:

Point of Sales, Toko Sembako, React JS, Payment Gateway, Midtrans

Correspondent Email:

kevinchocs15@gmail.com

Abstrak. Toko sembako, sebagai bagian dari Usaha Mikro, Kecil, dan Menengah (UMKM), berperan penting dalam mendukung perekonomian lokal dan nasional. Namun, kegiatan operasionalnya masih menghadapi berbagai kendala seperti pencatatan transaksi manual, terbatasnya metode pembayaran, serta pengelolaan stok yang belum efisien. Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi Point of Sale (POS) berbasis web yang terintegrasi dengan sistem payment gateway untuk meningkatkan efisiensi operasional, khususnya di Toko Adida Pratama. Aplikasi dikembangkan menggunakan ReactJS untuk antarmuka frontend, ExpressJS sebagai backend, dan Prisma ORM dalam pengelolaan basis data. Metode prototyping diterapkan untuk memastikan sistem dapat dikembangkan secara bertahap dan responsif terhadap kebutuhan pengguna. Integrasi dengan Midtrans melalui Snap API memungkinkan pembayaran non-tunai seperti QRIS, e-wallet, dan transfer bank. Hasil pengujian black-box menunjukkan bahwa seluruh fitur berjalan sesuai spesifikasi. Dengan demikian, aplikasi ini dapat menjadi solusi digital yang efektif dalam pengelolaan transaksi dan pembayaran di toko sembako.



JITET is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Abstract. Grocery stores, as part of Micro, Small and Medium Enterprises (MSMEs), play an important role in supporting the local and national economy. However, operational activities still face various obstacles such as manual transaction recording, limited payment methods, and inefficient stock management. This research aims to design and develop a web-based Point of Sale (POS) application that is integrated with a payment gateway system to increase operational efficiency, especially at the Adida Pratama Store. The application was developed using ReactJS for the frontend interface, ExpressJS as the backend, and Prisma ORM for database management. The prototyping method is applied to ensure the system can be developed in stages and is responsive to user needs. Integration with Midtrans via Snap API enables non-cash payments such as QRIS, e-wallet and bank transfers. Black-box testing results show that all features work according to specifications. Thus, this application can be an effective digital solution in managing transactions and payments in grocery stores.

1. PENDAHULUAN

Toko sembako sebagai bagian dari UMKM memiliki peran penting dalam ekonomi

Indonesia, terutama dalam penyediaan kebutuhan pokok dan penciptaan lapangan kerja. Menurut Kemenko Perekonomian

(2023), toko sembako adalah bisnis mikro prospektif dengan kontribusi signifikan terhadap pertumbuhan ekonomi [1]. BPS mencatat sektor ini menyumbang sekitar 15% PDB dan menjadi penyerap tenaga kerja besar. Namun, di tengah transformasi digital, sebagian besar toko sembako masih tertinggal secara teknologi. Masalah umum meliputi pencatatan manual, pengelolaan stok tidak efisien, dan keterbatasan metode pembayaran non-tunai. Hingga Mei 2022, hanya 29,6% UMKM telah bergabung dengan platform digital, menunjukkan masih banyak yang belum terdigitalisasi.[2]

Toko Adida Pratama di Karawang dipilih sebagai studi kasus karena mewakili toko sembako tradisional dengan omzet harian tinggi (3–5 juta rupiah) namun masih menggunakan sistem manual. Permasalahan utama meliputi pencatatan transaksi kertas, kesulitan menghitung pendapatan, belum mendukung pembayaran digital, dan keterlambatan pemantauan stok. Toko ini dipilih secara purposive karena relevan dengan tujuan penelitian dan memungkinkan pengumpulan data langsung dari pemilik.

Untuk mengatasi masalah tersebut, dibutuhkan sistem *Point of Sales* (POS) berbasis web yang terintegrasi dengan *payment gateway*, agar transaksi lebih efisien, akurat, dan cepat. Sistem ini memungkinkan pencatatan digital, pengelolaan stok otomatis, dan pembayaran non-tunai seperti QRIS atau e-wallet. Penelitian sebelumnya menunjukkan bahwa POS dapat membantu efisiensi transaksi, meskipun masih ada keterbatasan pada integrasi pembayaran dan fitur cetak otomatis. Oleh karena itu, penelitian ini bertujuan mengembangkan aplikasi POS berbasis web menggunakan framework ReactJS dengan integrasi *payment gateway*, serta mengujinya secara langsung melalui *Black Box Testing* untuk hasil yang lebih aplikatif dan siap diterapkan.

2. TINJAUAN PUSTAKA

2.1. Integrasi

Integrasi sistem merupakan proses menghubungkan berbagai sistem dan aplikasi komputer agar dapat bekerja secara terpadu, baik secara fisik maupun fungsional. [3]

2.2. Payment Gateway

Payment gateway merupakan salah satu mekanisme yang digunakan untuk memproses transaksi secara elektronik. Sistem ini bertindak sebagai perantara antara pelanggan, pelaku usaha, dan institusi perbankan dalam proses pembayaran digital [4].

2.3. Aplikasi

Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk menyelesaikan tugas-tugas khusus pengguna komputer. Hal ini menunjukkan bahwa aplikasi tidak hanya sekadar antarmuka, melainkan juga mengatur alur kerja dan proses di balik layar guna menghasilkan keluaran yang sesuai harapan.[5]

2.4. Point of Sales

Point Of Sales (POS) adalah tempat di mana transaksi diselesaikan, yakni saat pelanggan melakukan pembayaran atas barang atau jasa yang diperoleh, dan sistem kemudian menghasilkan tanda terima sebagai bukti transaksi.[6]

2.5. Website

Situs web adalah kumpulan informasi atau halaman yang dapat diakses oleh siapa saja melalui jaringan internet kapan saja dan di mana saja, asalkan terhubung secara online. Secara teknis, situs web terdiri dari berbagai halaman yang tergabung dalam satu domain atau subdomain tertentu [7].

