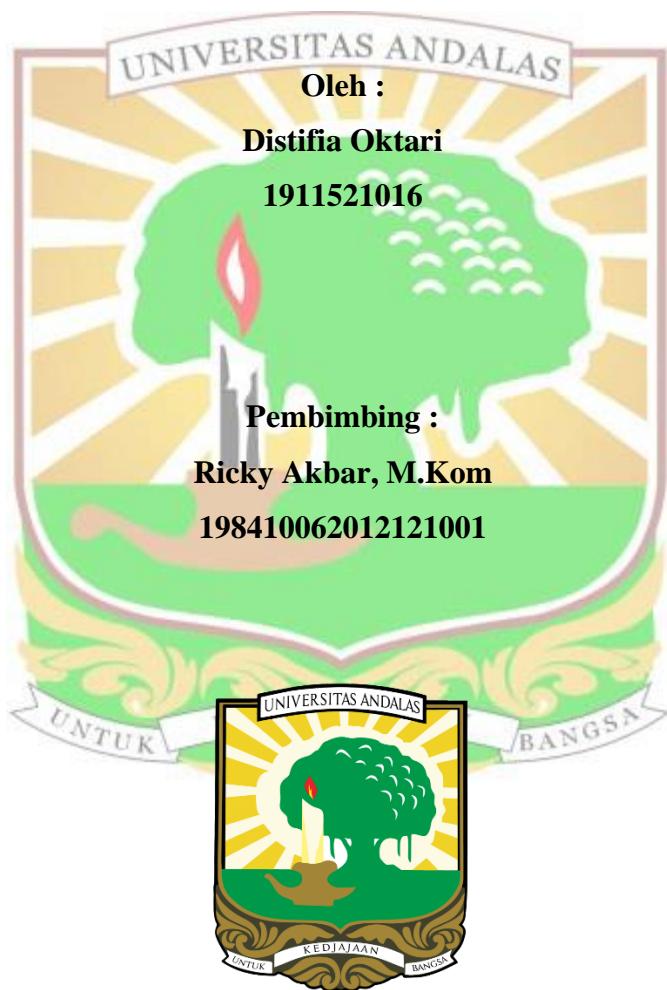


**PEMBANGUNAN APLIKASI DENGAN MODEL BISNIS *MARKEPLACE*  
UNTUK MEMBANTU PEMASARAN OLEH-OLEH MAKANAN  
PADA UMKM KOTA PAYAKUMBUH**

**TUGAS AKHIR**

Diajukan sebagai Salah Satu Syarat untuk Menyelesaikan Program Strata-I pada Departemen Sistem Informasi Fakultas Teknologi Informasi Universitas Andalas



**DEPARTEMEN SISTEM INFORMASI  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS  
2023**

## HALAMAN PENGESAHAN

### TUGAS AKHIR

#### PEMBANGUNAN APLIKASI DENGAN MODEL BISNIS *MARKEPLACE* UNTUK MEMBANTU PEMASARAN OLEH-OLEH MAKANAN PADA UMKM KOTA PAYAKUMBUH

Oleh

Distifia Oktari



Mengetahui,

Ketua Departemen Sistem Informasi

Husnil Kamil, M.T

NIP. 198201182008121002

## PERNYATAAN

Saya menyatakan bahwa laporan tugas akhir yang berjudul "**Pembangunan Aplikasi Dengan Model Bisnis Marketplace Untuk Membantu Pemasaran Oleh-Oleh Makanan Pada UMKM Kota Payakumbuh**" ini tidak terdapat karya yang pernah diajukan sebagai syarat menyelesaikan mata kuliah tugas akhir di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat saya yang pernah ditulis atau diterbitkan oleh orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.



## KATA PENGANTAR

Puji dan syukur penulis ucapkan atas kehadiran Allah SWT. karena berkat limpahan rahmat dan karunia-Nya penulis dapat menyelesaikan Laporan Tugas Akhir yang berjudul “Pembangunan Aplikasi Dengan Model Bisnis *Marketplace* Untuk Membantu Pemasaran Oleh-Oleh Makanan pada UMKM Kota Payakumbuh”. Pelaksanaan dan penulisan laporan tugas akhir ini bertujuan sebagai salah satu syarat untuk menyelesaikan Program Strata-I pada Departemen Sistem Informasi Fakultas Teknologi Informasi Universitas Andalas. Selama proses penggerjaan tugas akhir ini, penulis banyak menerima bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis mengucapkan rasa hormat dan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ricky Akbar, M.Kom. selaku pembimbing tugas akhir yang telah meluangkan waktu untuk memberikan bimbingan selama proses penggerjaan tugas akhir ini berlangsung.
2. Bapak Husnil Kamil, M.T. selaku ketua Departemen Sistem Informasi Universitas Andalas.
3. Seluruh Dosen Departemen Sistem Informasi Universitas Andalas yang tidak dapat penulis sebutkan satu per satu yang telah memberikan pelajaran, bimbingan serta masukan kepada penulis selama menempuh pendidikan.
4. Pihak UMKM daerah Payakumbuh Barat yang telah bersedia untuk diwawancara, dimintai data dan menjadikan UMKM nya menjadi objek penelitian.
5. Kedua orang tua dan kedua saudara penulis yang selalu memberikan dukungan dan senantiasa mendukung penulis dalam menyelesaikan tugas akhir ini.
6. *Partner* penulis yang telah menemani dan memberikan dukungan selama pembuatan tugas akhir ini.
7. Teman-teman penulis satu angkatan Sistem Informasi 19, terutama grup “Merger” yang selalu menemani, membantu dan memberi dukungan selama menghadapi masa perkuliahan ini.

Penulis menyadari bahwa dalam laporan tugas akhir ini masih jauh dari kata sempurna. Oleh karena itu, penulis berharap semua kritikan dan saran dari pembaca yang bersifat membangun dapat disampaikan melalui *email*: *distifiaoktari@gmail.com*. Semoga proposal ini dapat bermanfaat baik bagi penulis sendiri, maupun pembaca.

Padang, 18 September 2023



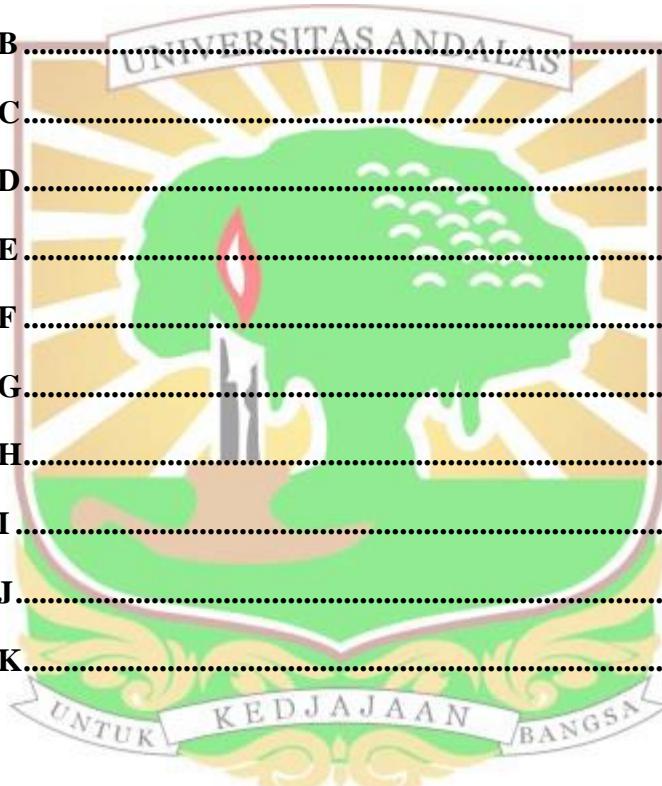
Distifia Oktari

## DAFTAR ISI

<b>HALAMAN PENGESAHAN.....</b>	<b>i</b>
<b>PERNYATAAN.....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR GAMBAR.....</b>	<b>viii</b>
<b>DAFTAR TABEL.....</b>	<b>x</b>
<b>ABSTRAK .....</b>	<b>xi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
1.6 Sistematika Penulisan.....	5
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>7</b>
2.1 <i>Marketplace</i> .....	7
2.2 Pemasaran.....	9
2.3 Usaha Mikro Kecil Menengah (UMKM) .....	10
2.4 Usaha Oleh-Oleh Makanan .....	11
2.5 Alat Pemodelan Aplikasi.....	11
2.5.1 <i>Business Process Model Notation (BPMN)</i> .....	11
2.5.2 <i>Use Case Diagram</i> .....	13
2.5.3 <i>Sequence Diagram</i> .....	15
2.5.4 <i>Entity Relationship Diagram (ERD)</i> .....	15
2.6 Perangkat Lunak Pendukung.....	17
2.6.1 <i>Database MySQL</i> .....	17

2.6.2 XAMPP .....	18
2.6.3 PHP.....	18
2.6.4 <i>Framework</i> Laravel .....	19
2.7 <i>Blackbox Testing</i> .....	21
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>23</b>
3.1 Objek Penelitian .....	23
3.2 Metode Pengumpulan Data .....	23
3.3 Metode Pengembangan Sistem .....	24
3.4 <i>Flowchart</i> Penelitian .....	26
<b>BAB IV ANALISIS SISTEM DAN PERANCANGAN .....</b>	<b>29</b>
4.1 Analisis Sistem .....	29
4.1.1 Sistem yang Berjalan .....	29
4.1.2 Sistem yang Diusulkan .....	34
4.1.3 Analisis Kebutuhan Fungsional .....	37
4.1.4 <i>Use Case Diagram</i> .....	39
4.1.5 Deskripsi Tugas Aktor .....	40
4.1.6 <i>Use Case Scenario</i> .....	43
4.1.7 <i>Sequence Diagram</i> .....	46
4.1.8 <i>Class Analysis</i> .....	49
4.2 Perancangan Sistem.....	52
4.2.1 Perancangan <i>Database</i> .....	52
4.2.2 Struktur Tabel <i>Database</i> .....	54
4.2.3 Arsitektur Aplikasi.....	56
4.2.4 <i>Class Diagram</i> .....	56
4.2.5 Perancangan Antarmuka Aplikasi .....	58
<b>BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM.....</b>	<b>61</b>
5.1 Implementasi Sistem .....	61
5.1.1 Pengkodean Program .....	62
5.1.2 Implementasi Antarmuka Aplikasi .....	72
5.2 Pengujian Sistem .....	78

5.2.1 Fokus Pengujian.....	79
5.2.2 Kasus Hasil Pengujian .....	80
5.3 Analisis Hasil Pengujian .....	91
<b>BAB VI PENUTUP .....</b>	<b>93</b>
6.1 Kesimpulan.....	93
6.2 Saran.....	93
<b>DAFTAR PUSTAKA .....</b>	<b>95</b>
<b>LAMPIRAN A .....</b>	<b>99</b>
<b>LAMPIRAN B .....</b>	<b>108</b>
<b>LAMPIRAN C .....</b>	<b>124</b>
<b>LAMPIRAN D .....</b>	<b>143</b>
<b>LAMPIRAN E .....</b>	<b>155</b>
<b>LAMPIRAN F .....</b>	<b>161</b>
<b>LAMPIRAN G .....</b>	<b>183</b>
<b>LAMPIRAN H .....</b>	<b>346</b>
<b>LAMPIRAN I .....</b>	<b>366</b>
<b>LAMPIRAN J .....</b>	<b>384</b>
<b>LAMPIRAN K .....</b>	<b>431</b>



## DAFTAR GAMBAR

Gambar 2. 1 Proses Bisnis <i>Marketplace</i> .....	8
Gambar 2. 2 Notasi yang Digunakan pada <i>Swimlane</i> .....	12
Gambar 2. 3 Notasi yang Ada pada <i>Connecting</i> .....	12
Gambar 2. 4 Notasi yang Ada pada <i>Artifacts</i> .....	13
Gambar 2. 5 Notasi yang Ada pada <i>Flow Object</i> .....	13
Gambar 2. 6 Gambaran <i>Blackbox Testing</i> .....	21
Gambar 3. 1 Siklus Pengembangan Metode <i>Waterfall</i> .....	24
Gambar 3. 2 <i>Flowchart</i> Penelitian.....	26
Gambar 4. 1 Proses Pemasaran yang Berjalan.....	30
Gambar 4. 2 Proses Transaksi Penjualan Produk Eceran yang Berjalan .....	31
Gambar 4. 3 Proses Transaksi Penjualan Produk Grosir yang Berjalan .....	32
Gambar 4. 4 Proses Pengelolaan Produk yang Berjalan.....	33
Gambar 4. 5 Proses Pemasaran Produk yang Diusulkan .....	34
Gambar 4. 6 Proses Transaksi Produk yang Diusulkan .....	35
Gambar 4. 7 Proses Pengelolaan Produk yang Diusulkan .....	37
Gambar 4. 8 <i>Use Case Diagram</i> Aplikasi .....	39
Gambar 4. 9 <i>Sequence Diagram</i> Kelola Perizinan UMKM .....	47
Gambar 4. 10 <i>Sequence Diagram</i> Melakukan Pembelian Produk.....	48
Gambar 4. 11 <i>Sequence Diagram</i> Mengelola Transaksi .....	49
Gambar 4. 12 <i>Class Analysis</i> Kelola Perizinan UMKM.....	50
Gambar 4. 13 <i>Class Analysis</i> Melakukan Pembelian Produk .....	51
Gambar 4. 14 <i>Class Analysis</i> Mengelola Transaksi.....	51
Gambar 4. 15 Perancangan <i>Database</i> .....	53
Gambar 4. 16 Arsitektur Aplikasi MVC .....	56
Gambar 4. 17 <i>Class Diagram</i> Aplikasi .....	57
Gambar 4. 18 Perancangan Antarmuka Data UMKM .....	58
Gambar 4. 19 Perancangan Antarmuka <i>Home</i> .....	59
Gambar 4. 20 Perancangan Antarmuka Keranjang.....	60
Gambar 5. 1 Kode Program <i>Routing</i> Aplikasi.....	62
Gambar 5. 2 Kode Program <i>Controller</i> Aplikasi.....	65

Gambar 5. 3 Kode Program <i>Model</i> Aplikasi.....	71
Gambar 5. 4 Kode Program <i>View</i> Aplikasi.....	72
Gambar 5. 5 Implementasi Halaman <i>Landing Page</i> .....	73
Gambar 5. 6 Implementasi Antarmuka Halaman <i>Login</i> .....	74
Gambar 5. 7 Implementasi Antarmuka Halaman Awal Pembeli .....	75
Gambar 5. 8 Implementasi Antarmuka Halaman Awal Penjual .....	76
Gambar 5. 9 Implementasi Antarmuka Halaman Awal Admin .....	77
Gambar 5. 10 Pengujian Masukan Penambahan UMKM (Benar).....	81
Gambar 5. 11 Pengujian Keluaran Penambahan UMKM (Benar) .....	82
Gambar 5. 12 Pengecekan Pengujian Penambahan UMKM (Benar).....	83
Gambar 5. 13 <i>Output</i> Menambahkan UMKM (Alternatif) .....	83
Gambar 5. 14 Pengujian Masukan Penambahan Produk (Benar) .....	85
Gambar 5. 15 Pengujian Keluaran Penambahan Produk (Benar) .....	86
Gambar 5. 16 Pengecekan Pengujian Menambahkan Produk (Benar) .....	86
Gambar 5. 17 <i>Output</i> Menambahkan Produk (Alternatif) .....	87
Gambar 5. 18 Pengujian Masukan Menambahkan Stok Produk (Benar).....	88
Gambar 5. 19 Pengujian Keluaran Menambahkan Stok Produk.....	89
Gambar 5. 20 <i>Output</i> Menambahkan Stok Produk (Alternatif) .....	90
Gambar 5. 21 <i>Output</i> Menambahkan Stok Produk (Alternatif) .....	91



## DAFTAR TABEL

Tabel 2. 1 Simbol <i>Sequence Diagram</i> .....	15
Tabel 2. 2 Komponen ERD.....	16
Tabel 2. 3 Fungsi pada MySQL .....	17
Tabel 4. 1 Deskripsi Tugas Aktor.....	40
Tabel 4. 2 <i>Use Case Scenario</i> Kelola Perizinan UMKM.....	44
Tabel 4. 3 <i>Use Case Scenario</i> Melakukan Pembelian Produk.....	45
Tabel 4. 4 <i>Use Case Scenario</i> Mengelola Transaksi.....	46
Tabel 4. 5 Struktur Tabel Pembeli .....	54
Tabel 4. 6 Struktur Tabel UMKM.....	54
Tabel 4. 7 Struktur Tabel Produk .....	55
Tabel 4. 8 Struktur Tabel Pembelian.....	55
Tabel 5. 1 Fokus Pengujian.....	79
Tabel 5. 2 Pengujian Menambahkan UMKM (Benar) .....	81
Tabel 5. 3 Pengujian Menambahkan UMKM (Alternatif).....	83
Tabel 5. 4 Pengujian Menambahkan Produk (Benar) .....	84
Tabel 5. 5 Pengujian Menambahkan Produk (Alternatif) .....	87
Tabel 5. 6 Pengujian Menambahkan Stok Produk (Benar) .....	88
Tabel 5. 7 Pengujian Menambahkan Stok Produk (Alternatif) .....	89
Tabel 5. 8 Pengujian Menambahkan Stok Produk (Alternatif) .....	90
Tabel 5. 9 Analisis Hasil Pengujian .....	91

## ABSTRAK

*Usaha Mikro Kecil Menengah (UMKM) adalah salah satu usaha yang saat ini berkembang di masyarakat. Masyarakat umumnya memanfaatkan potensi yang ada di daerah mereka untuk mengembangkannya menjadi sebuah usaha. Salah satu daerah yang memanfaatkan potensinya yaitu Kota Payakumbuh. Di daerah ini banyak sekali usaha oleh-oleh makanan yang dikembangkan langsung oleh masyarakat sekitar. Untuk saat ini proses pemasaran masih dilakukan secara konvensional, sedangkan produk olahan makanan pada daerah ini merupakan produk yang berpotensi memiliki harga jual yang tinggi dan merupakan salah satu produk makanan yang banyak diminati oleh masyarakat luas. Pemasaran yang dilakukan secara konvensional ini berdampak pada penjualan mereka, penjualan hanya dilakukan di daerah Payakumbuh dan sekitarnya sehingga dibutuhkanlah sebuah cara yang bisa melakukan pemasaran seluas-luasnya. Cara tersebut adalah dengan membangun aplikasi berbasis model bisnis marketplace pada UMKM oleh-oleh makanan di Kota Payakumbuh. Aplikasi dengan model bisnis marketplace merupakan aplikasi yang bisa digunakan sebagai platform bagi pelaku usaha untuk mempromosikan produknya dan bisa juga untuk melakukan transaksi penjualan dan pembelian produk secara online. Metode pengumpulan data yang digunakan berupa observasi, wawancara dan studi literatur, serta metode pengembangan sistem yang digunakan yaitu metode waterfall. Tahapan dari metode waterfall ini meliputi requirement, design, implementation, verification. Hasil yang diperoleh dari penelitian ini yaitu adanya sebuah aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan di Kota Payakumbuh yang dibangun menggunakan bahasa pemrograman php dengan framework laravel. Dengan adanya aplikasi ini menjadi sebuah solusi sebagai sarana promosi bagi para pelaku usaha UMKM serta mempermudah proses transaksi penjualan dan pembelian jarak jauh pada UMKM oleh-oleh makanan di Kota Payakumbuh.*

**Kata kunci:** UMKM, pemasaran, marketplace

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Sektor kewirausahaan menjadi salah satu sektor yang sangat penting di kalangan masyarakat saat ini, sektor ini menjadi salah satu sektor utama dalam menghasilkan pendapatan untuk masyarakat. Salah satu komponen dari sektor ini yaitu adanya Usaha Mikro Kecil dan Menengah. Usaha Mikro Kecil dan Menengah atau yang biasanya dikenal dengan UMKM adalah sebuah usaha yang yang didirikan dan dibentuk oleh perseorangan atau kelompok tanpa terikat dengan perusahaan lainnya. UMKM memiliki daya tahan dan fleksibilitas yang lebih baik terhadap krisis ekonomi sehingga menjadikan UMKM salah satu bisnis yang bergerak paling banyak dalam kepentingan masyarakat dan menjadikan UMKM menjadi sektor yang penting dalam perkembangan perekonomian Indonesia (Hamza & Agustien, 2019).

