UNIVERSITAS GUNADARMA

FAKULTAS TEKNOLOGI INDUSTRI

TULISAN ILMIAH

PEMBUATAN WEBSITE JUAL BELI MOBIL SHOWROOM YONO MOBILINDO MENGGUNAKAN LARAVEL

Nama	**: Muhammad Tarmidzi Bariq**
NPM	**: 51422161**
Jurusan	**: Informatika**
Pembimbing	**: Dr. Ahmad Sabri, S.Si., M.Si.**

Diajukan Guna Melengkapi Sebagian Syarat Dalam Mencapai Gelar Setara Sarjana Muda

Jakarta

2025

PERNYATAAN ORIGINALITAS DAN PUBLIKASI

Saya yang bertanda tangan di bawah ini,

Nama	**:**	**Muhammad Tarmidzi Bariq**
NPM	**:**	**51422161**
Judul PI	**:**	**PEMBUATAN WEBSITE JUAL BELI MOBIL SHOWROOM YONO MOBILINDO MENGGUNAKAN LARAVEL**
Tanggal Sidang	**:**	
Tanggal Lulus	**:**	

Menyatakan bahwa tulisan ini adalah merupakan hasil karya saya sendiri dan dapat dipublikasikan sepenuhnya oleh Universitas Gunadarma. Segala kutipan dalam bentuk apa pun telah mengikuti kaidah, etika yang berlaku. Mengenai isi dan tulisan adalah merupakan tanggung jawab Penulis, bukan Universitas Gunadarma. Demikian pernyataan ini dibuat dengan sebenarnya dan dengean penuh kesadaran.

Depok, 6 Juli 2025

(Muhammad Tarmidzi Bariq)

LEMBAR PENGESAHAN

Judul PI	:	Pembuatan Website Jual Beli Mobil Showroom Yono Mobilindo Menggunakan Laravel
Nama	:	Muhammad Tarmidzi Bariq
NPM	:	51422161
Tanggal Sidang	:	
Tanggal Lulus	:	

Menyetujui,

Pembimbing	Kasubag. Sidang PI
(Dr. Ahmad Sabri, S.Si., M.Si.)	( )

Ketua Jurusan

(Prof. Dr. Lintang Yuniar Banowosari, S.Kom., M.Sc.)

ABSTRAK

Muhammad Tarmidzi Bariq, 51422161

PEMBUATAN WEBSITE JUAL BELI MOBIL SHOWROOM YONO MOBILINDO MENGGUNAKAN LARAVEL

Tulisan Ilmiah. Informatika, Fakultas Teknologin Industri, Universitas Gunadarma, 2025

Kata Kunci : Jual Beli Mobil, Web Aplikasi, Showroom, Laravel, Informatika

Permasalahan kurangnya transparansi informasi kondisi kendaraan, potensi penipuan, dan keterbatasan transaksi konvensional pada jual beli mobil bekas menjadi alasan pengembangan solusi berbasis *website* untuk *showroom* Yono Mobilindo. Pengembangan sistem dilakukan secara terstruktur dengan metode *System Development Life Cycle* (SDLC) model *waterfall* yang meliputi tahapan requirement, desain, implementasi, pengujian, dan pemeliharaan. *Website* menyediakan fitur pencarian kendaraan sesuai kriteria, sistem ulasan pembeli, formulir penjualan mobil dengan unggah gambar, serta sistem pembayaran *down payment* (DP) yang terintegrasi Midtrans guna meminimalkan risiko transaksi. Pembuatan website didukung dengan tools PHP, Bootstrap, Laravel, Figma, Xampp, MySQL, Visual Studio Code, dan Midtrans untuk mendukung desain antarmuka, pengelolaan data, dan integrasi pembayaran. Metode pengujian *Black Box* digunakan untuk memastikan fungsionalitas sistem sesuai kebutuhan pengguna.

Daftar Pustaka (2020 – 2022)

KATA PENGANTAR

Puji syukur penulis persembahkan kepada Tuhan Yang Maha Esa atas limpahan rahmat dan karunia-Nya sehingga penulisan ilmiah berjudul *"*PEMBUATAN WEBSITE JUAL BELI MOBIL SHOWROOM YONO MOBILINDO MENGGUNAKAN LARAVEL*"* dapat diselesaikan dengan baik.

Penulisan ini disusun sebagai salah satu syarat untuk menyelesaikan tugas akhir pada Jurusan Informatika, Fakultas Teknologi Industri, Universitas Gunadarma.

Penulis menyadari bahwa pencapaian ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ibu Prof. Dr. E.S. Margianti, SE., MM., selaku Rektor Universitas Gunadarma.
2. Bapak Prof. Dr. Ing. Adang Suhendra, SSi., SKom., MSc., selaku Dekan Fakultas Teknologi Industri, Universitas Gunadarma.
3. Ibu Prof. Dr. Lintang Yuniar Banowosari, S.Kom., M.Sc., selaku ketua jurusan informatika Universitas Gunadarma.
4. Bapak Dr. Ahmad Sabri, S.Si., M.Si., selaku dosen pembimbing yang telah banyak memberikan bimbingan, arahan, waktu, tenaga dan kebaikan hati untuk membantu penulis dalam menyelesaikan Penulisan Ilmiah ini hingga selesai.
5. Kedua orang tua, Bapak Sridaryono dan Ibu Endriska Adrianis yang telah memberikan dukungan dan semangat serta doa restu kepada penulis.
6. Teman-teman kelas 3IA11 yang telah membantu dalam menyelesaikan Penulisan Ilmiah ini.
7. Komunitas musang yang telah memberikan masukan dalam berbagai bentuk selama penulisan.
8. Syaifudin Ilham selaku teman yang telah memberikan masukan dalam program aplikasi yang dibuat.

9. Seluruh teman dan sahabat penulis yang telah memberikan dukungan, saran, kritik, dan motivasi kepada penulis dalam membuat Penulisan Ilmiah.

Penulis menyadari bahwa karya ini masih memiliki kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang. Semoga tulisan ini dapat bermanfaat bagi semua pihak yang memerlukannya.

Depok, 6 Juli 2025

(Muhammad Tarmidzi Bariq)

DAFTAR ISI

Halaman Judul i

PERNYATAAN ORIGINALITAS DAN PUBLIKASI ii

LEMBAR PENGESAHAN iii

ABSTRAK iv

KATA PENGANTAR v

DAFTAR ISI vii

DAFTAR TABEL x

DAFTAR GAMBAR xi

BAB 1 PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Batasan Masalah 2

1.3 Tujuan 3

1.4 Metode Penelitian 3

1.5 Sistematika Penulisan 5

BAB 2 LANDASAN TEORI 6

2.1 Konsep Showroom 6

2.2. Teori dan Studi tentang Fitur Sistem 6

2.2.1 Sistem Pencarian dan Filter 6

2.2.2 Sistem Review dan Rating 6

2.2.3 Pembayaran Digital dan Sistem DP 7

2.2.4 Form Penjualan Mobil dan Upload Gambar 7

2.2.5 Informasi dan Dokumentasi Mobil 8

2.3 Teknologi Pengembangan Sistem 8

2.3.1 Bahasa Pemograman dan Tools Backend 8

2.3.2 Teknologi Frontend 10

2.3.3 Tools Pengembangan dan Pengujian 14

2.4 Data Flow Diagram 15

2.5 Entity Relationship Diagram 16

2.6 Activity Diagram 17

2.7 Use Case 18
2.8 Struktur Navigasi 19
2.8.1 Struktur Navigasi Linear 20
2.8.2 Struktur Navigasi Non-Linear 20
2.8.3 Struktur navigasi hirarki 20
2.8.4 Struktur Navigasi Campuran 21
2.9 Black Box Testing 21
2.10 Penelitian Terdahulu 22
BAB 3 PERANCANGAN DAN IMPLEMENTASI 24
3.1 Gambaran Umum Sistem 24
3.2 Analisis Kebutuhan 24
3.2.1 Kebutuhan Fungsional 25
3.2.2 Kebutuhan Non-Fungsional 27
3.3 Perancangan 28
3.3.1 Struktur Navigasi 28
3.3.2 Perancangan Database 31
3.3.3 Use Case Diagram 41
3.3.4 Activity Diagram 43
3.3.5 Perancangan Tampilan 48
3.4 Implementasi 75
3.4.1 Install Laravel 75
3.4.2 Install Laravel UI 76
3.4.3 Install Midtrans 77
3.4.4 Integrasi GitHub dalam Pengelolaan Kode 79
3.5 Tampilan Website 82
3.5.1 Tampilan Halaman Utama 82
3.5.2 Tampilan Halaman Jual Mobil 84
3.5.3 Tampilan Halaman Masuk 86
3.5.4 Tampilan Halaman Daftar 86
3.5.5 Tampilan Halaman Testimoni 87
3.5.6 Tampilan Halaman Dashboard Role Admin 89
3.5.7 Tampilan Halaman Mobil Role Admin 90
3.5.8 Tampilan Halaman Update Status Mobil Role Admin 90
3.5.9 Tampilan Halaman Tambah Mobil Role Admin 91
3.5.10 Tampilan Halaman Show Mobil Role Admin 93
3.5.11 Tampilan Halaman Edit Mobil Role Admin 94
3.5.12 Tampilan Halaman Down Payment Role Admin 96
3.5.13 Tampilan Halaman Show Down Payment Role Admin 96
3.5.14 Tampilan Halaman Add Refund Down Payment Role Admin 97
3.5.15 Tampilan Halaman Edit Refund Down Payment Role Admin 98
3.5.16 Tampilan Halaman Catatan Transaksi Pembelian Role Admin 98
3.5.17 Tampilan Halaman Tambah Transaksi Pembelian Role Admin 99

3.5.18 Tampilan Halaman Permintaan Penjualan Role Admin 100
3.5.19 Tampilan Halaman Update Status Permintaan Penjualan Role Admin 100
3.5.20 Tampilan Halaman Detail Permintaan Penjualan Role Admin..... 101
3.5.21 Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin..... 102
3.5.22 Tampilan Halaman Catatan Transaksi Penjualan Role Admin..... 102
3.5.23 Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin 103
3.5.24 Tampilan Halaman Dashboard Role User..... 104
3.5.25 Tampilan Halaman Down Payment Role User 104
3.5.26 Tampilan Halaman Detail Bayar Down Payment Role User..... 105
3.5.27 Tampilan Halaman Bayar Sekarang Down Payment Role User..... 106
3.5.28 Tampilan Halaman Detail Down Payment Role User 107
3.5.29 Tampilan Halaman Transaksi Pembelian Role User 108
3.5.30 Tampilan Halaman Testimoni Role User..... 109
3.5.31 Tampilan Halaman Permintaan Penjualan Role User..... 109
3.5.32 Tampilan Halaman Detail Permintaan Penjualan role User 110
3.5.33 Tampilan Halaman Update Alamat Role User..... 111
3.5.34 Tampilan Halaman Transaksi Penjualan Role User..... 111

3.6 Hosting 112

3.7 Uji Coba 112

BAB 4 PENUTUP..... 116

4.1 Kesimpulan 116

4.2 Saran..... 116

DAFTAR PUSTAKA 118

LAMPIRAN..... 120

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu 22
Tabel 3. 1 Kebutuhan Fungsional 25
Tabel 3. 2 Kebutuhan Non-Fungsional 27
Tabel 3. 3 Tabel Users 34
Tabel 3. 4 Tabel Cars Types 35
Tabel 3. 5 Tabel Cars 36
Tabel 3. 6 Tabel Car Photos 37
Tabel 3. 7 Tabel Down Payments 37
Tabel 3. 8 Tabel Offers 38
Tabel 3. 9 Tabel Offer Records 39
Tabel 3. 10 Tabel Refunds 39
Tabel 3. 11 Tabel Reviews 40
Tabel 3. 12 Tabel Sales Records 40
Tabel 3. 13 Uji Coba 112

DAFTAR GAMBAR

Gambar 1. 1 Metode Waterfall 4
Gambar 2. 1 Logo PHP 9
Gambar 2. 2 Logo Laravel 9
Gambar 2. 3 Logo MySQL 10
Gambar 2. 4 Logo Midtrans 10
Gambar 2. 5 Logo HTML 11
Gambar 2. 6 Logo CSS 11
Gambar 2. 7 Logo Javascript 12
Gambar 2. 8 Logo Bootstrap 12
Gambar 2. 9 Logo Figma 13
Gambar 2. 10 Screenshoot Tampilan AdminLTE 13
Gambar 2. 11 Logo Visual Studio Code 14
Gambar 2. 12 Logo Git dan Github 15
Gambar 2. 13 DFD 16
Gambar 2. 14 ERD 17
Gambar 2. 15 Activity Diagram 18
Gambar 2. 16 Use Case 19
Gambar 2. 17 Struktur Navigasi Linear (Sumber: J. V. Irwansyah, 2014 dalam Sanawiah & Hartiningsih, 2020) 20
Gambar 2. 18 Struktur Navigasi Non-Linear (Sumber: J. V. Irwansyah, 2014 dalam Sanawiah & Hartiningsih, 2020) 20
Gambar 2. 19 Struktur Navigasi Hirarki 21
Gambar 2. 20 Struktur Navigasi Campuran 21
Gambar 3. 1 Struktur Navigasi Web 29
Gambar 3. 2 Struktur Navigasi Role Admin 30
Gambar 3. 3 Struktur Navigasi Role User 31
Gambar 3. 4 DFD Level 0 32
Gambar 3. 5 DFD Level 1 33
Gambar 3. 6 ERD 34
Gambar 3. 7 Use Case Diagram Role User 42

Gambar 3. 8 Use Case Diagram Role Admin 43
Gambar 3. 9 Activity Diagram Login 44
Gambar 3. 10 Activity Diagram Down Payment 45
Gambar 3. 11 Activity Diagram Pembelian Mobil 46
Gambar 3. 12 Activity Diagram Penjualan Mobil 47
Gambar 3. 13 Perancangan Tampilan Halaman Utama 49
Gambar 3. 14 Perancangan Tampilan Halaman Jual Mobil 51
Gambar 3. 15 Perancangan Tampilan Halaman Masuk 52
Gambar 3. 16 Perancangan Tampilan Halaman Daftar 53
Gambar 3. 17 Perancangan Tampilan Halaman Testimoni 54
Gambar 3. 18 Perancangan Tampilan Dashboard Role Admin 55
Gambar 3. 19 Perancangan Tampilan Halaman Mobil Role Admin 56
Gambar 3. 20 Perancangan Tampulan Update Status Mobi Role Admin..... 56
Gambar 3. 21 Perancangan Tampilan Halaman Tambah Mobil Role Admin 57
Gambar 3. 22 Perancangan Tampilan Halaman Show Mobil Role Admin 57
Gambar 3. 23 Perancangan Tampilan Halaman Edit Mobil Role Admin..... 58
Gambar 3. 24 Perancangan Tampilan Halaman Down Payment Role Admin 59
Gambar 3. 25 Perancangan Tampilan Halaman Show Down Payment Role Admin 60
Gambar 3. 26 Perancangan Tampilan Halaman Refund Down Payment Role Admin 60
Gambar 3. 27 Perancangan Tampilan Halaman Edit Refund Down Payment Role Admin..... 61
Gambar 3. 28 Perancangan Tampilan Catatan Transaksi Pembelian Role Admin62
Gambar 3. 29 Perancangan Tampilan Halaman Tambah Transaksi Pembelian Role Admin..... 63
Gambar 3. 30 Perancangan Tampilan Halaman Permintaan Penjualan Role Admin 64
Gambar 3. 31 Perancangan Tampilan Jalaman Update Status Permintaan Penjualan Role Admin 64
Gambar 3. 32 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role Admin..... 65
Gambar 3. 33 Perancangan Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin 66

Gambar 3. 34 Perancangan Tampilan Halaman Catatan Transaksi Penjualan Role Admin 66
Gambar 3. 35 Perancangan Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin 67
Gambar 3. 36 Perancangan Tampilan Dashboard Role User 68
Gambar 3. 37 Perancangan Tampilan Halaman Down Payment Role User 69
Gambar 3. 38 Perancangan Tampilan Halaman Bayar Sekarang Down Payment Role User 69
Gambar 3. 39 Perancangan Tampilan Halaman Detail Bayar Down Payment Role User 70
Gambar 3. 40 Perancangan Tampilan Halaman Detail Down Payment Role User 71
Gambar 3. 41 Perancangan Tampilan Halaman Transaksi Pembelian Role User 71
Gambar 3. 42 Perancangan Tampilan Halaman Testimoni Role User 72
Gambar 3. 43 Perancangan Tampilan Halaman Permintaan Penjualan Role User73
Gambar 3. 44 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role User 73
Gambar 3. 45 Perancangan Tampilan Halaman Update Alamat Role User 74
Gambar 3. 46 Perancangan Tampilan Halaman Transaksi Penjualan Role User . 74
Gambar 3. 47 Install Laravel Via Composer 75
Gambar 3. 48 Membuat Proyek Laravel Baru 75
Gambar 3. 49 Direktori Proyek Laravel 75
Gambar 3. 50 Install Dependensi dan Kompilasi Aset front-end Laravel 76
Gambar 3. 51 Composer Run Dev 76
Gambar 3. 52 Install Laravel UI 76
Gambar 3. 53 Install Scaffolding Antarmuka Bootsrap 77
Gambar 3. 54 Install Dependensi dan Kompilasi Aset front-end Laravel UI 77
Gambar 3. 55 Install Midtrans Via Composer 77
Gambar 3. 56 Membuat File config/midtrans.php 78
Gambar 3. 57 Variable Environtment Midtrans 78
Gambar 3. 58 Kode MidtransHelper.php 79
Gambar 3. 59 Tombol New Repository Github 79
Gambar 3. 60 Membuat Repository Baru 80
Gambar 3. 61 Salin URL Repository 80

Gambar 3. 62 Perintah Git Init 81
Gambar 3. 63 Perintah Git Add 81
Gambar 3. 64 Perintah Git Remote 81
Gambar 3. 65 Perintah git branch -M main dan git push -u origin main 82
Gambar 3. 66 Tampilan Halaman Utama 83
Gambar 3. 67 Tampilan Halaman Jual Mobil 85
Gambar 3. 68 Tampilan Halaman Masuk 86
Gambar 3. 69 Tampilan Halaman Daftar 87
Gambar 3. 70 Tampilan Halaman Testimoni 88
Gambar 3. 71 Tampilan Halaman Dashboard Role Admin 89
Gambar 3. 72 Tampilan Halaman Mobil Role Admin 90
Gambar 3. 73 Tampilan Halaman Update Status Mobil Role Admin 90
Gambar 3. 74 Tampilan Halaman Tambah Mobil Role Admin 92
Gambar 3. 75 Tampilan Halaman Show Mobil Role Admin 93
Gambar 3. 76 Tampilan Halaman Edit Mobil Role Admin 95
Gambar 3. 77 Tampilan Halaman Down Payment Role Admin 96
Gambar 3. 78 Tampilan Halaman Show Down Payment Role Admin 97
Gambar 3. 79 Tampilan Halaman Add Refund Down Payment Role Admin 97
Gambar 3. 80 Tampilan Halaman Edit Refund Down Payment Role Admin 98
Gambar 3. 81 Tampilan Halaman Catatan Transaksi Pembelian Role Admin 99
Gambar 3. 82 Tampilan Halaman Tambah Transaksi Pembelian Role Admin 99
Gambar 3. 83 Tampilan Halaman Permintaan Penjualan Role Admin 100
Gambar 3. 84 Tampilan Halaman Update Status Permintaan Penjualan Role Admin 101
Gambar 3. 85 Tampilan Halaman Detail Permintaan Penjualan Role Admin 101
Gambar 3. 86 Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin 102
Gambar 3. 87 Tampilan Halaman Catatan Transaksi Penjualan Role Admin 103
Gambar 3. 88 Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin 103
Gambar 3. 89 Tampilan Halaman Dashboard Role User 104
Gambar 3. 90 Tampilan Halaman Down Payment Role User 105
Gambar 3. 91 Tampilan Halaman Detail Bayar Down Payment Role User 106

Gambar 3. 92 Tampilan Halaman Bayar Sekarang Down Payment Role User.. 107
Gambar 3. 93 Tampilan Halaman Detail Down Payment Role User 108
Gambar 3. 94 Tampilan Halaman Transaksi Pembelian Role User..... 108
Gambar 3. 95 Tampilan Halaman Testimoni Role User..... 109
Gambar 3. 96 Tampilan Halaman Permintaan Penjualan Role User..... 110
Gambar 3. 97 Tampilan Halaman Detail Permintaan Penjualan role User..... 110
Gambar 3. 98 Tampilan Halaman Update Alamat Role User..... 111
Gambar 3. 99 Tampilan Halaman Transaksi Penjualan Role User..... 112

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Kemajuan di bidang teknologi informasi berkontribusi terhadap peningkatan yang substansial dalam industri perdagangan kendaraan, terutama dalam aktivitas jual beli mobil bekas. Teknologi digital, khususnya internet, memberikan kemudahan bagi konsumen untuk mengakses berbagai informasi tentang kendaraan bekas secara lebih cepat dan efisien. Di Indonesia, pasar mobil bekas terus berkembang pesat karena permintaan masyarakat terhadap kendaraan pribadi yang lebih terjangkau terus meningkat. Dengan harga mobil baru yang relatif tinggi, mobil bekas menjadi pilihan yang lebih ekonomis bagi banyak orang.

Dalam konteks pasar mobil bekas, terdapat beberapa permasalahan yang menjadi hambatan utama dalam kelancaran transaksi, antara lain adalah kurangnya transparansi mengenai kondisi kendaraan, akuratan informasi terkait harga, serta potensi adanya penipuan antara penjual dan pembeli. Banyak pembeli merasa khawatir akan keaslian informasi mengenai mobil yang dijual seperti riwayat kecelakaan atau kondisi mesin yang sebenarnya

Beberapa penelitian terkait aplikasi jual beli mobil bekas berbasis *website* telah dilakukan untuk mengatasi masalah-masalah tersebut. Penelitian mengenai pengalaman pengguna aplikasi jual beli mobil bekas menunjukkan bahwa informasi berkualitas tinggi, seperti gambar dan deskripsi kendaraan yang jelas, dapat meningkatkan kepercayaan pengguna.

Penelitian ini dirancang menggunakan situs web melalui penerapan fitur pencarian untuk memudahkan pengguna dalam menentukan mobil sesuai dengan pilihan, fitur ulasan pembeli yang memberikan gambaran dari pengalaman pengguna sebelumnya, serta fitur *down payement* (DP) yang memungkinkan pembeli untuk memeriksa mobil terlebih dahulu sebelum melakukan pembelian,

aplikasi ini diharapkan dapat meningkatkan transparansi serta kepercayaan dalam proses jual beli. Selain itu, fitur jual mobil yang memudahkan penjual dalam menawarkan mobilnya, beserta informasi terkait mobil yang dijual yang mencakup kondisi kendaraan, harga, dan riwayat servis, diharapkan dapat menciptakan transaksi yang lebih aman dan efisien bagi kedua belah pihak.

Dengan penerapan solusi-solusi tersebut, diharapkan aplikasi jual beli mobil bekas berbasis situs web yang dikembangkan oleh Yono Mobilindo diharapkan mampu memberikan pengalaman yang lebih memuaskan bagi pengguna dan memperbesar kepercayaan di pasar mobil bekas yang semakin penuh tantangan.

1.2 Batasan Masalah

Cakupan penelitian ini dibatasi pada beberapa aspek spesifik yang akan dijelaskan pada uraian berikut.

1. Aplikasi hanya mencakup mobil bekas sebagai barang yang dijual.
2. Pembayaran pada aplikasi ini dibatasi pada metode pembayaran DP (*Down Payment*) yang memungkinkan pembeli untuk memeriksa mobil terlebih dahulu sebelum melakukan pembelian.
3. Aplikasi ini dilengkapi dengan fitur pencarian yang dirancang untuk mempermudah pengguna dalam menemukan kendaraan sesuai dengan kriteria yang diinginkan.
4. Informasi yang disediakan di aplikasi mencakup detail mobil yang dijual, termasuk kondisi kendaraan, harga, dan riwayat servis mobil.
5. Fitur ulasan pembeli hanya berlaku untuk pengguna yang telah menyelesaikan transaksi pembelian mobil melalui aplikasi.
6. Aplikasi ini hanya melayani transaksi jual beli mobil bekas yang berada dalam wilayah Jabodetabek.

1.3 Tujuan

Tujuan yang ingin dicapai dalam pelaksanaan penelitian ini dapat dirinci sebagai berikut:

1. Membuat aplikasi jual beli mobil bekas berbasis situs web yang dilengkapi dengan fitur pencarian, ulasan pembeli, sistem *down payement* (DP), fitur jual mobil, dan informasi detail mobil untuk mempermudah transaksi jual beli mobil bekas secara *online*.
2. Memecahkan masalah kurangnya transparansi informasi mengenai kondisi mobil dan riwayat transaksi dalam pasar mobil bekas, serta mengurangi potensi penipuan antara penjual dan pembeli.
3. Memberikan wawasan lebih dalam tentang penggunaan sistem pembayaran *down payement* (DP) dan ulasan untuk memastikan keamanan, kepercayaan, dan kualitas dalam transaksi jual beli kendaraan.

1.4 Metode Penelitian

Penelitian ini menerapkan pengembangan perangkat lunak *System Development Life Cycle* (SDLC) tipe *waterfall*. Pendekatan ini dipilih untuk memastikan pengembangan aplikasi dilakukan secara terstruktur dan sistematis melalui tahapan-tahapan yang jelas dan terarah.

Gambar 1. 1 Metode Waterfall

Menurut Romadhon, Yudhistira, & Mukrodin (2021), tahapan dalam model SDLC tipe waterfall adalah sebagai berikut:

1. *Requirement definition*

   Proses ini meliputi penentuan fitur, batasan, serta tujuan sistem dengan melakukan konsultasi langsung bersama pengguna. Semua elemen tersebut kemudian dirinci secara mendalam dan dijadikan sebagai spesifikasi sistem.

2. *System and software design*

   Pada tahap ini, arsitektur sistem dikembangkan berdasarkan kebutuhan yang telah ditentukan. Selain itu, dilakukan pula identifikasi serta pemodelan abstraksi dasar dari sistem perangkat lunak beserta hubungan antar komponennya.

3. *Implementation and unit testing*

   Pada tahap ini, desain perangkat lunak direalisasikan dalam bentuk unit program yang kemudian diuji untuk memastikan kesesuaian dengan spesifikasi yang telah ditetapkan.

4. *Integration and system testing*

   Pada tahap ini, unit-unit program diintegrasikan dan diuji secara keseluruhan sebagai satu sistem untuk memastikan pemenuhan semua

persyaratan yang telah ditetapkan. Setelah itu, sistem diserahkan kepada pengguna.

5. *Operation and maintance*

   Pada tahap ini, sistem dipasang dan mulai digunakan oleh pengguna. Selain itu, dilakukan perbaikan terhadap kesalahan yang belum teridentifikasi pada tahap sebelumnya serta pengembangan lanjutan berupa penambahan fitur dan fungsi baru.

1.5 Sistematika Penulisan

Penulisan ilmiah ini disusun berdasarkan sistematika berikut:

Bab 1 : Membahas latar belakang pentingnya pengembangan platform digital untuk jual beli mobil, terutama dalam menjawab kebutuhan masyarakat akan transaksi yang cepat, aman, dan transparan.

Bab 2 : Menguraikan teori-teori yang relevan, mulai dari konsep *showroom*, sistem pembayaran *online*, hingga fitur-fitur utama dalam aplikasi jual beli kendaraan.

Bab 3 : Proses analisa, perancangan, implementasi, dan uji coba tiap fitur utama dalam *website*.

Bab 4 : Bab terakhir ini berfungsi sebagai penutup dari penulisan ilmiah yang dibuat oleh penulis. Pada bagian ini, disampaikan kesimpulan dan saran yang berkaitan dengan pembuatan aplikasi yang telah dilakukan oleh penulis.

BAB 2
LANDASAN TEORI

2.1 Konsep *Showroom*

Showroom adalah bentuk usaha yang berhubungan erat dengan sektor ritel, berfungsi sebagai lokasi penjualan berbagai produk. Saat ini, *showroom* umumnya dimanfaatkan untuk memasarkan kendaraan, baik mobil baru maupun mobil bekas (Noorviandra, Sanjaya, & Aji, 2020).

2.2. Teori dan Studi tentang Fitur Sistem

Pada penelitian ini tersedia empat fitur yaitu sistem pencarian dan *filter,* sistem *review* dan *rating,* pembayaran dan sistem *down payment* (DP) dan fitur formulir penjualan mobil dan *upload* gambar.

2.2.1 Sistem Pencarian dan *Filter*

Fitur pencarian dan *filter* merupakan elemen esensial dalam aplikasi maupun situs web jual beli, yang berfungsi untuk mempermudah pengguna dalam menemukan produk yang sesuai dengan kriteria atau kebutuhannya secara lebih cepat dan efisien. Sistem ini memungkinkan pengguna memasukkan kata kunci (misalnya: brand mobil, model, atau tahun) untuk menampilkan hasil yang relevan dari seluruh data yang tersedia di *platform*.

2.2.2 Sistem *Review* dan *Rating*

Ulasan daring (*online review*) berperan sebagai media promosi yang strategis dan efektif dalam memperkuat aktivitas komunikasi pemasaran antara pelaku usaha dan konsumen. Keberadaan ulasan positif dari pelanggan dapat menjadi faktor penting dalam keputusan belanja secara *online*, karena mampu meningkatkan rasa percaya konsumen dan mendorong keinginan untuk melakukan pembelian di toko tersebut. Sedangkan *rating* merupakan bentuk standar simbolik yang

merepresentasikan tingkat kepuasan pembeli terhadap penjual atau penyedia layanan, biasanya ditunjukkan melalui jumlah bintang dari satu hingga lima, di mana semakin banyak bintang menunjukkan tingkat kepuasan yang lebih tinggi (Kamisa, Devita P, & Novita, 2022).

2.2.3 Pembayaran Digital dan Sistem DP

Secara umum, sistem pembayaran dapat diartikan sebagai mekanisme pemindahan dana dari pihak yang melakukan pembayaran kepada pihak penerima. Salah satu bentuk implementasinya adalah pembayaran digital, yang memanfaatkan teknologi dalam proses penyimpanan, pengelolaan, dan penerimaan dana dalam bentuk digital. Proses transaksi ini dilaksanakan melalui perangkat atau instrumen pembayaran elektronik yang mendukung efisiensi dan kecepatan transfer dana (Maulidah, Astuti, Nisa, Erlangga, & Hambarwati, 2024).

Penerapan fitur *down payment* (DP) atau uang muka merupakan komponen strategis dalam transaksi jual beli mobil secara daring. Fitur ini memungkinkan pembeli membayar sebagian dari total harga kendaraan sebagai bentuk komitmen awal. Setelah pembayaran *down payment* (DP) dilakukan, pembeli diberi kesempatan untuk memverifikasi kondisi kendaraan secara langsung sebelum menyelesaikan pelunasan. Pendekatan ini dapat meningkatkan rasa aman dan kepercayaan konsumen, terutama dalam transaksi bernilai tinggi seperti pembelian kendaraan bermotor.

2.2.4 *Form* Penjualan Mobil dan *Upload* Gambar

Dalam sistem jual beli mobil berbasis daring, formulir penjualan kendaraan merupakan elemen penting yang digunakan oleh penjual untuk menginput informasi terkait unit kendaraan yang akan ditawarkan kepada pemilik *showroom* Yono Mobilindo. Formulir ini umumnya memuat sejumlah data utama, antara lain brand, model, tahun pembuatan, jarak tempuh, jenis transmisi, harga yang ditawarkan, serta kondisi kendaraan secara keseluruhan. Penjual diharapkan mengisi seluruh informasi tersebut secara terperinci dan sistematis melalui formulir digital yang telah disediakan oleh sistem.

Selain data tekstual, *platform* juga menyediakan fitur unggah gambar yang memungkinkan penjual melampirkan dokumentasi visual kendaraan. Gambar yang diunggah mencakup tampilan *eksterior*, *interior*, dan komponen mesin guna memberikan deskripsi visual yang lebih akurat. Keberadaan foto pendukung ini bertujuan untuk meningkatkan transparansi dan memberikan gambaran menyeluruh kepada pemilik *showroom*, sehingga proses seleksi dan evaluasi kendaraan dapat dilakukan secara lebih objektif.

2.2.5 Informasi dan Dokumentasi Mobil

Dalam sistem jual beli mobil secara daring, penyediaan informasi dan dokumentasi kendaraan memiliki peran krusial dalam menjamin transparansi dan meningkatkan kepercayaan calon pembeli. Informasi kendaraan mencakup detail spesifikasi teknis seperti merek, tipe, tahun produksi, kapasitas mesin, jenis transmisi, jenis bahan bakar, dan jarak tempuh.

Dokumentasi visual yang mendukung informasi tersebut meliputi foto-foto kendaraan dari berbagai sudut, termasuk bagian *eksterior*, *interior*, mesin, ban, dan panel instrumen. Ketersediaan informasi yang lengkap dan dokumentasi visual yang memadai akan membantu pembeli dalam mengambil keputusan secara lebih rasional dan mengurangi risiko ketidaksesuaian antara ekspektasi dan kondisi kendaraan yang sebenarnya.

2.3 Teknologi Pengembangan Sistem

2.3.1 Bahasa Pemograman dan Tools Backend

2.3.1.1 PHP

PHP adalah bahasa pemrograman berbasis *scripting* yang berfungsi di sisi server dan dapat disisipkan langsung ke dalam struktur HTML. Bahasa ini banyak digunakan dalam pengembangan halaman web dinamis yang mendukung berbagai fitur interaktif, seperti buku tamu, statistik pengunjung, sistem polling, layanan

email, dan lainnya. PHP pertama kali diperkenalkan oleh Rasmus Lerdorf pada tahun 1994 sebagai solusi untuk pengelolaan konten web yang lebih fleksibel dan efisien. PHP versi awal secara resmi dirilis pada tahun 1995 (Krisbiantoro & Abda'u, 2021).

Gambar 2. 1 Logo PHP

2.3.1.2 Laravel

Laravel merupakan sebuah kerangka kerja aplikasi web yang dirancang dengan sintaks yang ringkas dan elegan. Framework ini memberikan fondasi dan struktur dasar dalam pengembangan aplikasi, memungkinkan pengembang untuk lebih fokus pada penciptaan fitur utama, sementara aspek teknis dan kompleksitas di latar belakang telah ditangani oleh Laravel (Laravel).

Gambar 2. 2 Logo Laravel

2.3.1.3 MySQL

MySQL merupakan *Relational Database Management System* (RDBMS) yang mampu menangani banyak database secara bersamaan dalam satu server. Dalam praktiknya, satu server MySQL dapat digunakan oleh berbagai pengguna untuk

mengelola database yang berbeda. Setiap database tersusun atas struktur penyimpanan data serta isi data itu sendiri (Silalahi, 2022).

Gambar 2. 3 Logo MySQL

2.3.1.4 Midtrans

Midtrans adalah penyedia layanan pembayaran digital yang berfungsi untuk memfasilitasi transaksi pembayaran secara *online* dengan aman dan mudah. Layanan ini digunakan oleh berbagai jenis bisnis, baik yang berbasis *e-commerce* maupun perusahaan dengan model bisnis berbasis transaksi *online* lainnya. Midtrans menyediakan berbagai fitur yang memudahkan pengguna, seperti integrasi dengan berbagai metode pembayaran, dukungan untuk pembayaran menggunakan kartu kredit, transfer bank, hingga dompet digital (Midtrans, n.d.).

Gambar 2. 4 Logo Midtrans

2.3.2 Teknologi Frontend

2.3.2.1 HTML

Muhardian (2016) menyatakan bahwa *Hypertext Markup Language* (HTML) merupakan bahasa pemrograman standar yang digunakan untuk merancang struktur halaman web agar dapat ditampilkan dan diakses melalui jaringan *internet*. Secara umum, tampilan dan konten halaman web yang ditampilkan kepada pengguna disusun menggunakan HTML, yang kemudian diinterpretasikan oleh sistem

komputer agar dapat dipahami dan digunakan secara interaktif. HTML telah menjadi standar global dalam pengembangan situs web, sehingga memungkinkan laman web ditampilkan secara konsisten pada berbagai perangkat, termasuk layar komputer (Limbong & Sriadhi, 2021).

Gambar 2. 5 Logo HTML

2.3.2.2 CSS

Cascading Style Sheets (CSS) merupakan bahasa pemrograman yang digunakan untuk mengatur dan menyusun berbagai elemen dalam halaman web agar tampilan dan tata letaknya sesuai dengan kebutuhan dan preferensi pengembang (Limbong & Sriadhi, 2021). CSS juga berfungsi untuk mengontrol aspek visual dari elemen-elemen tersebut, seperti warna, ukuran, tata letak, dan jenis huruf.

Gambar 2. 6 Logo CSS

2.3.2.3 Javascript

Marquis (2020) menyatakan JavaScript merupakan salah satu komponen utama dalam pengembangan *World Wide Web*, selain HTML dan CSS. Teknologi ini berperan penting dalam menciptakan interaktivitas pada halaman web dan menjadi elemen krusial dalam arsitektur aplikasi web modern. Pada tahap awal pengembangannya, JavaScript diimplementasikan secara khusus pada sisi klien (*client-side*) melalui peramban web (Limbong & Sriadhi, 2021).

Gambar 2. 7 Logo Javascript

2.3.2.4 Bootstrap

Spurlock (2013) menyatakan Bootstrap adalah sebuah framework yang digunakan untuk mengatur gaya dan estetika tampilan antarmuka pada halaman web. Dalam struktur pengembangan web, HTML berperan dalam menyusun tata letak halaman, sementara aspek visual dan penataan tampilannya dikendalikan melalui CSS, yang dapat dioptimalkan penggunaannya dengan bantuan Bootstrap (Ramadhan, 2023).

Gambar 2. 8 Logo Bootstrap

2.3.2.5 Figma

Figma merupakan sebuah platform berbasis web yang dirancang untuk mendukung proses desain kolaboratif dalam pengembangan produk digital, seperti situs web dan aplikasi seluler. Platform ini digunakan secara luas oleh desainer, manajer produk, pengembang, serta pemangku kepentingan lainnya dalam siklus pengembangan produk (Figma, n.d.).

Gambar 2. 9 Logo Figma

2.3.2.6 AdminLTE

AdminLTE merupakan proyek sumber terbuka yang dilisensikan di bawah lisensi MIT(*Massachusetts Institute of Technology*). Lisensi ini memberikan hak kepada pengguna untuk memanfaatkan perangkat lunak dengan fleksibilitas tersebut hampir tanpa batas, selama hak cipta tetap dicantumkan pada setiap salinan, baik sebagian maupun keseluruhan dari perangkat lunak (AdminLTE, 2022).

Gambar 2. 10 Screenshoot Tampilan AdminLTE

2.3.3 Tools Pengembangan dan Pengujian

2.3.3.1 Visual Studio Code

Visual Studio Code (VS Code) merupakan sebuah lingkungan penyuntingan kode yang digunakan dalam pengembangan aplikasi berbasis berbagai bahasa pemrograman. Aplikasi ini terintegrasi dengan GitHub Copilot, yaitu asisten berbasis kecerdasan buatan yang dirancang untuk membantu proses penulisan kode secara efisien. VS Code juga menyediakan fitur-fitur tambahan seperti agent mode, saran penyuntingan lanjutan, dukungan ekstensi, penyesuaian tema, serta akses melalui antarmuka web (VSCODE).

Gambar 2. 11 Logo Visual Studio Code

2.3.3.2 Git dan Github

Somasundaram (2013) menyatakan Git merupakan suatu sistem kontrol versi (*Version Control System* atau VCS) yang berfungsi untuk merekam setiap perubahan yang dilakukan terhadap suatu berkas atau kumpulan berkas dalam rentang waktu tertentu. Setiap perubahan tersebut dapat diberi penanda berupa *tag* agar dapat ditelusuri dan dipulihkan kembali di masa mendatang berdasarkan waktu perubahan yang telah dicatat. VCS memungkinkan pengembang untuk mengelola versi kode secara sistematis (Ramadhan, 2023).

Menurut Thung, Bissyandé, Lo, dan Jiang (2013) serta Wu, Kropczynski, Shih, dan Carroll (2014), GitHub merupakan sebuah platform jejaring sosial dalam bidang pemrograman (*social coding site*) yang memungkinkan pengembang perangkat lunak untuk memamerkan karya serta aktivitas mereka dalam satu tempat terintegrasi. Platform ini menggunakan *Distributed Version Control System*

(DVCS) berbasis Git untuk mencatat setiap perubahan yang dilakukan oleh pengguna. Proyek dan berkas pengembangan yang diunggah akan dikemas dalam repositori, sehingga memudahkan kolaborasi maupun pengembangan mandiri. Selain itu, pengguna dapat dengan bebas dan tanpa biaya berkontribusi pada repositori milik orang lain ataupun mengembangkan proyeknya sendiri (Ramadhan, 2023).

Gambar 2. 12 Logo Git dan Github

2.3.3.3 Hosting

Hosting merupakan layanan penyimpanan digital yang digunakan untuk menempatkan data atau *file* dari sebuah situs web agar dapat diakses melalui jaringan *internet*. Agar dapat diakses oleh pengguna secara luas, data yang tersimpan pada layanan web hosting harus terhubung dengan koneksi *internet*. Jenis data yang disimpan mencakup berbagai format, seperti gambar, dokumen, surat elektronik, video, maupun *file website* secara keseluruhan (Kurniansyah & Sinar , 2020).

2.4 Data Flow Diagram

Menurut Surono (2014) *Data Flow Diagram* (DFD) merupakan diagram yang digunakan untuk memodelkan alur perpindahan data di dalam suatu sistem informasi. Diagram ini menyajikan proses pengolahan data secara sistematis melalui simbol-simbol khusus, yang merepresentasikan entitas, proses, penyimpanan data, serta jalur arus data yang terjadi dalam aktivitas sistem bisnis tertentu (Muliadi, Andriani, & Irawan, 2020).

Keterangan	DeMarco and Yourdan Simbol	Gane and Sarson Simbol
Entitas Luar		
Proses		
Aliran data (data flow)		
Simpan data		

Gambar 2. 13 DFD

2.5 Entity Relationship Diagram

Menurut Saka (2019) *Entity Relationship Diagram* (ERD) merupakan model konseptual yang digunakan untuk menggambarkan hubungan antar entitas dalam proses perancangan basis data secara sistematis dan terstruktur. ERD bertujuan untuk memvisualisasikan entitas, atribut, dan hubungan antar data, sehingga memudahkan dalam memahami struktur data yang saling berelasi di dalam suatu sistem basis data (Noorviandra, Sanjaya, & Aji, 2020).

Notasi	Keterangan
Entitas	**Entitas**, adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
Relasi	**Relasi**, menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
Atribut	**Atribut**, berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)
————	**Garis**, sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

Gambar 2. 14 ERD

2.6 Activity Diagram

Menurut Munawar (2018), *Activity Diagram* adalah salah satu bentuk diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk memodelkan aspek dinamis dari sebuah sistem, termasuk logika prosedural, proses bisnis, dan aliran kerja. Diagram ini memiliki kemiripan fungsi dengan *flowchart*, namun keunggulannya terletak pada kemampuannya dalam menggambarkan perilaku sistem yang berjalan secara paralel. Tujuan utama dari pembuatan *Activity Diagram* adalah untuk memvisualisasikan alur aktivitas dalam sistem, menunjukkan urutan transisi dari satu aktivitas ke aktivitas lainnya, serta memodelkan percabangan dan aliran konkuren yang mungkin terjadi dalam proses sistem (Ramadhan, 2023).

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem,aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Gambar 2. 15 Activity Diagram

2.7 Use Case

Menurut Munawar (2018) diagram *Use Case* adalah representasi fungsi yang ada pada sistem yang dilihat dari sudut pandang pengguna. Proses kerjanya dilakukan dengan menggambarkan interaksi antara pengguna dan sistem yang digunakan dalam suatu skenario mengenai pemanfaatan sistem tersebut. *Use Case* menggambarkan analisis kebutuhan pada tingkat sistem melalui ilustrasi interaksi antara fungsionalitas sistem dan pengguna yang disebut aktor. *Use Case* disusun berdasarkan kebutuhan aktor dan difokuskan pada “Apa” yang perlu dilakukan oleh

perangkat lunak aplikasi, bukan pada “Bagaimana” perangkat lunak tersebut menjalankan fungsinya (Ramadhan, 2023).

Simbol	**Keterangan**
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan *use case*
	Use case : Abstraksi dan interaksi antara sistem dan aktor
	Association : Abstraksi dari penghubung antara aktor dengan use case
	Generalisasi : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
<<include>>	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
<<extend>>	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

Gambar 2. 16 Use Case

2.8 Struktur Navigasi

Struktur navigasi diartikan sebagai alur dalam suatu program yang merepresentasikan rancangan hubungan antar area berbeda, sehingga mempermudah proses pengorganisasian seluruh elemen dalam sebuah situs web (Sanawiah & Hartiningsih, 2020).

Berikut ini adalah cara untuk merancang struktur navigasi:

2.8.1 Struktur Navigasi Linear

Struktur navigasi linier memiliki alur cerita tunggal yang tersusun secara berurutan, menampilkan layar satu per satu sesuai urutan yang telah ditetapkan. Pada jenis struktur ini, pengguna hanya dapat berpindah ke halaman sebelumnya atau berikutnya secara langsung, tanpa melewati lebih dari satu halaman sekaligus (Sanawiah & Hartiningsih, 2020).

Gambar 2. 17 Struktur Navigasi Linear
(Sumber: J. V. Irwansyah, 2014 dalam Sanawiah & Hartiningsih, 2020)

2.8.2 Struktur Navigasi Non-Linear

Struktur navigasi non-linier atau tidak berurutan merupakan bentuk pengembangan dari navigasi linier. Pada struktur ini, diperbolehkan adanya navigasi dengan jalur bercabang (Sanawiah & Hartiningsih, 2020).

Gambar 2. 18 Struktur Navigasi Non-Linear
(Sumber: J. V. Irwansyah, 2014 dalam Sanawiah & Hartiningsih, 2020)

2.8.3 Struktur navigasi hirarki

Struktur navigasi hirarki, yang juga dikenal sebagai struktur bercabang, merupakan bentuk navigasi yang memanfaatkan percabangan untuk menampilkan data sesuai kriteria tertentu. Halaman pertama pada menu disebut *master page* (halaman utama), yang memiliki halaman percabangan bernama *slave page* (halaman pendukung). Ketika salah satu halaman pendukung dipilih atau diakses, halaman tersebut menjadi *master page* berikutnya, dan proses ini berlanjut secara hierarkis. Dalam struktur navigasi ini, tampilan tidak disusun secara linier (Sanawiah & Hartiningsih, 2020).

Gambar 2. 19 Struktur Navigasi Hirarki

2.8.4 Struktur Navigasi Campuran

Struktur navigasi campuran merupakan kombinasi dari ketiga bentuk navigasi sebelumnya, yaitu linier, non-linier, dan hirarki. Struktur ini juga dikenal sebagai struktur navigasi bebas. Jenis navigasi ini banyak diterapkan dalam pembuatan situs web karena mampu mendukung interaksi yang lebih kompleks dan fleksibel (Sanawiah & Hartiningsih, 2020).

Gambar 2. 20 Struktur Navigasi Campuran

2.9 Black Box Testing

Metode *black box* adalah teknik pengujian perangkat lunak yang dilakukan tanpa memerhatikan detail internal program yang diuji. Pengujian ini hanya memeriksa nilai masukan dan keluaran tanpa menelusuri kode sumber yang menghasilkan keluaran tersebut. Keunggulan metode *black box* terletak pada pelaksanaannya yang tidak memerlukan pengetahuan mendalam tentang bahasa pemrograman tertentu. Pengujian dilakukan dari sudut pandang pengguna, sehingga tercipta ketergantungan dan kerja sama antara pengembang (programmer) dan penguji (tester) (Shaleh, Prayogi, Pirdaus, Syawal, & Saifudin, 2021).

2.10 Penelitian Terdahulu

Penelitian terdahulu dijadikan sebagai acuan untuk mendukung landasan teori dan memberikan gambaran mengenai pengembangan sistem serupa. Tabel 2.1 berikut memuat ringkasan beberapa penelitian yang relevan beserta penjelasan singkat mengenai isi dan kontribusinya terhadap penelitian ini.

Tabel 2. 1 Penelitian Terdahulu

No	Penulis	Judul	Isi
1	Andy Kartasasmita, Yermias Duha, Ramalia Noratama Putri	Rancangan Bangun Aplikasi Pemasaran Mobil Bekas Berbasis Website	Penelitian terdahulu oleh penulis pada topik *Rancang Bangun Aplikasi Pemasaran Mobil Bekas Berbasis Website* mengusulkan solusi inovatif berupa aplikasi berbasis web yang memfasilitasi proses jual-beli mobil bekas secara lebih efisien dan praktis. Aplikasi tersebut dirancang untuk membantu penjual memasarkan kendaraan dengan menyediakan fitur pencarian berdasarkan merek, model, harga, dan tahun, serta memungkinkan unggahan informasi lengkap termasuk gambar, spesifikasi teknis, dan riwayat perawatan. Penelitian ini menunjukkan bahwa penerapan sistem berbasis website dapat mengurangi hambatan komunikasi antara penjual dan pembeli, meningkatkan transparansi transaksi, serta membuka peluang bisnis di industri mobil bekas.
2	Muhammad Ridho Ardiansyah ,Inaya Tri Septieni Koearito	Aplikasi Penjualan Sepeda Motor Berbasis Website Menggunakan Metode Prototype (studi kasus : CV Faris Motor)	Penelitian sebelumnya yang dilakukan oleh Ardiansyah dan Koearito (2022) merancang aplikasi penjualan sepeda motor berbasis website menggunakan metode *Prototype* dengan studi kasus pada CV Faris Motor. Aplikasi ini dibangun untuk mengatasi permasalahan pengolahan data penjualan yang sebelumnya masih dilakukan secara manual sehingga menimbulkan kesulitan dalam pencarian data, pengecekan stok, dan

			penyusunan laporan. Sistem yang dikembangkan memfasilitasi pencatatan data konsumen, transaksi penjualan (tunai dan kredit), serta riwayat pembayaran angsuran secara lebih terstruktur. Dengan pemanfaatan teknologi berbasis website, aplikasi ini diharapkan dapat meningkatkan efisiensi kerja pegawai, meminimalisir kesalahan pencatatan, dan mempercepat proses layanan administrasi penjualan.

BAB 3
PERANCANGAN DAN IMPLEMENTASI

3.1 Gambaran Umum Sistem

Website ini dirancang dengan sejumlah fitur fungsional untuk mendukung transaksi jual beli kendaraan secara daring. Fitur pencarian memungkinkan pengguna menelusuri mobil berdasarkan parameter tertentu seperti merek, tahun produksi, dan kisaran harga. Sistem ulasan dan penilaian dari pembeli sebelumnya disediakan guna meningkatkan kepercayaan calon konsumen. Selain itu, tersedia mekanisme pembayaran *down payment* (DP) yang memberikan kesempatan bagi pembeli untuk melakukan inspeksi unit secara langsung sebelum melanjutkan ke transaksi penuh, dengan jaminan sistem pembayaran yang aman. Fitur penjualan mobil memungkinkan pengguna mengisi formulir data kendaraan, mengunggah foto, serta melakukan negosiasi harga. Informasi kendaraan disajikan secara komprehensif, mencakup spesifikasi teknis, riwayat servis dan kepemilikan, serta kondisi unit yang didokumentasikan melalui foto *interior* dan *eksterior*.

3.2 Analisis Kebutuhan

Tahap analisis kebutuhan merupaka langkah awal yang krusial dalam proses pengembangan aplikasi, karena pada tahap in ditentukan fitur serta fungsionalitas utama yang harus dimiliki sistem agar mampu mencapai tujuan pengguna. Analisis ini mencakup identifikasi kebutuhan pengguna, serta kebutuhan fungsional dan non-fungsional, guna memastikan bahwa aplikasi dapat beroperasi secara optimal sesuai konteks penggunaannya.

3.2.1 Kebutuhan Fungsional

Kebutuhan fungsional mencakup spesifikasi kemampuan atau layanan yang harus disediakan oleh sistem, yang menggambarkan tindakan atau fungsi utama yang dijalankan oleh aplikasi. Layanan yang disediakan oleh system meliputi:

Tabel 3. 1 Kebutuhan Fungsional

Kategori	Kebutuhan	Deskripsi
HomePage Yono Mobilindo	Filter	Fitur pencarian berdasarkan kriteria seperti brand, model, tahun, dan jarak tempuh.
	Informasi Mobil Tersedia	Menampilkan daftar mobil yang tersedia beserta detail umum.
	Informasi Testimonial	Menampilkan ulasan dan rating dari pembeli sebelumnya.
	Detail Mobil	Informasi lengkap tentang mobil: spesifikasi, riwayat service, kondisi, dan foto.
	Form Pembayaran DP Mobil	Fitur untuk membayar DP secara aman dan terjamin, memberikan kesempatan pembeli mengecek kondisi mobil langsung.
Jual	Form Jual Mobil	Formulir untuk mengisi detail mobil yang ingin dijual, termasuk harga, dan fitur negosiasi.
Testimonial	Informasi Testimonial	Menampilkan ulasan dan rating dari pembeli sebelumnya.
Kontak	Informasi Kontak	Menampilkan Informasi Kontak Showroom
Masuk	Formulir Masuk	Menampilkan Formulir Masuk
Daftar	Formulir Daftar	Menampilkan Formulir Daftar
User Management System, Role: User		
Dashboard	Informasi Data	Menampilkan Informasi Dashboard
DownPayment	Informasi Data DownPayement	Menampilkan Informasi Data DownPayment
	Bayar Sekarang	Pengguna dapat melakukan pembayaran jika status mobil masih available dan status pembayaran pending
	Detail DownPayment	Menampilkan informasi yang mendetail dari DownPayment
Transaksi Pembelian	Informasi Data Transaksi Pembelian Mobil	Menampilkan informasi data transaksi pembelian mobil
	Testimoni	Menambahkan testimonial ketika sudah melakukan transaksi pembelian mobil
Permintaan Penjualan	Informasi Data Permintaan Penjualan Mobil	Menampilkan informasi data permintaan penjualan mobil

	Detail Permintaan Penjualan	menampilkan informasi permintaan penjualan secara mendetail
Transaksi Penjualan	Informasi Data Transaksi Penjualan Mobil	menampilkan informasi data transaksi penjualan mobil
User Management System, Role : Admin		
Dashboard	Informasi Data	Menampilkan informasi data dashboard
Mobil	Informasi Data Mobil	Menampilkan seluruh data mobil yang tersedia dalam showroom
	Tambah Mobil	Fitur untuk menambahkan data mobil baru ke dalam sistem melalui form input
	Edit Mobil	Memungkinkan pengguna untuk memperbarui atau memperbaiki informasi mobil yang sudah ada di dalam sistem.
	Show Mobil	Menampilkan detail lengkap dari satu mobil tertentu, seperti spesifikasi, status, foto, dan informasi riwayat lainnya.
	Hapus Mobil	Fitur untuk menghapus data mobil dari sistem jika mobil sudah tidak tersedia atau terjadi kesalahan input.
	Perbarui Status Mobil	Digunakan untuk mengubah status mobil, misalnya dari “available” menjadi “sold” atau “under_review” atau "pending_check", sesuai dengan perkembangan transaksi.
	Filter Cars	Fitur yang memungkinkan pengguna untuk memfilter daftar mobil berdasarkan kriteria tertentu
DownPayment	Informasi Data DownPayment	Menampilkan semua data pembayaran uang muka (DP) yang dilakukan oleh pembeli, termasuk nominal, tanggal pembayaran, status, dan metode pembayaran.
	Show DownPayment	Menampilkan detail lengkap dari satu transaksi Down Payment, seperti informasi mobil, pembeli, nominal, status pembayaran, dan tanggal.
	Refund	Fitur untuk mencatat atau memproses pengembalian uang DP kepada pembeli, baik sebagian maupun seluruhnya, beserta bukti transfer.
	Edit Refund	Fitur untuk perubahan pada nomor rekening penerima *refund* dan mengunggah bukti pembayaran yang relevan sebelum mengonfirmasi pengembalian dana.
	Filter DownPayment	Memungkinkan pengguna untuk menyaring daftar data Down Payment berdasarkan kriteria seperti status pembayaran, tanggal, atau nama pembeli.