2.6. Framework

Framework adalah kumpulan fungsi yang memberikan bantuan kepada pengembang perangkat lunak dalam menghadapi berbagai masalah pemrograman, seperti mengakses *database* [8].

2.7. Reactjs

React JS adalah sebuah pustaka (library) JavaScript bersifat open-source yang digunakan untuk membangun antarmuka pengguna (user interface) khususnya pada aplikasi web berbasis satu halaman (single-page application). React berfokus pada efisiensi dalam pembaruan dan penyajian data yang sering berubah, serta memudahkan pengembangan antarmuka yang kompleks dengan struktur yang lebih terorganisi [9].

2.8. Black Box Testing

Black box testing merupakan teknik pengujian perangkat lunak yang berfokus pada pengujian aspek fungsional tanpa melihat struktur internal kode. Metode ini bertujuan untuk menemukan kesalahan yang berkaitan dengan fungsi sistem, antarmuka pengguna, pengelolaan data, kinerja, serta proses awal dan akhir dari aplikasi.[10].

2.9. Database

Database adalah suatu kumpulan data yang berhubungan secara logika dan deskripsi dari data-data yang dirancang untuk memenuhi kebutuhan informasi dalam suatu organisasi [11]. *Database* menawarkan keuntungan berupa penyimpanan data dengan format yang fleksibel dan independen, sehingga perubahan pada struktur database tidak secara langsung memengaruhi aplikasi yang menggunakannya

2.10. Model Prototype

Model *Prototype* merupakan salah satu metode pengembangan sistem yang banyak digunakan untuk meminimalkan kesalahan sistem sejak tahap awal.[12] model prototype merupakan teknik pengembangan sistem yang menggunakan *prototype* sistem untuk menggambarkan sistem, sehingga pengguna atau pemilik sistem mempunyai gambaran pengembangan sistem yang akan dilakukan. Dengan pendekatan ini, pengguna dapat memberikan masukan sejak tahap awal perancangan, sehingga iterasi dan penyempurnaan sistem dapat dilakukan lebih cepat dan lebih tepat sasaran.

Dalam konteks pengembangan sistem informasi, penggunaan model prototype sangat efektif terutama ketika kebutuhan pengguna belum sepenuhnya terdokumentasi atau masih memungkinkan untuk berubah. Pendekatan ini menekankan keterlibatan aktif pengguna melalui pembuatan model awal (*prototype*) yang dapat dievaluasi dan diperbaiki sebelum implementasi akhir dilakukan.

2.11. Javascript

JavaScript adalah bahasa pemrograman yang digunakan untuk memanipulasi elemen HTML, sehingga meningkatkan interaktivitas halaman web [13]. *JavaScript* menambahkan logika dan perilaku dinamis pada struktur (*HTML*) dan tampilan (*CSS*). Contohnya,

JavaScript memungkinkan aksi seperti menampilkan seluruh isi artikel saat pengguna mengklik elemen `<h1>`.

2.12. Midtrans

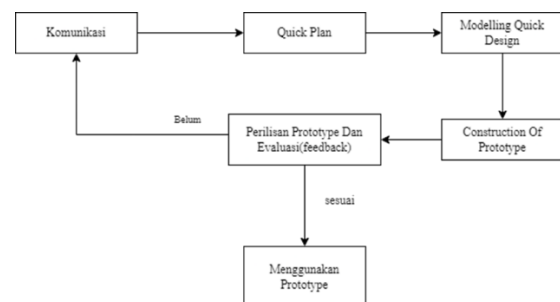
[14] *Midtrans* merupakan salah satu layanan *payment gateway* yang dirancang untuk memudahkan proses pengujian dan pelaksanaan pembayaran. Dengan sistem yang sederhana, pengguna cukup memasukkan kode transaksi dan menekan tombol bayar untuk menyelesaikan pembayaran secara otomatis.

2.13. Bootstrap

[15] *Bootstrap* adalah *framework front-end* yang membantu mempercepat dan menyederhanakan tampilan web dan seluler, khususnya dalam pengembangan web. *Bootstrap* menyediakan berbagai fitur berguna, termasuk HTML, CSS, dan *JavaScript* bawaan yang siap digunakan dan mudah dikembangkan. Dalam pengembangannya, *Bootstrap* juga menyediakan komponen antarmuka seperti tombol, navigasi, dan formulir yang dapat langsung digunakan serta mendukung tampilan yang responsif di berbagai ukuran layar, baik desktop maupun seluler.

3. METODE PENELITIAN

Metodologi penelitian yang diterapkan dalam penelitian ini adalah model *prototype*. Model *prototype* dimulai dengan pengumpulan kebutuhan dari pengguna terhadap perangkat lunak yang akan dibangun. Program prototype ini dievaluasi oleh pengguna secara berkelanjutan hingga spesifikasi yang sesuai dengan harapan pengguna ditemukan. Model *prototype* dipilih karena kebutuhan pengguna yang bisa berubah-ubah, dan pengguna memiliki peran aktif dalam proses pengembangan untuk mendeteksi potensi kesalahan lebih awal.



Gambar 1 Rancangan Penelitian

Metode Prototype memiliki lima tahapan pengembangan sistem seperti yang tertera pada Gambar 1. Diatas yaitu Komunikasi, *Quick Plan*, *Modelling Quick Design*, *Construction Of Prototype* dan *Perilisan Prototype* dan Evaluasi.