Kota Payakumbuh merupakan salah satu kota yang berada di provinsi Sumatra Barat dan merupakan kota yang menjadi pengembangan kawasan industri khususnya industri makanan. Pada kota ini terdapat sentra pusat oleh-oleh, dimana pada lokasi ini terdapat banyak sekali unit usaha oleh-oleh makanan khas minang. Salah satu contoh dari oleh-oleh makanannya yaitu produk sanjai. Sanjai merupakan ubi singkong yang diolah kedalam beberapa olahan yang menjadikannya makanan khas oleh-oleh provinsi Sumatra Barat dan Payakumbuh merupakan kota tempat produksi sanjai tersebut sehingga menjadikannya banyak sekali para pelaku usaha yang membuka usaha sanjai tersebut dari yang masih usaha kecil-kecilan hingga usaha yang sudah sangat besar. Pada daerah ini tidak hanya menjual sanjai, para pelaku usaha tersebut mengembangkan usaha mereka dengan menjual produk lainnya. Sehingga menjadikan UMKM ini sebagai penyerap banyak tenaga kerja dan memiliki kontribusi besar dalam pendapatan masyarakat sekitar.

Walaupun usaha ini sudah banyak dan memiliki kontribusi yang besar namun tetap saja masih memiliki kendala yang harus dihadapi. Berdasarkan wawancara yang telah dilakukan terhadap pelaku usaha UMKM di pusat sentra oleh-oleh Payakumbuh dengan mendatangi beberapa UMKM tersebut, kendala yang dihadapi

oleh para pelaku usaha tersebut sebagian besar terdapat dibagian promosi atau pemasaran dari produk yang mereka hasilkan. Sebagian besar dari pelaku usaha UMKM oleh-oleh makanan ini, pemasarannya masih dilakukan secara konvensional, tidak menggunakan *platform digital* apa pun. Mereka melakukan pemasaran hanya melalui spanduk yang dipasang di depan toko tempat mereka berjualan dan melalui promosi mulut ke mulut. Walaupun ada juga beberapa pelaku usaha yang sudah menggunakan media sosial seperti *instagram* dan *facebook* sebagai sarana promosi tetapi hal tersebut masih dirasa kurang efektif dikarenakan adanya keterbatasan pemahaman penggunaan teknologi oleh para pelaku usaha tersebut.

*Platform digital* merupakan sebuah tempat dimana individu atau kelompok dapat berkumpul untuk berbagi informasi, berdagang atau menawarkan suatu produk secara *online*, adanya platform digital ini dapat mengubah mekanisme bisnis yang awalnya bersifat konvensional menjadi bisnis yang menggunakan teknologi internet dan jaringan data (Natania & Dwijayanti, 2023). *Platform digital* yang digunakan dalam penelitian ini yaitu aplikasi berbasis *website* yang memanfaatkan model bisnis *marketplace* dalam penerapannya. *Marketplace* merupakan model bisnis yang memungkinkan aktivitas jual beli antara penjual dan pembeli secara *online* tanpa ada batas ruang, jarak dan waktu sehingga dirancang untuk meminimalisir proses bisnis yang kompleks sehingga tercipta efisiensi dan efektifitas (Yustiani & Yunanto, 2017).

Berdasarkan data statistika *market insight*, pada tahun 2022 jumlah pengguna lokal pasar *online* termasuk *e-commerce* dan *marketplace* didalamnya mencapai 178,94 juta, jumlah tersebut meningkat 12,79% dibandingkan pada tahun sebelumnya yang sebanyak 158,65 juta pengguna. Bank Indonesia mencatat nilai transaksi sebesar Rp476,3 triliun pada tahun 2022 tersebut. Dari data tersebut dapat disimpulkan bahwa perkembangan aplikasi berbasis model bisnis *marketplace* ini sangat pesat dan banyak pelaku usaha yang menggunakan sarana ini sebagai media promosi untuk kemajuan dan pembangunan bisnisnya.

Produk oleh-oleh makanan khas minang yang banyak terdapat di Kota Payakumbuh ini merupakan produk yang berpotensi memiliki harga jual yang tinggi dan produk yang sangat diminati oleh masyarakat luas sehingga seharusnya

dapat dipasarkan seluas-luasnya. Tetapi cara promosi mereka yang bersifat konvensional membawa dampak terhadap penjualan mereka, penjualan masih dilakukan ke daerah Payakumbuh dan sekitarnya. Terkadang ada di beberapa usaha UMKM yang produknya tidak habis terjual dan telah memasuki masa kadaluarsa sehingga harus dibuang ataupun diolah untuk dijadikan pakan ternak dan kemudian menimbulkan kerugian pada periode produksi UMKM tersebut. Dari hasil wawancara sebagian besar pelaku usaha belum ada yang menggunakan *platform digital website* menjadi salah satu sarana untuk melakukan proses penjualan produk mereka.

Untuk mengatasi masalah diatas, solusi yang dapat dilakukan yaitu diperlukannya sebuah *platform digital* untuk mempromosikan produk UMKM mereka agar usaha tersebut memiliki jangkauan yang lebih luas dan mampu menembus pasar ke seluruh wilayah yang ada di Indonesia. Salah satu *platform* yang bisa digunakan untuk melakukan promosi seluas-luasnya yaitu melalui aplikasi berbasiskan *website* dengan model bisnis *marketplace*.

Dalam mendukung penelitian ini, beberapa penelitian terdahulu dapat dijadikan referensi, diantaranya penelitian yang dilakukan oleh (Setiyawati, 2020) dengan judul “*Marketplace Larisumkm.com Sebagai Alat Peningkatan Promosi UMKM di Kota Salatiga*” menghasilkan aplikasi *website* berbasis *marketplace*. Penelitian ini bertujuan untuk membantu mempromosikan produk UMKM Kota Salatiga dan mengakomodasikan proses jual beli yang lebih efisien sehingga proses transaksi penjualan dan pembelian dapat dilakukan secara *online*. Kemudian penelitian yang dilakukan oleh (Ahsyar et al., 2020) dengan judul “*E-Marketplace Media Pengembangan Promosi Usaha Mikro Kecil dan Menengah Dinas Koperasi UMKM Kota Pekanbaru*” menghasilkan aplikasi e-*marketplace* berbasis *website* dengan menggunakan metode pengembangan sistem waterfall serta pendekatan analisis dan desain Object-Oriented Analysis and Design (OOAD) yang diuji menggunakan metode UAT. Penelitian ini bertujuan untuk mempromosikan produk dan memudahkan pihak dinas dalam mengelola data UMKM. Selanjutnya penelitian yang dilakukan oleh (Aulami & Ariani, 2022) dengan judul “*Aplikasi E-Marketplace Usaha Mikro Kecil dan Menengah (UMKM) berbasis mobile pada studi kasus Kabupaten Pesawaran*” menghasilkan aplikasi e-*marketplace* berbasis

website dan mobile dengan tujuan untuk mempermudah pengelolaan penjualan produk hasil olahan UMKM di daerah Kabupaten Pesawaran. Dari ketiga penilitian tersebut tentang pembuatan aplikasi dengan model bisnis *marketplace* untuk UMKM, terdapat sebuah kesamaan dalam hal tujuan yaitu sama-sama merancang dan membangun sistem penjualan dan pembelian produk hasil olahan UMKM serta untuk mempromosikan produk hasil UMKM pada daerah tersebut. Pada penelitian ini juga dibangun aplikasi *e-marketplace* pada Kota Payakumbuh untuk mempromosikan UMKM oleh-oleh makanan yang ada pada daerah ini menggunakan metode pengembangan sistem *waterfall* dan pendekatan analisa dan desain *Object-Oriented Analysis and Design* (OOAD). Perbedaan penelitian ini dengan ketiga penelitian sebelumnya yaitu pada penelitian ini terdapat penambahan fitur chat dan fitur laporan transaksi serta pengembangan lebih lanjut mengenai transaksi penjualan dan pembelian pada aplikasi.

Berdasarkan uraian diatas mendorong penulis untuk melakukan penelitian dengan merancang sebuah aplikasi berbasis *website* dengan model bisnis *marketplace* menggunakan bahasa pemrograman PHP dan *framework* laravel untuk UMKM oleh-oleh makanan di Kota Payakumbuh, Provinsi Sumatra Barat sebagai wadah untuk mempromosikan produknya secara *online*. Aplikasi ini bertujuan untuk memudahkan para pelaku usaha untuk menjual produk secara lebih luas lagi dan juga mempermudah para konsumen untuk membeli produk UMKM tersebut walaupun tidak datang secara langsung ke tempatnya. Penelitian ini berjudul “Pembangunan Aplikasi Dengan Model Bisnis *Marketplace* Untuk Membantu Pemasaran Oleh-Oleh Makanan Pada UMKM Kota Payakumbuh”.

## 1.2 Rumusan Masalah

Berdasarkan uraian masalah yang telah dijelaskan sebelumnya, didapatkanlah rumusan masalah pada penelitian ini yaitu bagaimana membangun aplikasi dengan model bisnis *marketplace* untuk membantu pemasaran oleh-oleh makanan pada UMKM di Kota Payakumbuh.

## 1.3 Batasan Masalah

Dalam penelitian ini, batasan masalah yang ditetapkan oleh penulis yaitu :

1. Dalam pembangunan aplikasi ini terbatas pada pengelolaan produk oleh pelaku usaha, penjualan produk, pengecekan pembayaran serta pembelian produk oleh pembeli tidak sampai ke pengembalian produk.
2. Dalam aplikasi ini para pelaku usaha terbatas pada pelaku usaha UMKM oleh-oleh makanan.
3. Model bisnis yang digunakan yaitu berbasis model bisnis *marketplace* sehingga setiap pelaku usaha bisa mendaftarkan UMKM nya melalui aplikasi.
4. Aplikasi yang dibangun berbasis web dengan menggunakan bahasa pemrograman PHP dan *framework* Laravel.
5. Aplikasi ini menggunakan *database* MySQL untuk menyimpan data.
6. Pengujian dilakukan untuk mengecek kesesuaian fungsional dengan kebutuhan yang diharapkan.

#### **1.4 Tujuan Penelitian**

Tujuan dari penelitian ini adalah membangun aplikasi dengan model bisnis *marketplace* yang dapat membantu para pelaku usaha UMKM untuk mempromosikan produknya, melakukan penjualan secara *online* dan menjangkau konsumen seluas-luasnya serta membantu para konsumen yang ingin membeli produk, tidak perlu datang langsung ke tempat usahanya sehingga dapat meningkatkan efektivitas dan efisiensi penjualan dan pembelian produk UMKM.

#### **1.5 Manfaat Penelitian**

Manfaat yang didapatkan dari penelitian ini yaitu agar aplikasi ini bisa menjadi wadah untuk mempromosikan produk dari para pelaku usaha UMKM dan juga dengan adanya aplikasi ini promosi bisa dilakukan secara luas tanpa hambatan apapun dan mempermudah pembeli untuk melakukan transaksi ketika ingin membeli produknya dari jarak jauh.

#### **1.6 Sistematika Penulisan**

Sistematika penulisan laporan penelitian tugas akhir ini dibagi menjadi enam bab, sebagai berikut :

## **BAB I: PENDAHULUAN**

Bab ini berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan laporan.

## **BAB II: TINJAUAN PUSTAKA**

Bab ini berisi landasan teori dan informasi pendukung yang diterapkan dalam penelitian.

## **BAB III: METODOLOGI PENELITIAN**

Bab ini menjelaskan mengenai objek penelitian, metode pengumpulan data, metode pengembangan sistem, dan *flowchart* penelitian pada UMKM oleh-oleh makanan yang ada di Kota Payakumbuh.

## **BAB IV: ANALISIS SISTEM DAN PERANCANGAN**

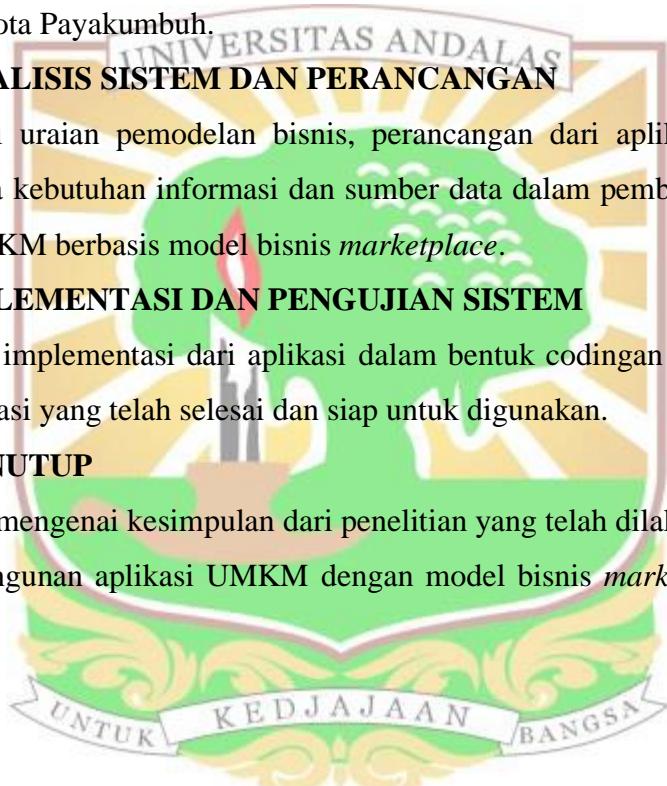
Bab ini berisi uraian pemodelan bisnis, perancangan dari aplikasi yang akan dibangun serta kebutuhan informasi dan sumber data dalam pembangunan sistem informasi UMKM berbasis model bisnis *marketplace*.

## **BAB V: IMPLEMENTASI DAN PENGUJIAN SISTEM**

Bab ini berisi implementasi dari aplikasi dalam bentuk codingan serta pengujian terhadap aplikasi yang telah selesai dan siap untuk digunakan.

## **BAB VI: PENUTUP**

Bab ini berisi mengenai kesimpulan dari penelitian yang telah dilakukan dan saran dalam pembangunan aplikasi UMKM dengan model bisnis *marketplace* di Kota Payakumbuh.



## **BAB II**

### **TINJAUAN PUSTAKA**

Pada bab ini menjelaskan mengenai teori-teori terkait dengan penelitian. Teori-teori ini berasal dari jurnal, buku serta artikel-artikel ilmiah yang berguna untuk membantu peneliti dalam menjalankan penelitian berdasarkan teori yang benar. Terdiri atas konsep dasar *marketplace*, pemasaran, UMKM, oleh-oleh makanan, pemodelan aplikasi hingga penjelasan tentang perangkat lunak pendukung.

#### **2.1 Marketplace**

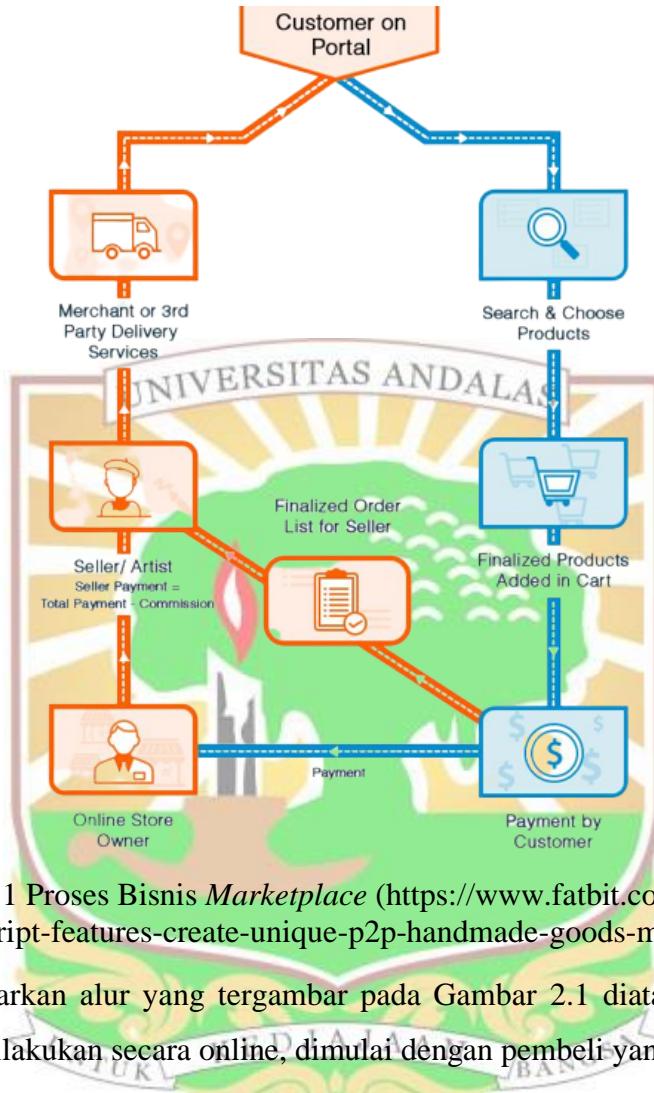
*Marketplace* merupakan salah satu model bisnis yang sedang berkembang pesat akibat penggunaan teknologi yang semakin meningkat tajam. Dengan menggunakan *marketplace* proses bisnis yang kompleks akan menjadi lebih efisien dan efektif. Konsep dari *marketplace* ini adalah para penjual dan pembeli dapat melakukan aktifitas jual beli dengan mudah dan praktis tanpa adanya batasan ruang dan waktu. Hal yang memengaruhi mengapa proses bisnis *marketplace* ini menjadi lebih efektif dikarenakan adanya fasilitas transaksi yang mempertemukan penjual dan pembeli tanpa harus bertemu (Artaya & Purworusmiardi, 2019).

