

LAMPIRAN

Lampiran 1 Surat Kesiediaan Pembimbing

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Muhammad Affan Tamami
NIM : 20090151
Program Studi : D IV Teknik Informatika

Pihak Kedua

Nama : Dega Surono Wibowo, S.T., M.Kom.
Status : Dosen Tetap
NIDN : 0607108202
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata / III/C

Pada hari ini Jum'at tanggal 15 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan melakukan bimbingan 1x dalam seminggu atau setidaknya 3x bimbingan dalam 1 bulan (dengan progress). Apabila saya tidak memenuhi syarat tersebut, maka saya tidak berhak meminta surat rekomendasi mengikuti sidang skripsi. Dan saya berjanji akan memenuhi persyaratan tersebut dan menyelesaikan penelitian (produk dan laporan) sesuai jadwal penelitian (tepat waktu).

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 15 Maret 2024

Pihak Pertama



Muhammad Affan Tamami

Pihak Kedua



Dega Surono Wibowo, S.T., M.Kom.
NIPY. 06.014.183

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.
NIPY. 09.015.225

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini :

Pihak Pertama

Nama : Muhammad Affan Tamami
NIM : 20090151
Program Studi : D IV Teknik Informatika

Pihak Kedua

Nama : Hepatika Zidny Ilmadina, S.Pd., M.Kom.
Status : Dosen
NIDN : 0618119101
Jabatan Fungsional : Asisten Ahli
Pangkat/Golongan : Penata Muda Tingkat I/ III/B

Pada hari ini Jum'at tanggal 15 Maret 2024 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 8 kali kepada Pihak Kedua. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.

Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 15 Maret 2024

Pihak Pertama



Muhammad Affan Tamami

Pihak Kedua



Hepatika Zidny Ilmadina, S.Pd., M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom.
NIPY. 09.015.225

Lampiran 2 Surat Pernyataan Pengajuan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. N a m a : Muhammad Affan Tamami
Kewarganegaraan : Indonesia
Alamat : Jl. Kates Raya 11, RT 09/ RW 02, Desa Tembok Banjaran, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah, 52194.
2. N a m a : Dega Surono Wibowo, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Perum Sapphire Regency Blok H No.1, RT.004/RW.001, Kelurahan Pulosari Kecamatan Brebes, Kabupaten Brebes, Provinsi Jawa Tengah, 52213.
3. N a m a : Hepatika Zidny Ilmadina, S.Pd.,M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jalan Kenanga Gang 1 No. 9, Kelurahan Mangkukusuman, Kecamatan Tegal Timur, Kota Tegal, Provinsi Jawa Tengah, 52123

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:
Berupa : Program Komputer
Berjudul : Rancang Bangun Sistem Informasi Manajemen Konveksi Berbasis Web pada viary store

Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);

- Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
- Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam Pasal 39;
- Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud dalam Pasal 41 dan 42;
- Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
- Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum, pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang kami mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.

4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas kami langgar, maka kami bersedia secara sukarela bahwa:
- permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau
 - Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
 - Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam berperkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, 15 Januari 2025



Muhammad Affan Tamami
Pemegang Hak Cipta*

Dega Surono Wibowo, S.T., M.Kom.
Pemegang Hak Cipta*

Hepatika Zidny Ilmadina, S.Pd., M.Kom.
Pemegang Hak Cipta*

* Semua pemegang hak cipta agar menandatangani diatas materai

Lampiran 3 Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini:

1. Nama : Muhammad Affan Tamami
Kewarganegaraan : Indonesia
Alamat : Jl. Kates Raya 11, RT 09/ RW 02, Desa Tembok Banjaran, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah, 52194
2. Nama : Dega Surono Wibowo, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Perum Sapphire Regency Blok H No.1, RT.004/RW.001, Kelurahan Pulosari Kecamatan Brebes, Kabupaten Brebes, Provinsi Jawa Tengah, 52213
3. Nama : Hepatika Zidny Ilmadina, S.Pd., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jalan Kenanga Gang 1 No. 9, Kelurahan Mangkukusuman, Kecamatan Tegal Timur, Kota Tegal, Provinsi Jawa Tengah, 52123

Adalah Pihak I selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada:

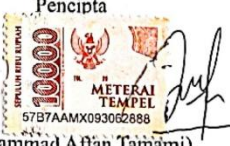
Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah Pihak II selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Rancang Bangun Sistem Informasi Manajemen Konveksi Berbasis Website(studi kasus viary store)". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikian Surat pernyataan ini kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Pemegang Hak Cipta
Kepala P3M

(Muhammad Fikri Hidayattullah, S.T., M.Kom.)

Tegal, 15 Januari 2025
Pencipta

(Muhammad Affan Tamami)


(Dega Surono Wibowo, S.T., M.Kom)


(Hepatika Zidny Ilmadina, S.Pd., M.Kom.)

Lampiran 4 Syarat Pengajuan HKI

MANUAL BOOK

RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
DAN PEMESANAN PADA KONVEKSI
BERBASIS WEBSITE

Oleh :

Muhammad Affan Tamami

Dega Surono Wibowo, S.T., M.Kom.

Hepatika Zidny Ilmadina, S.Pd., M.Kom

1. PENDAHULUAN

1.1. Tujuan Pembuatan Dokumen

Dokumen *user manual* ini dibuat untuk memberikan penjelasan dan Gambaran atau sebagai panduan kepada Pengguna mengenai penggunaan aplikasi Manajemen Konveksi Berbasis Website.

1.2. Deskripsi Umum Aplikasi

Aplikasi ini dirancang untuk mendukung operasional toko konveksi dalam mengelola data produk, pesanan, dan pengguna secara efisien. Sistem ini menggunakan framework Laravel untuk membangun aplikasi berbasis web yang responsif, aman, dan mudah digunakan. Aplikasi ini menyediakan fitur seperti manajemen data produk, pemrosesan pesanan, autentikasi dan otorisasi pengguna yang dibagi menjadi beberapa peran (owner, admin, dan user), yang masing-masing memiliki hak akses yang berbeda.

1.3. Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan Aplikasi Manajemen Konveksi Berbasis Website.

1. Bagian 1

Bagian ini berisi informasi umum yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem, dan deskripsi dokumen.

2. Bagian 2

Bagian ini berisi perangkat yang dibutuhkan untuk penggunaan aplikasi meliputi perangkat lunak.

3. Bagian 3

Bagian ini berisi panduan atau cara penggunaan.

2. PERANGKAT YANG DIBUTUHKAN

2.1. Perangkat Lunak

Untuk menjalankan aplikasi ini, berikut beberapa perangkat lunak yang dibutuhkan yaitu:

1. Aplikasi ini harus dijalankan pada platform web untuk memungkinkan user seperti melihat produk, memesan produk melalui perangkat yang terhubung dengan internet.

2. Untuk menjalankan aplikasi dengan lancar, diperlukan koneksi internet yang stabil.
3. Penyimpanan Data akan disimpan dengan aman diserver

3. MENU DAN CARA PENGGUNAAN

3.1. Struktur Menu

Ada 3 pengguna dalam aplikasi manajemen koveksi yaitu *admin,owner* dan *user*. Berikut Stuktur menu yang ada pada aplikasi manajemen konveksi adalah sebagai berikut:

User

1. Register
Menu ini digunakan untuk mendaftarkan akun baru. Pengguna harus mengisi form yang sudah disediakan seperti nama lengkap, username, email, nomor handphone dan *password* untuk membuat akun.
2. Verifikasi email
Halaman ini berfungsi untuk mengirim verifkasi email agar akun *user* dapat diverifikasi
3. Login
Menu ini digunakan untuk masuk ke aplikasi dengan akun yang sudah terdaftar. Pengguna harus memasukkan email dan kata sandi yang telah didaftarkan.
4. Lupa Password
Menu ini berfungsi bagi *user* yang lupa *password* dari akun mereka. Disini *user* akan disuruh memasukan email mereka yang nantinya akan ada email yang masuk ke email *user* yang berisi link untuk me reset *password* sang *user*.
5. Home
Menu ini akan menampilkan informasi mengenai aplikasi yang dapat anda lihat seperti informasi produk, informasi layanan
6. Kategori
Menu ini akan menampilkan kategori dari produk
7. Detail
Menu ini akan menampilkan informasi detail dari sebuah produk seperti, nama produk, deskripsi produk, harga produk, ukuran produk, dan jumlah produk

8. History

Halaman ini akan menampilkan halaman history, setelah user melakukan order maka user dapat melihat history

9. Layanan

Menampilkan informasi layanan yang disediakan pada toko konveksi

10. Profile User

Halaman ini akan menampilkan informasi pribadi pengguna yang telah terdaftar di aplikasi.

11. Logout

Setelah melakukan order, melihat riwayat order, Pengguna dapat logout dari akun mereka

3.2. Cara Penggunaan

User

1. Halaman Login

Dihalaman login ini, anda dapat masuk kedalam aplikasi dengan memasukan *email* dan *password* yang sudah terdaftar dan terverifikasi. Tampilan halaman login dapat dilihat pada Gambar 1. Berikut langkah-langkah nya:

- 1) Masukkan *email* dan *password*. Pastikan sudah terdaftar dan terverifikasi.
- 2) Klik button Login.
- 3) Aplikasi akan mengarahkan ke halaman *home*.

Catatan :

- 1) Jika anda belum memiliki akun, maka anda harus membuatnya terlebih dahulu dengan menekan link “Register”.
- 2) Jika anda lupa *password* akun anda, maka anda bisa menekan link “Lupa Password”

Gambar 1. Tampilan login dan register

2. Halaman Lupa Password

Dihalaman ini, *user* mengisi email yang terdaftar untuk mereset *password* akunnya yang lupa. Tampilan tersebut dapat dilihat pada Gambar 2. Berikut langkah-langkah nya :

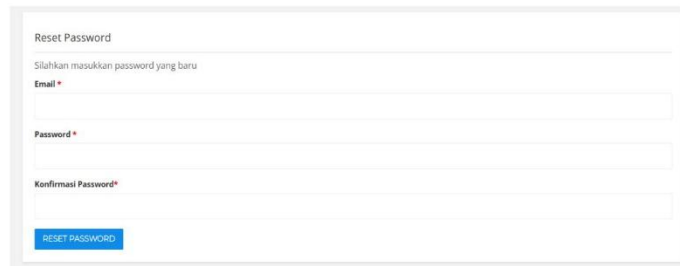
- 1) Terdapat inputan untuk *user* mengisinya dengan akun email yang lupa *password* nya.
- 2) *User* akan mendapatkan pesan email yang berisi link untuk mereset *password* *password* *user* yang lama

Gambar 2. Tampilan untuk Lupa *password*

3. Halaman Reset Password

Dihalaman ini, *user* mengisi *password* baru untuk mengubah *password* akunnya yang lupa. Tampilan tersebut dapat dilihat pada Gambar 3. Berikut langkah-langkah nya :

- 1) Terdapat inputan untuk *user* mengisi email dan *password baru* serta konfirmasi *password*. Setelah itu Klik tombol “Reset Password”.
- 2) *User* akan dialihkan ke halaman login.



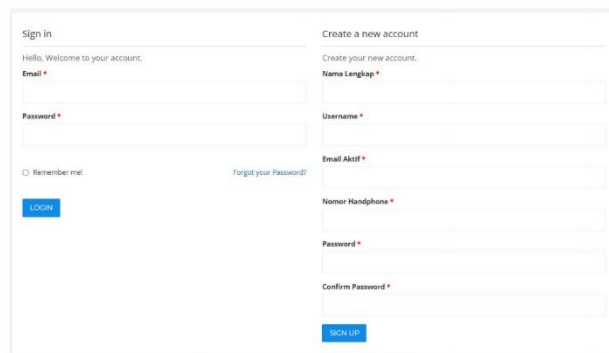
Gambar 3. Tampilan Reset *Password*

4. Halaman Registrasi

Pada halaman ini, anda akan diarahkan untuk membuat akun baru agar dapat masuk kedalam website. Tampilan tersebut dapat dilihat pada Gambar

4. Berikut langkah-langkah nya:

- 1) Terdapat empat formular untuk proses pendaftaran, masukkan informasi yang diperlukan seperti Nama, Username, Email, No handphone, dan Password. Klik button “Signup”
- 2) Setelah registrasi maka akan di arahkan ke halaman verifikasi email
- 3) Setelah email terverifikasi akan diarahkan ke halaman home

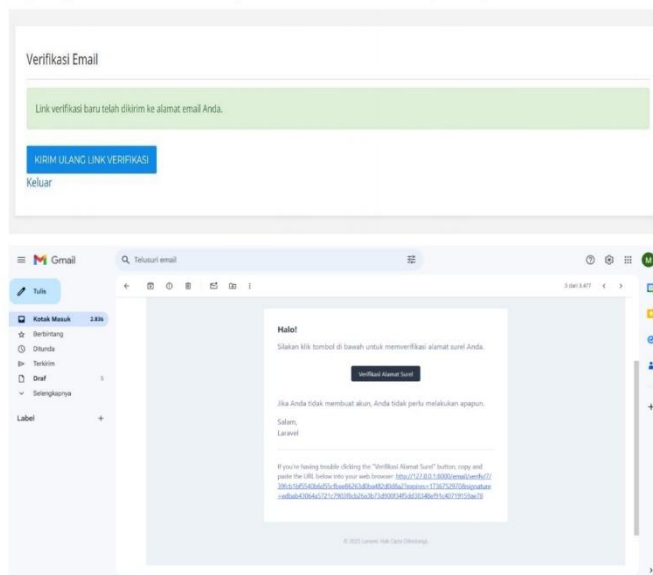


Gambar 4. Tampilan Register

5. Halaman Verifikasi Email

Halaman ini digunakan untuk memastikan bahwa alamat email yang Anda daftarkan valid dan dapat diakses oleh anda. Tampilan tersebut dapat dilihat pada Gambar 5. Berikut langkah-langkah nya:

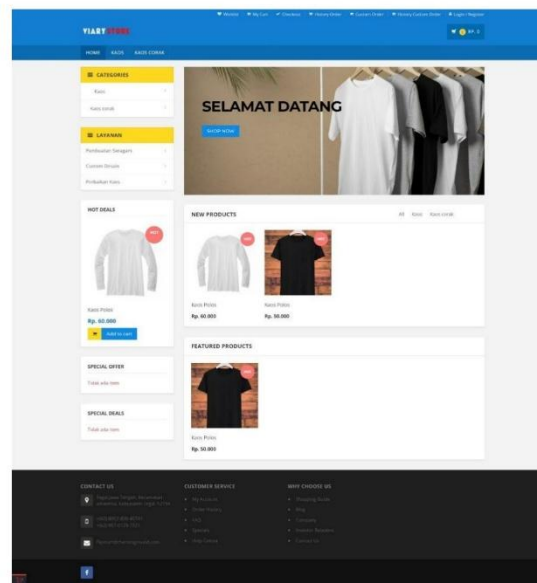
- 1) Periksa kotak masuk Gmail anda dan klik tombol verifikasi pada pesan.
- 2) Klik tombol yang bertuliskan “*Kirim ulang link verifikasi*”, tunggu hingga muncul pesan berwarna hijau, pesan verifikasi akan masuk ke kotak masuk Gmail anda.
- 3) Aplikasi akan menampilkan halaman utama (home).



Gambar 5 Tampilan Halaman *Verifikasi* Email

6. Halaman *Home*

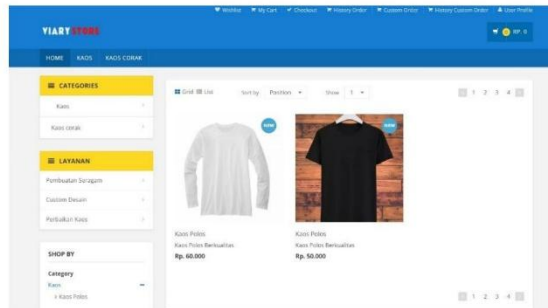
Halaman *home* merupakan halaman yang dilihat setelah anda berhasil login ke aplikasi. Dalam halaman *home*, akan menampilkan beberapa produk, informasi layanan, dan kategori produk. Tampilan tersebut dapat dilihat pada Gambar 6. dibawah ini.