Catatan Transaksi Pembelian	Informasi Catatan Transaksi Pembelian	Menampilkan seluruh riwayat transaksi penjualan mobil yang telah terjadi, termasuk informasi mobil, pembeli, penjual, harga jual, dan tanggal transaksi.
	Tambah Catatan Transaksi Pembelian	Fitur untuk membuat catatan baru atas transaksi penjualan mobil yang telah disetujui dan diselesaikan antara pembeli dan penjual.
	Hapus Catatan Transaksi Pembelian	Menghapus catatan transaksi penjualan dari sistem apabila terjadi kesalahan input atau transaksi dibatalkan.
Penjualan Mobil	Informasi Penjualan Mobil	Menampilkan daftar semua penjualan mobil yang telah terjadi, lengkap dengan informasi pembeli, penjual, mobil yang dijual, harga, serta tanggal transaksi.
	Show Detail Penjualan Mobil	Menampilkan detail lengkap dari satu transaksi penjualan mobil, seperti informasi mobil, pembeli, nominal, dan tanggal.
	Perbarui Status Penjualan Mobil	Fitur untuk mengubah status transaksi penjualan, seperti dari “accepted”, “pending”, “rejected”, sesuai dengan tahapan proses penjualan.
Catatan Transaksi Penjualan	Informasi Data Catatan Transaksi Penjualan	Menampilkan data penawaran penjualan yang dilakukan oleh pembeli terhadap mobil tertentu, sebelum terjadi transaksi final.
	Tambah Catatan Transaksi Penjualan	Fitur untuk mencatat penawaran pembeli terhadap mobil yang dijual, termasuk harga penawaran, tanggal, dan identitas pihak yang terlibat.

3.2.2 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merujuk pada karakteristik sistem yang mendukung operasional aplikasi, seperti keamanan, performa, dan ketersediaan, yang menjelaskan cara sistem seharusnya bekerja. Berikut ini kebutuhan non-fungsional sistem meliputi :

Tabel 3. 2 Kebutuhan Non-Fungsional

Perangkat Keras	
Spesifikasi	Detail
Nama Perangkat	Macbook Air M3

Prosesor	M3
Ram	16GB
Perangkat Lunak	
Spesifikasi	Detail
Sistem Operasi	macOS
Code Editor	Visual Studio Code
Server Local	Xampp
Framework	Laravel, Boostrap 5
Bahasa Pemograman	PHP, JavaScript,
Tampilan Dashboard	AdminLTE
API Pembayaran	Midtrans

3.3 Perancangan

3.3.1 Struktur Navigasi

3.3.1.1 Struktur Navigasi Halaman Web

Struktur navigasi aplikasi Yono Mobilindo dirancang secara hierarkis dengan menu utama yang membagi akses pengguna ke lima fitur utama, yaitu: beli mobil, jual mobil, testimoni, kontak, dan masuk/daftar.

Gambar 3. 1 Struktur Navigasi Web

3.3.1.2 Struktur Navigasi Halaman User Management System Role Admin

Struktur navigasi untuk peran admin dalam aplikasi dirancang secara modular guna mendukung pengelolaan data secara efisien. Navigasi utama terdiri dari beberapa menu inti, yaitu: *dashboard*, mobil, *down payment*, catatan transaksi pembelian, penjualan mobil, dan catatan transaksi penjualan.

Gambar 3. 2 Struktur Navigasi Role Admin

3.3.1.3 Struktur Navigasi Halaman User Management System Role User

Struktur navigasi untuk peran **user** dalam sistem ini disusun untuk mendukung aktivitas transaksi secara mandiri dan terarah. Navigasi utama mencakup lima menu utama, yaitu: *dashboard*, *down payment*, transaksi pembelian, permintaan penjualan, dan transaksi penjualan.

Gambar 3. 3 Struktur Navigasi Role User

3.3.2 Perancangan Database

3.3.2.1 Data Flow Diagram

1) Data Flow Diagram Level 0

DFD Level 0 menggambarkan proses utama dalam sistem jual beli mobil bekas Yono Mobilindo yang berinteraksi dengan dua entitas eksternal, yaitu *user* (pembeli/penjual) dan admin. Terdapat satu proses utama yaitu "Sistem Jual Beli Mobil Bekas Yono Mobilindo" yang mengelola berbagai aliran data sebagai berikut:

Gambar 3. 4 DFD Level 0

2) Data Flow Diagram Level 1

DFD Level 1 menggambarkan detail proses dari sistem utama menjadi beberapa sub-proses yang lebih spesifik. Sistem ini melibatkan dua entitas *eksternal*, yaitu user (pembeli/penjual) dan admin**,** serta sejumlah penyimpanan data. Adapun proses-proses utama yang tergambar meliputi:

Gambar 3. 5 DFD Level 1

3.3.2.2 Perancangan Database

Struktur basis data sistem Yono Mobilindo dirancang untuk mengelola proses jual beli mobil bekas secara menyeluruh. Basis data terdiri dari beberapa tabel utama yang saling terhubung, antara lain:

Gambar 3. 6 ERD

1) 3.4.1 Tabel users

Tabel users pada Tabel 3.3 digunakan untuk menyimpan informasi mengenai pengguna dalam sistem. Setiap kolom pada tabel ini memiliki fungsi khusus untuk mencatat data yang diperlukan dalam proses identifikasi, autentikasi, dan pengelolaan peran pengguna secara terstruktur.

Tabel 3. 3 Tabel Users

Tabel: users			
Nama	Jenis	Tak Ternilai	Bawaan

id	bigint(20)	Tidak	Tidak ada
name	varchar(30)	Tidak	Tidak ada
email	varchar(30)	Tidak	Tidak ada
email_verified_at	timestamp	NULL	NULL
password	varchar(255)	Tidak	Tidak ada
phone	varchar(15)	NULL	NULL
address	text	NULL	NULL
role	enum('admin', 'user')	user	user
remember_token	varchar(100)	NULL	NULL
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

2) Tabel cars_types

Tabel car_types pada Tabel 3.4 berfungsi untuk menyimpan informasi tentang jenis-jenis mobil yang terdapat dalam sistem. Setiap kolom dalam tabel ini memiliki peran yang berbeda dalam mencatat dan mengelola data mobil.

Tabel 3. 4 Tabel Cars Types

Tabel : car_types			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
tahun	varchar(5)	Tidak	Tidak ada
brand	varchar(50)	Tidak	Tidak ada
model	varchar(50)	Tidak	Tidak ada
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

3) Tabel cars

Tabel cars pada Tabel 3.5 menyimpan data tentang mobil yang tersedia dalam sistem. Setiap kolom memiliki tujuan spesifik untuk menyimpan informasi terkait mobil, seperti identifikasi, kondisi, dan informasi terkait transaksi.

Tabel 3. 5 Tabel Cars

Tabel : Cars			
Nama	Jenis	Tak Ternilai	Bawaan
id	Bigint(20)	Tidak	Tidak Ada
user_id	Bigint(20)	Tidak	Tidak Ada
brand	varchar(50)	Tidak	Tidak Ada
model	varchar(50)	Tidak	Tidak Ada
year	varchar(5)	Tidak	Tidak Ada
price	int(11)	Tidak	Tidak Ada
transmission	enum('manual', 'automatic')	Tidak	Tidak Ada
description	text	Ya	Null
service_history	date	Ya	Null
fuel_type	varchar(10)	Tidak	Tidak Ada
mileage	varchar(10)	Tidak	Tidak Ada
color	varchar(15)	Tidak	Tidak Ada
tax	date	Tidak	Tidak Ada
engine	int(11)	Tidak	Tidak Ada
seat	int(11)	Tidak	Tidak Ada
bpkb	tinyint(1)	Tidak	Tidak Ada
spare_key	tinyint(1)	Tidak	Tidak Ada
manual_book	tinyint(1)	Tidak	Tidak Ada
service_book	tinyint(1)	Tidak	Tidak Ada
sale_type	enum('user', 'showroom')	Tidak	Tidak Ada
status	enum('available','pending_check','sold','under_review')	Tidak	Tidak Ada
created_at	timestamp	Ya	Null
update_at	timestamp	Ya	Null

4) Tabel car_photos

Tabel car_photos pada Tabel 3.6 berfungsi untuk menyimpan informasi tentang foto-foto mobil, menghubungkan setiap foto dengan mobil yang relevan, dan melacak waktu pembuatan dan pembaruan foto tersebut. Tabel

ini penting untuk sistem yang menyimpan banyak foto terkait mobil, seperti galeri foto mobil di *showroom* atau sistem jual beli mobil.

Tabel 3. 6 Tabel Car Photos

Tabel: car_photos			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
car_id	bigint(20)	Tidak	Tidak ada
photo_url	varchar(255)	Tidak	Tidak ada
number	int(11)	Tidak	Tidak ada
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

5) Tabel down_payments

Tabel down_payments pada Tabel 3.7 berfungsi untuk melacak dan menyimpan informasi terkait pembayaran uang muka, termasuk detail transaksi, status pembayaran, metode yang digunakan, dan tanggal pembayaran. Tabel ini penting untuk sistem yang mengelola transaksi pembayaran, terutama dalam konteks penjualan mobil atau produk lainnya yang memerlukan pembayaran uang muka.

Tabel 3. 7 Tabel Down Payments

Tabel: down_payments			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
user_id	bigint(20)	Tidak	Tidak ada
car_id	bigint(20)	Tidak	Tidak ada
order_id	varchar(100)	NULL	NULL
amount	decimal(15,2)	Tidak	Tidak ada
payment_status	enum('pending', 'confirmed', 'cancelled', 'expired')	pending	pending
payment_method	varchar(20)	NULL	NULL

appointment_date	datetime	Tidak	Tidak ada
payment_date	datetime	NULL	NULL
snap_token	varchar(255)	NULL	NULL
refund_id	bigint(20)	NULL	NULL
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

6) Tabel offers

Tabel offers pada Tabel 3.8 berfungsi untuk mencatat tawaran harga yang diajukan oleh pengguna untuk mobil tertentu, lengkap dengan informasi mengenai merek, model, harga yang ditawarkan, status tawaran, serta catatan terkait inspeksi. Tabel ini sangat berguna dalam sistem jual beli mobil atau platform tawar-menawar yang melibatkan berbagai pihak.

Tabel 3. 8 Tabel Offers

Tabel: offers			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
user_id	bigint(20)	Tidak	Tidak ada
brand	varchar(50)	Tidak	Tidak ada
model	varchar(50)	Tidak	Tidak ada
year	int(11)	Tidak	Tidak ada
mileage	int(11)	Tidak	Tidak ada
offered_price	decimal(15,2)	Tidak	Tidak ada
location_inspection	text	NULL	NULL
status	enum('pending', 'accepted', 'rejected', 'sold')	pending	pending
inspection_date	datetime	NULL	NULL
note	text	NULL	NULL
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

7) Tabel offer_records

Tabel offer_records pada Tabel 3.9 berfungsi untuk mencatat informasi tentang transaksi yang terjadi antara pembeli dan penjual terkait tawaran harga mobil. Tabel ini mencatat harga jual, status transaksi, serta waktu transaksi tersebut dilakukan. Ini penting untuk melacak proses transaksi dalam sistem jual beli mobil atau platform tawar-menawar.

Tabel 3. 9 Tabel Offer Records

Tabel: offer_records			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
offer_id	bigint(20)	Tidak	Tidak ada
seller_id	bigint(20)	Tidak	Tidak ada
buyer_id	bigint(20)	Tidak	Tidak ada
sale_price	decimal(15,2)	Tidak	Tidak ada
sale_date	datetime	Tidak	Tidak ada
status	enum('completed', 'pending', 'cancelled')	Tidak	Tidak ada
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

8) Tabel refunds

Tabel refunds pada Tabel 3.10 berfungsi untuk melacak dan mencatat semua informasi yang berkaitan dengan pengembalian dana, termasuk status pengembalian, nomor rekening, dan bukti pembayaran. Tabel ini sangat berguna untuk memproses dan mengelola pengembalian dana dalam sistem transaksi.

Tabel 3. 10 Tabel Refunds

Tabel: refunds			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
no_rekening_refund	varchar(255)	Tidak	Tidak ada
refund_status	enum('refund', 'no_refund')	Tidak	Tidak ada

refund_payment_proof	varchar(255)	Tidak	Tidak ada
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

9) Tabel reviews

Tabel reviews pada Tabel 3.11 berfungsi untuk menyimpan informasi tentang ulasan yang diberikan oleh pengguna terhadap mobil atau transaksi. Tabel ini mencakup rating, komentar, dan foto yang diunggah oleh pengguna, serta waktu pembuatan dan pembaruan ulasan tersebut. Ini sangat penting untuk sistem yang mengelola *review* atau *feedback* dalam platform jual beli mobil atau layanan terkait.

Tabel 3. 11 Tabel Reviews

Tabel: reviews			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
user_id	bigint(20)	Tidak	Tidak ada
sales_record_id	bigint(20)	NULL	NULL
car_id	bigint(20)	NULL	NULL
rating	int(11)	5	5
comment	text	NULL	NULL
photo_review	varchar(255)	NULL	NULL
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

10) Tabel sales_records

Tabel sales_records pada Tabel 3.12 berfungsi untuk menyimpan catatan transaksi penjualan mobil, termasuk informasi tentang mobil yang dijual, pembeli dan penjual, harga jual, status transaksi, serta waktu transaksi tersebut dilakukan. Tabel ini sangat penting dalam sistem jual beli yang melibatkan transaksi antara penjual dan pembeli.

Tabel 3. 12 Tabel Sales Records

Tabel: sales_records			
Nama	Jenis	Tak Ternilai	Bawaan
id	bigint(20)	Tidak	Tidak ada
car_id	bigint(20)	Tidak	Tidak ada
buyer_id	bigint(20)	Tidak	Tidak ada
seller_id	bigint(20)	Tidak	Tidak ada
sale_price	decimal(15,2)	Tidak	Tidak ada
sale_date	datetime	Tidak	Tidak ada
status	enum('completed', 'pending', 'cancelled')	Tidak	Tidak ada
created_at	timestamp	NULL	NULL
updated_at	timestamp	NULL	NULL

3.3.3 Use Case Diagram

3.3.3.1 Use Case Diagram Role User

Use case diagram menggambarkan alur interaksi antara *user* dan berbagai fitur dalam sistem *showroom* mobil. Pengguna dapat mengakses halaman utama (*home*), melakukan *login* untuk mengakses fitur lebih lanjut, membeli mobil, menjual mobil, memberikan testimoni, melakukan pembayaran *down payment* (DP) serta melihat daftar mobil yang tersedia. *User* juga dapat memonitor riwayat transaksi pembelian, penjualan, dan permintaan penjualan mobil, serta mengakses *dashboard* untuk ringkasan aktivitas sistem. Diagram ini menunjukkan bagaimana pengguna berinteraksi dengan sistem untuk menyelesaikan berbagai transaksi terkait mobil.

Gambar 3. 7 Use Case Diagram Role User

3.3.3.2 Use Case Diagram Role Admin

Use case diagram role admin menggambarkan interaksi antara admin dan fitur-fitur dalam sistem. Admin dapat login ke sistem dan mengelola berbagai data, termasuk data mobil, pembayaran *down payment* (DP), transaksi pembelian dan penjualan mobil, serta permintaan penjualan mobil. Selain itu, admin juga dapat mengakses *dashboard* untuk mendapatkan ringkasan aktivitas sistem. Diagram ini menunjukkan bagaimana admin mengelola data dan transaksi yang terjadi di *showroom*.

Gambar 3. 8 Use Case Diagram Role Admin

3.3.4 Activity Diagram

3.3.4.1 Activity Diagram Login

Activity diagram ini menggambarkan alur aktivitas pengguna dan sistem dalam proses login ke dalam sistem.

Gambar 3. 9 Activity Diagram Login

3.3.4.2 Activity Diagram Down Payment

Activity diagram ini menggambarkan alur aktivitas pengguna (pembeli) saat melakukan pembayaran *down payment* (DP) mobil dalam sistem.

Gambar 3. 10 Activity Diagram Down Payment

3.3.4.3 Activity Diagram Pembelian Mobil

Activity diagram ini menjelaskan alur proses pembelian mobil oleh pembeli, termasuk pengecekan kondisi mobil dan pengisian ulasan setelah transaksi.

Gambar 3. 11 Activity Diagram Pembelian Mobil

3.3.4.4 Activity Diagram Penjualan Mobil

Activity diagram ini menggambarkan alur aktivitas yang dilakukan oleh penjual dalam proses penjualan mobil melalui sistem, termasuk tahapan validasi dan inspeksi oleh admin.

Gambar 3. 12 Activity Diagram Penjualan Mobil

3.3.5 Perancangan Tampilan

3.3.5.1 Perancangan Tampilan Halaman Utama

Tampilan antarmuka halaman utama pada sistem Yono Mobilindo dirancang untuk memberikan pengalaman pengguna yang informatif dan mudah dinavigasi. Struktur halaman ini terdiri dari beberapa bagian utama sebagai berikut:

Gambar 3. 13 Perancangan Tampilan Halaman Utama

3.3.5.2 Perancangan Tampilan Halaman Jual Mobil

Halaman jual mobil pada sistem Yono Mobilindo dirancang untuk memfasilitasi pengguna dalam mengajukan penjualan mobil secara mudah dan terstruktur. Struktur halaman terdiri dari beberapa bagian utama sebagai berikut:

Gambar 3. 14 Perancangan Tampilan Halaman Jual Mobil

3.3.5.3 Perancangan Tampilan Halaman Masuk

Halaman login pada sistem Yono Mobilindo dirancang secara sederhana dan fokus, guna memudahkan pengguna untuk melakukan autentikasi masuk ke dalam sistem.

Gambar 3. 15 Perancangan Tampilan Halaman Masuk

3.3.5.4 Perancangan Tampilan Halaman Daftar

Halaman daftar pada sistem Yono Mobilindo dirancang untuk memfasilitasi proses registrasi akun pengguna baru dengan tampilan yang sederhana dan responsif.

Gambar 3. 16 Perancangan Tampilan Halaman Daftar

3.3.5.5 Perancangan Tampilan Halaman Testimoni

Halaman testimonial pada sistem Yono Mobilindo dirancang untuk menampilkan ulasan dan pengalaman para pelanggan secara visual dan terstruktur, sebagai bentuk bukti sosial dan peningkatan kepercayaan pengguna baru.

Gambar 3. 17 Perancangan Tampilan Halaman Testimoni

3.3.5.6 Perancangan Tampilan Dashboard Role Admin

Halaman *dashboard* dalam admin Yono Mobilindo digunakan untuk menampilkan ringkasan data utama, seperti jumlah *down payment* (DP) terkonfirmasi, permintaan penjualan, dan mobil terjual. Antarmuka ini dilengkapi grafik transaksi serta kalender jadwal appointment untuk mendukung pemantauan aktivitas secara menyeluruh.

Gambar 3. 18 Perancangan Tampilan Dashboard Role Admin

3.3.5.7 Perancangan Tampilan Halaman Mobil Role Admin

Halaman data mobil dalam admin Yono Mobilindo digunakan untuk mengelola daftar mobil yang terdaftar dalam sistem. Antarmuka ini dilengkapi dengan fitur filter data yang memudahkan admin dalam melakukan pengawasan dan pengolahan data kendaraan.

Gambar 3. 19 Perancangan Tampilan Halaman Mobil Role Admin

3.3.5.8 Perancangan Tampilan Halaman Update Status Mobil Role Admin

Gambar 3. 20 Perancangan Tampulan Update Status Mobi Role Admin

3.3.5.9 Perancangan Tampilan Halaman Tambah Mobil Role Admin

Halaman *create* mobil merupakan bagian dari modul manajemen kendaraan dalam sistem Yono Mobilindo, yang digunakan oleh admin untuk menambahkan stok mobil baru ke dalam sistem.

Gambar 3. 21 Perancangan Tampilan Halaman Tambah Mobil Role Admin

3.3.5.10 Perancangan Tampilan Halaman Show Mobil Role Admin

Halaman car detail digunakan untuk menampilkan informasi lengkap mengenai satu unit mobil tertentu dalam sistem Yono Mobilindo.

Gambar 3. 22 Perancangan Tampilan Halaman Show Mobil Role Admin

3.3.5.11 Perancangan Tampilan Halaman Edit Mobil Role Admin

Halaman edit stock car pada sistem Yono Mobilindo digunakan oleh admin showroom untuk memperbarui informasi kendaraan yang sudah terdaftar di dalam sistem.

Gambar 3. 23 Perancangan Tampilan Halaman Edit Mobil Role Admin

3.3.5.12 Perancangan Tampilan Halaman Down Payment Role Admin

Halaman *down payment* (DP) dalam sistem Yono Mobilindo berfungsi sebagai pusat pengelolaan data pembayaran uang muka yang dilakukan oleh pengguna terhadap kendaraan yang diminati.

Gambar 3. 24 Perancangan Tampilan Halaman Down Payment Role Admin

3.3.5.13 Perancangan Tampilan Halaman Show Down Payment Role Admin

Halaman detail *down payment* (DP) menampilkan informasi lengkap mengenai transaksi pembayaran uang muka oleh pengguna terhadap unit mobil yang dipilih. Tampilan ini muncul dalam bentuk jendela detail atau *pop-up* yang ditujukan untuk kebutuhan verifikasi dan peninjauan cepat oleh admin.

Gambar 3. 25 Perancangan Tampilan Halaman Show Down Payment Role Admin

3.3.5.14 Perancangan Tampilan Halaman Refund Down Payment Role Admin

Halaman *refund down payment* (DP) digunakan oleh admin untuk memproses pengembalian dana kepada pengguna yang batal melanjutkan transaksi pembelian mobil.

Gambar 3. 26 Perancangan Tampilan Halaman Refund Down Payment Role Admin

3.3.5.15 Perancangan Tampilan Halaman Edit Refund Down Payment Role Admin

Halaman *edit refund down payment* (DP) ini dirancang untuk memungkinkan admin mengelola pengembalian dana bagi pengguna yang membatalkan pembelian mobil. Admin dapat melakukan perubahan pada nomor rekening penerima *refund* dan mengunggah bukti pembayaran yang relevan sebelum mengonfirmasi pengembalian dana.

Gambar 3. 27 Perancangan Tampilan Halaman Edit Refund Down Payment Role Admin

3.3.5.16 Perancangan Tampilan Halaman Catatan Transaksi Pembelian Role Admin

Halaman catatan transaksi pembelian ini digunakan oleh admin *showroom* untuk mencatat dan mengelola transaksi pembelian mobil. Setiap entri mencakup informasi seperti nomor transaksi, tanggal pembuatan, mobil yang dibeli, penjual, pembeli, harga jual, dan tanggal penjualan. Admin dapat menambah data baru melalui tombol *create* dan mengelola data yang sudah ada melalui kolom *action* yang memungkinkan pengeditan atau penghapusan transaksi.

Gambar 3. 28 Perancangan Tampilan Catatan Transaksi Pembelian Role Admin

3.3.5.17 Perancangan Tampilan Halaman Tambah Transaksi Pembelian Role Admin

Halaman *create* transaksi pembelian memungkinkan admin untuk memasukkan data transaksi baru, termasuk informasi tentang penjual, pembeli, mobil yang dibeli, harga jual, serta tanggal dan waktu penjualan. Admin dapat memilih data yang relevan melalui *dropdown* yang tersedia, dan setelah mengisi seluruh informasi yang diperlukan, data dapat disubmit untuk dicatat dalam sistem. Formulir ini memastikan bahwa setiap transaksi tercatat secara sistematis dan dapat dikelola dengan mudah.

Gambar 3. 29 Perancangan Tampilan Halaman Tambah Transaksi Pembelian Role Admin

3.3.5.18 Perancangan Tampilan Halaman Permintaan Penjualan Role Admin

Halaman permintaan penjualan digunakan oleh admin untuk mengelola permintaan penjualan mobil yang masuk. Di halaman ini, admin dapat memfilter permintaan berdasarkan tanggal pembuatan, status permintaan, dan nama mobil yang diajukan untuk dijual. Setiap entri menampilkan informasi seperti nomor permintaan, tanggal pembuatan, mobil yang diajukan, harga penawaran, tanggal inspeksi, serta status permintaan yang dapat diperbarui. Admin dapat melakukan tindakan lebih lanjut melalui kolom *action* untuk mengelola setiap permintaan penjualan yang ada.

Gambar 3. 30 Perancangan Tampilan Halaman Permintaan Penjualan Role Admin

3.3.5.19 Perancangan Tampilan Halaman Update Status Permintaan Penjualan Role Admin

Halaman ini digunakan oleh admin untuk memperbarui status permintaan penjualan mobil. Melalui formulir ini, admin dapat memilih status baru untuk permintaan penjualan yang sedang diproses.

Gambar 3. 31 Perancangan Tampilan Jalaman Update Status Permintaan Penjualan Role Admin

3.3.5.20 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role Admin

Halaman detail penjualan menampilkan informasi lengkap tentang permintaan penjualan mobil tertentu. Di halaman ini, admin dapat melihat rincian seperti nomor permintaan, status permintaan, nama dan kontak pembeli, harga mobil, tanggal pembelian, serta lokasi inspeksi.

Gambar 3. 32 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role Admin

3.3.5.21 Perancangan Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin

Halaman ini memungkinkan admin untuk menambahkan catatan tambahan terkait permintaan penjualan mobil. Admin dapat memasukkan informasi penting atau komentar terkait transaksi ke dalam kolom yang tersedia.

Gambar 3. 33 Perancangan Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin

3.3.5.22 Perancangan Tampilan Halaman Catatan Transaksi Penjualan Role Admin

Halaman catatan transaksi penjualan ini digunakan untuk mencatat setiap transaksi penjualan mobil yang dilakukan. Terdapat tabel yang menampilkan nomor transaksi, tanggal transaksi, mobil yang terjual, penjual, pembeli, harga jual, dan tanggal penjualan. Admin dapat menambah transaksi baru melalui tombol *Create*.

Gambar 3. 34 Perancangan Tampilan Halaman Catatan Transaksi Penjualan Role Admin

3.3.5.23 Perancangan Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin

Halaman create transaksi penjualan ini memungkinkan admin untuk mencatat transaksi penjualan mobil yang baru. Admin dapat mengisi informasi terkait pembeli, penjual, penawaran yang dipilih, harga jual, serta tanggal dan waktu penjualan.

Gambar 3. 35 Perancangan Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin

3.3.5.24 Perancangan Tampilan Dashboard Role User

Halaman *dashboard* pada panel *user* Yono Mobilindo digunakan untuk menampilkan ringkasan transaksi pembelian dan penjualan mobil, informasi *down payment* (DP) yang masih tertunda, serta daftar permintaan penjualan yang telah disetujui. Antarmuka ini memudahkan pengguna memeriksa detail transaksi, jadwal janji temu, lokasi *showroom*, dan status permintaan secara terstruktur.

Gambar 3. 36 Perancangan Tampilan Dashboard Role User

3.3.5.25 Perancangan Tampilan Halaman Down Payment Role User

Halaman *down payment* (DP) ini digunakan untuk memonitor pembayaran uang muka pada transaksi pembelian mobil. Tabel yang ada menampilkan informasi seperti nomor pembayaran, mobil yang dibeli, jumlah DP yang dibayarkan, tanggal pembayaran, serta status pembayaran.

Gambar 3. 37 Perancangan Tampilan Halaman Down Payment Role User

3.3.5.26 Perancangan Tampilan Halaman Bayar Sekarang Down Payment Role User

Halaman ini menampilkan rincian metode pembayaran untuk transaksi pembayaran *down payment* (DP). Berbagai opsi pembayaran yang tersedia, seperti transfer melalui bank atau pembayaran menggunakan *e-wallet*, dapat dipilih oleh pengguna.

Gambar 3. 38 Perancangan Tampilan Halaman Bayar Sekarang Down Payment Role User

3.3.5.27 Perancangan Tampilan Halaman Detail Bayar Down Payment Role User

Halaman ini menampilkan detail pembayaran *down payment* (DP) untuk transaksi pembelian mobil. Pengguna dapat melihat informasi lengkap mengenai transaksi, termasuk nomor transaksi, jenis mobil yang dibeli, informasi pembeli, serta detail pembayaran DP. Pengguna juga dapat melihat status pembayaran, dan alamat *showroom*. Setelah memverifikasi informasi, pengguna dapat melanjutkan proses pembayaran dengan menekan tombol bayar sekarang.

Gambar 3. 39 Perancangan Tampilan Halaman Detail Bayar Down Payment Role User

3.3.5.28 Perancangan Tampilan Halaman Detail Down Payment Role User

Halaman ini menunjukkan rincian lengkap mengenai pembayaran *down payment* (DP) untuk transaksi pembelian mobil. Informasi yang ditampilkan mencakup nomor pembayaran, mobil yang dibeli, data pembeli, dan status pembayaran.

Gambar 3. 40 Perancangan Tampilan Halaman Detail Down Payment Role User

3.3.5.29 Perancangan Tampilan Halaman Transaksi Pembelian Role User

Halaman transaksi pembelian mobil ini digunakan untuk memantau dan mengelola transaksi pembelian mobil. Tabel yang ada menampilkan informasi penting seperti nomor transaksi, tanggal transaksi, mobil yang dibeli, penjual, pembeli, harga jual, serta tanggal penjualan.

Gambar 3. 41 Perancangan Tampilan Halaman Transaksi Pembelian Role User

3.3.5.30 Perancangan Tampilan Halaman Testimoni Role User

Halaman testimoni transaksi ini memungkinkan pengguna untuk mencatat umpan balik atau testimoni setelah melakukan transaksi pembelian mobil. Di halaman ini, pengguna dapat mengisi komentar mengenai pengalaman transaksi, memberikan rating transaksi dalam skala 1-5, serta mengunggah foto *review* dari pengguna.

Gambar 3. 42 Perancangan Tampilan Halaman Testimoni Role User

3.3.5.31 Perancangan Tampilan Halaman Permintaan Penjualan Role User

Halaman permintaan penjualan mobil ini menampilkan informasi terkait penjualan mobil yang sedang diproses. Tabel ini mencakup nomor penjualan, mobil yang dijual, harga penawaran, tanggal penjualan, serta status permintaan (misalnya pending).

Gambar 3. 43 Perancangan Tampilan Halaman Permintaan Penjualan Role User

3.3.5.32 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role User

Halaman detail permintaan penjualan mobil ini menampilkan informasi rinci mengenai transaksi penjualan mobil tertentu. Informasi yang disediakan meliputi nomor penjualan, status penjualan, data pembeli, mobil yang dibeli, harga penawaran, serta lokasi *showroom* tempat transaksi dilakukan

Gambar 3. 44 Perancangan Tampilan Halaman Detail Permintaan Penjualan Role User

3.3.5.33 Perancangan Tampilan Halaman Update Alamat Role User

Halaman ini memungkinkan pengguna untuk memperbarui atau melengkapi informasi alamat mereka. Pengguna dapat memasukkan alamat lengkap pada kolom yang disediakan. Setelah alamat terisi, pengguna dapat menyimpan perubahan dengan menekan tombol *submit*, atau membatalkan perubahan dengan tombol batal.

Gambar 3. 45 Perancangan Tampilan Halaman Update Alamat Role User

3.3.5.34 Perancangan Tampilan Halaman Transaksi Penjualan Role User

Halaman transaksi penjualan mobil ini menunjukkan daftar transaksi penjualan mobil yang telah diselesaikan. Tabel ini mencakup informasi seperti nomor transaksi, jenis mobil yang dijual, pembeli, harga jual, tanggal transaksi, dan status transaksi.

Gambar 3. 46 Perancangan Tampilan Halaman Transaksi Penjualan Role User

3.4 Implementasi

3.4.1 Install Laravel

Langkah pertama dalam memulai pengembangan aplikasi Laravel adalah melakukan instalasi Laravel Installer melalui Composer. Composer merupakan *dependency* manager untuk PHP yang digunakan secara luas dalam ekosistem Laravel. Pada tahap ini, perintah pada Gambar 3.47 berikut dijalankan pada terminal:

```
1   composer global require laravel/installer
```

Gambar 3. 47 Install Laravel Via Composer

Setelah laravel installer berhasil diinstal, tahap selanjutnya adalah membuat proyek laravel baru dengan menggunakan perintah pada Gambar 3.48 berikut:

```
1   laravel new example-app
```

Copy to Clipboard

Gambar 3. 48 Membuat Proyek Laravel Baru

Setelah proyek Laravel berhasil dibuat, langkah berikutnya adalah masuk ke direktori proyek dan menjalankan beberapa perintah untuk mempersiapkan dependensi *front-end* serta memulai *environment* pengembangan. Berikut urutan perintah pada gambar Gambar 3.49 yang digunakan:

```
1   cd example-app
```

Gambar 3. 49 Direktori Proyek Laravel

Perintah ini menjalankan dua proses sekaligus. Pertama, npm install akan menginstal seluruh dependensi Node.js yang diperlukan oleh Laravel untuk pengelolaan asset seperti JavaScript, CSS, dan lainnya. Selanjutnya, npm run build

akan membangun (compile) file-file front-end berdasarkan konfigurasi yang ada, sehingga siap digunakan dalam tampilan antarmuka aplikasi. Perintah pada Gambar 3.50 dapat diketik di terminal:

```
2    npm install && npm run build
```

Gambar 3. 50 Install Dependensi dan Kompilasi Aset front-end Laravel

Perintah ini digunakan untuk menjalankan perintah pengembangan (development script) yang didefinisikan di dalam file composer.json, biasanya untuk menjalankan Laravel Vite secara lokal agar dapat melihat hasil pengembangan secara real-time. Perintah yang ditampilkan pada Gambar 3.51 dapat dieksekusi di terminal:

```
3    composer run dev
```

Gambar 3. 51 Composer Run Dev

3.4.2 Install Laravel UI

Untuk menambahkan antarmuka pengguna (*User Interface*) dasar pada aplikasi Laravel, dapat digunakan package tambahan bernama Laravel UI**.** *Package* ini menyediakan *scaffolding* untuk *authentication* dasar menggunakan teknologi *front-end* seperti Bootstrap. Perintah pada Gambar 3.52 berikut digunakan untuk menambahkan *package* Laravel UI ke dalam proyek:

```
composer require laravel/ui
```

Gambar 3. 52 Install Laravel UI

Perintah ini digunakan untuk menghasilkan *scaffolding* antarmuka pengguna berbasis Bootstrap sekaligus menambahkan fitur *authentication*. Setelah package pada Gambar 3.52 ini terinstal, dapat melanjutkan dengan menjalankan perintah tambahan seperti Gambar 3.53 berikut:

```
php artisan ui bootstrap --auth
```

Gambar 3. 53 Install Scaffolding Antarmuka Bootsrap

Langkah selanjutnya pada Gambar 3.54 mengelola dependensi *front-end* menggunakan *Node Package Manager* (NPM). Hal ini bertujuan untuk memastikan bahwa seluruh komponen antarmuka seperti Bootstrap, jQuery, dan file pendukung lainnya dapat berjalan dengan baik.

```
npm install && npm run dev
```

Gambar 3. 54 Install Dependensi dan Kompilasi Aset front-end Laravel UI

3.4.3 Install Midtrans

Pada Gambar 3.55 ditunjukkan perintah pada terminal untuk menambahkan dependensi Midtrans PHP melalui composer. Perintah ini digunakan untuk menginstal pustaka Midtrans yang memungkinkan integrasi sistem pembayaran Midtrans pada aplikasi berbasis PHP atau Laravel.

```
* History restored
tarmidzibariq@MacBook-Air-muhammad app % composer require midtrans/midtrans-php
```

Gambar 3. 55 Install Midtrans Via Composer

Pada Gambar 3.56 tersebut memperlihatkan file konfigurasi Midtrans pada Laravel yang memuat *server key*, *client key*, dan pengaturan mode produksi. Nilai-nilai tersebut diambil dari variabel .env agar dapat diatur dengan aman tanpa disimpan langsung pada kode sumber. Langkah kedua membuat file config/midtrans.php :

```php
<?php

return [
    'server_key' => env('MIDTRANS_SERVER_KEY'),
    'client_key' => env('MIDTRANS_CLIENT_KEY'),
    'is_production' => env('MIDTRANS_IS_PRODUCTION', false),
];

```

Gambar 3. 56 Membuat File config/midtrans.php

Pada Gambar 3.57 ditunjukkan konfigurasi variabel environment Midtrans pada berkas .env, yang berfungsi untuk menyimpan server key, client key, dan pengaturan mode produksi. Konfigurasi ini mempermudah pengelolaan kredensial dengan cara yang aman dan terpisah dari kode sumber. Tambahkan kode berikut pada berkas .env:

```
MIDTRANS_SERVER_KEY=server-key-anda
MIDTRANS_CLIENT_KEY=client-key-anda
MIDTRANS_IS_PRODUCTION=false
```

Gambar 3. 57 Variable Environtment Midtrans

Pada Gambar 3.58 menampilkan kode kelas MidtransHelper pada Laravel yang digunakan untuk menginisialisasi konfigurasi Midtrans. Metode statik init berfungsi untuk mengatur server key, mode produksi, serta pengaturan validasi dan 3D Secure agar integrasi pembayaran Midtrans dapat digunakan secara konsisten di seluruh aplikasi. Kode tersebut dituliskan pada berkas app/Helpers/MidtransHelper.php :

```php
<?php

namespace App\Helpers;

class MidtransHelper
{
    public static function init()
    {
        \Midtrans\Config::$serverKey = env('MIDTRANS_SERVER_KEY');
        \Midtrans\Config::$isProduction = env('MIDTRANS_IS_PRODUCTION', false);
        \Midtrans\Config::$serverKey = config('midtrans.server_key');
        \Midtrans\Config::$isProduction = config('midtrans.is_production');
        \Midtrans\Config::$isSanitized = true;
        \Midtrans\Config::$is3ds = true;
    }
}
```

Gambar 3. 58 Kode MidtransHelper.php

3.4.4 Integrasi GitHub dalam Pengelolaan Kode

Pada Gambar 3.57 tersebut memperlihatkan tampilan halaman *repository* di GitHub dengan tombol *new* yang berfungsi untuk membuat *repository* baru. Tombol ini digunakan untuk memulai penyimpanan dan pengelolaan kode sumber di GitHub.

Gambar 3. 59 Tombol New Repository Github

Pada Gambar 3.58 tersebut menampilkan antarmuka GitHub untuk membuat repository baru. Pengguna diminta mengisi nama pemilik, nama *repository*, deskripsi opsional, serta menentukan tingkat visibilitas *repository*, yaitu *public* atau *private*. Halaman ini digunakan sebagai langkah awal dalam pembuatan *repository* untuk pengelolaan kode sumber.

Gambar 3. 60 Membuat Repository Baru

Pada Gambar 3.59 menampilkan antarmuka GitHub pada menu *Code* untuk menyalin URL *repository*. Pengguna dapat memilih metode *clone* melalui HTTPS, SSH, atau GitHub CLI.

Gambar 3. 61 Salin URL Repository

Pada Gambar 3.60 tersebut menampilkan perintah terminal git init yang digunakan untuk menginisialisasi *repository* Git lokal pada direktori proyek. Perintah ini membuat struktur kontrol versi di dalam folder kerja sehingga memungkinkan pengelolaan perubahan kode sumber secara teratur dan terdokumentasi.

```
tarmidzibariq@MacBook-Air-muhammad app %
 * History restored

tarmidzibariq@MacBook-Air-muhammad app % git init
```

Gambar 3. 62 Perintah Git Init

Pada Gambar 3.61 tersebut menampilkan perintah yang digunakan untuk menambahkan seluruh perubahan berkas di direktori kerja. Perintah ini mempersiapkan berkas-berkas tersebut untuk dicatat pada commit selanjutnya, sehingga memastikan semua perubahan tersimpan dalam riwayat versi.

```
tarmidzibariq@MacBook-Air-muhammad app % git add .
```

Gambar 3. 63 Perintah Git Add

Pada Gambar 3.62 tersebut menampilkan perintah yang digunakan untuk menambahkan alamat *remote repository* pada GitHub ke repository lokal. Perintah ini mengaitkan repository lokal dengan repository *remote* bernama *origin*, sehingga memudahkan proses *push* dan *pull* antara keduanya.

```
tarmidzibariq@MacBook-Air-muhammad app % git remote add origin https://github.com/tarmidzibariq/ju
albelimobil_yonomobilindo.git
```

Gambar 3. 64 Perintah Git Remote

Pada Gambar 3.63 tersebut menampilkan perintah terminal git branch -M main dan git push -u origin main. Perintah pertama digunakan untuk menetapkan atau mengganti nama cabang utama menjadi *main*. Perintah kedua digunakan untuk mengirim (*push*) cabang *main* ke *remote repository* bernama *origin*, sekaligus mengatur pelacakan cabang *upstream* agar mempermudah sinkronisasi pada *push* atau *pull* berikutnya.

```
git branch -M main
git push -u origin main
```

Gambar 3. 65 Perintah git branch -M main dan git push -u origin main

3.5 Tampilan Website

3.5.1 Tampilan Halaman Utama

Halaman ini merupakan tampilan depan dari situs web *showroom* mobil, yang menyediakan informasi tentang mobil yang tersedia untuk dijual. Pengguna dapat mencari mobil berdasarkan model, jenis, dan tahun melalui kolom pencarian di bagian atas. Di bawahnya, terdapat daftar mobil yang tersedia beserta harga dan gambar mobil tersebut. Selain itu, terdapat bagian apa kata pelanggan kami, yang menampilkan testimoni dari pembeli sebelumnya, termasuk foto pelanggan yang telah membeli mobil. Di bagian bawah, terdapat informasi lebih lanjut mengenai *showroom*, termasuk testimonial dari pemilik *showroom*.

Gambar 3. 66 Tampilan Halaman Utama

3.5.2 Tampilan Halaman Jual Mobil

Halaman ini merupakan formulir untuk menjual mobil di situs web *showroom*. Pengguna diminta untuk mengisi informasi terkait mobil yang akan dijual, seperti brand, model, tahun, dan harga. Selain itu, pengguna juga diminta untuk memberikan detail inspeksi kendaraan dan lokasi inspeksi yang akan dilakukan.

Gambar 3. 67 Tampilan Halaman Jual Mobil

3.5.3 Tampilan Halaman Masuk

Halaman ini menampilkan formulir masuk untuk pengguna yang ingin mengakses akun mereka di situs Yono Mobilindo. Pengguna diminta untuk memasukkan alamat email dan kata sandi untuk masuk.

Gambar 3. 68 Tampilan Halaman Masuk

3.5.4 Tampilan Halaman Daftar

Halaman ini adalah formulir pendaftaran untuk pengguna baru yang ingin membuat akun di situs Yono Mobilindo. Pengguna diminta untuk mengisi informasi pribadi, termasuk nama lengkap, email, kata sandi, konfirmasi kata sandi, dan nomor telepon.

Gambar 3. 69 Tampilan Halaman Daftar

3.5.5 Tampilan Halaman Testimoni

Halaman ini menampilkan koleksi testimoni pelanggan yang telah membeli mobil dari *showroom*. Setiap testimonial dilengkapi dengan foto pelanggan yang menunjukkan kepuasan mereka setelah membeli mobil, serta rating bintang (1-5) untuk menggambarkan pengalaman mereka.

Gambar 3. 70 Tampilan Halaman Testimoni

3.5.6 Tampilan Halaman Dashboard Role Admin

Halaman *dashboard* pada panel admin Yono Mobilindo digunakan untuk menampilkan ringkasan status transaksi, termasuk jumlah *down payment* (DP) terkonfirmasi, permintaan penjualan yang pending, dan mobil yang telah terjual. Antarmuka ini dilengkapi grafik total dan jumlah transaksi untuk analisis visual, serta kalender jadwal appointment yang memudahkan pemantauan agenda pertemuan dengan pelanggan.

Gambar 3. 71 Tampilan Halaman Dashboard Role Admin

3.5.7 Tampilan Halaman Mobil Role Admin

Halaman ini menampilkan daftar mobil yang tersedia di *showroom*, lengkap dengan informasi mengenai merek, model, harga, dan status penjualannya. Admin dapat menggunakan filter untuk mencari mobil berdasarkan merek, tipe, atau status penjualan.

Gambar 3. 72 Tampilan Halaman Mobil Role Admin

3.5.8 Tampilan Halaman Update Status Mobil Role Admin

Halaman ini memungkinkan admin untuk memperbarui status mobil yang ada di *showroom*.

Gambar 3. 73 Tampilan Halaman Update Status Mobil Role Admin

3.5.9 Tampilan Halaman Tambah Mobil Role Admin

Halaman ini digunakan untuk menambahkan mobil baru ke dalam sistem inventaris *showroom*. Admin dapat mengisi berbagai informasi terkait mobil yang akan dijual, seperti merek, model, harga, deskripsi, tipe bahan bakar, jarak tempuh, warna, dan detail lainnya.

Gambar 3. 74 Tampilan Halaman Tambah Mobil Role Admin

3.5.10 Tampilan Halaman Show Mobil Role Admin

Halaman ini menunjukkan detail lengkap mobil yang tersedia di showroom. Informasi yang ditampilkan meliputi nomor mobil, merek, model, tahun, harga, transmisi, deskripsi, tipe bahan bakar, jarak tempuh, sejarah servis, dan data lainnya seperti warna, kapasitas mesin, serta apakah mobil tersebut dilengkapi dengan buku manual dan kunci cadangan.

Gambar 3. 75 Tampilan Halaman Show Mobil Role Admin

3.5.11 Tampilan Halaman Edit Mobil Role Admin

Halaman ini digunakan untuk mengedit informasi tentang mobil yang tersedia di *showroom*. Admin dapat memperbarui detail mobil, termasuk merek, model, tahun, harga, deskripsi, jenis bahan bakar, warna, dan informasi lainnya.

Gambar 3. 76 Tampilan Halaman Edit Mobil Role Admin

3.5.12 Tampilan Halaman Down Payment Role Admin

Halaman *Down Payment* (DP) ini digunakan untuk mengelola pembayaran uang muka dari pembeli yang bertransaksi di showroom.

Gambar 3. 77 Tampilan Halaman Down Payment Role Admin

3.5.13 Tampilan Halaman Show Down Payment Role Admin

Halaman ini menampilkan rincian detail untuk pembayaran *down payment* (DP) yang sedang diproses. Dalam tampilan ini, admin dapat melihat informasi lengkap mengenai transaksi DP, seperti nomor transaksi, nama pembeli, mobil yang dibeli, tanggal janji temu, lokasi *showroom*, serta jumlah pembayaran yang dilakukan.

Gambar 3. 78 Tampilan Halaman Show Down Payment Role Admin

3.5.14 Tampilan Halaman Add Refund Down Payment Role Admin

Halaman ini digunakan untuk mengelola proses pengembalian dana terkait pembayaran *down payment* (DP) dari pembeli. Admin dapat mengisi informasi terkait pengembalian dana, termasuk nomor rekening pengguna yang akan menerima pengembalian dana, serta mengunggah bukti pembayaran terkait.

Gambar 3. 79 Tampilan Halaman Add Refund Down Payment Role Admin

3.5.15 Tampilan Halaman Edit Refund Down Payment Role Admin

Halaman ini digunakan untuk mengedit informasi terkait pengembalian dana pada pembayaran *down payment* (DP). Admin dapat memperbarui nomor rekening pengguna yang menerima pengembalian dana serta mengunggah bukti pembayaran berupa gambar atau dokumen yang relevan.

Gambar 3. 80 Tampilan Halaman Edit Refund Down Payment Role Admin

3.5.16 Tampilan Halaman Catatan Transaksi Pembelian Role Admin

Halaman catatan transaksi pembelian ini berfungsi untuk mencatat setiap transaksi pembelian mobil yang dilakukan oleh pembeli.

Gambar 3. 81 Tampilan Halaman Catatan Transaksi Pembelian Role Admin

3.5.17 Tampilan Halaman Tambah Transaksi Pembelian Role Admin

Halaman catatan transaksi pembelian ini memungkinkan admin untuk mencatat transaksi pembelian mobil yang dilakukan. Admin dapat memilih penjual, pembeli, mobil yang dibeli, serta harga jualnya.

Gambar 3. 82 Tampilan Halaman Tambah Transaksi Pembelian Role Admin

3.5.18 Tampilan Halaman Permintaan Penjualan Role Admin

Halaman permintaan penjualan ini memungkinkan admin untuk mengelola permintaan penjualan mobil. Tabel yang ada menampilkan informasi terkait setiap permintaan, termasuk nomor permintaan, nama pembeli, mobil yang diminta, harga penawaran, dan tanggal inspeksi.

Gambar 3. 83 Tampilan Halaman Permintaan Penjualan Role Admin

3.5.19 Tampilan Halaman Update Status Permintaan Penjualan Role Admin

Halaman ini digunakan untuk memperbarui status permintaan penjualan mobil. Dalam dialog perbarui status permintaan penjualan, admin dapat memilih status baru dari daftar yang tersedia, seperti *pending*, *accepted*, *rejected*, atau *sold*, untuk menunjukkan status permintaan yang sedang dalam proses.

Gambar 3. 84 Tampilan Halaman Update Status Permintaan Penjualan Role Admin

3.5.20 Tampilan Halaman Detail Permintaan Penjualan Role Admin

Halaman ini menampilkan rincian detail penjualan mobil, yang berisi informasi lengkap tentang permintaan penjualan mobil tertentu. Admin dapat melihat data seperti nomor permintaan, nama pembeli, mobil yang diminta, harga penawaran, tanggal janji, dan lokasi inspeksi.

Gambar 3. 85 Tampilan Halaman Detail Permintaan Penjualan Role Admin

3.5.21 Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin

Halaman ini memungkinkan admin untuk menambahkan catatan tambahan terkait permintaan penjualan mobil. Dalam dialog tambah catatan *showroom,* admin dapat memasukkan informasi atau komentar yang relevan mengenai transaksi atau status permintaan tersebut.

Gambar 3. 86 Tampilan Halaman Tambah Catatan Permintaan Penjualan Role Admin

3.5.22 Tampilan Halaman Catatan Transaksi Penjualan Role Admin

Halaman catatan transaksi penjualan ini menampilkan daftar transaksi penjualan mobil yang telah dilakukan. Setiap entri mencakup informasi seperti tanggal transaksi, mobil yang dijual, penjual, pembeli, harga jual, dan tanggal penjualan.

Gambar 3. 87 Tampilan Halaman Catatan Transaksi Penjualan Role Admin

3.5.23 Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin

Halaman ini memungkinkan admin untuk mencatat transaksi penjualan mobil baru. Admin dapat memilih pembeli, penjual, dan penawaran yang telah disepakati, serta memasukkan harga jual dan tanggal transaksi.

Gambar 3. 88 Tampilan Halaman Tambah Catatan Transaksi Penjualan Role Admin

3.5.24 Tampilan Halaman Dashboard Role User

Halaman *dashboard* pada panel *user* Yono Mobilindo digunakan untuk menampilkan ringkasan transaksi pembelian dan penjualan mobil, detail *down payment* (DP) yang masih tertunda, serta daftar permintaan penjualan yang telah diterima. Antarmuka ini memudahkan pengguna memeriksa informasi pembayaran, jadwal janji temu, lokasi *showroom*, dan status permintaan penjualan secara terstruktur dan mudah diakses.

Gambar 3. 89 Tampilan Halaman Dashboard Role User

3.5.25 Tampilan Halaman Down Payment Role User

Halaman pembayaran *down payment* (DP) ini digunakan untuk mengelola pembayaran DP untuk transaksi pembelian mobil. Tabel ini menampilkan informasi terkait pembayaran DP, termasuk tanggal pembayaran, jumlah DP, mobil yang dibeli, serta status transaksi.

Gambar 3. 90 Tampilan Halaman Down Payment Role User

3.5.26 Tampilan Halaman Detail Bayar Down Payment Role User

Halaman detail pembayaran *down payment* (DP) ini menampilkan informasi rinci terkait pembayaran DP untuk pembelian mobil. Data yang ditampilkan meliputi nomor pembayaran, nama pembeli, mobil yang dibeli, jumlah DP yang dibayar, serta tanggal pembayaran.

Gambar 3. 91 Tampilan Halaman Detail Bayar Down Payment Role User

3.5.27 Tampilan Halaman Bayar Sekarang Down Payment Role User

Halaman ini menampilkan opsi pembayaran untuk transaksi *down payment* (DP). Pengguna dapat memilih berbagai metode pembayaran, seperti transfer melalui rekening bank BCA, kartu kredit, atau menggunakan metode pembayaran digital seperti virtual account.

Gambar 3. 92 Tampilan Halaman Bayar Sekarang Down Payment Role User

3.5.28 Tampilan Halaman Detail Down Payment Role User

Halaman detail *down payment* (DP) ini menampilkan informasi lengkap mengenai pembayaran DP yang telah dilakukan oleh pembeli. Pengguna dapat melihat nomor pembayaran, nama pembeli, mobil yang dibeli, jumlah pembayaran yang dilakukan, serta tanggal pembayaran.

Gambar 3. 93 Tampilan Halaman Detail Down Payment Role User

3.5.29 Tampilan Halaman Transaksi Pembelian Role User

Halaman transaksi pembelian mobil ini menampilkan daftar transaksi pembelian mobil yang telah dilakukan. Tabel ini menunjukkan informasi seperti nomor transaksi, mobil yang dibeli, penjual, harga jual, tanggal transaksi, serta status pembayaran.

Gambar 3. 94 Tampilan Halaman Transaksi Pembelian Role User

3.5.30 Tampilan Halaman Testimoni Role User

Halaman ini memungkinkan pengguna untuk memberikan testimoni setelah transaksi pembelian mobil selesai. Pengguna dapat memasukkan komentar mengenai pengalaman transaksi, memberikan rating (dari 1 hingga 5 bintang), dan mengunggah foto *review* jika ada.

Gambar 3. 95 Tampilan Halaman Testimoni Role User

3.5.31 Tampilan Halaman Permintaan Penjualan Role User

Halaman permintaan penjualan mobil ini menampilkan daftar mobil yang diajukan untuk dijual ke *showroom*. Setiap entri mencakup informasi penting seperti nomor permintaan, jenis mobil, penawaran harga, serta tanggal permintaan penjualan.

Gambar 3. 96 Tampilan Halaman Permintaan Penjualan Role User

3.5.32 Tampilan Halaman Detail Permintaan Penjualan role User

Halaman detail permintaan penjualan mobil ini memberikan rincian lengkap tentang permintaan penjualan mobil yang diajukan ke *showroom*.

Gambar 3. 97 Tampilan Halaman Detail Permintaan Penjualan role User

3.5.33 Tampilan Halaman Update Alamat Role User

Halaman ini memungkinkan pengguna untuk memperbarui atau menambahkan alamat tambahan pada rincian permintaan penjualan mobil.

Gambar 3. 98 Tampilan Halaman Update Alamat Role User

3.5.34 Tampilan Halaman Transaksi Penjualan Role User

Halaman transaksi penjualan mobil ini menampilkan informasi terkait transaksi penjualan mobil yang telah terjadi. Setiap transaksi mencakup nomor transaksi, tanggal transaksi, mobil yang dijual, penjual, harga jual, dan tanggal transaksi.

Gambar 3. 99 Tampilan Halaman Transaksi Penjualan Role User

3.6 Hosting

3.7 Uji Coba

Uji coba sistem menggunakan metode *black box testing* untuk memverifikasi bahwa semua fungsi berjalan sesuai spesifikasi tanpa melihat struktur internal kode. Pengujian dilakukan dengan memberikan masukan tertentu dan memeriksa keluaran yang dihasilkan. Seluruh fitur, baik untuk pengguna maupun admin, diuji untuk memastikan kesesuaian dengan kebutuhan sistem.