*E-marketplace* merupakan salah satu pengembangan *e-commerce* dimana *e-marketplace* menjadi media perantara yang mempertemukan antara penjual dan pembeli (Naovarat & Juntongjin, 2015). *E-marketplace* merupakan segala usaha yang dilakukan untuk melakukan pemasaran produk atau jasa melalui media internet atau jaringan www, dapat juga diartikan sebagai tempat atau wadah dalam media promosi (Sadgotra & Saputra, 2013).

Berikut diberikan beberapa keuntungan menggunakan model bisnis *marketplace* :

- a. Para pelaku bisnis atau usaha tidak perlu mengeluarkan biaya yang besar untuk membangun sistem karena telah disediakan oleh pihak *marketplace*.
- b. Para pelaku bisnis tidak perlu melakukan perbaikan secara berkala.
- c. Dengan menggunakan *marketplace* sudah termasuk kedalam ajang promosi.
- d. Memperkecil biaya transaksi.

Dalam menjalankan sebuah aplikasi dengan model bisnis *marketplace*, tentu saja memiliki sebuah proses bisnis. Proses bisnis *marketplace* dapat dilihat pada Gambar 2.1 berikut



Gambar 2. 1 Proses Bisnis *Marketplace* (<https://www.fatbit.com/fab/clone-worthy-script-features-create-unique-p2p-handmade-goods-marketplace/>)

Berdasarkan alur yang tergambar pada Gambar 2.1 diatas, proses bisnis *marketplace* dilakukan secara online, dimulai dengan pembeli yang bisa membuka aplikasi dan melihat lihat produk yang tersedia, lalu setelah itu pembeli akan mencari informasi mengenai produk yang akan dibeli dan melanjutkannya ke tahap transaksi. Ketika masuk ke proses transaksi, pembeli akan melakukan pembayaran dan menunggu proses pengiriman dari penjual. Sedangkan untuk alur penjual, penjual dimulai dengan mendaftarkan akun toko nya dan memasukkan produk beserta deskripsinya dengan lengkap, setelah itu jika ada penjualan yang terjadi maka pembeli akan mengecek pembayaran dan melakukan pengiriman.

## 2.2 Pemasaran

Pemasaran adalah sebuah proses dimana organisasi dapat menciptakan, menawarkan dan mempertukarkan produk yang bernilai kepada pihak lain mencakup kegiatan dalam penyampaian produk atau jasa dari pihak produsen ke pihak konsumen (Sudarsono, n.d., 2020). Pemasaran merupakan sebuah kegiatan dalam bentuk upaya promosi terhadap suatu produk, layanan, jasa yang dimiliki oleh sebuah perusahaan, menyampaikan nilai dari sebuah produk atau jasa agar sampai ke tangan konsumen dengan baik dan bagaimana hal tersebut dapat memberikan kepuasan kepada para konsumen (Fauzan et al., 2023).

Pemasaran adalah fungsi organisasi dan rangkaian proses untuk menciptakan, mengkomunikasikan dan memberikan nilai kepada pelanggan untuk mengelola hubungan pelanggan dengan cara yang menguntungkan organisasi dan pihak-pihak berkepentingan, pemasaran memiliki konsep bagaimana organisasi memuaskan kebutuhan pelanggan (Dr. H. Muhammad Yusuf Saleh et al., 2019). Sehingga dari berbagai pengertian pemasaran diatas, dapat disimpulkan bahwa pemasaran adalah rangkaian dari proses bisnis yang dilakukan oleh organisasi atau perusahaan untuk mengenalkan produk mereka pelanggan sehingga tujuan mereka tercapai.

Pemasaran digital merupakan pemasaran produk atau jasa yang dilakukan melalui media digital atau media internet, pemasaran dalam internet bukan hanya sekedar untuk meningkatkan penjualan tetapi juga sebagai promosi produk dan jasa, *branding*, dan membina hubungan pelanggan dengan jangkauan yang lebih luas secara efisien, efektif dan relevan (Bambang & Abdur, 2019).

Ada banyak cara pemasaran yang bisa dilakukan oleh para pelaku usaha, salah satunya yaitu pemasaran digital. Pemasaran digital dilakukan melalui sistem komputer online yang interaktif, yang menghubungkan pembeli dan penjual secara elektronik, dan merupakan bagian terpenting dari proses perdagangan secara elektronik dengan menggunakan *website*, *blog* atau media sosial. Pemasaran meliputi aktivitas – aktivitas yang berkaitan dengan penjualan, pengiklanan, promosi serta penentuan harga. Strategi pemasaran *online* atau sering disebut dengan *online marketing strategy* merupakan segala usaha (bisnis) yang dilakukan untuk melakukan pemasaran suatu produk atau jasa melalui atau menggunakan

media online, yakni media internet. Walaupun mengalami perubahan, pemasaran tidak bisa melepaskan diri dari tiga komponen yang selalu menyertai, yaitu konsumen, kompetitor dan perusahaan (Setiawati, 2017).

### **2.3 Usaha Mikro Kecil Menengah (UMKM)**

Usaha Mikro Kecil dan Menengah atau yang biasa dikenal dengan UMKM adalah sebuah usaha yang didirikan oleh perseorangan atau beberapa orang yang tidak terikat dengan perusahaan lainnya. Menurut UU no 20 tahun 2008 Bab I Pasal 1 menjelaskan pengertian dari UMKM yaitu :

- a. Usaha Mikro adalah usaha produktif milik orang perorangan dan/atau badan usaha perorangan yang memenuhi kriteria Usaha Mikro sebagaimana diatur dalam Undang-Undang ini.
- b. Usaha Kecil adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau bukan cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dari Usaha Menengah atau Usaha Besar yang memenuhi kriteria Usaha Kecil sebagaimana dimaksud dalam Undang-Undang ini.
- c. Usaha Menengah adalah usaha ekonomi produktif yang berdiri sendiri, yang dilakukan oleh orang perorangan atau badan usaha yang bukan merupakan anak perusahaan atau cabang perusahaan yang dimiliki, dikuasai, atau menjadi bagian baik langsung maupun tidak langsung dengan Usaha Kecil atau Usaha Besar dengan jumlah kekayaan bersih atau hasil penjualan tahunan sebagaimana diatur dalam Undang-Undang ini

Dengan ketentuan yang dimaksud dalam pasal tersebut adalah untuk usaha mikro maksimal penjualan sebanyak Rp 300.000.000,00 (tiga ratus juta rupiah), kemudian untuk usaha kecil maksimal penjualan sebanyak Rp 2.500.000.000,00 (dua miliar lima ratus juta rupiah) dan terakhir untuk usaha menengah maksimal penjualan sebanyak Rp 50.000.000.000,00 (lima puluh miliar rupiah).

## 2.4 Usaha Oleh-Oleh Makanan

Pada saat sekarang ini, berdasarkan data pada website resmi *Online Single Submission* (OSS) jumlah UMKM yang tersebar di seluruh Indonesia yaitu mencapai 8,71 juta unit hingga akhir tahun 2022 sehingga UMKM hampir menguasai sebagian besar dari perekonomian masyarakat dimana masyarakat membuka usaha berdasarkan potensi yang dimiliki ataupun berdasarkan sumber daya yang ada di sekitaran lingkungan mereka. Salah satu jenis UMKM yaitu UMKM produk oleh-oleh makanan khas minang yang banyak ditemui pada Kota Payakumbuh. Salah satu contoh produk olahan makanannya yaitu produk sanjai.

Sanjai merupakan salah satu olahan dari ubi singkong, salah satu makanan khas dari Provinsi Sumatra Barat. Kerupuk Sanjai adalah sejenis kerupuk dari singkong yang diparut tipis, dijemur hingga kering, lalu digoreng dan diberi bumbu, Kerupuk sanjai terdiri atas tiga rasa, yaitu kerupuk sanjai yang hanya diberi garam (kerupuk sanjai tawar), kerupuk sanjai yang diberikan gula merah (kerupuk sanjai saka) , dan kerupuk sanjai yang diberikan bumbu balado (kerupuk balado) (Fauzan & Jayanti, 2019). Rasanya yang enak dan renyah sehingga menjadikannya ciri khas dan oleh-oleh dari daerah ini bagi wisatawan yang berkunjung, hal itu membuat masyarakat sekitar daerah ini banyak yang membuka usaha oleh-oleh. Tidak hanya produk sanjai tetapi para pelaku usaha ini mengembangkan usahanya dengan menjual berbagai jenis produk lainnya. Usaha-usaha yang semakin berkembang inilah yang disebut dengan UMKM oleh-oleh makanan.

## 2.5 Alat Pemodelan Aplikasi

Dalam sub bab ini dijelaskan mengenai teori pendukung tentang pemodelan aplikasi yang akan dilakukan dalam penelitian, pemodelan dilakukan terdiri atas BPMN, *use case diagram*, *sequence diagram* dan *entity relationship diagram*.

### 2.5.1 Business Process Model Notation (BPMN)

(Dewi et al., 2017) menjelaskan bahwa BPMN adalah standar untuk memodelkan proses bisnis dan proses-proses *web services*. BPMN menggambarkan alur bisnis menggunakan notasi grafis yang saling berurutan dan menggunakan konsep *flowchart*. Salah satu *software* yang dapat digunakan dalam pembuatan BPMN yaitu bizagi, penggunaan bizagi dapat mempermudah pembuatan BPMN

karena bizagi menyediakan berbagai fitur dan elemen yang bisa digunakan dalam membuat BPMN serta dapat dipublikasikan kedalam beberapa bentuk format.

Diagram BPMN yang akan dibuat menggunakan bizagi memiliki beberapa elemen sebagai berikut :

a. *Swimlane*

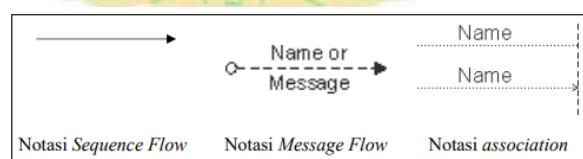
Elemen ini digunakan untuk mengatur dan memisahkan tugas-tugas dari proses yang sedang dibuat. Terdapat dua notasi, yaitu notasi *pool* dan notasi *lane*. Notasi *pool* digunakan sebagai *container* sedangkan notasi *lane* digunakan sebagai partisi dari suatu proses. Bentuk dari *swimlane* dapat dilihat pada Gambar 2.2 berikut



Gambar 2. 2 Notasi yang Digunakan pada *Swimlane* (Ismanto, Firman Hidayah, 2020)

b. *Connecting Object*

Elemen ini digunakan sebagai penghubung antar objek dalam arus proses bisnis. Terdapat tiga notasi yaitu *sequence flow* yang digunakan sebagai penghubung antar objek yang mengalir dalam satu proses, *message flow* yang digunakan sebagai penghubung antar objek yang mengalir dalam proses yang berbeda sedangkan *association* digunakan sebagai penghubung yang mengalir ke *artifact*. Bentuk dari *connecting object* dapat dilihat pada Gambar 2.3 berikut

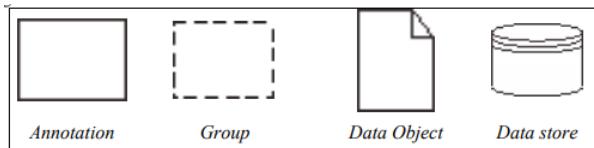


Gambar 2. 3 Notasi yang Ada pada *Connecting* (Ismanto, Firman Hidayah, 2020)

c. *Artifacts*

Elemen ini digunakan untuk memberikan informasi tambahan dalam suatu proses. Terdapat empat notasi dalam elemen ini yaitu *annotation* yang digunakan sebagai penjelasan dari suatu objek, *group* yang

digunakan untuk mengelompokkan beberapa objek, *data object* yang digunakan untuk aliran file yang diinput atau dihasilkan dari sebuah alur serta *data store* yang digunakan sebagai penanda sistem dan aplikasi yang dihasilkan dari sebuah alur. Bentuk dari *artifacts* dapat dilihat pada Gambar 2.4 berikut



Gambar 2. 4 Notasi yang Ada pada *Artifacts* (Ismanto, Firman Hidayah, 2020)

#### d. *Flow Object*

Elemen ini digunakan untuk menggambarkan objek yang mengalir pada suatu proses. Terdapat tiga notasi yang digunakan pada elemen ini yaitu *event* yang merupakan sebuah kejadian yang bersifat pasif, *activity* yang merupakan kegiatan yang bersifat aktif serta *gateway* yang merupakan pemecah dari beberapa aktifitas. Bentuk dari *flow object* dapat dilihat pada Gambar 2.5 berikut



Gambar 2. 5 Notasi yang Ada pada *Flow Object* (Ismanto, Firman Hidayah, 2020)

### 2.5.2 *Use Case Diagram*

*Use case* merupakan deskripsi fungsi dari sebuah sistem dari perspektif atau sudut pandang para pengguna sistem (Setiyani, 2021). *Use case diagram* digunakan untuk menggambarkan alur interaksi secara keseluruhan yang akan dilakukan oleh sebuah aktor didalam sebuah sistem yang akan dibuat.

*Use case diagram* juga digunakan untuk mengetahui rancangan fungsional apa saja yang akan berjalan didalam sistem dan aktor siapa saja yang berhak untuk mengakses fungsional tersebut.

Berikut bagian-bagian yang terdapat di dalam *use case diagram* :

a. *System*

Bagian ini merupakan batasan sistem dalam sebuah relasi dengan aktor-aktor yang berperan didalamnya dan fungsional yang dibutuhkan dalam sebuah sistem. Digambarkan dengan segi empat yang membatasi semua use case yang saling berinteraksi dalam satu sistem.

b. *Actor*

Bagian ini merupakan sebuah entitas yang berasal dari luar sistem yang mengendalikan sebuah fungsional. Bagian ini dapat berupa manusia, sistem atau perangkat yang menggunakan sistem tersebut.

c. *Use Case*

Bagian ini merupakan fitur kunci dari sebuah sistem. Setiap *use case* berisi fungsional atau perintah yang akan dijalankan oleh sistem tersebut, fungsional tersebut diletakkan didalam sebuah notasi yang berbentuk *elips*.

d. *Association*

Bagian ini menggambarkan interaksi yang terjadi antar setiap aktor tertentu dengan setiap *use case* tertentu pula. Digambarkan dengan garis antara aktor dengan use case yang saling berkaitan. Asosiasi berbentuk satu garis anak panah jika komunikasi yang dilakukan antar aktor tersebut satu arah tetapi jika komunikasi dilakukan dua arah maka digambarkan dengan tanpa anak panah.

e. *Dependency*

Bagian ini mengidentifikasi hubungan antara dua *use case* ketika satu *use case* memanggil satu *use case* lainnya. Terdapat dua *dependency* yaitu *include* dan *extend*. *Include* digunakan ketika satu *use case* menjadi syarat untuk terjadinya *use case* lainnya sedangkan *extend* digunakan ketika *use case* tersebut merupakan *use case* tambahan dari sebuah *use case*, dimana *use case* induk dapat berdiri sendiri tanpa *use case* tambahan tersebut.

### 2.5.3 Sequence Diagram

*Sequence diagram* adalah sebuah diagram yang dirancang sebagai gambaran dari alur interaksi antar objek yang harus disesuaikan dengan isi dari *use case* yang telah dibuat sebelumnya (Seidl et al., 2015).

*Sequence diagram* ini menggambarkan jenis interaksi urutan dari sekumpulan objek yang bekerja secara bersamaan, merincikan interaksi yang terjadi di dalam sebuah sistem, interaksi antar objek dalam konteks kolaborasi.

Berikut beberapa notasi yang terdapat di dalam *sequence diagram* yang dapat dilihat pada tabel 2.1

Tabel 2. 1 Simbol *Sequence Diagram* (Purwanto, 2014)

Gambar	Nama	Keterangan
	<i>Actor</i>	Aktor yang akan berinteraksi di dalam sistem
	<i>Entity class</i>	Menggambarkan hubungan yang akan dilakukan
	<i>Boundary class</i>	Menggambarkan sebuah gambaran dari form
	<i>Control class</i>	Penghubung antara boundary dan tabel
	<i>A focus of control &amp; a life line</i>	Tempat mulai dan berakhirnya sebuah pesan
	<i>A message</i>	Pengiriman pesan

### 2.5.4 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data

relasional (Yanto, 2016). Tujuan dari pembuatan ERD ini adalah agar mempermudah dalam memahami dan merancang database.

Berikut beberapa komponen yang terdapat di dalam ERD yang dapat dilihat pada tabel 2.2

Tabel 2.2 Komponen ERD (D. Edi and S. Betshani, 2009)

Simbol	Nama	Keterangan
	Entitas	Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain.
	Relasi	Hubungan yang terjadi antara entitas.
	Atribut	Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut.
	Relasi 1 : 1	Relasi yang menunjukkan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua.
	Relasi 1 : N	Relasi yang menunjukkan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berrelasi dengan banyak entitas pada himpunan entitas yang lain.
	Relasi N : N	Hubungan ini menunjukkan bahwa setiap entitas pada himpunan entitas yang pertama dapat berhubungan dengan banyak entitas pada himpunan entitas yang kedua, demikian juga sebaliknya.