Gambar 6 Tampilan *Home User*

7. Halaman *Kategori*

Halaman *Kategori* akan menampilkan beberapa produk dengan kategori yang sama. Tampilan tersebut dapat dilihat pada Gambar 7.

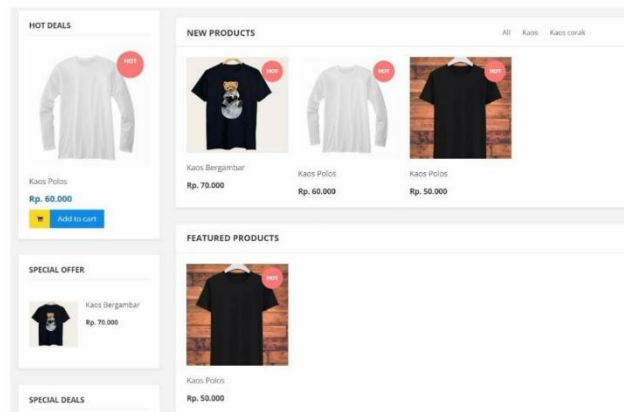


Gambar 7. Tampilan Halaman Kategori

8. Halaman Produk

Halaman *Produk* akan menampilkan beberapa produk yang dimana Pengguna dapat memilih dari sebuah produk. Tampilan tersebut dapat dilihat pada Gambar 8. Berikut langkah-langkah yang harus dilakukan :

- 1) *User* memilih produk
- 2) Website akan mengarahkan *user* ke menu detail dari produk yang dipilih.

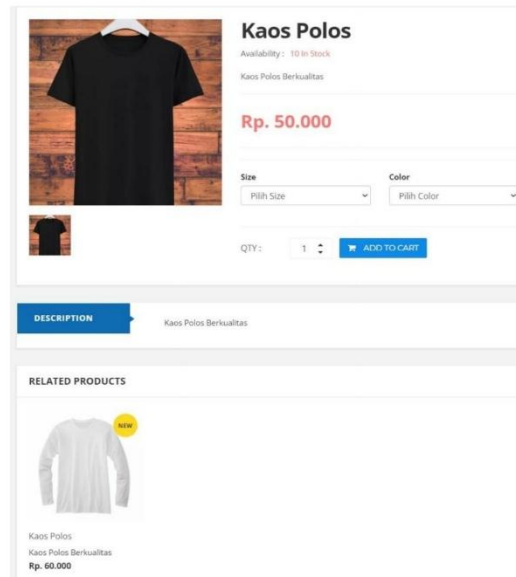


Gambar 8. Tampilan Produk

9. Halaman Detail

Halaman ini akan menampilkan detail terkait produk seperti nama produk, harga, stok, ukuran, warna, dan deskripsi. Berikut tampilan pada halaman ini dapat dilihat pada Gambar 9. Berikut langkah-langkah yang harus dilakukan :

- 1) Setelah *User* memilih produk maka akan tampil halaman dari detail produk tersebut
- 2) *User* dapat memilih ukuran, warna, serta jumlah produk yang akan di pesan.
- 3) Kemudian Klik button “Add To Cart”
- 4) Setelah user menekan button ”add to cart” Produk akan ditambahkan ke dalam keranjang

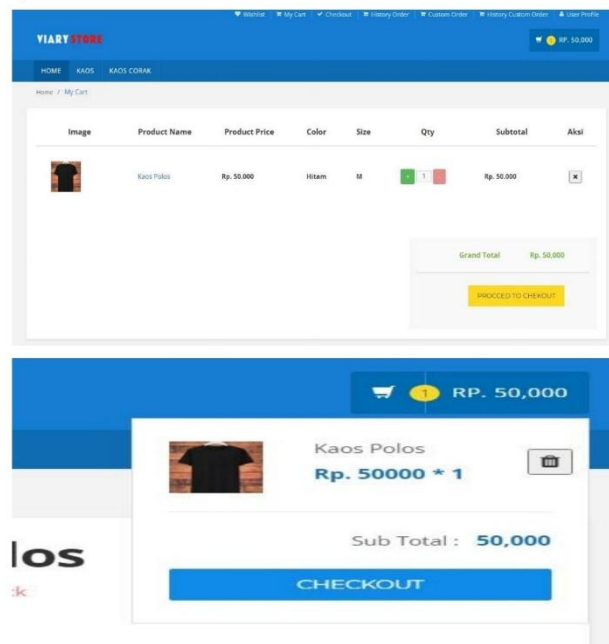


Gambar 9. Tampilan Detail Produk

10. Halaman Keranjang

Halaman ini akan menampilkan terkait produk seperti gambar produk, nama produk, harga, warna, ukuran, jumlah, dan subtotal. Berikut tampilan pada halaman ini dapat dilihat pada Gambar 10. Berikut langkah-langkah yang harus dilakukan :

- 1) Setelah *User* menambahkan produk kedalam keranjang maka akan tampil di halaman keranjang dari detail produk tersebut
- 2) Di halaman ini akan menampilkan seperti gambar produk, nama produk, harga, warna, ukuran, jumlah, subtotal, dan grand total
- 3) Kemudian *User* klik button “Procced To Chekout”
- 4) Jika user checkout dari menu mini cart maka user klik button ”checkout”
- 5) Website akan mengarahkan *user* ke halaman checkout dari produk yang dipilih

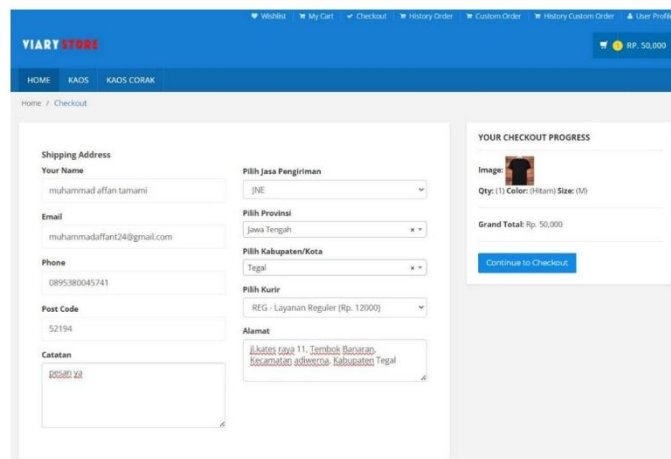


Gambar 10. Tampilan halaman keranjang

11. Halaman Checkout Produk

Halaman Checkout *Produk* akan menampilkan beberapa informasi data yang dimana user harus mengisi informasi dengan lengkap untuk memproses pesanan. Tampilan tersebut dapat dilihat pada Gambar 11. Berikut langkah-langkah yang harus dilakukan :

- 1) *User* mengisi informasi yang dibutuhkan seperti nama lengkap, email, phone, kode pos, catatan, serta alamat lengkap, kemudian *User* dapat memilih untuk jasa pengirimannya.
- 2) Setelah semua informasi data sudah lengkap kemudian klik button “Continue To Checkout”.
- 3) Website akan mengarahkan user ke halaman detail informasi pesanan yang akan dibuat.

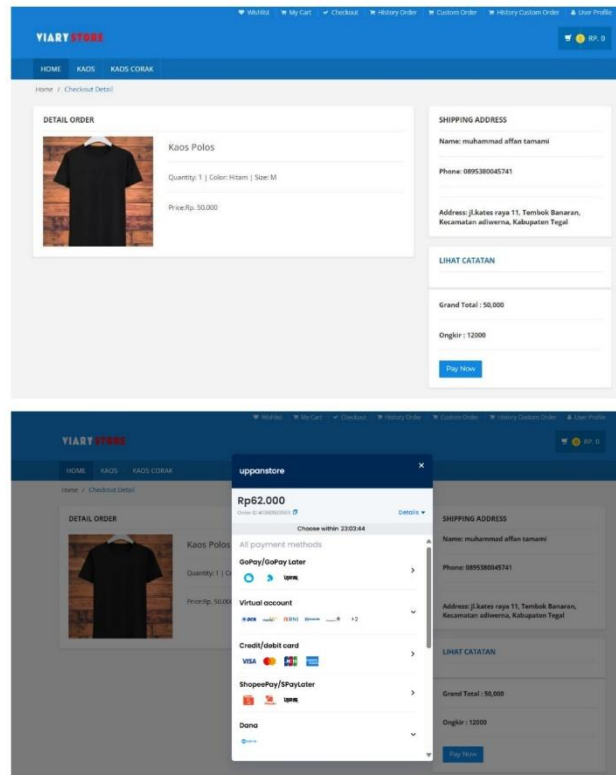


Gambar 11. Tampilan *user* checkout

12. Halaman Detail Checkout

Halaman ini akan menampilkan detail order produk seperti nama produk, harga, jumlah, ukuran, warna, dan informasi nama penerima, nomor handphone serta Alamat . Berikut tampilan pada halaman ini dapat dilihat pada Gambar 12. Berikut langkah-langkah yang harus dilakukan :

- 1) Setelah semuanya sesuai kemudian klik button “Pay now”.
- 2) Website akan menampilkan Pop up Pembayaran midtrans, menampilkan informasi jumlah harga yang harus dibayar, kemudian user dapat membayar agar pesanan dapat di proses.
- 3) Setelah pembayaran sukses maka akan diarahkan ke halaman history



Gambar 12. Tampilan Detail Checkout

13. Halaman History

Setelah melakukan order maka *user* dapat melihat dhalaman history Yang berisi, tanggal, total, payment, invoice, dan order status, *user* juga dapat melihat invoice order. Dapat dilihat pada gambar 13. dibawah ini.

VIARY STORE

HOME KAOS KAOS CORAK

Home / History Order

Date	Total	Payment	Invoice	Order	Action
20 January 2025	62000	qris	INV159666884418421	Success	View Invoice

Name: muhammad affan tamami
Email: muhammadaffant24@gmail.com
Phone: 0895380045741
Address: jl.kates raya 11, Tembok Banaran, Kecamatan adiwerna, Kabupaten Tegal
Post Code: 52194

#INV159666884418421
Order Date: 20 January 2025
Delivery Date:
Payment Type: qris

Products

No	Product Name	Size	Color	Quantity	Unit Price	Total
1	Kaos Polos	M	Hitam	1	Rp. 50.000	Rp. 50.000

Subtotal: 50000
Ongkir: 12000
Total: 62000
Payment Status: Success

Thanks For Buying Product!

Authority Signature:

Gambar 13. Tampilan history dan invoice

14. Halaman Detail History

Halaman ini akan menampilkan detail lengkap dari history order seperti nama produk, harga, jumlah, ukuran, warna, dan informasi nama penerima, nomor handphone serta Alamat, dapat dilihat pada gambar 14.

Name	Phone	Alamat	Note
muhammad affan tamami	0895380045741	Jl. Kates raja 11, Tembok Baranar, Kecamatan adiwerna, Kabupaten Tegal	pesan ya

Catatan : pesan ya

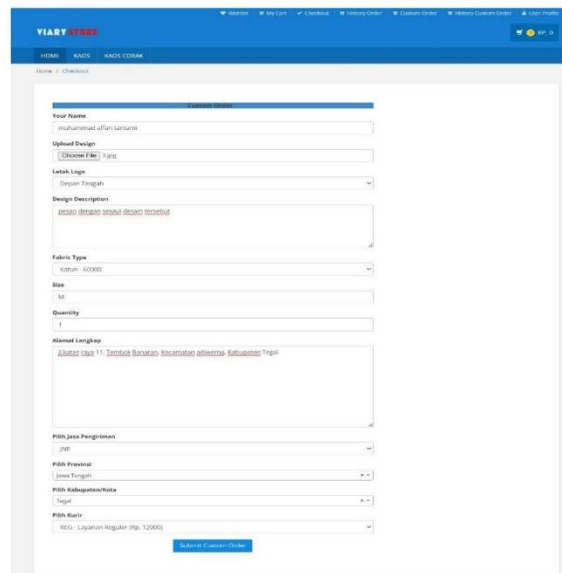
Image	Product Name	Color	Size	Quantity	Price	Subtotal
	Kaos Polo	Hitam	M	1	Rp. 50.000	Rp. 50.000

Gambar 14. Tampilan Detail History order

15. Halaman Custom Order

Halaman ini akan menampilkan beberapa informasi data yang dimana user harus mengisi data untuk memproses pesanan custom. Tampilan tersebut dapat dilihat pada Gambar 15. Berikut langkah-langkah yang harus dilakukan:

- 1) *User* mengisi informasi yang dibutuhkan dengan lengkap, kemudian *User* dapat memilih untuk jasa pengirimannya.
- 2) Setelah semua informasi data sudah lengkap kemudian klik button “Submit Custom Order”.
- 3) Website akan mengarahkan user ke halaman history dari custom order.



The screenshot displays the 'Custom Order' form on the VIARY CLOTHES website. The form is titled 'Custom Order' and includes the following fields and options:

- Your Name:** A text input field containing 'muhammad affan kurnia'.
- Upload Design:** A button labeled 'Choose File' and a file size indicator '1.3M'.
- Letak Logo:** A dropdown menu with 'Depan Tengah' selected.
- Design Description:** A text area containing 'desain dengan detail desain terdapat'.
- Fabric Type:** A dropdown menu with 'katun - 60/300' selected.
- Size:** A dropdown menu with 'M' selected.
- Quantity:** A text input field with '1' entered.
- Alamat Lengkap:** A text area with a placeholder 'alamat (jika T1. Tambahkan keterangan pengiriman kealamat Pengirim)'. Below the text area is a small table with one row: 'Alamat (jika T1. Tambahkan keterangan pengiriman kealamat Pengirim)'.
- Pilih Jasa Pengiriman:** A dropdown menu with 'JNE' selected.
- Pilih Provinsi:** A dropdown menu with 'Jawa Tengah' selected.
- Pilih Kabupaten/Kota:** A dropdown menu with 'Tegal' selected.
- Pilih Kurir:** A dropdown menu with 'REG - Lapangan Reguler (Rp. 12000)' selected.

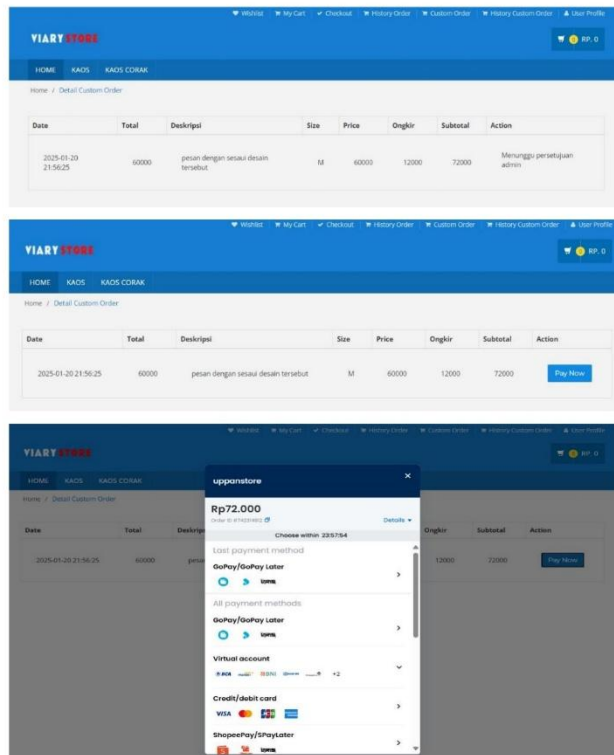
At the bottom of the form is a blue button labeled 'Submit Custom Order'.