Tabel 3. 13 Uji Coba

No	Fitur yang Diuji	Input/Uji Coba	Output yang Diharapkan	Hasil Uji	Status
1	Filter	Brand, Model, Tahun, Jarak Tempuh	Menampilkan mobil sesuai kriteria filter	Output sesuai	Lulus
2	Informasi Mobil Tersedia	-	Menampilkan daftar mobil yang tersedia	Output sesuai	Lulus

3	Informasi Testimonial	-	Menampilkan ulasan dan rating pembeli sebelumnya	Output sesuai	Lulus
4	Detail Mobil	-	Menampilkan spesifikasi, riwayat servis, kondisi, dan foto mobil	Output sesuai	Lulus
5	Form Pembayaran DP Mobil	Tanggal Janji, Waktu Janji, Jumlah DP	Data pembayaran DP tersimpan dan terverifikasi	Output sesuai	Lulus
6	Form Jual Mobil	Brand, Model, Tahun, Jarak Tempuh, Harga, Lokasi	Detail mobil untuk dijual tersimpan dalam sistem	Output sesuai	Lulus
7	Informasi Kontak	-	Menampilkan informasi kontak showroom	Output sesuai	Lulus
8	Formulir Masuk	Email, Password	Pengguna berhasil login dengan kredensial valid	Output sesuai	Lulus
9	Formulir Daftar	Nama, Email, Password, No Telepon	Akun pengguna baru berhasil dibuat	Output sesuai	Lulus
User Management System, Role User					
10	Dashboard (User)	-	Menampilkan informasi dashboard pengguna	Output sesuai	Lulus
11	Informasi Data DownPayment (User)	-	Menampilkan daftar pembayaran DP	Output sesuai	Lulus
12	Bayar Sekarang (User)	Nomor Telepon, Nama, Jumlah DP	Melakukan pembayaran DP dengan status updated	Output sesuai	Lulus
13	Detail DownPayment	-	Menampilkan informasi detail pembayaran DP	Output sesuai	Lulus
14	Informasi Data Transaksi Pembelian Mobil	-	Menampilkan daftar transaksi pembelian mobil	Output sesuai	Lulus
15	Testimoni (User)	Komentar, Rating, Foto	Menyimpan dan menampilkan testimonial pengguna	Output sesuai	Lulus
16	Informasi Data Permintaan Penjualan Mobil	-	Menampilkan daftar permintaan penjualan mobil	Output sesuai	Lulus
17	Detail Permintaan Penjualan	-	Menampilkan informasi permintaan penjualan secara mendetail	Output sesuai	Lulus

18	Informasi Data Transaksi Penjualan Mobil	-	Menampilkan daftar transaksi penjualan mobil	Output sesuai	Lulus
User Management System, Role Admin					
19	Dashboard (Admin)	-	Menampilkan data dashboard admin	Output sesuai	Lulus
20	Informasi Data Mobil	-	Menampilkan daftar mobil yang tersedia di showroom	Output sesuai	Lulus
21	Tambah Mobil	Data lengkap mobil	Data mobil baru berhasil disimpan	Output sesuai	Lulus
22	Edit Mobil	Data mobil diperbarui	Informasi mobil berhasil diperbarui	Output sesuai	Lulus
23	Show Mobil	-	Menampilkan detail lengkap mobil	Output sesuai	Lulus
24	Hapus Mobil	-	Data mobil terhapus dari sistem	Output sesuai	Lulus
25	Perbarui Status Mobil	Status	Status mobil berhasil diperbarui	Output sesuai	Lulus
26	Filter Cars	Brand, Sale Type, Status	Menampilkan daftar mobil sesuai filter	Output sesuai	Lulus
27	Informasi Data DownPayment (Admin)	-	Menampilkan semua data pembayaran DP	Output sesuai	Lulus
28	Show DownPayment	-	Menampilkan detail pembayaran DP	Output sesuai	Lulus
29	Add Refund	No Rekening, Foto	Data pengembalian uang DP berhasil disimpan	Output sesuai	Lulus
30	Edit Refund	No Rekening, Foto	Data refund berhasil diperbarui	Output sesuai	Lulus
31	Filter DownPayment	Tanggal dibuat, Status, Brand, Model	Menyaring data pembayaran DP sesuai kriteria	Output sesuai	Lulus
32	Informasi Catatan Transaksi Pembelian	-	Menampilkan riwayat transaksi penjualan mobil	Output sesuai	Lulus
33	Tambah Catatan Transaksi Pembelian	Penjual, Pembeli, Mobil, Terjual Diharga, Tanggal	Data catatan transaksi berhasil disimpan	Output sesuai	Lulus
34	Hapus Catatan Transaksi Pembelian	-	Data catatan transaksi berhasil dihapus	Output sesuai	Lulus
35	Informasi Penjualan Mobil	-	Menampilkan daftar semua transaksi penjualan mobil	Output sesuai	Lulus

36	Show Penjualan Mobil	-	Menampilkan detail lengkap transaksi penjualan mobil	Output sesuai	Lulus
37	Perbarui Status Penjualan Mobil	Status	Status transaksi penjualan berhasil diperbarui	Output sesuai	Lulus
38	Informasi Data Catatan Transaksi Penjualan	-	Menampilkan data penawaran penjualan mobil sebelum transaksi final	Output sesuai	Lulus
39	Tambah Catatan Transaksi Penjualan	Penjual, Pembeli, Mobil, Terjual Diharga, Tanggal	Data catatan penawaran berhasil disimpan	Output sesuai	Lulus

BAB 4
PENUTUP

4.1 Kesimpulan

Penelitian ini bertujuan merancang dan mengimplementasikan sistem jual beli mobil bekas berbasis website untuk mengatasi masalah transparansi, keamanan, dan efisiensi transaksi. Hasil penelitian ini dapat disimpulkan sebagai berikut:

1. Sistem yang dikembangkan menawarkan fitur pencarian kendaraan, ulasan pembeli, pembayaran *down payment* (DP), dan formulir penjualan mobil, memberikan solusi yang efisien, aman, dan transparan bagi pembeli dan penjual.
2. Pengembangan sistem mengikuti model *System Development Life Cycle* (SDLC) tipe *waterfall* yang terstruktur, melalui tahapan yang jelas mulai dari analisis kebutuhan, desain, implementasi, pengujian, hingga pemeliharaan, untuk memastikan aplikasi yang dikembangkan sesuai dengan kebutuhan pengguna.
3. Sistem menggunakan PHP, Laravel, MySQL, Midtrans, dan Bootstrap untuk mempermudah transaksi dan mengintegrasikan pembayaran digital yang aman.
4. Pengujian sistem dengan metode *black box* memastikan fungsionalitas sesuai dengan kebutuhan pengguna tanpa memperhatikan aspek teknis internal sistem.

Sistem yang dikembangkan berhasil menghadirkan solusi bagi permasalahan yang ada di pasar jual beli mobil bekas dengan menawarkan *platform* yang lebih transparan, efisien, dan aman. Dengan fitur-fitur yang mendukung kemudahan transaksi, sistem ini diharapkan dapat meningkatkan kepercayaan antara pembeli dan penjual serta mempercepat proses jual beli kendaraan.

4.2 Saran

Untuk meningkatkan kinerja dan pengalaman pengguna pada sistem jual beli mobil bekas berbasis *website* yang telah dikembangkan, beberapa saran perbaikan dan pengembangan fungsionalitas dapat diterapkan. Berikut adalah beberapa usulan yang dapat dipertimbangkan untuk meningkatkan aplikasi ini:

1. Penambahan notifikasi dan pengingat otomatis untuk pembeli dan penjual mengenai status transaksi, pembayaran, dan pembaruan informasi kendaraan dapat meningkatkan komunikasi dan memastikan transaksi berjalan lancar.
2. Mengembangkan program loyalitas yang memberikan diskon atau keuntungan lainnya kepada pengguna yang melakukan transaksi berulang atau memberikan ulasan positif dapat meningkatkan retensi pengguna dan membangun komunitas yang lebih solid.
3. Untuk memperluas jangkauan pasar, sistem dapat mengembangkan layanan untuk mencakup wilayah di luar Jabodetabek, serta memperkenalkan berbagai jenis kendaraan selain mobil bekas, seperti sepeda motor atau kendaraan niaga.

Dengan implementasi saran-saran ini, sistem jual beli mobil bekas berbasis *website* yang dikembangkan diharapkan dapat lebih optimal, menciptakan pengalaman yang lebih baik untuk pengguna, serta memperkuat posisi di pasar kendaraan bekas

DAFTAR PUSTAKA

Noorviandra, S. A., Sanjaya, M. B., & Aji, P. (2020). APLIKASI JUAL BELI DAN TUKAR TAMBAH MOBIL BEKAS PADA DYA MOTOR BANDUNG. *e-Proceeding of Applied Science : Vol.6, No.2 Agustus 2020*, 1578.

Kamisa, N., Devita P, A., & Novita, D. (2022). PENGARUH ONLINE CUSTOMER REVIEW DAN ONLINE CUSTOMER RATING TERHADAP KEPERCAYAAN KONSUMEN (Studi kasus: Pengguna Shopee di Bandar Lampung). *Journal of Economic and Business Research*, 23-24.

Maulidah, A. R., Astuti, R. P., Nisa, K., Erlangga, W., & Hambarwati, E. (2024). Perkembangan Sistem Pembayaran Digital : Pada Era Revolusi Industri 4.0 Di Indonesia. *JURNAL EKONOMI DAN BISNIS DIGITAL*, 800.

Krisbiantoro, D., & Abda'u, P. D. (2021). *Dasar Pemrograman Web dengan bahasa HTML, PHP, dan Database MySQL (Vol. 1).* Zahira Media Publisher.

Silalahi, F. D. (2022). *Manajemen Database MySQL (Structured Query Language).* semarang: Yayasan PRIMA Agus Teknik .

Limbong, T., & Sriadhi. (2021). *Pemrograman Web Dasar.* Yayasan Kita Menulis.

Ramadhan, R. (2023). *RANCANGAN BANGUN E-LEARNING DAN REPOSITORI DIGITAL DENGAN PLATFORM GITHUB (Studi Kasus: Centratama Telekomunikasi Indonesia).* Jakarta: Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Laravel. (n.d.). *Installation - Laravel Documentation.* Laravel. Retrieved from https://laravel.com/docs/12.x

Kurniansyah, M. I., & S. S. (2020). Sistem Pendukung Keputusan Pemilihan Server Hosting dan Domain. *Jurnal Sistem Komputer dan Informatika (JSON)*, 14-24.

Muliadi, Andriani, M., & Irawan, H. (2020). PERANCANGAN SISTEM INFORMASI PEMESANAN KAMAR HOTEL BERBASIS WEBSITE (WEB) MENGGUNAKAN DATA FLOW DIAGRAM (DFD). *JURNAL INTEGRASI SISTEM INDUSTRI* , 113.

VSCODE. (n.d.). *Visual Studio Code - Code Editing. Redefined.* Retrieved from https://code.visualstudio.com/Search?q=what+is+vscode

AdminLTE. (2022). *Licence.* Retrieved from https://adminlte.io/docs/3.2/license.html

Sanawiah, S., & Hartiningsih, W. B. (2020). Sistem Informasi Verifikasi Dan Validasi Penempatan Jabatan Pelaksana Pada Pemerintah Provinsi Dki Jakarta. *Jurnal Sistem Informasi dan Telematika (Telekomunikasi, Multimedia dan Informatika)*, 52.

Midtrans. (n.d.). *Home*. Retrieved from Midtrans: https://midtrans.com/

Shaleh, I. A., Prayogi, J., Pirdaus, P., Syawal, R., & Saifudin, A. (2021). Pengujian Black Box pada Sistem Informasi Penjualan Buku BerbasisWeb dengan Teknik Equivalent Partitions. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 39.

Figma. (n.d.). *help*. Retrieved from https://help.figma.com/: https://help.figma.com/hc/en-us/articles/14563969806359-What-is-Figma

LAMPIRAN

helpers/MidtransHelper.php

```
<?php
namespace App\Helpers;
class MidtransHelper
{
    public static function init()
    {
        \Midtrans\Config::$serverKey = env('MIDTRANS_SERVER_KEY');
        \Midtrans\Config::$isProduction = env('MIDTRANS_IS_PRODUCTION', false);
        \Midtrans\Config::$serverKey = config('midtrans.server_key');
        \Midtrans\Config::$isProduction = config('midtrans.is_production');
        \Midtrans\Config::$isSanitized = true;
        \Midtrans\Config::$is3ds = true;
    }
}
```

Http/Controllers/Admin/CarPhotoController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\CarPhoto;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Storage;
class CarPhotoController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    public function destroy($id)
    {
        $photo = CarPhoto::findOrFail($id);
        $carId = $photo->car_id;
        if (Storage::disk('public')->exists('car_photos/' . $photo->photo_url)) {
            Storage::disk('public')->delete('car_photos/' . $photo->photo_url);
        }
        $photo->delete();
```

```
        // SUSUN ULANG NOMOR FOTO YANG TERSISA
        $remainingPhotos = CarPhoto::where('car_id', $carId)->orderBy('number')->get();
        foreach ($remainingPhotos as $index => $p) {
            $p->update(['number' => $index + 1]);
        }
        return response()->json(['message' => 'Photo deleted']);
    }
}
```

Http/Controllers/Admin/CarsController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\CarPhoto;
use App\Models\CarType;
use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;
class CarsController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Display a listing of the resource.
     */
    public function index(Request $request)
    {
        $query = Car::query();
        // Filter brand (brand dari tabel cartype)
        if ($request->brand) {
            $query->where('brand', $request->brand);
        }
        // Filter sale_type
        if ($request->sale_type) {
            $query->where('sale_type', $request->sale_type);
        }
        // Filter status
        if ($request->status) {
            $query->where('status', $request->status);
        }
        $cars = $query->orderBy('id', 'desc')->paginate(10);
        // Ambil brand unik dari CarType
        $brands = CarType::distinct()->pluck('brand');
        // Daftar pilihan sale_type dan status (bisa juga dari config/db)
        $sale_types = ['showroom' => 'Showroom', 'admin' => 'Admin'];
        $statuses = [
            'available' => 'Available',
```

```
        'pending_check' => 'Pending Check',
        'sold' => 'Sold',
        'under_review' => 'Under Review',
    ];
    return view('admin.cars.index', compact('cars', 'brands', 'sale_types', 'statuses'));
}
public function create()
{
    $user = User::all();
    $brands = CarType::distinct()->pluck('brand');
    return view('admin.cars.create', compact('user', 'brands'));
}
public function store(Request $request)
{
    $request->validate([
        'user_id' => 'required|exists:users,id',
        'brand' => 'required|string|max:50',
        'model' => 'required|string|max:50',
        'year' => 'required|integer|min:1886|max:' . date('Y'),
        'price' => 'required|integer|min:0',
        'transmission' => 'required|in:manual,automatic',
        'description' => 'nullable|string',
        'service_history' => 'nullable|date',
        'fuel_type' => 'required|string|max:50',
        'mileage' => 'required|string|max:50',
        'color' => 'required|string|max:50',
        'tax' => 'required|date',
        'engine' => 'required|integer|max:10000',
        'seat' => 'required|integer|max:20',
        'bpkb' => 'nullable|boolean',
        'spare_key' => 'nullable|boolean',
        'manual_book' => 'nullable|boolean',
        'service_book' => 'nullable|boolean',
        'sale_type' => 'required|in:user,showroom',
        'photos.*' => 'image|mimes:jpeg,png,jpg|max:2048',
    ]);
    try {
        $request->merge([
            'bpkb' => $request->has('bpkb'),
            'spare_key' => $request->has('spare_key'),
            'manual_book' => $request->has('manual_book'),
            'service_book' => $request->has('service_book'),
        ]);
        DB::transaction(function () use ($request) {
            $car = Car::create([
                'user_id' => $request->user_id,
                'brand' => $request->brand,
                'model' => $request->model,
                'year' => $request->year,
                'price' => $request->price,
```

```
                'transmission' => $request->transmission,
                'description' => $request->description,
                'service_history' => $request->service_history,
                'fuel_type' => $request->fuel_type,
                'mileage' => $request->mileage,
                'color' => $request->color,
                'tax' => $request->tax,
                'engine' => $request->engine,
                'seat' => $request->seat,
                'sale_type' => $request->sale_type,
                'status' => "pending_check",
                // kelengkapan tambahan
                'bpkb' => $request->bpkb,
                'spare_key' => $request->spare_key,
                'manual_book' => $request->manual_book,
                'service_book' => $request->service_book,
            ]);
            $photos = $request->file('photos', []);
            $orders = $request->input('photo_order', []);
            foreach ($photos as $i => $photo) {
                $filename = $photo->hashName();
                $photo->storeAs('car_photos', $filename, 'public');
                CarPhoto::create([
                    'car_id' => $car->id,
                    'photo_url' => $filename,
                    'number' => $orders[$i] ?? ($i + 1),
                ]);
            }
        });
        return redirect()->route('admin.cars.index')->with('success', 'Mobil berhasil
ditambahkan.');
    } catch (\Exception $e) {
        // Jika error, kembali ke form dengan pesan error
        return back()->withInput()->withErrors(['error' => 'Gagal menyimpan data: ' . $e-
>getMessage()]);
    }
}
public function show($id)
{
    $car = Car::with('user', 'carPhoto')->findOrFail($id);
    return view('admin.cars.partials.modal_content', compact('car'));
}
public function edit(string $id)
{
    $car = Car::with('carPhoto')->findOrFail($id);
    $user = User::all();
    $brands = CarType::distinct()->pluck('brand');
    return view('admin.cars.edit', compact('car','user', 'brands'));
}
public function update(Request $request, string $id)
```

```
{
    $car = Car::findOrFail($id);
    $request->validate([
        'user_id' => 'required|exists:users,id',
        'brand' => 'required|string|max:50',
        'model' => 'required|string|max:50',
        'year' => 'required|integer|min:1886|max:' . date('Y'),
        'price' => 'required|integer|min:0',
        'transmission' => 'required|in:manual,automatic',
        'description' => 'nullable|string',
        'service_history' => 'nullable|date',
        'fuel_type' => 'required|string|max:50',
        'mileage' => 'required|string|max:50',
        'color' => 'required|string|max:50',
        'tax' => 'required|date',
        'engine' => 'required|integer|max:10000',
        'seat' => 'required|integer|max:20',
        'bpkb' => 'nullable|boolean',
        'spare_key' => 'nullable|boolean',
        'manual_book' => 'nullable|boolean',
        'service_book' => 'nullable|boolean',
        'sale_type' => 'required|in:user,showroom',
        'photos.*' => 'image|mimes:jpeg,png,jpg|max:2048',
    ]);
    // 🔄 Atur nilai default boolean
    $request->merge([
        'bpkb' => $request->has('bpkb'),
        'spare_key' => $request->has('spare_key'),
        'manual_book' => $request->has('manual_book'),
        'service_book' => $request->has('service_book'),
    ]);
    // update data
    $car->update($request->all());
    // 🔄 Update urutan foto lama
    $existingPhotoIds = $request->input('existing_photo_ids', []);
    $existingOrders = $request->input('existing_photo_order', []);
    foreach ($existingPhotoIds as $i => $photoId) {
        CarPhoto::where('id', $photoId)->update([
            'number' => $existingOrders[$i] ?? ($i + 1),
        ]);
    }
    // 🔄 Upload foto baru
    $newPhotos = $request->file('photos', []);
    $newOrders = $request->input('new_photo_order', []);
    foreach ($newPhotos as $i => $photo) {
        $filename = $photo->hashName();
        $photo->storeAs('car_photos', $filename, 'public');
        CarPhoto::create([
            'car_id' => $car->id,
            'photo_url' => $filename,
```

```
            'number' => $newOrders[$i] ?? ($i + 1),
        ]);
    }
    return redirect()->route('admin.cars.index')->with('success', 'ID #' . $car->id . ' Car
updated successfully.');
  }
  /**
   * Remove the specified resource from storage.
   */
  public function destroy($id)
  {
    $car = Car::findOrFail($id);
    $car->delete();
    return redirect()->route('admin.cars.index')->with('success', 'ID #' . $car->id . ' Car
deleted successfully.');
  }
  /**
   * Update the status of the car via AJAX.
   */
  public function editStatus($id)
  {
    $car = Car::findOrFail($id);
    return view('admin.cars.partials.update_status_modal', compact('car'));
  }
  public function updateStatus(Request $request, $id)
  {
    // Validasi bahwa status adalah salah satu dari 4 nilai yang diperbolehkan
    $request->validate([
        'status' => 'required|in:available,pending_check,sold,under_review'
    ]);
    // Ambil data mobil berdasarkan ID
    $car = Car::findOrFail($id);
    // Update statusnya
    $car->status = $request->status;
    $car->save();
    // Kembalikan response JSON (bisa dipakai untuk notifikasi success di frontend)
    return response()->json(['message' => 'Status updated']);
  }
}
```

Http/Controllers/Admin/DashboardController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\DownPayment;
use App\Models\Offer;
use App\Models\OfferRecord;
use App\Models\SalesRecord;
use App\Models\User;
```

```
use Carbon\Carbon;
use Illuminate\Http\Request;
class DashboardController extends Controller
{
   public function __construct()
   {
      $this->middleware('auth');
      $this->middleware('checkrole:admin');
   }
   public function index()
   {
   // < START FITUR STATISTIK>
      // Menghitung jumlah down payment dengan status confirmed
      $downPayment = DownPayment::where('payment_status', 'confirmed')
         ->count();
      // Menghitung jumlah permintaan penjualan dengan status pending
      $offer = Offer::where('status', 'pending')
         ->count();
      // Menghitung jumlah mobil terjual
      $car = Car::where('status', 'sold')
         ->count();
   // < END FITUR STATISTIK>
   // <START FITUR GRAFIK>
      // Range: 7 bulan terakhir
      $startDate = Carbon::now()->subMonths(6)->startOfMonth();
      $endDate = Carbon::now()->endOfMonth();
      // Ambil bulan terakhir 7 bulan (string YYYY-MM-01)
      $months = collect(range(0, 6))->map(function ($i) {
         return Carbon::now()->subMonths(6 - $i)->format('Y-m-01');
      });
      // SALES_RECORDS: ambil COUNT & SUM per bulan
      $salesData = SalesRecord::selectRaw("DATE_FORMAT(sale_date, '%Y-%m-01')
as month, COUNT(*) as count, SUM(sale_price) as total")
         ->whereBetween('sale_date', [$startDate, $endDate])
         ->where('status', 'completed')
         ->groupBy('month')
         ->pluck('count', 'month')
         ->toArray();
      $salesAmountData = SalesRecord::selectRaw("DATE_FORMAT(sale_date, '%Y-
%m-01') as month, SUM(sale_price) as total")
         ->whereBetween('sale_date', [$startDate, $endDate])
         ->where('status', 'completed')
         ->groupBy('month')
         ->pluck('total', 'month')
         ->toArray();
      // OFFER_RECORDS: ambil COUNT & SUM per bulan
      $offerData =OfferRecord::selectRaw("DATE_FORMAT(sale_date, '%Y-%m-01')
as month, COUNT(*) as count, SUM(sale_price) as total")
         ->whereBetween('sale_date', [$startDate, $endDate])
         ->where('status', 'completed')
```

```
            ->groupBy('month')
            ->pluck('count', 'month')
            ->toArray();
        $offerAmountData =OfferRecord::query()
            ->selectRaw("DATE_FORMAT(sale_date, '%Y-%m-01') as month,
SUM(sale_price) as total")
            ->whereBetween('sale_date', [$startDate, $endDate])
            ->where('status', 'completed')
            ->groupBy('month')
            ->pluck('total', 'month')
            ->toArray();
        // Normalize: isi 0 kalau data kosong
        $salesCountSeries = $months->map(fn($month) => $salesData[$month] ?? 0);
        $salesAmountSeries = $months->map(fn($month) => $salesAmountData[$month]
?? 0);
        $offerCountSeries = $months->map(fn($month) => $offerData[$month] ?? 0);
        $offerAmountSeries = $months->map(fn($month) => $offerAmountData[$month]
?? 0);
    // < END FITUR GRAFIK>
    // <START FITUR TANGGAL>
        // DOWN PAYMENTS
        $downPayments = DownPayment::with('user')
            // ->where('appointment_date', '>=', now()->subMonths(1))
            ->whereNotNull('appointment_date')
            ->where('payment_status', 'confirmed')
            ->whereNull('refund_id') // Pastikan tidak ada refund
            ->orderBy('appointment_date')
            ->get()
            ->map(function ($item) {
                return [
                    'id'      => 'dp-' . $item->id,
                    'title'   => 'DP: ' . $item->user->name . ' - Rp' . number_format($item-
>amount, 0, ',', '.'),
                    'start'   => $item->appointment_date->format('Y-m-d\TH:i:s'),
                    'color'   => '#DC2525', // MERAH
                    'source'  => 'downpayment',
                ];
            });
        // OFFERS
        $offers = Offer::with('user')
            ->whereNotNull('inspection_date')
            ->where('status', 'accepted')
            ->orderBy('inspection_date')
            ->get()
            ->map(function ($item) {
                return [
                    'id'      => 'offer-' . $item->id,
                    'title'   => 'Offer: ' . $item->user->name . ' - ' . $item->brand . ' ' . $item-
>model,
                    'start'   => $item->inspection_date->format('Y-m-d\TH:i:s'),
```

```
                'color'   => '#20c997', // HIJAU
                'source'  => 'offer',
                'url'     => route('admin.offer.show', $item->id), // URL untuk detail offer
            ];
        });
    // Gabungkan jadi satu array
    $appointments = $downPayments->merge($offers)->values();
// <END FITUR TANGGAL>
    return view('admin.dashboard.index', [
        'downPayment'  => $downPayment,
        'offer'        => $offer,
        'car'          => $car,
        'months'             => $months->values(),
        'salesCountSeries'   => $salesCountSeries->values(),
        'salesAmountSeries'  => $salesAmountSeries->values(),
        'offerCountSeries'   => $offerCountSeries->values(),
        'offerAmountSeries'  => $offerAmountSeries->values(),
        'appointments' => $appointments,
    ]);
  }
}
```

Http/Controllers/Admin/DownPaymentController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\DownPayment;
use App\Models\Refund;
use Carbon\Carbon;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Storage;
class DownPaymentController extends Controller
{
  public function __construct()
  {
    $this->middleware('auth');
  }
  public function index(Request $request)
  {
    $query = DownPayment::query();

     // Filter by keyword (brand atau model mobil)
    if ($request->filled('keyword')) {
        $query->whereHas('car', function ($q) use ($request) {
            $q->where('brand', 'like', '%' . $request->keyword . '%')
            ->orWhere('model', 'like', '%' . $request->keyword . '%');
        });
    }
    if ($request->filled('status')) {
        [$type, $value] = explode(':', $request->status);
```

```
            if ($type === 'payment') {
                $query->where('payment_status', $value);
            }
            if ($type === 'payment' && $value === 'confirmed') {
                $query->whereNotIn('payment_status', ['pending', 'cancelled', 'expired'])
                    ->whereNull('refund_id');
            }
            // Jika statusnya refund, kita cek relasi refund
            if ($type === 'refund') {
                $query->whereHas('refund', function ($q) use ($value) {
                    $q->where('refund_status', $value);
                });
            }
        }
        // Filter by date range
        if ($request->filled('date_range')) {
            $dates = explode(' to ', $request->date_range);
            $startDate = trim($dates[0]);
            $endDate = trim($dates[1] ?? $dates[0]);
            $start = Carbon::parse($startDate)->startOfDay();
            $end = Carbon::parse($endDate)->endOfDay();
            $query->whereBetween('created_at', [$start, $end]);
        }
        // Ambil data down payment dengan relasi user dan car
        $downPayments = $query->with(['user', 'car'])->latest()->paginate(10);
        $statuses = [
            'pending' => 'Pending',
            'confirmed' => 'Confirmed',
            'cancelled' => 'Cancelled',
            'expired' => 'Expired',
            'refund' => 'Refunded',
        ];
        // $downPayments = DownPayment::with(['user','car'])->orderBy('id', 'desc')-
>paginate(10);
        return view('admin.downPayment.index', compact('downPayments', 'statuses'));
    }
    public function show($id)
    {
        $downPayment = DownPayment::with(['user', 'car'])->findOrFail($id);
        return view('admin.downPayment.partials.modal_content',
compact('downPayment'));
    }
    public function addRefund($id)
    {
        $downPayment = DownPayment::findOrFail($id);
        return view('admin.downPayment.refund', compact('downPayment'));
    }
    public function storeRefund(Request $request, $id)
    {
```

```
        $request->validate([
            'no_rekening_refund' => 'required|string|max:255',
            'refund_payment_proof' => 'required|image|mimes:jpeg,png,jpg|max:2048',

        ]);
        $refund_payment_proof = $request->file('refund_payment_proof');        // Ambil
file
        $filename = $refund_payment_proof->hashName();                    // Generate nama
unik (berbasis hash)
        $refund_payment_proof->storeAs('refund', $filename, 'public');        // Simpan file
ke storage/app/public/refund

        $refund = Refund::create([
            'no_rekening_refund' => $request->no_rekening_refund,
            'refund_payment_proof' => $filename,
            'status' => 'refund',
        ]);
        $downPayment = DownPayment::findOrFail($id);
        $downPayment->update([
            'refund_id' => $refund->id,
        ]);
        if ($refund->refund_status = 'refund') {
            // Update status mobil menjadi sold
            $downPayment->car->update(['status' => 'available']);
        }
        return redirect()->route('admin.downPayment.index')->with('success', 'Down
payment refunded successfully.');
    }
    public function editRefund($id)
    {
        $downPayment = DownPayment::with('refund')->findOrFail($id);
        return view('admin.downPayment.editRefund', compact('downPayment'));
    }
    public function updateRefund(Request $request, $id)
    {
        $request->validate([
            'no_rekening_refund' => 'required|string|max:255',
            'refund_payment_proof' => 'nullable|image|mimes:jpeg,png,jpg|max:2048',
        ]);
        $downPayment = DownPayment::with('refund')->findOrFail($id);
        // Jika belum ada refund, buat baru
        if (!$downPayment->refund) {
            $refund = new Refund();
        } else {
            $refund = $downPayment->refund;
        }
        // Update no rekening
        $refund->no_rekening_refund = $request->no_rekening_refund;
        // Jika ada file baru
        if ($request->hasFile('refund_payment_proof')) {
```

```
            // Hapus file lama jika ada
            if ($refund->refund_payment_proof && Storage::disk('public')->exists('refund/' .
$refund->refund_payment_proof)) {
                Storage::disk('public')->delete('refund/' . $refund->refund_payment_proof);
            }
            $filename = $request->file('refund_payment_proof')->hashName();
            $request->file('refund_payment_proof')->storeAs('refund', $filename, 'public');
            $refund->refund_payment_proof = $filename;
        }
        $refund->refund_status = 'refund'; // update atau tetap sama
        $refund->save();
        if ($refund->refund_status = 'refund') {
            // Update status mobil menjadi sold
            $downPayment->car->update(['status' => 'available']);
        }
        // Pastikan DownPayment terhubung ke refund
        $downPayment->refund_id = $refund->id;
        $downPayment->save();
        return redirect()->route('admin.downPayment.index')->with('success', 'Refund
berhasil diperbarui.');
    }
}
```

Http/Controllers/Admin/OfferController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\Offer;
use Carbon\Carbon;
use Illuminate\Http\Request;
class OfferController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    public function index(Request $request)
    {
        $query = Offer::query();
        // Filter status
        if ($request->filled('status')) {
            $query->where('status', $request->status);
        }
        // Filter keyword brand/model
        if ($request->filled('keyword')) {
            $query->where(function($q) use ($request) {
                $q->where('brand', 'like', '%' . $request->keyword . '%')
                ->orWhere('model', 'like', '%' . $request->keyword . '%');
            });
        }
```

```
    // Filter by date range
    if ($request->filled('date_range')) {
        $dates = explode(' to ', $request->date_range);
        $startDate = trim($dates[0]);
        $endDate = trim($dates[1] ?? $dates[0]);
        $start = Carbon::parse($startDate)->startOfDay();
        $end = Carbon::parse($endDate)->endOfDay();
        $query->whereBetween('created_at', [$start, $end]);
    }
    // $offers = Offer::orderB/y('id', 'desc')->paginate(10);
    $offers = $query->latest()->paginate(10);
    $statuses = [
        'accepted' => 'Accepted',
        'pending' => 'Pending',
        'rejected' => 'Rejected',
        'sold' => 'Sold',
    ];
    return view('admin.offer.index', compact('offers', 'statuses'));
}
public function show($id)
{
    $offer = Offer::findOrFail($id);
    return view('admin.offer.show', compact('offer'));
}
public function updateNote(Request $request, $id)
{
    // Validasi input
    $request->validate([
        'note' => 'required|string|max:255',
    ]);
    // Ambil data offer berdasarkan ID
    $offer = Offer::findOrFail($id);
    // Update catatan
    $offer->note = $request->note;
    $offer->save();
    // Kembalikan response JSON (bisa dipakai untuk notifikasi success di frontend)
     return back()->with('success', 'Catatan berhasil ditambahkan.');
}
public function editStatus($id)
{
    $offer = Offer::findOrFail($id);
    return view('admin.offer.partials.update_status_modal', compact('offer'));
}
public function updateStatus(Request $request, $id)
{
    // Validasi bahwa status adalah salah satu dari 4 nilai yang diperbolehkan
    $request->validate([
        'status' => 'required|in:accepted,pending,rejected,sold',
    ]);
    // Ambil data mobil berdasarkan ID
```

```
        $offer = Offer::findOrFail($id);
        // Update statusnya
        $offer->status = $request->status;
        $offer->save();
        // Kembalikan response JSON (bisa dipakai untuk notifikasi success di frontend)
        return response()->json([
            'message' => 'ID #' . $offer->id . ' Update Status Offer successfully.',
            'status' => 'success'
        ]);
    }
}
```

Http/Controllers/Admin/OfferRecordController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\Offer;
use App\Models\OfferRecord;
use App\Models\User;
use Illuminate\Http\Request;
class OfferRecordController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
        $this->middleware('checkrole:admin');
    }
    public function index()
    {
        $offerRecords = OfferRecord::with(['offer', 'buyerOfferRecord', 'salerOfferRecord'])
            ->orderBy('created_at', 'desc')
            ->paginate(10);
        return view('admin.offerRecord.index', compact('offerRecords'));
    }
    public function create()
    {
        $offers = Offer::where('status', 'accepted')->orderBy('created_at', 'desc')->get();
        // $offers = Offer::where('status', 'accepted')->orderBy('created_at', 'desc')->get();
        $users = User::all();
        // You can add logic here to fetch necessary data for creating an offer record
        return view('admin.offerRecord.create', compact('offers', 'users'));
    }
    public function store(Request $request)
    {
        // dd($request->all());
        $request->validate([
            'offer_id' => 'required|exists:cars,id',
            'buyer_id' => 'required|exists:users,id',
            'seller_id' => 'required|exists:users,id',
            'sale_price' => 'required|numeric|min:0',
```

```
            'sale_date' => 'required|date',
        ]);
        $offerRecord = OfferRecord::create([
            'offer_id' => $request->offer_id,
            'buyer_id' => $request->buyer_id,
            'seller_id' => $request->seller_id,
            'sale_price' => $request->sale_price,
            'sale_date' => $request->sale_date,
            'status' => 'completed', // Assuming the status is set to completed
        ]);
        // Update the car status to 'sold'

        $offer = Offer::findOrFail($request->offer_id);
        if ($offer->status !== 'sold') {
            $offer->update(['status' => 'sold']);
        }
        return redirect()->route('admin.offerRecord.index')->with('success', 'Sales record
created successfully.');
    }
    public function destroy($id){
        $offerRecord = OfferRecord::findOrFail($id);
        // dd($offerRecord->offer_id);
        $offer = Offer::findOrFail($offerRecord->offer_id);
        if ($offer->status === 'sold') {
            $offer->update(['status' => 'accepted']);
        }
        $offerRecord->delete();
        return redirect()->route('admin.offerRecord.index')->with('success', 'Sales record
created successfully.');
    }
}
```

Http/Controllers/Admin/SalesRecordController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\Offer;
use App\Models\SalesRecord;
use App\Models\User;
use Illuminate\Http\Request;
class SalesRecordController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
        $this->middleware('checkrole:admin');
    }

    public function index()
```

```
    {
        $salesRecords = SalesRecord::with(['car', 'buyer', 'saler'])
            ->orderBy('created_at', 'desc')
            ->paginate(10);
        return view('admin.salesRecord.index', compact('salesRecords'));
    }

    public function create()
    {
        $cars = Car::where('status', 'under_review')->orderBy('created_at', 'desc')->get();
        // $offers = Offer::where('status', 'accepted')->orderBy('created_at', 'desc')->get();
        $users = User::all();
        return view('admin.salesRecord.create', compact('cars', 'users'));
    }
    public function store(Request $request)
    {
        // dd($request->all());
        $request->validate([
            'car_id' => 'required|exists:cars,id',
            'buyer_id' => 'required|exists:users,id',
            'seller_id' => 'required|exists:users,id',
            'sale_price' => 'required|numeric|min:0',
            'sale_date' => 'required|date',
        ]);
        $salesRecord = SalesRecord::create([
            'car_id' => $request->car_id,
            'buyer_id' => $request->buyer_id,
            'seller_id' => $request->seller_id,
            'sale_price' => $request->sale_price,
            'sale_date' => $request->sale_date,
            'status' => 'completed', // Assuming the status is set to completed
        ]);
        // Update the car status to 'sold'
        $car = Car::findOrFail($request->car_id);
        if ($car->status !== 'sold') {
            $car->update(['status' => 'sold']);
        }
        return redirect()->route('admin.salesRecord.index')->with('success', 'Sales record
created successfully.');
    }
    public function destroy($id)
    {
        $salesRecord = SalesRecord::findOrFail($id);
        $salesRecord->delete();
        return redirect()->route('admin.salesRecord.index')->with('success', 'Sales record
deleted successfully.');
    }
}
```

Http/Controllers/Admin/UsersController.php

```
<?php
namespace App\Http\Controllers\Admin;
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use App\Models\User;
class UsersController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    public function index(Request $request)
    {
        $search = $request->input('search');
        $address = $request->input('address');
        $role = $request->input('role');
        $users = User::when($role, function ($query, $role) {
            return $query->where('role', $role);
        })
            ->when($search, function ($query, $search) {
                return $query->where('name', 'like', "%{$search}%")
                    ->orWhere('email', 'like', "%{$search}%");
            })
            ->when($address, function ($query, $address) {
                return $query->where('address', 'like', "%{$address}%");
            })
            ->orderBy('id', 'desc')
            ->paginate(10);
        return view('admin.users.index', compact('users'));
    }
    public function create()
    {
        return view('admin.users.create');
    }
    public function store(Request $request)
    {
        $request->validate([
            'name' => 'required|string|max:50',
            'email' => 'required|string|email|max:50|unique:users',
            'password' => 'required|string|min:8|confirmed',
            'phone' => 'required|string|max:15',
            'address' => 'required|string|max:255',
            'role' => 'required|string|max:5',
        ]);
        $user = User::create([
            'name' => $request->name,
            'email' => $request->email,
            'password' => Hash::make($request->password),
            'phone' => $request->phone,
```

```
            'address' => $request->address,
            'role' => $request->role,
        ]);
        // dd($request->all());
        return redirect()->route('admin.users.index')->with('success', 'ID #'.$user->id.' User
created successfully.');
    }
    public function edit($id)
    {
        $user = User::findOrFail($id);
        return view('admin.users.edit', compact('user'));
    }
    public function update(Request $request, $id)
    {
        $user = User::findOrFail($id);
        $request->validate([
            'name' => 'required|string|max:50',
            'email' => 'required|string|email|max:50|unique:users,email,' . $user->id,
            'phone' => 'required|string|max:15',
            'address' => 'required|string|max:255',
            'role' => 'required|string|max:5',
        ]);
        $user->name = $request->name;
        $user->email = $request->email;
        if ($request->password) {
            $user->password = Hash::make($request->password);
        }
        $user->phone = $request->phone;
        $user->address = $request->address;
        $user->role = $request->role;
        $user->save();
        return redirect()->route('admin.users.index')->with('success', 'ID #'.$user->id.' User
updated successfully.');
    }
    public function destroy($id)
    {
        $user = User::findOrFail($id);
        $user->delete();
        return redirect()->route('admin.users.index')->with('success', 'ID #'.$user->id.' User
deleted successfully.');
    }
}
```

Http/Controllers/Auth/ConfirmPasswordController.php

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\ConfirmsPasswords;
class ConfirmPasswordController extends Controller
{
```

```
    /*
    |--------------------------------------------------------------------------
    | Confirm Password Controller
    |--------------------------------------------------------------------------
    |
    | This controller is responsible for handling password confirmations and
    | uses a simple trait to include the behavior. You're free to explore
    | this trait and override any functions that require customization.
    |
    */
    use ConfirmsPasswords;
    /**
     * Where to redirect users when the intended url fails.
     *
     * @var string
     */
    protected $redirectTo = '/home';
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('auth');
    }
}
```

Http/Controllers/Auth/ForgotPasswordController.php

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\SendsPasswordResetEmails;
class ForgotPasswordController extends Controller
{
    /*
    |--------------------------------------------------------------------------
    | Password Reset Controller
    |--------------------------------------------------------------------------
    |
    | This controller is responsible for handling password reset emails and
    | includes a trait which assists in sending these notifications from
    | your application to your users. Feel free to explore this trait.
    |
    */
    use SendsPasswordResetEmails;
}
```

Http/Controllers/Auth/LoginController.php

```
<?php
```

```
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\AuthenticatesUsers;
use Illuminate\Http\Request;
class LoginController extends Controller
{
    /*
    |--------------------------------------------------------------------------
    | Login Controller
    |--------------------------------------------------------------------------
    |
    | This controller handles authenticating users for the application and
    | redirecting them to your home screen. The controller uses a trait
    | to conveniently provide its functionality to your applications.
    |
    */
    use AuthenticatesUsers;
    /**
     * Where to redirect users after login.
     *
     * @var string
     */
    protected $redirectTo = '/home';
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('guest')->except('logout');
        $this->middleware('auth')->only('logout');
    }
    public function showLoginForm(Request $request)
    {
        if ($request->has('redirect')) {
            $redirect = $request->redirect;
            // Hindari menyimpan redirect ke /login atau ke /user/dashboard
            if (!str_contains($redirect, '/login') && !str_contains($redirect, '/dashboard')) {
                session(['url.intended' => $redirect]);
            }
        }
        if (isset($redirect) && str_contains($redirect, '/admin')) {
            return view('auth.login');
        }
        // Hindari redirect loop kalau asalnya dari halaman butuh login
        if (url()->previous() === url()->current() || str_contains(url()->previous(),
'/dashboard')) {
            return redirect('/');
        }
```

```
        return redirect()->back()->with('showLoginModal', true);
    }
}
```

Http/Controllers/Auth/RegisterController.php

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use App\Models\User;
use Illuminate\Foundation\Auth\RegistersUsers;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
class RegisterController extends Controller
{
    /*
    |--------------------------------------------------------------------------
    | Register Controller
    |--------------------------------------------------------------------------
    |
    | This controller handles the registration of new users as well as their
    | validation and creation. By default this controller uses a trait to
    | provide this functionality without requiring any additional code.
    |
    */
    use RegistersUsers;
    /**
     * Where to redirect users after registration.
     *
     * @var string
     */
    protected $redirectTo = '/home';
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('guest');
    }
    /**
     * Get a validator for an incoming registration request.
     *
     * @param  array  $data
     * @return \Illuminate\Contracts\Validation\Validator
     */
    protected function validator(array $data)
    {
        return Validator::make($data, [
            'name' => ['required', 'string', 'max:50'],
```

```
            'email' => ['required', 'string', 'email', 'max:50', 'unique:users'],
            'password' => ['required', 'string', 'min:8', 'confirmed'],
            'phone' => ['required', 'numeric', 'digits_between:10,15'],
        ]);
    }
    /**
     * Create a new user instance after a valid registration.
     *
     * @param  array  $data
     * @return \App\Models\User
     */
    protected function create(array $data)
    {
        return User::create([
            'name' => $data['name'],
            'email' => $data['email'],
            'phone' => $data['phone'],
            'password' => Hash::make($data['password']),
        ]);
    }
    public function showRegistrationForm()
    {
        return view('web.auth.register');
    }
}
```

Http/Controllers/Auth/ResetPasswordController.php

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\ResetsPasswords;
class ResetPasswordController extends Controller
{
    /*
    |--------------------------------------------------------------------------
    | Password Reset Controller
    |--------------------------------------------------------------------------
    |
    | This controller is responsible for handling password reset requests
    | and uses a simple trait to include this behavior. You're free to
    | explore this trait and override any methods you wish to tweak.
    |
    */
    use ResetsPasswords;
    /**
     * Where to redirect users after resetting their password.
     *
     * @var string
     */
    protected $redirectTo = '/home';
```

```
}
```

Http/Controllers/Auth/VerificationController.php

```
<?php
namespace App\Http\Controllers\Auth;
use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\VerifiesEmails;
class VerificationController extends Controller
{
   /*
   |--------------------------------------------------------------------------
   | Email Verification Controller
   |--------------------------------------------------------------------------
   |
   | This controller is responsible for handling email verification for any
   | user that recently registered with the application. Emails may also
   | be re-sent if the user didn't receive the original email message.
   |
   */
   use VerifiesEmails;
   /**
    * Where to redirect users after verification.
    *
    * @var string
    */
   protected $redirectTo = '/home';
   /**
    * Create a new controller instance.
    *
    * @return void
    */
   public function __construct()
   {
      $this->middleware('auth');
      $this->middleware('signed')->only('verify');
      $this->middleware('throttle:6,1')->only('verify', 'resend');
   }
}
```

- Http/Controllers/User/DashboardController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Http\Controllers\Controller;
use App\Models\DownPayment;
use App\Models\Offer;
use App\Models\OfferRecord;
use App\Models\SalesRecord;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class DashboardController extends Controller
{
   public function __construct()
```

```
        {
            $this->middleware('auth');
        }
        public function index(){
            $downPayment = DownPayment::with('car') // kalau relasi ke Car
                ->where('payment_status', 'pending')
                ->where('user_id', Auth::id())
                ->orderBy('id', 'desc')
                ->first();

            $offers = Offer::where('status', 'accepted')
                ->where('user_id', Auth::id())
                ->orderBy('id', 'desc')->get();
            $salesRecord = SalesRecord::where('buyer_id', Auth::id())
                ->where('status', 'completed')
                ->count();
            $offerRecord = OfferRecord::where('seller_id', Auth::id())
                ->where('status', 'completed')
                ->count();
            return view('user.dashboard.index', compact('downPayment', 'offers',
    'salesRecord', 'offerRecord'));
        }
    }
```

Http/Controllers/User/DownPaymentController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Helpers\MidtransHelper;
use App\Http\Controllers\Controller;
use App\Models\DownPayment;
use Illuminate\Database\Console\Migrations\StatusCommand;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Midtrans\Snap;
use Midtrans\Config;
use Midtrans\Transaction;
class DownPaymentController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    /**
     * Display a listing of the resource.
     */
    public function index()
    {
        $downPayments = DownPayment::with('car')->where('user_id', Auth::id())-
>orderBy('id', 'desc')->paginate(10);
        return view('user.downPayment.index', compact('downPayments'));
```

```
    }

    public function checkout($id){
        $snapToken = 'xxx';
        $downPayments = DownPayment::with('car')->where('user_id', Auth::id())-
>findOrFail($id);
        return view('user.downPayment.checkout', compact('downPayments', 'snapToken'));
    }
}
```

- Http/Controllers/User/OfferController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Http\Controllers\Controller;
use App\Models\Offer;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class OfferController extends Controller
{
    public function __construct()
    {
        $this->middleware('auth');
    }
    public function index()
    {
        $offers = Offer::with('user')->where('user_id', Auth::id())->orderBy('id', 'desc')-
>get();
        // Logic to display a list of offers
        return view('user.offer.index',compact('offers'));
    }
    public function show($id)
    {
        $offer = Offer::with('user')->where('user_id', Auth::id())->findOrFail($id);
        // Logic to display a specific offer
        return view('user.offer.show', compact('offer'));
    }
}
```

Http/Controllers/User/PaymentController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Helpers\MidtransHelper;
use App\Http\Controllers\Controller;
use App\Models\Car;
use Illuminate\Http\Request;
use App\Models\DownPayment;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Http;
use Midtrans\Config;
use Midtrans\Snap;
use Midtrans\Notification;
```

```
use Illuminate\Support\Facades\Log;
use Midtrans\Transaction;
class PaymentController extends Controller
{
   public function checkout($id)
   {
      $downPayment = DownPayment::with(['user', 'car'])->where('user_id', Auth::id())-
>findOrFail($id);
      MidtransHelper::init();
      // Cek status order lama dulu
      if ($downPayment->order_id) {
         try {
            $midtransStatus = Transaction::status($downPayment->order_id);
            $transaction = $midtransStatus->transaction_status ?? null;
            if ($transaction === 'settlement') {
               $downPayment->update([
                  'payment_status' => 'confirmed',
                  'payment_date' => $midtransStatus->transaction_time,
                  'payment_method' => $midtransStatus->payment_type,
                  'snap_token' => null,
               ]);
               $downPayment->car->update(['status' => 'under_review']);
               return redirect()->route('user.downPayment.show', $id)
                  ->with('success', 'Pembayaran berhasil!');
            } elseif ($transaction === 'expire' || $transaction === 'cancel') {
               $downPayment->update([
                  'payment_status' => $transaction === 'expire' ? 'expired' : 'cancelled',
                  'snap_token' => null,
               ]);
            } elseif ($transaction === 'pending') {
               // masih pending → pakai snap_token lama
               return view('user.downPayment.checkout', [
                  'downPayments' => $downPayment,
                  'snapToken' => $downPayment->snap_token,
               ]);
            }
         } catch (\Throwable $e) {
            // fallback → buat order baru
         }
      }
      // Jika belum ada order atau sudah expire/cancel
      $orderId = 'DP-' . $downPayment->id . '-' . time();
      $params = [
         'transaction_details' => [
            'order_id' => $orderId,
            'gross_amount' => (int) $downPayment->amount,
         ],
         'customer_details' => [
            'first_name' => $downPayment->user->name,
            'email' => $downPayment->user->email,
```

```
            'phone' => $downPayment->user->phone,
        ],
    ];
    $snapToken = Snap::getSnapToken($params);
    $downPayment->update([
        'order_id' => $orderId,
        'snap_token' => $snapToken,
        'payment_status' => 'pending',
    ]);
    return view('user.downPayment.checkout', [
        'downPayments' => $downPayment,
        'snapToken' => $snapToken,
    ]);
}
public function changeStatus($id) {
    $downPayment = DownPayment::with(['user', 'car'])->where('user_id', Auth::id())-
>findOrFail($id);
    MidtransHelper::init();
    $statusFromMidtrans = null;
    try {
        $statusFromMidtrans = Transaction::status($downPayment->order_id);
    } catch (\Throwable $th) {
    }
    $transaction = $statusFromMidtrans->transaction_status ?? "pending";
    $status = "pending";
    $paymentMethod = null;
    if ($transaction == 'settlement') {
        $status = 'confirmed';
        $paymentMethod = $statusFromMidtrans->payment_type ?? null;
    } elseif ($transaction == 'expire') {
        $status = 'expired';
    }elseif ($transaction == 'cancel') {
        $status = 'cancelled';
    } elseif ($transaction == 'pending') {
        $status = 'pending';
    }
    if ($transaction != 'pending') {
        DownPayment::where("id", $id)
        ->update([
            "payment_status" => $status,
            "payment_date" => $statusFromMidtrans->transaction_time ?? null,
            "payment_method" => $paymentMethod,
            'snap_token' => null
        ]);
    }else {
        $orderId = 'DP-' . $downPayment->id . '-' . time();
        // Siapkan parameter transaksi
        $params = [
            'transaction_details' => [
                'order_id' => $orderId,
```

```
                'gross_amount' => (int) $downPayment->amount,
            ],
            'customer_details' => [
                'first_name' => $downPayment->user->name,
                'email' => $downPayment->user->email,
                'phone' => $downPayment->user->phone,
            ],
        ];
        $snapToken = !empty($downPayment->snap_token) ? $snapToken =
$downPayment->snap_token : $snapToken = Snap::getSnapToken($params);
        DownPayment::where('id', $id)
        ->where('payment_status', 'pending')
        ->update([
            'order_id' => $orderId,
            'snap_token' => $snapToken
        ]);
        $newDownPayment = DownPayment::with(['user', 'car'])->where('user_id',
Auth::id())->findOrFail($id);
        return view('user.downPayment.checkout', [
            'downPayments' => $newDownPayment,
            'snapToken' => $snapToken
        ]);
    }

    if ($status == 'confirmed') {
        // Update status mobil menjadi sold
        $downPayment->car->update(['status' => 'under_review']);
    }
    $newDownPayment = DownPayment::with(['user', 'car'])->where('user_id',
Auth::id())->findOrFail($id);
    return view('user.downPayment.checkout', [
        'downPayments' => $newDownPayment,
        'snapToken' => ""
    ]);
  }
  // Webhook Midtrans
  public function notificationHandler(Request $request)
  {
    MidtransHelper::init();
    Log::info('Webhook Midtrans MASUK:', $request->all());
    try {
        $notif = new Notification();
        $transaction = $notif->transaction_status;
        $orderId = $notif->order_id;
        // Format order_id = DP-5-1718000000
        $id = explode('-', $orderId)[1] ?? null;
        $payment = DownPayment::find($id);
        if (!$payment) {
            Log::error("❌ DownPayment ID {$id} tidak ditemukan.");
            return response()->json(['message' => 'Not found'], 404);
```

```
            }
            // Proses status dari Midtrans
            $status = "pending";
            $paymentMethod = null;
            if ($transaction == 'settlement') {
                $status = 'confirmed';
                $paymentMethod = $data['payment_type'] ?? null;
            } elseif ($transaction == 'expire') {
                $status = 'expired';
            }elseif ($transaction == 'cancel') {
                $status = 'cancelled';
            } elseif ($transaction == 'pending') {
                $status = 'pending';
            }

            DownPayment::where("id", $id)
                ->update([
                    "payment_status" => $status,
                    "payment_date" => $notif->transaction_time ?? null,
                    "payment_method" => $paymentMethod,
                ]);
            $payment->save();
            Log::info("✅ Status untuk ID {$id} diupdate menjadi {$payment-
>payment_status}");
            return response()->json(['message' => 'Success']);
        } catch (\Exception $e) {
            Log::error('🔥 ERROR Webhook: ' . $e->getMessage());
            return response()->json(['message' => $e->getMessage()], 500);
        }
    }
    public function checkMidtransStatus($id)
    {
        $downPayment = DownPayment::findOrFail($id);

        $car = Car::findOrFail($downPayment->car_id);
        if ($car->status === 'sold' || $car->status === 'under_review' || $car->status ===
'pending_check') {
            return response()->json(['status' => $car->status], 200);
        }
        $serverKey = config('services.midtrans.server_key');
        $encodedKey = base64_encode($serverKey . ':');
        $response = Http::withHeaders([
            'Accept' => 'application/json',
            'Authorization' => 'Basic ' . $encodedKey
        ])->get("https://api.sandbox.midtrans.com/v2/{$downPayment->order_id}/status");
        if ($response->successful()) {
            $data = $response->json();
            // dd($data);
            // Convert status Midtrans ke sistem lokal (optional)
            if ($data["status_code"] == 200) {
```

```
            $status = match ($data['transaction_status']) {
                'settlement' => 'confirmed',
                'expire' => 'cancelled',
                'pending' => 'pending',
                default => 'unknown',
            };
            $transaction = $data['transaction_status'];
            if ($transaction == 'settlement') {
                $status = 'confirmed';
                $paymentMethod = $data['payment_type'] ?? null;
            } elseif ($transaction == 'expire') {
                $status = 'expired';
            }elseif ($transaction == 'cancel') {
                $status = 'cancelled';
            } elseif ($transaction == 'pending') {
                $status = 'pending';
            }
            DownPayment::where("id", $id)
                ->update([
                    "payment_status" => $status,
                    "payment_date" => $data['transaction_time'] ?? null,
                    "payment_method" => $paymentMethod,
                    "snap_token" => null
                ]);
            return response()->json(['status' => $status]);
        }else {
            $id = DownPayment::where("id", $id)
            ->update([
                'snap_token' => null
            ]);
        return response()->json(
            [
                'status' => 'error',
                'snap_token' => $downPayment->snap_token
            ],
            $response->status());
        }
    }
    DownPayment::where("id", $id)
        ->update([
            'snap_token' => null
    ]);
    return response()->json(['status' => 'error'], $response->status());
  }
}
```

Http/Controllers/User/ProfileController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Http\Controllers\Controller;
```

```
use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class ProfileController extends Controller
{
   public function __construct()
   {
      $this->middleware('auth');
   }
   public function updateAddress(Request $request, $id)
   {
      // Validasi input
      $request->validate([
         'address' => 'required|string|max:255',
      ]);
      // Cari user berdasarkan ID
      $user = User::findOrFail($id);
      // Pastikan hanya user yang sedang login yang bisa update datanya sendiri
      if ($user->id !== Auth::id()) {
         abort(403, 'Tidak diizinkan memperbarui data ini.');
      }
      // Update address
      $user->address = $request->address;
      $user->save();
      return back()->with('success', 'Alamat berhasil diperbarui.');
   }
}
```