## 2.6 Perangkat Lunak Pendukung

Dalam sub bab ini dijelaskan mengenai teori pendukung tentang perangkat lunak pendukung yang akan digunakan dalam penelitian ini terdiri atas *database* mysql, xampp, bahasa pemrograman php dan *framework* Laravel.

### 2.6.1 Database MySQL

Basis data merupakan suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, yang diorganisasikan berdasarkan skema dan struktur tertentu dan dengan software untuk melakukan manipulasi pada kegunaan tertentu (Pamungkas, 2017). Sehingga basis data ini kumpulan dari informasi yang berguna dan disimpan dalam komputer yang tersusun atas beberapa elemen. Berikut beberapa fungsi dalam penggunaan basis data :

- a. Mengelompokkan data sehingga informasi lebih mudah untuk dimengerti.
- b. Mencegah adanya duplikasi data.
- c. Penyimpanan yang sistematis sehingga menjaga kualitas dari data yang diakses oleh pengguna.
- d. Sangat digunakan dalam mekanisme penyimpanan data khusunya konsep big data.

MySQL merupakan sebuah *database server* dan sebuah program yang dapat mengakses suatu *database* MySQL yang berposisi sebagai *server*, yang berarti program berposisi sebagai *client* atau dapat disimpulkan bahwa MySQL merupakan *database* yang dapat digunakan sebagai *client* maupun *server*. MySQL merupakan bagian dari *Database Management System* (DBMS).

MySQL menyediakan berbagai fasilitas yang dapat digunakan untuk mendokumentasikan dan memanipulasi *server* dengan mengetikkan fungsi tertentu yang dapat dilihat seperti pada tabel 2.3

Tabel 2. 3 Fungsi pada MySQL (Haris Saputro, 2012)

Nama	Fungsi	Keterangan
<i>Help</i>	(\h)	Menampilkan file bantuan pada MySQL
?	(\?)	Jenis lain dari perintah <i>help</i>
<i>Clear</i>	(\c)	Membersihkan dan menggagalkan semua perintah yang telah berjalan pada <i>prompt</i>

Tabel 2. 3 Fungsi pada MySQL (Haris Saputro, 2012) (Lanjutan)

<i>Connect</i>	(\r)	Refresh koneksi pada <i>server host</i>
<i>Ego</i>	(\G)	Menampilkan data secara horizontal
<i>Go</i>	(\g)	Untuk eksekusi
<i>Tee</i>	(\T)	Mengatur tempat file yang akan didokumentasikan
<i>Note</i>	(\t)	Mendokumentasikan semua <i>query</i>
<i>Print</i>	(\p)	Mencetak <i>query</i>
<i>Prompt</i>	(\R)	Mengubah <i>prompt</i> sesuai keinginan
<i>Source</i>	(\.)	Eksekusi <i>query</i> eksternal dengan file yang berekstensi .sql
<i>Use</i>	(\u)	Memasuki maupun mengganti <i>database</i> yang akan digunakan

## 2.6.2 XAMPP

XAMPP merupakan suatu *software* yang bersifat *open source* dan kombinasi dari software pengembangan seperti MySQL dan PHP (Lengkong et al., 2022). XAMPP juga merupakan *tool* yang menyediakan paket perangkat lunak ke dalam satu buah paket.

Dari dua pengertian diatas dapat ditarik kesimpulan bahwa XAMPP merupakan sebuah *tool* yang membantu perangkat lunak berbasis *open source* yang menggabungkan *Apache web server*, MySQL, PHP dalam satu aplikasi yang sama sehingga dapat mempermudah penggunanya. Program ini tersedia dalam GNU General Public License dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

## 2.6.3 PHP

Bahasa pemrograman pada dasarnya merupakan sebuah alat yang digunakan oleh para programmer untuk membuat sebuah aplikasi yang membantu pekerjaan manusia. Salah satu bahasa yang sangat sering digunakan untuk membuat sebuah *website* yaitu bahasa pemrograman *Hypertext Preprocessor* atau yang lebih dikenal

dengan bahasa PHP, bahasa ini sudah menggunakan konsep pemrograman berorientasi objek.

PHP (*Hypertext Preprocessor*) adalah kode atau skrip yang akan dieksekusi pada *server side* (Sutaji, 2012 dalam Wibowo, 2015). *Class* di dalam PHP secara sederhana adalah kumpulan variabel dan fungsi dalam suatu variabel (Sidik, 2012 dalam Wibowo, 2015). PHP dapat dikatakan sebagai pemrograman *interpreter* dikarenakan PHP menterjemahkan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung saat kode program tersebut dijalankan. Selain itu PHP juga disebut sebagai pemrograman *server side programming* dikarenakan proses yang dilakukan dijalankan pada sisi *server*, tidak pada sisi *client* dan PHP bersifat *open source* sehingga dapat dikembangkan sesuai kebutuhan dari para pengguna.

Beberapa keunggulan yang dimiliki oleh bahasa pemrograman PHP :

- a. PHP merupakan *open source* sehingga dapat dengan mudah untuk dikembangkan dan disesuaikan dengan kebutuhan.
- b. Mempunyai *syntax* yang *user-friendly*.
- c. Penulisan *script* PHP yang menyatu dengan HTML sehingga memudahkan dalam pembuatannya.
- d. PHP kompatibel di banyak platform.

Kemampuan yang dimiliki bahasa pemrograman PHP :

- a. Dapat menerima file upload dan menghasilkan atau mengakses *file*.
- b. Mendukung HTTP *cookie* sehingga dapat menyimpan data pada browser.
- c. PHP mendukung koneksi ke *database*.
- d. Memiliki struktur *error handling*.

#### 2.6.4 Framework Laravel

*Framework* adalah sekumpulan *library* yang memiliki fungsi dan kemampuan yang dapat menunjang kebutuhan dalam pengembangan sistem, *framework* ini dikembangkan oleh para *developer* yang tergabung dalam komunitas

*framework*. Salah satu contoh *framework* yang dapat digunakan dalam bahasa pemrograman PHP yaitu *framework* Laravel. (Endra et al., 2021)

Laravel adalah sebuah MVC *web development framework* yang didesain untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan dan perbaikan serta meningkatkan produktifitas pekerjaan dengan sintak yang bersih dan fungsional yang dapat mengurangi banyak waktu untuk implementasi (Widodo & Purnomo, 2016 dalam Luthfi, 2017). Laravel merupakan *framework* dengan versi PHP yang *up-to-date*, karena Laravel mensyaratkan PHP versi 5.3 keatas. Laravel merupakan *framework* PHP yang menekankan pada kesederhanaan dan fleksibilitas pada desainnya (Naista, 2016 dalam Luthfi, 2017). Laravel dikembangkan oleh Taylor Otwell pada bulan Juni 2011 dan terus berkembang hingga saat ini. *Framework* ini menyediakan fungsi-fungsi kode di dalam *library* yang dapat di *install* ke dalam Laravel itu sendiri.

Beberapa komponen fungsi yang terdapat di dalam Laravel :

a. *Migration*

Sebuah fungsi yang bertugas untuk interaksi dengan *database*. Skema *database* direpresentasikan di dalam sebuah *class* dan dapat berjalan pada *database* seperti MySQL, PostgreSQL, MSSQL dan SQLITE.

b. *Eloquent*

Fungsi ini merupakan pengimplementasian *active record* pada Laravel serta berbasis pemrograman berorientasi objek.

c. *Command Line Interface*

Disebut juga sebagai artisan, sehingga pengembang dapat berinteraksi dengan aplikasi untuk sebuah aksi yang akan dijalankan.

d. *Blade Template Engine*

Fungsi ini memberikan estetika dan kebersihan kode program pada *view* secara parsial.

e. *Reverse Routing*

Fungsi yang menjalankan hubungan timbal balik antara *link* dan *route*. Sehingga ketika ada perubahan pada *route* akan otomatis tersambung dengan *link* yang relevan.

f. *Restful Controller*

Fungsi yang memberikan pilihan untuk memisahkan logika dalam merespon permintaan GET dan POST.

g. *Unit Testing*

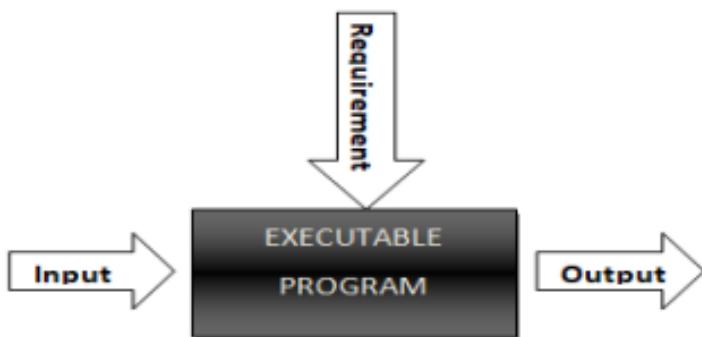
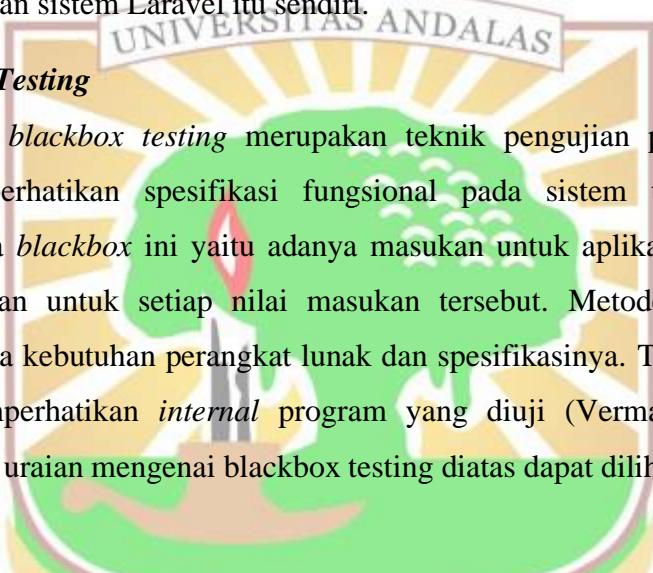
Fungsi yang bertujuan untuk mencari *error* dalam sebuah sistem, memiliki banyak tes sehingga mempermudah dalam mendeteksi dan mencegah regresi.

h. *Automatic Pagination*

Pengelolaan halaman yang dilakukan secara otomatis dan terintegrasi dengan sistem Laravel itu sendiri.

## 2.7 Blackbox Testing

Pengujian *blackbox testing* merupakan teknik pengujian perangkat lunak dengan memperhatikan spesifikasi fungsional pada sistem tersebut. Fokus pengujian pada *blackbox* ini yaitu adanya masukan untuk aplikasi dan keluaran yang diharapkan untuk setiap nilai masukan tersebut. Metode pengujian ini didasarkan pada kebutuhan perangkat lunak dan spesifikasinya. Teknik pengujian ini tidak memperhatikan *internal* program yang diuji (Verma et al., 2017). Gambaran dari uraian mengenai blackbox testing diatas dapat dilihat pada Gambar 2.6 berikut.



Gambar 2. 6 Gambaran *Blackbox Testing* (Verma et al., 2017)

Tipe-tipe pengujian yang terdapat pada *blackbox testing* :

1. *Equivalence partitioning*

Teknik yang digunakan untuk mendesain kasus uji, dibuatkan semua nilai masukan yang terbagi menjadi beberapa partisi yaitu partisi valid dan tidak valid dan pengujian fungsional dilakukan berdasarkan kasus uji tersebut.

2. *Boundary Value Analysis*

Teknik ini melaksanakan pengujian berdasarkan nilai inputan untuk setiap domain kasus uji.

3. *Cause Effect Graphing*

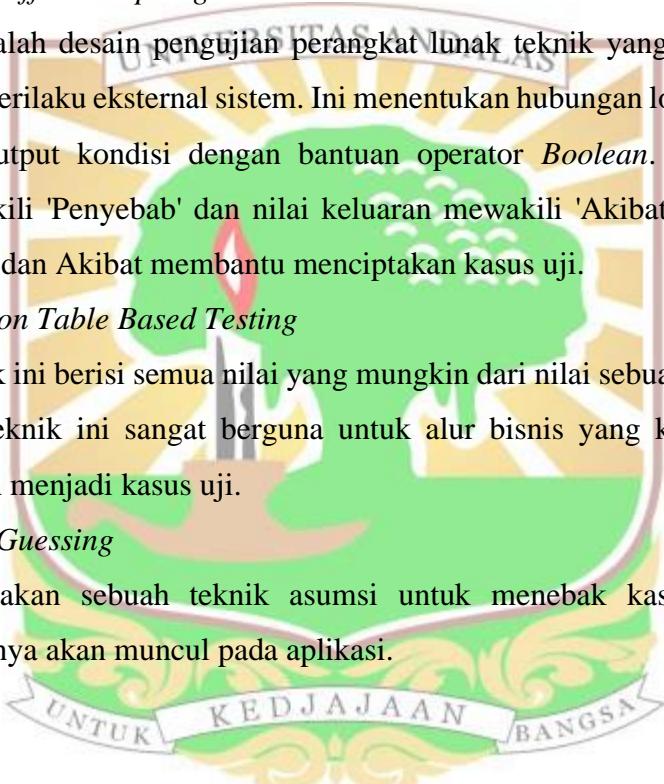
Ini adalah desain pengujian perangkat lunak teknik yang berkonsentrasi pada perilaku eksternal sistem. Ini menentukan hubungan logis antara input dan output kondisi dengan bantuan operator *Boolean*. Nilai masukan mewakili 'Penyebab' dan nilai keluaran mewakili 'Akibat'. Itu hubungan Sebab dan Akibat membantu menciptakan kasus uji.

4. *Decision Table Based Testing*

Teknik ini berisi semua nilai yang mungkin dari nilai sebuah kondisi kasus uji. Teknik ini sangat berguna untuk alur bisnis yang kompleks untuk diubah menjadi kasus uji.

5. *Error Guessing*

Merupakan sebuah teknik asumsi untuk menebak kasus *error* yang sekiranya akan muncul pada aplikasi.



## **BAB III**

### **METODOLOGI PENELITIAN**

Pada bab ini menjelaskan mengenai metodologi penelitian yang terdiri atas objek penelitian, metode pengumpulan data, metode pengembangan sistem, dan *flowchart* penelitian yang dilakukan selama proses penelitian berlangsung.

#### **3.1 Objek Penelitian**

Dalam penelitian ini objek yang dibahas yaitu pembangunan aplikasi dengan model bisnis *marketplace* untuk UMKM berbagai unit bisnis pelaku usaha oleh-oleh makanan yang ada di Kota Payakumbuh. Objek ini diambil karena usaha UMKM di kota tersebut masih menggunakan cara manual dalam penjualan produk mereka dan pemasaran produk juga tidak dilakukan menggunakan *platform* teknologi sehingga pemasaran hanya dijangkau disekitaran daerah tersebut. Sedangkan produk yang mereka jual merupakan salah satu produk yang berpotensi memiliki nilai jual yang tinggi karena produk yang dihasilkan berkualitas dan rasanya yang enak serta memiliki banyak peminat dikalangan masyarakat. Oleh karena itu dibutuhkan sebuah *platform* yang bisa mereka gunakan untuk sebagai sarana penjualan dan pemasaran seluas-luasnya.

#### **3.2 Metode Pengumpulan Data**

Beberapa metode pengumpulan data yang digunakan dalam penelitian ini sebagai berikut :

a. Observasi

Observasi dilakukan dengan mendatangi dan mengamati langsung daerah sekitar sentra pusat oleh-oleh di Kota Payakumbuh dan didapatkan hasil yaitu terdapat banyak sekali para pelaku usaha UMKM oleh-oleh makanan yang ada di Kota Payakumbuh ini dan dilakukan pengamatan mengenai proses bisnis yang berjalan pada usaha ini.

b. Wawancara

Wawancara dilakukan untuk mendapatkan informasi yang lebih spesifik mengenai data yang dibutuhkan, wawancara dilakukan secara

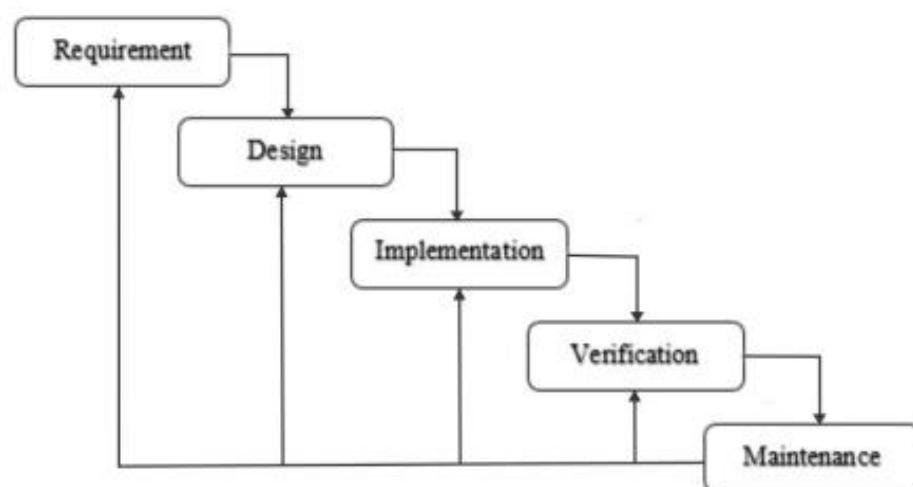
langsung terhadap pemilik-pemilik UMKM oleh-oleh makanan yang ada di Kota Payakumbuh.

c. Studi Literatur

Studi literatur dilakukan melalui pencarian mengenai teori-teori terkait dan penelitian terdahulu yang sejenis bersumber dari jurnal ilmiah, situs internet dan juga bacaan lain yang dapat mendukung penelitian ini.