Gambar 15. Tampilan Submit Custom Order

16. Halaman History Custom Order

Halaman ini akan menampilkan detail lengkap dari history custom order seperti tanggal, total harga, deskripsi, size, price, ongkir dan subtotal. dapat dilihat pada gambar 16. Berikut Langkah-langkah yang harus dilakukan :

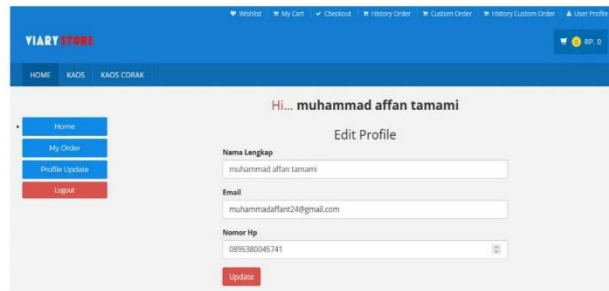
- 1) Setelah user melakukan pesanan custom order, setelah itu menunggu admin untuk mneyetujui pesanan custom order tersebut.
- 2) Apabila sudah disetujui maka user dapat melakukan pembayaran dengan cara Klik button “Pay Now”.
- 3) Setelah pembayaran sukses maka akan diarahkan ke halaman history custom order, user juga dapat melihat invoice dari custom order yang telah dibuat.



Gambar 16. Tampilan history Custom order dan Pembayaran

17. Halaman Profile User

Pada halaman ini menampilkan detail dari profil *User*. Terdapat nama *lengkap*, email dan nomor handphone. Klik button update jika ingin merubah data. Berikut tampilan halaman profile dapat dilihat pada Gambar 17.



The screenshot displays the 'User Profile' page of the VIARY 1992 application. The top navigation bar includes links for Wishlist, My Cart, Checkout, History Order, Custom Order, History Custom Order, and User Profile. The main header shows the user's name 'Hi... muhammad affan tamami' and a balance of Rp. 0. The left sidebar contains buttons for Home, My Order, Profile Update, and Logout. The 'Edit Profile' section on the right features three input fields: 'Nama Lengkap' (muhammad affan tamami), 'Email' (muhammadaffan214@gmail.com), and 'Nomor Hp' (0895380045741). An 'Update' button is located at the bottom of the form.

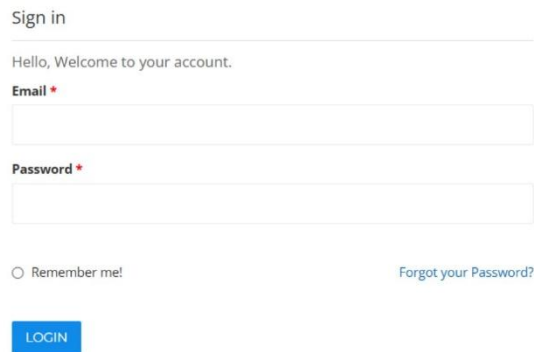
Gambar 17. Tampilan Profile *User*

Admin

1. Halaman Login

Dihalaman login ini, admin dapat masuk kedalam *dashboard* dengan memasukan *email* dan *password* yang sudah didaftarkan. Tampilan halaman login admin dapat dilihat pada Gambar 18. Berikut langkah-langkah nya :

- 1) Masukkan *email* dan *password admin*.
- 2) Klik button Login
- 3) Aplikasi akan mengarahkan ke halaman *dashboard*.



Sign in

Hello, Welcome to your account.

Email *

Password *

☐ Remember me! [Forgot your Password?](#)

LOGIN

Gambar 18. Tampilan Halaman Login admin

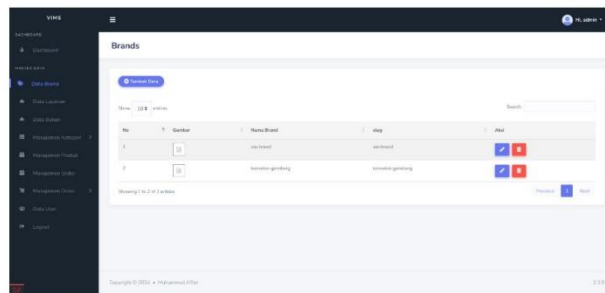
2. Halaman Dashboard

Setelah admin login Halaman ini merupakan halaman awal yang menampilkan total produk, total order sukses, total order pending dan total user, admin juga dapat melihat grafik transaksi bulanan. Tampilan gambar halaman ini dapat dilihat pada Gambar 19 berikut ini.



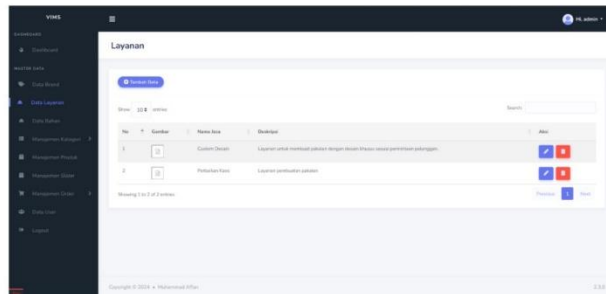
3. Halaman Data Brand

Pada halaman ini akan menampilkan semua data brand yang ditampilkan pada halaman cekout produk saat *user* akan memesan produk. Disini admin bisa mengubah, menambahkan, dan menghapus data yang diinginkan. Tampilan halaman ini bisa dilihat pada Gambar 20 berikut.



4. Halaman Layanan

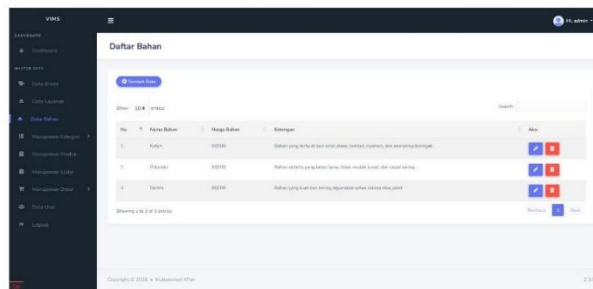
Pada halaman ini akan menampilkan data layanan pada website. Dihalaman ini juga, admin dapat menambahkan data baru, mengubah data, dan menghapusnya. Tampilan halaman ini bisa dilihat pada Gambar 21 dibawah ini



Gambar 21. Tampilan Halaman Layanan Admin

5. Halaman Data Bahan

Pada halaman ini akan menampilkan data bahan pada website yang akan digunakan untuk *user* pada form bahan custom order. Di halaman ini juga, admin dapat menambahkan data baru, mengubah data, dan menghapusnya. Tampilan halaman ini bisa dilihat pada Gambar 22 dibawah ini



Gambar 22. Tampilan Halaman Data bahan Admin

6. Halaman Data Kategori

Pada halaman ini akan menampilkan data semua kategori pada website. Di halaman ini juga, admin dapat menambahkan data kategori baru, mengubah data kategori, melihat data kategori, dan menghapusnya. Tampilan halaman ini bisa dilihat pada Gambar 23 dibawah ini. Berikut langkah-langkah dalam menambahkan kategori baru :

▪ Kategori

- 1) Pada menu kategori Tekan tombol “Tambah data”
- 2) Isi data kategori seperti nama kategori, icon kategori.
- 3) Tekan tombol “Simpan” untuk menyimpannya.

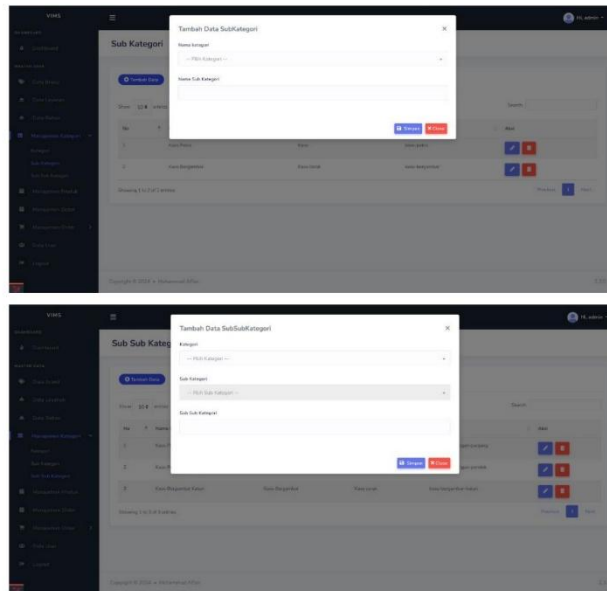
▪ SubKategori

- 1) Pada menu kategori tekan tombol “Tambah data”
- 2) Pilih nama kategori yang sudah ditambahkan sebelumnya di menu kategori.
- 3) Kemudian isi nama subkategori
- 4) Tekan tombol “Simpan” untuk menyimpannya.

▪ SubSubKategori

- 1) Pada menu kategori tekan tombol “Tambah data”
- 2) Pilih nama kategori yang sudah ditambahkan sebelumnya di menu kategori.
- 3) Pilih nama sub kategori yang sudah ditambahkan sebelumnya di menu SubKategori.
- 4) Kemudian isi nama subsubkategori
- 5) Tekan tombol “Simpan” untuk menyimpannya.



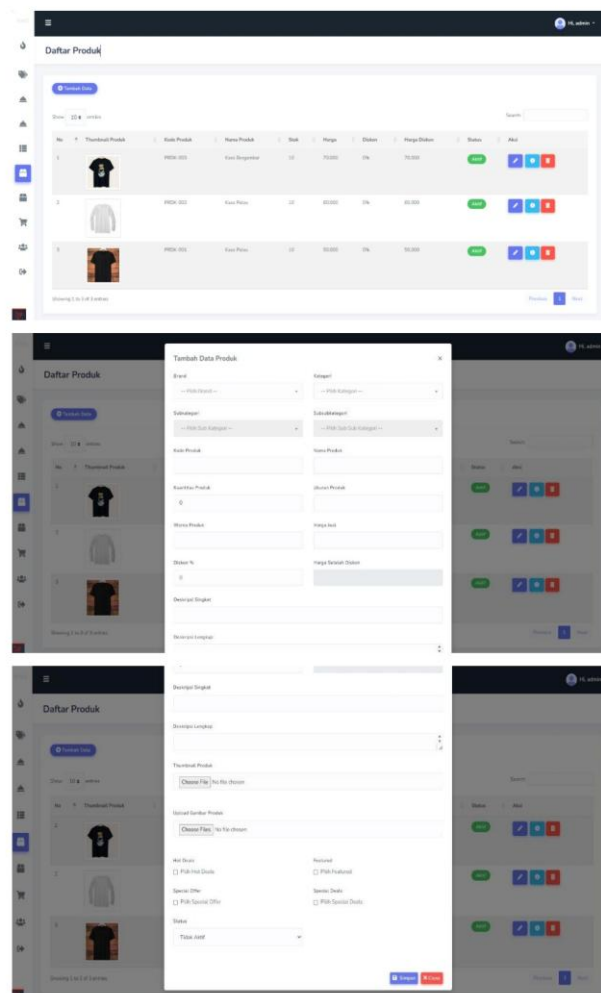


Gambar 23. Tampilan Halaman Kategori Admin

7. Halaman Produk

Pada halaman ini akan menampilkan data produk pada website. Di halaman ini juga, admin dapat menambahkan data produk baru, mengubah data produk, melihat data produk, dan menghapusnya. Tampilan halaman ini bisa dilihat pada Gambar 24 dibawah ini. Berikut langkah-langkah dalam menambahkan produk baru :

- 1) Tekan tombol “Tambah Data”
- 2) Isi data produk seperti brand, kategori, subkategori, subsubkategori, kode produk, nama produk. dan lain-lain.
- 3) Tekan tombol “Simpan” untuk menyimpannya.

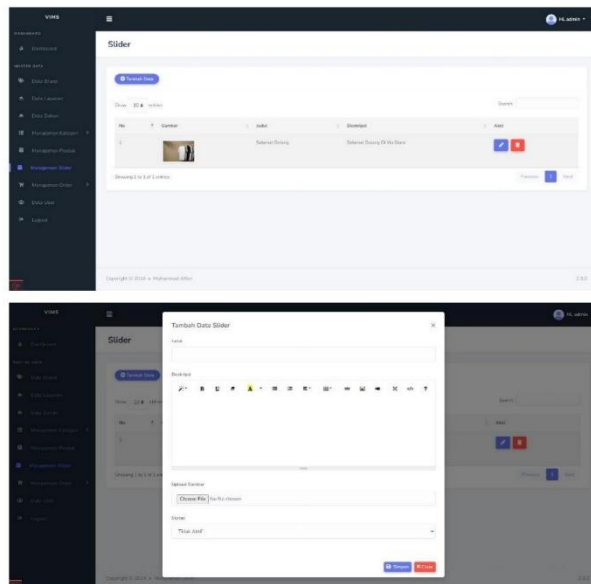


Gambar 24. Tampilan Halaman produk admin

8. Halaman Slider

Pada halaman ini akan menampilkan data slide yang akan ditampilkan pada halaman home user pada website. Di halaman ini juga, admin dapat menambahkan data slider baru, mengubah data slider, dan menghapusnya. Tampilan halaman ini bisa dilihat pada Gambar 25 dibawah ini. Berikut langkah-langkah dalam menambahkan slider baru :

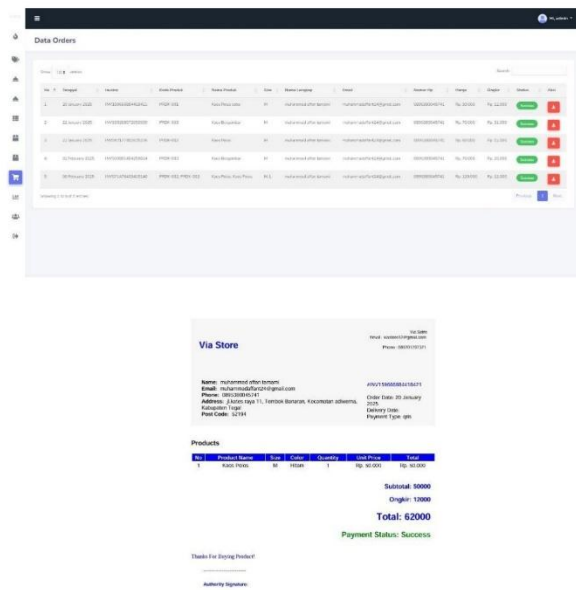
- 1) Tekan tombol “Tambah Data”.
- 2) Isi data seperti judul, deskripsi, upload gambar, dan dropdown status aktif dan tidak aktif yang dimana agar dapat ditampilkan pada halaman home user.
- 3) Tekan tombol “Simpan” untuk menyimpannya



Gambar 25. Tampilan Halaman Slider Admin

9. Halaman Order

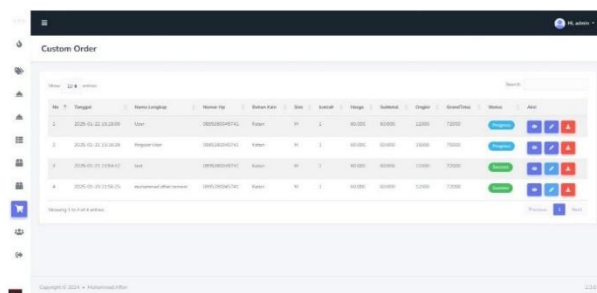
Pada halaman ini akan menampilkan data order dan transaksi yang telah dilakukan user pada halaman cekhout produk pada website. Di halaman ini admin dapat melihat data order dari user serta melihat status pembayaran. admin dapat melihat invoice dari order. Tampilan halaman ini bisa dilihat pada Gambar 26 dibawah ini.