3.1. Komunikasi

Pada tahap ini, penulis bekerja sama dengan pengguna atau pihak terkait untuk memahami, mengetahui kebutuhan sistem dengan beberapa cara, yaitu observasi, wawancara dan studi literatur. Informasi yang di peroleh dari ketiga cara ini menjadikan dasar dalam desain dan pengembangan *prototype*, serta akan diulang setiap iterasi agar sistem sesuai dengan kebutuhan dan keinginan pengguna atau klien.

3.2. Quick Plan

Tahap ini adalah tahap lanjutan dari komunikasi, setelah komunikasi selesai, maka penulis akan membuat desain sederhana yang menggambarkan tentang sistem sesuai dengan kebutuhan yang ada pada tahap komunikasi. Pembuatan design sederhana akan menggunakan *tools* Figma.

3.3. Modelling Quick Design

Dalam proses ini, penulis memanfaatkan berbagai alat bantu UML, seperti *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, dan *Sequence Diagram*, yang disusun berdasarkan kebutuhan dan informasi yang penulis peroleh dari *user*. Tujuan dari perancangan ini bukan hanya untuk menggambarkan alur kerja sistem, interaksi antar komponen, serta hubungan antara aktor dengan sistem, tetapi juga untuk memastikan setiap elemen desain dapat mendukung implementasi sistem secara optimal.

3.4. Construction of Prototype

Pada tahap ini penulis mengembangkan prototipe dan melakukan pengkodean yaitu membangun web berdasarkan desain dan kebutuhan yang telah ditentukan dalam pengkodean ini. Penulis menggunakan *tools Visual Studio Code*. Dengan mengimplementasikan fitur-fitur inti dalam bentuk awal yang dapat berfungsi.

3.5. Perilisan Prototype dan Evaluasi

Pada tahap ini penulis akan melakukan pertemuan dengan pengguna untuk mempresentasikan prototipe dengan tujuan untuk mendapatkan umpan balik yang konstruktif dari user. Penulis juga memperkenalkan *prototype* yang telah dibuat dan memberikan kesempatan kepada pengguna untuk mencobanya. Pertemuan ini juga bertujuan untuk mendiskusikan pengalaman dari pengguna, termasuk fitur yang berfungsi dengan baik dan hal yang harus diperbaiki.

4. HASIL DAN PEMBAHASAN

4.1. Komunikasi dan Perencanaan Cepat

Tahap ini diawali dengan pengumpulan dan analisis data untuk mengidentifikasi permasalahan serta kebutuhan pengguna, dalam hal ini pemilik Toko Adida Pratama. Komunikasi dilakukan secara intensif guna memperoleh pemahaman menyeluruh terhadap proses bisnis yang berjalan. Hasil analisis digunakan sebagai dasar dalam perancangan awal sistem

4.1.1 Identifikasi Kebutuhan Pengguna

Dari hasil observasi dan wawancara dengan narasumber, identifikasi kebutuhan sebagai berikut:

1. Sistem manajemen toko yang mudah digunakan Sistem memiliki antarmuka yang sederhana dan mudah dipahami oleh pengguna, terutama oleh pemilik toko yang belum terbiasa dengan teknologi digital.
2. Fitur pencatatan transaksi otomatis Sistem dapat mencatat setiap transaksi penjualan secara otomatis untuk menghindari pencatatan manual yang rentan terhadap kesalahan.
3. Pengelolaan stok barang secara digital dan real-time Sistem mampu mencatat, mengurangi, dan memperbarui jumlah stok barang secara otomatis setiap kali terjadi transaksi.
4. Menyediakan berbagai metode pembayaran Sistem menyediakan opsi pembayaran non-tunai seperti transfer bank dan QRIS untuk memudahkan pelanggan bertransaksi.
5. Pembuatan laporan penjualan, pembelian, produk dan supplier Sistem mampu menghasilkan riwayat penjualan dan pembelian harian, mingguan, dan bulanan untuk memudahkan analisis dan evaluasi bisnis.

6. Kemudahan pencarian dan pengelolaan produk Pengguna dapat mencari produk, menambah, mengubah, atau menghapus data produk dengan cepat.
7. Fitur pencetakan nota otomatis Setelah transaksi selesai, sistem dapat mencetak nota sebagai bukti pembayaran bagi pelanggan dan dokumentasi toko.

4.1.2 Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah langkah dari beberapa kegiatan yang akan diperlukan saat membangun sistem aplikasi. Merujuk dari solusi sistem yang sudah dianalisa sebelumnya, peneliti dapat menyimpulkan beberapa fitur yang dibutuhkan pada sistem dapat dilihat pada Tabel 1 berikut

Tabel 1 Kebutuhan Fungsional

No	Kebutuhan
1	Login
2	Halaman <i>Dashboard</i>
3	Katalog
4	Transaksi
5	Laporan
6	Kelola Produk
7	Kelola Kategori
9	Kelola <i>Supplier</i>
10	Kasir
11	Riwayat Penjualan
12	Riwayat Pembelian
13	Laporan Produk
14	Laporan Penjualan
15	Laporan Supplier
16	Logout

4.1.3 Analisis Kebutuhan Non Fungsional

Kebutuhan non-fungsional merupakan kebutuhan teknis yang diperlukan dalam pengembangan aplikasi, pengujian aplikasi, dan penggunaan aplikasi yang dikembangkan. Berikut adalah tabel kebutuhan non-fungsional untuk pengembang pada Tabel 2