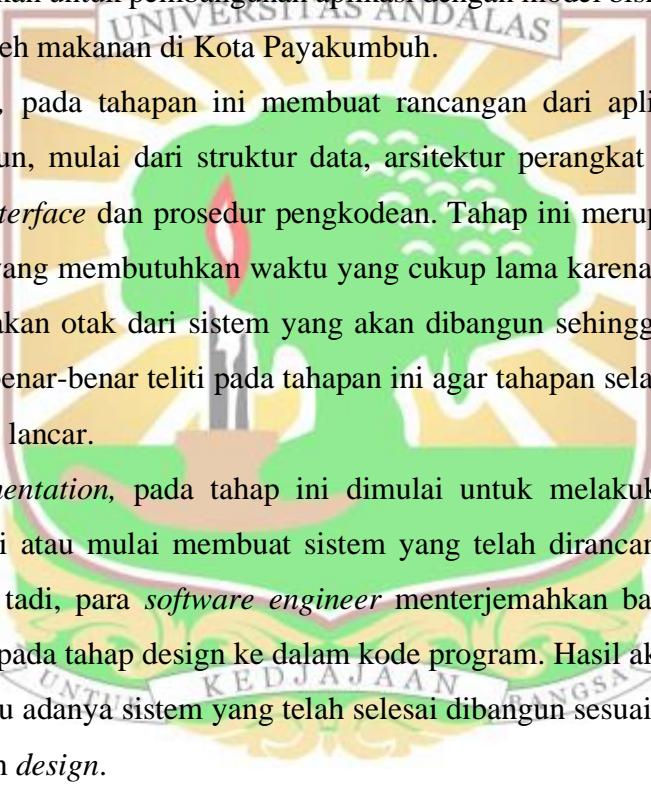
### 3.3 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini yaitu metode pengembangan sistem *waterfall*. Menurut (Pressman, 2012 dalam Wahid, 2020). *Metode Waterfall* atau yang dikenal dengan model air terjun merupakan sebuah metode pengembangan yang dilakukan secara sekuensial. Metode ini bersifat sistematis dan Langkah-langkah pengembangan sistem harus secara berurutan untuk dilakukan. Dalam penelitian ini metode *waterfall* digunakan hanya sampai ke tahap *verification* atau tahap pengujian dikarenakan tujuan dalam penelitian ini yaitu untuk membangun aplikasi dengan model bisnis *marketplace* hingga siap digunakan dan tidak sampai ke tahapan *maintanance*. Metode pengembangan *waterfall* dapat dilihat pada gambar 3.1 berikut :



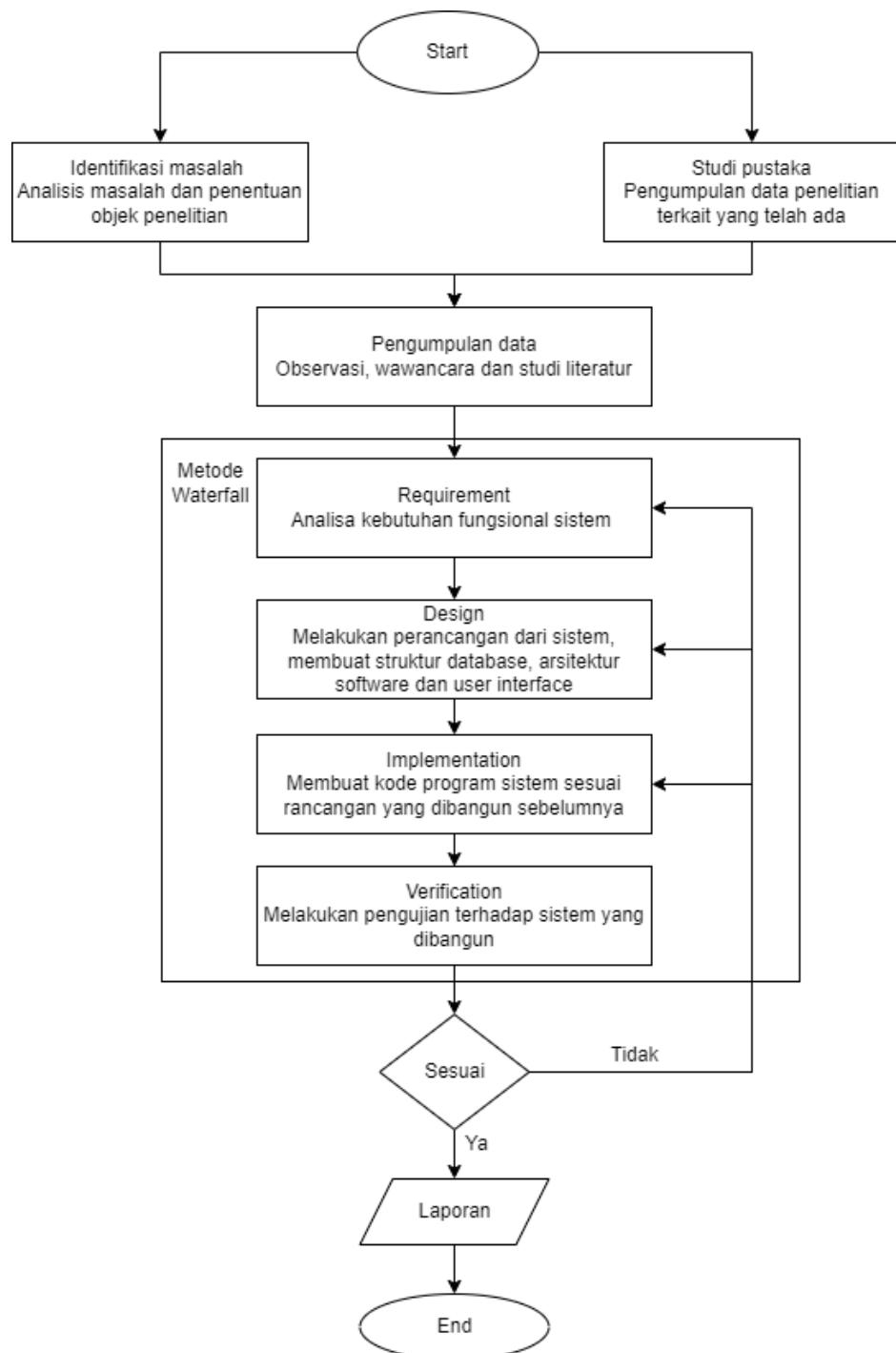
Gambar 3. 1 Siklus Pengembangan Metode *Waterfall* (Pressman 2012 dalam Wahid, 2020)

Berdasarkan Gambar 3.1 dapat dijabarkan tahapan yang ada dalam siklus pengembangan metode waterfall :

- 
- a. *Requirement*, sebuah tahapan awal dalam pengembangan sistem. Dalam tahap ini dilakukan pengumpulan data-data yang dibutuhkan selama pengembangan dan juga dilakukan analisis fungsional yang akan dibuat kedalam sebuah sistem. Tujuan dari tahap ini yaitu untuk mengetahui keinginan dari pengguna aplikasi sehingga aplikasi yang dikembangkan benar-benar sesuai dengan kebutuhan dari para penggunanya. Dalam penelitian ini dilakukan pengumpulan data UMKM, jenis produk-produk yang dijual, harga produk untuk tiap masing-masing UMKM serta pembukuan pencatatan penjualan produk tersebut. Data-data ini akan digunakan untuk pembangunan aplikasi dengan model bisnis pada UMKM oleh-oleh makanan di Kota Payakumbuh.
  - b. *Design*, pada tahapan ini membuat rancangan dari aplikasi yang akan dibangun, mulai dari struktur data, arsitektur perangkat lunak, tampilan *user interface* dan prosedur pengkodean. Tahap ini merupakan salah satu tahap yang membutuhkan waktu yang cukup lama karena dalam tahap ini merupakan otak dari sistem yang akan dibangun sehingga *system analysis* harus benar-benar teliti pada tahapan ini agar tahapan selanjutnya berjalan dengan lancar.
  - c. *Implementation*, pada tahap ini dimulai untuk melakukan pengkodean aplikasi atau mulai membuat sistem yang telah dirancang pada tahapan design tadi, para *software engineer* menterjemahkan bahasa yang telah dibuat pada tahap design ke dalam kode program. Hasil akhir dari tahapan ini yaitu adanya sistem yang telah selesai dibangun sesuai rancangan pada tahapan *design*.
  - d. *Verification*, Melakukan pengujian terhadap aplikasi dengan model bisnis *marketplace* untuk UMKM yang telah dibangun apakah telah memenuhi persyaratan sistem atau belum. Semua fungsional yang telah diimplementasikan harus diuji agar tidak terdapat *error* dalam penggunaan aplikasi. Pengujian ini akan dilakukan oleh pihak penjual dari pelaku usaha UMKM dan juga oleh calon pembeli produk.

### 3.4 Flowchart Penelitian

Pada *flowchart* penelitian, ditampilkan alur dari penelitian yang akan dilaksanakan berdasarkan metode *waterfall*, gambaran dari *flowchart* penelitian ini terdapat pada Gambar 3.2 berikut



Gambar 3. 2 *Flowchart* Penelitian

Berdasarkan Gambar 3.2 dapat dijabarkan tahapan dalam flowchart penelitian yaitu :

a. Identifikasi Masalah

Melakukan analisis masalah pada objek penelitian yang telah ditentukan untuk mendapatkan dasar dalam pembangunan aplikasi berbasis model bisnis *marketplace* pada penelitian ini.

b. Studi Pustaka

Dilakukan pengumpulan data-data penelitian terkait yang telah ada yang sesuai dengan konteks penelitian dan dapat dijadikan sebagai bahan referensi dalam penelitian.

c. Pengumpulan Data

Dilakukan pengumpulan data melalui observasi, wawancara dan studi Pustaka untuk mendapatkan data-data yang diperlukan selama proses penelitian berlangsung.

d. Tahap *Requirement*

Menganalisa kebutuhan data dan proses bisnis yang sedang berlangsung menggunakan *Business Process Modelling Notation* (BPMN). Merancang analisa fungsional yang akan dibuat serta pemodelan UML yang terdiri atas *use case diagram*, *use case scenario*, *sequence diagram* dan *class analysis*.

e. Tahap *Design*

Merancang aplikasi dengan membuat struktur database menggunakan *Entity Relationship Diagram* (ERD), arsitektur aplikasi, tampilan *user interface* hingga *class diagram*.

f. Tahap *Implementation*

Melakukan pembuatan kode program aplikasi menggunakan bahasa pemrograman PHP dengan framework Laravel. Hasil akhir dari tahap ini akan menghasilkan sebuah aplikasi UMKM berbasis model bisnis *marketplace*.

g. Tahap *Verification*

Melakukan pengujian dari setiap fungsional yang telah berhasil dibuat menggunakan metode pengujian *black box* dan memastikan semua fungsional tidak lagi terdapat *error*.

h. Laporan

Dilakukan pembuatan laporan tentang pembangunan aplikasi berbasis model bisnis *marketplace* UMKM di Kota Payakumbuh dari proses perancangan aplikasi sampai ke aplikasi tersebut siap digunakan.



## BAB IV

### ANALISIS SISTEM DAN PERANCANGAN

Pada bab ini menjelaskan mengenai hasil dari analisa kebutuhan dan perancangan dari aplikasi berbasis model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh. Pada analisis sistem ini menggunakan BPMN (*Business Process Model and Notation*), analisis kebutuhan fungsional sistem, *use case diagram*, deskripsi tugas aktor, *use case scenario*, *sequence diagram* dan *class analysis*. Sedangkan pada perancangan terdiri atas ERD (*Entity Relationship Diagram*), struktur tabel *database*, arsitektur aplikasi, *class diagram* dan perancangan antarmuka sistem.

#### 4.1 Analisis Sistem

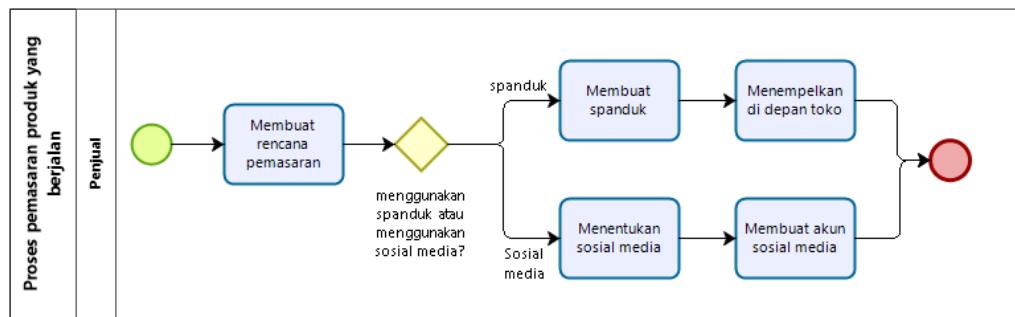
Pada sub bab ini menjelaskan mengenai tahapan analisis dari sistem yang akan dibangun, dimulai dari analisis sistem yang sedang berjalan hingga usulan analisis sistem yang diusulkan. Analisis sistem ini terdiri atas kedudukan sistem, proses bisnis yang sedang berjalan dan diusulkan menggunakan BPMN (*Business Process Model and Notation*), kebutuhan fungsional sistem, *use case diagram*, deskripsi tugas aktor, *use case scenario*, *sequence diagram* dan *class analysis*.

##### 4.1.1 Sistem yang Berjalan

Proses bisnis dari pembangunan aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh yang sedang berjalan dimodelkan dengan menggunakan BPMN (*Business Process Model and Notation*). Terdapat empat kegiatan utama yaitu proses pemasaran, transaksi produk eceran, proses transaksi produk grosir dan proses pengelolaan produk.

###### 4.1.1.1 BPMN Proses Pemasaran yang Berjalan

Proses pemasaran yang berjalan ini dilakukan oleh aktor penjual sebagai pihak yang akan menjual produk. Pemodelan proses bisnis pemasaran yang berjalan ini dapat dilihat pada Gambar 4.1



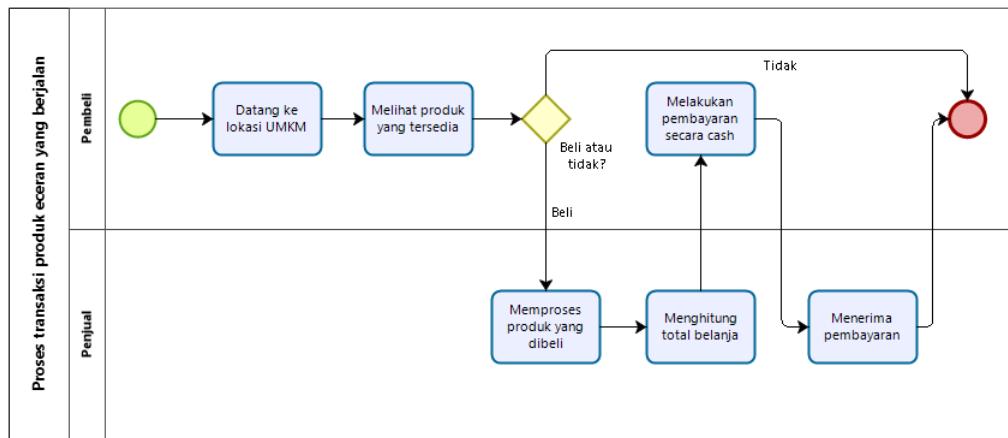
Gambar 4. 1 Proses Pemasaran yang Berjalan

Berdasarkan Gambar 4.1 berikut penjelasan rinci mengenai alur pemasaran yang berjalan :

1. Penjual membuat rencana mengenai pemasaran yang akan dilakukan.
2. Terdapat kondisi apakah penjual akan menggunakan spanduk atau menggunakan sosial media menjadi sarana promosinya.
3. Jika menggunakan spanduk, maka penjual akan membuat spanduk dan menempatkannya di depan toko usaha mereka.
4. Jika menggunakan sosial media, penjual akan menentukan sosial media apa yang digunakan dan membuat akun sosial media tersebut.
5. Proses pemasaran selesai.

#### **4.1.1.2 BPMN Proses Transaksi Penjualan Produk Eceran yang Berjalan**

Proses transaksi produk eceran yang berjalan ini dilakukan oleh dua aktor yaitu penjual sebagai pihak yang akan menjual produk UMKM nya dan pembeli sebagai pihak yang akan membeli produk tersebut dimana proses ini dimulai dari pembeli. Pemodelan proses bisnis transaksi penjualan produk eceran yang berjalan dapat dilihat pada Gambar 4.2



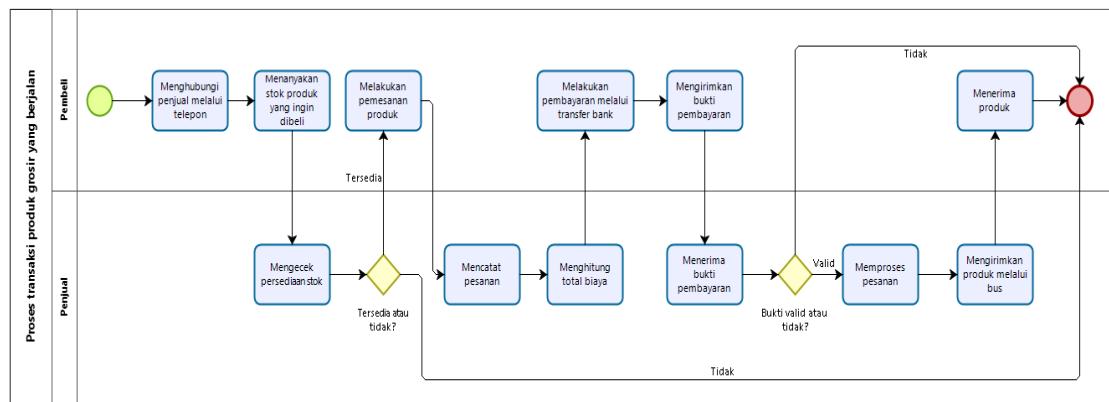
Gambar 4. 2 Proses Transaksi Penjualan Produk Eceran yang Berjalan

Berdasarkan Gambar 4.2 berikut penjelasan rinci mengenai alur proses transaksi produk eceran yang berjalan :

1. Pembeli datang langsung ke lokasi UMKM.
2. Pembeli melihat produk yang tersedia.
3. Pembeli akan memutuskan untuk jadi membeli produk atau tidak.
4. Jika jadi, maka penjual akan memproses produk yang akan dibeli oleh pembeli tersebut.
5. Penjual akan menghitung total biaya dari belanja pembeli.
6. Pembeli akan melakukan pembayaran secara tunai.
7. Penjual akan menerima pembayaran.
8. Proses transaksi produk eceran selesai.