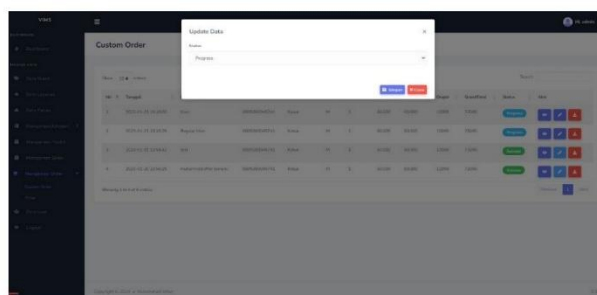
Gambar 26. Tampilan Halaman Order Admin dan invoice

10. Halaman Custom Order

Pada halaman ini akan menampilkan data Custom Order dan transaksi yang telah dilakukan user pada halaman Custom Order untuk memesan produk pada website. Di halaman ini admin dapat melihat data Custom order dari user, memberikan status progres pada pesanan, serta dapat melihat status pembayaran atau transaksi jika user telah melakukan pembayaran maka status pembayaran secara otomatis menjadi sukses. admin juga bisa melihat invoice dari pesanan tersebut berupa pdf. Tampilan halaman ini bisa dilihat pada Gambar 27 dan gambar 28 dibawah ini



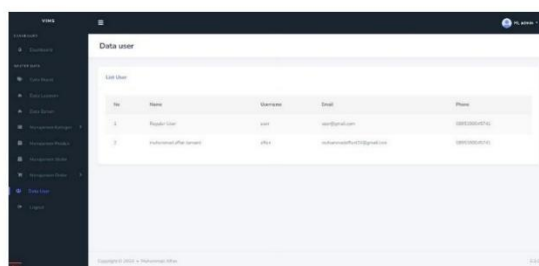
Gambar 27. Tampilan Halaman Custom Order Admin



Gambar 28. Tampilan Halaman Custom Order Memberikan Status

11. Halaman Data *User Customer*

Pada halaman ini akan menampilkan data dari user yang telah melakukan registrasi melalui website, Tampilan halaman ini bisa dilihat pada Gambar 29 dibawah ini.



Gambar 29. Tampilan Halaman data user customer

Owner

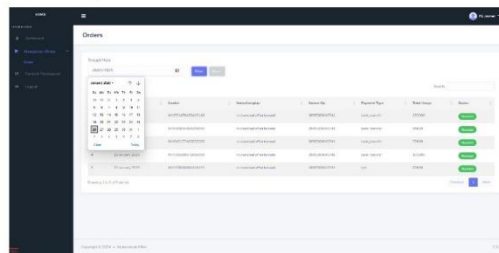
1. Halaman Login

Dihalaman login ini, owner dapat masuk kedalam *dashboard* dengan memasukan *email* dan *password* yang sudah didaftarkan, Tampilan halaman login admin dapat dilihat pada Gambar 30. Berikut langkah-langkahnya :

- 1) Masukkan *email* dan *password owner*.
- 2) Klik button Login
- 3) Aplikasi akan mengarahkan ke halaman *dashboard*

2. Halaman data Order

Pada halaman ini akan menampilkan data order dan hanya transaksi yang telah sukses, selain itu owner dapat memfilter data berdasarkan tanggal, bulan dan tahun, jika data ada maka akan ditampilkan, jika data tidak ada, maka tidak akan ditampilkan.

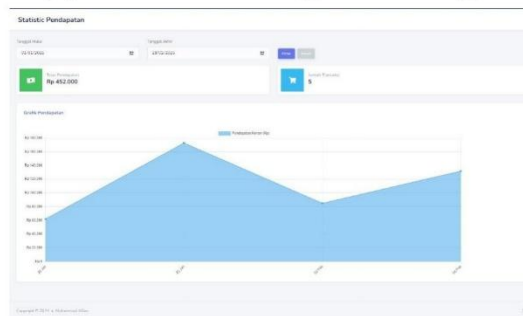


No	Nama	Alamat	No. HP	Email	Status
1	John Doe	123 Main St	08123456789	john.doe@example.com	Sukses
2	Jane Smith	456 Elm St	08987654321	jane.smith@example.com	Sukses
3	Bob Johnson	789 Oak St	08765432109	bob.johnson@example.com	Sukses
4	Alice Brown	101 Pine St	08543210987	alice.brown@example.com	Sukses

Gambar 30. Tampilan Halaman Owner Data order

3. Halaman statistik pendapatan

Pilih Statistik pendapatan di menu sidebar, didalam menu statistik pendapatan menampilkan statistik pendapatan yang dicapai, data yang di tampilkan yaitu total pendapatan dan banyaknya transaksi, sekaligus data pendapatan juga akan ditampilkan dengan grafik chart. Data yang akan ditampilkan juga bisa diatur berdasarkan tanggal mulai dan tanggal akhir.



Gambar 31. Tampilan Halaman Owner Statistik Pendapatan

DOKUMEN TEKNIS

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
DAN PEMESANAN PADA KONVEKSI
BERBASIS WEBSITE**

Oleh :

Muhammad Affan Tamami

Dega Surono Wibowo, S.T., M.Kom.

Hepatika Zidny Ilmadina, S.Pd., M.Kom.

Profile

Aplikasi manajemen konveksi ini dirancang untuk memfasilitasi penjualan produk konveksi seperti baju atau kaos secara online dengan tampilan yang user-friendly dan fungsionalitas yang lengkap. Sistem ini memungkinkan Pengguna untuk menelusuri berbagai produk seperti baju, melakukan pembelian, serta mendapatkan informasi detail tentang setiap produk, termasuk harga, ukuran, deskripsi, gambar produk. Aplikasi ini juga mendukung fitur keranjang belanja, pembayaran yang aman melalui integrasi dengan payment gateway Midtrans, serta pengelolaan transaksi secara otomatis. Disisi admin, terdapat dashboard yang memudahkan dalam pengelolaan produk, kategori, dan transaksi, termasuk penambahan, pengeditan, dan penghapusan data, selain itu dashboard admin dilengkapi dengan fitur untuk memantau manajemen stok produk. Rancang bangun aplikasi ini juga memperhatikan aspek keamanan dan kemudahan pengguna sehingga diharapkan dapat memberikan pengalaman berbelanja yang nyaman dan aman bagi customer serta efisiensi dalam penegelolaan toko bagi administrator.

Manfaat

1. customer dapat dengan mudan mencari dan membeli baju atau kaos dari mana saja dan kapan saja tanpa harus mengunjungi toko fisik, meningkatkan kenyamanan dan aksesibilitas bagi pengguna.
2. Aplikasi ini menampilkan berbagai pilihan baju dengan informasi lengkap, termasuk harga, ukuran, deskripsi dan gambar produk, yang membantu customer membuat keputusan pembelian yang lebih baik.
3. Dengan intergasi payment gateway Midtrans, aplikasi ini menawarkan berbagai metode pembayaran yang aman dan nyaman, meningkatkan kepercayaan customer dalam bertransaksi.
4. Melalui dashboard admin, pengelola toko dapat mudah mengelola inventaris, memantau transaksi, menganalisis penjualan untuk pengambilan keputusan yang lebih tepat.
5. Aplikasi ini mengotomatisasi banyak proses penting, seperti pembaruan status transaksi dan manajemen stok, yang mengurangi kesalahan manual, meningkatkan efisiensi oeprasional.

Spesifikasi Teknis

Spesifikasi Teknis, meliputi:

1. Modul Pengguna
2. *Source Code*

Berikut uraian spesifikasi untuk pembangunan aplikasi :

1. Visual Studi Code
2. Xampp
3. Web Browser (Chrome, Edge)
4. Framework Laravel
5. Windows 11

Berikut uraian spesifikasi modul:

1. Modul Untuk Pengguna
2. *Source Code*

1) User Interface.

1. Tampilan Login

Pada tampilan halaman login menggunakan form yang menggunakan method 'POST' dan mengirim data ke route 'login'. Memiliki 2 inputan yaitu email yang bertipe 'email' dan password yang bertipe 'password'. Pada login juga ada link 'Lupa Password' yang berfungsi bagi *user* yang lupa password akun mereka. Disana juga terdapat form 'Register' yang diperuntukan bagi *user* yang ingin membuat akun. Kedua inputan *user* yang dimasukan yaitu email dan password akan di olah di bagian *Controller* untuk di cek dan disamakan dengan data yang ada di database apakah ada data *user* yang sesuai data inputan. Tampilan *Source Code* halaman login dapat dilihat pada Gambar 1.1.

```

1  <section id="frontend_main_master">
2  <div class="container">
3    <div class="breadcrumb">
4      <div class="breadcrumb-inner">
5        <div class="list-inline list-unstyled">
6          <div class="text">[< a href="/" > Home</a> />]
7          <div class="active">login</div>
8        </div>
9      </div>
10    </div>
11    <div class="body-content">
12      <div class="sign-in-page">
13        <div class="row">
14          <div class="col-md-4 col-sm-6 sign-in">
15            <div class="sign-in-box">
16              <div class="text">Hello, Welcome to your account.</div>
17              <div class="form-group">
18                <input type="text" name="email" class="form-control unicode-form-control text input">
19                <div class="invalid-feedback" style="display: none;">

```

Gambar 1. 1 Source Code Dan Tampilan Login

2. Tampilan Register

Tampilan registrasi ini menampilkan formulir yang terdiri dari empat input yaitu nama, kota, provinsi, email, dan password. Masing-masing dikelola oleh input yang bertipe 'text' untuk kolom nama lengkap, ,username, email, dan password, serta divalidasi di bagian *Controller* menggunakan 'validate' untuk menyimpan dan memvalidasi inputan yang diisikan ke dalam form register diisi dengan benar. Halaman menyediakan tombol *Register* yang terhubung ke fungsi 'store' pada *Controller* yang menerima inputan *user* dan mengirim data ke server. *Source Code* tersebut dapat dilihat pada Gambar 1.2.

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_register);

    // Initialize variables
    String name, city, province, email, password;

    // Find views
    TextView nameView = findViewById(R.id.name);
    TextView cityView = findViewById(R.id.city);
    TextView provinceView = findViewById(R.id.province);
    TextView emailView = findViewById(R.id.email);
    TextView passwordView = findViewById(R.id.password);
    TextView confirmPasswordView = findViewById(R.id.confirm_password);
    TextView signUpView = findViewById(R.id.sign_up);

    // Set text colors
    nameView.setTextColor(Color.BLACK);
    cityView.setTextColor(Color.BLACK);
    provinceView.setTextColor(Color.BLACK);
    emailView.setTextColor(Color.BLACK);
    passwordView.setTextColor(Color.BLACK);
    confirmPasswordView.setTextColor(Color.BLACK);
    signUpView.setTextColor(Color.BLACK);

    // Set background color
    signUpView.setBackgroundColor(Color.LIGHT_GRAY);

    // Set text size
    nameView.setTextSize(16);
    cityView.setTextSize(16);
    provinceView.setTextSize(16);
    emailView.setTextSize(16);
    passwordView.setTextSize(16);
    confirmPasswordView.setTextSize(16);
    signUpView.setTextSize(16);

    // Set hint text
    nameView.setHint("Nama Lengkap");
    cityView.setHint("Kota");
    provinceView.setHint("Provinsi");
    emailView.setHint("Email");
    passwordView.setHint("Password");
    confirmPasswordView.setHint("Konfirmasi Password");

    // Set input type
    emailView.setInputType(InputType.TYPE_EMAIL);
    passwordView.setInputType(InputType.TYPE_PASSWORD);
    confirmPasswordView.setInputType(InputType.TYPE_PASSWORD);

    // Set focus
    nameView.requestFocus();

    // Set click listener for sign up button
    signUpView.setOnClickListener(new View.OnClickListener() {
        @Override public void onClick() {
            // Get input values
            name = nameView.getText().toString().trim();
            city = cityView.getText().toString().trim();
            province = provinceView.getText().toString().trim();
            email = emailView.getText().toString().trim();
            password = passwordView.getText().toString().trim();
            confirmPassword = confirmPasswordView.getText().toString().trim();

            // Validate input
            if (name.isEmpty() || city.isEmpty() || province.isEmpty() || email.isEmpty() || password.isEmpty() || confirmPassword.isEmpty()) {
                Toast.makeText(Register.this, "All fields are required", Toast.LENGTH_SHORT).show();
                return;
            }

            // Check if email is already registered
            boolean isEmailRegistered = false;
            for (User user : users) {
                if (user.getEmail().equals(email)) {
                    isEmailRegistered = true;
                    break;
                }
            }

            // Check if password and confirm password match
            if (!password.equals(confirmPassword)) {
                Toast.makeText(Register.this, "Password and confirm password do not match", Toast.LENGTH_SHORT).show();
                return;
            }

            // Register user
            User user = new User(name, city, province, email, password);
            users.add(user);

            // Show success message
            Toast.makeText(Register.this, "Registration successful", Toast.LENGTH_SHORT).show();

            // Clear input fields
            nameView.setText("");
            cityView.setText("");
            provinceView.setText("");
            emailView.setText("");
            passwordView.setText("");
            confirmPasswordView.setText("");