Http/Controllers/User/TransactionOfferRecordController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Http\Controllers\Controller;
use App\Models\OfferRecord;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class TransactionOfferRecordController extends Controller
{
   public function __construct()
   {
      $this->middleware('auth');
   }
   public function index(Request $request)
   {
      $offerRecords = OfferRecord::with(['offer', 'buyerOfferRecord',
'salerOfferRecord'])->where('seller_id', Auth::id())->orderBy('created_at', 'desc')-
>paginate(10);;

      return view('user.transactionOfferRecord.index', compact('offerRecords'));
   }
}
```

Http/Controllers/User/TransactionSalesRecordController.php

```
<?php
namespace App\Http\Controllers\User;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\Review;
use App\Models\SalesRecord;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class TransactionSalesRecordController extends Controller
{
   public function __construct()
   {
      $this->middleware('auth');
   }
   public function index(Request $request)
   {
      $salesRecords = SalesRecord::with(['car', 'buyer', 'saler'])->where('buyer_id',
Auth::id())->orderBy('created_at', 'desc')->paginate(10);;

      return view('user.transactionSalesRecord.index', compact('salesRecords'));
   }
   public function createTesti($id) {
      $salesRecord = SalesRecord::findOrFail($id);
      $car = Car::findOrFail($salesRecord->car_id);
      $existingReview = Review::where('sales_record_id', $salesRecord->id)
         ->where('car_id', $car->id)
         ->first();
      if ($existingReview) {
         return redirect()->route('user.transactionSalesRecord.index')->with('error',
'Testimoni untuk transaksi ID #' . $salesRecord->id . ' ini sudah pernah dibuat.');
      }
      return view('user.transactionSalesRecord.testimoni', compact('salesRecord'));
   }
   public function storeTesti(Request $request, $id)
   {
      $user_id = Auth::id();
      $salesRecord = SalesRecord::findOrFail($id);
      $car = Car::findOrFail($salesRecord->car_id);
      $request->validate([
         'rating' => 'required|numeric|min:1|max:5',
         'comment' => 'required|string|max:255',
         'photo_review' => 'required|image|mimes:jpeg,png,jpg|max:2048',
      ]);
      // Cek apakah sudah ada review dengan kombinasi sales_record_id dan car_id
      $existingReview = Review::where('sales_record_id', $salesRecord->id)
         ->where('car_id', $car->id)
         ->first();
      if ($existingReview) {
```

```
            return redirect()->route('user.transactionSalesRecord.index')->with('error',
'Testimoni untuk transaksi ini sudah pernah dibuat.');
        }
        // Upload foto review
        $photo_review_proof = $request->file('photo_review');         // Ambil file
        $filename = $photo_review_proof->hashName();                  // Generate nama
unik (berbasis hash)
        $photo_review_proof->storeAs('photo_review', $filename, 'public');
        // Simpan review ke database
        Review::create([
            'user_id' => $user_id,
            'sales_record_id' => $salesRecord->id,
            'car_id' => $car->id,
            'rating' => $request->rating,
            'comment' => $request->comment,
            'photo_review' => $filename,
        ]);
        return redirect()->route('user.transactionSalesRecord.index')->with('success',
'Testimoni berhasil disimpan.');
    }
}
```

Http/Controllers/Web/DetailCarController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Http\Controllers\Controller;
use App\Models\Car;
use Illuminate\Http\Request;
class DetailCarController extends Controller
{
    public function index($id)  {
        $car = Car::with('carPhoto', 'mainPhoto')->findOrFail($id);

        if ($car->status == "available") {
            return view('web.detailMobil', compact('car'));
        }else{
            return  view('errors.404');
        }
    }
}
```

Http/Controllers/Web/DownPaymentController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Helpers\MidtransHelper;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\DownPayment;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
```

```
use Midtrans\Snap;
class DownPaymentController extends Controller
{
   public function __construct()
   {
      $this->middleware('auth');
   }

   public function store(Request $request)
   {
      $user = Auth::user();
      if (!$user || $user->role !== 'user') {
         return redirect()->back()->with('error', 'Admin tidak diperbolehkan melakukan
pembayaran DP.');
      }
      // Validasi input
      $request->validate([
         'appointment_date' => 'required|date',
         'appointment_time' => 'required',
         'amount' => 'required|numeric|min:500000',
         'car_id' => 'required|exists:cars,id',
      ]);
      $car = Car::findOrFail($request->car_id);

      // Gabungkan tanggal dan waktu
      $appointmentDateTime = $request->appointment_date . ' ' . $request-
>appointment_time;
      // Jika kolom di database `appointment_date` bertipe DATETIME
      // Maka konversi ke format datetime valid (pakai Carbon)
      $appointmentDateTime = \Carbon\Carbon::createFromFormat('Y-m-d H:i',
$appointmentDateTime);
      // Simpan data ke tabel (misal DownPayment)
      $store = DownPayment::create([
         'car_id' => $car->id,
         'user_id' => Auth::guard('web')->id(),
         'appointment_date' => $appointmentDateTime,
         'amount' => $request->amount,
      ]);
      $store->save();

      return redirect()->route('user.downPayment.checkout' , $store->id);
   }
   public function show($id){
      $car = Car::with('mainPhoto')->findOrFail($id);

      if ($car->status == "available") {
         return view('web.downPayment', compact('car'));
      }else{
         return  view('errors.404');
      }
```

```
    }
    private function goToPaymentView($id) {
        $downPayment = DownPayment::with(['user', 'car'])->where('user_id', Auth::id())-
>findOrFail($id);
        // Konfigurasi Midtrans
        MidtransHelper::init();
        $orderId = 'DP-' . $downPayment->id . '-' . time();
        // Siapkan parameter transaksi
        $params = [
            'transaction_details' => [
                'order_id' => $orderId,
                'gross_amount' => (int) $downPayment->amount,
            ],
            'customer_details' => [
                'first_name' => $downPayment->user->name,
                'email' => $downPayment->user->email,
                'phone' => $downPayment->user->phone,
            ],
        ];
        $snapToken = Snap::getSnapToken($params);
        DownPayment::where('id', $id)
        ->where('payment_status', 'pending')
        ->update(['order_id' => $orderId]);
        return view('user.downPayment.checkout', [
            'downPayments' => $downPayment,
            'snapToken' => $snapToken
        ]);
    }
}
```

Http/Controllers/Web/HomeController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Http\Controllers\Controller;
use App\Models\Car;
use App\Models\CarType;
use App\Models\Review;
use Illuminate\Http\Request;
class HomeController extends Controller
{
    public function index(Request $request){
        // Ambil 3 review terbaru yang sudah disetujui
        $reviews = Review::
            with(['car', 'user'])
            ->inRandomOrder()
            ->take(3)
            ->get();

        $query = Car::query();
        // Filter pencarian bebas
```

```
        if ($request->filled('q')) {
            $query->where(function ($q) use ($request) {
                $q->where('brand', 'like', '%' . $request->q . '%')
                ->orWhere('model', 'like', '%' . $request->q . '%')
                ->orWhere('year', 'like', '%' . $request->q . '%');
            });
        }
        // Filter berdasarkan brand
        if ($request->filled('brand')) {
            $query->where('brand', $request->brand);
        }
        // Filter berdasarkan jarak tempuh
        if ($request->filled('mileage')) {
            [$min, $max] = explode('-', $request->mileage);
            $query->whereBetween('mileage', [(int) $min, (int) $max]);
        }
        // Filter berdasarkan tahun
        if ($request->filled('year')) {
            $query->where('year', $request->year);
        }
        $cars = $query->with('mainPhoto')->where('status','available')->orderBy('created_at',
'desc')->paginate(12);
        $brands = CarType::distinct()->pluck('brand');
        return view('web.home', compact('cars', 'brands', 'reviews'));
    }
}
```

Http/Controllers/Web/JualMobilController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Http\Controllers\Controller;
use App\Models\CarType;
use App\Models\Offer;
use App\Models\Review;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class JualMobilController extends Controller
{
    /**
     * Display a listing of the resource.
     */
    public function index()
    {
        $brands = CarType::distinct()->pluck('brand');
        $reviews = Review::
            with(['car', 'user'])
            ->inRandomOrder()
            ->take(3)
            ->get();
        return view('web.jualMobil', compact('brands', 'reviews'));
```

```
    }
    /**
     * Show the form for creating a new resource.
     */
    public function create()
    {

    }
    /**
     * Store a newly created resource in storage.
     */
    public function store(Request $request)
    {
        $user = Auth::user();
        if (!$user || $user->role !== 'user') {
            return redirect()->back()->with('error', 'Admin tidak diperbolehkan melakukan
pembayaran DP.');
        }
        $request->validate([
            'brand' => 'required|string|max:255',
            'model' => 'required|string|max:255',
            'year' => 'required|digits:4|integer|min:2000|max:' . date('Y'),
            'mileage' => 'required|string',
            'offered_price' => 'required|numeric|min:1000000',
            'location_inspection' => 'required|string',
            'inspection_date' => 'required|date|after_or_equal:today',
        ]);
        $offer = Offer::create([
            'user_id' => Auth::id(),
            'brand' => $request->brand,
            'model' => $request->model,
            'year' => $request->year,
            'mileage' => $request->mileage,
            'offered_price' => $request->offered_price,
            'location_inspection' => $request->location_inspection,
            'inspection_date' => $request->inspection_date,
            'status' => 'pending',
        ]);
        return redirect()->route('user.offer.show', $offer->id)->with('success', 'Data
Penjualan mobil berhasil disubmit! Silahkan Tunggu Konfirmasi dari Admin Kontak:
081220745317');
    }
    /**
     * Display the specified resource.
     */
    public function show(string $id)
    {
        //
    }
    /**
```

```
     * Show the form for editing the specified resource.
     */
    public function edit(string $id)
    {
        //
    }
    /**
     * Update the specified resource in storage.
     */
    public function update(Request $request, string $id)
    {
        //
    }
    /**
     * Remove the specified resource from storage.
     */
    public function destroy(string $id)
    {
        //
    }
}
```

Http/Controllers/Web/KontakController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Http\Controllers\Controller;
use Illuminate\Http\Request;
class KontakController extends Controller
{
    public function index() {
        return view('web.kontak');
    }
}
```

Http/Controllers/Web/TestimonialController.php

```
<?php
namespace App\Http\Controllers\Web;
use App\Http\Controllers\Controller;
use App\Models\Review;
use Illuminate\Http\Request;
class TestimonialController extends Controller
{
    public function index(){
        $reviews = Review::
            with(['car', 'user'])
            ->orderBy('created_at', 'desc')
            ->get();
        return view('web.testimonial', compact('reviews'));
    }
}
```

Http/Controllers/Controller.php

```
<?php
namespace App\Http\Controllers;
use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
use Illuminate\Foundation\Validation\ValidatesRequests;
use Illuminate\Routing\Controller as BaseController;
class Controller extends BaseController
{
   use AuthorizesRequests, ValidatesRequests;
}
```

Http/Controllers/HomeController.php

```
<?php
namespace App\Http\Controllers;
use Illuminate\Support\Facades\Auth;
use Illuminate\Http\Request;
use App\Http\Controllers\Admin\DashboardController as
AdminDashboardController;
use App\Http\Controllers\User\DashboardController as
UserDashboardController;
class HomeController extends Controller
{
   /**
    * Create a new controller instance.
    *
    * @return void
    */
   public function __construct()
   {
      $this->middleware('auth');
   }
   /**
    * Show the application dashboard.
    *
    * @return \Illuminate\Contracts\Support\Renderable
    */
   public function index()
   {
      if (Auth::user()->role == 'admin') {
         $dashboard = new AdminDashboardController();
         return $dashboard->index();
      } else {
         $dashboard = new UserDashboardController();
         return $dashboard->index();
      }
   }
}
```

Http/Middleware/CheckRole.php

```
<?php
namespace App\Http\Middleware;
use Closure;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
class CheckRole
{
   public function handle(Request $request, Closure $next, $role)
   {
      // Cek apakah pengguna sudah login dan apakah role-nya sesuai dengan yang
diminta
      if (!Auth::check() || Auth::user()->role !== $role) {
         // Jika tidak sesuai, redirect ke halaman utama atau halaman lainnya
         return redirect('/');
      }
      return $next($request);
   }
}
```

Models/Car.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
use \Illuminate\Database\Eloquent\Factories\HasFactory;
class Car extends Model
{
   use HasFactory;
   protected $fillable = [
      'brand',
      'model',
      'year',
      'price',
      'transmission',
      'description',
      'service_history',
      'fuel_type',
      'mileage',
      'color',
      'tax',
      'engine',
      'seat',
      'bpkb',
      'spare_key',
      'manual_book',
      'service_book',
      'sale_type',
      'status',
      'user_id',
   ];
   // Define the relationship with the User model
```

```
    public function user()
    {
        return $this->belongsTo(User::class);
    }
    // Define the relationship with the Sales model
    public function salesRecords()
    {
        // return $this->hasMany(SalesRecord::class);
        return $this->hasOne(SalesRecord::class);
    }
    // Define the relationship with the DownPayment model
    public function downPayment()
    {
        return $this->hasMany(DownPayment::class);
    }
    // Define the relationship with the reviews model
    public function review()
    {
        // return $this->hasMany(Review::class);
        return $this->hasOne(Review::class);
    }
    // Define the relationship with the Car_Photo model
    public function carPhoto()
    {
        return $this->hasMany(CarPhoto::class)->orderBy('number');
    }
    // Relasi khusus untuk foto utama
    public function mainPhoto()
    {
        return $this->hasOne(CarPhoto::class)->orderBy('number', 'asc');
    }
}
```

Models/CarPhoto.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Factories\HasFactory;
class CarPhoto extends Model
{
    use HasFactory;

    protected $fillable = [
        'car_id',
        'photo_url',
        'number'
    ];
    // defining the relationship with the Car model
    public function car()
    {
```

```
        return $this->belongsTo(Car::class);
    }
}
```

Models/CarType.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
class CarType extends Model
{
    use HasFactory;
    protected $fillable = [
        'tahun',
        'brand',
        'model',
    ];
}
```

Models/DownPayment.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
class DownPayment extends Model
{
    use HasFactory;
    protected $fillable = [
        'user_id',
        'car_id',
        'amount',
        'appointment_date',
        'payment_status',
        'payment_date',
        'order_id',
        'snap_token',
        'payment_method',
        'refund_id',
    ];
    protected $casts = [
        'payment_date' => 'datetime',
        'appointment_date' => 'datetime',
    ];
    // Define the relationship with the User model
    public function user()
    {
        return $this->belongsTo(User::class);
    }
    // Define the relationship with the Car model
    public function car()
```

```
    {
        return $this->belongsTo(Car::class);
    }

    // Define the relationship with the Refund model
    public function refund()
    {
        return $this->belongsTo(Refund::class);
    }
}
```

Models/Offer.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
use \Illuminate\Database\Eloquent\Factories\HasFactory;
class Offer extends Model
{
    use hasFactory;
    protected $fillable = [
        'user_id',
        'brand',
        'model',
        'year',
        'mileage',
        'offered_price',
        'status',
        'location_inspection',
        'inspection_date',
        'note',
    ];
    protected $casts = [
        'inspection_date' => 'datetime',
    ];
    // Define the relationship with the User model
    public function user()
    {
        return $this->belongsTo(User::class);
    }

    // Define the relationship with the OfferRecord model
    public function offerRecords()
    {
        // return $this->hasMany(OfferRecord::class);
        return $this->hasOne(OfferRecord::class);
    }
}
```

Models/OfferRecord.php

```
<?php
```

```
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
class OfferRecord extends Model
{
   protected $fillable = [
      'offer_id',
      'buyer_id',
      'seller_id',
      'sale_price',
      'sale_date',
      'status',
   ];
   // Define the relationship with the Car model
   public function offer()
   {
      return $this->belongsTo(Offer::class);
   }
   // Define the relationship with the User model
   public function buyerOfferRecord()
   {
      return $this->belongsTo(User::class, 'buyer_id');
   }

   public function salerOfferRecord()
   {
      return $this->belongsTo(User::class, 'seller_id');
   }
}
```

Models/Refund.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
class Refund extends Model
{
   use HasFactory;

   protected $fillable = [
      'no_rekening_refund',
      'refund_status',
      'refund_payment_proof',
   ];
   public function downPayment()
   {
      return $this->hasOne(DownPayment::class);
   }
}
```

Models/Review.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Factories\HasFactory;
class Review extends Model
{
   use HasFactory;
   protected $fillable = [
      'user_id',
      'sales_record_id',
      'car_id',
      'rating',
      'comment',
      'photo_review',
      'status',
   ];
   // defining the relationship with the User model
   public function user()
   {
      return $this->belongsTo(User::class);
   }

   // defining the relationship with the Car model
   public function car()
   {
      return $this->belongsTo(Car::class);
   }
   // defining the relationship with the SalesRecord model
   public function salesRecord()
   {
      return $this->belongsTo(SalesRecord::class);
   }
}
```

Models/SalesRecord.php

```
<?php
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
class SalesRecord extends Model
{
   use HasFactory;
   protected $fillable = [
      'car_id',
      'buyer_id',
      'seller_id',
      'sale_price',
      'sale_date',
      'status',
   ];
```

```
    // Define the relationship with the Car model
    public function car()
    {
        return $this->belongsTo(Car::class);
    }
    // Define the relationship with the User model
    public function buyer()
    {
        return $this->belongsTo(User::class, 'buyer_id');
    }

    public function saler()
    {
        return $this->belongsTo(User::class, 'seller_id');
    }
    public function reviews()
    {
        return $this->hasOne(Review::class);
    }
}
```

Models/User.php

```
<?php
namespace App\Models;
// use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
class User extends Authenticatable
{
    /** @use HasFactory<\Database\Factories\UserFactory> */
    use HasFactory, Notifiable;
    /**
     * The attributes that are mass assignable.
     *
     * @var list<string>
     */
    protected $fillable = [
        'name',
        'email',
        'password',
        'phone',
        'address',
        'role',
    ];
    protected $hidden = [
        'password',
        'remember_token',
    ];
    protected function casts(): array
```

```
    {
        return [
            'email_verified_at' => 'datetime',
            'password' => 'hashed',
        ];
    }
    // Define the relationship with the Offer model
    public function offer()
    {
        return $this->hasMany(Offer::class);
    }
    // Define the relationship with the Car model
    public function car()
    {
        return $this->hasMany(Car::class);
    }
    // Define the relationship with the SalesRecord model
    public function purchases()
    {
        return $this->hasMany(SalesRecord::class, 'buyer_id');
    }
    public function sales()
    {
        return $this->hasMany(SalesRecord::class, 'seller_id');
    }
    // Define the relationship with the OfferRecord model
    public function purchasesOfferRecord()
    {
        return $this->hasMany(OfferRecord::class, 'buyer_id');
    }
    public function salesOfferRecord()
    {
        return $this->hasMany(OfferRecord::class, 'seller_id');
    }
    // Define the relationship with the DownPayment model
    public function downPayment()
    {
        return $this->hasMany(DownPayment::class);
    }
    // Define the relationship with the Review model
    public function review()
    {
        return $this->hasMany(Review::class);
    }
    // Define the relationship with the Car_Photo model
    public function carPhoto()
    {
        return $this->hasMany(CarPhoto::class);
    }
}
```

Providers/AppServiceProvider.php

```
<?php
namespace App\Providers;
use Carbon\Carbon;
use Illuminate\Support\ServiceProvider;
class AppServiceProvider extends ServiceProvider
{
    /**
     * Register any application services.
     */
    public function register(): void
    {
        //
    }
    /**
     * Bootstrap any application services.
     */
    public function boot(): void
    {
        Carbon::setLocale('id');
    }
}
```

bootstrap/app.php

```
<?php
use Illuminate\Foundation\Application;
use Illuminate\Foundation\Configuration\Exceptions;
use Illuminate\Foundation\Configuration\Middleware;
return Application::configure(basePath: dirname(__DIR__))
    ->withRouting(
        web: __DIR__.'/../routes/web.php',
        api: __DIR__.'/../routes/api.php',
        commands: __DIR__.'/../routes/console.php',
        health: '/up',
    )
    ->withMiddleware(function (Middleware $middleware) {
        $middleware->alias([
            'checkrole' => \App\Http\Middleware\CheckRole::class,
        ]);
})
    ->withExceptions(function (Exceptions $exceptions) {
        //
    })->create();
```

bootstrap/providers.php

```
<?php
return [
    App\Providers\AppServiceProvider::class,
];
```

config/midtrans.php

```
<?php
return [
    'server_key' => env('MIDTRANS_SERVER_KEY'),
    'client_key' => env('MIDTRANS_CLIENT_KEY'),
    'is_production' => env('MIDTRANS_IS_PRODUCTION', false),
];
```

database/migrations/0001_01_01_000000_create_users_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('name',30);
            $table->string('email',30)->unique();
            $table->timestamp('email_verified_at')->nullable();
            $table->string('password');
            $table->string('phone',15)->nullable();
            $table->text('address')->nullable();
            $table->enum('role', ['admin', 'user'])->default('user');
            $table->rememberToken();
            $table->timestamps();
        });
        Schema::create('password_reset_tokens', function (Blueprint $table) {
            $table->string('email')->primary();
            $table->string('token');
            $table->timestamp('created_at')->nullable();
        });
        Schema::create('sessions', function (Blueprint $table) {
            $table->string('id')->primary();
            $table->foreignId('user_id')->nullable()->index();
            $table->string('ip_address', 45)->nullable();
            $table->text('user_agent')->nullable();
            $table->longText('payload');
            $table->integer('last_activity')->index();
        });
    }
    /**
     * Reverse the migrations.
     */
```

```
    public function down(): void
    {
        Schema::dropIfExists('users');
        Schema::dropIfExists('password_reset_tokens');
        Schema::dropIfExists('sessions');
    }
};
```

database/migrations/0001_01_01_000001_create_cache_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('cache', function (Blueprint $table) {
            $table->string('key')->primary();
            $table->mediumText('value');
            $table->integer('expiration');
        });
        Schema::create('cache_locks', function (Blueprint $table) {
            $table->string('key')->primary();
            $table->string('owner');
            $table->integer('expiration');
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('cache');
        Schema::dropIfExists('cache_locks');
    }
};
```

database/migrations/2025_04_27_0000111_create_refunds_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
```

```
    */
    public function up(): void
    {
        Schema::create('refunds', function (Blueprint $table) {
            $table->id();
            $table->string('no_rekening_refund');
            $table->enum('refund_status',['refund','no_refund']);
            $table->string('refund_payment_proof');
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('refunds');
    }
};
```

database/migrations/2025_04_27_015941_create_cars_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('cars', function (Blueprint $table) {
            $table->id();
            // relationship with users table
            $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
            $table->string('brand',50);
            $table->string('model',50);
            $table->string('year',5);
            $table->integer('price');
            $table->enum('transmission', ['manual', 'automatic']);
            $table->text('description')->nullable();
            $table->date('service_history')->nullable();
            $table->string('fuel_type',10);
            $table->string('mileage',10);
            $table->string('color',15);
            $table->date('tax');
            $table->integer('engine');
            $table->integer('seat');
            $table->boolean('bpkb');
```

```
            $table->boolean('spare_key');
            $table->boolean('manual_book');
            $table->boolean('service_book');
            $table->enum('sale_type',['user','showroom']);
            $table->enum('status',['available','pending_check','sold','under_review']);
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('cars');
    }
};
```

database/migrations/2025_04_27_021547_create_offers_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('offers', function (Blueprint $table) {
            $table->id();
            $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
            $table->string('brand',50);
            $table->string('model',50);
            $table->integer('year');
            $table->integer('mileage');
            $table->decimal('offered_price', 15, 2);
            $table->text('location_inspection')->nullable();
            $table->enum('status', ['pending', 'accepted', 'rejected','sold'])->default('pending');
            $table->dateTime('inspection_date')->nullable();
            $table->text('note')->nullable();
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('offers');
```

```
    }
};
```

database/migrations/2025_04_27_044106_create_sales_records_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('sales_records', function (Blueprint $table) {
            $table->id();
            $table->foreignId('car_id')->constrained('cars')->onDelete('cascade');
            $table->foreignId('buyer_id')->constrained('users')->onDelete('cascade');
            $table->foreignId('seller_id')->constrained('users')->onDelete('cascade');
            $table->decimal('sale_price', 15, 2);
            $table->dateTime('sale_date');
            $table->enum('status', ['completed', 'pending', 'cancelled']);
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('sales_records');
    }
};
```

database/migrations/2025_04_27_044406_create_down_payments_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('down_payments', function (Blueprint $table) {
            $table->id();
            $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
```

```
            $table->foreignId('car_id')->constrained('cars')->onDelete('cascade');
            $table->string('order_id', 100)->unique()->nullable();
            $table->decimal('amount', 15, 2);
            $table->enum('payment_status', ['pending', 'confirmed', 'cancelled', 'expired'])-
>default('pending');
            $table->string('payment_method',20)->nullable();
            $table->dateTime('appointment_date');
            $table->dateTime('payment_date')->nullable();
            $table->string('snap_token', 255)->unique()->nullable();
            // Perbaikan foreign key refund_id
            $table->unsignedBigInteger('refund_id')->nullable();
            $table->foreign('refund_id')->references('id')->on('refunds')->onDelete('set null');
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('down_payments');
    }
};
```

database/migrations/2025_04_27_044406_create_reviews_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('reviews', function (Blueprint $table) {
            $table->id();
            $table->foreignId('user_id')->constrained('users')->onDelete('cascade');
            $table->foreignId('sales_record_id')->nullable()->constrained('sales_records')-
>onDelete('cascade');
            $table->foreignId('car_id')->nullable()->constrained('cars')->onDelete('cascade');
            $table->integer('rating')->nullable()->default(5);
            $table->text('comment')->nullable();
            $table->string('photo_review',255)->nullable();
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
```

```
    */
    public function down(): void
    {
        Schema::dropIfExists('reviews');
    }
};
```

database/migrations/2025_04_27_071045_create_car_photos_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('car_photos', function (Blueprint $table) {
            $table->id();
            $table->foreignId('car_id')->constrained('cars')->onDelete('cascade');
            $table->string('photo_url',255);
            $table->integer('number');
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('car_photos');
    }
};
```

database/migrations/2025_05_15_224602_create_car_types_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('car_types', function (Blueprint $table) {
            $table->id();
```

```
            $table->string('tahun',5);
            $table->string('brand',50);
            $table->string('model',50);
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('car_types');
    }
};
```

database/migrations/2025_06_17_154403_create_offer_records_table.php

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('offer_records', function (Blueprint $table) {
            $table->id();
            $table->foreignId('offer_id')->constrained('offers')->onDelete('cascade');
            $table->foreignId('seller_id')->constrained('users')->onDelete('cascade');
            $table->foreignId('buyer_id')->constrained('users')->onDelete('cascade');
            $table->decimal('sale_price', 15, 2);
            $table->dateTime('sale_date');
            $table->enum('status', ['completed', 'pending', 'cancelled']);
            $table->timestamps();
        });
    }
    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('offer_records');
    }
};
```

seeders/CarSeeder.php

```
<?php
namespace Database\Seeders;
```

```
use App\Models\Car;
use App\Models\CarPhoto;
use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
class CarSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        $car = Car::create([
            'user_id' => 1,
            'brand' => 'Toyota',
            'model' => 'AVANZA VELOZ',
            'year' => '2021',
            'price' => '100000000',
            'transmission' => 'automatic',
            'description' => 'Toyota Alphard menjadi salah satu mobil MPV mewah dengan
angka penjualan yang cukup tinggi.
                Ukuran yang besar dengan interior yang mewah menjadikan mobil ini sebagai
mobil standar bagi
                kalangan menengah atas di Indonesia. Segera jadwalkan untuk informasi lebih
lanjut.',
            'service_history' => '2024-05-20',
            'fuel_type' => 'bensin',
            'mileage' => '100000',
            'color' => 'silver',
            'tax' => '2025-05-20',
            'engine' => '2500',
            'bpkb'=> true,
            'spare_key'=> true,
            'manual_book'=> true,
            'service_book'=> true,
            'sale_type' => 'showroom',
            'seat' => '4',
            'status' => 'available',
        ]);
        foreach (range(1, 5) as $i) {
            CarPhoto::create([
                'car_id' => $car->id,
                'photo_url' => "$i.jpg",
                'number' => $i,
            ]);
        }
        $car = Car::create([
            'user_id' => 1,
            'brand' => 'Dhaihatsu',
            'model' => 'Ayla',
            'year' => '2021',
```

```
            'price' => '90000000',
            'transmission' => 'automatic',
            'description' => 'Toyota Alphard menjadi salah satu mobil MPV mewah dengan
angka penjualan yang cukup tinggi.
                Ukuran yang besar dengan interior yang mewah menjadikan mobil ini sebagai
mobil standar bagi
                kalangan menengah atas di Indonesia. Segera jadwalkan untuk informasi lebih
lanjut.',
            'service_history' => '2024-05-20',
            'fuel_type' => 'bensin',
            'mileage' => '100000',
            'color' => 'silver',
            'tax' => '2025-05-20',
            'engine' => '2500',
            'bpkb'=> true,
            'spare_key'=> true,
            'manual_book'=> true,
            'service_book'=> true,
            'sale_type' => 'showroom',
            'seat' => '4',
            'status' => 'available',
        ]);
        foreach (range(1, 5) as $i) {
            CarPhoto::create([
                'car_id' => $car->id,
                'photo_url' => "$i.jpg",
                'number' => $i,
            ]);
        }
    }
}
```

seeders/CarTypeSeeder.php

```
<?php
namespace Database\Seeders;
use App\Models\CarType;
use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\File;
class CarTypeSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run()
    {
        $filePath =
database_path('seeders/data/used_car_tahun_brand_model_unique_sorted.csv');
        if (!File::exists($filePath)) {
            $this->command->error("File CSV tidak ditemukan di: {$filePath}");
```

```
            return;
        }
        $csvData = array_map('str_getcsv', file($filePath));
        $header = array_shift($csvData);
        foreach ($csvData as $row) {
            $data = array_combine($header, $row);
            CarType::create([
                'tahun' => $data['tahun'],
                'brand' => $data['brand'],
                'model' => $data['model'],
                'created_at' => now(),
                'updated_at' => now(),
            ]);
        }
        $this->command->info('Data cars_type berhasil di-seed dari CSV.');
    }
}
```

seeders/ReviewSeeder.php

```
<?php
namespace Database\Seeders;
use App\Models\Review;
use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
class ReviewSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        for ($i = 1; $i <= 15; $i++) {
            Review::create([
                'user_id' => 2,
                'sales_record_id' => null,
                'car_id' => null,
                'rating' => 5,
                'photo_review' => "$i.jpeg",
            ]);
        }
    }
}
```

seeders/UsersSeeder.php

```
<?php
namespace Database\Seeders;
use App\Models\User;
use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\Hash;
```

```
class UsersSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        User::create([
            'name' => 'Admin Showroom',
            'email' => 'admin@mail.com',
            'password' => Hash::make('admin123'),
            'role' => 'admin',
        ]);
        User::create([
            'name' => 'User Bariq',
            'email' => 'bariq@mail.com',
            'password' => Hash::make('bariq123'),
            'phone' => '081220733216',
            'address' => 'Depok',
            'role' => 'user',
        ]);
    }
}
```

resources/views/admin/cars/partials/modal_content.blade.php

```
<div class="modal-header">
  <h5 class="modal-title" id="carModalLabel">Car Detail: {{ $car->brand }} {{ $car-
>model }}</h5>
  <button type="button" class="btn-close" data-bs-dismiss="modal" aria-
label="Close"></button>
</div>
<div class="modal-body">
  <table class="table table-bordered text-capitalize">
    <tr><th>ID</th><td>{{ $car->id }}</td></tr>
    <tr><th>User</th><td>{{ $car->user->name ?? 'N/A' }}</td></tr>
    <tr><th>Brand</th><td>{{ $car->brand }}</td></tr>
    <tr><th>Model</th><td>{{ $car->model }}</td></tr>
    <tr><th>Year</th><td>{{ $car->year }}</td></tr>
    <tr><th>Price</th><td>{{ 'Rp '.number_format($car->price, 0, ',', '.') }}</td></tr>
    <tr><th>Transmission</th><td>{{ $car->transmission }}</td></tr>
    <tr><th>Description</th><td>{{ $car->description }}</td></tr>
    <tr><th>Fuel Type</th><td>{{ $car->fuel_type }}</td></tr>
    <tr><th>Mileage</th><td>{{ number_format($car->mileage, 0, ',', '.') }}</td></tr>
    <tr><th>Service History</th><td>{{ \Carbon\Carbon::parse($car->service_history)-
>translatedFormat('d F Y') }}</td></tr>
    <tr><th>Tax</th><td>{{ \Carbon\Carbon::parse($car->tax)->translatedFormat('d F
Y') }}</td></tr>
    <tr><th>Tax</th><td>{{ $car->seat . ' People' }}</td></tr>
    <tr><th>Tax</th><td>{{ number_format($car->engine, 0, ',', '.') . ' CC' }}</td></tr>
    <tr><th>Color</th><td>{{ $car->color }}</td></tr>
```

```
    <tr><th>BPKB</th><td>{{ $car->bpkb ? 'Yes' : 'No' }}</td></tr>
    <tr><th>Spare Key</th><td>{{ $car->spare_key ? 'Yes' : 'No' }}</td></tr>
    <tr><th>Manual Book</th><td>{{ $car->manual_book ? 'Yes' : 'No' }}</td></tr>
    <tr><th>Service Book</th><td>{{ $car->service_book ? 'Yes' : 'No' }}</td></tr>
    <tr><th>Sale Type</th><td>{{ $car->sale_type }}</td></tr>
    <tr><th>Status</th><td>{{ $car->status }}</td></tr>
    <tr><th>Created At</th><td>{{ \Carbon\Carbon::parse($car->created_at)-
>translatedFormat('d F Y H:i') }}</td></tr>
    <tr><th>Updated At</th><td>{{ \Carbon\Carbon::parse($car->updated_at )-
>translatedFormat('d F Y H:i') }}</td></tr>
  </table>
  <h6 class="fw-bold mb-2">Foto Mobil</h6>
  @if ($car->carPhoto->count())
    <div class="row mb-3">
      @foreach ($car->carPhoto->sortBy('number') as $photo)
        <div class="col-md-4 text-center">
          @if (Storage::disk('public')->exists('car_photos/' . $photo->photo_url))
            <img src="{{ asset('storage/car_photos/' . $photo->photo_url) }}"
                 alt="Car Photo"
                 class="img-fluid rounded shadow-sm mb-2"
                 style="max-height: 180px;">
          @else
            <p class="text-danger fw-bold">
                 File Terhapus, Silahkan upload ulang.
            </p>
          @endif
        </div>
      @endforeach
    </div>
  @else
    <p class="text-muted">Belum ada foto yang diunggah.</p>
  @endif
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Close</button>
</div>
```

resources/views/admin/cars/partials/update_status_modal.blade.php

```
<div class="modal-header">
  <h5 class="modal-title">Update Car Status</h5>
  <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
</div>
<div class="modal-body">
  <form id="form-update-status">
    @csrf
    <input type="hidden" name="car_id" id="modal-car-id" value="{{ $car->id }}">
    <div class="mb-3">
      <label for="new-status" class="form-label">Select Status</label>
      <select name="status" id="new-status" class="form-select">
```

```
        @foreach(['available', 'pending_check', 'sold', 'under_review'] as $status)
          <option value="{{ $status }}" {{ $car->status === $status ? 'selected' : '' }}>
            {{ ucfirst($status) }}
          </option>
        @endforeach
      </select>
    </div>
  </form>
</div>
<div class="modal-footer">
  <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Cancel</button>
  <button type="submit" form="form-update-status" class="btn btn-
primary">Save</button>
</div>
```

resources/views/admin/cars/create.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css"
rel="stylesheet" />
<link href="https://cdn.jsdelivr.net/npm/select2-bootstrap-5-theme@1.2.0/dist/select2-
bootstrap-5-theme.min.css"
   rel="stylesheet" />
   <style>
      .check-badge {
      display: flex;
      align-items: center;
      gap: 6px;
      padding: 8px 12px;
      border-radius: 8px;
      border: 1px solid #ccc;
      background-color: #f9f9f9;
      cursor: pointer;
      transition: 0.2s;
      user-select: none;
   }
   .check-badge:hover {
      background-color: #e2e6ea;
   }
   .check-badge input {
      display: none;
   }
   .check-badge.active {
      background-color: #cce5ff;
      border-color: #66afe9;
   }
   .check-badge i {
      font-size: 1rem;
```

```
    }
    .check-toggle-btn {
        margin-bottom: 10px;
    }
    </style>
@endpush
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Create New Stock Car </div>
            </div>
            <!--end::Header-->
            <!--begin::Error Alert-->
            @if ($errors->any())
            <div class="alert alert-danger">
                <ul class="mb-0">
                    @foreach ($errors->all() as $error)
                    <li>{{ $error }}</li>
                    @endforeach
                </ul>
            </div>
            @endif
            <!--end::Error Alert-->
            <!--begin::Form-->
            <form action="{{ route('admin.cars.store') }}" method="POST"
enctype="multipart/form-data">
                @csrf
                <div class="card-body">
                    <div class="mb-3">
                        <label for="user_id" class="form-label">User</label>
                        <select class="form-control select2 @error('user_id') is-invalid @enderror
" name="user_id"
                            required>
                            <option value="">-- Pilih User --</option>
                            @foreach ($user as $u)
                            <option value="{{ $u->id }}"
                                {{ old('user_id', auth()->user()->id) == $u->id ? 'selected' : '' }}>
                                {{ $u->name }} ({{ $u->email }})
                            </option>
                            @endforeach
                        </select>
                    </div>
                    <div class="mb-3">
                        <label for="brand" class="form-label">Brand</label>
                        <select name="brand" id="brand" class="form-control" required>
                            <option value="">-- Pilih Brand --</option>
                            @foreach ($brands as $brand)
```

```
                <option value="{{ $brand }}">{{ $brand }}</option>
                @endforeach
            </select>
            @error('brand')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
            </span>
            @enderror
        </div>
        <div class="mb-3">
            <label for="model" class="form-label">Model</label>
            <select name="model" id="model" class="form-control" required
disabled>
                <option value="">-- Pilih Model --</option>
            </select>
            @error('model')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
            </span>
            @enderror
        </div>
        <div class="mb-3">
            <label for="year" class="form-label">Year</label>
            <select class="form-control select2Year @error('year') is-invalid
@enderror" name="year"
                id="year" required>
                <option value="">-- Pilih Tahun --</option>
                @for ($year = 2025; $year >= 2004; $year--)
                <option value="{{ $year }}" {{ old('year') == $year ? 'selected' : ''
}}>{{ $year }}
                </option>
                @endfor
            </select>
            @error('year')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
            </span>
            @enderror
        </div>
        <div class="mb-3">
            <label for="price_format" class="form-label">Price</label>
            <input type="text" class="form-control @error('price') is-invalid
@enderror" id="price_format"
                value="{{ old('price') ? number_format(old('price'), 0, ',', '.') : '' }}"
required />
            <input type="hidden" name="price" id="price" value="{{ old('price') }}"
/>
            @error('price')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
```

```
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="transmission" class="form-label">Transmission</label>
                <select name="transmission" id="transmission"
                    class="form-control @error('transmission') is-invalid @enderror"
required>
                    <option value="">-- Pilih Transmission --</option>
                    <option value="automatic" {{ old('transmission') == 'automatic' ?
'selected' : '' }}>
                        Automatic</option>
                    <option value="manual" {{ old('transmission') == 'manual' ? 'selected' :
'' }}>Manual
                    </option>
                </select>
                @error('transmission')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="description" class="form-label">Description</label>
                <textarea cols="10" rows="2" class="form-control @error('description')
is-invalid @enderror"
                    id="description" name="description" required>{{ old('description')
}}</textarea>
                @error('description')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="fuel_type" class="form-label">Fuel Type</label>
                <input type="text" class="form-control @error('fuel_type') is-invalid
@enderror" id="fuel_type"
                    name="fuel_type" value="{{ old('fuel_type') }}" required />
                @error('fuel_type')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="mileage" class="form-label">Mileage</label>
                <input type="text" class="form-control @error('mileage') is-invalid
@enderror"
                    id="mileage_format"
```

```
                    value="{{ old('mileage') ? number_format(old('mileage'), 0, ',', '.') : ''
}}" required />
                <input type="hidden" name="mileage" id="mileage" value="{{
old('mileage') }}" />
                @error('mileage')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="color" class="form-label">Color</label>
                <input type="text" class="form-control @error('color') is-invalid
@enderror" id="color"
                    name="color" value="{{ old('color') }}" required />
                @error('color')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="service_history" class="form-label">Last Service</label>
                <input type="date" class="form-control @error('service_history') is-
invalid @enderror"
                    id="service_history" name="service_history" value="{{
old('service_history') }}" required />
                @error('service_history')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="tax" class="form-label">Tax</label>
                <input type="date" class="form-control @error('tax') is-invalid
@enderror" id="tax" name="tax"
                    value="{{ old('tax') }}" required />
                @error('tax')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="engine" class="form-label">Engine</label>
                <input type="number" class="form-control @error('engine') is-invalid
@enderror" id="engine"
                    name="engine" value="{{ old('engine') }}" required />
                @error('engine')
```

```
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="seat" class="form-label">Seat</label>
                <input type="number" class="form-control @error('seat') is-invalid
@enderror" id="seat"
                    name="seat" value="{{ old('seat') }}" required />
                @error('seat')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label class="form-label">Completeness of Documents &
Accessories</label>
                <div class="d-flex flex-wrap gap-2">
                    <label class="check-badge {{ old('bpkb') ? 'active' : '' }}">
                        <input type="checkbox" name="bpkb" id="bpkb" value="1"
                            {{ old('bpkb') ? 'checked' : '' }}>
                        <i class="bi bi-check-square"></i> BPKB
                    </label>
                    <label class="check-badge {{ old('spare_key') ? 'active' : '' }}">
                        <input type="checkbox" name="spare_key" id="spare_key"
value="1"
                            {{ old('spare_key') ? 'checked' : '' }}>
                        <i class="bi bi-key"></i> Kunci Cadangan
                    </label>
                    <label
                        class="check-badge {{ old('manual_book') ? 'active' : '' }}">
                        <input type="checkbox" name="manual_book" id="manual_book"
value="1"
                            {{ old('manual_book') ? 'checked' : '' }}>
                        <i class="bi bi-book"></i> Manual Book
                    </label>
                    <label
                        class="check-badge {{ old('service_book') ? 'active' : '' }}">
                        <input type="checkbox" name="service_book" id="service_book"
value="1"
                            {{ old('service_book') ? 'checked' : '' }}>
                        <i class="bi bi-journal-check"></i> Service Book
                    </label>
                </div>
                <button type="button" class="btn btn-sm btn-outline-primary mt-2 check-
toggle-btn"
                    onclick="toggleChecklist()">☑ All</button>
            </div>
```

```
                <div class="mb-3">
                    <label for="sale_type" class="form-label">Sale Type</label>
                    <select name="sale_type" id="sale_type"
                        class="form-control @error('sale_type') is-invalid @enderror"
required>
                        <option value="">-- Pilih sale_type --</option>
                        <option value="showroom" {{ old('sale_type') == 'showroom' ?
'selected' : '' }}>Showroom
                        </option>
                        <option value="user" {{ old('sale_type') == 'user' ? 'selected' : ''
}}>User</option>
                    </select>
                    @error('sale_type')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <label for="photos" class="form-label">Photo Mobil</label>
                <div id="photo-wrapper" class="row sortable-photo-wrapper">
                    {{-- Template satu photo --}}
                    <div class="col-md-3 mb-3 photo-group">
                        <div class="card position-relative text-center h-100 shadow-sm p-2">
                            <img src="{{ asset('image/NoImage.png') }}" class="img-preview
img-fluid mb-2"
                                style="max-height: 150px;">
                            <input type="file" name="photos[]" class="form-control photo-input
mb-2"
                                accept="image/*" required>
                            <input type="hidden" name="photo_order[]" value="1"
class="photo-order">
                            <button type="button" class="btn btn-danger btn-sm w-100 btn-
remove-photo">
                                <i class="bi bi-trash"></i> Hapus
                            </button>
                        </div>
                    </div>
                </div>
                <div class="d-flex justify-content-between mt-2">
                    <button type="button" id="btn-add-photo" class="btn btn-info px-4">+
Tambah Foto</button>
                </div>
            </div>
            <div class="card-footer">
                <button type="submit" class="btn btn-primary">Submit</button>
            </div>
        </form>
        <!--end::Form-->
    </div>
```

```
    </div>
    <!--end::Container-->
</div>
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/js/select2.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/sortablejs@1.15.0/Sortable.min.js"></script>
<script>
    $(document).ready(function () {
        $('.select2, .select2Year').select2({
            theme: 'bootstrap-5',
            width: '100%',
            allowClear: true,
            placeholder: '-- Pilih --',
        });
        $('#brand, #model').select2({
            theme: 'bootstrap-5',
            width: '100%',
            allowClear: true,
            placeholder: '-- Pilih --',
        });
        $('#brand').on('change', function () {
            const brand = $(this).val();
            $('#model').prop('disabled', true).empty().append('<option
value="">Loading...</option>');
            if (brand) {
                $.get('/api/models', {
                    brand: brand
                }, function (data) {
                    $('#model').empty().append('<option value="">-- Pilih Model --</option>');
                    $.each(data, function (i, model) {
                        $('#model').append(
                            `<option value="${model}">${model}</option>`);
                    });
                    $('#model').prop('disabled', false).trigger('change');
                });
            } else {
                $('#model').empty().append('<option value="">-- Pilih Model --
</option>').prop(
                    'disabled', true);
            }
        });
    });
    // Format input price to Rupiah
    function formatRupiah(angka) {
        return angka.replace(/\D/g, "")
            .replace(/\B(?=(\d{3})+(?!\d))/g, ".");
    }
    document.addEventListener('DOMContentLoaded', function () {
```

```
    const inputFormatted = document.getElementById('price_format');
    const inputHidden = document.getElementById('price');
    inputFormatted.addEventListener('input', function () {
        let clean = this.value.replace(/\D/g, "");
        this.value = formatRupiah(clean);
        inputHidden.value = clean;
    });
    const inputMileageFormatted = document.getElementById('mileage_format');
    const inputMileageHidden = document.getElementById('mileage');
    inputMileageFormatted.addEventListener('input', function () {
        let clean = this.value.replace(/\D/g, "");
        this.value = formatRupiah(clean);
        inputMileageHidden.value = clean;
    });
});
// Menangani penambahan dan penghapusan foto
document.addEventListener('DOMContentLoaded', function () {
    const wrapper = document.getElementById('photo-wrapper');
    const btnAdd = document.getElementById('btn-add-photo');
    // Tambah foto baru
    btnAdd.addEventListener('click', function () {
        const newCol = document.createElement('div');
        newCol.classList.add('col-md-3', 'mb-3', 'photo-group');
        newCol.innerHTML = `
        <div class="card position-relative text-center h-100 shadow-sm p-2">
            <img src="{{ asset('image/NoImage.png') }}" class="img-preview img-fluid
mb-2" style="max-height: 150px;">
            <input type="file" name="photos[]" class="form-control photo-input mb-2"
accept="image/*" required>
            <input type="hidden" name="photo_order[]" value="0" class="photo-order">
            <button type="button" class="btn btn-danger btn-sm w-100 btn-remove-
photo">
                <i class="bi bi-trash"></i> Hapus
            </button>
        </div>
    `;
        wrapper.appendChild(newCol);
    });
    // Preview image
    wrapper.addEventListener('change', function (e) {
        if (e.target.classList.contains('photo-input')) {
            const preview = e.target.closest('.photo-group').querySelector('.img-preview');
            const file = e.target.files[0];
            if (file) {
                const reader = new FileReader();
                reader.onload = (e) => preview.src = e.target.result;
                reader.readAsDataURL(file);
            }
        }
    });
```