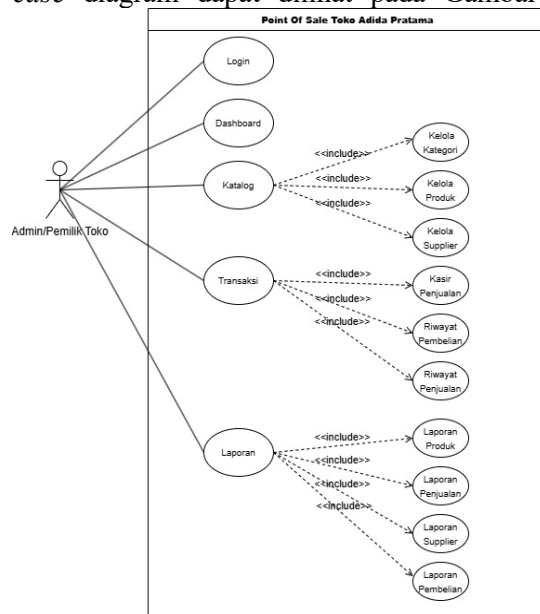
Tabel 2 Kebutuhan Non Fungsional

No	Nama Perangkat	Jenis
1	Visual Studio Code	Perangkat Lunak
2	XAMPP	Perangkat Lunak
3	JavaScript	Perangkat Lunak

4	React JS dan Vite	Perangkat Lunak
5	Bootstrap	Perangkat Lunak
6	Brave Browser	Perangkat Lunak
7	Figma	Perangkat Lunak
9	Draw.io	Perangkat Lunak
10	Internet	Perangkat Lunak
11	Laptop	Perangkat Keras

4.1.4 Use Case Diagram

Use Case Diagram digunakan untuk memodelkan interaksi antara aktor dan sistem, serta menjelaskan jenis interaksi yang terjadi antara pengguna dan sistem. Pada aplikasi *Point of Sale* Toko Adida Pratama, rancangan *use case diagram* dapat dilihat pada Gambar 2

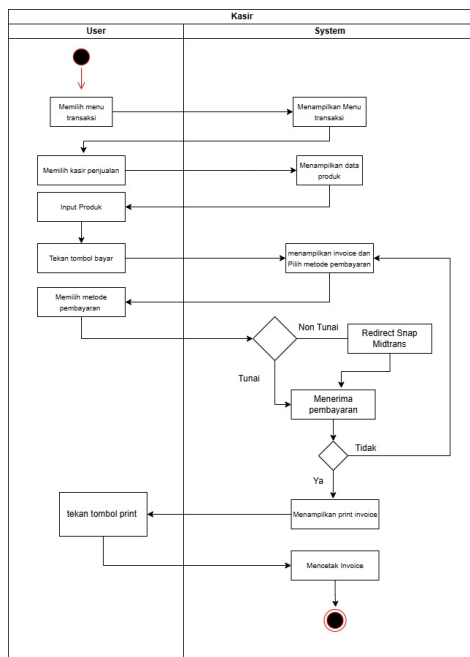


Gambar 2 Use Case Diagram POS

Pada use case diagram di atas, terdapat satu aktor yang berinteraksi dengan sistem, yaitu pemilik (admin).

4.1.5 Activity Diagram

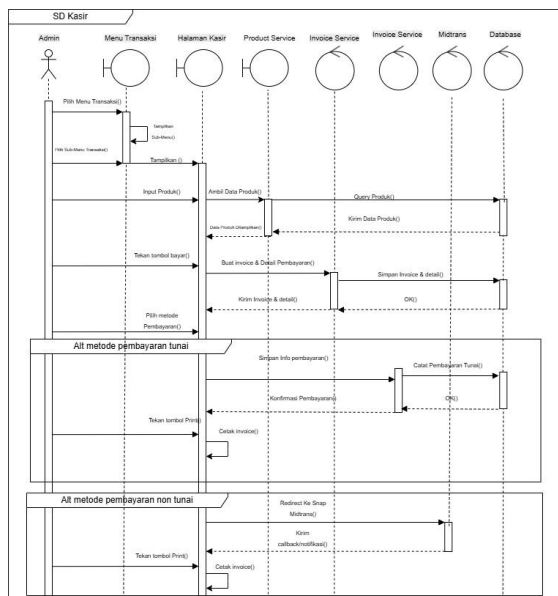
Activity Diagram merupakan diagram yang menggambarkan alur kerja atau aktivitas dalam suatu sistem. Diagram ini menyusun urutan proses dari awal hingga akhir melalui simbol-simbol khusus yang dihubungkan oleh panah sebagai penunjuk arah aliran aktivitas.



Gambar 3 Activity Diagram Kasir

4.1.6 Sequence Diagram

Sequence Diagram adalah sebuah diagram yang menggambarkan urutan atau alur dari interaksi antara objek dalam suatu sistem. Diagram ini memperlihatkan bagaimana objek saling berinteraksi dalam hal pesan atau pesan yang dikirimkan di antara objek tersebut.