            // Focus on email field
            emailView.requestFocus();
        }
    });
}
```

Create a new account

Create your new account.

Nama Lengkap *

Username *

Email Aktif *

Nomor Handphone *

Password *

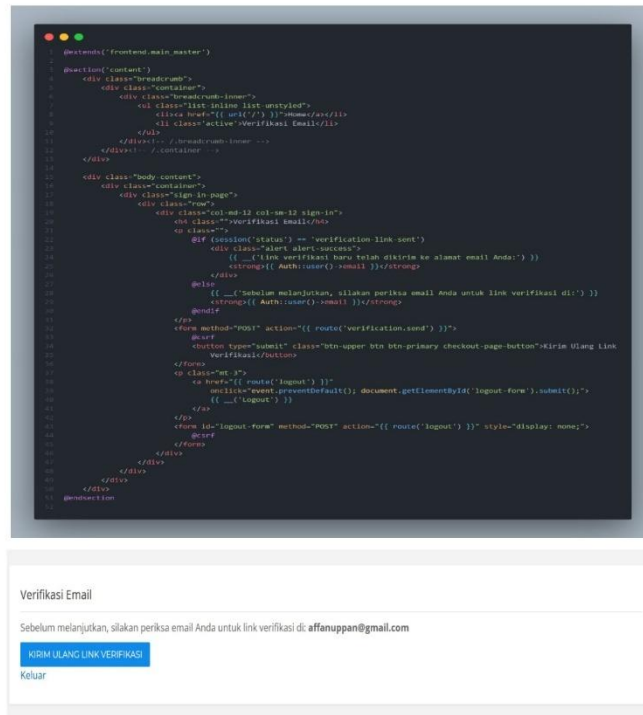
Confirm Password *

[SIGN UP](#)

Gambar 1. 2. *Source Code* dan Tampilan Register

3. Tampilan Verifikasi Email

Pada tampilan halaman verifikasi Email, form menggunakan method 'POST' dan mengirim data ke route 'verify.email'. Jika ada pesan informasi, pesan resend, atau pesan kesalahan dari session, pesan tersebut akan ditampilkan di bagian atas form. Pada halaman ini juga terdapat judul 'Verifikasi Email' dan pesan “Sebelum Melanjutkan, Silahkan periksa email anda untuk link verifikasi” yang ditampilkan di bagian atas form. Terdapat pula form lain untuk mengirim permintaan pengiriman ulang verifikasi ke route `resend.otp` dengan tombol 'Kirim Ulang Link Verifikasi'. Tampilan *source code* halaman verifikasi Email dapat dilihat pada Gambar 1.3



Gambar 1. 3. Source Code dan Tampilan Verifikasi Email

4. Tampilan Lupa Password

Pada tampilan halaman lupa password, form menggunakan method 'POST' dan mengirim data ke route 'password.request'. Form ini memiliki satu inputan yaitu email yang bertipe 'text' dimana *user* dapat memasukkan email mereka untuk mendapatkan link reset password. Jika ada kesalahan input pada email, pesan kesalahan akan ditampilkan di bawah input email tersebut. Ketika user berhasil mengirim permintaan reset password, pesan sukses akan ditampilkan. Pada halaman ini juga terdapat judul 'LUPA PASSWORD' yang ditampilkan di bagian atas form. Tombol 'Request Reset Password' digunakan untuk mengirim permintaan reset password. Tampilan *source code* halaman reset password dapat dilihat pada Gambar 1.4.

5. Tampilan Rreset Password

Pada tampilan halaman reset password, form menggunakan method 'POST' dan mengirim data ke route 'password.update'. Form ini memiliki dua inputan yaitu password baru dan konfirmasi password. Kedua inputan tersebut bertipe 'password' dimana user harus memasukkan password baru dan mengonfirmasi password baru mereka. Jika ada kesalahan dalam input, pesan kesalahan akan ditampilkan di atas form. Pada halaman ini juga terdapat judul 'Reset Password' yang ditampilkan di bagian atas form. Tombol 'RESET PASSWORD' digunakan untuk mengirim permintaan reset password. Tampilan *source code* halaman reset password dapat dilihat pada Gambar 1.3.



Gambar 1. 5. Source Code dan Tampilan Reset Password

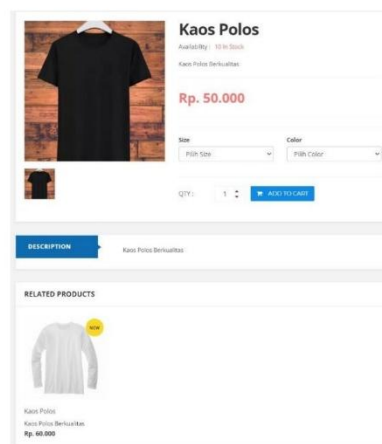
6. Tampilan Home

Pada tampilan halaman home, terdapat berbagai bagian yang menampilkan daftar produk untuk Pengguna, di mana setiap item produk memiliki nama, harga, dan icon keranjang untuk menampilkan popup informasi dari data produk yang dipilih kemudian terdapat button add to cart untuk memasukan produk kedalam keranjang, tekan pada gambar produk atau nama produk untuk dapat melihat lebih detail terkait dari produk tersebut termasuk deskripsi, ukuran, dan harga, selain itu pada halaman home juga menampilkan beberapa kategori. Tampilan *source code* tampilan home dapat dilihat pada Gambar 1.5.

7. Tampilan Detail

Pada tampilan halaman detail, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method detail pada `IndexController`. Pada method detail data produk diambil dari model produk. Di halaman ini pengguna dapat melihat informasi lengkap mengenai produk tertentu yang mereka pilih. Halaman ini menampilkan gambar produk, nama produk, ukuran, jumlah, dan deskripsi produk tersebut. Terdapat juga tombol untuk menambahkan produk ke dalam keranjang pengguna. Selain itu, pengguna dapat melihat produk lain dari kategori yang sama. *Source Code* tersebut dapat dilihat pada Gambar 1.7.



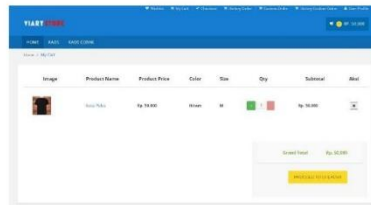


Gambar 1. 7. Source Code dan Tampilan Detail Produk

8. Tampilan Keranjang

Pada halaman keranjang, pengguna dapat melihat daftar produk yang telah mereka simpan ke dalam keranjang. Setiap produk ditampilkan dengan nama, gambar produk, harga, warna, size, dan tombol untuk mengatur jumlah produk yang akan dibeli serta tombol untuk menghapus produk dari daftar keranjang. Tombol "x" memungkinkan pengguna untuk menghapus produk dari daftar keranjang mereka tanpa perlu meninggalkan halaman. *Source Code* tersebut dapat dilihat pada Gambar 1.8.



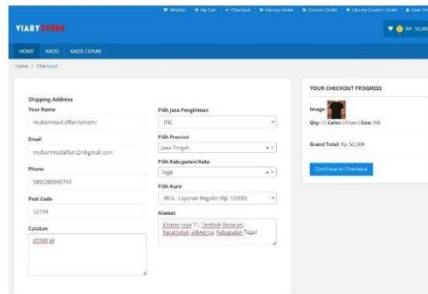


Gambar 1. 8. Source Code dan Tampilan Keranjang

9. Tampilan Checkout

Pada halaman checkout, menampilkan beberapa informasi data yang dimana user harus mengisi informasi dengan lengkap untuk memproses pesanan. Setiap produk ditampilkan dengan nama, gambar produk, harga, warna, size, kemudian terdapat pilihan untuk memilih jasa pengiriman. *Source Code* tersebut dapat dilihat pada Gambar 1.9.





Gambar 1. 9. *Source Code* dan Tampilan Checkout

10. Tampilan Detail Checkout

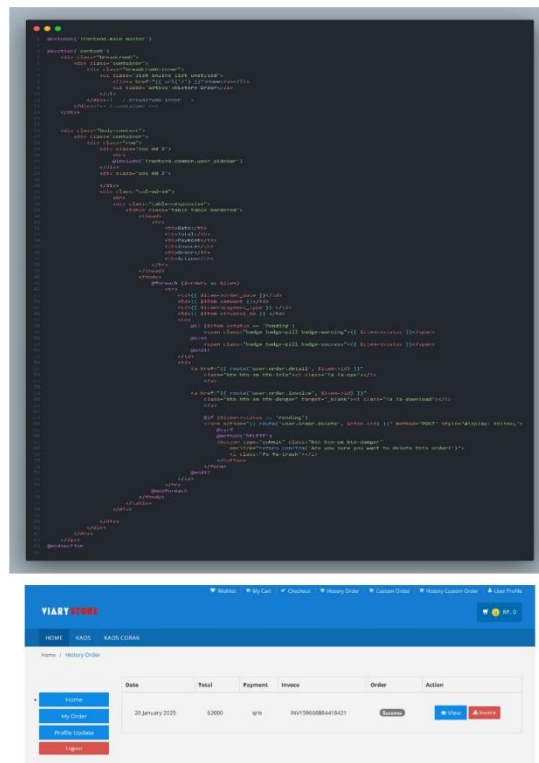
Pada tampilan halaman detail checkout, pengguna dapat melihat informasi lengkap mengenai produk yang mereka pilih. Halaman ini menampilkan gambar produk, harga, jumlah, ukuran, warna dan informasi nama penerima, nomor handphone serta Alamat, dan button 'Pay now' dengan menggunakan fungsi `Route::post()` yang mengarahkan ke method `checkoutstore` pada controller `UserCheckoutController`, dan pada tampilan view kemudian dihubungkan ke api `midtrans` untuk menampilkan popup pembayaran dari `midtrans`. *Source Code* tersebut dapat dilihat pada Gambar 1.10.



Gambar 1. 10. Source Code dan Tampilan Detail Checkout

11. Tampilan History

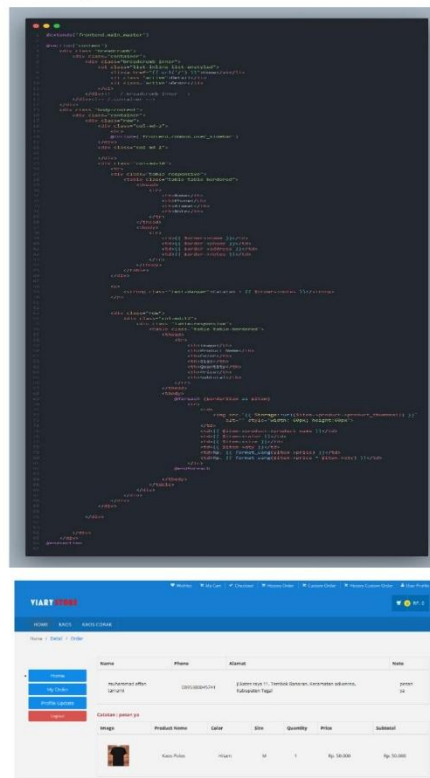
Untuk menampilkan histori dari order pengguna dengan menggunakan laravel, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method `index` pada `UserOrderController`. Pada method `index`, data histori order diambil dari model `order`, kemudian disaring berdasarkan `user_id` yang sedang terautentikasi. Tampilan histori order ditampilkan dalam tabel yang responsive, dengan status order ditampilkan menggunakan badge, seperti 'sukses' jika order berhasil dilakukan. *Source Code* tersebut dapat dilihat pada Gambar 1.11.



Gambar 1. 11. *Source Code* dan Tampilan History Order

12. Tampilan Detail History

Untuk menampilkan detail histori dari order pengguna dengan menggunakan laravel, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method `orderDetail` pada `UserOrderController`. Pada method `orderDetail`, data histori order diambil dari model `order`, kemudian disaring berdasarkan `user_id` yang sedang terautentikasi. Tampilan detail histori order ditampilkan dalam tabel yang responsive, dengan menampilkan detail lengkap, seperti, nama produk, harga, jumlah, ukuran, dan informasi nama penerima. *Source Code* tersebut dapat dilihat pada Gambar 1.12.



Gambar 1. 12. *Source Code* dan Tampilan Detail History

13. Tampilan Custom order

Untuk menampilkan form dari Customorder pengguna dengan menggunakan laravel, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method index pada `UserCustomOrderController`. Pada method index, data histori order diambil dari model order, Tampilan Custom order ditampilkan dalam tabel yang responsive. *Source Code* tersebut dapat dilihat pada Gambar 1.13.



Gambar 1. 13. *Source Code* dan Tampilan Custom Order

14. Tampilan history Custom order