```
        // Hapus photo group
        wrapper.addEventListener('click', function (e) {
            if (e.target.closest('.btn-remove-photo')) {
                const photo = e.target.closest('.photo-group');
                photo.remove();
            }
        });
        // Drag & Drop dengan SortableJS
        new Sortable(wrapper, {
            animation: 150,
            draggable: '.photo-group',
            onEnd: updatePhotoOrder,
        });
        // Update order saat submit
        document.querySelector('form').addEventListener('submit', updatePhotoOrder);
        function updatePhotoOrder() {
            const groups = wrapper.querySelectorAll('.photo-group');
            groups.forEach((group, index) => {
                group.querySelector('.photo-order').value = index + 1;
            });
        }
    });
    // ceklis button kelengkapan
    function toggleChecklist() {
        const labels = document.querySelectorAll('.check-badge');
        const allChecked = [...labels].every(label => label.classList.contains('active'));
        labels.forEach(label => {
            const checkbox = label.querySelector('input[type="checkbox"]');
            checkbox.checked = !allChecked;
            label.classList.toggle('active', !allChecked);
        });
    }
    document.addEventListener('DOMContentLoaded', () => {
        document.querySelectorAll('.check-badge input[type="checkbox"]').forEach(cb => {
            cb.addEventListener('change', function () {
                this.closest('.check-badge').classList.toggle('active', this.checked);
            });
        });
    });
</script>
@endpush
@endsection
```

resources/views/admin/cars/edit.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/css/select2.min.css" rel="stylesheet" />
```

```
<link href="https://cdn.jsdelivr.net/npm/select2-bootstrap-5-
theme@1.2.0/dist/select2-bootstrap-5-theme.min.css" rel="stylesheet" />
<style>
        .check-badge {
        display: flex;
        align-items: center;
        gap: 6px;
        padding: 8px 12px;
        border-radius: 8px;
        border: 1px solid #ccc;
        background-color: #f9f9f9;
        cursor: pointer;
        transition: 0.2s;
        user-select: none;
    }
    .check-badge:hover {
        background-color: #e2e6ea;
    }
    .check-badge input {
        display: none;
    }
    .check-badge.active {
        background-color: #cce5ff;
        border-color: #66afe9;
    }
    .check-badge i {
        font-size: 1rem;
    }
    .check-toggle-btn {
        margin-bottom: 10px;
    }
    </style>
@endpush
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Edit #{{ $car->id }} Stock Car </div>
            </div>
            <!--end::Header-->
            <!--begin::Error Alert-->
            @if ($errors->any())
            <div class="alert alert-danger">
                <ul class="mb-0">
                    @foreach ($errors->all() as $error)
                    <li>{{ $error }}</li>
                    @endforeach
                </ul>
```

```
        </div>
        @endif
        <!--end::Error Alert-->
        <!--begin::Form-->
        <form action="{{ route('admin.cars.update', $car->id) }}"
method="POST" enctype="multipart/form-data">
          @csrf
          @method('PUT')
          <div class="card-body">
            <div class="mb-3">
              <label for="user_id" class="form-label">User</label>
              <select class="form-control select2 @error('user_id') is-invalid
@enderror" name="user_id" required>
                <option value="">-- Pilih User --</option>
                @foreach ($user as $u)
                  <option value="{{ $u->id }}"
                    {{ old('user_id', $car->user_id) == $u->id ? 'selected' : '' }}>
                    {{ $u->name }} ({{ $u->email }})
                  </option>
                @endforeach
              </select>
              @error('user_id')
              <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
              </span>
              @enderror
            </div>
            <div class="mb-3">
              <label for="brand" class="form-label">Brand</label>
              <select name="brand" id="brand" class="form-control" required>
                <option value="">-- Pilih Brand --</option>
                @foreach ($brands as $brand)
                  <option value="{{ $brand }}" {{ old('brand', $car->brand)
== $brand ? 'selected' : '' }}>
                    {{ $brand }}
                  </option>
                @endforeach
              </select>

              @error('brand')
              <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
              </span>
              @enderror
            </div>
            <div class="mb-3">
              <label for="model" class="form-label">Model</label>
              <select name="model" id="model" class="form-control" required
disabled>
                <option value="">-- Pilih Model --</option>
```

```
                </select>
                @error('model')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="year" class="form-label">Year</label>
                <select class="form-control select2Year @error('year') is-invalid
@enderror" name="year" id="year" required>
                    <option value="">-- Pilih Tahun --</option>
                    @for ($year = 2025; $year >= 2004; $year--)
                        <option value="{{ $year }}" {{ old('year', $car->year) ==
$year ? 'selected' : '' }}>{{ $year }}</option>
                    @endfor
                </select>
                @error('year')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="price_format" class="form-label">Price</label>
                <input type="text" class="form-control @error('price') is-invalid
@enderror" id="price_format"
                        value="{{ old('price', $car->price) ?
number_format(old('price',$car->price), 0, ',', '.') : '' }}" required />

                <input type="hidden" name="price" id="price" value="{{
old('price',$car->price) }}" />

                @error('price')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            {{-- {{ dd($car->transmission) }} --}}
            <div class="mb-3">
                <label for="transmission" class="form-
label">Transmission</label>
                <select name="transmission" id="transmission" class="form-
control select2 @error('transmission') is-invalid @enderror" required>
                    <option value="">-- Pilih Transmission --</option>
                    <option value="automatic" {{ old('transmission', $car-
>transmission) == 'automatic' ? 'selected' : '' }}>Automatic</option>
                    <option value="manual" {{ old('transmission', $car-
>transmission) == 'manual' ? 'selected' : '' }}>Manual</option>
```

```
                    </select>
                    @error('transmission')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="description" class="form-label">Description</label>
                    <textarea cols="10" rows="2" class="form-control
@error('description') is-invalid @enderror"
                        id="description" name="description" required>{{
old('description',$car->description) }}</textarea>
                    @error('description')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="service_history" class="form-label">Last
Service</label>
                    <input type="date" class="form-control @error('service_history')
is-invalid @enderror"
                        id="service_history" name="service_history" value="{{
old('service_history', $car->service_history) }}" required />
                    @error('service_history')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="fuel_type" class="form-label">Fuel Type</label>
                    <input type="text" class="form-control @error('fuel_type') is-
invalid @enderror"
                        id="fuel_type" name="fuel_type" value="{{ old('fuel_type',
$car->fuel_type) }}" required />
                    @error('fuel_type')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="color" class="form-label">Color</label>
                    <input type="text" class="form-control @error('color') is-invalid
@enderror"
                        id="color" name="color" value="{{ old('color', $car->color)
}}" required />
```

```
                @error('color')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="mileage" class="form-label">Mileage</label>
                <input type="text" class="form-control @error('mileage') is-
invalid @enderror" id="mileage_format"
                    value="{{ old('mileage',$car->mileage) ?
number_format(old('mileage',$car->mileage), 0, ',', '.') : '' }}" required />

                <input type="hidden" name="mileage" id="mileage" value="{{
old('mileage',$car->mileage) }}" />

                @error('mileage')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
             <div class="mb-3">
                <label for="tax" class="form-label">Tax</label>
                <input type="date" class="form-control @error('tax') is-invalid
@enderror" id="tax"
                    name="tax" value="{{ old('tax', $car->tax) }}" required />
                @error('tax')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="engine" class="form-label">Engine</label>
                <input type="number" class="form-control @error('engine') is-
invalid @enderror" id="engine"
                    name="engine" value="{{ old('engine', $car->engine) }}"
required />
                @error('engine')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="seat" class="form-label">Seat</label>
                <input type="number" class="form-control @error('seat') is-
invalid @enderror" id="seat"
                    name="seat" value="{{ old('seat', $car->seat) }}" required />
```

```
                @error('seat')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label class="form-label">Completeness of Documents &
Accessories</label>
                <div class="d-flex flex-wrap gap-2">
                    <label class="check-badge {{ old('bpkb', $car->bpkb ?? false) ?
'active' : '' }}">
                        <input type="checkbox" name="bpkb" id="bpkb" value="1"
                            {{ old('bpkb', $car->bpkb ?? false) ? 'checked' : '' }}>
                        <i class="bi bi-check-square"></i> BPKB
                    </label>
                    <label class="check-badge {{ old('spare_key', $car->spare_key
?? false) ? 'active' : '' }}">
                        <input type="checkbox" name="spare_key" id="spare_key"
value="1"
                            {{ old('spare_key', $car->spare_key ?? false) ? 'checked' : ''
}}>
                        <i class="bi bi-key"></i> Kunci Cadangan
                    </label>
                    <label
                        class="check-badge {{ old('manual_book', $car-
>manual_book ?? false) ? 'active' : '' }}">
                        <input type="checkbox" name="manual_book"
id="manual_book" value="1"
                            {{ old('manual_book', $car->manual_book ?? false) ?
'checked' : '' }}>
                        <i class="bi bi-book"></i> Manual Book
                    </label>
                    <label
                        class="check-badge {{ old('service_book', $car-
>service_book ?? false) ? 'active' : '' }}">
                        <input type="checkbox" name="service_book"
id="service_book" value="1"
                            {{ old('service_book', $car->service_book ?? false) ?
'checked' : '' }}>
                        <i class="bi bi-journal-check"></i> Service Book
                    </label>
                </div>
                <button type="button" class="btn btn-sm btn-outline-primary mt-2
check-toggle-btn"
                    onclick="toggleChecklist()">☑ All</button>
            </div>
            <div class="mb-3">
                <label for="sale_type" class="form-label">Sale Type</label>
```

```
                <select name="sale_type" id="sale_type" class="form-control
select2 @error('sale_type') is-invalid @enderror" required>
                    <option value="">-- Pilih sale_type --</option>
                    <option value="showroom" {{ old('sale_type',$car->sale_type)
== 'showroom' ? 'selected' : '' }}>Showroom</option>
                    <option value="user" {{ old('sale_type',$car->sale_type) ==
'user' ? 'selected' : '' }}>User</option>
                </select>
                @error('sale_type')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div id="photo-wrapper" class="row sortable-photo-wrapper">
            <label class="form-label">Photo Cars</label>
                @if ($car->carPhoto->count())
                    @foreach ($car->carPhoto->sortBy('number') as $photo)
                        <div class="col-md-4 mb-3 photo-group" data-photo-id="{{
$photo->id }}">
                            <div class="border rounded p-3 shadow-sm position-
relative text-center h-100">
                                @if (Storage::disk('public')->exists('car_photos/' .
$photo->photo_url))
                                    <img src="{{ asset('storage/car_photos/' . $photo-
>photo_url) }}" class="img-preview img-fluid mb-3" style="max-height:
150px;" />
                                    <input type="hidden" name="existing_photo_ids[]"
value="{{ $photo->id }}">
                                    <input type="hidden" name="existing_photo_order[]"
class="photo-order" value="{{ $photo->number }}">
                                    <button type="button"
                                        class="btn btn-danger w-100 btn-delete-photo"
                                        data-photo-id="{{ $photo->id }}">
                                        🗑 Hapus
                                    </button>
                                @else
                                    <p class="text-danger fw-bold">
                                        File tidak ditemukan. Silakan upload ulang.
                                    </p>
                                @endif
                            </div>
                        </div>
                    @endforeach
                @else
                    <p class="text-muted">Belum ada foto yang diunggah.</p>
                @endif
            </div>
            <div class="d-flex justify-content-between mt-2">
```

```
                <button type="button" id="btn-add-photo" class="btn btn-info px-
4">Add Photo</button>
            </div>
        </div>
        <div class="card-footer">
            <button type="submit" class="btn btn-primary">Submit</button>
        </div>
    </form>
    <!--end::Form-->
  </div>
</div>
<!--end::Container-->
</div>
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/js/select2.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/sortablejs@1.15.0/Sortable.min.js"></script>
<script>
  const selectedModel = @json(old('model', $car->model));
  const selectedBrand = @json(old('brand', $car->brand));
  $(document).ready(function() {
    $('.select2, .select2Year').select2({
      theme: 'bootstrap-5',
      width: '100%',
      allowClear: true,
      placeholder: '-- Pilih --',

    });
    $('#brand, #model').select2({
      theme: 'bootstrap-5',
      width: '100%',
      allowClear: true,
      placeholder: '-- Pilih --',
    });
    function loadModelOptions(brand, selectedModel = null) {
      if (!brand) {
        $('#model').empty().append('<option value="">-- Pilih Model --
</option>').prop('disabled', true);
        return;
      }
      $('#model').prop('disabled', true).empty().append('<option
value="">Loading...</option>');
      $.get('/api/models', { brand: brand }, function (data) {
        $('#model').empty().append('<option value="">-- Pilih Model --
</option>');
        $.each(data, function (i, model) {
          const selected = (model === selectedModel) ? 'selected' : '';
```

```
                $('#model').append(`<option value="${model}"
${selected}>${model}</option>`);
            });
            $('#model').prop('disabled', false).trigger('change');
        });
    }
    // ⏳ Load saat halaman dibuka jika brand sudah terisi
    if (selectedBrand) {
        loadModelOptions(selectedBrand, selectedModel);
    }
    // 🔁 Load ulang saat brand diubah
    $('#brand').on('change', function () {
        const brand = $(this).val();
        loadModelOptions(brand);
    });
});
// Format input price to Rupiah
function formatRupiah(angka) {
    return angka.replace(/\D/g, "")
        .replace(/\B(?=(\d{3})+(?!\d))/g, ".");
}
document.addEventListener('DOMContentLoaded', function () {
    const inputFormatted = document.getElementById('price_format');
    const inputHidden = document.getElementById('price');
    inputFormatted.addEventListener('input', function () {
        let clean = this.value.replace(/\D/g, "");
        this.value = formatRupiah(clean);
        inputHidden.value = clean;
    });
    const inputMileageFormatted =
document.getElementById('mileage_format');
    const inputMileageHidden = document.getElementById('mileage');

    inputMileageFormatted.addEventListener('input', function () {
        let clean = this.value.replace(/\D/g, "");
        this.value = formatRupiah(clean);
        inputMileageHidden.value = clean;
    });
});
// Menangani penambahan dan penghapusan foto
document.addEventListener('DOMContentLoaded', function () {
    const wrapper = document.getElementById('photo-wrapper');
    const btnAdd = document.getElementById('btn-add-photo');
    // Tambah foto baru
    btnAdd.addEventListener('click', function () {
        const newCol = document.createElement('div');
        newCol.classList.add('col-md-3', 'mb-3', 'photo-group');
        newCol.innerHTML = `
        <div class="card position-relative text-center h-100 shadow-sm p-2">
```

```
            <img src="{{ asset('image/NoImage.png') }}" class="img-preview
img-fluid mb-2" style="max-height: 150px;">
            <input type="file" name="photos[]" class="form-control photo-input
mb-2" accept="image/*" required>
            <input type="hidden" name="photo_order[]" value="0" class="photo-
order">
            <button type="button" class="btn btn-danger btn-sm w-100 btn-
remove-photo">
                <i class="bi bi-trash"></i> Hapus
            </button>
        </div>
    `;
        wrapper.appendChild(newCol);
    });
    // Preview image
    wrapper.addEventListener('change', function (e) {
        if (e.target.classList.contains('photo-input')) {
            const preview = e.target.closest('.photo-group').querySelector('.img-
preview');
            const file = e.target.files[0];
            if (file) {
                const reader = new FileReader();
                reader.onload = (e) => preview.src = e.target.result;
                reader.readAsDataURL(file);
            }
        }
    });
    // Hapus photo group
    wrapper.addEventListener('click', function (e) {
        if (e.target.closest('.btn-remove-photo')) {
            const photo = e.target.closest('.photo-group');
            photo.remove();
        }
    });
    // Drag & Drop dengan SortableJS
    new Sortable(wrapper, {
        animation: 150,
        draggable: '.photo-group',
        onEnd: updatePhotoOrder,
    });
    // Update order saat submit
    document.querySelector('form').addEventListener('submit',
updatePhotoOrder);
    function updatePhotoOrder() {
        const groups = wrapper.querySelectorAll('.photo-group');
        groups.forEach((group, index) => {
            group.querySelector('.photo-order').value = index + 1;
        });
    }
});
```

```
        // Hapus foto yang sudah ada
        $(document).on('click', '.btn-delete-photo', function () {
            const photoId = $(this).data('photo-id');
            if (!confirm('Yakin ingin menghapus foto ini?')) return;
            $.ajax({
                url: `/admin/cars-photo/${photoId}`,
                type: 'DELETE',
                data: { _token: '{{ csrf_token() }}' },
                success: function () {
                    alert('Foto berhasil dihapus');
                    location.reload(); // atau remove elemen foto dari DOM
                },
                error: function () {
                    alert('Gagal menghapus foto');
                }
            });
        });

        // ceklis button kelengkapan
        function toggleChecklist() {
            const labels = document.querySelectorAll('.check-badge');
            const allChecked = [...labels].every(label =>
    label.classList.contains('active'));
            labels.forEach(label => {
                const checkbox = label.querySelector('input[type="checkbox"]');
                checkbox.checked = !allChecked;
                label.classList.toggle('active', !allChecked);
            });
        }
        document.addEventListener('DOMContentLoaded', () => {
            document.querySelectorAll('.check-badge
    input[type="checkbox"]').forEach(cb => {
                cb.addEventListener('change', function () {
                    this.closest('.check-badge').classList.toggle('active', this.checked);
                });
            });
        });
    </script>
    @endpush
    @endsection
```

resources/views/admin/cars/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        <div class="card mb-4">
            <div class="card-body">
                {{-- Alert Notifikasi Sukses --}}
                @if (session('success'))
```

```
            <div class="alert alert-success alert-dismissible fade show" role="alert">
                {{ session('success') }}
                <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
            </div>
            @endif
            {{-- Tombol Create mobil baru --}}
            <div class="d-flex justify-content-between mb-3">
                <a href="{{ route('admin.cars.create') }}" class="btn btn-
primary">Create</a>
            </div>
            {{-- Form Filter pencarian berdasarkan Brand, Sale Type, dan Status --}}
            <form method="GET" action="{{ route('admin.cars.index') }}" class="row g-3
mb-4">
                <div class="col-md-4">
                    <label for="brand" class="form-label">Brand</label>
                    <select class="form-select" name="brand" id="brand">
                        <option value="">-- All Brands --</option>
                        @foreach ($brands as $brand)
                        <option value="{{ $brand }}" {{ request('brand') == $brand ? 'selected'
: '' }}>
                            {{ $brand }}
                        </option>
                        @endforeach
                    </select>
                </div>
                <div class="col-md-4">
                    <label for="sale_type" class="form-label">Sale Type</label>
                    <select class="form-select" name="sale_type" id="sale_type">
                        <option value="">-- All Sale Types --</option>
                        @foreach ($sale_types as $key => $label)
                        <option value="{{ $key }}" {{ request('sale_type') == $key ? 'selected' :
'' }}>
                            {{ $label }}
                        </option>
                        @endforeach
                    </select>
                </div>
                <div class="col-md-4">
                    <label for="status" class="form-label">Status</label>
                    <select class="form-select" name="status" id="status">
                        <option value="">-- All Statuses --</option>
                        @foreach ($statuses as $key => $label)
                        <option value="{{ $key }}" {{ request('status') == $key ? 'selected' : ''
}}>
                            {{ $label }}
                        </option>
                        @endforeach
                    </select>
                </div>
```

```
                {{-- Tombol Submit Filter dan Reset --}}
                <div class="col-12">
                    <a href="{{ route('admin.cars.index') }}" class="btn btn-
secondary">Reset</a>
                    <button type="submit" class="btn btn-success">Filter</button>
                </div>
            </form>
            {{-- Tabel mobil dengan fitur DataTables --}}
            <div class="table-responsive">
                <table id="cars-table" class="table table-bordered">
                    <thead>
                        <tr>
                            <th>ID</th>
                            <th>Brand</th>
                            <th>Model</th>
                            <th>Price</th>
                            <th>Sale Type</th>
                            <th>Status</th>
                            <th>Action</th>
                        </tr>
                    </thead>
                    <tbody>
                        @forelse ($cars as $car)
                        <tr>
                            <td>{{ $car->id }}</td>
                            <td>{{ $car->brand }}</td>
                            <td>{{ $car->model }}</td>
                            <td>{{ number_format($car->price, 0, ',', '.') }}</td>
                            <td>{{ ucfirst($car->sale_type) }}</td>
                            <td>
                                <!-- Label status saat ini -->
                                @php
                                $status = $car->status;
                                $badgeClass = match($status) {
                                'available' => 'bg-success',
                                'pending_check' => 'bg-warning text-dark',
                                'sold' => 'bg-danger',
                                'under_review' => 'bg-info text-dark',
                                default => 'bg-secondary',
                                };
                                @endphp
                                <span class="badge {{ $badgeClass }} status-label status-label-{{
$car->id }}"
                                    data-id="{{ $car->id }}"
                                    data-status="{{ $status }}">
                                    {{ ucfirst($status) }}
                                </span>
                                {{-- Jika status adalah 'pending_check', tampilkan tombol update
status --}}
                                <!-- Tombol untuk membuka modal update status -->
```

```
                        <button type="button" class="btn btn-sm btn-outline-primary mt-1
btn-edit-status"
                            data-id="{{ $car->id }}" {{-- ID mobil --}} data-status="{{
$car->status }}"
                            {{-- Status saat ini --}} data-bs-toggle="modal" data-bs-
target="#statusModal">
                            Update Status
                        </button>
                    </td>
                    <td>
                        {{-- Tombol untuk membuka modal detail mobil --}}
                        <button type="button" class="btn btn-sm btn-info btn-show-detail"
                            data-id="{{ $car->id }}" data-bs-toggle="modal" data-bs-
target="#carModal">
                            Show
                        </button>
                        {{-- Tombol edit --}}
                        <a href="{{ route('admin.cars.edit', $car->id) }}"
                            class="btn btn-sm btn-warning">Edit</a>
                        {{-- Form untuk delete dengan konfirmasi --}}
                        <form action="{{ route('admin.cars.destroy', $car->id) }}"
method="POST"
                            style="display:inline-block;">
                            @csrf
                            @method('DELETE')
                            <button type="submit" class="btn btn-sm btn-danger"
                                onclick="return confirm('Are you sure?')">Delete</button>
                        </form>
                    </td>
                </tr>
                @empty
                <tr>
                    <td colspan="7" class="text-center">No Cars Found.</td>
                </tr>
                @endforelse
            </tbody>
        </table>
    </div>
    <!-- Modal kosong untuk show, akan diisi konten AJAX -->
    <div class="modal fade" id="carModal" tabindex="-1" aria-
labelledby="carModalLabel" aria-hidden="true">
        <div class="modal-dialog modal-lg">
            <div class="modal-content" id="carModalContent">
                <!-- Konten modal akan di-load via AJAX -->
                <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                    </div>
                </div>
            </div>
```

```
                </div>
            </div>
            <!-- Modal update status (kosong, isi via AJAX) -->
            <div class="modal fade" id="statusModal" tabindex="-1" aria-
labelledby="statusModalLabel"
                aria-hidden="true">
                <div class="modal-dialog">
                    <div class="modal-content" id="statusModalContent">
                        <div class="modal-body text-center">
                            <div class="spinner-border text-primary" role="status">
                                <span class="visually-hidden">Loading...</span>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
        {{-- Pagination --}}
        <div class="card-footer clearfix">
            <div class="pagination pagination-sm m-0 float-end">
                {{ $cars->appends(request()->query())->links('pagination::bootstrap-5') }}
            </div>
        </div>
    </div>
  </div>
</div>
@endsection
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
  $(document).ready(function () {
    // ================================
    // 1. Show detail mobil ke dalam modal
    // ================================
    $('.btn-show-detail').on('click', function () {
        let carId = $(this).data('id');
        let modalContent = $('#carModalContent');
        modalContent.html(`
      <div class="modal-body text-center">
        <div class="spinner-border text-primary" role="status">
          <span class="visually-hidden">Loading...</span>
        </div>
      </div>
    `);
        $.ajax({
            url: `/admin/cars/${carId}`, // Show route: menampilkan detail mobil
            method: 'GET',
            success: function (response) {
                modalContent.html(response); // Load ke dalam modal
            },
```

```
        error: function () {
            modalContent.html(
                '<div class="modal-body text-danger">Failed to load data.</div>'
                );
        }
    });
});
// ================================
// 2. Load form update status via AJAX modal
// ================================
$(document).on('click', '.btn-edit-status', function () {
    const carId = $(this).data('id');
    const modalContent = $('#statusModalContent');
    modalContent.html(`
  <div class="modal-body text-center">
    <div class="spinner-border text-primary" role="status">
      <span class="visually-hidden">Loading...</span>
    </div>
  </div>
`);
    $.ajax({
        url: `/admin/cars/${carId}/edit-status`, // Route untuk ambil form partial
        method: 'GET',
        success: function (html) {
            modalContent.html(html);
        },
        error: function () {
            modalContent.html(
                '<div class="modal-body text-danger">Gagal memuat form status.</div>'
                );
        }
    });
});
// ================================
// 3. Submit update status via AJAX
// ================================
$(document).on('submit', '#form-update-status', function (e) {
    e.preventDefault();
    const carId = $('#modal-car-id').val(); // dari input hidden
    const newStatus = $('#new-status').val();
    $.ajax({
        url: `/admin/cars/${carId}/update-status`, // Route untuk update status
        method: 'POST',
        data: {
            _token: '{{ csrf_token() }}',
            status: newStatus
        },
        success: function (res) {
            $('#statusModal').modal('hide');
            // Update tampilan badge status di table
```

```
                const badge = $(`.status-label-${carId}`);
                badge.text(newStatus.charAt(0).toUpperCase() + newStatus.slice(1));
                // Bersihkan class lama dan tambahkan class baru
                badge.removeClass('bg-success bg-warning text-dark bg-danger bg-info text-
dark bg-secondary');
                badge.addClass(getBadgeClass(newStatus));
                // alert('Status updated successfully!');
            },
            error: function (xhr) {
                console.error('Update failed:', xhr.status, xhr.responseText);
                alert('Failed to update status');
            }
        });
        function getBadgeClass(status) {
            switch (status) {
                case 'available': return 'bg-success';
                case 'pending_check': return 'bg-warning text-dark';
                case 'sold': return 'bg-danger';
                case 'under_review': return 'bg-info text-dark';
                default: return 'bg-secondary';
            }
        }
    });
  });
</script>
@endpush
```

resources/views/admin/dashboard/index.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
  <!-- FullCalendar CSS -->
  <link href="https://cdn.jsdelivr.net/npm/fullcalendar@6.1.9/index.global.min.css"
rel="stylesheet">
@endpush
<!--begin::App Content-->
<div class="app-content">
  <!--begin::Container-->
  <div class="container-fluid">
    <!--begin::Row-->
    <div class="row">
      <!--begin::Col-->
      <div class="col-lg-4 col-6">
        <!--begin::Small Box Widget 1-->
        <div class="small-box text-bg-success">
          <div class="inner">
            <h3>{{$downPayment}}</h3>
            <p>Down Payment Status Confirmed</p>
          </div>
```

```
                <svg class="small-box-icon" fill="currentColor"
xmlns="http://www.w3.org/2000/svg"
                    viewBox="0 0 576 512">
                    <!--!Font Awesome Free 6.7.2 by @fontawesome -
https://fontawesome.com License - https://fontawesome.com/license/free Copyright 2025
Fonticons, Inc.-->
                    <path
                        d="M64 64C28.7 64 0 92.7 0 128L0 384c0 35.3 28.7 64 64 64l448
0c35.3 0 64-28.7 64-64l0-256c0-35.3-28.7-64-64-64L64 64zm64 320l-64 0 0-64c35.3 0
64 28.7 64 64zM64 192l0-64 64 0c0 35.3-28.7 64-64 64zM448 384c0-35.3 28.7-64 64-
64l0 64-64 0zm64-192c-35.3 0-64-28.7-64-64l64 0 0 64zM288 160a96 96 0 1 1 0 192 96
96 0 1 1 0-192z" />
                </svg>
                <a href="{{ route('admin.downPayment.index', ['status' =>
'payment:confirmed']) }}"
                    class="small-box-footer link-light link-underline-opacity-0 link-underline-
opacity-50-hover">
                    More info <i class="bi bi-link-45deg"></i>
                </a>
            </div>
            <!--end::Small Box Widget 1-->
        </div>
        <!--end::Col-->
        <div class="col-lg-4 col-6">
            <!--begin::Small Box Widget 2-->
            <div class="small-box text-bg-warning">
                <div class="inner">
                    <h3>{{$offer}}</h3>
                    <p>Permintaan Penjualan Status Pending</p>
                </div>
                <svg class="small-box-icon" fill="currentColor"
xmlns="http://www.w3.org/2000/svg"
                    viewBox="0 0 448 512">
                    <!--!Font Awesome Free 6.7.2 by @fontawesome -
https://fontawesome.com License - https://fontawesome.com/license/free Copyright 2025
Fonticons, Inc.-->
                    <path
                        d="M224 0c-17.7 0-32 14.3-32 32l0 19.2C119 66 64 130.6 64 208l0
18.8c0 47-17.3 92.4-48.5 127.6l-7.4 8.3c-8.4 9.4-10.4 22.9-5.3 34.4S19.4 416 32 416l384
0c12.6 0 24-7.4 29.2-18.9s3.1-25-5.3-34.4l-7.4-8.3C401.3 319.2 384 273.9 384 226.8l0-
18.8c0-77.4-55-142-128-156.8L256 32c0-17.7-14.3-32-32-32zm45.3 493.3c12-12 18.7-
28.3 18.7-45.3l-64 0-64 0c0 17 6.7 33.3 18.7 45.3s28.3 18.7 45.3 18.7s33.3-6.7 45.3-
18.7z" />
                </svg>
                <a href="{{ route('admin.offer.index', ['status' => 'pending']) }}"
                    class="small-box-footer link-light link-underline-opacity-0 link-underline-
opacity-50-hover">
                    More info <i class="bi bi-link-45deg"></i>
                </a>
            </div>
```

```
            <!--end::Small Box Widget 2-->
        </div>
        <!--end::Col-->
        <div class="col-lg-4 col-6">
            <!--begin::Small Box Widget 3-->
            <div class="small-box text-bg-danger">
                <div class="inner">
                    <h3>{{$car}}</h3>
                    <p>Mobil Terjual</p>
                </div>
                <svg class="small-box-icon" fill="currentColor"
xmlns="http://www.w3.org/2000/svg"
                    viewBox="0 0 640 512">
                    <!--!Font Awesome Free 6.7.2 by @fontawesome -
https://fontawesome.com License - https://fontawesome.com/license/free Copyright 2025
Fonticons, Inc.-->
                    <path
                        d="M171.3 96L224 96l0 96-112.7 0 30.4-75.9C146.5 104 158.2 96
171.3 96zM272 192l0-96 81.2 0c9.7 0 18.9 4.4 25 12l67.2 84L272 192zm256.2 1L428.2
68c-18.2-22.8-45.8-36-75-36L171.3 32c-39.3 0-74.6 23.9-89.1 60.3L40.6 196.4C16.8
205.8 0 228.9 0 256L0 368c0 17.7 14.3 32 32 32l33.3 0c7.6 45.4 47.1 80 94.7 80s87.1-
34.6 94.7-80l130.7 0c7.6 45.4 47.1 80 94.7 80s87.1-34.6 94.7-80l33.3 0c17.7 0 32-14.3
32-32l0-48c0-65.2-48.8-119-111.8-127zM434.7 368a48 48 0 1 1 90.5 32 48 48 0 1 1 -
90.5-32zM160 336a48 48 0 1 1 0 96 48 48 0 1 1 0-96z" />
                    </svg>
                <a href="{{ route('admin.cars.index', ['status' => 'sold'] ) }}"
                    class="small-box-footer link-dark link-underline-opacity-0 link-underline-
opacity-50-hover">
                    More info <i class="bi bi-link-45deg"></i>
                </a>
            </div>
            <!--end::Small Box Widget 3-->
        </div>
        <!--end::Col-->
    </div>
    <!--end::Row-->
    <!--begin::Row-->
    <div class="row">
        <!-- Start col -->
        <div class="col-lg-6 connectedSortable">
            <div class="card mb-4">
                <div class="card-header">
                    <h3 class="card-title">Total Transaksi</h3>
                </div>
                <div class="card-body">
                    <div id="transaction-count-chart"></div>
                </div>
            </div>
        </div>
        <div class="col-lg-6 connectedSortable">
```

```
            <div class="card mb-4">
                <div class="card-header">
                    <h3 class="card-title">Jumlah Transaksi</h3>
                </div>
                <div class="card-body">
                    <div id="sales-amount-chart"></div>
                </div>
            </div>
        </div>
        <!-- End col -->
        <!-- Start col -->
        <div class="col-lg-12">
            <div class="card mb-4">
            <div class="card-header">
                <h3 class="card-title">Jadwal Appointment</h3>
            </div>
            <div class="card-body">
                <div id="calendar"></div>
            </div>
        </div>
        {{-- End Col --}}
    </div>
    <!-- /.row (main row) -->
  </div>
  <!-- Modal kosong, akan diisi konten AJAX -->
    <div class="modal fade" id="downPaymentModal" tabindex="-1" aria-
labelledby="downPaymentModalLabel"
        aria-hidden="true">
        <div class="modal-dialog modal-lg">
            <div class="modal-content" id="downPaymentModalContent">
                <!-- Konten modal akan di-load via AJAX -->
                <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                    </div>
                </div>
            </div>
        </div>
    </div>
  <!--end::Container-->
</div>
<!--end::App Content-->
@push('scripts')
{{-- Jquery --}}
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<!-- apexcharts -->
<script src="https://cdn.jsdelivr.net/npm/apexcharts@3.37.1/dist/apexcharts.min.js"
  integrity="sha256-+vh8GkaU7C9/wbSLIcwq82tQ2wTf44aOHA8HlBMwRI8="
crossorigin="anonymous"></script>
<!-- FullCalendar JS -->
```

```
<script
src="https://cdn.jsdelivr.net/npm/fullcalendar@6.1.9/index.global.min.js"></script>
<!-- ChartJS -->
<script>
   // NOTICE!! DO NOT USE ANY OF THIS JAVASCRIPT
   // IT'S ALL JUST JUNK FOR DEMO
   // ++++++++++++++++++++++++++++++++++++++++++++
   const months = @json($months);
   const salesCountData = @json($salesCountSeries);
   const salesAmountData = @json($salesAmountSeries);
   const offerCountData = @json($offerCountSeries);
   const offerAmountData = @json($offerAmountSeries);
   function formatRupiah(value) {
      return new Intl.NumberFormat('id-ID', { style: 'currency', currency: 'IDR'
}).format(value);
   }
   const transactionCountOptions = {
      series: [
         { name: 'Catatan Transaksi Pembelian', data: salesCountData },
         { name: 'Catatan Transaksi Penjualan', data: offerCountData },
      ],
      chart: { type: 'area', height: 300 },
      xaxis: { type: 'datetime', categories: months },
      tooltip: {
         // y: {
         //    formatter: formatRupiah
         // },
         x: {
            format: 'MMMM yyyy'
         }
      }
   };
   new ApexCharts(document.querySelector('#transaction-count-chart'),
transactionCountOptions).render();
    const salesAmountOptions = {
      series: [
         { name: 'Catatan Transaksi Pembelian', data: salesAmountData },
         { name: 'Catatan Transaksi Penjualan', data: offerAmountData },
      ],
      chart: { type: 'area', height: 300 },
      xaxis: { type: 'datetime', categories: months },
      tooltip: {
         y: {
            formatter: formatRupiah
         },
         x: {
            format: 'MMMM yyyy'
         }
      }
   };
```

```
    new ApexCharts(document.querySelector('#sales-amount-chart'),
salesAmountOptions).render();
</script>
<script>
    function showDownPaymentDetail(id) {
        let modal = new
bootstrap.Modal(document.getElementById('downPaymentModal'));
        let modalContent = $('#downPaymentModalContent');
        modalContent.html(`
            <div class="modal-body text-center">
                <div class="spinner-border text-primary" role="status">
                    <span class="visually-hidden">Loading...</span>
                </div>
            </div>
        `);
        modal.show();
        $.ajax({
            url: `/admin/downPayment/${id}`,
            method: 'GET',
            success: function (response) {
                modalContent.html(response);
            },
            error: function () {
                modalContent.html(
                    '<div class="modal-body text-danger">Failed to load data.</div>'
                );
            }
        });
    }
    $(document).ready(function () {
        // Tombol di tabel
        $('.btn-show-detail').on('click', function () {
            let downPaymentId = $(this).data('id');
            showDownPaymentDetail(downPaymentId);
        });
    });

    document.addEventListener('DOMContentLoaded', function() {
        const calendarEl = document.getElementById('calendar');
        const calendar = new FullCalendar.Calendar(calendarEl, {
            initialView: 'dayGridMonth',
            headerToolbar: {
                left: 'prev,next today',
                center: 'title',
                right: 'dayGridMonth,timeGridWeek,listWeek'
            },
            events: @json($appointments),
            eventClick: function(info) {
                // Panggil modal detail sesuai sumber
                const eventId = info.event.id;
```

```
            const source = info.event.extendedProps.source;
            if (source === 'downpayment') {
                showDownPaymentDetail(eventId.replace('dp-', ''));
            } else if (source === 'offer') {
                window.open(info.event.url, '_blank');
                info.jsEvent.preventDefault();
            }
            info.jsEvent.preventDefault();
        },
        locale: 'id',
        nowIndicator: true,
        height: 650
    });
    calendar.render();
});
</script>
@endpush
@endsection
```

resources/views/admin/downPayment/partials/modal_content.blade.php

```
<div class="modal-header">
    <h5 class="modal-title" id="carModalLabel">Down Payment DP #{{ $downPayment-
>id }}</h5>
    <button type="button" class="btn-close" data-bs-dismiss="modal" aria-
label="Close"></button>
</div>
<div class="modal-body">
    <table class="table table-bordered text-capitalize">
        <tr>
            <th>ID</th>
            <td>{{ $downPayment->id }}</td>
        </tr>
        <tr>
            <th>Created At</th>
            <td>{{ \Carbon\Carbon::parse($downPayment->created_at)->format('d M Y -
H:i') }}</td>
        </tr>
        <tr>
            <th>Nama User</th>
            <td>{{ $downPayment->user->name }}</td>
        </tr>
        <tr>
            <th>No. HP User</th>
            <td>{{ $downPayment->user->phone }}</td>
        </tr>
        <tr>
            <th>Mobil</th>
            <td>#{{ $downPayment->car_id }} {{ $downPayment->car->brand }} {{
$downPayment->car->model }}</td>
        </tr>
```

```
        <tr>
            <th>Appointment Date</th>
            @if ($downPayment->payment_status === 'confirmed' )
                <td><span class=" bg-info">{{ \Carbon\Carbon::parse($downPayment-
>appointment_date)->format('d M Y - H:i') }}</span></td>

            @else
                <td>{{ \Carbon\Carbon::parse($downPayment->appointment_date)->format('d
M Y - H:i') }}</td>

            @endif
        </tr>
        <tr>
            <th>Lokasi Showroom</th>
            <td>Lokasi: Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU TAPOS,
DEPOK<br>
                Kontak Showroom: 081220745317</td>
        </tr>
        <tr>
            <th colspan="2" class="text-muted">Informasi Pembayaran</th>
        </tr>
        <tr>
            <th>Order ID</th>
            <td>{{ $downPayment->order_id ?? '-' }}</td>
        </tr>
        <tr>
            <th>Status Pembayaran</th>
            <td>
                @switch($downPayment->payment_status)
                @case('pending')
                <span class="badge bg-warning text-dark">Pending</span>
                @break
                @case('confirmed')
                    @if(optional($downPayment->refund)->refund_status === 'refund')
                        <span class="badge bg-success">Confirmed</span> | <span class="badge
bg-secondary">Refund</span>
                    @else
                        <span class="badge bg-success">Confirmed</span>
                    @endif
                    @break
                @case('cancelled')
                <span class="badge bg-danger">Cancelled</span>
                @break
                @case('expired')
                <span class="badge bg-danger">Expired</span>
                @break
                @default
                <span class="badge bg-secondary">{{ ucfirst($downPayment-
>payment_status) }}</span>
                @endswitch
```

```
            </td>
        </tr>
        <tr>
            <th>Metode Pembayaran</th>
            <td>{{ $downPayment->payment_method ?? 'Belum Dibayar'}}</td>
        </tr>
        <tr>
            <th>Amount</th>
            <td>Rp {{ number_format($downPayment->amount, 0, ',', '.') }}</td>
        </tr>
        <tr>
            <th>Tanggal Pembayaran</th>
            <td>
                @if ($downPayment->payment_date)
                {{ \Carbon\Carbon::parse($downPayment->payment_date)->format('d M Y -
H:i') }}
                @else
                -
                @endif
            </td>
        </tr>
        @if ($downPayment->refund_id !== null && $downPayment->payment_status ===
'confirmed')
        <tr>
            <th colspan="2" class="text-muted">Informasi Refund</th>
        </tr>
        <tr>
            <th>No Rekening Refund</th>
            <td>{{ $downPayment->refund ? $downPayment->refund->no_rekening_refund :
'-' }}</td>
        </tr>
        <tr>
            <th>Bukti Pembayaran Refund</th>
            <td>
                @if ($downPayment->refund && $downPayment->refund-
>refund_payment_proof)
                <img src="{{ asset('storage/refund/' . $downPayment->refund-
>refund_payment_proof) }}"
                    alt="Bukti Pembayaran Refund" class="img-thumbnail" style="max-height:
200px; object-fit: contain;">
                @else
                -
                @endif
            </td>
        </tr>
        <tr>
            <th>Status Refund</th>
            <td>
                @if ($downPayment->refund)
                @switch($downPayment->refund->refund_status)
```

```
            @case('refund')
            <span class="badge bg-danger">Refund</span>
            @break
            @case('no_refund')
            <span class="badge bg-danger">No Refund</span>
            @break
            @default
            <span class="badge bg-secondary">{{ ucfirst($downPayment->refund->status)
}}</span>
            @endswitch
            @else
            -
            @endif
          </td>
        </tr>
        @endif
    </table>
</div>
<div class="modal-footer">
    <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Tutup</button>
</div>
```

resources/views/admin/downPayment/editRefund.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Edit Refund #{{$downPayment->id}} </div>
            </div>
            <!--end::Header-->
            <!--begin::Error Alert-->
            @if ($errors->any())
            <div class="alert alert-danger">
                <ul class="mb-0">
                    @foreach ($errors->all() as $error)
                    <li>{{ $error }}</li>
                    @endforeach
                </ul>
            </div>
            @endif
            <!--end::Error Alert-->
            <!--begin::Form-->
            <form action="{{ route('admin.downPayment.updateRefund', $downPayment-
>id) }}" method="POST"
                enctype="multipart/form-data">
```

```
            @csrf
            {{-- @method('PUT') --}}
            <div class="card-body">
                <div class="mb-3">
                    <label for="no_rekening_refund" class="form-label">No Rekening
User</label>
                    <textarea class="form-control @error('no_rekening_refund') is-invalid
@enderror"
                        id="no_rekening_refund" name="no_rekening_refund"
                        required>{{ old('no_rekening_refund', $downPayment->refund ?
$downPayment->refund->no_rekening_refund : '') }}
                    </textarea>
                    @error('no_rekening_refund')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="refund_payment_proof" class="form-label">Bukti
Pembayaran Refund</label>
                    <div class="mb-2">
                    <img id="preview-refund" src="{{ $downPayment->refund &&
$downPayment->refund->refund_payment_proof
                            ? asset('storage/refund/' . $downPayment->refund-
>refund_payment_proof)
                            : asset('image/NoImage.png') }}"
                        alt="Preview Bukti Refund" style="max-height: 200px; object-fit:
contain;"
                        class="img-thumbnail">
                    </div>
                    <input type="file" class="form-control @error('refund_payment_proof')
is-invalid @enderror"
                        id="refund_payment_proof" name="refund_payment_proof"
                        onchange="previewImage(event)" >
                    <small class="text-muted">Silakan upload ulang bukti refund jika form
gagal disubmit.</small>
                    @error('refund_payment_proof')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
            </div>
            <div class="card-footer">
                <button type="submit" class="btn btn-primary">Submit</button>
            </div>
        </form>
        <!--end::Form-->
    </div>
```

```
    </div>
    <!--end::Container-->
</div>
@push('scripts')
<script>
    function previewImage(event) {
        const input = event.target;
        const preview = document.getElementById('preview-refund');
        if (input.files && input.files[0]) {
            const reader = new FileReader();
            reader.onload = function (e) {
                preview.src = e.target.result;
                preview.style.display = 'block';
            }
            reader.readAsDataURL(input.files[0]);
        }
    }
</script>
@endpush
@endsection
```

resources/views/admin/downPayment/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        <div class="card mb-4">
            <div class="card-body">
                {{-- Alert Notifikasi Sukses --}}
                @if (session('success'))
                <div class="alert alert-success alert-dismissible fade show" role="alert">
                    {{ session('success') }}
                    <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
                </div>
                @endif
                <form method="GET" action="{{ route('admin.downPayment.index') }}"
class="row g-3 mb-4">
                    {{-- Filter Tanggal (Created At) --}}
                    <div class="col-md-4">
                        <label for="date_range" class="form-label">Created At</label>
                        <input type="text" name="date_range" id="date_range" class="form-
control" value="{{ request('date_range') }}" placeholder="Pilih rentang tanggal">
                    </div>

                    {{-- Filter Status --}}
                    <div class="col-md-4">
                        <label for="status" class="form-label">Status</label>
                        <select name="status" class="form-select">
                            <option value="">-- Semua Status --</option>
```

```
                    @foreach ($statuses as $key => $label)
                    <option value="{{ 'payment:'.$key }}" {{ request('status') ==
'payment:'.$key ? 'selected' : '' }}>
                        {{ $label }}
                    </option>
                    @endforeach
                  </select>
                </div>
                {{-- Filter Nama Mobil --}}
                <div class="col-md-4">
                  <label for="keyword" class="form-label">Nama Mobil</label>
                  <input type="text" class="form-control" name="keyword" id="keyword"
value="{{ request('keyword') }}"
                    placeholder="Masukkan nama mobil...">
                </div>
                {{-- Tombol Filter --}}
                <div class="col-12">
                  <a href="{{ route('admin.downPayment.index') }}" class="btn btn-
secondary">Reset</a>
                  <button type="submit" class="btn btn-success">Filter</button>
                </div>
              </form>
              {{-- Tabel mobil dengan fitur DataTables --}}
              <div class="table-responsive">
                <table id="cars-table" class="table table-bordered">
                  <thead>
                    <tr>
                      <th>ID</th>
                      <th>Created At</th>
                      <th>User</th>
                      <th>Car</th>
                      <th>Amount DP</th>
                      <th>Appointment Date </th>
                      <th>Appointment Time</th>
                      <th>status</th>
                      <th>Action</th>
                    </tr>
                  </thead>
                  <tbody>
                    @forelse ($downPayments as $downPayment)
                    <tr>
                      <td>{{$downPayment->id}}</td>
                      <td>{{ \Carbon\Carbon::parse($downPayment->created_at)-
>translatedFormat('d F Y H:i') }}
                      </td>
                      <td>{{$downPayment->user->name}} </td>
                      <td>{{ $downPayment->car->brand . ' ' . $downPayment->car-
>model }}</td>
                      <td>Rp. {{ number_format($downPayment->amount, 0, ',', '.')
}}</td>
```

```
                    {{-- @php
                        dd($downPayment->appointment_date);
                    @endphp --}}
                    <td>{{ \Carbon\Carbon::parse($downPayment->appointment_date)-
>translatedFormat('d F Y') }}
                    </td>
                    <td>{{ \Carbon\Carbon::parse($downPayment->appointment_date)-
>translatedFormat('H:i') .' WIB' }}
                    </td>
                    <td>
                        @switch($downPayment->payment_status)
                        @case('pending')
                        <span class="badge bg-warning text-dark">Pending</span>
                        @break
                        @case('confirmed')
                            @if(optional($downPayment->refund)->refund_status ===
'refund')
                                <span class="badge bg-success">Confirmed</span> | <span
class="badge bg-secondary">Refund</span>
                            @else
                                <span class="badge bg-success">Confirmed</span>
                            @endif
                        @break
                        @case('cancelled')
                        <span class="badge bg-danger">Cancelled</span>
                        @break
                        @case('expired')
                        <span class="badge bg-danger">Expired</span>
                        @break
                        @default
                        <span class="badge bg-secondary">{{ ucfirst($item-
>payment_status) }}</span>
                        @endswitch
                    </td>
                    <td>
                        {{-- Tombol untuk membuka modal detail mobil --}}
                        <button type="button" class="btn btn-sm btn-info btn-show-detail"
                            data-id="{{ $downPayment->id }}" data-bs-toggle="modal"
                            data-bs-target="#downPaymentModal">
                            Show
                        </button>
                        {{-- Tombol untuk edit down payment --}}
                        @if ( $downPayment->payment_status === 'confirmed' &&
$downPayment->refund_id === null)
                            <a href="{{ route('admin.downPayment.addRefund',
$downPayment->id) }}"
                                class="btn btn-sm btn-secondary">
                                Refund
                            </a>
```

```
                        @elseif ($downPayment->payment_status === 'confirmed' &&
$downPayment->refund_id !== null)
                          <a href="{{ route('admin.downPayment.editRefund',
$downPayment->id) }}"
                             class="btn btn-sm btn-secondary">
                             Edit Refund
                          </a>
                        @endif
                      </td>
                    </tr>
                    @empty
                    <tr>
                      <td colspan="9" class="text-center">No down payments found.</td>
                    </tr>
                    @endforelse
                  </tbody>
                </table>
              </div>
              <!-- Modal kosong, akan diisi konten AJAX -->
              <div class="modal fade" id="downPaymentModal" tabindex="-1" aria-
labelledby="downPaymentModalLabel"
                aria-hidden="true">
                <div class="modal-dialog modal-lg">
                  <div class="modal-content" id="downPaymentModalContent">
                    <!-- Konten modal akan di-load via AJAX -->
                    <div class="modal-body text-center">
                      <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
            {{-- Pagination --}}
            <div class="card-footer clearfix">
              <div class="pagination pagination-sm m-0 float-end">
                {{ $downPayments->appends(request()->query())-
>links('pagination::bootstrap-5') }}
              </div>
            </div>
</div>
</div>
</div>
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/flatpickr/dist/flatpickr.min.css">
<script src="https://cdn.jsdelivr.net/npm/flatpickr"></script>
<script src="https://cdn.jsdelivr.net/npm/flatpickr/dist/plugins/rangePlugin.js"></script>
<script>
```

```
    flatpickr("#date_range", {
        mode: "range",
        dateFormat: "Y-m-d",
        defaultDate: @json(explode(' to ', request('date_range')))
    });
    $(document).ready(function () {
        // ================================
        // 1. Show detail downPayment ke dalam modal
        // ================================
        $('.btn-show-detail').on('click', function () {
            let downPaymentId = $(this).data('id');
            let modalContent = $('#downPaymentModalContent');
            modalContent.html(`
          <div class="modal-body text-center">
            <div class="spinner-border text-primary" role="status">
              <span class="visually-hidden">Loading...</span>
            </div>
          </div>
        `);
            $.ajax({
                url: `/admin/downPayment/${downPaymentId}`, // Show route: menampilkan
detail mobil
                method: 'GET',
                success: function (response) {
                    modalContent.html(response); // Load ke dalam modal
                },
                error: function () {
                    modalContent.html(
                        '<div class="modal-body text-danger">Failed to load data.</div>'
                    );
                }
            });
        });

    });
</script>
@endpush
@endsection
```

resources/views/admin/downPayment/refund.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Refund #{{$downPayment->id}} </div>
            </div>
```

```
        <!--end::Header-->
        <!--begin::Error Alert-->
        @if ($errors->any())
        <div class="alert alert-danger">
            <ul class="mb-0">
                @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
                @endforeach
            </ul>
        </div>
        @endif
        <!--end::Error Alert-->
        <!--begin::Form-->
        <form action="{{ route('admin.downPayment.storeRefund', $downPayment->id)
}}" method="POST"
            enctype="multipart/form-data">
            @csrf
            <div class="card-body">
                <div class="mb-3">
                    <label for="no_rekening_refund" class="form-label">No Rekening
User</label>
                    <textarea class="form-control @error('no_rekening_refund') is-invalid
@enderror"
                        id="no_rekening_refund" name="no_rekening_refund"
                        required>{{ old('no_rekening_refund', $downPayment->refund ?
$downPayment->refund->no_rekening_refund : '') }}
                    </textarea>
                    @error('no_rekening_refund')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                <div class="mb-3">
                    <label for="refund_payment_proof" class="form-label">Bukti
Pembayaran Refund</label>
                    <div class="mb-2">
                    <img id="preview-refund" src="{{ $downPayment->refund &&
$downPayment->refund->refund_payment_proof
                            ? asset('storage/refund/' . $downPayment->refund-
>refund_payment_proof)
                            : asset('image/NoImage.png') }}"
                        alt="Preview Bukti Refund" style="max-height: 200px; object-fit:
contain;"
                        class="img-thumbnail">
                    </div>
                    <input type="file" class="form-control @error('refund_payment_proof')
is-invalid @enderror"
                        id="refund_payment_proof" name="refund_payment_proof"
                        onchange="previewImage(event)" required>
```

```
                @error('refund_payment_proof')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
        </div>
        <div class="card-footer">
            <button type="submit" class="btn btn-primary">Submit</button>
        </div>
    </form>
    <!--end::Form-->
  </div>
</div>
<!--end::Container-->
</div>
@push('scripts')
<script>
  function previewImage(event) {
    const input = event.target;
    const preview = document.getElementById('preview-refund');
    if (input.files && input.files[0]) {
      const reader = new FileReader();
      reader.onload = function (e) {
        preview.src = e.target.result;
        preview.style.display = 'block';
      }
      reader.readAsDataURL(input.files[0]);
    }
  }
</script>
@endpush
@endsection
```

resources/views/admin/offer/partials/update_status_modal.blade.php

```
<div class="modal-header">
  <h5 class="modal-title">Perbarui Status Permintaan Penjualan</h5>
  <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
</div>
<div class="modal-body">
  <form id="form-update-status">
    @csrf
    <input type="hidden" name="offer_id" id="modal-offer-id" value="{{ $offer-
>id }}">
    <div class="mb-3">
      <label for="new-status" class="form-label">Select Status</label>
      <select name="status" id="new-status" class="form-select">
        @foreach(['accepted', 'pending', 'rejected', 'sold'] as $status)
          <option value="{{ $status }}" {{ $offer->status === $status ? 'selected' : ''
}}>
```

```
                {{ ucfirst($status) }}
              </option>
            @endforeach
          </select>
        </div>
      </form>
    </div>
    <div class="modal-footer">
      <button type="button" class="btn btn-secondary" data-bs-
    dismiss="modal">Cancel</button>
      <button type="submit" form="form-update-status" class="btn btn-
    primary">Save</button>
    </div>
```

resources/views/admin/offer/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <div class="container-fluid">
    <div class="card mb-4">
      <div class="card-body">
        {{-- Alert Notifikasi Sukses --}}
        @if (session('success'))
        <div class="alert alert-success alert-dismissible fade show" role="alert">
          {{ session('success') }}
          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>
        @endif
        {{-- Form Filter pencarian berdasarkan Brand, Sale Type, dan Status --}}
        <form method="GET" action="{{ route('admin.offer.index') }}" class="row g-
3 mb-4">
          <div class="col-md-4">
            <label for="date_range" class="form-label">Created At</label>
            <input type="text" name="date_range" id="date_range" class="form-
control" value="{{ request('date_range') }}" placeholder="Pilih rentang tanggal">
          </div>
          <div class="col-md-4">
            <label for="status" class="form-label">Status</label>
            <select class="form-select" name="status" id="status">
              <option value="">-- All Status --</option>
              @foreach ($statuses as $key => $label)
              <option value="{{ $key }}" {{ request('status') == $key ? 'selected' : ''
}}>
                {{ $label }}
              </option>
              @endforeach
            </select>
          </div>
          {{-- Filter Nama Mobil (Brand/Model) --}}
          <div class="col-md-4">
```

```
                <label for="keyword" class="form-label">Car Name</label>
                <input type="text" name="keyword" id="keyword" class="form-control"
                    value="{{ request('keyword') }}" placeholder="Search brand/model...">
            </div>
            {{-- Tombol Submit Filter dan Reset --}}
            <div class="col-12">
                <a href="{{ route('admin.cars.index') }}" class="btn btn-
secondary">Reset</a>
                <button type="submit" class="btn btn-success">Filter</button>
            </div>
        </form>
        {{-- Tabel mobil dengan fitur DataTables --}}
        <div class="table-responsive">
            <table id="cars-table" class="table table-bordered">
                <thead>
                    <tr>
                        <th>ID</th>
                        <th>Create At</th>
                        <th>User</th>
                        <th>Car</th>
                        <th>Offer Price</th>
                        <th>Inspection Date</th>
                        <th>Status</th>
                        <th>Action</th>
                    </tr>
                </thead>
                <tbody>
                    @forelse ($offers as $offer)
                    <tr>
                        <td>{{ $offer->id }}</td>
                        <td>{{ \Carbon\Carbon::parse($offer->created_at)-
>translatedFormat('d F Y H:i') }}</td>
                        <td>{{$offer->user->name}}</td>
                        <td>{{ $offer->brand . ' ' . $offer->model }}</td>
                        <td>{{ 'Rp ' . number_format($offer->offered_price, 0, ',', '.') }}</td>
                        <td>{{ \Carbon\Carbon::parse($offer->inspection_date)-
>translatedFormat('d F Y H:i') }}</td>
                        <td>
                            @php
                                $status = $offer->status;
                                $badgeClass = match($status) {
                                    'pending' => 'bg-warning',
                                    'accepted' => 'bg-success',
                                    'rejected' => 'bg-danger',
                                    default => 'bg-secondary',
                                };
                            @endphp
                            <span class="badge {{ $badgeClass }} status-label-{{ $offer->id
}}">
                                {{ ucfirst($status) }}
```

```
                        </span>
                        <button type="button" class="btn btn-sm btn-outline-primary mt-1
btn-edit-status"
                            data-id="{{ $offer->id }}" data-status="{{ $offer->status }}"
                            data-bs-toggle="modal" data-bs-target="#statusModal">
                            Update Status
                        </button>
                    </td>
                    <td>
                        {{-- Optional action --}}
                        <a href="{{ route('admin.offer.show', $offer->id) }}" class="btn
btn-sm btn-info">
                            Show
                    </td>
                </tr>
                {{-- Jika tidak ada data mobil --}}
                @empty
                <tr>
                    <td colspan="8" class="text-center">No Offers Found.</td>
                </tr>
                @endforelse
            </tbody>
        </table>
    </div>
    {{-- Modal kosong yang akan diisi via AJAX --}}
    <div class="modal fade" id="statusModal" tabindex="-1" aria-
labelledby="statusModalLabel" aria-hidden="true">
        <div class="modal-dialog">
            <div class="modal-content" id="statusModalContent">
                <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                    </div>
                </div>
            </div>
        </div>
    </div>
    {{-- Pagination --}}
    <div class="card-footer clearfix">
        <div class="pagination pagination-sm m-0 float-end">
            {{ $offers->appends(request()->query())->links('pagination::bootstrap-5')
}}
        </div>
    </div>
  </div>
 </div>
</div>
</div>
@endsection
@push('scripts')
```

```
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/flatpickr/dist/flatpickr.min.css">
<script src="https://cdn.jsdelivr.net/npm/flatpickr"></script>
<script src="https://cdn.jsdelivr.net/npm/flatpickr/dist/plugins/rangePlugin.js"></script>
<script>
    flatpickr("#date_range", {
        mode: "range",
        dateFormat: "Y-m-d",
        defaultDate: @json(explode(' to ', request('date_range')))
    });
    $(document).ready(function () {
        // Load form update status ke dalam modal
        $(document).on('click', '.btn-edit-status', function () {
            const offerId = $(this).data('id');
            const modalContent = $('#statusModalContent');
            modalContent.html(`
                <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                    </div>
                </div>
            `);
            $.ajax({
                url: `/admin/offer/${offerId}/edit-status`,
                method: 'GET',
                success: function (html) {
                    modalContent.html(html);
                },
                error: function () {
                    modalContent.html(
                        '<div class="modal-body text-danger">Gagal memuat form status.</div>'
                    );
                }
            });
        });
        // Submit form update status
        $(document).on('submit', '#form-update-status', function (e) {
            e.preventDefault();
            const offerId = $('#modal-offer-id').val();
            const newStatus = $('#new-status').val();
            $.ajax({
                url: `/admin/offer/${offerId}/update-status`,
                method: 'POST',
                data: {
                    _token: '{{ csrf_token() }}',
                    status: newStatus
                },
                success: function (res) {
                    $('#statusModal').modal('hide');
                    const badge = $(`.status-label-${offerId}`);
```

```
                badge.text(newStatus.charAt(0).toUpperCase() + newStatus.slice(1));
                badge.removeClass('bg-warning bg-success bg-danger bg-secondary')
                    .addClass(getBadgeClass(newStatus));
                alert(res.message);
            },
            error: function (xhr) {
                console.error('Update failed:', xhr.status, xhr.responseText);
                alert('Failed to update status');
            }
        });
        function getBadgeClass(status) {
            switch (status) {
                case 'pending': return 'bg-warning';
                case 'accepted': return 'bg-success';
                case 'rejected': return 'bg-danger';
                default: return 'bg-secondary';
            }
        }
    });
});
</script>
@endpush
```

resources/views/admin/offer/show.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        @if (session('success'))
            <div class="alert alert-success alert-dismissible fade show" role="alert">
                {{ session('success') }}
                <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
            </div>
        @endif
        <div class="card mb-4">
            <div class="card-body">
                <h3 class="fw-bold mb-4">Detail Penjualan Mobil {{ $offer->brand  . ' ' .
$offer->model . ' ' . $offer->year ?? '-' }} </h3>
                <table class="table table-bordered">
                    <tr>
                        <th>ID Pembayaran</th>
                        <td>{{ '#'. $offer->id ?? '-'}}</td>
                    </tr>
                    <tr>
                        <th>Status Penjualan</th>
                        <td>
                            @php
                                $status = $offer->status;
                                $badgeClass = match($status) {
                                    'pending' => 'bg-warning',
```

```
                    'accepted' => 'bg-success',
                    'rejected' => 'bg-danger',
                    default => 'bg-secondary',
                  };
                @endphp
                <span class="badge {{ $badgeClass }} status-label-{{ $offer->id }}">
                  {{ ucfirst($status) }}
                </span>
                <button type="button" class="btn btn-sm btn-outline-primary mt-1 btn-
edit-status"
                  data-id="{{ $offer->id }}" data-status="{{ $offer->status }}"
                  data-bs-toggle="modal" data-bs-target="#statusModal">
                  Update Status
                </button>
              </td>
            </tr>
            <tr>
              <th>Nama Pengguna</th>
              <td>{{ $offer->user->name ?? '-' }}</td>
            </tr>
            <tr>
              <th>Email</th>
              <td>{{ $offer->user->email ?? '-' }}</td>
            </tr>
            <tr>
              <th>Nomor Telepon</th>
              <td>{{ $offer->user->phone ?? '-' }}</td>
            </tr>
            <tr>
              <th>Mobil</th>
              <td>{{ $offer->brand  . ' ' . $offer->model . ' ' . $offer->year ?? '-' }}</td>
            </tr>
            <tr>
              <th>Penawaran Harga Mobil</th>
              <td>Rp {{ number_format($offer->offered_price, 0, ',', '.') }}</td>
            </tr>
            <tr>
              <th>Tanggal Janji</th>
              <td>{{ \Carbon\Carbon::parse($offer->inspection_date)->format('d M Y')
}}</td>
            </tr>
            <tr>
              <th>Jam Janji</th>
              <td>{{ \Carbon\Carbon::parse($offer->inspection_date)->format('H:i T')
}}</td>
            </tr>
            <tr>
              <th>Lokasi Inspeksi</th>
              <td>
```

```
                    @if($offer->status === 'accepted' && $offer->location_inspection ===
'Datang Ke Showroom')
                    Datang Ke Showroom<br>
                    Lokasi: Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU
TAPOS, DEPOK<br>
                    Kontak Admin: 081220745317
                    @elseif($offer->location_inspection === 'Rumah')
                    Rumah<br>
                    Lokasi: {{ Auth::user()->address ?? 'Alamat tidak tersedia' }}
                    <br>
                    @else
                    {{ $offer->location_inspection }}
                    @endif
                  </td>
                </tr>
                <tr>
                  <th>Catatan Showroom</th>
                  <td>
                    @if($offer->status === 'accepted' && $offer->note === null)
                    <span class="text-danger">Catatan belum diisi</span>
                    @endif
                    {{ $offer->note  }}
                    <br>
                    <button class="btn btn-sm btn-info" data-bs-toggle="modal" data-bs-
target="#catatanModal">
                      Tambahkan Catatan
                    </button>
                  </td>
                </tr>
              </table>
            </div>
            {{-- Modal kosong yang akan diisi via AJAX --}}
            <div class="modal fade" id="statusModal" tabindex="-1" aria-
labelledby="statusModalLabel" aria-hidden="true">
              <div class="modal-dialog">
                <div class="modal-content" id="statusModalContent">
                  <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                      <span class="visually-hidden">Loading...</span>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
<!-- Modal Lengkapi Alamat -->
<div class="modal fade" id="catatanModal" tabindex="-1" aria-
labelledby="catatanModalLabel" aria-hidden="true">
```