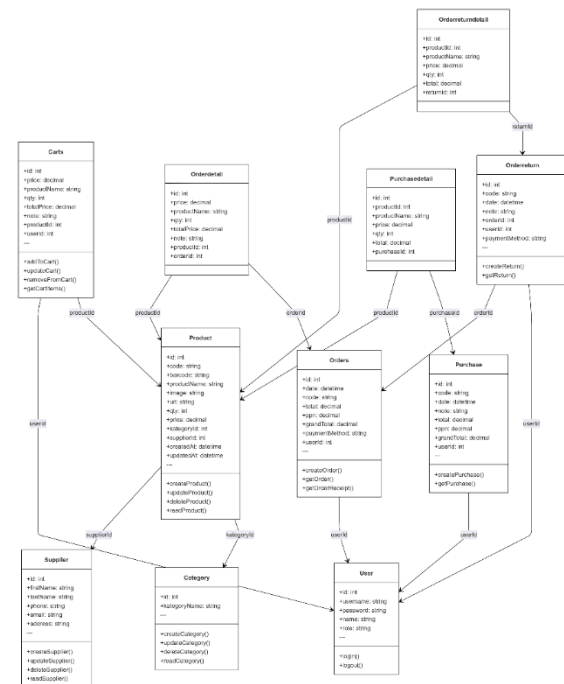


Gambar 4 Sequence Diagram Kasir

4.1.7 Class Diagram

Struktur sistem dalam Point Of Sales (POS) ini memfasilitasi interaksi antara berbagai objek yang saling terkait, yang dapat

diilustrasikan melalui sebuah *class diagram* seperti yang terlihat pada Gambar 5



Gambar 5 Class Diagram

4.2. Implementasi Payment Gateway

Integrasi payment gateway dilakukan menggunakan layanan Midtrans untuk mendukung pembayaran non-tunai seperti QRIS, e-wallet, dan transfer bank. Setelah transaksi diproses, sistem akan mengarahkan pengguna ke halaman pembayaran dan mencatat status pembayaran secara otomatis guna meningkatkan efisiensi dan kemudahan transaksi. Berikut merupakan implementasi dari kode program pada proses integrasi payment gateway ke dalam sistem Point of Sales.

```

1. import java.util.*;
2. import java.io.*;
3. import java.net.*;
4. import java.sql.*;
5. import java.util.*;
6. import java.util.*;
7. import java.util.*;
8. import java.util.*;
9. import java.util.*;
10. import java.util.*;
11. import java.util.*;
12. import java.util.*;
13. import java.util.*;
14. import java.util.*;
15. import java.util.*;
16. import java.util.*;
17. import java.util.*;
18. import java.util.*;
19. import java.util.*;
20. import java.util.*;
21. import java.util.*;
22. import java.util.*;
23. import java.util.*;
24. import java.util.*;
25. import java.util.*;
26. import java.util.*;
27. import java.util.*;
28. import java.util.*;
29. import java.util.*;
30. import java.util.*;
31. import java.util.*;
32. import java.util.*;
33. import java.util.*;
34. import java.util.*;
35. import java.util.*;
36. import java.util.*;
37. import java.util.*;
38. import java.util.*;
39. import java.util.*;
40. import java.util.*;
41. import java.util.*;
42. import java.util.*;
43. import java.util.*;
44. import java.util.*;
45. import java.util.*;
46. import java.util.*;
47. import java.util.*;
48. import java.util.*;
49. import java.util.*;
50. import java.util.*;
51. import java.util.*;
52. import java.util.*;
53. import java.util.*;
54. import java.util.*;
55. import java.util.*;
56. import java.util.*;
57. import java.util.*;
58. import java.util.*;
59. import java.util.*;
60. import java.util.*;
61. import java.util.*;
62. import java.util.*;
63. import java.util.*;
64. import java.util.*;
65. import java.util.*;
66. import java.util.*;
67. import java.util.*;
68. import java.util.*;
69. import java.util.*;
70. import java.util.*;
71. import java.util.*;
72. import java.util.*;
73. import java.util.*;
74. import java.util.*;
75. import java.util.*;
76. import java.util.*;
77. import java.util.*;
78. import java.util.*;
79. import java.util.*;
80. import java.util.*;
81. import java.util.*;
82. import java.util.*;
83. import java.util.*;
84. import java.util.*;
85. import java.util.*;
86. import java.util.*;
87. import java.util.*;
88. import java.util.*;
89. import java.util.*;
90. import java.util.*;
91. import java.util.*;
92. import java.util.*;
93. import java.util.*;
94. import java.util.*;
95. import java.util.*;
96. import java.util.*;
97. import java.util.*;
98. import java.util.*;
99. import java.util.*;
100. import java.util.*;

```

Gambar 6 Implementasi *payment Gateway*

4.3. Implementasi *Prototype*

Implementasi *prototype* merupakan tahap di mana rancangan *prototype* yang telah disesuaikan dengan kebutuhan pengguna direalisasikan menjadi sistem yang berfungsi secara nyata. Pada tahap ini, desain antarmuka dan alur fitur yang sebelumnya dirancang dalam bentuk *prototype* diimplementasikan ke dalam aplikasi Point of Sales. Hasilnya ditampilkan dalam bentuk halaman-halaman sistem yang dapat langsung digunakan oleh pengguna sesuai dengan fungsinya masing-masing.

4.3.1. Halaman Login

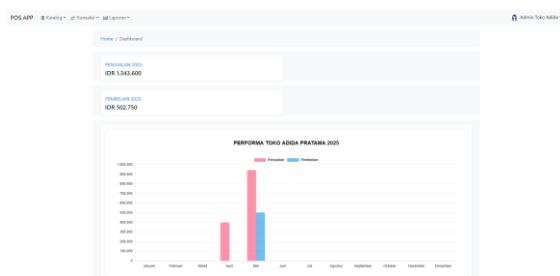
Halaman Login adalah halaman yang digunakan oleh admin untuk masuk kedalam sistem. Halaman Login menampilkan form username dan Password sebagaimana yang ditunjukkan dalam Gambar 7.

The screenshot shows a login interface with a light blue background. At the top, it says 'Selamat Datang!' and 'Silakan login untuk melanjutkan'. Below this are two input fields: 'Username' with the placeholder 'Masukkan username' and 'Password' with the placeholder 'Masukkan password'. A blue 'Login' button is at the bottom.