Untuk menampilkan histori dari custom order pengguna dengan menggunakan laravel, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method history pada `UserCustomOrderController`. Pada method history, data histori custom order diambil dari model `CustomOrder`, kemudian disaring berdasarkan `user_id` yang sedang terautentikasi. Tampilan detail histori order ditampilkan dalam tabel yang responsive, dengan menampilkan detail lengkap, seperti, total, size, harga, ongkir. *Source Code* tersebut dapat dilihat pada Gambar 1.14.

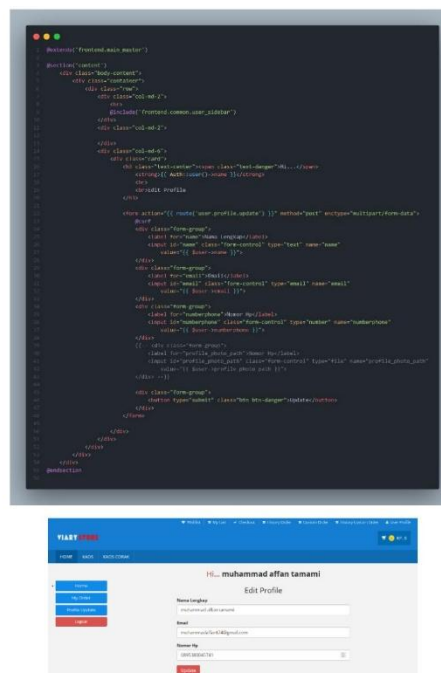


Gambar 1. 14. *Source Code* dan Tampilan History Custom Order

15. Tampilan detail history Custom order

Untuk menampilkan detail histori dari custom order pengguna dengan menggunakan laravel, langkah pertama adalah membuat sebuah route dengan menggunakan fungsi `Route::get()` yang mengarahkan ke method detail pada `UserCustomOrderController`. Pada method detail, data detail histori custom order diambil dari model `CustomOrder`, kemudian disaring berdasarkan `user_id` yang sedang terautentikasi. Tampilan detail histori order ditampilkan dalam tabel yang responsive, dengan menampilkan detail lengkap, seperti total, deskripsi, size, harga, ongkir dan subtotal, terdapat button 'pay now' dengan menggunakan fungsi `Route::post()` yang mengarahkan ke method `customOrderStore` pada `UserCustomOrderController` yang dihubungkan ke API midtrans untuk menampilkan popup pembayaran dari midtrans, apabila pengguna belum melakukan pembayaran. *Source Code* tersebut dapat dilihat pada Gambar 1.14.

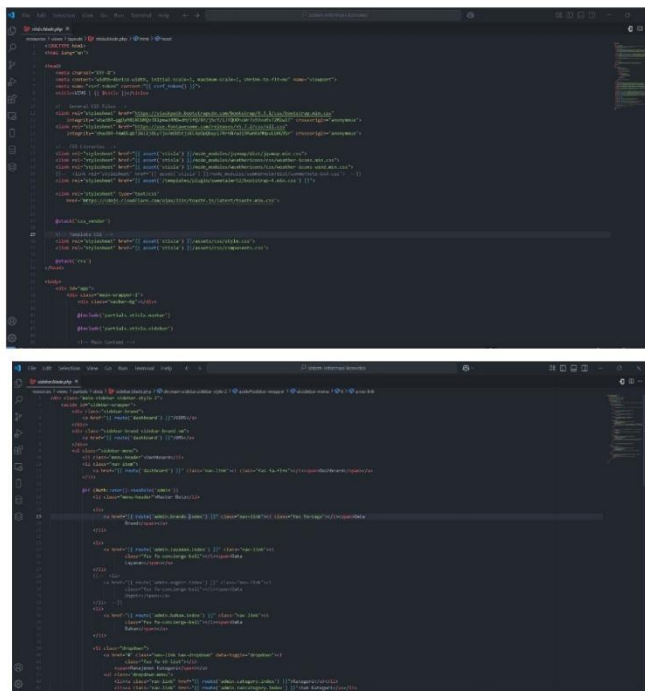
Metode ini mengirimkan data tersebut ke halaman view profile untuk ditampilkan. Selain itu, menggunakan `Route::post()` yang mengarah ke fungsi metode `userProfileUpdate` memproses pembaruan informasi profil berdasarkan data yang dikirimkan melalui form, termasuk validasi untuk memastikan nama diisi yang benar. Setelah berhasil diperbarui, pengguna akan diarahkan kembali ke halaman profil dengan pemberitahuan sukses, memastikan fungsionalitas lengkap untuk mengelola dan memperbarui profil dengan mudah dan aman.. *Source Code* tersebut dapat dilihat pada Gambar 1.16.

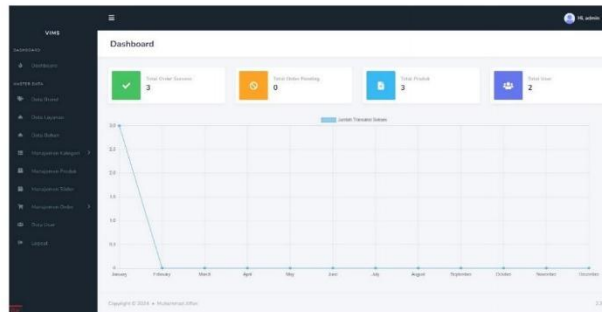


Gambar 1. 16. *Source Code* dan Tampilan Profile User

17. Tampilan Halaman Dashboard Admin

Template Blade stisla.blade.php dirancang untuk menyediakan pengalaman dashboard yang komperhensif bagi Pengguna admin dalam aplikasi laravel. Template ini menggunakan Bootstrap dan gaya CSS kustom untuk tata letak yang responsive dan menarik secara visual. Template dimulai dengan navbar yang mencakup sidebar yang berisi tautan navigasi untuk berbagai fungsionalitas admin seperti mengelola produk. Setiap tautan navigasi route() dikonfigurasi untuk mengarahkan pengguna ke tugas atau halaman tertentu. Template juga mencakup fitur logout untuk manajemen sesi pengguna, menggunakan JavaScript untuk interaksi pengguna yang lebih baik dan mengintegrasikan plugin Chart.js dan DataTables untuk visualisasi dan manajemen data. Konfigurasi ini memastikan bahwa admin dapat dengan efisien menjalajahi dan mengelola tugas dalam aplikasi sambil mempertahankan antarmuka yang konsisten dan ramah pengguna. Source Code dashboard bisa dilihat pada gambar 1. 17.





Gambar 1. 17. Tampilan *Source Code* Dashboard Admin

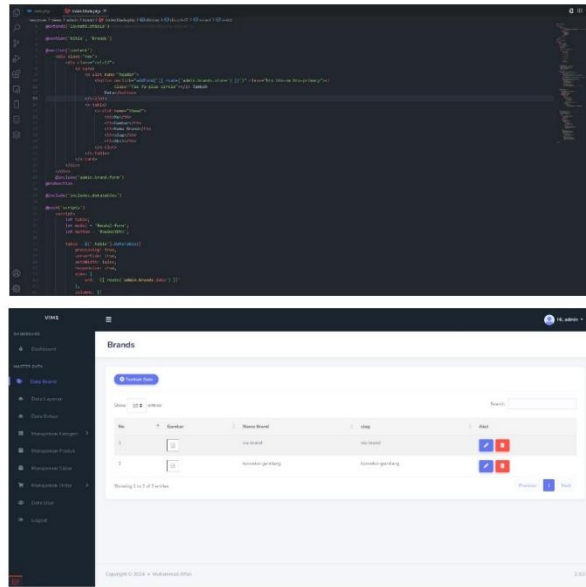
18. Tampilan Brand Admin

Halaman data brand adalah bagian dari antarmuka admin yang digunakan untuk mengelola brand produk. Halaman ini mengadopsi layout utama dari `stisla.blade.php` dalam admin, menampilkan nama brand produk dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri brand. Form penambahan brand memungkinkan pengguna untuk memasukkan nama brand baru dan menyimpannya melalui aksi POST ke route `'admin.brands.store'`.

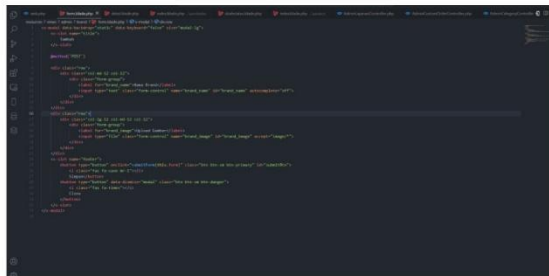
Setiap baris tabel menampilkan nomor urut, nama brand, serta tombol edit dan hapus. Tombol edit mengarahkan pengguna ke halaman modal bootstrap `form.blade.php` untuk mengubah detail brand yang sudah ada, sementara tombol hapus menghapus brand dengan metode DELETE. Teknologi yang digunakan meliputi PHP Blade Templating untuk integrasi data dinamis dari backend dengan tata letak HTML, Bootstrap untuk responsivitas dan tata letak UI, serta Laravel Routing untuk mengarahkan pengguna ke fungsi CRUD yang sesuai dalam model brand. Halaman ini memfasilitasi administrasi efektif dan efisien dari brand produk dalam sistem aplikasi manajemen konveksi.

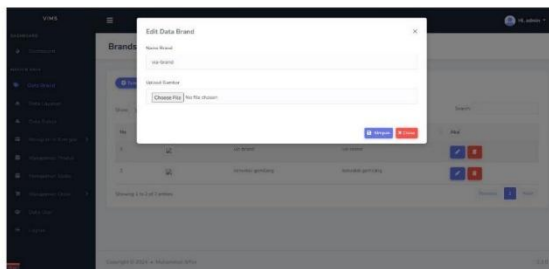
Halaman edit brand menyediakan formulir untuk mengubah detail sebuah brand yang sudah ada. Dengan menggunakan metode POST ke `admin.brands.update`, halaman ini menampilkan kolom input yang memuat nama brand yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "simpan". Implementasi menggunakan PHP Blade Templating untuk menampilkan formulir dengan data yang diisi sebelumnya dari model produk, Bootstrap untuk tata letak dan elemen UI seperti kolom input dan tombol "Save", serta Laravel Routing untuk memproses perubahan yang dilakukan

pada brand. Ini memungkinkan admin untuk mengelola dan memperbarui informasi brand produk secara efisien dalam aplikasi manajemen konveksi. *Source code* dapat dilihat pada Gambar 1.18. dan Gambar 1.19. berikut ini.



Gambar 1. 18. Tampilan *Source Code* data Brand Admin





Gambar 1. 19. Tampilan *Source Code* data Brand Edit

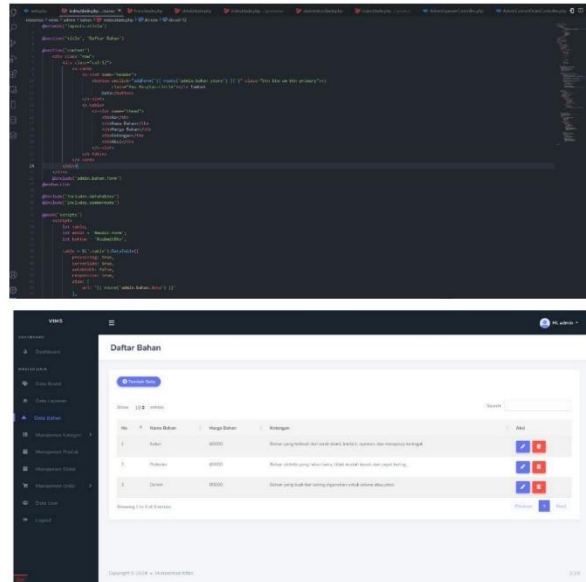
19. Tampilan data Bahan Admin

Halaman data bahan adalah bagian dari antarmuka admin yang digunakan untuk mengelola brand produk. Halaman ini mengadopsi layout utama dari `stisla.blade.php` dalam admin, menampilkan nama brand produk dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri bahan. Form penambahan brand memungkinkan pengguna untuk memasukkan nama brand baru dan menyimpannya melalui aksi POST ke route `'admin.bahans.store'`.

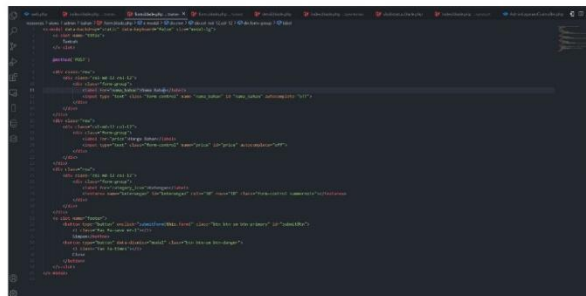
Setiap baris tabel menampilkan nomor urut, nama brand, serta tombol edit dan hapus. Tombol edit mengarahkan pengguna ke halaman modal bootstrap `form.blade.php` untuk mengubah detail brand yang sudah ada, sementara tombol hapus menghapus brand dengan metode DELETE. Teknologi yang digunakan meliputi PHP Blade Templating untuk integrasi data dinamis dari backend dengan tata letak HTML, Bootstrap untuk responsivitas dan tata letak UI, serta Laravel Routing untuk mengarahkan pengguna ke fungsi CRUD yang sesuai dalam model brand. Halaman ini memfasilitasi administrasi efektif dan efisien dari brand produk dalam sistem aplikasi manajemen konveksi.

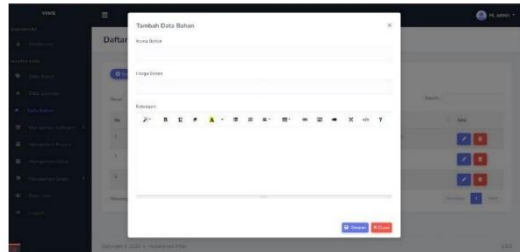
Halaman edit data bahan menyediakan formulir untuk mengubah detail sebuah brand yang sudah ada. Dengan menggunakan metode PUT ke `admin.bahans.update`, halaman ini menampilkan kolom input yang memuat nama brand yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "simpan". Implementasi menggunakan PHP Blade Templating untuk menampilkan formulir dengan data yang diisi sebelumnya dari model produk, Bootstrap untuk tata letak dan elemen UI seperti kolom input dan tombol "Save", serta Laravel Routing untuk memproses perubahan yang dilakukan pada bahan. Ini memungkinkan admin untuk mengelola dan memperbarui informasi

brand produk secara efisien dalam aplikasi manajemen konveksi. *Source code* dapat dilihat pada Gambar 1.20, Gambar 1.21 dan Gambar 1.22. berikut ini.

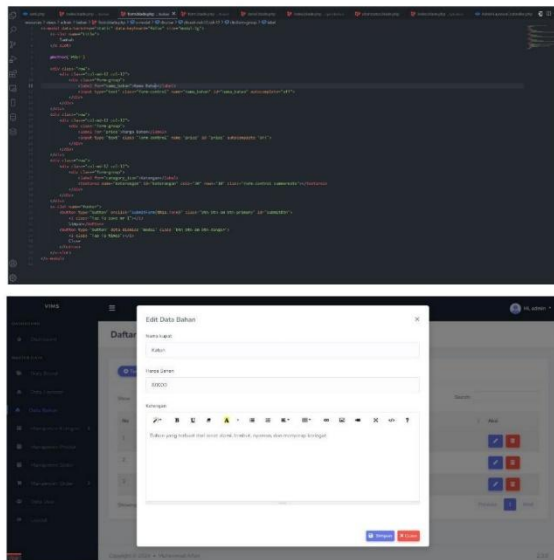


Gambar 1. 20. Tampilan *Source Code* Index bahan





Gambar 1. 21. Tampilan *Source Code* Create Data Bahan



Gambar 1. 22. Tampilan *Source Code* Edit Data Bahan

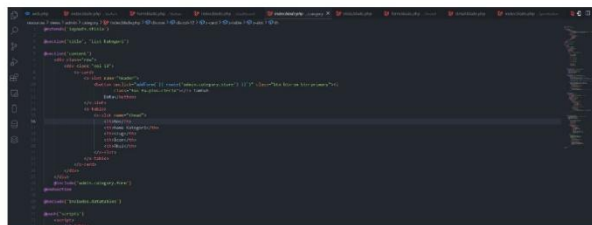
20. Tampilan Kategori Admin

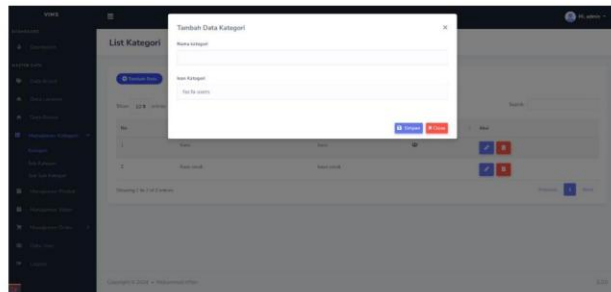