#### 4.1.1.3 BPMN Proses Transaksi Penjualan Produk Grosir yang Berjalan

Proses transaksi penjualan produk grosir yang berjalan ini dilakukan oleh dua aktor yaitu penjual sebagai pihak yang akan menjual produk UMKM nya dan pembeli sebagai pihak yang akan membeli produk tersebut dimana proses ini dimulai dari pembeli. Pemodelan proses bisnis transaksi penjualan produk grosir yang berjalan dapat dilihat pada Gambar 4.3



Gambar 4. 3 Proses Transaksi Penjualan Produk Grosir yang Berjalan

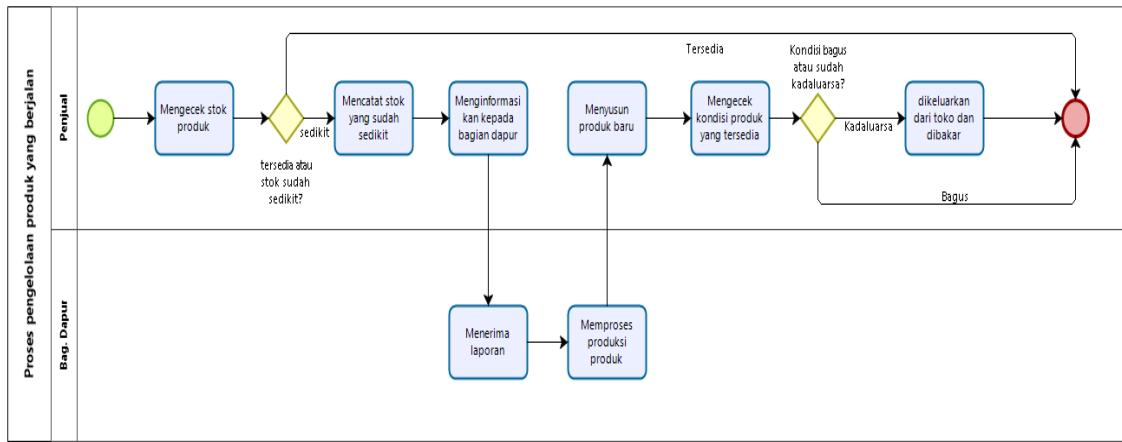
Berdasarkan Gambar 4.3 berikut penjelasan rinci mengenai alur proses transaksi produk grosir yang berjalan :

1. Pembeli menghubungi penjual melalui telepon.
2. Pembeli menanyakan stok produk yang akan dibeli.
3. Penjual akan melakukan pengecekan stok.
4. Jika tidak tersedia maka proses pembelian tidak jadi berlangsung.
5. Jika tersedia maka pembeli akan melakukan pemesanan produk yang diinginkan.
6. Penjual akan mencatat pesanan dari pembeli.
7. Penjual akan menghitung total belanja dari pembeli dan mengirimkan jumlahnya pada pembeli.
8. Pembeli melakukan pembayaran melalui transfer bank dan mengirimkan buktinya pada penjual.
9. Penjual akan menerima bukti tersebut dan akan mengecek apakah bukti tersebut valid atau tidak.
10. Jika valid maka penjual akan memproses pesanan.
11. Penjual akan mengirimkan pesanan melalui bus.
12. Pembeli menerima pesanan.
13. Proses transaksi produk grosir selesai.

#### 4.1.1.4 BPMN Proses Pengelolaan Produk yang Berjalan

Proses pengelolaan produk yang berjalan ini dilakukan oleh dua aktor yaitu penjual sebagai pihak yang akan mengecek produk UMKM nya dan

bagian dapur yang akan memproduksi ulang produk yang sudah menipis stoknya dimana proses ini dimulai dari penjual. Pemodelan proses bisnis pengelolaan produk yang berjalan dapat dilihat pada Gambar 4.4



Gambar 4. 4 Proses Pengelolaan Produk yang Berjalan

Berdasarkan Gambar 4.4 berikut penjelasan rinci mengenai alur proses pengelolaan produk yang berjalan :

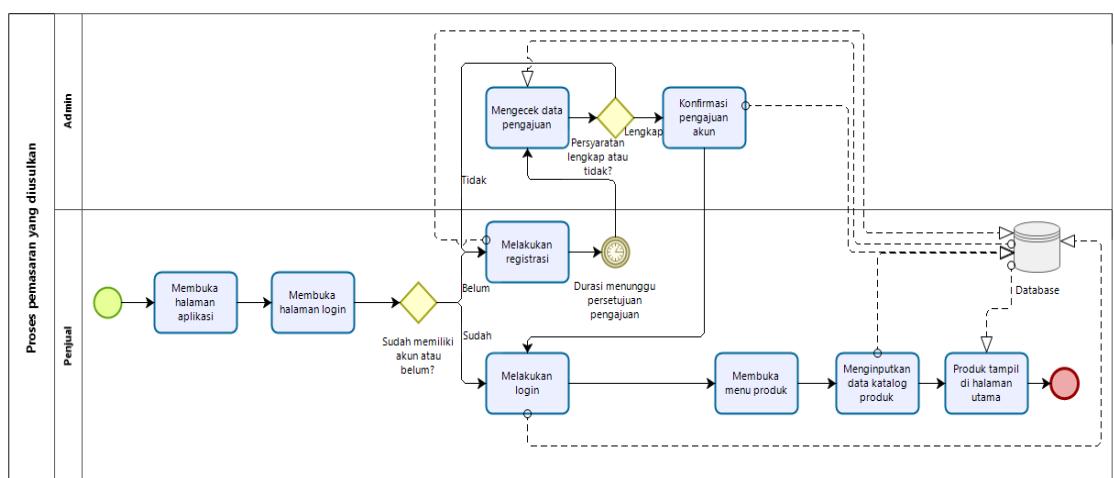
1. Penjual melakukan pengecekan stok produk UMKM.
2. Jika tersedia maka proses berakhiri.
3. Jika stok sudah sedikit, maka penjual akan mencatat produk-produk yang stok nya sudah sedikit.
4. Penjual akan menginformasikan hal tersebut kepada bagian dapur.
5. Bagian dapur akan menerima laporan mengenai stok produk yang sudah sedikit.
6. Bagian dapur akan melakukan proses produksi produk tersebut.
7. Setelah diproduksi, penjual akan menata kembali produk-produk tersebut.
8. Penjual melakukan pengecekan kondisi dari produk.
9. Jika masih bagus, maka proses akan berakhiri.
10. Jika produk sudah kadaluarsa maka produk tersebut akan dikeluarkan dari toko dan produk akan dibakar.
11. Proses pengelolaan produk selesai.

#### 4.1.2 Sistem yang Diusulkan

Berdasarkan proses bisnis yang berjalan sebelumnya, maka dirancang sistem yang diusulkan dengan menggunakan model bisnis *marketplace* dimana kegiatan utamanya ada tiga yaitu proses pemasaran, proses transaksi produk dan proses pengelolaan produk. BPMN (*Business Process Model and Notation*) dari ketiga proses kegiatan utama tersebut dijelaskan pada poin dibawah ini.

##### 4.1.2.1 BPMN Proses Pemasaran Produk yang Diusulkan

Proses pemasaran produk yang diusulkan ini dilakukan oleh dua aktor yaitu penjual sebagai pihak yang akan mempromosikan produknya dan admin sebagai pihak yang akan mengkonfirmasi akun dari penjual dimana proses ini dimulai dari penjual. Pemodelan proses bisnis pemasaran produk yang diusulkan dapat dilihat pada Gambar 4.5



Gambar 4. 5 Proses Pemasaran Produk yang Diusulkan

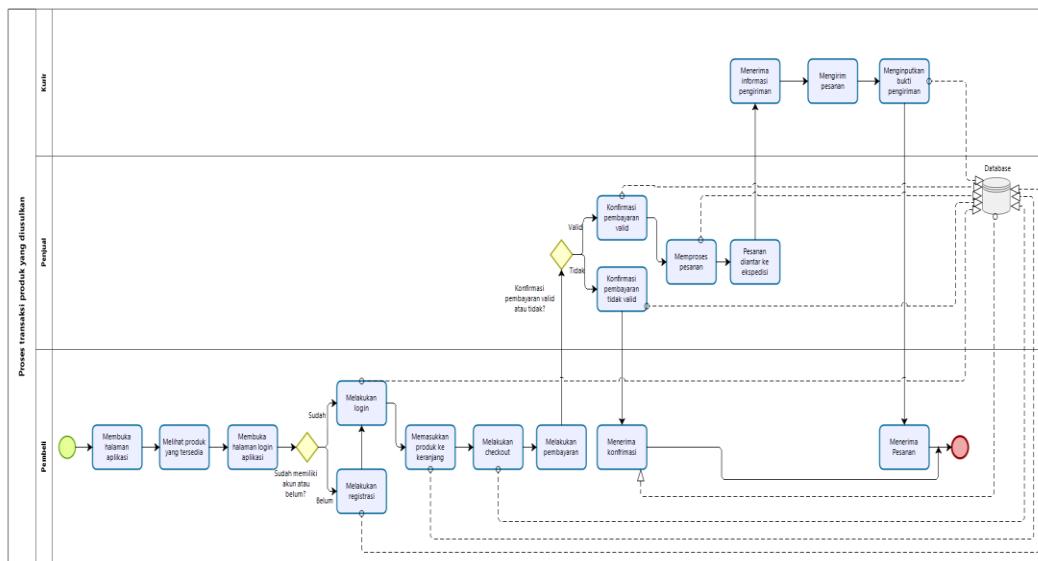
Berdasarkan Gambar 4.5 berikut penjelasan rinci mengenai alur proses pemasaran produk yang diusulkan :

1. Penjual membuka halaman aplikasi.
2. Penjual membuka halaman login, lalu muncul form untuk melakukan login.
3. Jika belum mempunyai akun, maka harus melakukan registrasi dengan mengisikan data yang diminta.

4. Admin akan mengecek data registrasi apakah persyaratan yang dibutuhkan lengkap atau tidak.
5. Jika tidak, maka penjual harus melakukan registrasi ulang kembali.
6. Jika lengkap, maka admin akan mengkonfirmasi akun penjual.
7. Jika sudah memiliki akun, penjual melakukan login.
8. Penjual membuka halaman menu produk.
9. Penjual menginputkan data katalog produk.
10. Produk yang telah ditambahkan akan tampil di halaman utama aplikasi.
11. Proses pemasaran produk selesai.

#### 4.1.2.2 BPMN Proses Transaksi Produk yang Diusulkan

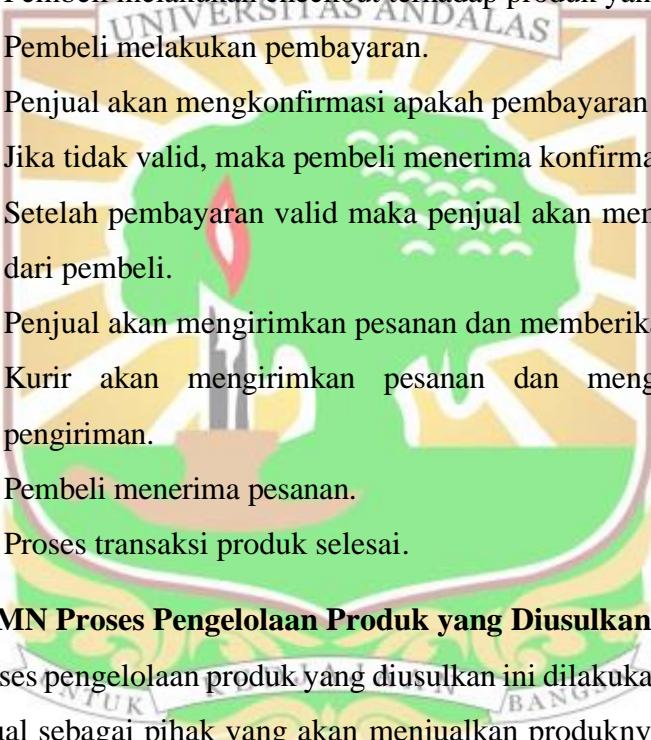
Proses transaksi produk yang diusulkan ini dilakukan oleh dua aktor yaitu penjual sebagai pihak yang akan menjualkan produknya dan pembeli sebagai pihak yang akan membeli produk dimana proses ini dimulai dari pembeli. Pemodelan proses bisnis transaksi produk yang diusulkan dapat dilihat pada Gambar 4.6



Gambar 4. 6 Proses Transaksi Produk yang Diusulkan

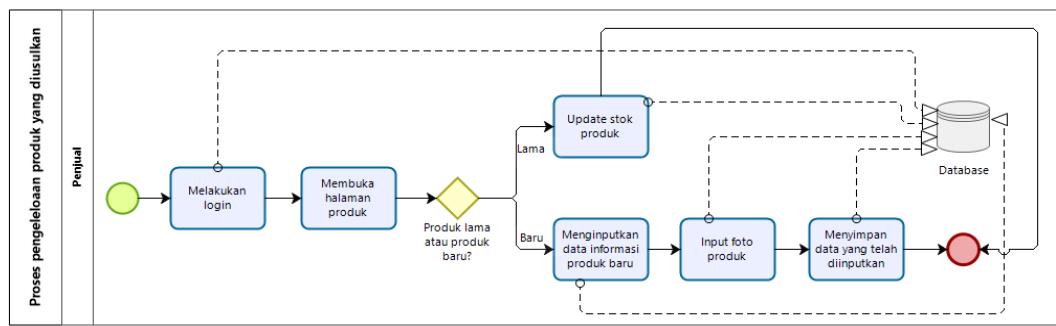
Berdasarkan Gambar 4.6 berikut penjelasan rinci mengenai alur proses transaksi produk yang diusulkan :

1. Pembeli membuka halaman aplikasi.

- 
2. Pembeli melihat produk yang tersedia di katalog aplikasi.
  3. Pembeli diarahkan ke halaman login jika ingin melakukan transaksi pembelian.
  4. Jika pembeli belum memiliki akun, maka harus melakukan registrasi terlebih dahulu.
  5. Jika sudah memiliki akun, maka lakukan login dengan mengisi form login.
  6. Pembeli memilih produk yang akan dibeli dengan memasukkan produk tersebut ke keranjang.
  7. Pembeli melakukan checkout terhadap produk yang akan dibeli.
  8. Pembeli melakukan pembayaran.
  9. Penjual akan mengkonfirmasi apakah pembayaran valid atau tidak.
  10. Jika tidak valid, maka pembeli menerima konfirmasi.
  11. Setelah pembayaran valid maka penjual akan memproses pesanan dari pembeli.
  12. Penjual akan mengirimkan pesanan dan memberikan no resi.
  13. Kurir akan mengirimkan pesanan dan menginputkan bukti pengiriman.
  14. Pembeli menerima pesanan.
  15. Proses transaksi produk selesai.

#### 4.1.2.3 BPMN Proses Pengelolaan Produk yang Diusulkan

Proses pengelolaan produk yang diusulkan ini dilakukan oleh satu aktor yaitu penjual sebagai pihak yang akan menjualkan produknya dimana proses ini dimulai dengan penjual yang melakukan login ke aplikasi. Pemodelan proses bisnis pengelolaan produk yang diusulkan dapat dilihat pada Gambar 4.7



Gambar 4. 7 Proses Pengelolaan Produk yang Diusulkan

Berdasarkan Gambar 4.7 berikut penjelasan rinci mengenai alur proses pengelolaan produk yang diusulkan :

1. Penjual melakukan login ke aplikasi.
2. Penjual membuka halaman produk dari UMKM nya.
3. Produk yang akan diinputkan apakah produk lama atau produk baru.
4. Jika produk lama, maka penjual akan mengupdate dari data produk yang telah ada.
5. Jika produk baru, maka penjual akan menginputkan data informasi mengenai produk.
6. Penjual menginputkan foto produk.
7. Penjual menyimpan data yang telah diinputkan.
8. Proses pengelolaan produk selesai.

#### 4.1.3 Analisis Kebutuhan Fungsional

Berdasarkan hasil analisis proses bisnis pembangunan aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh yang diusulkan sebelumnya dapat dirumuskan beberapa kebutuhan fungsional dimana pada aplikasi ini terdapat empat aktor yang terlibat yaitu admin aplikasi, pihak penjual atau pemilik UMKM, pihak pembeli serta pihak kurir yang akan mengantarkan pesanan. Kebutuhan fungsional setiap aktor dapat dijelaskan sebagai berikut :

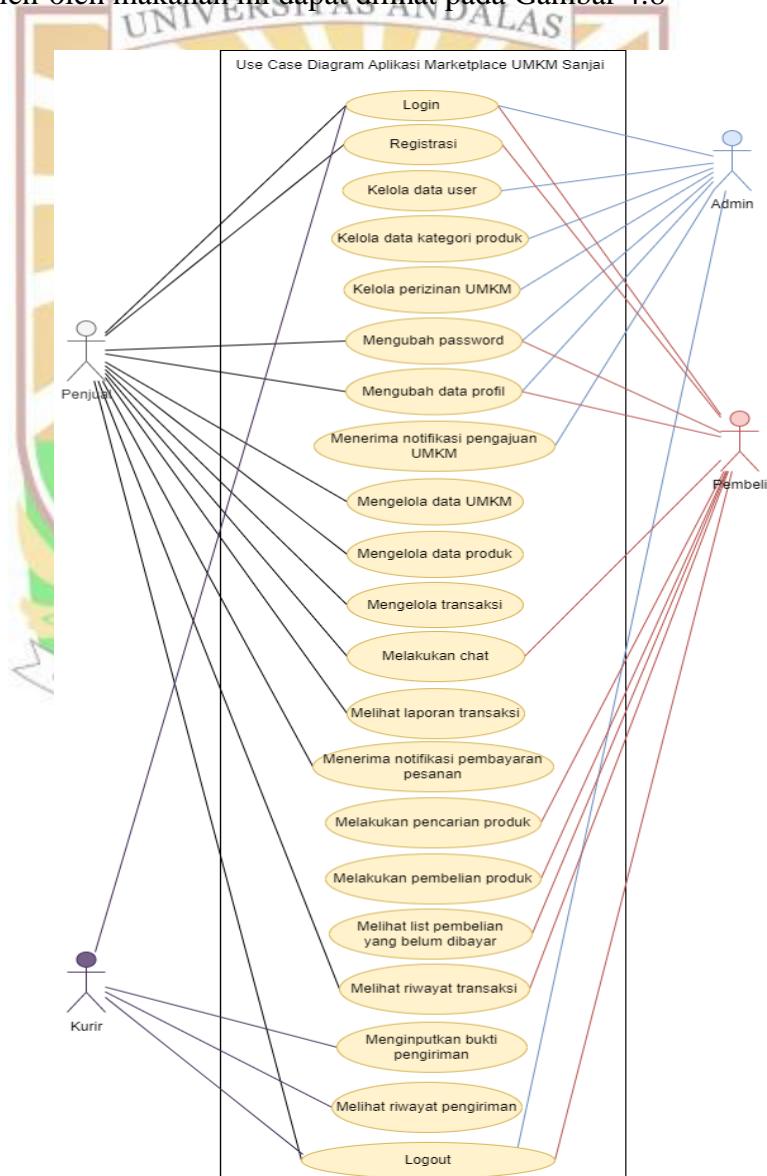
1. Kebutuhan Fungsional Admin Aplikasi
  - a. *Login*