```
  <div class="modal-dialog">
    <form method="POST" action="{{ route('admin.offer.updateNote', $offer->id) }}">
        @csrf
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="catatanModalLabel">Tambahkan Catatan
Showroom</h5>
                <button type="button" class="btn-close" data-bs-dismiss="modal" aria-
label="Tutup"></button>
            </div>
            <div class="modal-body">
                <div class="mb-3">
                    <textarea name="note" class="form-control" rows="3" required>{{
old('address', Auth::user()->address) }}</textarea>
                </div>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Cancel</button>
                <button type="submit" class="btn btn-success">Save</button>
            </div>
        </div>
    </form>
  </div>
</div>
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
    $(document).ready(function () {
        // Load form update status ke dalam modal
        $(document).on('click', '.btn-edit-status', function () {
            const offerId = $(this).data('id');
            const modalContent = $('#statusModalContent');
            modalContent.html(`
                <div class="modal-body text-center">
                    <div class="spinner-border text-primary" role="status">
                        <span class="visually-hidden">Loading...</span>
                    </div>
                </div>
            `);
            $.ajax({
                url: `/admin/offer/${offerId}/edit-status`,
                method: 'GET',
                success: function (html) {
                    modalContent.html(html);
                },
                error: function () {
                    modalContent.html(
                        '<div class="modal-body text-danger">Gagal memuat form status.</div>'
                    );
```

```
            }
        });
    });
    // Submit form update status
    $(document).on('submit', '#form-update-status', function (e) {
        e.preventDefault();
        const offerId = $('#modal-offer-id').val();
        const newStatus = $('#new-status').val();
        $.ajax({
            url: `/admin/offer/${offerId}/update-status`,
            method: 'POST',
            data: {
                _token: '{{ csrf_token() }}',
                status: newStatus
            },
            success: function (res) {
                $('#statusModal').modal('hide');
                const badge = $(`.status-label-${offerId}`);
                badge.text(newStatus.charAt(0).toUpperCase() + newStatus.slice(1));
                badge.removeClass('bg-warning bg-success bg-danger bg-secondary')
                    .addClass(getBadgeClass(newStatus));
                alert(res.message);
            },
            error: function (xhr) {
                console.error('Update failed:', xhr.status, xhr.responseText);
                alert('Failed to update status');
            }
        });
        function getBadgeClass(status) {
            switch (status) {
                case 'pending': return 'bg-warning';
                case 'accepted': return 'bg-success';
                case 'rejected': return 'bg-danger';
                default: return 'bg-secondary';
            }
        }
    });
});
</script>
@endpush
@endsection
```

resources/views/admin/offerRecord/create.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/flatpickr/dist/flatpickr.min.css">
@endpush
<div class="app-content">
```

```
<div class="container-fluid">
    <div class="card card-primary card-outline mb-4">
        <div class="card-header">
            <div class="card-title">Transaction User Offer</div>
        </div>
        @if ($errors->any())
        <div class="alert alert-danger">
            <ul class="mb-0">
                @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
                @endforeach
            </ul>
        </div>
        @endif
        <form action="{{ route('admin.offerRecord.store') }}" method="POST">
            @csrf
            <div class="card-body">
                {{-- Buyer --}}
                <div class="mb-3">
                    <label for="buyer_id" class="form-label">Buyer</label>
                    <select class="form-select @error('buyer_id') is-invalid @enderror"
id="buyer_id"
                        name="buyer_id" required>
                        <option value="">-- Select Buyer --</option>
                        @foreach($users as $user)

                        <option value="{{ $user->id }}"
                            {{ old('user_id', auth()->user()->id) == $user->id ? 'selected' : '' }}>
                            {{ $user->name }}
                        </option>
                        @endforeach
                    </select>
                    @error('buyer_id')
                    <span class="invalid-feedback">{{ $message }}</span>
                    @enderror
                </div>
                {{-- Seller --}}
                <div class="mb-3">
                    <label for="seller_id" class="form-label">Seller</label>
                    <select class="form-select @error('seller_id') is-invalid @enderror"
id="seller_id"
                        name="seller_id" required>
                        <option value="">-- Select Seller --</option>
                        @foreach($users as $user)
                        @if(old('buyer_id') != $user->id) {{-- Exclude user yang sama dengan
seller --}}
                        <option value="{{ $user->id }}" {{ old('seller_id') == $user->id ?
'selected' : '' }}>
                            {{ $user->name }}
                        </option>
```

```
                    @endif
                    @endforeach
                </select>
                @error('seller_id')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
            </div>
            {{-- Car --}}
            <div class="mb-3">
                <label for="offer_id" class="form-label">Offer</label>
                <select class="form-select @error('offer_id') is-invalid @enderror"
id="offer_id" name="offer_id" required>
                    <option value="">-- Select Offer --</option>
                    @foreach($offers as $offer)
                    <option value="{{ $offer->id }}" data-price="{{ $offer->offered_price
}}"
                        {{ old('offer_id') == $offer->id ? 'selected' : '' }}>
                        {{ $offer->brand }} {{ $offer->model }}
                    </option>
                    @endforeach
                </select>
                @error('offer_id')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
            </div>
            {{-- Sale Price --}}
            <div class="mb-3">
                <label for="offered_price" class="form-label fw-medium">Sale
Price</label>
                <input type="text" class="form-control py-2 @error('offered_price') is-
invalid @enderror"
                    id="offered_price_format"
                    value="{{ old('offered_price') ? number_format(old('offered_price'), 0,
',', '.') : '' }}" />
                <input type="hidden" name="sale_price" id="offered_price" value="{{
old('offered_price') }}" />
                @error('offered_price')
                <span class="invalid-feedback" role="alert"><strong>{{ $message
}}</strong></span>
                @enderror
            </div>
            {{-- Sale Date --}}
            <div class="mb-3">
                <label for="sale_date" class="form-label">Sale Date</label>
                <input type="text"
                    class="form-control @error('sale_date') is-invalid @enderror"
                    id="sale_date"
                    name="sale_date"
                     value="{{ old('sale_date', date('Y-m-d\TH:i')) }}"
                    required>
```

```
                @error('sale_date')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
              </div>
            </div>
            <div class="card-footer">
              <button type="submit" class="btn btn-primary">Submit</button>
            </div>
          </form>
        </div>
      </div>
    </div>
@push('scripts')
<script src="https://cdn.jsdelivr.net/npm/flatpickr"></script>
<script src="https://npmcdn.com/flatpickr/dist/l10n/id.js"></script>
<script>
  function formatRupiah(angka) {
    // Hapus karakter selain angka dan titik (untuk desimal)
    let numberString = angka.replace(/[^\d.]/g, "");
    // Pisahkan antara bagian sebelum dan sesudah koma (jika ada)
    let parts = numberString.split(".");
    // Ambil hanya bagian sebelum titik desimal
    let integerPart = parts[0];
    // Format ke rupiah (dengan titik setiap 3 digit)
    return integerPart ? 'Rp ' + integerPart.replace(/\B(?=(\d{3})+(?!\d))/g, ".") : '';
  }
  document.addEventListener('DOMContentLoaded', function () {
    // Filter buyer options based on selected seller
    // Filter buyer options based on selected seller
    const sellerSelect = document.getElementById('seller_id');
    const buyerSelect = document.getElementById('buyer_id');
    function filterSellerOptions() {
      const selectedBuyer = buyerSelect.value;
      for (let option of sellerSelect.options) {
        option.hidden = false;
        if (option.value === selectedBuyer) {
          option.hidden = true;
        }
      }
      // Jika buyer saat ini sama dengan seller, reset buyer
      if (sellerSelect.value === selectedBuyer) {
        sellerSelect.value = "";
      }
    }
    buyerSelect.addEventListener('change', filterSellerOptions);
    // Jalankan saat pertama kali load (jika ada old value)
    filterSellerOptions();
    const offerSelect = document.getElementById('offer_id');
    const priceInput = document.getElementById('offered_price');
    const inputFormatted = document.getElementById('offered_price_format');
```

```
        inputFormatted.addEventListener('input', function () {
            let clean = this.value.replace(/\D/g, "");
            this.value = formatRupiah(clean);
            priceInput.value = clean;
        });
        function updatePrice() {
            const selectedOption = offerSelect.options[offerSelect.selectedIndex];
            const price = selectedOption.getAttribute('data-price');
            if (price) {
                priceInput.value = price;
                inputFormatted.value = formatRupiah(price);
            } else {
                priceInput.value = '';
                inputFormatted.value = '';
            }
        }
        offerSelect.addEventListener('change', updatePrice);
        updatePrice(); // Jalankan jika ada old value
    });
    flatpickr("#sale_date", {
        enableTime: true,
        dateFormat: "Y-m-d H:i",
        // minDate: "{{ date('Y-m-d') }}",
        locale: "id"
    });
</script>
@endpush
@endsection
```

resources/views/admin/offerRecord/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        <div class="card mb-4">
            <div class="card-body">
                {{-- Alert Notifikasi Sukses --}}
                @if (session('success'))
                <div class="alert alert-success alert-dismissible fade show" role="alert">
                    {{ session('success') }}
                    <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
                </div>
                @endif
                {{-- Tombol Create mobil baru --}}
                <div class="d-flex justify-content-between mb-3">
                    <a href="{{ route('admin.offerRecord.create') }}" class="btn btn-
primary">Create</a>
                </div>
```

```
{{-- Tabel mobil dengan fitur DataTables --}}
<div class="table-responsive">
  <table id="cars-table" class="table table-bordered">
    <thead>
      <tr>
        <th>ID</th>
        <th>Create At</th>
        <th>Car</th>
        <th>Saler</th>
        <th>Buyer</th>
        <th>Sale Price</th>
        <th>Sale Date</th>
        <th>Action</th>
      </tr>
    </thead>
    <tbody>
      @forelse ($offerRecords as $offerRecord)
      <tr>
        <td>{{ $offerRecord->id }}</td>
        <td>{{ \Carbon\Carbon::parse($offerRecord->created_at)-
>translatedFormat('d F Y H:i') }}</td>
        <td><a href="{{route('admin.offer.show',$offerRecord->offer-
>id)}}" >{{ $offerRecord->offer->brand . ' ' . $offerRecord->offer->model . ' ' .
$offerRecord->offer->year }}</a></td>
        <td>{{ $offerRecord->salerOfferRecord->name }}</td>
        <td>{{ $offerRecord->buyerOfferRecord->name }}</td>
        <td>Rp. {{ number_format($offerRecord->sale_price, 0, ',', '.')
}}</td>
        <td>{{ \Carbon\Carbon::parse($offerRecord->sale_date)-
>translatedFormat('d F Y') }}</td>
        <td>
          <form action="{{ route('admin.offerRecord.destroy', $offerRecord)
}}" method="POST" class="d-inline">
            @csrf
            @method('DELETE')
            <button type="submit" class="btn btn-danger btn-sm"
onclick="return confirm('Are you sure you want to delete this record?')">Delete</button>
          </form>
        </td>
      </tr>
      @empty
        <tr>
          <td colspan="8" class="text-center">No offer records found.</td>
        </tr>
      @endforelse
    </tbody>
  </table>
</div>

{{-- Pagination --}}
```

```
            <div class="card-footer clearfix">
                <div class="pagination pagination-sm m-0 float-end">
                    {{ $offerRecords->appends(request()->query())-
>links('pagination::bootstrap-5') }}
                </div>
            </div>
        </div>
    </div>
  </div>
</div>
@endsection
```

resources/views/admin/salesRecord/create.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <!--begin::Container-->
  <div class="container-fluid">
    <div class="card card-primary card-outline mb-4">
      <!--begin::Header-->
      <div class="card-header">
        <div class="card-title">Transaction Sales Record Car</div>
      </div>
      <!--end::Header-->
      <!--begin::Error Alert-->
      @if ($errors->any())
      <div class="alert alert-danger">
        <ul class="mb-0">
          @foreach ($errors->all() as $error)
          <li>{{ $error }}</li>
          @endforeach
        </ul>
      </div>
      @endif
      <!--end::Error Alert-->
      <!--begin::Form-->
      <form action="{{ route('admin.salesRecord.store') }}" method="POST">
        @csrf
        <div class="card-body">
          {{-- Seller --}}
          <div class="mb-3">
            <label for="seller_id" class="form-label">Seller</label>
            <select class="form-select @error('seller_id') is-invalid @enderror"
id="seller_id"
              name="seller_id" required>
              <option value="">-- Select Seller --</option>
              @foreach($users as $user)
              <option value="{{ $user->id }}"
                {{ old('user_id', auth()->user()->id) == $user->id ? 'selected' : '' }}>
                {{ $user->name }}
```

```
                    </option>
                    @endforeach
                </select>
                @error('seller_id')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
            </div>
            {{-- Buyer --}}
            {{-- Buyer --}}
            <div class="mb-3">
                <label for="buyer_id" class="form-label">Buyer</label>
                <select class="form-select @error('buyer_id') is-invalid @enderror"
id="buyer_id"
                    name="buyer_id" required>
                    <option value="">-- Select Buyer --</option>
                    @foreach($users as $user)
                    @if(old('seller_id') != $user->id) {{-- Exclude user yang sama dengan
seller --}}
                    <option value="{{ $user->id }}" {{ old('buyer_id') == $user->id ?
'selected' : '' }}>
                        {{ $user->name }}
                    </option>
                    @endif
                    @endforeach
                </select>
                @error('buyer_id')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
            </div>
            {{-- Car --}}
            <div class="mb-3">
                <label for="car_id" class="form-label">Car</label>
                <select class="form-select @error('car_id') is-invalid @enderror"
id="car_id" name="car_id"
                    required>
                    <option value="">-- Select Car --</option>
                    @foreach($cars as $car)
                    <option value="{{ $car->id }}" data-price="{{ $car->price }}"
                        {{ old('car_id') == $car->id ? 'selected' : '' }}>
                        {{ $car->brand }} {{ $car->model }}
                    </option>
                    @endforeach
                </select>
            </div>

            <div class="mb-3">
                <label for="sale_price" class="form-label fw-medium">Sale Price</label>
                <input type="text" class="form-control py-2 @error('sale_price') is-invalid
@enderror"
                    id="sale_price_format"
```

```
                    value="{{ old('sale_price') ? number_format(old('sale_price'), 0, ',', '.') :
" }}"
                    required />
                <input type="hidden" name="sale_price" id="sale_price" value="{{
old('sale_price') }}" />
                @error('price')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            {{-- Sale Date --}}
            <div class="mb-3">
                <label for="sale_date" class="form-label">Sale Date & Time</label>
                <input type="datetime-local"
                    class="form-control @error('sale_date') is-invalid @enderror"
                    id="sale_date"
                    name="sale_date"
                    value="{{ old('sale_date', \Carbon\Carbon::now()->format('Y-m-
d\TH:i')) }}"
                    min="{{ \Carbon\Carbon::now()->format('Y-m-d\TH:i') }}"
                    required />
                @error('sale_date')
                <span class="invalid-feedback">{{ $message }}</span>
                @enderror
            </div>
        </div>
    </div>
    <div class="card-footer">
        <button type="submit" class="btn btn-primary">Submit</button>
    </div>
    </form>

    <!--end::Form-->
  </div>
</div>
<!--end::Container-->
</div>
@push('scripts')
<script>
  // Format input offered_price to Rupiah
  function formatRupiah(angka) {
    let numberString = angka.replace(/\D/g, "");
    let formatted = numberString.replace(/\B(?=(\d{3})+(?!\d))/g, ".");
    return numberString ? 'Rp ' + formatted : '';
  }
  document.addEventListener('DOMContentLoaded', function () {
    // Filter buyer options based on selected seller
    const sellerSelect = document.getElementById('seller_id');
    const buyerSelect = document.getElementById('buyer_id');
```

```
        function filterBuyerOptions() {
            const selectedSeller = sellerSelect.value;
            for (let option of buyerSelect.options) {
                option.hidden = false;
                if (option.value === selectedSeller) {
                    option.hidden = true;
                }
            }
            // Jika buyer saat ini sama dengan seller, reset buyer
            if (buyerSelect.value === selectedSeller) {
                buyerSelect.value = "";
            }
        }
        sellerSelect.addEventListener('change', filterBuyerOptions);
        // Jalankan saat pertama kali load (jika ada old value)
        filterBuyerOptions();
        // Update sale_price based on selected car
        const carSelect = document.getElementById('car_id');
        const priceInput = document.getElementById('sale_price');
        const inputFormatted = document.getElementById('sale_price_format');
        inputFormatted.addEventListener('input', function () {
            let clean = this.value.replace(/\D/g, "");
            this.value = formatRupiah(clean);
            priceInput.value = clean;
        });
        function updatePrice() {
            const selectedOption = carSelect.options[carSelect.selectedIndex];
            const price = selectedOption.getAttribute('data-price');
            if (price) {
                priceInput.value = price;
                inputFormatted.value = formatRupiah(price);
            } else {
                priceInput.value = '';
            }
        }
        carSelect.addEventListener('change', updatePrice);
        // Jika sebelumnya sudah terisi (old input), jalankan sekali
        updatePrice();
    });
</script>
@endpush
@endsection
```

resources/views/admin/salesRecord/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        <div class="card mb-4">
            <div class="card-body">
```

```
            {{-- Alert Notifikasi Sukses --}}
            @if (session('success'))
            <div class="alert alert-success alert-dismissible fade show" role="alert">
               {{ session('success') }}
               <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
            </div>
            @endif
            {{-- Tombol Create mobil baru --}}
            <div class="d-flex justify-content-between mb-3">
               <a href="{{ route('admin.salesRecord.create') }}" class="btn btn-
primary">Create</a>
            </div>

            {{-- Tabel mobil dengan fitur DataTables --}}
            <div class="table-responsive">
               <table id="cars-table" class="table table-bordered">
                  <thead>
                     <tr>
                        <th>ID</th>
                        <th>Create At</th>
                        <th>Car</th>
                        <th>Saler</th>
                        <th>Buyer</th>
                        <th>Sale Price</th>
                        <th>Sale Date</th>
                        <th>Action</th>
                     </tr>
                  </thead>
                  <tbody>
                     @forelse ($salesRecords as $salesRecord)
                     <tr>
                        <td>{{ $salesRecord->id }}</td>
                        <td>{{ \Carbon\Carbon::parse($salesRecord->created_at)-
>translatedFormat('d F Y H:i') }}</td>
                         <td><a href="" class="btn-show-detail" data-id="{{ $salesRecord-
>car->id }}"
                               data-bs-toggle="modal"
                               data-bs-target="#carModal">{{ $salesRecord->car->brand . ' ' .
$salesRecord->car->model . ' ' . $salesRecord->car->year }}</a>
                        </td>
                        <td>{{ $salesRecord->saler->name }}</td>
                        <td>{{ $salesRecord->buyer->name }}</td>
                        <td>Rp. {{ number_format($salesRecord->sale_price, 0, ',', '.')
}}</td>
                        <td>{{ \Carbon\Carbon::parse($salesRecord->sale_date)-
>translatedFormat('d F Y H:i') }}</td>
                        <td>
                           <form action="{{ route('admin.salesRecord.destroy', $salesRecord)
}}" method="POST" class="d-inline">
```

```
                        @csrf
                        @method('DELETE')
                        <button type="submit" class="btn btn-danger btn-sm"
onclick="return confirm('Are you sure you want to delete this record?')">Delete</button>
                    </form>
                  </td>
                </tr>
                @empty
                <tr>
                  <td colspan="8" class="text-center">No sales records found.</td>
                </tr>
                @endforelse

              </tbody>
            </table>
          </div>
          <!-- Modal kosong untuk show, akan diisi konten AJAX -->
          <div class="modal fade" id="carModal" tabindex="-1" aria-
labelledby="carModalLabel" aria-hidden="true">
            <div class="modal-dialog modal-lg">
              <div class="modal-content" id="carModalContent">
                <!-- Konten modal akan di-load via AJAX -->
                <div class="modal-body text-center">
                  <div class="spinner-border text-primary" role="status">
                    <span class="visually-hidden">Loading...</span>
                  </div>
                </div>
              </div>
            </div>
          </div>
          {{-- Pagination --}}
          <div class="card-footer clearfix">
            <div class="pagination pagination-sm m-0 float-end">
              {{ $salesRecords->appends(request()->query())-
>links('pagination::bootstrap-5') }}
            </div>
          </div>
        </div>
      </div>
    </div>
</div>
@push('scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script>
    $(document).ready(function () {
    // ================================
    // 1. Show detail mobil ke dalam modal
    // ================================
    $('.btn-show-detail').on('click', function () {
      let carId = $(this).data('id');
```

```
            let modalContent = $('#carModalContent');
            modalContent.html(`
          <div class="modal-body text-center">
            <div class="spinner-border text-primary" role="status">
              <span class="visually-hidden">Loading...</span>
            </div>
          </div>
        `);
            $.ajax({
                url: `/admin/cars/${carId}`, // Show route: menampilkan detail mobil
                method: 'GET',
                success: function (response) {
                    modalContent.html(response); // Load ke dalam modal
                },
                error: function () {
                    modalContent.html(
                        '<div class="modal-body text-danger">Failed to load data.</div>'
                        );
                }
            });
        });
    });
    </script>
@endpush
@endsection
```

resources/views/admin/Users/create.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Create User</div>
            </div>
            <!--end::Header-->
            <!--begin::Error Alert-->
            @if ($errors->any())
                <div class="alert alert-danger">
                    <ul class="mb-0">
                        @foreach ($errors->all() as $error)
                            <li>{{ $error }}</li>
                        @endforeach
                    </ul>
                </div>
            @endif
            <!--end::Error Alert-->
            <!--begin::Form-->
```

```
        <form action="{{ route('admin.users.store') }}" method="POST"
enctype="multipart/form-data">
            @csrf
            <div class="card-body">
                <div class="mb-3">
                    <label for="exampleInputName" class="form-label">Name</label>
                    <input type="text" class="form-control @error('name') is-invalid
@enderror" id="exampleInputName" name="name" value="{{ old('name') }}" required
/>
                    @error('name')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>

                <div class="mb-3">
                    <label for="exampleInputEmail1" class="form-label">Email
address</label>
                    <input type="email" class="form-control @error('email') is-invalid
@enderror" id="exampleInputEmail1" name="email" value="{{ old('email') }}" required
/>
                    @error('email')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>

                <div class="mb-3">
                    <label for="exampleInputPassword1" class="form-
label">Password</label>
                    <input type="password" class="form-control @error('password') is-invalid
@enderror" id="exampleInputPassword1" name="password" required />
                    @error('password')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>

                <div class="mb-3">
                    <label for="exampleInputPasswordConfirm" class="form-label">Confirm
Password</label>
                    <input type="password" class="form-control
@error('password_confirmation') is-invalid @enderror"
id="exampleInputPasswordConfirm" name="password_confirmation" required />
                    @error('password_confirmation')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
```

```
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="role" class="form-label">Role</label>
                <div class="form-check">
                    <input class="form-check-input" type="radio" name="role"
id="role_admin" value="admin" {{ old('role') == 'admin' ? 'checked' : '' }} />
                    <label class="form-check-label" for="role_admin">Admin</label>
                </div>
                <div class="form-check">
                    <input class="form-check-input" type="radio" name="role"
id="role_user" value="user" {{ old('role') == 'user' ? 'checked' : '' }} />
                    <label class="form-check-label" for="role_user">User</label>
                </div>
                @error('role')
                <span class="text-danger small d-block mt-1">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="phone" class="form-label">Phone</label>
                <input type="number" class="form-control @error('phone') is-invalid
@enderror" id="phone" name="phone" value="{{ old('phone') }}" required />
                @error('phone')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div class="mb-3">
                <label for="address" class="form-label">Address</label>
                <textarea cols="10" rows="2" class="form-control @error('address') is-
invalid @enderror" id="address" name="address" required>{{ old('address')
}}</textarea>
                @error('address')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
        </div>
        <div class="card-footer">
            <button type="submit" class="btn btn-primary">Submit</button>
        </div>
    </form>
    <!--end::Form-->
</div>
```

```
    </div>
    <!--end::Container-->
</div>
@endsection
```

resources/views/admin/Users/edit.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <!--begin::Container-->
    <div class="container-fluid">
        <div class="card card-primary card-outline mb-4">
            <!--begin::Header-->
            <div class="card-header">
                <div class="card-title">Edit User ID #{{ $user->id }}</div>
            </div>
            <!--end::Header-->
            <!--begin::Error Alert-->
            @if ($errors->any())
                <div class="alert alert-danger">
                    <ul class="mb-0">
                        @foreach ($errors->all() as $error)
                            <li>{{ $error }}</li>
                        @endforeach
                    </ul>
                </div>
            @endif
            <!--end::Error Alert-->
            <!--begin::Form-->
            <form action="{{ route('admin.users.update',$user->id) }}" method="POST"
enctype="multipart/form-data">
                @csrf
                @method('PUT')
                <div class="card-body">
                    <div class="mb-3">
                        <label for="exampleInputName" class="form-label">Name</label>
                        <input type="text" class="form-control @error('name') is-invalid
@enderror" id="exampleInputName" name="name" value="{{ $user->name }}" required
/>
                        @error('name')
                        <span class="invalid-feedback" role="alert">
                            <strong>{{ $message }}</strong>
                        </span>
                        @enderror
                    </div>

                    <div class="mb-3">
                        <label for="exampleInputEmail1" class="form-label">Email
address</label>
```

```
                <input type="email" class="form-control @error('email') is-invalid
@enderror" id="exampleInputEmail1" name="email" value="{{ $user->email }}"
required />
                @error('email')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="exampleInputPassword1" class="form-
label">Password</label>
                <input type="password" class="form-control @error('password') is-invalid
@enderror" id="exampleInputPassword1" name="password"  />
                @error('password')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="exampleInputPasswordConfirm" class="form-label">Confirm
Password</label>
                <input type="password" class="form-control
@error('password_confirmation') is-invalid @enderror"
id="exampleInputPasswordConfirm" name="password_confirmation"  />
                @error('password_confirmation')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="role" class="form-label">Role</label>
                <div class="form-check">
                    <input class="form-check-input" type="radio" name="role"
id="role_admin" value="admin" {{ $user->role == 'admin' ? 'checked' : '' }} />
                    <label class="form-check-label" for="role_admin">Admin</label>
                </div>
                <div class="form-check">
                    <input class="form-check-input" type="radio" name="role"
id="role_user" value="user" {{ $user->role == 'user' ? 'checked' : '' }} />
                    <label class="form-check-label" for="role_user">User</label>
                </div>
                @error('role')
                <span class="text-danger small d-block mt-1">
                    <strong>{{ $message }}</strong>
```

```
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="phone" class="form-label">Phone</label>
                <input type="number" class="form-control @error('phone') is-invalid
@enderror" id="phone" name="phone" value="{{ $user->phone }}" required />
                @error('phone')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>

            <div class="mb-3">
                <label for="address" class="form-label">Address</label>
                <textarea cols="10" rows="2" class="form-control @error('address') is-
invalid @enderror" id="address" name="address" required>{{ $user->address
}}</textarea>
                @error('address')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
        </div>
        <div class="card-footer">
            <button type="submit" class="btn btn-primary">Submit</button>
        </div>
    </form>
    <!--end::Form-->
  </div>
 </div>
 <!--end::Container-->
</div>
@endsection
```

resources/views/admin/Users/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
 <!--begin::Container-->
 <div class="container-fluid">
  <div class="card mb-4">
   {{-- <div class="card-header"><h3 class="card-title">Bordered Table</h3></div> --
}}
   <div class="card-body">
    {{-- Alert Notifikasi Sukses --}}
    @if (session('success'))
```

```
        <div class="alert alert-success alert-dismissible fade show" role="alert">
          {{ session('success') }}
          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>
      @endif
      {{-- End Alert Notifikasi Sukses --}}
      <div class="d-flex justify-content-between mb-3">
        <a href="{{ route('admin.users.create') }}" class="btn btn-primary">Create</a>
      </div>
      {{-- Form Search --}}
      <form action="{{ route('admin.users.index') }}" method="GET" class="row g-3
align-items-center mb-3" role="search">
        <div class="col-md-4">
          <input type="text" name="search" class="form-control" placeholder="Search by
name or email" value="{{ request('search') }}">
        </div>
        <div class="col-md-4">
          <select name="role" class="form-select">
            <option value="">All Roles</option>
            <option value="admin" {{ request('role') == 'admin' ? 'selected' : ''
}}>Admin</option>
            <option value="user" {{ request('role') == 'user' ? 'selected' : '' }}>User</option>
          </select>
        </div>
        <div class="col-md-4">
          <input type="text" name="address" class="form-control" placeholder="Search
Address" value="{{ request('address') }}">
        </div>
        <div class="col-12">
          <button type="submit" class="btn btn-success">Search</button>
          <a href="{{ route('admin.users.index') }}" class="btn btn-secondary">Reset</a>
        </div>
      </form>
      {{-- End Form Search --}}
      {{-- Table --}}
      <div class="table-responsive">
        <table class="table table-bordered">
          <thead>
            <tr>
              <th style="width: 10px">ID</th>
              <th>Created At</th>
              <th>Name</th>
              <th>Email</th>
              <th>Role</th>
              <th>Phone</th>
              <th>Address</th>
              <th>Action</th>
            </tr>
          </thead>
```

```
          <tbody>
            @forelse ($users as $item)
              <tr class="align-middle">
                <td>{{ $item->id }}</td>
                <td>{{ \Carbon\Carbon::parse($item->created_at)->translatedFormat('d F Y
H:i') }}</td>
                <td>{{ $item->name }}</td>
                <td>{{ $item->email }}</td>
                <td>{{ $item->role }}</td>
                <td>{{ $item->phone  ?? '-'}}</td>
                <td>{{ $item->address ?? '-' }}</td>
                <td>
                  <a href="{{ route('admin.users.edit', $item->id) }}" class="btn btn-warning
btn-sm">Edit</a>
                  <form action="{{ route('admin.users.destroy', $item->id) }}"
method="POST" class="d-inline">
                    @csrf
                    @method('DELETE')
                    <button type="submit" class="btn btn-danger btn-sm" onclick="return
confirm('Are you sure you want to delete this user?')">Delete</button>
                  </form>
              </tr>
            @empty
              <tr>
                <td colspan="8" class="text-center">No users found.</td>
              </tr>
            @endforelse
          </tbody>
        </table>
      </div>
      {{-- End Table --}}
    </div>
    <div class="card-footer clearfix">
      <div class="pagination pagination-sm m-0 float-end">
        {{ $users->appends(['search' => request('search')])->links('pagination::bootstrap-5')
}}
      </div>
    </div>
  </div>
 </div>
</div>
@endsection
```

resources/views/auth/passwords/confirm.blade.php

```
@extends('layouts.app')
@section('content')
<div class="container">
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card">
```

```
            <div class="card-header">{{ __('Confirm Password') }}</div>
            <div class="card-body">
                {{ __('Please confirm your password before continuing.') }}
                <form method="POST" action="{{ route('password.confirm') }}">
                    @csrf
                    <div class="row mb-3">
                        <label for="password" class="col-md-4 col-form-label text-md-end">{{
__('Password') }}</label>
                        <div class="col-md-6">
                            <input id="password" type="password" class="form-control
@error('password') is-invalid @enderror" name="password" required
autocomplete="current-password">
                            @error('password')
                                <span class="invalid-feedback" role="alert">
                                    <strong>{{ $message }}</strong>
                                </span>
                            @enderror
                        </div>
                    </div>
                    <div class="row mb-0">
                        <div class="col-md-8 offset-md-4">
                            <button type="submit" class="btn btn-primary">
                                {{ __('Confirm Password') }}
                            </button>
                            @if (Route::has('password.request'))
                                <a class="btn btn-link" href="{{ route('password.request') }}">
                                    {{ __('Forgot Your Password?') }}
                                </a>
                            @endif
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
  </div>
</div>
@endsection
```

resources/views/auth/passwords/email.blade.php

```
@extends('layouts.app')
@section('content')
<div class="container">
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card">
        <div class="card-header">{{ __('Reset Password') }}</div>
        <div class="card-body">
            @if (session('status'))
                <div class="alert alert-success" role="alert">
```

```
                {{ session('status') }}
            </div>
        @endif
        <form method="POST" action="{{ route('password.email') }}">
            @csrf
            <div class="row mb-3">
                <label for="email" class="col-md-4 col-form-label text-md-end">{{
__('Email Address') }}</label>
                <div class="col-md-6">
                    <input id="email" type="email" class="form-control @error('email')
is-invalid @enderror" name="email" value="{{ old('email') }}" required
autocomplete="email" autofocus>
                    @error('email')
                        <span class="invalid-feedback" role="alert">
                            <strong>{{ $message }}</strong>
                        </span>
                    @enderror
                </div>
            </div>
            <div class="row mb-0">
                <div class="col-md-6 offset-md-4">
                    <button type="submit" class="btn btn-primary">
                        {{ __('Send Password Reset Link') }}
                    </button>
                </div>
            </div>
        </form>
    </div>
  </div>
 </div>
</div>
</div>
@endsection
```

resources/views/errors/404.blade.php

```
@extends('layouts.app')
@section('content')
<div class="container text-center py-5">
  {{-- <img src="{{asset('image/YONO-MOBILINDO-LOGO.svg')}}" alt="YONO
MOBILINDO LOGO"> --}}
  <h1 class="display-4 text-danger">404</h1>
  <h3 class="mb-3">Halaman Tidak Ditemukan</h3>
  <p class="text-muted mb-4">Maaf, link yang Anda buka tidak tersedia atau sudah
dipindahkan.</p>
  <a href="{{ route('home') }}" class="btn btn-login">Kembali ke Beranda</a>
</div>
@endsection
```

resources/views/layouts/app.blade.php

```
<!doctype html>
```

```
<html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
<head>
   <meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <!-- CSRF Token -->
   <meta name="csrf-token" content="{{ csrf_token() }}">
   <title>{{ config('app.name', 'YONO MOBILINDO') }}</title>
   <!-- Fonts -->
   <link rel="dns-prefetch" href="//fonts.bunny.net">
   <link href="https://fonts.bunny.net/css?family=Nunito" rel="stylesheet">
   {{-- Icon --}}
   <link rel="icon" type="image/png" href="{{ asset('image/YONO-MOBILINDO-
LOGO.svg') }}">
   <!-- Bootstrap & Vite -->
   @vite(['resources/sass/app.scss', 'resources/js/app.js'])
   <!-- font montserat -->
   <link rel="preconnect" href="https://fonts.googleapis.com" />
   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
   <link
href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..
900&display=swap"
      rel="stylesheet" />
   <style>
      :root {
         --background: #fffdf6;
         --black: #171513;
         --grey: #867e77;
         --primary: #27548a;
         --yellow: #fcf259;
      }
      * {
         margin: 0;
         padding: 0;
         box-sizing: border-box;
         font-family: "Montserrat", sans-serif;
         color: var(--black);
      }
      body {
         background-color: var(--background);
      }

      #app {
         display: flex;
         flex-direction: column;
         min-height: 100vh;
      }
      .btn-login {
         background-color: var(--primary);
         color: var(--yellow);
         font-weight: bold;
```

```
            border: none;
            transition: background-color 0.3s ease, color 0.3s ease, transform 0.2s ease;
        }
        .btn-login:hover {
            /* Warna lebih gelap dari primary */
            background-color: #1f3f6b;
            /* Warna teks putih */
            color: #fff;
            /* Efek bayangan halus */
            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.2);
        }
        header {
            background-color: #ffffff;
            padding: 1rem 0;
            box-shadow: 0 2px 6px rgba(0, 0, 0, 0.05);
            text-align: center;
        }
        header img {
            max-width: 200px;
            height: auto;
        }
        main {
            flex: 1;
            padding: 2rem 1rem;
        }
        footer {
            background-color: #032145;
            color: #fff;
            text-align: center;
            padding: 1rem;
            font-size: 14px;
        }
    </style>
</head>
<body>
    <div id="app">
        <header>
            <img src="{{ asset('image/YONO-MOBILINDO-LOGO.svg') }}" alt="YONO
MOBILINDO LOGO">
        </header>
        <main class="container">
            @yield('content')
        </main>
        <footer>
            &copy; {{ date('Y') }} YONO MOBILINDO. All rights reserved.
        </footer>
    </div>
</body>
</html>
```

resources/views/layouts/contentHeader.blade.php

```
<!--begin::App Content Header-->
<div class="app-content-header">
   <!--begin::Container-->
   <div class="container-fluid">
      <!--begin::Row-->
      <div class="row">
         <div class="col-sm-6">
            <h3 class="mb-0">
               @switch(true)
                  @case(request()->routeIs('admin.dashboard') || request()->routeIs('home-
cms') )
                     Dashboard
                     @break
                  @case(request()->routeIs('admin.users.*'))
                     Users
                     @break
                  @case(request()->routeIs('admin.cars.*'))
                     Mobil
                     @break
                  @case(request()->routeIs('admin.downPayment.edit'))
                     Refund DP
                     @break
                  @case(request()->routeIs('admin.downPayment.*'))
                     Down Payment(DP)
                     @break
                  @case(request()->routeIs('admin.offer.*'))
                     Permintaan Penjualan
                     @break
                  @case(request()->routeIs('admin.salesRecord.*'))
                     Catatan Transaksi Pembelian
                     @break
                  @case(request()->routeIs('admin.offerRecord.*'))
                     Catatan Transaksi Penjualan
                     @break
                  @case(request()->routeIs('user.dashboard') || request()->routeIs('home-
cms') )
                     Dashboard
                     @break
                  @case(request()->routeIs('user.downPayment.*'))
                     Pembayaran DP
                     @break
                  @case(request()->routeIs('user.offer.*'))
                     Penjualan Mobil
                     @break
                  @case(request()->routeIs('user.transactionSalesRecord.*'))
                     Transaksi Pembelian Mobil
                     @break
                  @case(request()->routeIs('user.transactionOfferRecord.*'))
                     Transaksi Penjualan Mobil
                     @break
```

```
                @default
                    Halaman
            @endswitch
        </h3>
    </div>
    <div class="col-sm-6">
        <ol class="breadcrumb float-sm-end">
            <li class="breadcrumb-item">
                <a href="{{ route('home-cms') }}">Home</a>
            </li>
            @switch(true)
                {{-- admin.Dashboard --}}
                @case(request()->routeIs('admin.dashboard') || request()->routeIs('home-
cms') )
                    <li class="breadcrumb-item active" aria-
current="page">Dashboard</li>
                    @break
                {{-- admin.users.index --}}
                @case(request()->routeIs('admin.users.index'))
                    <li class="breadcrumb-item active" aria-current="page">Users</li>
                    @break
                {{-- admin.users.create --}}
                @case(request()->routeIs('admin.users.create'))
                    <li class="breadcrumb-item">
                        <a href="{{ route('admin.users.index') }}">Users</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Create</li>
                    @break
                {{-- admin.users.edit --}}
                @case(request()->routeIs('admin.users.edit'))
                    <li class="breadcrumb-item">
                        <a href="{{ route('admin.users.index') }}">Users</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Edit</li>
                    @break
                {{-- admin.cars.index --}}
                @case(request()->routeIs('admin.cars.index'))
                    <li class="breadcrumb-item active" aria-current="page">Mobil</li>
                    @break
                {{-- admin.cars.create --}}
                @case(request()->routeIs('admin.cars.create'))
                    <li class="breadcrumb-item">
                        <a href="{{ route('admin.cars.index') }}">Mobil</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Create</li>
                    @break
                {{-- admin.cars.edit --}}
                @case(request()->routeIs('admin.cars.edit'))
                    <li class="breadcrumb-item">
                        <a href="{{ route('admin.cars.index') }}">Mobil</a>
```

```
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Edit</li>
                    @break
                {{-- admin.offer.index --}}
                @case(request()->routeIs('admin.offer.index'))
                    <li class="breadcrumb-item active" aria-current="page">Permintaan
Penjualan</li>
                    @break
                {{-- admin.offer.show --}}
                @case(request()->routeIs('admin.offer.show'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('admin.offer.index')}}">Permintaan Penjualan</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Show</li>
                    @break

                {{-- admin.downPayment.index --}}
                @case(request()->routeIs('admin.downPayment.index'))
                    <li class="breadcrumb-item active" aria-current="page">Down
Payment(DP)</li>
                    @break
                {{-- admin.downPayment.addRefund --}}
                @case(request()->routeIs('admin.downPayment.addRefund'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('admin.downPayment.index')}}">Down
Payment</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Refund</li>
                    @break
                {{-- admin.downPayment.editRefund --}}
                @case(request()->routeIs('admin.downPayment.editRefund'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('admin.downPayment.index')}}">Down
Payment</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Edit
Refund</li>
                    @break
                {{-- admin.salesRecord.index --}}
                @case(request()->routeIs('admin.salesRecord.index'))
                    <li class="breadcrumb-item active" aria-current="page">Catatan
Transaksi Pembelian</li>
                    @break
                {{-- admin.salesRecord.create --}}
                @case(request()->routeIs('admin.salesRecord.create'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('admin.salesRecord.index')}}">Catatan Transaksi
Pembelian</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Create</li>
```

```
                    @break
                {{-- admin.offerRecord.index --}}
                @case(request()->routeIs('admin.offerRecord.index'))
                    <li class="breadcrumb-item active" aria-current="page">Catatan
Transaksi Penjualan</li>
                    @break
                {{-- admin.offerRecord.create --}}
                @case(request()->routeIs('admin.offerRecord.create'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('admin.offerRecord.index')}}">Catatan Transaksi
Penjualan</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Create</li>
                    @break

                {{-- user.dashboard --}}
                @case(request()->routeIs('user.dashboard') || request()->routeIs('home-
cms') )
                    <li class="breadcrumb-item active" aria-
current="page">Dashboard</li>
                    @break
                {{--user.downPayment.index  --}}
                @case(request()->routeIs('user.downPayment.index'))
                    <li class="breadcrumb-item active" aria-current="page">Down
Payment</li>
                    @break
                {{-- user.downPayment.checkout --}}
                @case(request()->routeIs('user.downPayment.checkout'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('user.downPayment.index')}}">Down Payment</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Checkout</li>
                    @break
                @case(request()->routeIs('user.downPayment.show'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('user.downPayment.index')}}">Down Payment</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Show</li>
                    @break
                {{--user.Offer.index  --}}
                @case(request()->routeIs('user.offer.index'))
                    <li class="breadcrumb-item active" aria-current="page">Penjualan
Mobil</li>
                    @break
                {{-- user.offer.show --}}
                @case(request()->routeIs('user.offer.show'))
                    <li class="breadcrumb-item active" aria-current="page">
                        <a href="{{route('user.offer.index')}}">Penjualan Mobil</a>
                    </li>
                    <li class="breadcrumb-item active" aria-current="page">Show</li>
```

```
                    @break

                    {{-- user.transactionSalesRecord.index --}}
                    @case(request()->routeIs('user.transactionSalesRecord.index'))

                    <li class="breadcrumb-item active" aria-current="page">Transaksi
Pembelian Mobil</li>
                    @break

                    @case(request()->routeIs('user.transactionSalesRecord.createTesti'))
                    <li class="breadcrumb-item active" aria-current="page">
                       <a href="{{route('user.transactionSalesRecord.index')}}">Transaksi
Pembelian Mobil</a>
                    </li>

                    <li class="breadcrumb-item active" aria-current="page">Testimoni</li>
                    @break
                  {{-- user.transactionOfferRecord.index --}}
                  @case(request()->routeIs('user.transactionOfferRecord.index'))

                    <li class="breadcrumb-item active" aria-current="page">Transaksi
Penjualan Mobil</li>
                    @break

                  @default
                    <li class="breadcrumb-item active" aria-current="page">Halaman</li>
                @endswitch
              </ol>
            </div>
        </div>
        <!--end::Row-->
    </div>
    <!--end::Container-->
</div>
```

resources/views/layouts/css.blade.php

```
<!--begin::Fonts-->
<link
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/@fontsource/source-sans-3@5.0.12/index.css"
integrity="sha256-tXJfXfp6Ewt1ilPzLDtQnJV4hclT9XuaZUKyUvmyr+Q="
crossorigin="anonymous"
/>
<!--end::Fonts-->
<!--begin::Third Party Plugin(OverlayScrollbars)-->
<link
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/overlayscrollbars@2.10.1/styles/overlayscrollbars.min.
css"
integrity="sha256-tZHrRjVqNSRyWg2wbppGnT833E/Ys0DHWGwT04GiqQg="
```

```
crossorigin="anonymous"
/>
<!--end::Third Party Plugin(OverlayScrollbars)-->
<!--begin::Third Party Plugin(Bootstrap Icons)-->
<link
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.min.css"
integrity="sha256-9kPW/n5nn53j4WMRYAxe9c1rCY96Oogo/MKSVdKzPmI="
crossorigin="anonymous"
/>
<!--end::Third Party Plugin(Bootstrap Icons)-->
<!--begin::Required Plugin(AdminLTE)-->
<link rel="stylesheet" href="{{ asset('adminLTE/dist/css/adminlte.css')}}" />
<!--end::Required Plugin(AdminLTE)-->
<!-- apexcharts -->
<linkz
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/apexcharts@3.37.1/dist/apexcharts.css"
integrity="sha256-4MX+61mt9NVvvuPjUWdUdyfZfxSB1/Rf9WtqRHgG5S0="
crossorigin="anonymous"
/>
<!-- jsvectormap -->
<link
rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/jsvectormap@1.5.3/dist/css/jsvectormap.min.css"
integrity="sha256-+uGLJmmTKOqBr+2E6KDYs/NRsHxSkONXFHUL0fy2O/4="
crossorigin="anonymous"
/>
<!-- Font Awesome CDN -->
<link
  href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css"
  rel="stylesheet"
/>
<!-- Head -->
{{-- <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css"
rel="stylesheet" />
<link href="https://cdn.jsdelivr.net/npm/select2-bootstrap-5-theme@1.2.0/dist/select2-
bootstrap-5-theme.min.css" rel="stylesheet" /> --}}
```

resources/views/layouts/footer.blade.php

```
<footer class="app-footer">
  <!--begin::To the end-->
  <div class="float-end d-none d-sm-inline">Anything you want</div>
  <!--end::To the end-->
  <!--begin::Copyright-->
  <strong>
    Copyright &copy; 2014-2024 
    <a href="https://adminlte.io" class="text-decoration-none">AdminLTE.io</a>.
  </strong>
  All rights reserved.
```

```
  <!--end::Copyright-->
</footer>
```

resources/views/layouts/js.blade.php

```
<!--begin::Third Party Plugin(OverlayScrollbars)-->
<script
src="https://cdn.jsdelivr.net/npm/overlayscrollbars@2.10.1/browser/overlayscrollbars.bro
wser.es6.min.js" integrity="sha256-
dghWARbRe2eLlIJ56wNB+b760ywulqK3DzZYEpsg2fQ="
crossorigin="anonymous"></script>
<!--end::Third Party Plugin(OverlayScrollbars)-->
<!--begin::Required Plugin(popperjs for Bootstrap 5)-->
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"
integrity="sha384-
I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/vR+Vc4jQkC+hVqc2pM8ODewa9r"
crossorigin="anonymous"></script>
<!--end::Required Plugin(popperjs for Bootstrap 5)-->
<!--begin::Required Plugin(Bootstrap 5)-->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js"
integrity="sha384-
0pUGZvbkm6XF6gxjEnlmuGrJXVbNuzT9qBBavbLwCsOGabYfZo0T0to5eqruptLy"
crossorigin="anonymous"></script>
<!--end::Required Plugin(Bootstrap 5)-->
<!--begin::Required Plugin(AdminLTE)-->
<script src="{{ asset('adminLTE/dist/js/adminlte.js') }}"></script>
<!--end::Required Plugin(AdminLTE)-->
<!--begin::OverlayScrollbars Configure-->
<script>
  const SELECTOR_SIDEBAR_WRAPPER = '.sidebar-wrapper';
  const Default = {
    scrollbarTheme: 'os-theme-light',
    scrollbarAutoHide: 'leave',
    scrollbarClickScroll: true
  };
  document.addEventListener('DOMContentLoaded', function() {
    const sidebarWrapper =
document.querySelector(SELECTOR_SIDEBAR_WRAPPER);
    if (sidebarWrapper && typeof OverlayScrollbarsGlobal?.OverlayScrollbars !==
'undefined') {
      OverlayScrollbarsGlobal.OverlayScrollbars(sidebarWrapper, {
        scrollbars: {
          theme: Default.scrollbarTheme
          , autoHide: Default.scrollbarAutoHide
          , clickScroll: Default.scrollbarClickScroll
        , }
      , });
    }
  });
</script>
```

```
<!--end::OverlayScrollbars Configure-->
<!-- Footer Scripts -->
{{-- <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/js/select2.min.js"></script> --}}
```

resources/views/layouts/master.blade.php

```
<!doctype html>
<html lang="en">
  <!--begin::Head-->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Yono Mobilindo</title>
    <!--begin::Primary Meta Tags-->
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta name="title" content="Yono Mobilindo Administrator" />
    {{-- <meta name="author" content="ColorlibHQ" /> --}}

    <meta
      name="description"
      content="Administrator YonoMobilindo"
    />
    <meta
      name="keywords"
      content="Administrator YonoMobilindo"/>
    {{-- icon --}}
    <link rel="icon" type="image/png" href="{{ asset('image/YONO-MOBILINDO-
LOGO.svg') }}">
    <!--end::Primary Meta Tags-->

    @include('layouts.css')
    @stack('styles')
  </head>
  <!--end::Head-->

  <!--begin::Body-->
  <body class="layout-fixed sidebar-expand-lg bg-body-tertiary">
    <!--begin::App Wrapper-->
    <div class="app-wrapper">
      <!--begin::Header-->
      @include('layouts.navbar')
      <!--end::Header-->
      <!--begin::Sidebar-->
      @include('layouts.sidebar')
      <!--end::Sidebar-->
      <!--begin::App Main-->
      <main class="app-main">
        <!--begin::App Content Header-->
        @include('layouts.contentHeader')
        <!--end::App Content Header-->
```

```
      <!--begin::App Content-->
      <div class="container">

      </div>
      @yield('content')
      <!--end::App Content-->
    </main>
    <!--end::App Main-->
    <!--begin::Footer-->
    @include('layouts.footer')
    <!--end::Footer-->
  </div>
  <!--end::App Wrapper-->
  <!--begin::Script-->
  @include('layouts.js')
  @stack('scripts')
  <!--end::Script-->
 </body>
 <!--end::Body-->
</html>
```

resources/views/layouts/navbar.blade.php

```
<nav class="app-header navbar navbar-expand bg-body">
 <!--begin::Container-->
 <div class="container-fluid">
  <!--begin::Start Navbar Links-->
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link" data-lte-toggle="sidebar" href="#" role="button">
        <i class="bi bi-list"></i>
      </a>
    </li>
  </ul>
  <!--end::Start Navbar Links-->
  <!--begin::End Navbar Links-->
  <ul class="navbar-nav ms-auto">
    <!--end::Notifications Dropdown Menu-->
    <!--begin::Fullscreen Toggle-->
    <li class="nav-item">
      <a class="nav-link" href="#" data-lte-toggle="fullscreen">
        <i data-lte-icon="maximize" class="bi bi-arrows-fullscreen"></i>
        <i data-lte-icon="minimize" class="bi bi-fullscreen-exit" style="display:
none"></i>
      </a>
    </li>
    <!--end::Fullscreen Toggle-->
    <!--begin::User Menu Dropdown-->
    <li class="nav-item dropdown user-menu">
      <a href="#" class="nav-link dropdown-toggle" data-bs-toggle="dropdown">
        <img
```

```
          src="{{ asset('adminLTE/dist/assets/img/user2-160x160.jpg') }}"
          class="user-image rounded-circle shadow"
          alt="User Image"
        />
        <span class="d-none d-md-inline">Hi, {{ Auth::user()->name }}</span>
      </a>
      <ul class="dropdown-menu dropdown-menu-lg dropdown-menu-end">
        <!--begin::User Image-->
        <li class="user-header text-bg-primary">
          <img
            src="{{asset('adminLTE/dist/assets/img/user2-160x160.jpg') }}"
            class="rounded-circle shadow"
            alt="User Image"
          />
          <p>
            {{ Auth::user()->name }}
            <small>Member since  {{ Auth::user()->created_at->format('d M Y')}}</small>
          </p>
        </li>
        <!--end::User Image-->
        <!--begin::Menu Footer-->
        <li class="user-footer">

          <a class="btn btn-default btn-flat float-end" href="{{ route('logout') }}"
             onclick="event.preventDefault();
                      document.getElementById('logout-form').submit();">
             {{ __('Logout') }}
          </a>
          <form id="logout-form" action="{{ route('logout') }}" method="POST" class="d-
none">
             @csrf
          </form>
        </li>
        <!--end::Menu Footer-->
      </ul>
    </li>
    <!--end::User Menu Dropdown-->
  </ul>
  <!--end::End Navbar Links-->
 </div>
 <!--end::Container-->
</nav>
```

resources/views/layouts/sidebar.blade.php

```
<aside class="app-sidebar bg-body-secondary shadow" data-bs-theme="dark">
 <!--begin::Sidebar Brand-->
 <div class="sidebar-brand">
  <!--begin::Brand Link-->
  <a href="{{route('home')}}" class="brand-link">
   <!--begin::Brand Image-->
```

```
      <img
        src="{{ asset('image/YONO MOBILINDO LOGO 2.png') }}"
        alt="YONO MOBILINDO"
        class="brand-image  shadow"
      />
      <!--end::Brand Image-->
      <!--begin::Brand Text-->
      <span class="brand-text fw-light">YONOMOBILINDO</span>
      <!--end::Brand Text-->
    </a>
    <!--end::Brand Link-->
  </div>
  <!--end::Sidebar Brand-->
  <!--begin::Sidebar Wrapper-->
  <div class="sidebar-wrapper">
    <nav class="mt-2">
      <!--begin::Sidebar Menu-->
      <ul
        class="nav sidebar-menu flex-column"
        data-lte-toggle="treeview"
        role="menu"
        data-accordion="false"
      >
      @auth
        @if(auth()->user()->role === 'admin')
          <li class="nav-item">
            <a href="{{ route('admin.dashboard') }}" class="nav-link {{ request()-
>routeIs('admin.dashboard') || request()->routeIs('home-cms')   ? 'active' : '' }}">
              <i class="nav-icon bi bi-speedometer"></i>
              <p>
                Dashboard
              </p>
            </a>
          </li>
          <li class="nav-item">
            <a href="{{ route('admin.cars.index') }}" class="nav-link {{ request()-
>routeIs('admin.cars.*') ? 'active' : '' }}">
              <i class=" nav-icon fa-solid fa-car"></i>
              <p>
                Mobil
              </p>
            </a>
          </li>
          <li class="nav-item">
            <a href="{{ route('admin.downPayment.index') }}" class="nav-link {{ request()-
>routeIs('admin.downPayment.*') ? 'active' : '' }}">
              <i class="nav-icon fa-solid fa-money-bill"></i>
              <p>
                DownPayment(DP)
              </p>
```

```
          </a>
        </li>
        <li class="nav-item">
          <a href="{{ route('admin.salesRecord.index') }}" class="nav-link {{ request()-
>routeIs('admin.salesRecord.*') ? 'active' : '' }}">
            {{-- <i class="nav-icon fa-solid fa-cart"></i> --}}
            <i class="fa-solid fa-cart-shopping"></i>
            <p>
              Catatan Transaksi Pembelian
            </p>
          </a>
        </li>
        <li class="nav-item">
          <a href="{{ route('admin.offer.index') }}" class="nav-link {{ request()-
>routeIs('admin.offer.*') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-store"></i>
            <p>
              Permintaan Penjualan Mobil
            </p>
          </a>
        </li>
        <li class="nav-item">
          <a href="{{ route('admin.offerRecord.index') }}" class="nav-link {{ request()-
>routeIs('admin.offerRecord.*') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-shop"></i>
            <p>
              Catatan Transaksi Penjualan
            </p>
          </a>
        </li>
        @elseif(auth()->user()->role === 'user')
        <li class="nav-item">
          <a href="{{ route('user.dashboard')}}" class="nav-link {{ request()-
>routeIs('user.dashboard') || request()->routeIs('home-cms') ? 'active' : '' }}">
            <i class="nav-icon bi bi-speedometer"></i>
            <p>
              Dashboard
            </p>
          </a>
        </li>
        <li class="nav-item">
          <a href="{{route('user.downPayment.index')}}" class="nav-link {{ request()-
>routeIs('user.downPayment.*') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-money-bill"></i>
            <p>
              DownPayment(DP)
            </p>
          </a>
        </li>
        <li class="nav-item">
```

```
          <a href="{{route('user.transactionSalesRecord.index')}}" class="nav-link {{
request()->routeIs('user.transactionSalesRecord.index') || request()-
>routeIs('user.transactionSalesRecord.createTesti') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-cart-shopping"></i>
            <p>
              Transaksi Pembelian
            </p>
          </a>
        </li>
        <li class="nav-item">
          <a href="{{route('user.offer.index')}}" class="nav-link {{ request()-
>routeIs('user.offer.*') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-car-side"></i>
            <p>
              Permintaan Penjualan
            </p>
          </a>
        </li>
        <li class="nav-item">
          <a href="{{route('user.transactionOfferRecord.index')}}" class="nav-link {{
request()->routeIs('user.transactionOfferRecord.*') ? 'active' : '' }}">
            <i class="nav-icon fa-solid fa-cart-shopping"></i>
            <p>
              Transaksi Penjualan
            </p>
          </a>
        </li>
      @endif
    @endauth
    </ul>
    <!--end::Sidebar Menu-->
  </nav>
 </div>
 <!--end::Sidebar Wrapper-->
</aside>
```

resources/views/user/dashboard/index.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
  <style>
      .hover-shadow:hover {
          box-shadow: 0 0.5rem 1rem rgba(0,0,0,0.15) !important;
          transform: translateY(-2px);
          transition: all 0.3s;
      }
  </style>
@endpush
<div class="app-content">
  <div class="container-fluid">
```