Halaman dashboard katagori ini dirancang untuk memungkinkan pengelolaan jenis kategori dalam aplikasi dengan menggunakan Laravel. Halaman utama menampilkan daftar kategori dengan opsi untuk menambah, mengedit, dan menghapus kategori. Desain halaman menggunakan Bootstrap untuk tampilan responsif yang baik. Setiap kategori ditampilkan dalam tabel dengan kolom-kolom yang mencakup informasi seperti nama kategori. Pengguna dapat menambahkan kategori baru dengan mengklik tombol "Tambah Data", yang mengarahkan mereka ke halaman formulir untuk mengisi detail kategori seperti nama Kategori.

Halaman ini juga memungkinkan pengguna untuk mengedit informasi nama kategori dengan mengklik ikon pensil, yang mengarahkan mereka ke formulir prapengisian dengan data kategori yang dapat diubah.

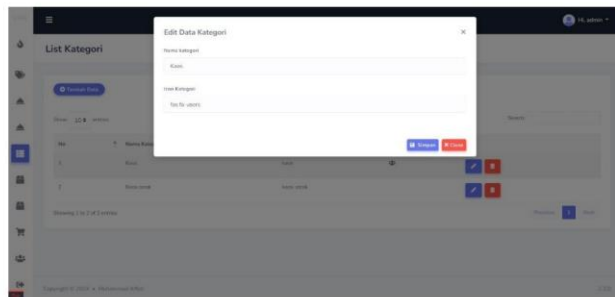
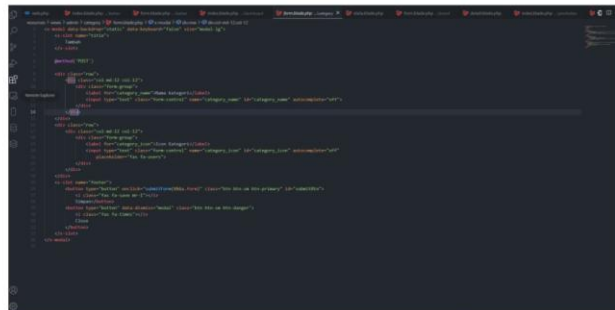
Proses penghapusan kategori dilakukan dengan mengklik ikon sampah yang mengarahkan pengguna ke konfirmasi penghapusan. Semua operasi ini diimplementasikan dengan menggunakan Laravel dan PHP untuk logika backend, sementara tampilan frontend menggunakan Blade templating engine untuk integrasi yang mulus. Fitur pagination disediakan secara otomatis oleh Laravel untuk menangani banyaknya data kategori, dengan tautan navigasi di bagian bawah tabel untuk navigasi yang mudah.

Secara keseluruhan, dashboard kategori ini dirancang untuk memberikan pengalaman pengelolaan produk yang intuitif dan efisien dalam aplikasi berbasis web menggunakan Laravel. *Source code* dapat dilihat pada Gambar 1.23., Gambar 1.24. berikut ini.





Gambar 1. 23. Tampilan *Source Code* Tambah Data Kategori



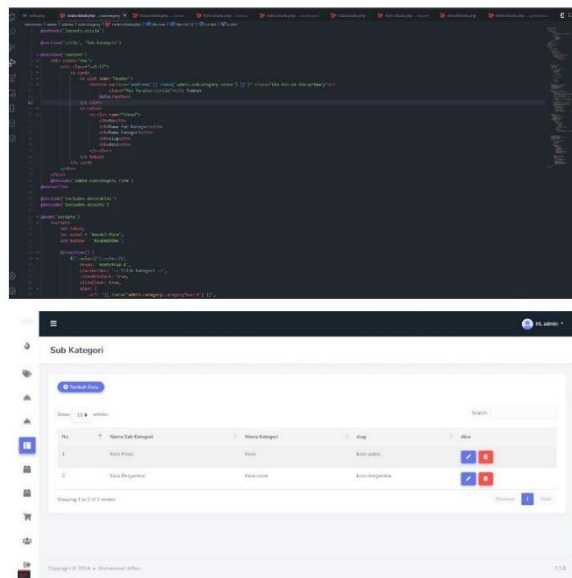
Gambar 1. 24. Tampilan *Source Code* Edit Kategori

21. Tampilan SubKategori Admin

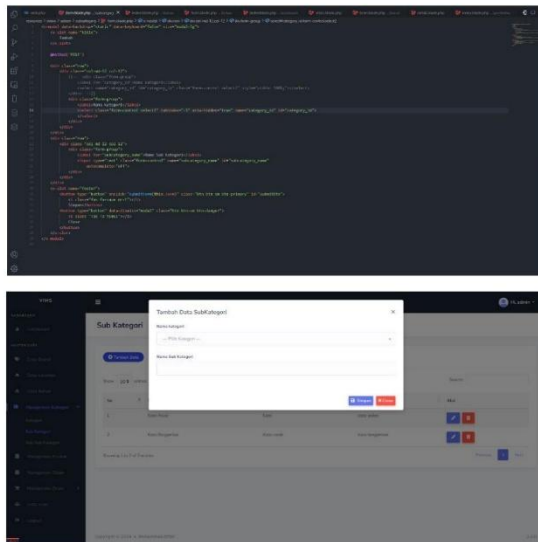
Halaman ini mengadopsi layout utama dari `stisla.blade.php` dalam admin, menampilkan nama SubKategori, nama Kategori, slug dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri SubKategori. Form penambahan SubKategori memungkinkan pengguna untuk memasukkan nama SubKategori baru dan menyimpannya melalui aksi POST ke route `'admin.subcategory.store'`.

Tombol edit mengarahkan pengguna ke halaman modal bootstrap `form.blade.php` untuk mengubah detail nama kategori dan subkategori yang sudah ada, sementara tombol hapus menghapus subkategori dengan metode DELETE.

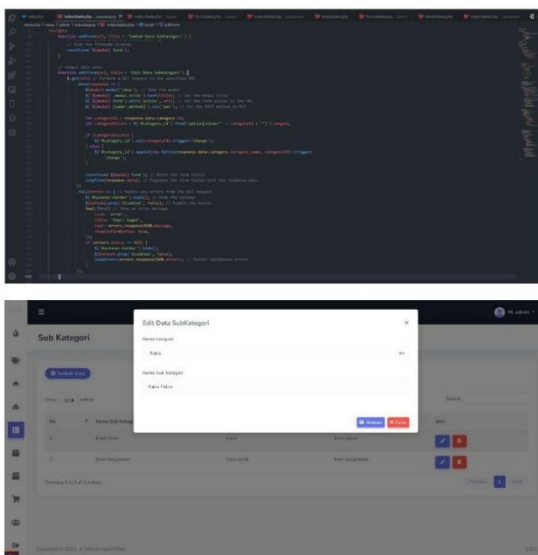
Halaman edit Subkategori menyediakan formulir untuk mengubah detail sebuah kategori dan subkategori yang sudah ada. Dengan menggunakan metode PUT ke `admin.subcategory.update`, halaman ini menampilkan kolom input yang memuat nama subkategori dan dropdown dari data kategori yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "simpan". *Source code* dapat dilihat pada Gambar 1.25., Gambar 1.26, Gambar 1.27, berikut ini.



Gambar 1. 25. Tampilan *Source Code* SubKategori



Gambar 1. 26. Tampilan *Source Code* Create Subkategori



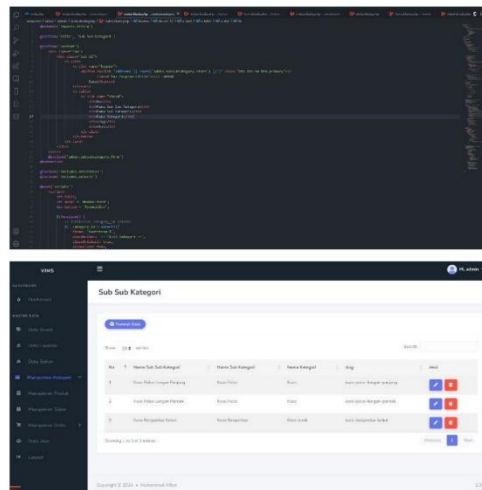
Gambar 1. 27. Tampilan *Source Code* Edit Subkategori

22. Tampilan SubSubKategori Admin

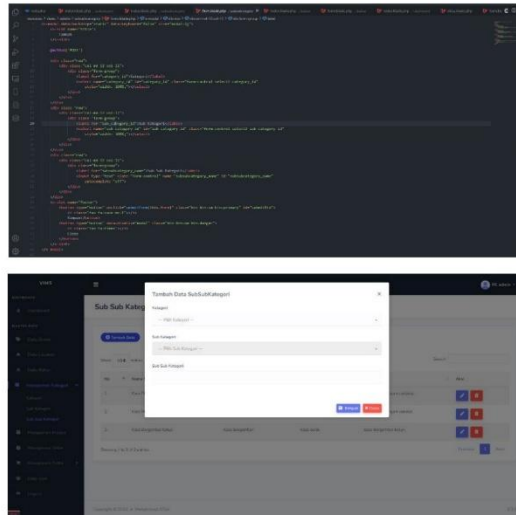
Halaman ini mengadopsi layout utama dari `stisla.blade.php` dalam admin, menampilkan nama SubSubKategori, nama SubKategori, nama Kategori slug dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri SubKategori. Form penambahan SubKategori memungkinkan pengguna untuk memasukkan nama SubKategori baru dan menyimpannya melalui aksi POST ke route `'admin.subsubcategory.store'`.

Tombol edit mengarahkan pengguna ke halaman modal bootstrap `form.blade.php` untuk mengubah detail nama kategori, nama subkategori, dan nama subsubkategori yang sudah ada, sementara tombol hapus menghapus subsubkategori dengan metode DELETE.

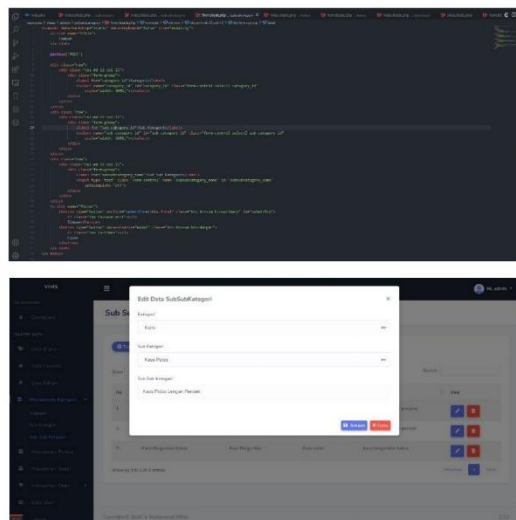
Halaman edit SubSubKategori menyediakan formulir untuk mengubah detail sebuah kategori, subkategori dan subsubkategori yang sudah ada. Dengan menggunakan metode PUT ke `admin.subsubcategory.update`, halaman ini menampilkan kolom input yang memuat dropdown dari data kategori, dropdown dari data subkategori dan inputan nama subsubkategori yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "simpan" *Source code* dapat dilihat pada Gambar 1.28., Gambar 1.29, dan Gambar 1.30 berikut ini.



Gambar 1. 28. Tampilan Source Code SubSubkategori



Gambar 1. 29. Tampilan *Source Code* Create SubSubKategori



Gambar 1. 30. Tampilan *Source Code* Edit SubSubKategori

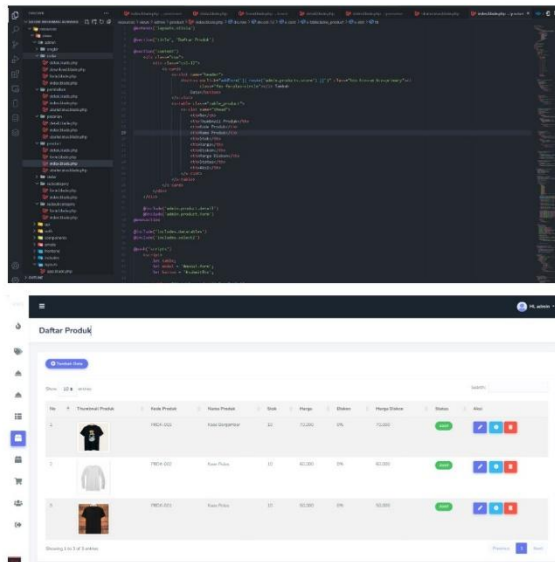
23. Tampilan Manajemen Produk Admin

Halaman dashboard produk ini dirancang untuk memungkinkan pengelolaan produk dalam aplikasi dengan menggunakan Laravel. Halaman utama menampilkan daftar produk dengan opsi untuk menambah, melihat detail, mengedit, dan menghapus produk. Desain halaman menggunakan Bootstrap untuk tampilan responsif yang baik. Setiap produk ditampilkan dalam tabel dengan kolom-kolom yang mencakup informasi seperti thumbnail produk, kode produk, nama produk, stok, harga, diskon dan status. Pengguna dapat menambahkan produk baru dengan mengklik tombol "Tambah Data", yang mengarahkan mereka ke halaman formulir untuk mengisi detail produk seperti nama produk, brand, kategori, harga, dan lainnya kemudian disimpan melalui aksi POST ke route 'admin.products.store'.

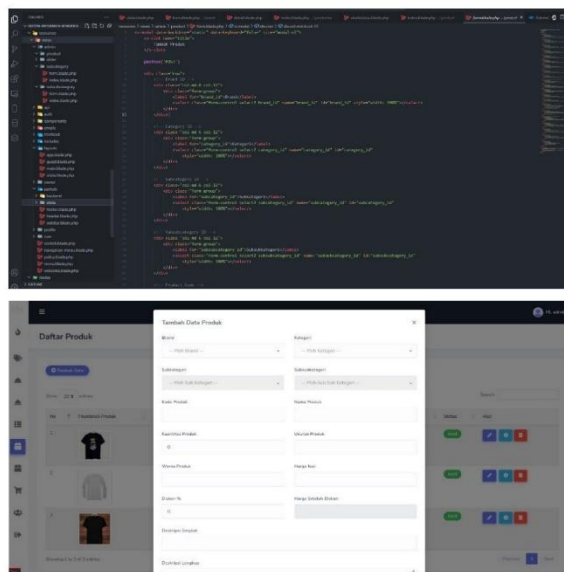
Untuk melihat detail produk, pengguna dapat mengklik ikon "i" untuk membuka halaman detail yang menampilkan informasi lebih lanjut seperti nama produk, kode produk, deskripsi singkat, kategori dan lainnya. Halaman ini juga memungkinkan pengguna untuk mengedit informasi produk dengan mengklik ikon pensil, yang mengarahkan mereka ke formulir prapengisian dengan data produk yang dapat diubah.

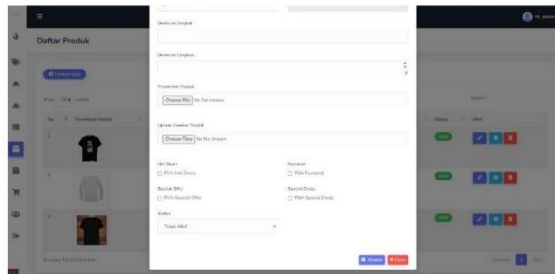
Proses penghapusan produk dilakukan dengan mengklik ikon sampah yang mengarahkan pengguna ke konfirmasi penghapusan. Semua operasi ini diimplementasikan dengan menggunakan Laravel dan PHP untuk logika backend, sementara tampilan frontend menggunakan Blade templating engine untuk integrasi yang mulus. Fitur pagination disediakan secara otomatis oleh Laravel untuk menangani banyaknya data produk, dengan tautan navigasi di bagian bawah tabel untuk navigasi yang mudah.

Secara keseluruhan, dashboard produk ini dirancang untuk memberikan pengalaman pengelolaan Produk yang intuitif dan efisien dalam aplikasi berbasis web menggunakan Laravel. *Source code* dapat dilihat pada Gambar 1.31., Gambar 1.32., Gambar 1.33. dan Gambar 1.34. berikut ini.

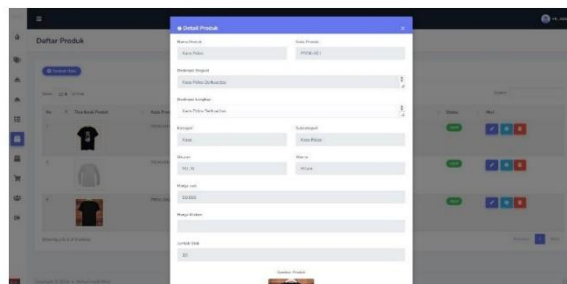
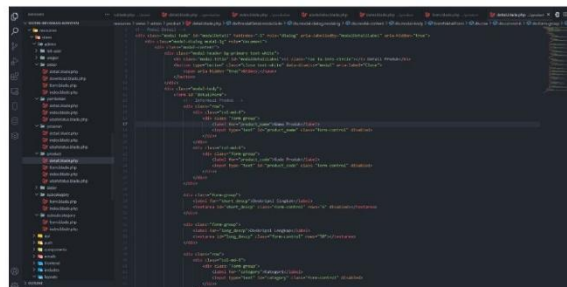


Gambar 1. 31. Tampilan *Source Code* Produk

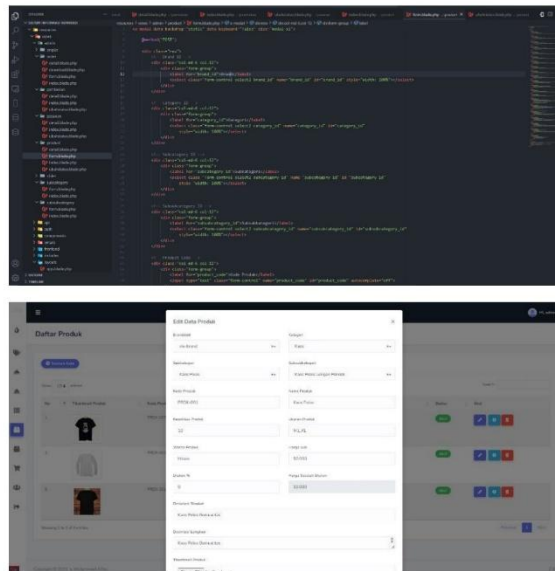




Gambar 1. 32. Tampilan *Source Code* Create Produk



Gambar 1. 33. Tampilan *Source Code* Detail Produk

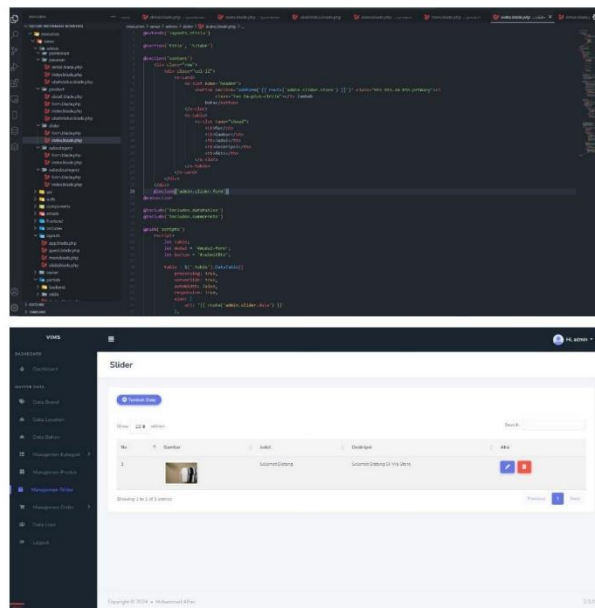


Gambar 1. 34. Tampilan *Source Code* Edit Produk

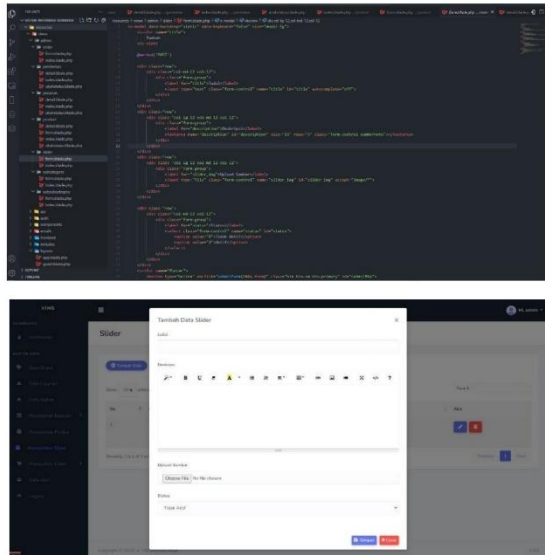
24. Tampilan Manajemen Slider admin

Halaman data slide adalah bagian dari antarmuka admin yang digunakan untuk mengelola slider yang akan ditampilkan di home user pada website. Halaman ini mengadopsi layout utama dari `stisla.blade.php` dalam admin, menampilkan gambar slider dalam sebuah tabel dengan opsi untuk menambah, mengedit, dan menghapus setiap entri slider. Form penambahan slider memungkinkan pengguna untuk memasukkan gambar slide baru dan menyimpannya melalui aksi POST ke route `'admin.slider.store'`.

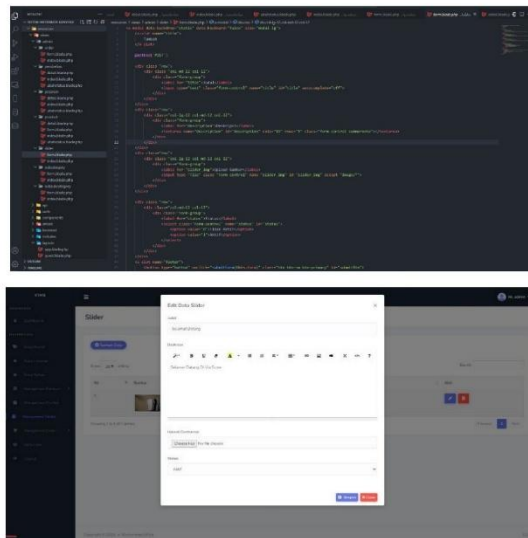