- 
- The logo of Universitas Andalas features a green shield-shaped base with a yellow border. Inside the shield, there is a stylized tree with red flowers and green leaves. Above the tree, a yellow sunburst pattern radiates from behind a green hill. The words "UNIVERSITAS ANDALAS" are written in a white, serif font along the top edge of the shield. Below the shield, a decorative banner with floral patterns contains the words "KEDAJAAN" and "BANGSA".
- b. Mengelola data *user*
  - c. Mengubah *password*
  - d. Mengubah data profil admin
  - e. Mengelola data kategori produk
  - f. Mengelola perizinan UMKM
  - g. Menerima notifikasi pengajuan UMKM
  - h. *Logout*
  - 2. Kebutuhan Fungsional Penjual
    - a. *Login*
    - b. Registrasi
    - c. Mengubah *password*
    - d. Mengubah data profil penjual
    - e. Mengelola data UMKM
    - f. Mengelola data produk
    - g. Mengelola transaksi pembelian
    - h. Melakukan percakapan / *chat*
    - i. Melihat laporan transaksi
    - j. Melihat riwayat transaksi
    - k. Menerima notifikasi pembayaran pesanan
    - l. *Logout*
  - 3. Kebutuhan Fungsional Pembeli
    - a. *Login*
    - b. Registrasi
    - c. Mengubah *password*
    - d. Mengubah data profil
    - e. Melakukan percakapan / *chat*
    - f. Melakukan pencarian produk
    - g. Mengelola pembelian produk
    - h. Melihat *list* pembelian yang belum dibayar
    - i. Melihat riwayat pembelian produk
    - j. *Logout*
  - 4. Kebutuhan Fungsional Kurir

- a. Login
- b. Menginputkan bukti pengiriman
- c. Melihat riwayat pengiriman
- d. Logout

#### 4.1.4 Use Case Diagram

Berdasarkan analisa kebutuhan fungsional beserta aktor yang terlibat, maka dapat dirumuskan sebuah *use case* yang menggambarkan hubungan antara pengguna dengan fungsional yang ada pada sistem ini. *Use case diagram* perancangan sistem aplikasi dengan model bisnis *marketplace* pada UMKM oleh-oleh makanan ini dapat dilihat pada Gambar 4.8



Gambar 4. 8 Use Case Diagram Aplikasi

#### 4.1.5 Deskripsi Tugas Aktor

Pada *use case diagram* diatas dapat dilihat bahwa terdapat empat aktor yang berperan dalam aplikasi ini yaitu admin aplikasi, penjual, pembeli dan kurir. Masing-masing aktor menjalankan fungsional sesuai dengan kebutuhannya sendiri, berikut dijelaskan deskripsi dari tugas masing-masing aktor tersebut.

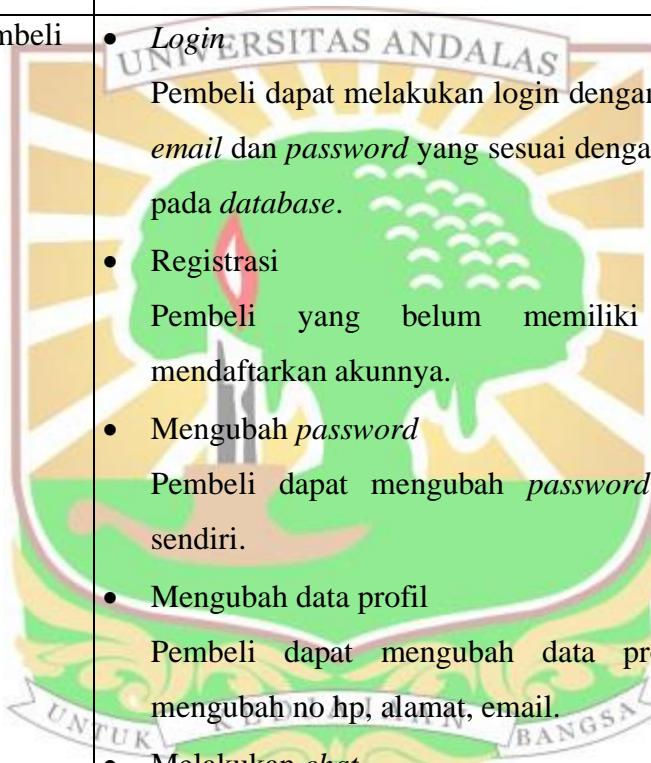
Tabel 4. 1 Deskripsi Tugas Aktor

No	Aktor	Deskripsi Tugas
1	Admin	<ul style="list-style-type: none"><li>• <i>Login</i> Admin dapat melakukan <i>login</i> dengan menginputkan <i>email</i> dan <i>password</i> yang sesuai dengan data yang ada pada <i>database</i>.</li><li>• <i>Kelola data user</i> Admin dapat menambah, mengedit dan menghapus data <i>user</i> yang terdaftar dalam aplikasi ini.</li><li>• Mengubah <i>password</i> Admin dapat mengubah <i>password</i> dari akunnya.</li><li>• Mengubah data profil Admin dapat mengubah data profil diri.</li><li>• <i>Kelola data kategori produk</i> Admin dapat melakukan pengelolaan yaitu dapat menambahkan, mengedit dan menghapus data kategori produk dari produk yang dijualkan oleh UMKM oleh-oleh makanan.</li><li>• <i>Kelola perizinan UMKM</i> Admin dapat mengkonfirmasi pengajuan pendaftaran akun oleh penjual, apakah akan menyetujui pengajuan atau menolaknya.</li><li>• <i>Menerima notifikasi pengajuan UMKM</i> Admin dapat menerima notifikasi jika ada pengajuan UMKM.</li></ul>

Tabel 4.1 Deskripsi Tugas Aktor (Lanjutan)

		<ul style="list-style-type: none"> <li>• <i>Logout</i> Admin dapat keluar dari aplikasi dengan menekan tombol <i>logout</i>.</li> </ul>
2	Penjual	<ul style="list-style-type: none"> <li>• <i>Login</i> Penjual dapat melakukan login dengan menginputkan <i>email</i> dan <i>password</i> yang sesuai dengan data yang ada pada <i>database</i>.</li> <li>• <i>Registrasi</i> Penjual yang belum memiliki akun, dapat melakukan registrasi dengan menginputkan data dan persyaratan tertentu.</li> <li>• Mengubah <i>password</i> Penjual dapat mengubah <i>password</i> akunnya sendiri.</li> <li>• Mengubah data profil Penjual dapat mengubah data profil diri.</li> <li>• Mengelola data UMKM Penjual dapat menambahkan, mengedit serta menghapus data UMKM.</li> <li>• Mengelola data produk Penjual dapat menambahkan, mengubah dan menghapus data produk.</li> <li>• Mengelola transaksi Penjual dapat melakukan pengelolaan terhadap transaksi yang masuk seperti mengkonfirmasi bukti pembayaran yang dikirimkan oleh pembeli.</li> <li>• Melakukan <i>chat</i> Penjual dapat melakukan <i>chat</i> melalui aplikasi.</li> <li>• Melihat laporan transaksi Penjual dapat melihat laporan transaksi bulanan penjualan produk.</li> </ul>

Tabel 4.1 Deskripsi Tugas Aktor (Lanjutan)

		<ul style="list-style-type: none"> <li>• Melihat riwayat transaksi Penjual dapat melihat riwayat transaksi penjualan.</li> <li>• Menerima notifikasi pembayaran pesanan Penjual menerima notifikasi ketika ada pembayaran pesanan oleh pembeli.</li> <li>• <i>Logout</i> Penjual dapat keluar dari aplikasi dengan menekan tombol <i>logout</i>.</li> </ul>
3	Pembeli	 <ul style="list-style-type: none"> <li>• <i>Login</i> Pembeli dapat melakukan login dengan menginputkan <i>email</i> dan <i>password</i> yang sesuai dengan data yang ada pada <i>database</i>.</li> <li>• Registrasi Pembeli yang belum memiliki akun dapat mendaftarkan akunnya.</li> <li>• Mengubah <i>password</i> Pembeli dapat mengubah <i>password</i> dari akunnya sendiri.</li> <li>• Mengubah data profil Pembeli dapat mengubah data profilnya seperti mengubah no hp, alamat, email.</li> <li>• Melakukan <i>chat</i> Pembeli dapat melakukan <i>chat</i> dengan penjual untuk menanyakan produk.</li> <li>• Melakukan pencarian produk Pembeli dapat melakukan pencarian produk yang diinginkan di halaman <i>dashboard</i>.</li> <li>• Mengelola pembelian produk Pembeli dapat melakukan pembelian produk dengan menambahkan produk tersebut kedalam keranjang</li> </ul>

Tabel 4.1 Deskripsi Tugas Aktor (Lanjutan)

		<p>kemudian melakukan <i>checkout</i> dan mengirimkan bukti transfer pembayaran produk.</p> <ul style="list-style-type: none"> <li>• Melihat riwayat pembelian Pembeli dapat melihat riwayat pembelian yang telah selesai.</li> <li>• Melihat <i>list</i> pembelian yang belum dibayar Pembeli dapat melihat daftar pembelian yang belum dibayar.</li> <li>• <i>Logout</i> Pembeli dapat keluar dari aplikasi dengan menekan tombol <i>logout</i>.</li> </ul>
4	Kurir	<ul style="list-style-type: none"> <li>• <i>Login</i> Kurir dapat melakukan login dengan menginputkan <i>email</i> dan <i>password</i> yang sesuai dengan data yang ada pada <i>database</i>.</li> <li>• Menginputkan bukti pengiriman Kurir dapat mengakses halaman pengiriman dan menginputkan bukti pengiriman.</li> <li>• Melihat riwayat pengiriman Kurir dapat melihat riwayat pengiriman yang telah diselesaikan.</li> </ul>

#### 4.1.6 Use Case Scenario

*Use case scenario* menjelaskan mengenai langkah-langkah yang akan dilakukan user ketika akan melakukan fungsional di dalam aplikasi. *Use case scenario* yang akan ditampilkan pada sub bab ini sebanyak tiga *use case scenario*, yaitu mengelola perizinan UMKM, melakukan pembelian produk dan mengelola transaksi. *Use case scenario* lainnya dapat dilihat pada Lampiran B.

#### 4.1.6.1 Use Case Scenario Kelola Perizinan UMKM

*Use case scenario* kelola perizinan UMKM ini dilakukan oleh aktor admin ketika ingin mengkonfirmasi pengajuan akun dari penjual. *Use case scenario* kelola perizinan UMKM ini dapat dilihat pada Tabel 4.2 berikut.

Tabel 4. 2 *Use Case Scenario* Kelola Perizinan UMKM

<i>Use case</i>	Kelola perizinan UMKM
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil login
<i>Flow of Event</i>	<ol style="list-style-type: none"><li>1. Aktor memilih halaman pengajuan perlu konfirmasi</li><li>2. Sistem menampilkan halaman pengajuan yang perlu konfirmasi</li><li>3. Aktor mengklik tombol konfirmasi pengajuan pada data pengajuan yang ingin dikonfirmasi</li><li>4. Sistem menampilkan <i>list</i> pilihan untuk menerima pengajuan atau menolak pengajuan</li><li>5. Aktor memilih terima pengajuan</li><li>6. Sistem menyimpan data konfirmasi</li></ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"><li>7. Aktor menekan pilihan menolak pengajuan</li><li>8. Sistem menyimpan data konfirmasi penolakan</li></ol>
<i>Exit Condition</i>	Sistem menampilkan halaman pengajuan

#### 4.1.6.2 Use Case Scenario Melakukan Pembelian Produk

*Use case scenario* melakukan pembelian produk ini dilakukan oleh aktor pembeli ketika ingin membeli produk yang tersedia. *Use case scenario* melakukan pembelian produk dapat dilihat pada Tabel 4.3 berikut.

Tabel 4. 3 *Use Case Scenario* Melakukan Pembelian Produk

<i>Use case</i>	Melakukan pembelian produk
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor menekan produk yang akan dibeli</li> <li>2. Sistem menampilkan halaman rincian produk</li> <li>3. Aktor menginputkan variasi ukuran dan jumlah produk yang ingin dibeli</li> <li>4. Aktor menekan tombol tambah ke keranjang</li> <li>5. Sistem menampilkan halaman keranjang pembeli</li> <li>6. Aktor menekan tombol <i>checkout</i></li> <li>7. Sistem menampilkan halaman ringkasan pesanan</li> <li>8. Aktor menekan tombol buat pesanan</li> <li>9. Sistem menampilkan rincian transaksi dan tenggat waktu pembayaran</li> <li>10. Aktor melakukan pembayaran</li> <li>11. Sistem menampilkan konfirmasi pembayaran berhasil</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>12. Aktor tidak jadi melakukan pembayaran</li> <li>13. Sistem menampilkan pesan pembelian <i>expired</i></li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman pembelian produk

#### 4.1.6.3 *Use Case Scenario* Mengelola Transaksi

*Use case scenario* mengelola transaksi ini dilakukan oleh aktor penjual ketika ingin mengkonfirmasi pesanan yang masuk. *Use case scenario* mengelola transaksi dapat dilihat pada Tabel 4.4 berikut.

Tabel 4. 4 *Use Case Scenario* Mengelola Transaksi

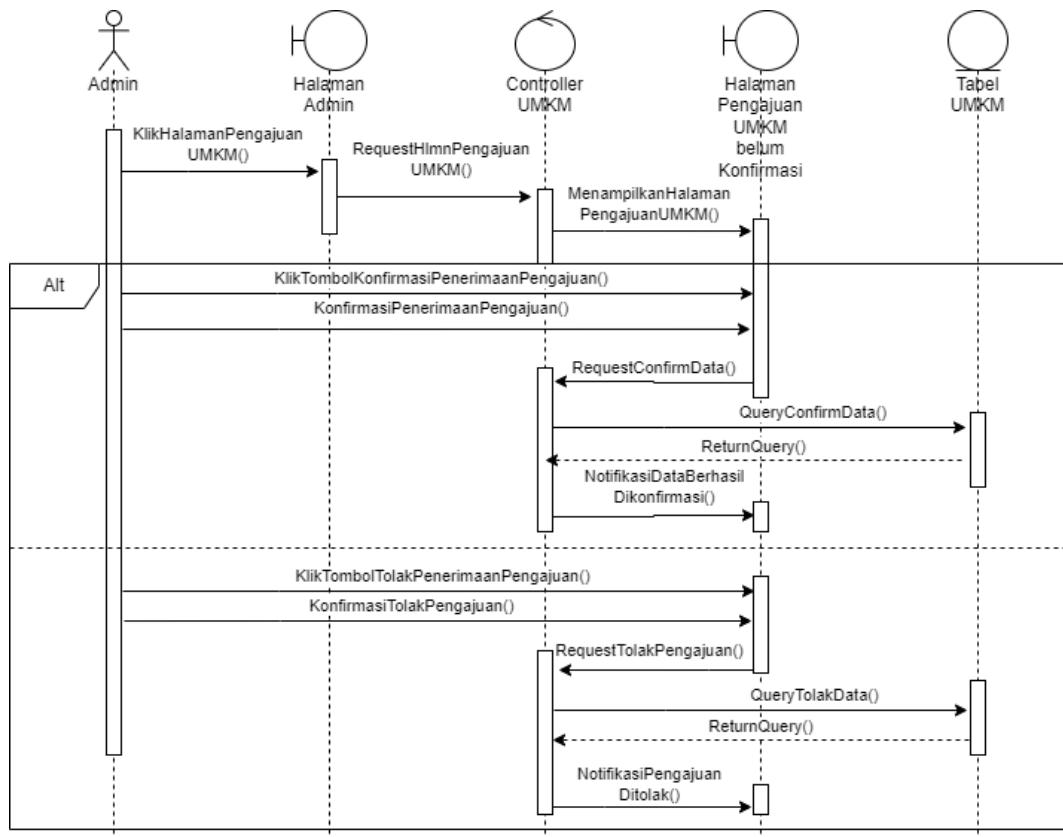
<i>Use case</i>	Mengelola transaksi
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman pemesanan</li> <li>2. Sistem menampilkan halaman daftar pesanan dari pembeli</li> <li>3. Aktor mengecek bukti pembayaran yang dikirimkan dengan mengakses halaman detail transaksi</li> <li>4. Aktor mengkonfirmasi pesanan</li> <li>5. Sistem menyimpan data konfirmasi pesanan</li> <li>6. Aktor mengklik detail transaksi pada transaksi yang sudah dikonfirmasi</li> <li>7. Sistem menampilkan halaman detail transaksi</li> <li>8. Aktor menginputkan noresi</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>9. Aktor mengkonfirmasi tolak transaksi</li> <li>10. Sistem menyimpan data konfirmasi penolakan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman transaksi

#### 4.1.7 *Sequence Diagram*

Pada sub bab ini terdapat interaksi semua proses yang terjadi didalam sistem yang dijelaskan didalam *sequence diagram*. *Sequence diagram* yang akan ditampilkan pada sub bab ini sebanyak tiga *sequence diagram*, yaitu mengelola perizinan UMKM, melakukan pembelian produk dan mengelola transaksi. *Sequence diagram* lainnya dapat dilihat pada Lampiran C.

#### 4.1.7.1 Sequence Diagram Kelola Perizinan UMKM

Sequence diagram kelola perizinan UMKM dapat dilihat pada Gambar 4.9 dibawah ini.