```
<div class="row">
    <div class="col-12 col-sm-6 col-lg-4 mb-3">
        <div class="bg-light border rounded shadow-sm p-3 d-flex align-items-center
hover-shadow position-relative">
            <div class="bg-success text-white rounded d-flex justify-content-center
align-items-center" style="width: 50px; height: 50px;">
                <i class="bi bi-cart-fill fs-4"></i>
            </div>
            <div class="ms-3 flex-grow-1">
                <div class="fw-bold text-dark">Transaksi Pembelian Mobil</div>
                <div class="text-muted">Total: {{$salesRecord}}</div>
                    <a href="{{ route('user.transactionSalesRecord.index') }}" class="text-
decoration-none">
                        <small class="text-success d-block mt-1">More info <i class="bi bi-
arrow-right"></i></small>
                    </a>
                </div>
            </div>
    </div>
    <div class="col-12 col-sm-6 col-lg-4 mb-3">
        <div class="bg-light border rounded shadow-sm p-3 d-flex align-items-center
hover-shadow position-relative">
            <div class="bg-primary text-white rounded-circle d-flex justify-content-
center align-items-center" style="width: 50px; height: 50px;">
                <i class="bi bi-cart-fill fs-4"></i>
            </div>
            <div class="ms-3 flex-grow-1">
                <div class="fw-bold text-dark">Transaksi Penjualan Mobil</div>
                <div class="text-muted">Total: {{ $offerRecord }}</div>
                    <a href="{{ route('user.transactionOfferRecord.index') }}" class="text-
decoration-none">
                        <small class="text-primary d-block mt-1">More info <i class="bi bi-
arrow-right"></i></small>
                    </a>
                </div>
            </div>
    </div>
    {{-- Pembayaran DP Pending --}}
    <div class="col-12 mb-4">
        @if ($downPayment && $downPayment->payment_status === 'pending')
        <div class="card  card-warning collapsed-card">
            <div class="card-header" data-lte-toggle="card-collapse">
                <h3 class="card-title">{{ '#'. $downPayment->id ?? '-'}} Down Payment
Pending</h3>
                <div class="card-tools">
                    <button type="button" class="btn btn-tool" data-lte-toggle="card-
collapse">
                        <i data-lte-icon="expand" class="bi bi-plus-lg"></i>
                        <i data-lte-icon="collapse" class="bi bi-dash-lg"></i>
                    </button>
```

```
                </div>
                <!-- /.card-tools -->
            </div>
            <!-- /.card-header -->
            <div class="card-body">
                <table class="table table-bordered">
                    <tr>
                        <th>ID Pembayaran</th>
                        <td>{{ '#'. $downPayment->id ?? '-'}}</td>
                    </tr>
                    <tr>
                        <th>Dibuat</th>
                        <td>{{ \Carbon\Carbon::parse($downPayment->created_at)-
>format('d M Y') }}</td>
                    </tr>
                    <tr>
                        <th>Mobil</th>
                        <td class="text-uppercase">#{{ $downPayment->car_id }}
                            {{ $downPayment->car->brand  . ' ' . $downPayment->car->model
. ' ' . $downPayment->car->year . ' '. $downPayment->car->color ?? '-' }}
                        </td>
                    </tr>
                    <tr>
                        <th>Tanggal Janji</th>
                        <td>{{ \Carbon\Carbon::parse($downPayment->appointment_date)-
>format('d M Y') }}</td>
                    </tr>
                    <tr>
                        <th>Jam Janji</th>
                        <td>{{\Carbon\Carbon::parse($downPayment->appointment_date)-
>format('H:i') . ' WIB'  }}
                        </td>
                    </tr>
                    <tr>
                        <th>Lokasi Showroom</th>
                        <td>Lokasi: Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU
BARU TAPOS, DEPOK<br>
                            Kontak Showroom: 081220745317</td>
                    </tr>
                    <tr>
                        <th colspan="2" class="text-muted">Informasi Pembayaran</th>
                    </tr>
                    <tr>
                        <th>Order ID</th>
                        <td>{{ $downPayment->order_id ?? '-' }}</td>
                    </tr>
                    <tr>
                        <th>Jumlah Pembayaran</th>
                        <td>Rp {{ number_format($downPayment->amount, 0, ',', '.')
}}</td>
```

```
                    </tr>
                  </table>
                  {{-- Tombol bayar --}}
                  @if($downPayment->payment_status == 'pending' && $downPayment-
>car->status == "available")
                  <div class="mt-4">
                    <a href="{{route('user.downPayment.checkout', $downPayment-
>id)}}"
                      class="btn btn-success mt-4">
                      <i class="fas fa-credit-card"></i> Bayar Sekarang
                    </a>
                  </div>
                  @endif
                </div>
                <!-- /.card-body -->
              </div>
              @endif
            </div>
            {{-- Permintaan Penjualan Mobil Accepted --}}
            <div class="col-12 mb-4">
              <div class="card ">
                <div class="card-header border-0">
                <h3 class="card-title">Permintaan Penjualan Mobil</h3>
                </div>
                <div class="card-body table-responsive p-0">
                <table class="table table-striped align-middle">
                  <thead>
                  <tr>
                    <th>ID</th>
                    <th>Mobil</th>
                    <th>Penawaran Harga</th>
                    <th>Tanggal Janjian</th>
                    <th>Detail</th>
                  </tr>
                  </thead>
                    <tbody>
                      @forelse ($offers as $offer)
                        <tr>
                          <td># {{$offer->id}}</td>
                          <td>
                            {{ $offer->brand }} {{ $offer->model }} {{ $offer->year }}
                            @switch($offer->status)
                              @case('pending')
                                <span class="badge bg-warning text-
dark">Pending</span>
                                @break
                              @case('accepted')
                                <span class="badge bg-success">Accepted</span>
                                @break
                              @case('rejected')
```

```
                                <span class="badge bg-danger">Rejected</span>
                                @break
                            @default
                                <span class="badge bg-secondary">{{ ucfirst($item-
>status) }}</span>
                            @endswitch
                        </td>
                        <td>Rp {{ number_format($offer->offered_price, 0, ',', '.')
}}</td>
                        <td>
                            {{ $offer->inspection_date ? \Carbon\Carbon::parse($offer-
>inspection_date)->format('d M Y') : '-' }}
                        </td>
                        <td>
                            <a href="{{ route('user.offer.show', $offer->id) }}" class="btn
btn-sm btn-primary">Detail</a>
                        </td>
                    </tr>
                @empty
                    <tr>
                        <td colspan="5" class="text-center">Data tidak tersedia.</td>
                    </tr>
                @endforelse
            </tbody>
        </table>
        </div>
    </div>
  </div>
</div>

  </div>
</div>
@endsection
```

resources/views/user/downPayment/checkout.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <div class="container-fluid">
    {{-- Alert Notifikasi Sukses --}}
    @if (session('success'))
    <div class="alert alert-success alert-dismissible fade show" role="alert">
      {{ session('success') }}
      <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
    </div>
    @endif
    {{-- Alert Notifikasi Error --}}
    @if (session('error'))
    <div class="alert alert-danger alert-dismissible fade show" role="alert">
```

```
        {{ session('error') }}
        <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
    </div>
    @endif
    <div class="card mb-4">
        <div class="card-body">
            <h3 class="fw-bold mb-4">Detail Pembayaran</h3>
            <table class="table table-bordered">
                <tr>
                    <th>ID Pembayaran</th>
                    <td>{{ '#'. $downPayments->id ?? '-'}}</td>
                </tr>
                <tr>
                    <th>Dibuat</th>
                    <td>{{ \Carbon\Carbon::parse($downPayments->created_at)->format('d
M Y') }}</td>
                </tr>
                <tr>
                    <th>Mobil</th>
                    <td class="text-uppercase">#{{ $downPayments->car_id }} {{
$downPayments->car->brand  . ' ' . $downPayments->car->model . ' ' . $downPayments-
>car->year . ' '. $downPayments->car->color ?? '-' }}</td>
                </tr>
                <tr>
                    <th>Nama Pengguna</th>
                    <td>{{ $downPayments->user->name ?? '-' }}</td>
                </tr>
                <tr>
                    <th>Email</th>
                    <td>{{ $downPayments->user->email ?? '-' }}</td>
                </tr>
                <tr>
                    <th>Nomor Telepon</th>
                    <td>{{ $downPayments->user->phone ?? '-' }}</td>
                </tr>

                <tr>
                    <th>Tanggal Janji</th>
                    <td>{{ \Carbon\Carbon::parse($downPayments->appointment_date)-
>format('d M Y') }}</td>
                </tr>
                <tr>
                    <th>Jam Janji</th>
                    <td>{{\Carbon\Carbon::parse($downPayments->appointment_date)-
>format('H:i') . ' WIB'  }}</td>
                </tr>
                <tr>
                    <th>Lokasi Showroom</th>
```

```
                <td>Lokasi: Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU
TAPOS, DEPOK<br>
                  Kontak Showroom: 081220745317</td>
              </tr>
              <tr>
                <th colspan="2" class="text-muted">Informasi Pembayaran</th>
              </tr>
               <tr>
                <th>Order ID</th>
                <td>{{ $downPayments->order_id ?? '-' }}</td>
              </tr>
              <tr>
                <th>Status Pembayaran DP</th>
                <td>
                  @switch($downPayments->payment_status)
                    @case('pending')
                      <span class="badge bg-warning text-dark">Pending</span>
                      @break
                    @case('confirmed')
                      @if(optional($downPayments->refund)->refund_status ===
'refund')
                        <span class="badge bg-success">Confirmed</span> | <span
class="badge bg-secondary">Refund</span>
                      @else
                        <span class="badge bg-success">Confirmed</span>
                      @endif
                      @break
                    @case('cancelled')
                      <span class="badge bg-danger">Cancelled</span>
                      @break
                    @case('expired')
                      <span class="badge bg-danger">Expired</span>
                      @break
                    @default
                      <span class="badge bg-secondary">{{ ucfirst($item-
>payment_status) }}</span>
                  @endswitch
                </td>
              </tr>
              <tr>
                <th>Metode Pembayaran</th>
                <td>{{ $downPayments->payment_method ?? 'Belum Dibayar'}}</td>
              </tr>
              <tr>
                <th>Jumlah Pembayaran</th>
                <td>Rp {{ number_format($downPayments->amount, 0, ',', '.') }}</td>
              </tr>
              <tr>
                <th>Tanggal Pembayaran</th>
```

```
                <td>{{ $downPayments->payment_date ?
\Carbon\Carbon::parse($downPayments->payment_date)->format('d M Y H:i') : '-'
}}</td>
            </tr>
            @if ($downPayments->payment_status == 'confirmed')


            @endif
            @if ($downPayments->refund_id !== null && $downPayments-
>payment_status === 'confirmed')
                <tr>
                    <th colspan="2" class="text-muted">Informasi Refund</th>
                </tr>
                <tr>
                    <th>No Rekening Refund</th>
                    <td>{{ $downPayments->refund ? $downPayments->refund-
>no_rekening_refund : '-' }}</td>
                </tr>
                <tr>
                    <th>Bukti Pembayaran Refund</th>
                    <td>
                        @if ($downPayments->refund && $downPayments->refund-
>refund_payment_proof)
                        <img src="{{ asset('storage/refund/' . $downPayments->refund-
>refund_payment_proof) }}"
                            alt="Bukti Pembayaran Refund" class="img-thumbnail"
style="max-height: 200px; object-fit: contain;">
                        @else
                        -
                        @endif
                    </td>
                </tr>
                @endif
            </table>
            {{-- Tombol bayar --}}
            @if($downPayments->payment_status == 'pending' && $downPayments->car-
>status == "available")
            <div class="mt-4">
                <button id="pay-button" class="btn btn-success mt-4">
                    <i class="fas fa-credit-card"></i> Bayar Sekarang
                </button>
            </div>
            @endif
        </div>
    </div>
  </div>
</div>
@push('scripts')
<script src="https://app.sandbox.midtrans.com/snap/snap.js" data-client-key="{{
config('midtrans.client_key') }}"></script>
```

```
<script type="text/javascript">
   document.getElementById('pay-button')?.addEventListener('click', function () {
      fetch(`/api/check-midtrans-status/{{ $downPayments->id }}`)
      .then(response => response.json())
            .then(data => {
               console.log("Cek status:", data.status);
               if (data.status == "under_review" || data.status == "sold" || data.status ==
"pending_check") {
                  alert("Mobil tidak tersedia")
                  window.location = "{{ route('user.downPayment.index') }}"
               }else if (data.status === 'confirmed' || data.status === 'cancelled') {
                  window.location.href = "{{ route('user.downPayment.changeStatus',
$downPayments->id) }}";
               }else if (data.status == "error") {
                  snap.pay(data.snap_token, {
                     onSuccess: function(result){
                        // alert("Pembayaran berhasil!");
                        window.location.href = "{{
route('user.downPayment.changeStatus' , $downPayments->id) }}";
                     },
                     onPending: function(result){
                        // alert("Menunggu pembayaran.");
                        window.location.href = "{{ route('user.downPayment.checkout' ,
$downPayments->id) }}";
                     },
                     onError: function(result){
                        // alert("Pembayaran gagal.");
                        window.location.href = "{{
route('user.downPayment.changeStatus' , $downPayments->id) }}";
                     },
                     onClose: function(){
                        alert("Kamu menutup popup tanpa menyelesaikan pembayaran.");
                     },
                  });
               }
            })
            .catch(error => {
               snap.pay('{{ $snapToken }}', {
                  onSuccess: function(result){
                     // alert("Pembayaran berhasil!");
                     window.location.href = "{{ route('user.downPayment.changeStatus' ,
$downPayments->id) }}";
                  },
                  onPending: function(result){
                     // alert("Menunggu pembayaran.");
                     window.location.href = "{{ route('user.downPayment.checkout' ,
$downPayments->id) }}";
                  },
                  onError: function(result){
                     // alert("Pembayaran gagal.");
```

```
                    window.location.href = "{{ route('user.downPayment.changeStatus' ,
$downPayments->id) }}";
                },
                onClose: function(){
                    alert("Kamu menutup popup tanpa menyelesaikan pembayaran.");
                },
            });
        });
    });
</script>
@endpush
@endsection
```

resources/views/user/downPayment/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
    <div class="container-fluid">
        <div class="card mb-4">
            <div class="card-body">

                {{-- Alert Notifikasi Sukses --}}
                @if (session('success'))
                <div class="alert alert-success alert-dismissible fade show" role="alert">
                    {{ session('success') }}
                    <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
                </div>
                @endif
                {{-- Alert Notifikasi Error --}}
                @if (session('error'))
                <div class="alert alert-danger alert-dismissible fade show" role="alert">
                    {{ session('error') }}
                    <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
                </div>
                @endif
                <div class="table-responsive">
                    <table class="table table-bordered">
                        <thead>
                            <tr>
                                <th style="width: 10px">ID</th>
                                <th>Dibuat</th>
                                <th>Mobil</th>
                                <th>Jumlah DP</th>
                                <th>Tanggal Janjian</th>
                                <th>Status</th>
                                <th>Action</th>
                            </tr>
                        </thead>
```

```
                <tbody>
                    @forelse ($downPayments as $item)
                    <tr>
                        <td>{{ $item->id }}</td>
                        <td>{{ \Carbon\Carbon::parse($item->created_at)-
>translatedFormat('d F Y H:i') }}</td>
                        <td class="text-capitalize">
                            {{ $item->car->brand . ' ' . $item->car->model . ' ' . $item->car-
>year }}
                        </td>
                        <td>{{ 'Rp ' . number_format($item->amount , 0, ',', '.') }}</td>
                        <td>{{ \Carbon\Carbon::parse($item->appointment_date)-
>translatedFormat('d F Y H:i') }}</td>
                        <td>
                            @if ($item->car->status === 'available' || $item->car->status ===
'under_review' || $item->car->status === 'sold' )
                                @switch($item->payment_status)
                                    @case('pending')
                                        <span class="badge bg-warning text-
dark">Pending</span>
                                        @break
                                    @case('confirmed')
                                        @if(optional($item->refund)->refund_status === 'refund')
                                            <span class="badge bg-success">Confirmed</span> |
<span class="badge bg-secondary">Refund</span>
                                        @else
                                            <span class="badge bg-success">Confirmed</span>
                                        @endif
                                        @break
                                    @case('cancelled')
                                        <span class="badge bg-danger">Cancelled</span>
                                        @break
                                    @case('expired')
                                        <span class="badge bg-danger">Expired</span>
                                        @break
                                    @default
                                        <span class="badge bg-secondary">{{ ucfirst($item-
>payment_status) }}</span>
                                @endswitch

                            @else
                                <span class="badge bg-secondary">Mobil Tidak
Tersedia</span>
                            @endif
                        </td>
                        <td>
                            @if ($item->payment_status == 'pending' && $item->car->status
=== 'available')
                            <a href="{{ route('user.downPayment.checkout' , $item->id)}}"
class="btn btn-sm btn-primary mb-1" data-id="{{ $item->id }}">
```

```
                        Bayar Sekarang
                    </a>
                    @else
                    <a href="{{ route('user.downPayment.show' , $item->id)}}"
class="btn btn-sm btn-outline-secondary">Detail</a>
                    @endif
                  </td>
                </tr>
                @empty
                <tr>
                  <td colspan="7" class="text-center">Data tidak tersedia.</td>
                </tr>
                @endforelse
              </tbody>
            </table>
          </div>
          {{-- Pagination --}}
          <div class="card-footer clearfix">
            <div class="pagination pagination-sm m-0 float-end">
              {{ $downPayments->appends(request()->query())-
>links('pagination::bootstrap-5') }}
            </div>
          </div>
        </div>
      </div>
    </div>
</div>
@endsection
@push('scripts')
<script src="https://app.sandbox.midtrans.com/snap/snap.js" data-client-key="{{
config('midtrans.client_key') }}"></script>
<script>
  document.addEventListener('DOMContentLoaded', function () {
    const payButtons = document.querySelectorAll('.pay-now-btn');
    payButtons.forEach(button => {
      button.addEventListener('click', function (e) {
        e.preventDefault();
        const dpId = this.dataset.id;
        fetch(`/user/payment/${dpId}/token`)
          .then(response => response.json())
          .then(data => {
            if (data.snapToken) {
              window.snap.pay(data.snapToken, {
                onSuccess: function (result) {
                  alert('Pembayaran berhasil!');
                  location.reload();
                },
                onPending: function (result) {
                  alert('Menunggu pembayaran!');
                },
```

```
                    onError: function (result) {
                        alert('Pembayaran gagal!');
                    },
                    onClose: function () {
                        alert('Kamu menutup popup tanpa menyelesaikan pembayaran');
                    }
                });
            } else {
                alert('Gagal mengambil token pembayaran');
            }
        })
        .catch(error => {
            console.error('Error:', error);
            alert('Terjadi kesalahan saat mengambil token pembayaran.');
        });
    });
  });
});
</script>
@endpush
```

resources/views/user/offer/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <div class="container-fluid">
    <div class="card mb-4">
      <div class="card-body">
        <div class="table-responsive">
          <table class="table table-bordered">
            <thead>
              <tr>
                <th style="width: 10px">ID</th>
                <th>Dibuat</th>
                <th>Mobil</th>
                <th>Penawaran Harga</th>
                <th>Tanggal Janjian</th>
                <th>Status</th>
                <th>Action</th>
              </tr>
            </thead>
            <tbody>
              @forelse ($offers as $item)
              <tr>
                <td>{{$item->id}}</td>
                <td>{{ \Carbon\Carbon::parse($item->created_at)-
>translatedFormat('d F Y H:i') }}</td>
                <td class="text-capitalize">
                    {{ $item->brand . ' ' . $item->model . ' ' . $item->year }}
                </td>
```

```
                    <td>{{ 'Rp ' . number_format($item->offered_price, 0, ',', '.') }}</td>
                    <td>{{ \Carbon\Carbon::parse($item->appointment_date)-
>translatedFormat('d F Y H:i T') }}</td>
                    <td>
                      @switch($item->status)
                        @case('pending')
                          <span class="badge bg-warning text-dark">Pending</span>
                          @break
                        @case('accepted')
                          <span class="badge bg-success">Accepted</span>
                          @break
                        @case('rejected')
                          <span class="badge bg-danger">Rejected</span>
                          @break
                        @default
                          <span class="badge bg-secondary">{{ ucfirst($item->status)
}}</span>
                      @endswitch
                    </td>
                    <td>
                      <a href="{{route('user.offer.show',$item->id)}}" class="btn btn-
sm btn-info">Detail</a>

                  </tr>
                  @empty
                  <tr>
                    <td colspan="7" class="text-center">Data tidak tersedia.</td>
                  </tr>
                  @endforelse
                </tbody>
              </table>
            </div>
          </div>
        </div>
      </div>
</div>
@endsection
```

resources/views/user/offer/show.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <div class="container-fluid">
      @if (session('success'))
        <div class="alert alert-success alert-dismissible fade show" role="alert">
          {{ session('success') }}
          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>
      @endif
```

```
<div class="card mb-4">
    <div class="card-body">
        <h3 class="fw-bold mb-4">Detail Penjualan Mobil {{ $offer->brand  . ' ' .
$offer->model . ' ' . $offer->year ?? '-' }} </h3>
        <table class="table table-bordered">
            <tr>
                <th>ID Penjualan</th>
                <td>{{ '#'. $offer->id ?? '-'}}</td>
            </tr>
            <tr>
                <th>Status Penjualan</th>
                <td>
                    <span class="badge
                        @if($offer->status == 'pending') bg-warning
                        @elseif($offer->status == 'accepted') bg-success
                        @elseif($offer->status == 'rejected') bg-danger
                        @endif">
                        {{ ucfirst($offer->status) }}
                    </span>
                </td>
            </tr>
            <tr>
                <th>Nama Pengguna</th>
                <td>{{ $offer->user->name ?? '-' }}</td>
            </tr>
            <tr>
                <th>Email</th>
                <td>{{ $offer->user->email ?? '-' }}</td>
            </tr>
            <tr>
                <th>Nomor Telepon</th>
                <td>{{ $offer->user->phone ?? '-' }}</td>
            </tr>
            <tr>
                <th>Mobil</th>
                <td>{{ $offer->brand  . ' ' . $offer->model . ' ' . $offer->year ?? '-' }}</td>
            </tr>
            <tr>
                <th>Tanggal Janji</th>
                <td>{{ \Carbon\Carbon::parse($offer->inspection_date)->format('d M Y')
}}</td>
            </tr>
            <tr>
                <th>Jam Janji</th>
                <td>{{ \Carbon\Carbon::parse($offer->inspection_date)->format('H:i T')
}}</td>
            </tr>
            <tr>
                <th>Penawaran Harga Mobil</th>
                <td>Rp {{ number_format($offer->offered_price, 0, ',', '.') }}</td>
```

```
                </tr>
                <tr>
                    <th>Lokasi Inspeksi</th>
                    <td>
                        @if($offer->status === 'accepted' && $offer->location_inspection ===
'Datang Ke Showroom')
                        Datang Ke Showroom<br>
                        Lokasi: Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU
TAPOS, DEPOK<br>
                        Kontak Admin: 081220745317
                        @elseif($offer->location_inspection === 'Rumah')
                        Rumah<br>
                        Lokasi: {{ Auth::user()->address ?? 'Alamat tidak tersedia' }}
                        <br>
                        <button class="btn btn-sm btn-info mt-2" data-bs-toggle="modal" data-
bs-target="#alamatModal">
                            Lengkapi Alamat
                        </button>
                        @else
                        {{ $offer->location_inspection }}
                        @endif
                    </td>
                </tr>
                <tr>
                    @if ($offer->note != null)
                        <th>Catatan Showroom</th>
                        <td>{{$offer->note}}</td>
                    @endif
                </tr>
            </table>
        </div>
    </div>
  </div>
</div>
<!-- Modal Lengkapi Alamat -->
<div class="modal fade" id="alamatModal" tabindex="-1" aria-
labelledby="alamatModalLabel" aria-hidden="true">
 <div class="modal-dialog">
  <form method="POST" action="{{ route('user.updateAdress', $offer->user->id) }}">
    @csrf
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="alamatModalLabel">Lengkapi Alamat
Anda</h5><small> (bisa tambahkan sharelok)</small>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-
label="Tutup"></button>
      </div>
      <div class="modal-body">
        <div class="mb-3">
```

```
                <label for="alamat" class="form-label">Alamat Lengkap <small> (bisa
tambahkan sharelok)</small></label>
                <textarea name="address" class="form-control" rows="3" required>{{
old('address', Auth::user()->address) }}</textarea>
            </div>
        </div>
        <div class="modal-footer">
            <button type="submit" class="btn btn-success">Simpan</button>
            <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">Batal</button>
        </div>
    </div>
  </form>
 </div>
</div>
@endsection
```

resources/views/user/transactionOfferRecord/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
  <div class="container-fluid">
    <div class="card mb-4">
      <div class="card-body">
        {{-- Alert Notifikasi Sukses --}}
        @if (session('success'))
        <div class="alert alert-success alert-dismissible fade show" role="alert">
          {{ session('success') }}
          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>
        @endif
        @if (session('error'))
        <div class="alert alert-danger alert-dismissible fade show" role="alert">
          {{ session('error') }}
          <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>
        @endif
        {{-- Alert Notifikasi Gagal --}}
        {{-- Tabel mobil dengan fitur DataTables --}}
        <div class="table-responsive">
          <table id="cars-table" class="table table-bordered">
            <thead>
              <tr>
                <th>ID</th>
                <th>Dibuat</th>
                <th>Mobil</th>
                <th>Pembeli</th>
                <th>Harga</th>
```

```
                    <th>Tanggal Terjual</th>
                    <th>Status</th>
                  </tr>
                </thead>
                <tbody>
                  @forelse ($offerRecords as $offerRecord)
                  <tr>
                    <td>{{ $offerRecord->id }}</td>
                    <td>{{ \Carbon\Carbon::parse($offerRecord->created_at)-
>translatedFormat('d F Y H:i') }}
                    </td>
                    <td>{{ $offerRecord->offer->brand . ' ' . $offerRecord->offer-
>model . ' ' . $offerRecord->offer->year }}</a>
                    </td>
                    <td>{{ $offerRecord->buyerOfferRecord->name }}</td>
                    <td>Rp. {{ number_format($offerRecord->sale_price, 0, ',', '.')
}}</td>
                    <td>{{ \Carbon\Carbon::parse($offerRecord->sale_date)-
>translatedFormat('d F Y H:i') }}
                    </td>
                    <td>
                      @if ($offerRecord->status == 'pending')
                      <span class="badge bg-warning">Pending</span>
                      @elseif ($offerRecord->status == 'completed')
                      <span class="badge bg-success">Completed</span>
                      @elseif ($offerRecord->status == 'cancelled')
                      <span class="badge bg-danger">Cancelled</span>
                      @else
                      <span class="badge bg-secondary">Unknown</span>
                      @endif
                    </td>
                  </tr>
                  @empty
                  <tr>
                    <td colspan="8" class="text-center">Data Tidak Tersedia.</td>
                  </tr>
                  @endforelse
                </tbody>
              </table>
            </div>
            {{-- Pagination --}}
            <div class="card-footer clearfix">
              <div class="pagination pagination-sm m-0 float-end">
                {{ $offerRecords->appends(request()->query())-
>links('pagination::bootstrap-5') }}
              </div>
            </div>
          </div>
        </div>
      </div>
```

```
</div>
@endsection
```

resources/views/user/transactionSalesRecord/index.blade.php

```
@extends('layouts.master')
@section('content')
<div class="app-content">
   <div class="container-fluid">
      <div class="card mb-4">
         <div class="card-body">
            {{-- Alert Notifikasi Sukses --}}
            @if (session('success'))
            <div class="alert alert-success alert-dismissible fade show" role="alert">
               {{ session('success') }}
               <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
            </div>
            @endif
            @if (session('error'))
            <div class="alert alert-danger alert-dismissible fade show" role="alert">
               {{ session('error') }}
               <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
            </div>
            @endif
            {{-- Alert Notifikasi Gagal --}}
            {{-- Tabel mobil dengan fitur DataTables --}}
            <div class="table-responsive">
               <table id="cars-table" class="table table-bordered">
                  <thead>
                     <tr>
                        <th>ID</th>
                        <th>Dibuat</th>
                        <th>Mobil</th>
                        <th>Penjual</th>
                        <th>Harga</th>
                        <th>Tanggal Dibeli</th>
                        <th>Status</th>
                        <th>Aksi</th>
                     </tr>
                  </thead>
                  <tbody>
                     @forelse ($salesRecords as $salesRecord)
                     <tr>
                        <td>{{ $salesRecord->id }}</td>
                        <td>{{ \Carbon\Carbon::parse($salesRecord->created_at)-
>translatedFormat('d F Y H:i') }}
                        </td>
                        <td>{{ $salesRecord->car->brand . ' ' . $salesRecord->car->model . ' '
. $salesRecord->car->year }}</td>
```

```
                        <td>{{ $salesRecord->saler->name }}</td>
                        <td>Rp. {{ number_format($salesRecord->sale_price, 0, ',', '.')
}}</td>
                        <td>{{ \Carbon\Carbon::parse($salesRecord->sale_date)-
>translatedFormat('d F Y H:i') }}
                        </td>
                        <td>
                           @if ($salesRecord->status == 'pending')
                           <span class="badge bg-warning">Pending</span>
                           @elseif ($salesRecord->status == 'completed')
                           <span class="badge bg-success">Completed</span>
                           @elseif ($salesRecord->status == 'cancelled')
                           <span class="badge bg-danger">Cancelled</span>
                           @else
                           <span class="badge bg-secondary">Unknown</span>
                           @endif
                        </td>
                           @php
                              $existingReview =
App\Models\Review::where('sales_record_id', $salesRecord->id)
                                 ->where('car_id', $salesRecord->car->id)
                                 ->first();
                           @endphp
                        <td>
                           @if ($salesRecord->status == 'completed' && $existingReview ==
false)
                           <a href="{{ route('user.transactionSalesRecord.createTesti',
$salesRecord->id) }}"
                              class="btn btn-primary btn-sm">Testimoni</a>
                           @endif
                        </td>
                     </tr>
                     @empty
                     <tr>
                        <td colspan="9" class="text-center">Data Tidak Tersedia</td>
                     </tr>
                     @endforelse
                  </tbody>
               </table>
            </div>
            {{-- Pagination --}}
            <div class="card-footer clearfix">
               <div class="pagination pagination-sm m-0 float-end">
                  {{$salesRecords->appends(request()->query())-
>links('pagination::bootstrap-5') }}
               </div>
            </div>
         </div>
      </div>
   </div>
```

```
</div>
@endsection
```

resources/views/user/transactionSalesRecord/testimoni.blade.php

```
@extends('layouts.master')
@section('content')
@push('styles')
   <style>
      .star-rating {
         display: flex;
         flex-direction: row-reverse;
         justify-content: flex-end;
      }
      .star-rating input {
         display: none;
      }
      .star-rating label {
         font-size: 2rem;
         color: #ddd;
         cursor: pointer;
         transition: color 0.2s;
      }
      .star-rating input:checked ~ label,
      .star-rating label:hover,
      .star-rating label:hover ~ label {
         color: #ffc107;
      }
   </style>
@endpush
<div class="app-content">
   <!--begin::Container-->
   <div class="container-fluid">
      <div class="card card-primary card-outline mb-4">
         <!--begin::Header-->
         <div class="card-header">
            <div class="card-title">Testimoni Car</div>
         </div>
         <!--end::Header-->
         <!--begin::Error Alert-->
         @if ($errors->any())
         <div class="alert alert-danger">
            <ul class="mb-0">
               @foreach ($errors->all() as $error)
               <li>{{ $error }}</li>
               @endforeach
            </ul>
         </div>
         @endif
         <!--end::Error Alert-->
         <!--begin::Form-->
```

```
        <form action="{{ route('user.transactionSalesRecord.storeTesti', $salesRecord-
>id) }}"
            method="POST" enctype="multipart/form-data">
            @csrf
            <div class="card-body">
                {{-- Comment --}}
                <div class="mb-3">
                    <label for="comment" class="form-label">Komentar</label>
                    <textarea class="form-control @error('comment') is-invalid @enderror"
id="comment"
                        name="comment" rows="3" required>{{ old('comment') }}</textarea>
                    @error('comment')
                    <span class="invalid-feedback" role="alert"><strong>{{ $message
}}</strong></span>
                    @enderror
                </div>
                {{-- Rating --}}
                <div class="mb-3">
                    <label class="form-label">Rating (1 - 5)</label>
                    <div class="star-rating">
                        @for ($i = 5; $i >= 1; $i--)
                            <input
                                type="radio"
                                name="rating"
                                id="rating-{{ $i }}"
                                value="{{ $i }}"
                                {{ old('rating') == $i ? 'checked' : '' }}
                                required
                            >
                            <label for="rating-{{ $i }}">
                                ★
                            </label>
                        @endfor
                    </div>
                    @error('rating')
                    <span class="invalid-feedback d-block" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                </div>
                {{-- Photo Review --}}
                <div class="mb-3">
                    <label for="photo_review" class="form-label">Foto Review</label>
                    <div class="mb-2">
                        <img id="preview-photo" src="#" alt="Preview Foto" style="max-
height: 200px; object-fit: contain;"
                            class="img-thumbnail d-none">
                    </div>
                    <input type="file" class="form-control @error('photo_review') is-invalid
@enderror"
```

```
                    id="photo_review" name="photo_review"
onchange="previewImage(event, 'preview-photo')" required>
                @error('photo_review')
                <span class="invalid-feedback" role="alert"><strong>{{ $message
}}</strong></span>
                @enderror
              </div>
            </div>
            <div class="card-footer">
              <button type="submit" class="btn btn-primary">Submit</button>
            </div>
          </form>
          <!--end::Form-->
        </div>
    </div>
    <!--end::Container-->
</div>
@push('scripts')
<script>
function previewImage(event, previewId) {
    const input = event.target;
    const reader = new FileReader();
    reader.onload = function(){
        const imgElement = document.getElementById(previewId);
        imgElement.src = reader.result;
        imgElement.classList.remove('d-none');
    }
    reader.readAsDataURL(input.files[0]);
}
</script>
@endpush
@endsection
```

resources/views/web/auth/login.blade.php

```
<!-- Modal Login -->
 <div class="modal fade" id="loginModal" tabindex="-1" aria-
labelledby="loginModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content p-4">
            <div class="modal-header border-0">
                <h5 class="modal-title w-100 text-center"
id="loginModalLabel">Masuk</h5>
            </div>
            <div class="modal-body">
              <form method="POST" action="{{ route('login') }}">
              @csrf
                  <div class="mb-3">
                      <label for="email" class="form-label fw-medium">Email*</label>
```

```
                    <input type="email" class="form-control py-2 border-0 @error('email')
is-invalid @enderror" id="email" placeholder="Masukkan Email" name="email"
value="{{ old('email') }}" required autocomplete="email" autofocus>
                  @error('email')
                    <span class="invalid-feedback" role="alert">
                      <strong>{{ $message }}</strong>
                    </span>
                  @enderror
                </div>
                <div class="mb-3">
                    <label for="password" class="form-label fw-medium">Kata
Sandi*</label>
                    <input type="password" class="form-control py-2 border-0
@error('password') is-invalid @enderror" id="password"
                      placeholder="Masukkan Kata Sandi" name="password" required
autocomplete="current-password">
                </div>
                <div class="mb-3 form-check">
                    <input class="form-check-input" type="checkbox" name="remember"
id="remember" {{ old('remember') ? 'checked' : '' }}>
                    <label class="form-check-label " for="remember">Ingat Saya</label>
                </div>
                <button type="submit" class="btn btn-login w-100 mb-2">Masuk</button>

                <div class="text-center register-link">Belum Mempunyai Akun? <a
href="{{ route('register') }}">Daftar</a></div>
              </form>
            </div>
        </div>
    </div>
 </div>
```

resources/views/web/auth/register.blade.php

```
<!DOCTYPE html>
<html lang="id">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1">
   <title>Form Daftar YONOMOBILINDO</title>
   {{-- icon --}}
   <link rel="icon" type="image/png" href="{{ asset('image/YONO-MOBILINDO-
LOGO.svg') }}">
   <!-- font montserat -->
   <link rel="preconnect" href="https://fonts.googleapis.com" />
   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
   <link
href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..
900&display=swap"
       rel="stylesheet" />
```

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
<style>
    :root {
        --background: #fffdf6;
        --black: #171513;
        --grey: #867e77;
        --primary: #27548a;
        --yellow: #fcf259;
    }
    * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
        font-family: "Montserrat", sans-serif;
        color: var(--black);
    }
    body {
        background-color: var(--background);
        background-color: #f8f9fa;
        /* height: 100vh; */
        padding: 20px 0;
        display: flex;
        align-items: center;
        justify-content: center;
    }
    .register-box {
        background-color: white;
        border: 1px solid #ccc;
        padding: 2.5rem;
        width: 100%;
        max-width: 500px;
        box-shadow: 0 0 10px rgba(0, 0, 0, 0.05);
        /* margin: 50px 0; */
    }
    .form-control {
        background-color: #f1f1f1;
        border: none;
    }
    .btn-daftar {
        background-color: var(--primary);
        color: var(--yellow);
        font-weight: bold;
        border: none;
        transition: background-color 0.3s ease, color 0.3s ease, transform 0.2s ease;
    }
    .btn-daftar:hover {
        /* Warna lebih gelap dari primary */
        background-color: #1f3f6b;
        /* Warna teks putih */
```

```
            color: #fff;
            /* Efek bayangan halus */
            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.2);
        }
    </style>
</head>
<body>
    <div class="register-box text-center rounded">
        <a href="{{route('home')}}">
            <img src="{{asset('image/Logo-Mobilindo-2.svg')}}" alt="YONOMOBILINDO
LOGO"
                class="img-fluid mx-auto d-block mb-3" style="max-width: 200px;">
        </a>
        <h3 class="form-heading mb-1">Daftar</h3>
        <p class="text-muted mb-4">Segera mendaftar dan dapatkan keuntungan</p>
        <form method="POST" action="{{route('register')}}">
            @csrf
            <div class="mb-3 text-start">
                <label class="form-label fw-medium" for="name">Nama Lengkap*</label>
                <input type="text" id="name" class="form-control @error('name') is-invalid
@enderror" name="name" placeholder="Masukkan Nama Lengkap"
value="{{old('name')}}" required>
                @error('name')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                @enderror
            </div>
            <div class="mb-3 text-start">
                <label class="form-label fw-medium" for="email">Email*</label>
                <input type="email" id="email" class="form-control @error('email') is-invalid
@enderror" name="email" placeholder="Masukkan Email"  value="{{old('email')}}"
required>
                @error('email')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                @enderror
            </div>
            <div class="mb-3 text-start">
                <label class="form-label fw-medium" for="password">Kata Sandi*</label>
                <input type="password" class="form-control @error('password') is-invalid
@enderror" name="password" placeholder="Masukkan Kata Sandi" autocomplete="new-
password" required>
                @error('password')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                @enderror
            </div>
```

```
        <div class="mb-3 text-start">
            <label class="form-label fw-medium" for="konfirmasi-katasandi">Konfirmasi
Kata Sandi*</label>
            <input type="password" id="konfirmasi-katasandi" class="form-control"
name="password_confirmation" placeholder="Masukkan Konfirmasi Kata Sandi"
required autocomplete="new-password" required>
        </div>
        <div class="mb-4 text-start">
            <label class="form-label fw-medium" for="phone">No. Telpon*</label>
            <input type="number" id="phone" class="form-control @error('phone') is-
invalid @enderror" value="{{ old('phone') }}" name="phone" placeholder="Masukkan
No. Telpon" required>
        </div>
        <button type="submit" class="btn btn-daftar w-100">Daftar</button>
        <div class="text-center register-link mt-2">Sudah Mempunyai Akun? <a
href="{{ route('login') }}">Login</a></div>
    </form>
  </div>
  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></scri
pt>
</body>
</html>
```

resources/views/web/layouts/css.blade.php

```
<!-- font montserat -->
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
<link
href="https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..
900&display=swap"
  rel="stylesheet" />
<!-- boostrap -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
<!-- cdn font awesome -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css">
{{-- css --}}
<style>
  :root {
    --background: #fffdf6;
    --black: #171513;
    --grey: #867e77;
    --primary: #27548a;
    --yellow: #fcf259;
  }
  * {
    margin: 0;
    padding: 0;
```

```
    box-sizing: border-box;
    font-family: "Montserrat", sans-serif;
    color: var(--black);
}
body {
    background-color: var(--background);
}
/* Navbar styles */
#navbar {
    position: fixed;
    width: 100%;
    z-index: 1000;
    border-bottom: 1px solid black;
}
#navbar .navbar-nav .nav-link {
    font-weight: 700;
    color: var(--grey) !important;
    font-size: 16px;
    border-bottom: 2px solid transparent;
    transition: color 0.3s, border-bottom 0.3s;
}
#navbar .navbar-nav .nav-link.active,
#navbar .navbar-nav .nav-link:hover {
    color: var(--primary) !important;
    border-bottom: 2px solid var(--primary);
}
.btn-login {
    background-color: var(--primary);
    color: var(--yellow);
    font-weight: bold;
    border: none;
    transition: background-color 0.3s ease, color 0.3s ease, transform 0.2s ease;
}
.btn-login:hover {
    /* Warna lebih gelap dari primary */
    background-color: #1f3f6b;
    /* Warna teks putih */
    color: #fff;
    /* Efek bayangan halus */
    box-shadow: 0 4px 12px rgba(0, 0, 0, 0.2);
}
#navbar .logo-text {
    font-style: italic;
    font-weight: 700;
    color: var(--primary);
}
#navbar .logo-text span {
    color: var(--yellow);
}
@media (min-width: 992px) {
```

```
    #navbar .navbar-nav .nav-item+.nav-item {
        margin-left: 30px;
    }
    #navbar .btn-login {
        margin-left: 30px;
    }
}
/* Custom hamburger icon for toggler */
.navbar-toggler {
    border: none;
    background: transparent;
    padding: 4px;
    width: 40px;
    height: 32px;
    display: flex;
    flex-direction: column;
    justify-content: space-around;
    align-items: center;
    cursor: pointer;
    box-shadow: none;
    transition: all 0.3s ease;
}
.navbar-toggler:focus {
    outline: none;
    box-shadow: none;
}
.navbar-toggler:hover {
    background-color: rgba(39, 84, 138, 0.1);
    border-radius: 4px;
}
.navbar-toggler span {
    display: block;
    width: 30px;
    height: 3px;
    background-color: var(--primary);
    border-radius: 2px;
    transition: all 0.3s ease-in-out;
    position: relative;
    transform-origin: center;
}
/* Hamburger animation - collapsed state (☰) */
.navbar-toggler.collapsed span:nth-child(1) {
    transform: rotate(0) translate(0, 0);
}
.navbar-toggler.collapsed span:nth-child(2) {
    opacity: 1;
    transform: scale(1);
}
.navbar-toggler.collapsed span:nth-child(3) {
    transform: rotate(0) translate(0, 0);
```

```
    }
    /* Hamburger animation - expanded state (✕) */
    .navbar-toggler:not(.collapsed) span:nth-child(1) {
        transform: rotate(45deg) translate(5px, 5px);
    }
    .navbar-toggler:not(.collapsed) span:nth-child(2) {
        opacity: 0;
        transform: scale(0);
    }
    .navbar-toggler:not(.collapsed) span:nth-child(3) {
        transform: rotate(-45deg) translate(5px, -5px);
    }
    .modal .form-control {
        background-color: #eeeeee;
        border-radius: 5px;
    }

    #footer {
        background-color: #032145;
    }
    .custom-link {
        color: rgba(255, 255, 255, 0.5);
        text-decoration: none;
        transition: color 0.2s ease-in-out;
    }
    .custom-link:hover {
        color: #ffffff; /* Putih penuh saat hover */
        text-decoration: none;
    }
    /* #reqCar{
      background-color: #ffffff;
    } */
</style>
```

resources/views/web/layouts/footer.blade.php

```
<section id="footer">
    <footer class="py-5 text-white">
        <div class="container">
            <div class="row align-items-start">
                <!-- Logo -->
                <div class="col-md-3 mb-4 mb-md-0">
                    <img src="{{ asset('image/YONO MOBILINDO LOGO 2.png')}}"
alt="Yono Mobilindo Logo"
                        class="img-fluid" style="max-width: 180px;">
                </div>
                <!-- Info Lokasi & Kontak -->
                <div class="col-md-9 mb-4 mb-md-0">
                    <h3 class="fw-bold text-white">Yono Mobilindo Showroom
Indonesia</h3>
                    <div class="row">
```

```
                <div class="col-md-8">
                  <div class="d-flex align-items-start mt-4">
                    <div class="me-3">
                      <span
                        class="d-inline-flex justify-content-center align-items-center
bg-warning rounded-circle"
                        style="width: 40px; height: 40px;">
                        <i class="fas fa-map-marker-alt text-primary"></i>
                      </span>
                    </div>
                    <div>
                      <p class="mb-1 fw-semibold text-white">Lokasi</p>
                      <p class="mb-0 text-white-50">Pekapuran Jl 1000 RT 06/05 NO
200 SUKAMAJU BARU
                        TAPOS, DEPOK</p>
                    </div>
                  </div>
                  <div class="d-flex align-items-start mt-3">
                    <div class="me-3">
                      <span
                        class="d-inline-flex justify-content-center align-items-center
bg-warning rounded-circle"
                        style="width: 40px; height: 40px;">
                        <i class="fas fa-user text-primary"></i>
                      </span>
                    </div>
                    <div>
                      <p class="mb-1 fw-semibold text-white">Kontak</p>
                      <p class="mb-0 text-white-50">081220745317</p>
                    </div>
                  </div>
                </div>
                <div class="col-md-4 mt-4">
                  <h6 class="fw-semibold text-white">Lainnya</h6>
                  <ul class="list-unstyled mt-3">
                    <li class="mb-2 text-white-50">
                      <a href="{{ route('home') }}" class="custom-link">
                        <i class="fas fa-car-side me-2 text-white-50"></i> Beli Mobil
                      </a>
                    </li>
                    <li class="mb-2 text-white-50">
                      <a href="{{ route('web.jualMobil.index') }}" class="custom-link">
                        <i class="fas fa-car me-2 text-white-50"></i> Jual Mobil
                      </a>
                    </li>
                    <li class="mb-2 text-white-50">
                      <a href="{{ route('web.testimonial') }}" class="custom-link">
                        <i class="fas fa-comments me-2 text-white-50"></i> Testimoni
                      </a>
                    </li>
```

```
                    <li class="text-white-50">
                        <a href="{{ route('web.kontak') }}" class="custom-link">
                            <i class="fas fa-phone me-2 text-white-50"></i> Kontak
                        </a>
                    </li>
                </ul>
            </div>
        </div>
    </div>
</div>
</div>
</footer>
</section>
```

resources/views/web/layouts/js.blade.php

```
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></scri
pt>
```

resources/views/web/layouts/master.blade.php

```
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <title>Yono Mobilindo</title>
    <meta name="title" content="Yono Mobilindo" />
    {{-- icon --}}
    <link rel="icon" type="image/png" href="{{ asset('image/YONO-MOBILINDO-
LOGO.svg') }}">
    @include('web.layouts.css')
    @stack('web-styles')
</head>
<body>
    {{-- navbar --}}
    @include('web.layouts.navbar')
    {{-- modal Login --}}
    @include('web.auth.login')
    {{-- content --}}
    @yield('web-content')
    <!-- start footer -->
   @include('web.layouts.footer')
    <!-- end footer -->
    {{-- js --}}
    @include('web.layouts.js')
    @stack('web-scripts')
</body>
</html>
```

resources/views/web/layouts/navbar.blade.php

```
<section id="navbar">
    <nav class="navbar navbar-expand-lg bg-white shadow-sm py-3 ">
        <div class="container">
            <!-- Logo -->
            <a class="navbar-brand d-flex align-items-center" href="{{route('home')}}">
                <img src="{{asset('image/Logo-Mobilindo-2.svg')}}" alt="Logo" width="150"
class="me-2" />
            </a>
            <!-- Custom toggle button -->
            <button class="navbar-toggler collapsed" type="button" data-bs-
toggle="collapse"
                data-bs-target="#navbarNav" aria-controls="navbarNav" aria-
expanded="false"
                aria-label="Toggle navigation">
                <span></span>
                <span></span>
                <span></span>
            </button>

            <!-- Menu -->
            <div class="collapse navbar-collapse justify-content-end" id="navbarNav">
                <ul class="navbar-nav mb-2 mb-lg-0">
                    <li class="nav-item">
                        <a class="nav-link {{ request()->routeIs('home') || request()-
>routeIs('web.detailMobil') || request()->routeIs('web.downPayment.show') ? 'active' : ''
}}" href="{{route('home')}}">BELI MOBIL</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link {{ request()->routeIs('web.jualMobil.*') ? 'active' : ''
}}" href="{{ route('web.jualMobil.index')}}">JUAL MOBIL</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link {{request()->routeIs('web.testimonial') ? 'active' : ''}}"
href="{{route('web.testimonial')}}">TESTIMONIAL</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link {{request()->routeIs('web.kontak') ? 'active' : ''}}"
href="{{route('web.kontak')}}">KONTAK</a>
                    </li>
                </ul>
                {{-- Jika belum login --}}
                @guest
                <a href="{{ route('login', ['redirect' => request()->fullUrl()]) }}" type="button"
class="btn btn-login px-5" data-bs-toggle="modal" data-bs-target="#loginModal">
                    MASUK
                </a>
                @endguest
                {{-- Jika sudah login --}}
                @auth
                <a href="{{ route('home-cms') }}" class="btn btn-login px-4">
```