Gambar 7 Implementasi *payment Gateway*

4.3.2. Halaman Dashboard

Halaman *Dashboard* adalah halaman yang berisi ringkasan dari keseluruhan isi dari *platform* ini, berisi tentang keuntungan penjualan dan data pembelian selama periode tahun.



Gambar 8 Halaman Dashboard

4.3.3. Halaman Kategori

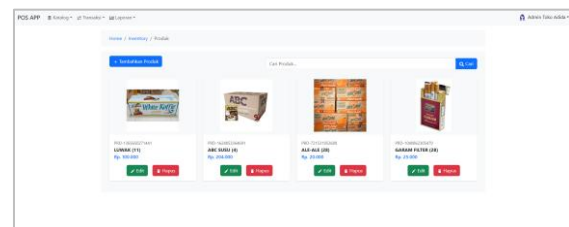
Halaman Kelola Kategori adalah halaman yang digunakan oleh admin untuk mengelola Kategori seperti tambah, ubah dan hapus kategori produk.

#	Kategori	Aksi
1	Baru	+ Tambah # Edit # Hapus
2	Makanan	+ Tambah # Edit # Hapus
3	Minuman	+ Tambah # Edit # Hapus

Gambar 9 Halaman Kelola Kategori

4.3.4. Halaman Kelola Produk

Halaman Kelola Produk adalah halaman yang digunakan oleh admin untuk mengelola produk seperti menambah, mengubah dan menghapus produk



Gambar 10 Halaman Kelola Produk

4.3.5. Halaman Kelola Supplier

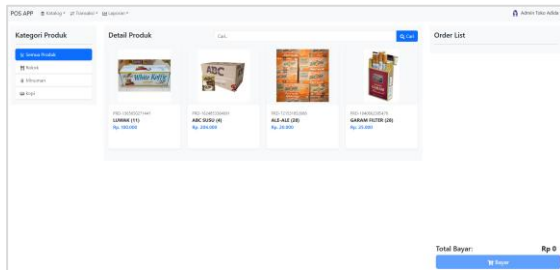
Halaman Kelola *Supplier* adalah halaman yang digunakan oleh admin untuk mengelola *Supplier* seperti menambah, mengubah dan menghapus *Supplier*

#	Nama	Alamat	Email	Aksi
1	Suplier PT	Jember	suplier@pt.com	+ Tambah # Edit # Hapus
2	Suplier PT	Jember	suplier@pt.com	+ Tambah # Edit # Hapus

Gambar 11 Halaman Kelola supplier

4.3.6. Halaman Kasir Penjualan

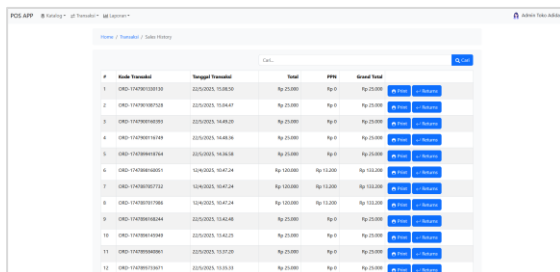
Halaman Kasir Penjualan adalah halaman yang digunakan oleh admin untuk melakukan transaksi pada *platform*



Gambar 12 Halaman Kasir Penjualan

4.3.7. Halaman Riwayat Penjualan

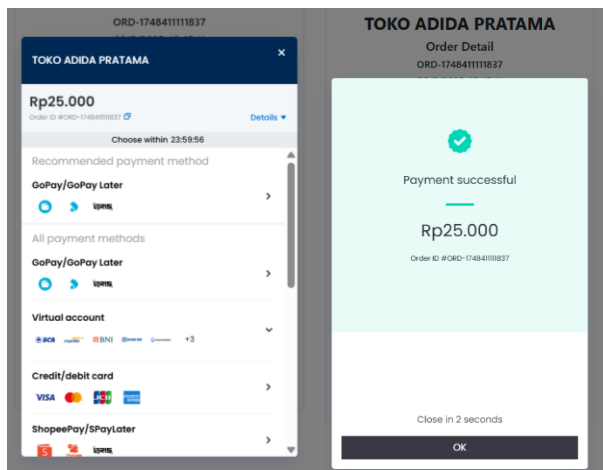
Halaman Riwayat Penjualan adalah halaman yang digunakan admin untuk mengecek riwayat transaksi yang sudah dilakukan.



Gambar 13 Halaman Riwayat Penjualan

4.3.8. Halaman Payment Gateway

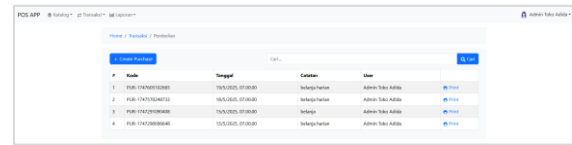
Halaman ini digunakan untuk memproses pembayaran non-tunai melalui integrasi dengan Midtrans, seperti QRIS, e-wallet, dan transfer bank, guna mempermudah dan mempercepat transaksi.