Halaman edit data slide menyediakan formulir untuk mengubah detail sebuah slide yang sudah ada. Dengan menggunakan metode PUT ke `admin.slider.update`, halaman ini menampilkan kolom input yang memuat gambar slide, dan dropdown aktif atau tidak aktif yang sedang diedit. Pengguna dapat melakukan perubahan dan menyimpannya menggunakan tombol "simpan". *Source code* dapat dilihat pada Gambar 1.35., Gambar 1.36, dan Gambar 1.37 berikut ini.



Gambar 1. 35. Tampilan *Source Code* Slider



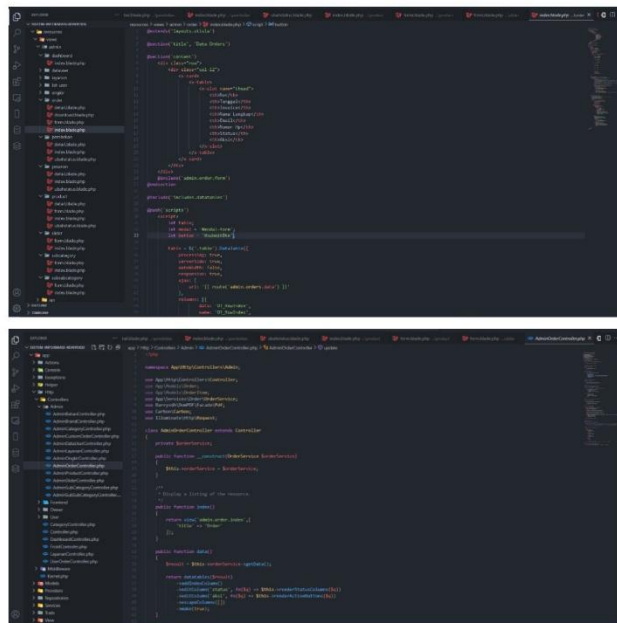
Gambar 1. 36. Tampilan *Source Code* Create Slider

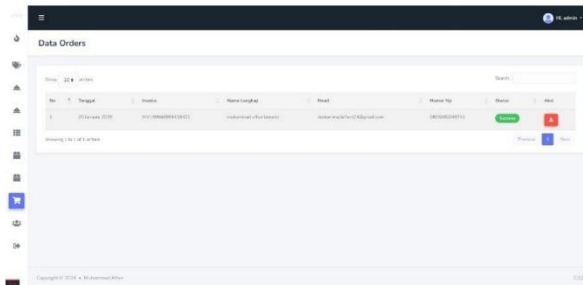


Gambar 1. 37. Tampilan *Source Code* Edit Slider

25. Tampilan Manajemen Order

Pada halaman ini, data order produk ditampilkan dengan detail yang relevan. Saat halaman dimuat, `index()` method dari `AdminOrderController` dijalankan. `AdminOrderController` adalah kontroler untuk mengelola data pesanan di sisi admin, menyediakan fitur seperti menampilkan daftar pesanan (dengan data ditampilkan secara dinamis menggunakan `DataTables`), melihat detail pesanan, serta mengunduh faktur dalam format PDF dengan nama file yang dinamis. Data pesanan dikelola melalui layanan `OrderService`, yang memisahkan logika bisnis dari kontroler. Halaman `index` menampilkan tabel data pesanan dengan kolom seperti tanggal, invoice, nama, email, status, dan tombol aksi yang dapat digunakan untuk mengunduh faktur. Skrip JavaScript mengintegrasikan `DataTables` untuk pemrosesan data secara server-side. *Source code* dapat dilihat pada Gambar 1.38 berikut ini.



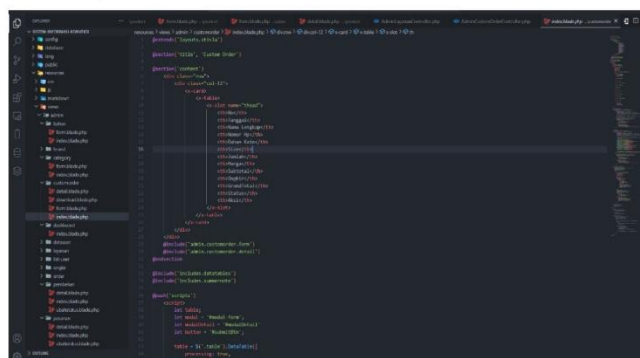


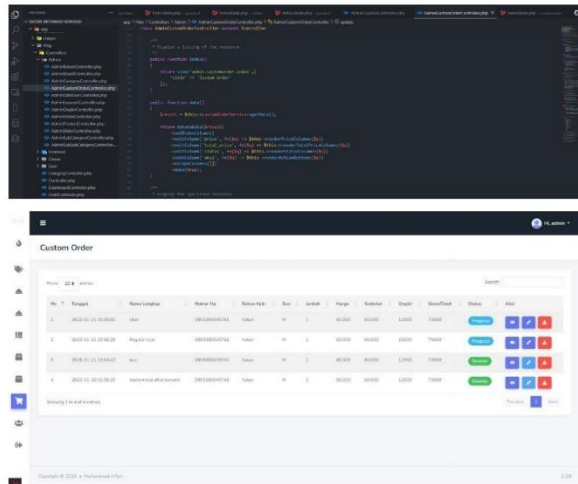
Gambar 1. 38. Tampilan *Source Code* Order

26. Tampilan Custom Order

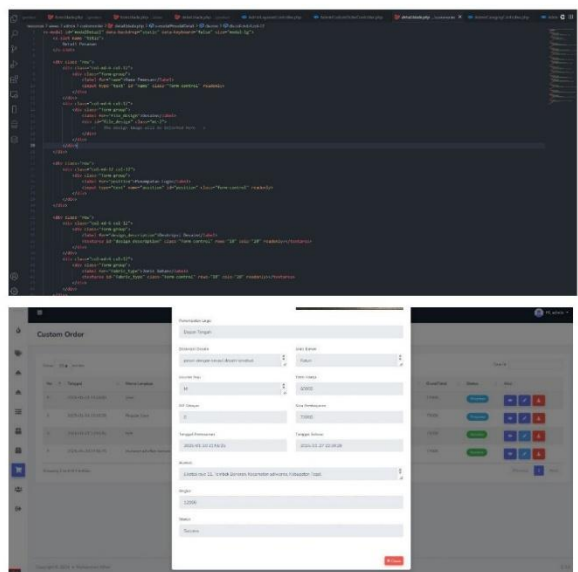
Halaman custom order ini dirancang untuk memungkinkan pengelolaan data custom order dalam aplikasi dengan menggunakan Laravel. Desain halaman menggunakan Bootstrap untuk tampilan responsif yang baik. Setiap data custom order ditampilkan dalam tabel dengan kolom-kolom yang mencakup informasi seperti tanggal pesanan, nama lengkap, nomor handphone, jenis bahan, size, harga dan lain lain.

Untuk melihat detail data custom order, pengguna dapat mengklik ikon “mata” untuk membuka halaman detail yang menampilkan informasi lebih lanjut seperti nama pemesan, ukuran , harga, tanggal pesanan dan lainnya. Controller AdminCustomOrderController bertanggung jawab untuk menangani berbagai operasi, seperti menampilkan data pesanan, mengunduh file desain, menghasilkan invoice dalam format PDF, memperbarui status pesanan. *Source code* dapat dilihat pada Gambar 1.39, Gambar 1.40 berikut ini.





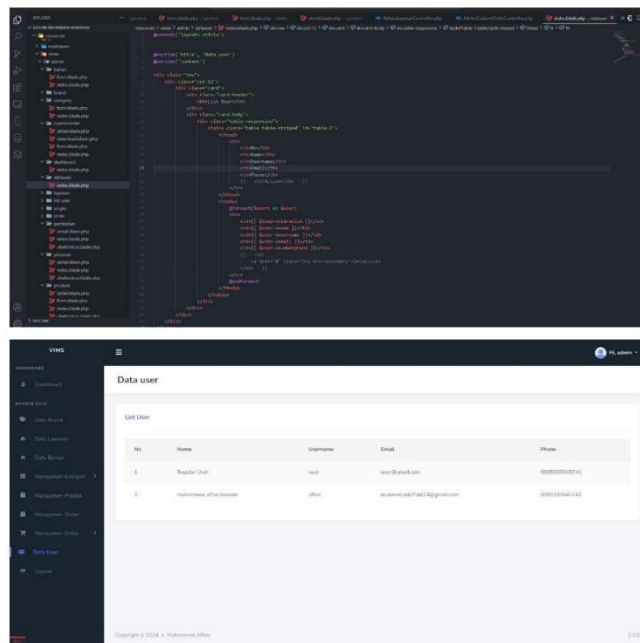
Gambar 1. 39. Tampilan *Source Code* CustomOrder



Gambar 1. 40. Tampilan *Source Code* Detail CustomOrder

27. Tampilan Data User *Customer*

Halaman ini menggunakan untuk menampilkan daftar pengguna dengan informasi seperti nama, username, email, dan nomor telepon. Data pengguna diambil dari database menggunakan model User dengan filter peran user, yang diproses oleh controller AdminDataUserController. Tampilan tabel pada halaman ini responsif, dilengkapi fitur iterasi otomatis menggunakan `$loop->iteration` untuk menampilkan nomor urut. Struktur data disajikan dalam tabel HTML yang dapat dengan mudah dikembangkan untuk menambahkan fitur tambahan, seperti detail pengguna atau tindakan lainnya. Halaman ini dirancang untuk membantu administrator mengelola data pengguna secara efisien. *Source code* dapat dilihat pada Gambar 1.41.



Gambar 1. 41. Tampilan *Source Code* Data User Customer

Lampiran 5 Sertifikat HKI Terbit

 REPUBLIK INDONESIA KEMENTERIAN HUKUM	
SURAT PENCATATAN CIPTAAN	
Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:	
Nomor dan tanggal permohonan	: EC00202524234, 21 Februari 2025
Pencipta	
Nama	: Muhammad Affan Tamami, Dega Surono Wibowo, S.T., M.Kom. dkk
Alamat	: Jl. Kates Raya 11, RT 09/ RW 02, Desa Tembok Banjaran, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah, Adiwerna, Tegal, Jawa Tengah, 52194
Kewarganegaraan	: Indonesia
Pemegang Hak Cipta	
Nama	: Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Alamat	: Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana, Margadana, Tegal, Jawa Tengah, 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Rancang Bangun Sistem Informasi Manajemen Konveksi Berbasis Website
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 21 Februari 2025, di Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor pencatatan	: 000863597
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL u.b Direktur Hak Cipta dan Desain Industri  Agung Damarsasongko,SH.,MH. NIP. 196912261994031001
Disclaimer: Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.	

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Muhammad Affan Tamami	Jl. Kates Raya 11, RT 09/ RW 02, Desa Tembok Banjaran, Kecamatan Adiwerna, Kabupaten Tegal, Provinsi Jawa Tengah, Adiwerna, Tegal
2	Dega Surono Wibowo, S.T., M.Kom.	Perum Sapphire Regency Blok H No.1, RT.004/RW.001, Kelurahan Pulosari Kecamatan Brebes, Kabupaten Brebes, Provinsi Jawa Tengah, Brebes, Brebes
3	Hepatika Zidny Ilmadina, S.Pd.,M.Kom	Jalan Kenanga Gang 1 No. 9, Kelurahan Mangkukusuman, Kecamatan Tegal Timur, Kota Tegal, Provinsi Jawa Tengah, Tegal Timur, Tegal



Lampiran 6. Foto Wawancara



Lampiran 7 Lembar Bimbingan



SARJANA TERAPAN TEKNIK INFORMATIKA POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN SKRIPSI

Nama : Muhammad Affan Tamami
 Nim : 20090151
 No. Ponsel : 0895380045741
 Judul TA : Rancang Bangun Sistem Informasi Manajemen Konveksi Berbasis Website (Studi Kasus Viary Store)
 Dosen Pembimbing I : Dega Suroño Wibowo, S.T., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	30/10/2024	- penggantian judul		af
2.	4/11/2024	- perancangan dan UML Use case diperbaiki sedikit		g-
3.	24/12/2024	- custom order by sistem		g.
4.	24/12/2024	- Output belum masuk sistem		g
5.	2/1/2025	- output diperbaiki dengan (jwb/jnt/ status) - api		g
6.	6/1/2025	- berikan buku manual book		g
7	24/2/2025	- perbaiki diagram use case sequence, activity - sesuai dengan standar yg benar		g



SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA

8.	5/3 2025	perbaiki	class diagram -	f
9.	6/3 2025	Laporan	Or	f
Silahkan - Dyffer Lideny				

Tegal, 6 - Maret - 2025
Dosen Pembimbing I

Dega Surono Wibowo, S.T., M.Kom.
NIPY. 06.014.183



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN SKRIPSI

Nama : Muhammad Affan Tamami
 Nim : 20090151
 No. Ponsel : 0895380045741
 Judul TA : Rancang Bangun Sistem Informasi Manajemen Konveksi Berbasis Website (Studi Kasus Viary Store)
 Dosen Pembimbing II : Hepatika Zidny Ilmadina, S.Pd., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan yang perlu dilakukan	Paraf Pembimbing
1.	30/10/2024	* rancangan sistem	* karena di konveksi yg menerima custom ; maka sistem harus mampu mengakomodir desain tulisan / gambar / logo pada kaos custom	
2.	24/12/2024	* rancangan sistem	Fitur. - tipe bahan => pakai drop down sesuai jenis bahan yg ada - harus ada penempatan logo desain harus ditaruh dimana	
3.	3/1/2024	* aplikasi / sistem	* perhatikan hal yg berkaitan dg informasi yg perlu di-upload customer => cek judul !! label * Harga untuk baju custom => pelanggan belum tau ; harga ada dimana ? bisa tau harga dimana ? krn pas checkout hanya harga ongkir	



**SARJANA TERAPAN TEKNIK INFORMATIKA
POLITEKNIK HARAPAN BERSAMA**

4.	13/1/2025	* aplikasi	* harga total tidak perlu input lagi oleh admin, jadi admin tinggal fllk billing * urus Hfkl	
5.	21/1/2025	* aplikasi	* hfkl	
6.	28/2/25	* laporan	* revisi	
7.	14/03/2025	* revisi laporan	acc	

Tegal, 2025
Dosen Pembimbing II



Hepatika Zidny Ilmadina, S.Pd., M.Kom.