```
                DASHBOARD
            </a>
            @endauth
        </div>
    </div>
  </nav>
</section>
```

resources/views/web/detailMobil.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
<style>
  #detail-mobil {
    padding-top: 120px;
  }
  /* Buat wrapper untuk kontrol ukuran tetap */
  .main-img-wrapper {
    width: 100%;
    height: 300px;
    /* Ukuran konsisten */
    overflow: hidden;
    border-radius: 8px;
  }
  .main-img {
    width: 100%;
    height: 100%;
    object-fit: contain;
    /* Gambar tidak dipotong */
    object-position: center;
  }
  .thumb-img {
    width: 80px;
    cursor: pointer;
    border: 2px solid transparent;
    transition: border 0.2s;
  }
  .thumb-img.active,
  .thumb-img:hover {
    border-color: #27548a;
  }
  .thumbnail-scroll {
    overflow-x: auto;
    white-space: nowrap;
    scrollbar-width: none;
    -ms-overflow-style: none;
  }
  .thumbnail-scroll::-webkit-scrollbar {
    display: none;
  }
```

```
.scroll-button {
   width: 35px;
   height: 100%;
   background: rgba(255, 255, 255, 0.8);
   border: none;
   position: absolute;
   top: 0;
   z-index: 10;
}
.scroll-left {
   left: 0;
}
.scroll-right {
   right: 0;
}
#tutorial .border {
   background-color: var(--background);
}
#tutorial .number {
   font-size: 48px;
   color: var(--grey, #867e77);
   font-weight: 700;
}
#tutorial .step-text {
   font-size: 16px;
   color: var(--black);
   font-weight: 700;
}
#tutorial img.step-icon {
   width: 70px;
}
@media (max-width: 576px) {
   #tutorial .number {
      font-size: 20px;
   }
   #tutorial .step-text {
      font-size: 12px;
      color: var(--black);
      font-weight: 700;
   }
   #tutorial img.step-icon {
      width: 40px;
   }
   #tutorial .step-text {
      font-size: 10px;
   }
   #tutorial .step-header {
      padding-left: 0.5rem !important;
      padding-right: 0.5rem !important;
   }
```

```
        #tutorial .step-body {
            padding: 1rem 0.5rem;
        }
    }
</style>
@endpush
<!-- start detail Mobil -->
<section id="detail-mobil">
    <div class="container py-5">
        <div class="row g-4">
            <!-- Gambar Mobil -->
            <div class="col-lg-6">
                <!-- Wrapper gambar utama -->
                <div class="main-img-wrapper mb-3">
                    <img id="mainImage" src="{{ $car->mainPhoto && Storage::disk('public')-
>exists('car_photos/' . $car->mainPhoto->photo_url)
                    ? asset('storage/car_photos/' . $car->mainPhoto->photo_url)
                    : asset('image/NoImage.png') }}" alt="{{ $car->brand }} {{ $car->model }}"
class="main-img"
                        data-bs-toggle="modal" data-bs-target="#zoomModal" alt="Mobil"
style="cursor: zoom-in;">
                </div>
                <!-- Modal Zoom -->
                <div class="modal fade" id="zoomModal" tabindex="-1" aria-
labelledby="zoomModalLabel"
                    aria-hidden="true">
                    <div class="modal-dialog modal-dialog-centered modal-xl">
                        <div class="modal-content bg-transparent border-0">
                            <div class="modal-body text-center p-0">
                                <img id="zoomImage" src="{{ $car->mainPhoto &&
Storage::disk('public')->exists('car_photos/' . $car->mainPhoto->photo_url)
                    ? asset('storage/car_photos/' . $car->mainPhoto->photo_url)
                    : asset('image/NoImage.png') }}" alt="{{ $car->brand }} {{ $car->model }}"
class="img-fluid w-100"
                                    alt="Zoom Preview">
                            </div>
                        </div>
                    </div>
                </div>
                <!-- Thumbnail scroll area -->
                <div class="position-relative">
                    <!-- Scroll Buttons -->
                    <button type="button" class="scroll-button scroll-left"
onclick="scrollThumbnails(-1)">
                        <i class="fas fa-chevron-left"></i>
                    </button>
                    <button type="button" class="scroll-button scroll-right"
onclick="scrollThumbnails(1)">
                        <i class="fas fa-chevron-right"></i>
                    </button>
```

```
                <!-- Thumbnails -->
                <div id="thumbnailContainer" class="thumbnail-scroll d-flex gap-2 px-5">
                    <!-- Thumbnails: Tambahkan lebih dari 5 untuk test -->
                    @if ($car->carPhoto->count())
                    @foreach ($car->carPhoto->sortBy('number') as $photo)
                    <img src="{{ asset('storage/car_photos/' . $photo->photo_url) }}"
class="thumb-img rounded"
                        onclick="switchImage(this)">
                    @endforeach
                    @endif
                </div>
            </div>
        </div>
        <!-- Detail Mobil -->
        <div class="col-lg-6">
            <h3 class="fw-bold text-capitalize">{{$car->year}} - {{$car->brand}} {{$car-
>model}} {{$car->color}}
            </h3>
            <p class="text-capitalize">{{ number_format($car->mileage, 0, ',', '.') }} KM |
{{ $car->transmission }}
            </p>
            <h5 class="text-danger fw-bold">Rp {{ number_format($car->price, 0, ',', '.')
}}</h5>

            <div class="d-flex justify-content-between border-bottom border-1 border-dark
py-2">
                <strong>Warna</strong>
                <span class="text-capitalize">{{$car->color}}</span>
            </div>
            <div class="d-flex justify-content-between border-bottom border-1 border-dark
py-2">
                <strong>Jenis Bahan Bakar</strong>
                <span class="text-capitalize">{{$car->fuel_type}}</span>
            </div>
            <div class="py-2">
                <strong class="fw-bold mb-1 mt-3">Lokasi Mobil:</strong>
                <p class="text-muted">Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU
BARU TAPOS DEPOK</p>
            </div>
            <!-- Tombol -->
            <div class="d-flex align-items-center gap-3 mt-3">
                @auth
                <a href="{{ route('web.downPayment.show' ,$car->id) }}" class="btn btn-
primary fw-semibold px-5 py-2"
                    style="background-color: var(--primary);">
                    <span style="color: var(--yellow);">DP Sekarang</span><br>
                    <small class="fw-normal text-white" style="font-size: 12px;">Min: Rp
500.000</small>
                </a>
```

```
                @else
                <a href="{{ route('login', ['redirect' => request()->fullUrl()]) }}" class="btn
btn-outline-primary fw-semibold px-5 py-2" >
                    <span>Login untuk DP</span><br>
                    <small class="fw-normal text-muted" style="font-size: 12px;">Akses fitur
hanya untuk pengguna
                        terdaftar</small>
                </a>
                @endauth
                <button class="btn btn-outline-secondary rounded-circle" style="width:
45px; height: 45px;"
                    onclick="shareLink()" title="Bagikan">
                    <i class="fas fa-share-alt"></i>
                </button>
            </div>
        </div>
    </div>
  </div>
</section>
<!-- end detail mobil -->
<!-- start tutorial -->
<section id="tutorial" class="py-5">
  <div class="container text-center bg-white py-5">
    <h3 class="fw-bold mb-5">PROSES BELI MOBIL</h3>
    <div class="row g-3 justify-content-center">
      <!-- STEP 1 -->
      <div class="col-4 col-lg-3">
        <div class="border rounded-3 h-100">
          <div
            class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
            <span class="step-text">STEP</span>
            <span class="number">01</span>
          </div>
          <div class="step-body py-4">
            <img src="Isi Data Mobil.svg" alt="DP" class="step-icon mb-3">
            <p class="fw-medium step-text mb-0">Melakukan DP</p>
          </div>
        </div>
      </div>
      <!-- STEP 2 -->
      <div class="col-4 col-lg-3">
        <div class="border rounded-3 h-100">
          <div
            class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
            <span class="step-text">STEP</span>
            <span class="number">02</span>
          </div>
          <div class="step-body py-4">
```

```
                <img src="Frame 3 1.svg" alt="Test Drive" class="step-icon mb-3">
                <p class="fw-medium step-text mb-0">Test Drive</p>
              </div>
            </div>
          </div>
          <!-- STEP 3 -->
          <div class="col-4 col-lg-3">
            <div class="border rounded-3 h-100">
              <div
                class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
                <span class="step-text">STEP</span>
                <span class="number">03</span>
              </div>
              <div class="step-body py-4">
                <img src="Frame 4 1.svg" alt="Pembayaran" class="step-icon mb-3">
                <p class="fw-medium step-text mb-0">Selesaikan<br>Pembayaran</p>
              </div>
            </div>
          </div>
        </div>
      </div>
</section>
<!-- end tutorial -->
<!-- start Kelengkapan -->
<section id="Tools" class=" ">
    <div class="container pt-5 border-top border-1 border-dark">
        <h3 class="fw-bold mb-4">Kelengkapan</h3>
        <div class="row g-3">
            <!-- Item 1 -->
            <div class="col-6 col-md-3">
                <div class="bg-white rounded shadow-sm p-3 text-center h-100">
                    <h6 class="fw-bold mb-1">BPKB</h6>
                    <p class="mb-0">{{ $car->bpkb ? 'Ya' : 'Tidak' }}</p>
                </div>
            </div>
            <!-- Item 2 -->
            <div class="col-6 col-md-3">
                <div class="bg-white rounded shadow-sm p-3 text-center h-100">
                    <h6 class="fw-bold mb-1">Kunci Serep</h6>
                    <p class="mb-0">{{ $car->spare_key ? 'Ya' : 'Tidak' }}</p>
                </div>
            </div>
            <!-- Item 3 -->
            <div class="col-6 col-md-3">
                <div class="bg-white rounded shadow-sm p-3 text-center h-100">
                    <h6 class="fw-bold mb-1">Buku Manual</h6>
                    <p class="mb-0">{{ $car->manual_book ? 'Ya' : 'Tidak' }}</p>
                </div>
            </div>
```

```
            <!-- Item 4 -->
            <div class="col-6 col-md-3">
                <div class="bg-white rounded shadow-sm p-3 text-center h-100">
                    <h6 class="fw-bold mb-1">Buku Service</h6>
                    <p class="mb-0">{{ $car->service_book ? 'Ya' : 'Tidak' }}</p>
                </div>
            </div>
        </div>
    </div>
</section>
<!-- end Kelengkapan -->
<!-- Start Ringkasan -->
<section class="mt-5">
    <div class="container">
        <h3 class="fw-bold mb-4">Ringkasan</h3>
        <div class="row">
            <div class="col-md-6">
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Brand dan Merek</span>
                    <span class="text-capitalize">{{$car->brand . ' ' . $car->model}} </span>
                </div>
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Tahun Produksi</span>
                    <span>{{$car->year}}</span>
                </div>
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Transmisi</span>
                    <span class="text-capitalize">{{$car->transmission}}</span>
                </div>
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Bahan Bakar</span>
                    <span class="text-capitalize">{{$car->fuel_type}}</span>
                </div>
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Warna</span>
                    <span class="text-capitalize">{{$car->color}}</span>
                </div>
            </div>
            <div class="col-md-6">
                <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                    <span class="fw-semibold">Jarak Tempuh</span>
                    <span>{{ number_format($car->mileage, 0, ',', '.') . ' KM' }}</span>
                </div>
```

```
            <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                <span class="fw-semibold">Jumlah Tempat Duduk</span>
                <span>{{$car->seat .' Orang'}}</span>
            </div>
            <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                <span class="fw-semibold">CC Mesin</span>
                <span>{{ number_format($car->engine, 0, ',', '.') . ' CC' }}</span>
            </div>
            <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                <span class="fw-semibold">Pajak</span>
                <span>{{ \Carbon\Carbon::parse($car->tax)->translatedFormat('d F Y') }}
</span>
            </div>
            <div class="mb-3 pb-2 border-bottom border-1 border-dark d-flex justify-
content-between">
                <span class="fw-semibold">Servis Terakhir</span>
                <span>{{ \Carbon\Carbon::parse($car->service_history)-
>translatedFormat('d F Y') }}</span>
            </div>
        </div>
    </div>
  </div>
</section>
<!-- End Ringkasan -->
<!-- Start Deskripsi -->
<section class="mt-4">
  <div class="container">
    <h3 class="fw-bold mb-4">Deskripsi</h3>
    <p style="line-height: 1.8;">
      {{$car->description}}
    </p>
    <p style="line-height: 1.8;">
    </p>
  </div>
</section>
<!-- End Deskripsi -->
<!-- Start Keamanan -->
<section class="my-5 text-center">
  <div class="container">
    <h3 class="fw-bold mb-5">Semua Bebas</h3>
    <div class="row justify-content-center g-4">
      <!-- Bebas Kebakaran -->
      <div class="col-12 col-md-4">
        <img src="{{ asset('image/Frame 100.svg')}}" width="120" alt="Bebas
Kebakaran" class="mb-3">
        <h6 class="fw-bold">Bebas Kebakaran</h6>
        <p class="mb-0">
```

```
                Telah melalui inspeksi menyeluruh untuk memastikan bebas dari riwayat
kebakaran.
            </p>
        </div>
        <!-- Bebas Kecelakaan -->
        <div class="col-12 col-md-4">
            <img src="{{ asset('image/Frame 101.svg')}}" width="120" alt="Bebas
Kecelakaan" class="mb-3">
            <h6 class="fw-bold">Bebas Kecelakaan</h6>
            <p class="mb-0">
                Terjaga dari kerusakan akibat kecelakaan, kondisi prima!
            </p>
        </div>
        <!-- Bebas Kebanjiran -->
        <div class="col-12 col-md-4">
            <img src="{{ asset('image/Frame 102.svg')}}" width="120" alt="Bebas
Kebanjiran" class="mb-3">
            <h6 class="fw-bold">Bebas Kebanjiran</h6>
            <p class="mb-0">
                Dijamin bukan mobil bekas banjir, aman dan nyaman dikendarai!
            </p>
        </div>
    </div>
  </div>
</section>
<!-- End Keamanan -->
@if(session('showLoginModal'))
<script>
  document.addEventListener('DOMContentLoaded', function () {
    const loginModal = new bootstrap.Modal(document.getElementById('loginModal'));
    loginModal.show();
  });
</script>
@endif
@push('web-scripts')
<script>
  function switchImage(el) {
    const main = document.getElementById("mainImage");
    const zoom = document.getElementById("zoomImage");
    // Set gambar utama
    main.src = el.src;
    // Set gambar zoom juga
    zoom.src = el.src;
    // Highlight aktif
    document.querySelectorAll(".thumb-img").forEach(img =>
img.classList.remove("active"));
    el.classList.add("active");
  }
  function scrollThumbnails(direction) {
    const container = document.getElementById('thumbnailContainer');
```

```
        const scrollAmount = 100; // atau sesuaikan dengan ukuran thumbnail
        container.scrollBy({
            left: direction * scrollAmount,
            behavior: 'smooth'
        });
    }
    document.addEventListener('DOMContentLoaded', function () {
        const container = document.getElementById('thumbnailContainer');
        const scrollLeftBtn = document.querySelector('.scroll-left');
        const scrollRightBtn = document.querySelector('.scroll-right');
        function updateScrollButtons() {
            scrollLeftBtn.style.display = container.scrollLeft > 0 ? 'flex' : 'none';
            scrollRightBtn.style.display =
                container.scrollWidth > container.clientWidth + container.scrollLeft ?
                'flex' : 'none';
        }
        container.addEventListener('scroll', updateScrollButtons);
        window.addEventListener('resize', updateScrollButtons);
        updateScrollButtons();
    });
    function shareLink() {
        const currentUrl = "{{ url()->current() }}";
        if (navigator.share) {
            navigator.share({
                title: 'Lihat Mobil Ini',
                text: 'Cek mobil ini yuk!',
                url: currentUrl
            }).then(() => {
                console.log("Berhasil dibagikan");
            }).catch((err) => {
                console.warn("Gagal membagikan:", err);
            });
        } else {
            // Fallback untuk browser lama: copy to clipboard
            navigator.clipboard.writeText(currentUrl).then(() => {
                alert("Link berhasil disalin ke clipboard!");
            }).catch((err) => {
                alert("Gagal menyalin link: " + err);
            });
        }
    }
</script>
@endpush
@endsection
```

resources/views/web/downPayment.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
<style>
```

```
    #formDP {
        padding-top: 170px;
        padding-bottom: 70px;
    }
    #formDP .form-control {
        background-color: #eeeeee;
        border-radius: 5px;
    }
    #formDP img {
        width: 100%;
        height: auto;
        border-radius: 5px;
    }
    #formDP .btn-submit {
        background-color: var(--primary);
        color: #fcf259;
        font-weight: 600;
        padding: 10px 20px;
        border-radius: 5px;
        border: none;
        transition: background-color 0.3s ease;
        margin-top: 20px;
    }
    .amount-display {
        font-weight: 500;
        font-size: 16px;
    }
</style>
@endpush
{{-- start formDP --}}
<section id="formDP">
    <div class="container">
        <h3 class="fw-bold mb-5">Pembayaran DP</h3>
        <div class="row">
            <form id="formDPSubmit" action="{{route('web.downPayment.store')}}"
method="POST" class="row">
                <!-- Kolom kiri: Form Tanggal dan Waktu -->
                @csrf
                <input type="hidden" name="car_id" value="{{ $car->id }}">
                <div class="col-lg-7">
                    <div class="card border-0 p-3 shadow rounded-3">
                        <div class="card-body">
                            <div class="mb-3">
                                <label for="tanggal" class="form-label fw-medium">Tanggal Janji
Temu</label>
                                <input type="date"
                                    name="appointment_date"
                                    class="form-control py-2 @error('appointment_date') is-invalid
@enderror"
                                    value="{{ old('appointment_date')}}"
```

```
                    id="tanggal"
                    min="{{ \Carbon\Carbon::now()->format('Y-m-d') }}">
                @error('appointment_date')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
            <div>
                <label for="waktu" class="form-label fw-medium">Waktu</label>
                <input type="time"
                    class="form-control py-2 @error('appointment_time') is-invalid
@enderror"
                    name="appointment_time" value="{{old('appointment_time')}}"
id="waktu">
                @error('appointment_time')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
        </div>
    </div>
    <!-- Kolom untuk jumlah DP -->
    <div class="card border-0 p-3 shadow rounded-3 my-5">
        <div class="card-body">
            <div>
                <label for="jumlahDP" class="form-label fw-medium">Masukkan
Jumlah DP || Minimal DP Rp
                    500.000</label>
                {{-- <input type="text" class="form-control py-2 amount-display"
                    placeholder="Ketik jumlah DP" name="amount" id="jumlahDP"
/> --}}
                <input type="text" class="form-control @error('amount') is-invalid
@enderror"
                    id="amount_format" value="" required />
                <input type="hidden" name="amount" id="amount" value="{{
old('amount') }}" />
                @error('amount')
                <span class="invalid-feedback" role="alert">
                    <strong>{{ $message }}</strong>
                </span>
                @enderror
            </div>
        </div>
    </div>
</div>
<!-- Kolom kanan: Info Mobil -->
<div class="col-lg-5">
    <div class="card border-0 p-3 shadow rounded-3">
```

```
                <div class="card-body">
                    <div class="row">
                        <div class="col-md-5 mb-3 mb-md-0">
                            <img src="{{ $car->mainPhoto && Storage::disk('public')-
>exists('car_photos/' . $car->mainPhoto->photo_url)
            ? asset('storage/car_photos/' . $car->mainPhoto->photo_url)
            : asset('image/NoImage.png') }}" alt="{{ $car->brand }} {{ $car->model }}"
class=" img-fluid" />
                        </div>
                        <div class="col-md-7">
                            <p class="fw-bold text-capitalize">
                                {{$car->year . ' - ' . $car->brand . ' ' . $car->model . ' ' . $car-
>color}}</p>
                            <p class="text-capitalize">
                                {{ number_format($car->mileage, 0, ',', '.') . ' KM | ' . $car-
>transmission }}
                            </p>
                        </div>
                    </div>
                    <div class="mt-md-4 mt-1">
                        <p class="fw-medium mb-1">Lokasi Mobil: </p>
                        <p>Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU
TAPOS DEPOK</p>
                    </div>
                </div>
            </div>
            <!-- Tombol Submit -->
            <div class="row">
                <div class="col-12">
                    <button type="submit" class="btn btn-submit w-100">Submit</button>
                </div>
            </div>
        </div>
    </form>
  </div>
 </div>
</section>
{{-- end formDp --}}
@push('web-scripts')
<script
src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap/5.3.0/js/bootstrap.bundle.min.js"></
script>
<script>
  // Format input jumlah DP
  // const jumlahDPInput = document.getElementById('jumlahDP');
  // jumlahDPInput.addEventListener('input', function (e) {
  //     let value = e.target.value.replace(/[^\d]/g, ''); // Hapus selain angka
  //     if (value) {
  //         // Format number with thousand separator
  //         let formatted = parseInt(value).toLocaleString('id-ID');
```

```
    //        e.target.value = 'Rp ' + formatted;
    //    } else {
    //        e.target.value = ''; // kosongkan, biar placeholder "Ketik jumlah DP" muncul
lagi
    //    }
    // });
    // Format input amount to Rupiah
    function formatRupiah(angka) {
        let numberString = angka.replace(/\D/g, "");
        let formatted = numberString.replace(/\B(?=(\d{3})+(?!\d))/g, ".");
        return numberString ? 'Rp ' + formatted : '';
    }
    document.addEventListener('DOMContentLoaded', function () {
        const inputFormatted = document.getElementById('amount_format');
        const inputHidden = document.getElementById('amount');
        inputFormatted.addEventListener('input', function () {
            let clean = this.value.replace(/\D/g, "");
            this.value = formatRupiah(clean);
            inputHidden.value = clean;
        });
    });
    document.getElementById('formDPSubmit').addEventListener('submit', function (e) {
        e.preventDefault();
        const tanggal = document.getElementById('tanggal').value;
        const waktu = document.getElementById('waktu').value;
        const jumlahDP = document.getElementById('amount').value;
        if (!tanggal || !waktu || jumlahDP.trim() === '') {
            alert('Mohon lengkapi semua field yang diperlukan!');
            return;
        }
        const dpAmount = parseInt(jumlahDP.replace(/[^\d]/g, ''));
        if (dpAmount < 500000) {
            alert('Minimal DP adalah Rp 500.000');
            return;
        }
        this.submit(); // Lanjutkan submit
    });
</script>
@endpush
@endsection
```

resources/views/web/home.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
<!-- CSS Select2 -->
<!-- CSS Select2 -->
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css"
rel="stylesheet" />
<!-- Jika kamu pakai Bootstrap 5 dan ingin tampilan serasi -->
```

```
<link href="https://cdn.jsdelivr.net/npm/select2-bootstrap-5-theme@1.3.0/dist/select2-
bootstrap-5-theme.min.css"
   rel="stylesheet" />
<style>
   /* padding dan tinggi agar konsisten dengan py-3 input */
   .select2-container--bootstrap-5 .select2-selection {
      padding: 1rem 1rem;
      /* Sama seperti py-3 */
      height: auto !important;
      min-height: 48px;
      /* Sesuaikan dengan input py-3 */
      border: 1px solid #ced4da;
      border-radius: 0.375rem;
      display: flex;
      align-items: center;
      /* Vertikal centering */
   }
   /* Hapus padding ganda dari teks */
   .select2-container--bootstrap-5 .select2-selection__rendered {
      padding: 0;
      margin: 0;
      line-height: normal;
   }
   /* Hapus padding dari arrow dropdown */
   .select2-container--bootstrap-5 .select2-selection__arrow {
      top: 50%;
      transform: translateY(-50%);
   }
   #form {
      padding-top: 120px;
   }
   #mobil {
      margin: 50px 0;
   }
   #mobil .card:hover {
      box-shadow: 0 0 15px rgba(0,0,0,0.1);
      transform: translateY(-2px);
      transition: all 0.2s ease-in-out;
   }
   #testimonial {
      /* margin-top: 50px; */
      /* padding: 100px 0; */
      background: linear-gradient(to bottom, var(--background), var(--yellow));
   }
   #testimonial h5 {
      font-size: 18px;
      font-weight: 500;
   }
</style>
@endpush
```

```
{{-- start form --}}
<section id="form">
   <div class="container my-5">
      <form action="{{ route('home') }}" method="GET">
         <div class="row g-2 align-items-center">
            <div class="col-md-9 col-sm-12">
               <input name="q" type="text" class="form-control py-3" placeholder="Cari
mobil..."
                  value="{{ request('q') }}">
            </div>
            <div class="col-md-3 col-sm-12">
               <button type="submit" class="btn w-100 py-3"
                  style="background-color: #fff04f; font-weight: bold;">CARI</button>
            </div>
         </div>
         <div class="row g-2 mt-3">
            <div class="col-4 col-sm-3">
               <select name="brand" class="form-select py-3 brand" data-placeholder="--
Pilih Brand --">
                  <option selected disabled>Brand</option>
                  {{-- <option></option> --}}
                  @foreach ($brands as $brand)
                  <option value="{{ $brand }}" {{ request('brand') == $brand ? 'selected' : "
}}>{{ $brand }}
                  </option>
                  @endforeach
               </select>
            </div>
            <div class="col-4 col-sm-3">
               <select name="mileage" class="form-select py-3 mileage">
                  <option selected disabled>Jarak Tempuh</option>
                  @for ($i = 0; $i < 100000; $i +=10000) @php $min=$i; $max=$i +
10000; $value="$min-$max" ;
                     @endphp <option value="{{ $value }}" {{ request('mileage') == $value
? 'selected' : " }}>
                     {{ number_format($min, 0, ',', '.') }} - {{ number_format($max, 0, ',', '.')
}} km
                     </option>
                     @endfor
                     {{-- Opsi tambahan: 100.000+ km --}}
                     <option value="100000-999999" {{ request('mileage') == '100000-
999999' ? 'selected' : " }}>
                        100.000+ km
                     </option>
               </select>
            </div>
            <div class="col-4 col-sm-3">
               <select name="year" class="form-select py-3 year">
                  <option selected disabled>Tahun</option>
                  @for ($year = 2025; $year >= 2000; $year--)
```

```
                <option value="{{ $year }}" {{ request('year') == $year ? 'selected' : ''
}}>{{ $year }}
                </option>
                @endfor
                <!-- Tambahkan tahun lainnya -->
            </select>
        </div>
    </div>
  </form>
 </div>
</section>
<!-- end form -->
<!-- start list mobil-->
<section id="mobil">
 <div class="container my-5">
  <h3 class="fw-bold mb-4">Mobil Tersedia</h3>
  <div class="row g-4">
   <!-- Card 1 -->
   @forelse ($cars as $item)
   <div class="col-md-4 col-6">
    <a href="{{route('web.detailMobil', $item->id)}}" class="text-decoration-
none">
     <div class="card shadow-sm h-100 rounded-top">
      <div
       class="ratio ratio-4x3 bg-light d-flex align-items-center justify-content-
center text-dark rounded-top">
       <img src="{{ $item->mainPhoto && Storage::disk('public')-
>exists('car_photos/' . $item->mainPhoto->photo_url)
     ? asset('storage/car_photos/' . $item->mainPhoto->photo_url)
     : asset('image/NoImage.png') }}" alt="{{ $item->brand }} {{ $item->model
}}"
        class="img-fluid object-fit-cover w-100 h-100 rounded-top">
      </div>
      <div class="card-body">
       <h6 class="card-title fw-bold text-capitalize">{{ $item->brand }} {{
$item->model }}
        {{ $item->year }} </h6>
       <p class="mb-2 text-capitalize" style="font-size: 14px;">
        {{ $item->year }} | {{ $item->mileage }} km | {{ $item-
>transmission }}
       </p>
       <p class="fw-bold mb-0">Rp {{ number_format($item->price, 0, ',', '.')
}}</p>
      </div>
     </div>
    </a>
   </div>
   @empty
   <div class="col-12 text-center">
```

```
            <h4 class="text-muted mb-3 text-capitalize">{{ request('q') ?? ' ' }} {{
request('brand') ?? ' ' }}
                {{ request('mileage') ?? ' ' }} {{ request('year') ?? ' ' }}Tidak
Ditemukan</h4>
            <a href="{{ route('home') }}" class="btn btn-outline-secondary">Reset
Filter</a>
        </div>
        @endforelse
      </div>
    </div>
</section>
<!-- end list mobil -->
<!-- start testimonial -->
<section id="testimonial">
    <div class="container py-5">
      <h3 class="text-center fw-bold text-primary mb-5">Apa Kata Pelanggan
Kami</h3>
      <div class="row g-4 justify-content-center">
        <!-- Testimonial Card -->
        @forelse ($reviews as $review)
          <div class="col-md-4 col-6">
            <div class="card border-0 shadow-sm rounded-4 overflow-hidden">
              <img src="{{asset('storage/photo_review/' . $review->photo_review)}}"
class="img-fluid" alt="Foto Pelanggan">
              <div class="card-body text-center">
                <h5 class="card-title">{{$review->user->name}}</h5>
                <div class="mb-2 text-warning">
                  @for ($i = 0; $i < $review->rating; $i++)
                    <i class="fas fa-star"></i>
                  @endfor
                </div>
                <p class="card-text text-muted" style="font-size: 14px;">
                  {{ $review->comment }}
                </p>
              </div>
            </div>
          </div>
          @empty
          <div class="col-12 text-center">
            <h4 class="text-muted mb-3">Belum Ada Testimonial</h4>
          </div>
        @endforelse
      </div>
      <!-- Tombol -->
      <div class="text-end mt-4">
        <a href="{{route('web.testimonial')}}" class="text-dark  text-decoration-none d-
inline-flex align-items-center">
          Lihat Yang Lain
          <span class="ms-2">→</span>
        </a>
```

```
        </div>
    </div>
</section>
<!-- end testimonial -->
<!-- Start About -->
<section id="about">
    <div class="py-5">
        <div class="container">
            <div class="row align-items-start">
                <!-- Kolom Foto -->
                <div class="col-lg-5 mb-4 mb-lg-0">
                    <div class="border rounded-3 overflow-hidden shadow-sm" style="height:
100%; max-height: 350px;">
                        <img src="{{asset('image/Showroom.jpeg')}}"
                            alt="YONOMOBILINDO SHOWROOM" class="img-fluid w-100 h-
100 object-fit-cover">
                    </div>
                </div>
                <!-- Kolom Teks -->
                <div class="col-lg-7">
                    <h4 class="fw-bold">Pak Sridaryono - Pemilik Showroom</h4>
                    <h3 class="fw-bold mb-4 text-uppercase">YONOMOBILINDO</h3>
                    <p class="text-muted">
                        Dengan pengalaman lebih dari 15 tahun di dunia otomotif, saya
berkomitmen untuk menyediakan
                        mobil
                        berkualitas dengan harga terbaik di pasaran.
                        Setiap mobil yang ada di showroom ini telah melalui proses seleksi ketat
dan pemeriksaan
                        menyeluruh untuk
                        memastikan kondisinya benar-benar prima.
                    </p>
                    <p class="text-muted">
                        Kepuasan pelanggan adalah prioritas utama kami. Kami sangat terbuka
dalam memberikan
                        informasi tentang
                        kondisi mobil secara detail dan jujur.
                        Kami juga menyediakan layanan purna jual dan garansi untuk memberikan
rasa aman dan nyaman
                        bagi para
                        pelanggan.
                    </p>
                    <!-- Statistik -->
                    <div class="row text-center mt-4">
                        <div class="col-4">
                            <h4 class="fw-bold">150+</h4>
                            <p class="mb-0">Mobil Terjual</p>
                        </div>
                        <div class="col-4">
                            <h4 class="fw-bold">98%</h4>
```

```
                <p class="mb-0">Pelanggan Puas</p>
            </div>
            <div class="col-4">
                <h4 class="fw-bold">15+</h4>
                <p class="mb-0">Tahun Pengalaman</p>
            </div>
        </div>
    </div>
  </div>
</section>
<!-- End About -->
<!-- start form request Mobil -->
{{-- <section id="reqCar">
  <div class="py-5">
    <div class="container">
      <div class="mx-auto p-5 shadow rounded-3 bg-white" style="max-width:
900px;">
        <h3 class="fw-bold text-center mb-2">Belum Ketemu Mobilnya? Kami Bantu
Cari!</h3>
        <p class="text-center text-muted mb-4">Isi detail mobil yang Anda cari, tim
kami akan segera
          menghubungi Anda.
        </p>
        <form>
          <div class="mb-3">
            <label for="carRequest" class="form-label fw-semibold">Merek, Mobil,
Tahun<span
                class="text-danger">*</span></label>
            <input type="text" id="carRequest" class="form-control form-control-lg
shadow-sm border-dark"
              placeholder="Cth. Daihatsu Xenia 2024" required>
          </div>
          <div class="text-end">
            <button type="submit" class="btn px-4 py-2 fw-bold text-primary"
              style="background-color: var(--yellow);">
              Beritahu saya
            </button>
          </div>
        </form>
      </div>
    </div>
  </div>
</section> --}}
<!-- end form request Mobil -->
@if(session('showLoginModal'))
<script>
  document.addEventListener('DOMContentLoaded', function () {
    const loginModal = new bootstrap.Modal(document.getElementById('loginModal'));
```

```
        loginModal.show();
    });
</script>
@endif
@push('web-scripts')
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/js/select2.min.js"></script>
<script>
    $(document).ready(function () {
        $('.brand, .mileage , .year').select2({
            theme: 'bootstrap-5',
            width: '100%',
            allowClear: true,
            placeholder: '-- Pilih --',
        });
    });
</script>
@endpush
@endsection
```

resources/views/web/jualMobil.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
{{-- select 2 --}}
<link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css"
rel="stylesheet" />
<link href="https://cdn.jsdelivr.net/npm/select2-bootstrap-5-theme@1.2.0/dist/select2-
bootstrap-5-theme.min.css"
    rel="stylesheet" />
{{-- flatPickr --}}
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/flatpickr/dist/flatpickr.min.css">
<style>
        .select2-container--bootstrap-5 .select2-selection {
        padding: 0.5em 1rem;
        /* Sama seperti py-3 */
        height: auto !important;
        /* min-height: 48px; */
        /* Sesuaikan dengan input py-3 */
        border: 0px solid #ced4da;
        border-radius: 0.375rem;
        background-color:#eeeeee;
        display: flex;
        align-items: center;
        /* Vertikal centering */
    }
    /* Hapus padding ganda dari teks */
    .select2-container--bootstrap-5 .select2-selection__rendered {
        padding: 0;
```

```
    margin: 0;
    /* line-height: normal; */
}
/* Hapus padding dari arrow dropdown */
.select2-container--bootstrap-5 .select2-selection__arrow {
    top: 50%;
    transform: translateY(-50%);
}
#formJualMobil {
    padding-top: 170px;
}
#formJualMobil .form-control {
    background-color: #eeeeee;
    border-radius: 5px;
}
#formJualMobil .btn-submit {
    background-color: var(--primary);
    color: #fcf259;
    font-weight: 600;
    padding: 10px 20px;
    border-radius: 5px;
    border: none;
    transition: background-color 0.3s ease;
    margin-top: 20px;
}
#tutorial .border {
    background-color: var(--background);
}
#tutorial .number {
    font-size: 48px;
    color: var(--grey, #867e77);
    font-weight: 700;
}
#tutorial .step-text {
    font-size: 16px;
    color: var(--black);
    font-weight: 700;
}
#tutorial img.step-icon {
    width: 70px;
}
@media (max-width: 576px) {
    #tutorial .number {
        font-size: 20px;
    }
    #tutorial .step-text {
        font-size: 12px;
        color: var(--black);
        font-weight: 700;
    }
```

```
        #tutorial img.step-icon {
            width: 40px;
        }
        #tutorial .step-text {
            font-size: 10px;
        }
        #tutorial .step-header {
            padding-left: 0.5rem !important;
            padding-right: 0.5rem !important;
        }
        #tutorial .step-body {
            padding: 1rem 0.5rem;
        }
    }
    #testimonial {
        /* margin-top: 50px; */
        /* padding: 100px 0; */
        background: linear-gradient(to bottom, var(--background), var(--yellow));
    }
    #testimonial h5 {
        font-size: 18px;
        font-weight: 500;
    }
</style>
@endpush
<!-- start formJualMobil -->
<section id="formJualMobil">
    <div class="container">
        <h3 class="fw-bold mb-5">FORM JUAL MOBIL</h3>
        <form action="{{route('web.jualMobil.store')}}" method="POST">
            @csrf
            <div class="row d-flex align-items-start">
                <div class="col-md-6 " style="font-size: 20px;">
                    <p class="fw-semibold mb-2">Informasi Mobil</p>
                    <p class="text-muted">Masukkan Spesifikasi Mobil Anda Secara
Lengkap</p>
                </div>
                <div class="col-md-6">
                    <div class="row">
                        <div class="col-md-6">
                            <div class="mb-3">
                                <label for="brand" class="form-label fw-medium">Brand*</label>
                                <select name="brand" id="brand" class="form-control py-2 border-0"
required>
                                    <option selected disabled>Ketik Atau Pilih</option>
                                    {{-- <option value="">-- Pilih Brand --</option> --}}
                                    @foreach ($brands as $brand)
                                    <option value="{{ $brand }}">{{ $brand }}</option>
                                    @endforeach
                                </select>
```

```
                    </div>
                </div>
                <div class="col-md-6">
                    <div class="mb-3">
                        <label for="model" class="form-label fw-medium">Model*</label>
                        <select name="model" id="model" class="form-control py-2 border-
0" required disabled>
                            <option selected disabled>Ketik Atau Pilih</option>
                            {{-- <option selected disabled>Brand</option> --}}
                        </select>
                    </div>
                </div>
                <div class="col-md-6">
                    <div class="mb-3">
                        <label for="year" class="form-label fw-medium ">Tahun*</label>
                        <select name="year" id="year" class="form-control py-2 border-0"
required>
                            <option selected disabled>Ketik Atau Pilih</option>
                            {{-- <option value="">Ketik Atau Pilih</option> --}}
                            @for ($year = 2025; $year >= 2000; $year--)
                                <option value="{{ $year }}" {{ request('year') == $year ?
'selected' : '' }}>{{ $year }}
                                </option>
                            @endfor
                        </select>
                    </div>
                </div>
                <div class="col-md-6">
                    <div class="mb-3">
                        <label for="mileage_format" class="form-label fw-medium">Jarak
Tempuh (KM)*</label>
                        {{-- <input type="number" name="mileage" id="mileage"
                            class="form-control py-2 @error('mileage') is-invalid @enderror"
                            value="{{ old('mileage') }}" placeholder="Ketik" required /> --}}
                        <input type="text" class="form-control @error('mileage') is-invalid
@enderror" id="mileage_format" value="{{ old('mileage') ?
number_format(old('mileage'), 0, ',', '.') : '' }}" required />
                        <input type="hidden" name="mileage" id="mileage" value="{{
old('mileage') }}" />
                    </div>
                </div>
                <div class="col-md-6">
                    <div class="mb-3">

                        <label for="offered_price_format" class="form-label fw-
medium">Penawaran Harga*</label>
                        <input type="text" class="form-control py-2 @error('offered_price')
is-invalid @enderror" id="offered_price_format"
                            value="{{ old('offered_price') ?
number_format(old('offered_price'), 0, ',', '.') : '' }}" required />
```

```
                    <input type="hidden" name="offered_price" id="offered_price"
value="{{ old('offered_price') }}" />
                    @error('price')
                    <span class="invalid-feedback" role="alert">
                        <strong>{{ $message }}</strong>
                    </span>
                    @enderror
                  </div>
                </div>
              </div>
            </div>
          </div>
          <div class="row d-flex align-items-start mt-5">
            <div class="col-md-6 " style="font-size: 20px;">
              <p class="fw-semibold mb-2">Detail Inspeksi</p>
              <p class="text-muted">Isi Lokasi dan Tanggal Inspeksi untuk Validasi
Kondisi Kendaraan Anda</p>
            </div>
            <div class="col-md-6">
              <div class="row">
                <div class="col-md-6">
                  <div class="mb-3">
                    <label for="location_inspection" class="form-label fw-
medium">Lokasi Inspeksi*</label>
                    <select name="location_inspection" id="location_inspection"
                        class="form-control py-2 border-0" required>
                        <option value="">Ketik Atau Pilih</option>
                        <option value="Rumah">Rumah</option>
                        <option value="Datang Ke Showroom">Datang Ke
Showroom</option>
                    </select>
                  </div>
                </div>
                <div class="col-md-6">
                  <div class="mb-3">
                    <label for="inspection_date" class="form-label fw-
medium">Tanggal & Waktu Inspeksi*</label>
                    <input type="datetime-local"
                        name="inspection_date"
                        class="form-control py-2 border-0"
                        id="inspection_date"
                        placeholder="Ketik"
                        value="{{ old('inspection_date', \Carbon\Carbon::now()-
>format('Y-m-d\TH:i')) }}"
                        min="{{ \Carbon\Carbon::now()->format('Y-m-d\TH:i') }}"
                        required>
                  </div>
                </div>
              </div>
            </div>
```

```
            </div>
            @auth
                <div class="justify-content-end d-flex mt-3">
                    <button type="submit" class="btn btn-submit col-12 col-md-
3">Submit</button>
                </div>
                @else
                <div class="justify-content-end d-flex mt-3">
                    <a href="{{ route('login', ['redirect' => request()->fullUrl()]) }}" class="btn
btn-outline-primary fw-semibold  py-2" >
                        <span>Login untuk Jual Mobil</span><br>
                        <small class="fw-normal text-muted" style="font-size: 12px;">Akses fitur
hanya untuk pengguna
                            terdaftar</small>
                    </a>
                </div>
            @endauth
        </form>
    </div>
</section>
<!-- end formJualMobil -->
<!-- start tutorial -->
<section id="tutorial" class="py-5 mt-5">
    <div class="container text-center">
        <h3 class="fw-bold mb-5">Proses Beli Mobil</h3>
        <div class="row g-3">
            <!-- STEP 1 -->
            <div class="col-3">
                <div class="border rounded-3 h-100">
                    <div
                        class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
                        <span class="step-text">STEP</span>
                        <span class="number">01</span>
                    </div>
                    <div class="step-body py-2 py-md-4">
                        <img src="{{asset('image/Isi Data Mobil.svg')}}" alt="DP" class="step-
icon mb-3">
                        <p class="fw-medium step-text mb-0">Isi Data Mobil & isi detail
inspeksi</p>
                    </div>
                </div>
            </div>
            <!-- STEP 2 -->
            <div class="col-3">
                <div class="border rounded-3 h-100">
                    <div
                        class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
                        <span class="step-text">STEP</span>
```

```
                <span class="number">02</span>
            </div>
            <div class="step-body py-2 py-md-4">
                <img src="{{ asset('image/VERIFIKASI.svg')}}" alt="Verifikasi &
Konfirmasi Jadwal"
                    class="step-icon mb-3">
                <p class="fw-medium step-text mb-0">Verifikasi & Konfirmasi
Jadwal</p>
            </div>
        </div>
    </div>
    <!-- STEP 3 -->
    <div class="col-3">
        <div class="border rounded-3 h-100">
            <div
                class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
                <span class="step-text">STEP</span>
                <span class="number">03</span>
            </div>
            <div class="step-body py-2 py-md-4">
                <img src="{{asset('image/Inspeksi.svg')}}" alt="Proses Inspeksi Mobi"
class="step-icon mb-3">
                <p class="fw-medium step-text mb-0">Proses Inspeksi Mobil</p>
            </div>
        </div>
    </div>
    <!-- STEP 4 -->
    <div class="col-3">
        <div class="border rounded-3 h-100">
            <div
                class="py-1 py-md-3 border-bottom d-flex align-items-center justify-
content-between px-3 step-header">
                <span class="step-text">STEP</span>
                <span class="number">04</span>
            </div>
            <div class="step-body py-2 py-md-4">
                <img src="{{asset('image/Konfirmasi.svg')}}" alt="Konfirmasi dan
Terima Penawaran"
                    class="step-icon mb-3">
                <p class="fw-medium step-text mb-0">Konfirmasi dan Terima
Penawaran</p>
            </div>
        </div>
    </div>
  </div>
 </div>
</section>
<!-- end tutorial -->
<!-- start testimonial -->
```

```
<section id="testimonial">
   <div class="container py-5">
      <h3 class="text-center fw-bold text-primary mb-5">Apa Kata Pelanggan
Kami</h3>
      <div class="row g-4 justify-content-center">
         <!-- Testimonial Card -->
         @forelse ($reviews as $review)
            <div class="col-md-4 col-6">
               <div class="card border-0 shadow-sm rounded-4 overflow-hidden">
                  <img src="{{asset('storage/photo_review/' . $review->photo_review)}}"
class="img-fluid" alt="Foto Pelanggan">
                  <div class="card-body text-center">
                     <h5 class="card-title">{{$review->user->name}}</h5>
                     <div class="mb-2 text-warning">
                        @for ($i = 0; $i < $review->rating; $i++)
                           <i class="fas fa-star"></i>

                        @endfor

                     </div>
                     <p class="card-text text-muted" style="font-size: 14px;">
                        {{ $review->comment }}
                     </p>
                  </div>
               </div>
            </div>
            @empty
            <div class="col-12 text-center">
               <h4 class="text-muted mb-3">Belum Ada Testimonial</h4>
            </div>
         @endforelse
      </div>
      <!-- Tombol -->
      <div class="text-end mt-4">
         <a href="{{route('web.testimonial')}}" class="text-dark  text-decoration-none d-
inline-flex align-items-center">
            Lihat Yang Lain
            <span class="ms-2">→</span>
         </a>
      </div>
   </div>
</section>
<!-- end testimonial -->
@if(session('showLoginModal'))
<script>
   document.addEventListener('DOMContentLoaded', function () {
      const loginModal = new bootstrap.Modal(document.getElementById('loginModal'));
      loginModal.show();
   });
</script>
```

```
@endif
@push('web-scripts')
{{-- select2 --}}
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-
rc.0/dist/js/select2.min.js"></script>
{{-- flatpickr --}}
<script src="https://cdn.jsdelivr.net/npm/flatpickr"></script>
<script src="https://npmcdn.com/flatpickr/dist/l10n/id.js"></script>
<script>
    $(document).ready(function () {
        $('#brand, #model, #year').select2({
            theme: 'bootstrap-5',
            width: '100%',
            allowClear: true,
            placeholder: 'Ketik Atau Pilih',
        });
        $('#brand').on('change', function () {
            const brand = $(this).val();
            $('#model').prop('disabled', true).empty().append('<option
value="">Loading...</option>');
            if (brand) {
                $.get('/api/models', {
                    brand: brand
                }, function (data) {
                    $('#model').empty().append('<option value="">-- Pilih Model --</option>');
                    $.each(data, function (i, model) {
                        $('#model').append(
                            `<option value="${model}">${model}</option>`);
                    });
                    $('#model').prop('disabled', false).trigger('change');
                });
            } else {
                $('#model').empty().append('<option value="">-- Pilih Model --
</option>').prop(
                    'disabled', true);
            }
        });
    });

    // Format input offered_price to Rupiah
    function formatRupiah(angka) {
        let numberString = angka.replace(/\D/g, "");
        let formatted = numberString.replace(/\B(?=(\d{3})+(?!\d))/g, ".");
        return numberString ? 'Rp ' + formatted : '';
    }
    document.addEventListener('DOMContentLoaded', function () {
        const inputFormatted = document.getElementById('offered_price_format');
        const inputHidden = document.getElementById('offered_price');
        inputFormatted.addEventListener('input', function () {
```

```
            let clean = this.value.replace(/\D/g, "");
            this.value = formatRupiah(clean);
            inputHidden.value = clean;
        });
    });
        // Format input mileage to Rupiah
    function formatMileage(angka) {
        return angka.replace(/\D/g, "")
            .replace(/\B(?=(\d{3})+(?!\d))/g, ".");
    }
    document.addEventListener('DOMContentLoaded', function () {
        const inputFormatted = document.getElementById('mileage_format');
        const inputHidden = document.getElementById('mileage');
        inputFormatted.addEventListener('input', function () {
            let clean = this.value.replace(/\D/g, "");
            this.value = formatMileage(clean);
            inputHidden.value = clean;
        });
    });
    // flatpickr
    flatpickr("#inspection_date", {
        enableTime: true,
        dateFormat: "Y-m-d H:i",
        minDate: "{{ date('Y-m-d') }}",
        locale: "id"
    });
</script>
@endpush
@endsection
```

resources/views/web/kontak.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
<style>
    #kontak {
        padding-top: 120px;
    }
</style>
@endpush
<section id="kontak">
    <div class="container py-5 text-center">
        <h3 class="fw-bold mb-5">
            Kontak Kami
        </h3>
        <div class="mb-2">
            <p class="mb-1 fw-semibold ">Lokasi</p>
            <p class="mb-0 ">Pekapuran Jl 1000 RT 06/05 NO 200 SUKAMAJU BARU
                TAPOS, DEPOK</p>
        </div>
```

```
        <div class="mb-2">
            <p class="mb-1 fw-semibold">Kontak</p>
            <p class="mb-0 ">081220745317</p>
        </div>
    </div>
</section>
@endsection
```

resources/views/web/testimonial.blade.php

```
@extends('web.layouts.master')
@section('web-content')
@push('web-styles')
<style>

    #testimonial {
        padding-top: 120px;
        /* padding: 100px 0; */
        /* background: linear-gradient(to bottom, var(--background), var(--yellow)); */
    }
    #testimonial h5 {
        font-size: 18px;
        font-weight: 500;
    }
    #testimonial .masonry {
        column-count: 3;
        column-gap: 1rem;
    }
    #testimonial .masonry .item {
        break-inside: avoid;
        margin-bottom: 1rem;
    }
    @media (max-width: 768px) {
        #testimonial .card-title {
            font-size: 12px;
        }
        #testimonial .card-text {
            font-size: 8px;
        }
        #testimonial .fa-star {
            font-size: 8px;
        }
    }
</style>
@endpush
<section id="testimonial">
    <div class="container py-5">
        <h3 class="text-center fw-bold mb-5">Apa Kata Pelanggan Kami</h3>
        <div class="masonry">
            @forelse ($reviews as $review)
                <div class="item">
```

```
                <!-- card di sini -->
                <div class="card border-0 shadow-sm rounded-4 overflow-hidden">
                    <img src="{{asset('storage/photo_review/' . $review->photo_review)}}"
class="img-fluid" alt="Foto Pelanggan">
                    <div class="card-body text-center">
                        <h5 class="card-title mb-0 mb-md-2">{{$review->user->name}}</h5>
                        <div class="mb-md-2 mb-1 text-warning ">
                            @for ($i = 0; $i < $review->rating; $i++)
                                <i class="fas fa-star"></i>

                            @endfor

                        </div>
                        <p class="card-text text-muted">
                            {{ $review->comment }}
                        </p>
                    </div>
                </div>
            </div>
            @empty
            <div class="col-12 text-center">
                <h4 class="text-muted mb-3">Belum Ada Testimonial</h4>
            </div>
        @endforelse
    </div>
    {{-- <div class="row g-4 justify-content-center">
        <!-- Testimonial Card -->
        @forelse ($reviews as $review)
            <div class="col-md-4 col-sm-6">
                <div class="card border-0 shadow-sm rounded-4 overflow-hidden">
                    <img src="{{asset('storage/photo_review/' . $review->photo_review)}}"
class="img-fluid" alt="Foto Pelanggan">
                    <div class="card-body text-center">
                        <h5 class="card-title">{{$review->user->name}}</h5>
                        <div class="mb-2 text-warning">
                            @for ($i = 0; $i < $review->rating; $i++)
                                <i class="fas fa-star"></i>

                            @endfor

                        </div>
                        <p class="card-text text-muted" style="font-size: 14px;">
                            {{ $review->comment }}
                        </p>
                    </div>
                </div>
            </div>
            @empty
            <div class="col-12 text-center">
                <h4 class="text-muted mb-3">Belum Ada Testimonial</h4>
```

```
            </div>
          @endforelse
        </div> --}}
        {{-- <!-- Tombol -->
        <div class="text-end mt-4">
          <a href="#" class="text-dark  text-decoration-none d-inline-flex align-items-
center">
            Lihat Yang Lain
            <span class="ms-2">→</span>
          </a>
        </div> --}}
    </div>
</section>
@endsection
```

routes/api.php

```
<?php
use App\Http\Controllers\User\PaymentController;
use Illuminate\Support\Facades\Route;
Route::post('/midtrans/webhook', [PaymentController::class, 'notificationHandler']);
Route::get('/check-midtrans-status/{id}', [PaymentController::class,
'checkMidtransStatus']);
```

routes/web.php

```
<?php
use App\Http\Controllers\Admin\CarPhotoController;
use App\Http\Controllers\Admin\CarsController;
use App\Http\Controllers\Admin\DashboardController;
use App\Http\Controllers\Admin\OfferController as AdminOfferController;
use App\Http\Controllers\Admin\UsersController;
use App\Http\Controllers\Auth\LoginController;
use App\Http\Controllers\User\DownPaymentController as
UserDownPaymentController;
use App\Http\Controllers\User\OfferController;
use App\Http\Controllers\User\PaymentController;
use App\Http\Controllers\User\ProfileController;
use App\Http\Controllers\User\TransactionOfferRecordController;
use App\Http\Controllers\User\TransactionSalesRecordController;
use App\Http\Controllers\Web\DetailCarController;
use App\Http\Controllers\Web\DownPaymentController;
use App\Http\Controllers\Web\HomeController;
use App\Http\Controllers\Web\JualMobilController;
use App\Http\Controllers\Web\KontakController;
use App\Http\Controllers\Web\TestimonialController;
use Illuminate\Support\Facades\Route;
use App\Http\Middleware\CheckRole;
use Illuminate\Http\Request;
use App\Models\CarType;
use Illuminate\Support\Facades\Auth;
// route authentication
```

```
Auth::routes();
Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home-
cms');
// API route to get car models based on brand
Route::get('/api/models', function (Request $request) {
    $brand = $request->query('brand');
    if (!$brand) {
        return response()->json([]);
    }
    $models = CarType::where('brand', $brand)
        ->orderBy('model', 'asc')
        ->pluck('model')
        ->unique()
        ->values();
    return response()->json($models);
});
//group route with prefix "admin" with middleware "auth" and "checkrole:admin"
Route::prefix('admin')->middleware(['auth', 'checkrole:admin'])->group(function () {
    // Dashboard Route
    Route::get('dashboard', [DashboardController::class, 'index'])-
>name('admin.dashboard');
    // Users Route
    Route::resource('users', UsersController::class)->names('admin.users');
    // Cars Route
    Route::resource('cars', CarsController::class)->names('admin.cars');
    // Cars Route with custom methods
    Route::get('cars/{id}/edit-status', [CarsController::class, 'editStatus']);
    Route::post('cars/{id}/update-status', [CarsController::class, 'updateStatus']);
    // Car Photos Route
    Route::resource('cars-photo', CarPhotoController::class)->names('admin.carPhotos')-
>except(['show', 'edit', 'update', 'create', 'store']);
    // Offer Route
    Route::get('offer',[AdminOfferController::class, 'index'])->name('admin.offer.index');
    Route::get('offer/{id}',[AdminOfferController::class, 'show'])-
>name('admin.offer.show');
    Route::get('offer/{id}/edit-status', [AdminOfferController::class, 'editStatus']);
    Route::post('offer/{id}/update-status', [AdminOfferController::class, 'updateStatus']);
    Route::post('offer/updateNote/{id}', [AdminOfferController::class, 'updateNote'])-
>name('admin.offer.updateNote');
    // Down Payment Route
    Route::resource('downPayment',
App\Http\Controllers\Admin\DownPaymentController::class)->only(['index', 'show'])-
>names('admin.downPayment');
    Route::get('downPayment/addRefund/{id}',
[App\Http\Controllers\Admin\DownPaymentController::class, 'addRefund'])-
>name('admin.downPayment.addRefund');
    Route::post('downPayment/storeRefund/{id}',
[App\Http\Controllers\Admin\DownPaymentController::class, 'storeRefund'])-
>name('admin.downPayment.storeRefund');
```

```
    Route::get('downPayment/editRefund/{id}',
[App\Http\Controllers\Admin\DownPaymentController::class, 'editRefund'])-
>name('admin.downPayment.editRefund');
    Route::post('downPayment/updateRefund/{id}',
[App\Http\Controllers\Admin\DownPaymentController::class, 'updateRefund'])-
>name('admin.downPayment.updateRefund');
    // Sales Records Route
    Route::resource('sales-records',
App\Http\Controllers\Admin\SalesRecordController::class)-
>names('admin.salesRecord')->except(['show', 'edit', 'update']);
    // Offer Records Route
    Route::resource('offer-records',
App\Http\Controllers\Admin\OfferRecordController::class)-
>names('admin.offerRecord')->except(['show', 'edit', 'update']);
});
//group route with prefix "user" with middleware "auth" and "checkrole:user"
Route::prefix('user')->middleware(['auth', 'checkrole:user'])->group(function () {
    // Dashboard Route
    Route::get('dashboard', [App\Http\Controllers\User\DashboardController::class,
'index'])->name('user.dashboard');
    // Down Payment Route
    Route::get('downPayment', [UserDownPaymentController::class, 'index'])-
>name('user.downPayment.index');
    Route::get('downPayment/{id}', [UserDownPaymentController::class, 'checkout'])-
>name('user.downPayment.show');
    Route::get('downPayment/checkout/{id}', [PaymentController::class, 'checkout'])-
>name('user.downPayment.checkout');
    Route::get('downPayment/change/{id}', [PaymentController::class, 'changeStatus'])-
>name('user.downPayment.changeStatus');
    // Offer Route
    Route::resource('offer', OfferController::class)->names('user.offer')->except(['edit',
'update', 'create', 'destroy']);
    // updateAddress Route
    Route::post('updateUser/{id}', [ProfileController::class, 'updateAddress'])-
>name('user.updateAdress');
    // Transaction Sales Record Route
    Route::get('transaction-sales-record', [TransactionSalesRecordController::class,
'index'])->name('user.transactionSalesRecord.index');
    Route::get('transaction-sales-record/createTesti/{id}',
[TransactionSalesRecordController::class, 'createTesti'])-
>name('user.transactionSalesRecord.createTesti');
    Route::post('transaction-sales-record/storeTesti/{id}',
[TransactionSalesRecordController::class, 'storeTesti'])-
>name('user.transactionSalesRecord.storeTesti');
    // Transaction Offer Record Route
    Route::get('transaction-offer-record', [TransactionOfferRecordController::class,
'index'])->name('user.transactionOfferRecord.index');
});
Route::post('/midtrans/notification', [PaymentController::class, 'notificationHandler']);
//Home Route
```

```
Route::get('/', [HomeController::class, 'index'])->name('home');
// Route::get('/show-login', [LoginController::class, 'showLoginForm'])-
>name('showLoginModal');
Route::prefix('web')->group(function () {
    // jual mobil
    Route::get('jualMobil', [JualMobilController::class, 'index'])-
>name('web.jualMobil.index');
    Route::post('jualMobil', [JualMobilController::class, 'store'])-
>name('web.jualMobil.store')->middleware(['auth', 'checkrole:user']);
    // testimonial
    Route::get('testimonial', [TestimonialController::class, 'index'])-
>name('web.testimonial');
    // kontak
    Route::get('kontak', [KontakController::class, 'index'])->name('web.kontak');
    Route::get('/detailMobil/{id}', [DetailCarController::class,'index'])-
>name('web.detailMobil');
    // Down Payment Route
    Route::resource('downPayment', DownPaymentController::class)-
>names('web.downPayment')->middleware(['auth', 'checkrole:user'])->except(['index',
'edit', 'update', 'create']);
});
```

.env

```
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:2ZIVknKFvBDBl49E82zXHa8XifFMaDl7wzEbME62hbQ=
APP_DEBUG=true
APP_URL=http://localhost
APP_LOCALE=en
APP_FALLBACK_LOCALE=en
APP_FAKER_LOCALE=en_US
APP_MAINTENANCE_DRIVER=file
# APP_MAINTENANCE_STORE=database
PHP_CLI_SERVER_WORKERS=4
BCRYPT_ROUNDS=12
LOG_CHANNEL=stack
LOG_STACK=single
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=yono_mobilindo
DB_USERNAME=root
DB_PASSWORD=
SESSION_DRIVER=database
SESSION_LIFETIME=120
SESSION_ENCRYPT=false
SESSION_PATH=/
SESSION_DOMAIN=null
```

```
BROADCAST_CONNECTION=log
FILESYSTEM_DISK=local
QUEUE_CONNECTION=database
CACHE_STORE=database
# CACHE_PREFIX=
MEMCACHED_HOST=127.0.0.1
REDIS_CLIENT=phpredis
REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379
MAIL_MAILER=log
MAIL_SCHEME=null
MAIL_HOST=127.0.0.1
MAIL_PORT=2525
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_FROM_ADDRESS="hello@example.com"
MAIL_FROM_NAME="${APP_NAME}"
AWS_ACCESS_KEY_ID=
AWS_SECRET_ACCESS_KEY=
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=
AWS_USE_PATH_STYLE_ENDPOINT=false
VITE_APP_NAME="${APP_NAME}"
MIDTRANS_SERVER_KEY=SB-Mid-server-rn2RSNKwISLfFNY536LjKb3X
MIDTRANS_CLIENT_KEY=SB-Mid-client-EhW68inFMJ6O2eJ0
MIDTRANS_IS_PRODUCTION=false
```