Gambar 14 Halaman Riwayat Pembelian

4.3.9. Halaman Riwayat Pembelian

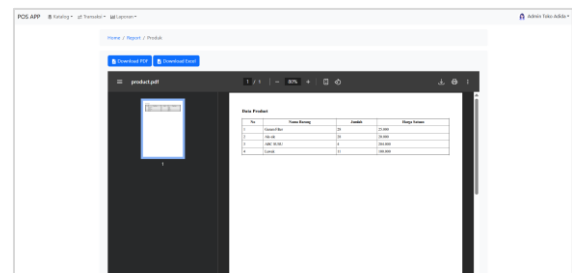
Halaman Riwayat Pembelian adalah halaman yang digunakan admin untuk mengecek riwayat pembelian dari *supplier*



Gambar 15 Halaman Riwayat Pembelian

4.3.10. Halaman Laporan Produk

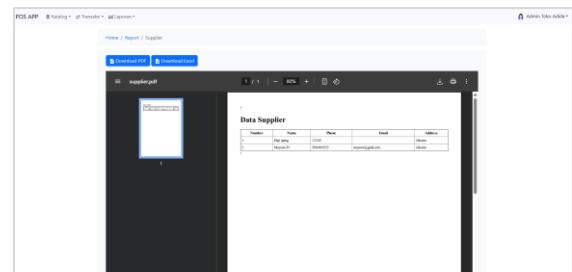
Halaman Laporan Produk adalah halaman yang digunakan admin untuk melihat dan mengunduh laporan produk.



Gambar 16 Halaman Laporan Produk

4.3.11. Halaman Laporan Supplier

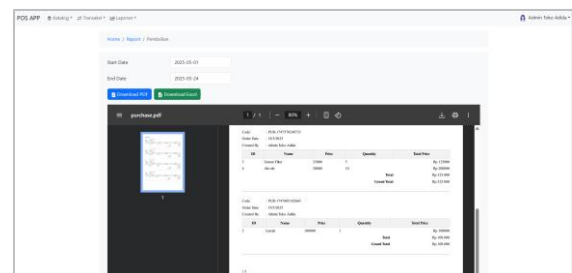
Halaman Laporan *Supplier* adalah halaman yang digunakan admin untuk melihat dan mengunduh laporan *supplier*.



Gambar 17 Halaman Laporan Supplier

4.3.12. Halaman Laporan Pembelian

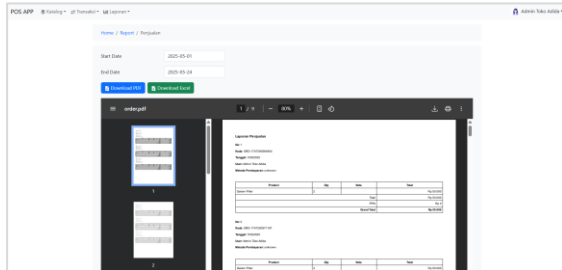
Halaman Laporan pembelian adalah halaman yang digunakan admin untuk melihat dan mengunduh laporan pembelian dari *supplier* berdasarkan rentang waktu tertentu



Gambar 18 Halaman Laporan Pembelian

4.3.13. Halaman Laporan Penjualan

Halaman Laporan pembelian adalah halaman yang digunakan admin untuk melihat dan mengunduh laporan penjualan berdasarkan rentang waktu tertentu



Gambar 19 Halaman Laporan Penjualan

4.3 Pengujian

Pada tahap ini, sistem diuji secara menyeluruh untuk memastikan semua fitur dan fungsi bekerja dengan benar serta mengidentifikasi dan memperbaiki potensi kesalahan. Pengujian ini meliputi black box testing dan user acceptance testing.

Pengujian Black Box dilakukan untuk memastikan bahwa setiap fitur dalam sistem berjalan sesuai dengan kebutuhan fungsional yang telah dirancang, tanpa memeriksa struktur internal dari kode program. Fokus utama pengujian ini adalah mengamati apakah output yang dihasilkan sesuai dengan input yang diberikan oleh pengguna. Berikut adalah rangkuman hasil *Black Box Testing*.

Tabel 3 Pengujian Black Box

No	Komponen Yang diuji	Hasil yang diharapkan	Hasil Pengujian
1	Login	Menampilkan halaman dashboard jika data login benar dan menampilkan pesan login gagal jika data yang dimasukkan salah	Sesuai Harapan
2	Kelola Kategori	Menampilkan halaman menu kategori dan fitur dapat	Sesuai Harapan

		berfungsi dengan baik	
3	Kelola Produk	Menampilkan halaman menu Produk dan fitur dapat berfungsi dengan baik	Sesuai Harapan
4	Kelola Supplier	Menampilkan halaman menu Produk dan fitur dapat berfungsi dengan baik	Sesuai Harapan
5	Kasir Penjualan	Menampilkan halaman menu kasir penjualan dan fitur dapat berfungsi dengan baik	Sesuai Harapan
6	Riwayat Penjualan	Menampilkan halaman menu riwayat penjualan dan fitur dapat berfungsi dengan baik	Sesuai Harapan
7	Riwayat Pembelian	Menampilkan halaman menu riwayat pembelian dan fitur dapat berfungsi dengan baik	Sesuai Harapan
8	Laporan Produk	Menampilkan halaman menu Laporan Produk dan fitur dapat berfungsi dengan baik	Sesuai Harapan
9	Laporan Supplier	Menampilkan halaman menu Laporan Supplier dan fitur dapat berfungsi dengan baik	Sesuai Harapan

10	Laporan Pembelian	Menampilkan halaman menu Laporan Pembelian dan fitur dapat berfungsi dengan baik	Sesuai Harapan
11	Logout	Menampilkan halaman menu Logout dan fitur dapat berfungsi dengan baik	Sesuai Harapan

5. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan mengenai Integrasi payment gateway pada aplikasi Point Of Sales berbasis web menggunakan framework reactjs di Toko adida pratama, maka dapat disimpulkan sebagai berikut:

- Penelitian ini berhasil merancang dan mengembangkan aplikasi Point Of Sales (POS) berbasis web menggunakan React JS dan Express JS, dengan Prisma ORM untuk pengelolaan database. Metode prototyping memungkinkan iterasi yang responsif terhadap masukan pengguna. Aplikasi ini berhasil menjadi solusi atas permasalahan proses transaksi yang sebelumnya masih dilakukan secara konvensional, serta memberikan sistem yang lebih efisien, terstruktur, dan terintegrasi dibandingkan metode manual.
- Aplikasi berhasil diintegrasikan dengan Midtrans menggunakan pendekatan Snap API, yang memungkinkan berbagai metode pembayaran digital seperti QRIS, e-wallet, dan transfer bank melalui mekanisme popup. Integrasi ini memberikan solusi pembayaran non-tunai yang aman, praktis, dan sesuai dengan kebutuhan operasional toko.
- Berdasarkan pengujian dengan metode *black box*, seluruh skenario yang diuji pada masing-masing fungsionalitas menunjukkan hasil yang sesuai dengan spesifikasi sistem. Seluruh fitur berjalan dengan baik tanpa ditemukannya bug atau kesalahan selama proses pengujian.

- Saran dalam pengembangan sistem meliputi perlunya pembaruan teknologi secara berkala agar sistem tetap mengikuti perkembangan teknologi terkini. Selain itu, penambahan fitur baru dapat dilakukan sesuai kebutuhan di masa mendatang. Pemeliharaan sistem yang rutin juga penting untuk memastikan kinerja tetap stabil dan optimal.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik. (2021). *Statistik Indonesia 2021*. Jakarta: Badan Pusat Statistik. Tersedia di: <https://www.bps.go.id/publication/2021/02/26/1c6bdb5fadc774ebf6c5b3db/statistik-indonesia-2021.html>
- [2] Kementerian Koperasi dan UKM Republik Indonesia. (2022). Peta Jalan Transformasi Digital UMKM 2021–2024.
- [3] W. and D. I. Sensuse, *Model Integrasi Sistem dengan Pendekatan Metode Service Oriented Architecture dan Model View Controller pada Pusat Penelitian Perkembangan Iptek Lembaga Ilmu Pengetahuan Indonesia*, Skripsi tidak dipublikasikan, Universitas Indonesia, 2022.
- [4] I. Fauzi dan I. H. Ikasari, "Rancang Bangun Penerapan Teknologi Aplikasi Payment Gateway pada Sistem Pembayaran Berbasis Web (Studi Kasus: Toko Bandar Aki)," *Jurnal Informatika MULTI*, vol. 1, no. 3, 2023.
- [5] A. S. Nugraha, Sapri, Arius, S. Kurniawansyah, and A. S. J. Komitek, "Building an Online-Based Point of Sale Application at the Anton Store (Membangun Aplikasi Point of Sale Berbasis Online di Toko Anton)," *JURNAL KOMITEK*, vol. 1, no. 2, 2021. [Online]. Available: <http://andika.vadapp.my.id>
- [6] D. Abner, J. Gerung, and D. Redaksi, "Perancangan Sistem Informasi Point of Sale Berbasis Website pada Toko Arpan Electric," *Informasi Artikel*, 2022.
- [7] D. C. P. Yuda, A. S. Y. Irawan, dan E. H. Nurkifli, "Rancang Bangun Sistem Point of Sales Berbasis Web Menggunakan Framework Laravel pada Percetakan Radjawali Digital Printing," *JITET*, vol. 12, no. 3, pp. 2434–2443, 2024.
- [8] W. I. Rahayu, J. M. Bintang, dan D. A. Pramana, "Implementasi Framework Laravel pada

- Perancangan Aplikasi Sistem Pendaftaran Programming Course Roblox," *Jurnal Teknik Informatika*, vol. 15, no. 1, pp. 18–24, Jan. 2023.
- [9] Nasution dan L. Iswari, "Penerapan React JS Pada Pengembangan Front-End Aplikasi Startup Ubaform," *Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi (SENTIKA)*, Universitas Islam Indonesia, Yogyakarta, 2021.
- [10] Y. D. Wijaya dan M. W. Astuti, "Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan PT INKA (Persero) Berbasis Equivalence Partitions," *Jurnal Digital Teknologi Informasi*, vol. 4, 2021.
- [11] N. E. Alfia dan B. Waseso, "Perancangan Aplikasi Retensi Data Pada Database MySQL (Studi Kasus: PT. Telkomsigma)," *Jurnal Sistem Informasi dan E-Bisnis (JUSIBI)*, vol. 2, no. 3, pp. 364–374, Mar. 2020.
- [12] T. Q. Fadillah, T. Suratno, dan Mauladi, "Rancang Bangun Sistem Informasi Administrasi Tahanan dan Barang Bukti Menggunakan Model Prototype pada Kepolisian Daerah Jambi," *Jurnal Sains dan Sistem Informasi (JUSS)*, vol. 2, no. 1, pp. 36–44, Agustus 2019.
- [13] M. dan A. R. Fadhli, "Penggunaan JavaScript dalam Pengembangan Website Responsif dan Interaktif untuk UMKM," pp. 56–65, 2022.
- [14] E. Febriyanto, U. Rahardja, N. Alnabawi, D. Stmik, R. Jurusan, M. S. Informasi, S. Informasi, S. Raharja, dan J. Jenderal, "Penerapan Midtrans sebagai Sistem Verifikasi Pembayaran pada Website iPanda," *Jurnal Informatika UPGRIS*, vol. 4, no. 2, 2021.
- [15] C. Perdana, Maharani, dan M. A. Wijaya, "Implementasi Framework Bootstrap 5 Pada Perancangan Front-End Website MC BRO di PT X," *Jurnal Sistem Informasi Galuh (JSIG)*, vol. 2, no. 1, pp. 30–43, Jan. 2024.