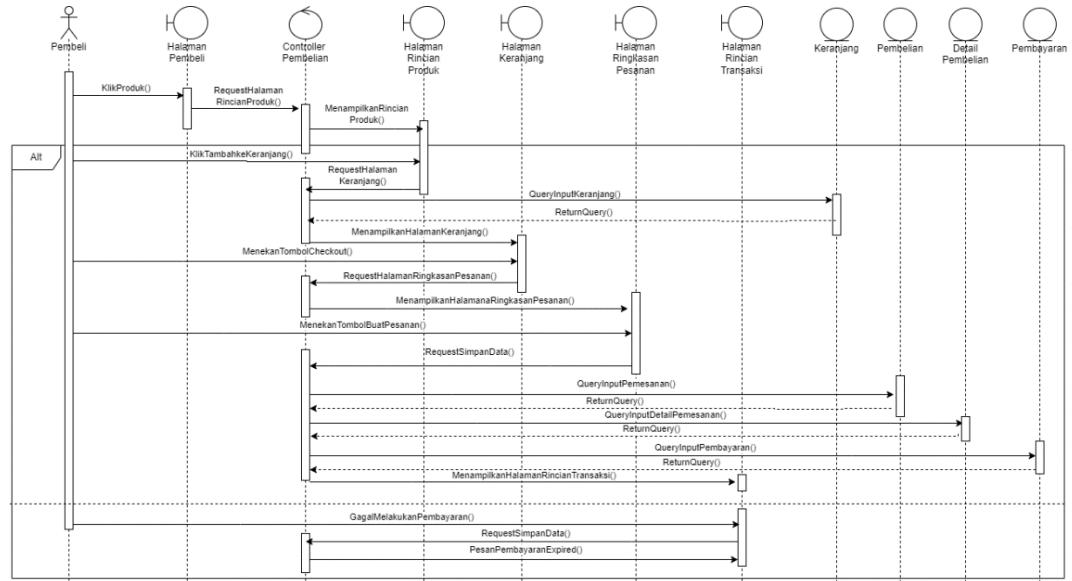


Gambar 4. 9 Sequence Diagram Kelola Perizinan UMKM

Berdasarkan Gambar 4.9, dapat dijelaskan bahwa proses kelola perizinan UMKM dilakukan oleh admin. Admin mengakses halaman pengajuan UMKM, kemudian admin menekan tombol konfirmasi terima pengajuan, data konfirmasi tersebut akan disimpan ke tabel UMKM melalui controller UMKM. Jika admin menekan tombol tolak pengajuan, maka konfirmasi tolak pengajuan juga akan dikirimkan ke tabel UMKM melalui controller UMKM.

#### 4.1.7.2 Sequence Diagram Melakukan Pembelian Produk

Sequence diagram melakukan pembelian produk dapat dilihat pada Gambar 4.10 dibawah ini.

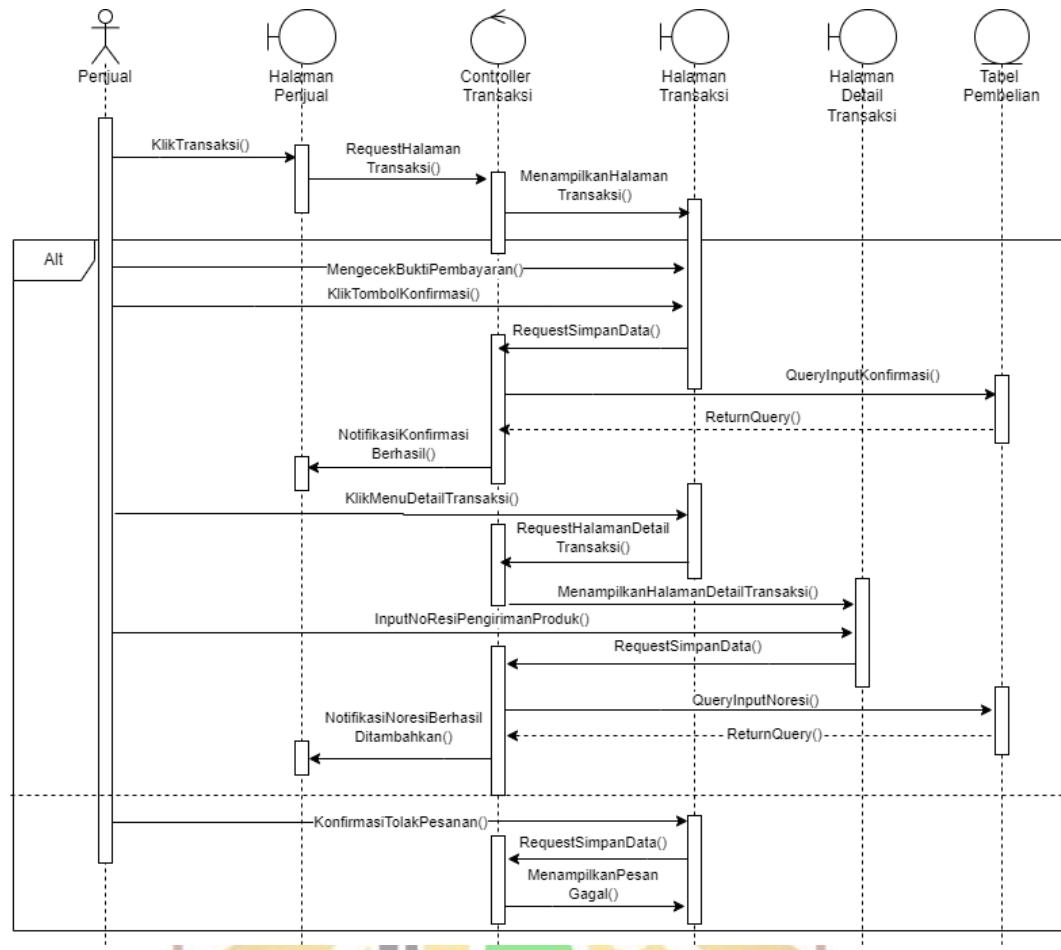


Gambar 4. 10 *Sequence Diagram* Melakukan Pembelian Produk

Berdasarkan Gambar 4.10, dapat dijelaskan bahwa proses pembelian dilakukan oleh pembeli. Proses dimulai ketika pembeli menekan tombol rincian produk, kemudian sistem akan menampilkan halaman rincian produk, lalu pembeli akan menekan tombol tambah ke keranjang. Pembeli akan menekan tombol *checkout* di halaman keranjang, di halaman checkout akan muncul detail dari total biaya yang akan dibayar kemudian pembeli menekan tombol buat pesanan sehingga akan muncul halaman ringkasan pesanan lalu pembeli akan diarahkan untuk menginputkan bukti pembayaran. Jika pembeli tidak melakukan pembayaran selama rentang waktu yang ditentukan maka akan muncul pesan *expired*.

#### 4.1.7.3 *Sequence Diagram* Mengelola Transaksi

*Sequence diagram* mengelola transaksi dapat dilihat pada Gambar 4.11 dibawah ini.



Gambar 4. 11 Sequence Diagram Mengelola Transaksi

Berdasarkan Gambar 4.11, dapat dijelaskan bahwa proses pengelolaan transaksi dilakukan oleh penjual. Proses dimulai ketika penjual mengakses halaman transaksi, kemudian pada halaman tersebut penjual akan mengecek bukti transaksi yang dikirimkan oleh pembeli. Jika bukti valid, maka penjual akan mengkonfirmasi transaksi dan hasil konfirmasi akan disimpan ke tabel pembelian melalui *controller* transaksi. Jika telah dikonfirmasi valid, maka penjual akan membuka halaman detail transaksi dan menambahkan noresi ke transaksi tersebut. Inputan noresi ini akan disimpan ke tabel pembelian melalui *controller* transaksi. Jika bukti tidak valid, maka penjual akan melakukan konfirmasi tolak pesanan.

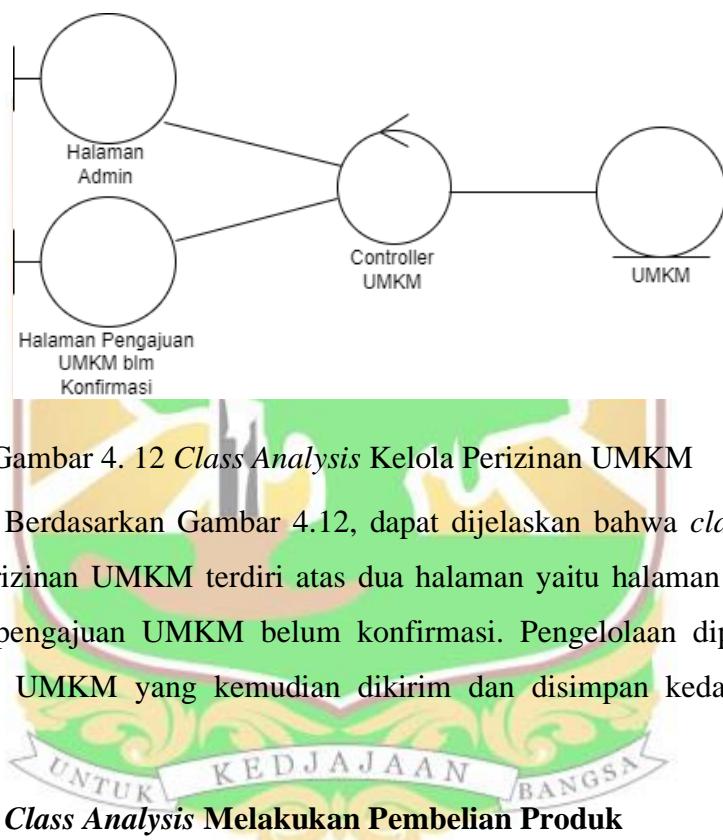
#### 4.1.8 Class Analysis

Pada sub bab ini dijelaskan *class analysis* yang menggambarkan semua kelas yang terlibat dalam setiap proses dan *class analysis* ini dibuat

berdasarkan hasil *sequence diagram* yang terdapat pada sub bab 4.1.7. *Class analysis* yang akan ditampilkan pada sub bab ini sebanyak tiga *class analysis*, yaitu mengelola perizinan UMKM, melakukan pembelian produk dan mengelola transaksi. *Class analysis* lainnya dapat dilihat pada Lampiran D.

#### 4.1.8.1 *Class Analysis Kelola Perizinan UMKM*

*Class analysis* kelola perizinan UMKM dapat dilihat pada Gambar 4.12 dibawah ini.

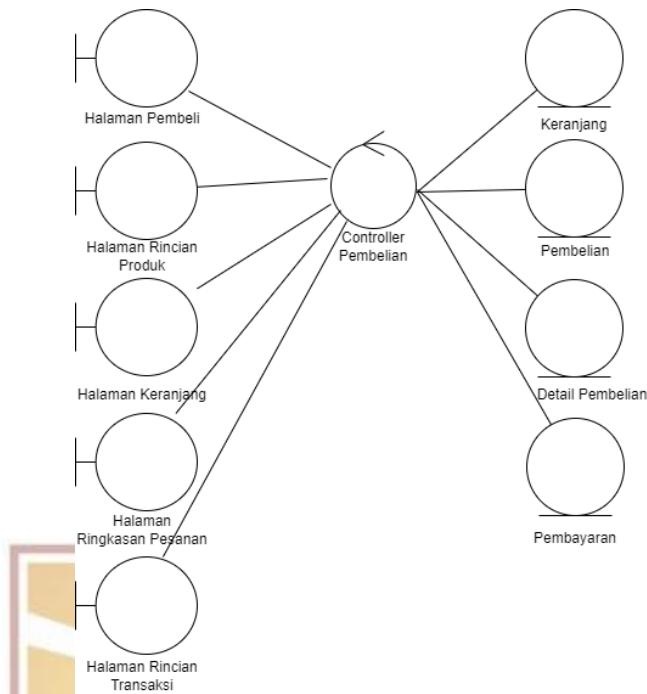


Gambar 4. 12 *Class Analysis Kelola Perizinan UMKM*

Berdasarkan Gambar 4.12, dapat dijelaskan bahwa *class analysis* kelola perizinan UMKM terdiri atas dua halaman yaitu halaman admin dan halaman pengajuan UMKM belum konfirmasi. Pengelolaan diproses oleh *controller* UMKM yang kemudian dikirim dan disimpan kedalam model UMKM.

#### 4.1.8.2 *Class Analysis Melakukan Pembelian Produk*

*Class analysis* melakukan pembelian produk dapat dilihat pada Gambar 4.13 dibawah ini.

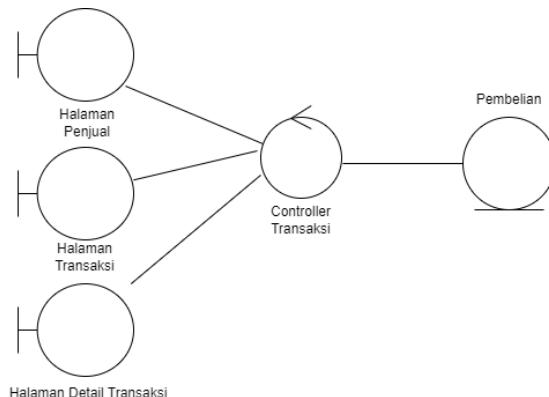


Gambar 4. 13 *Class Analysis* Melakukan Pembelian Produk

Berdasarkan Gambar 4.13, dapat dijelaskan bahwa *class analysis* melakukan pembelian produk terdiri atas lima halaman yaitu halaman pembeli, halaman rincian produk, halaman keranjang, halaman ringkasan pesanan dan halaman rincian transaksi. Pengelolaan diproses oleh *controller* pembelian yang kemudian dikirim dan disimpan kedalam model keranjang dan model pembelian.

#### 4.1.8.3 *Class Analysis* Mengelola Transaksi

*Class analysis* mengelola transaksi dapat dilihat pada Gambar 4.14 dibawah ini.



Gambar 4. 14 *Class Analysis* Mengelola Transaksi

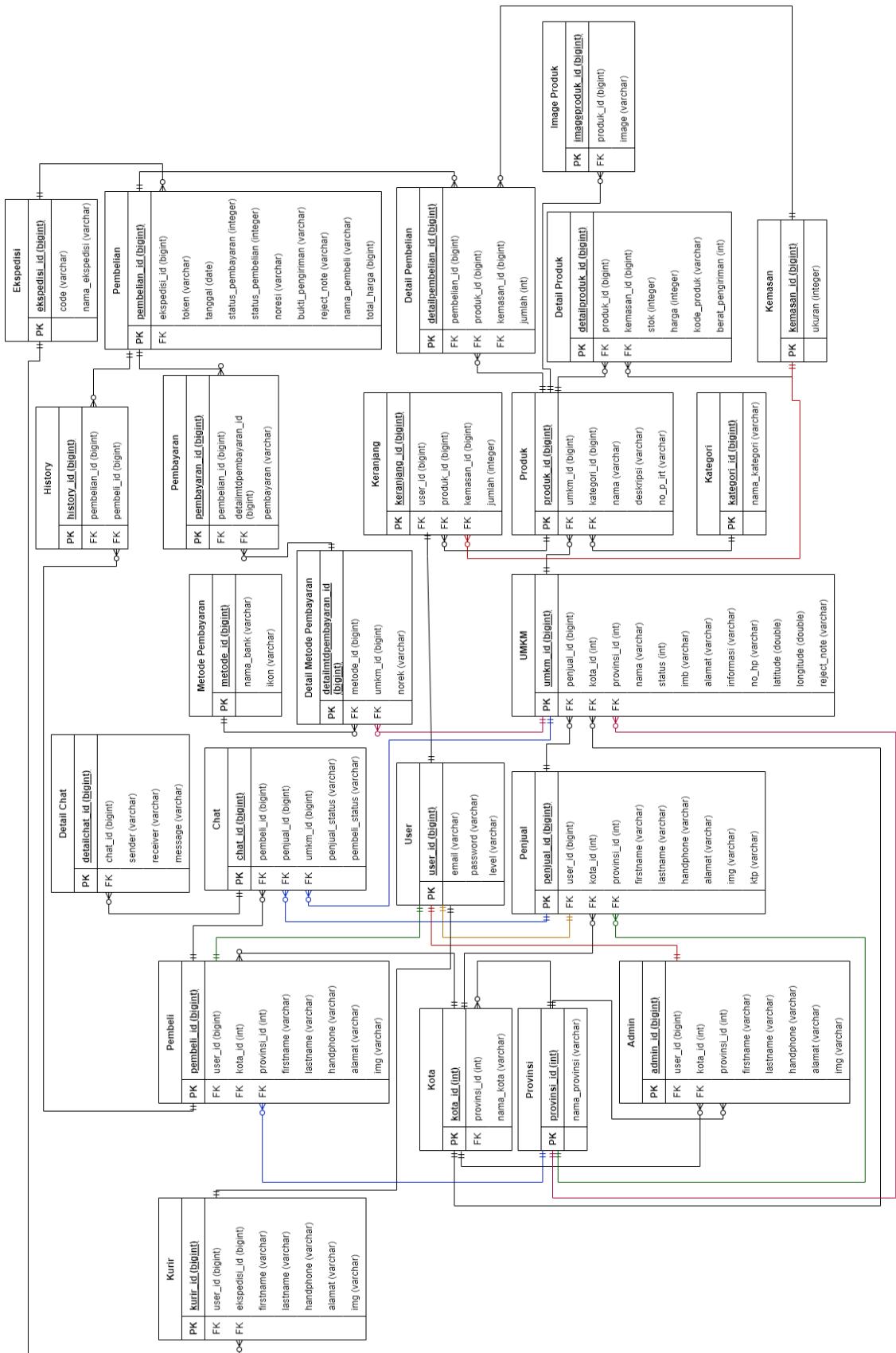
Berdasarkan Gambar 4.14, dapat dijelaskan bahwa *class analysis* mengelola transaksi terdiri atas tiga halaman yaitu halaman penjual, halaman transaksi dan halaman detail transaksi. Pengelolaan diproses oleh *controller* transaksi yang kemudian dikirim dan disimpan kedalam model pembelian.

## 4.2 Perancangan Sistem

Pada sub bab ini dijelaskan mengenai perancangan sistem yang akan dibuat, perancangan ini dibuat berdasarkan hasil analisis dari tahapan sebelumnya. Perancangan ini terdiri atas perancangan *database* dalam bentuk ERD (*Entity Relationship Diagram*), struktur tabel *database*, arsitektur aplikasi, *class diagram* dan perancangan *user interface* dari aplikasi yang dibangun.

### 4.2.1 Perancangan Database

Perancangan *database* dilakukan untuk mengatur alur data yang dibutuhkan dalam proses bisnis yang terjadi didalam aplikasi. Perancangan *database* ini dilakukan menggunakan pemodelan dengan tools dari ERD (*Entity Relationship Diagram*) yang bertujuan untuk menghubungkan antar entitas berdasarkan atributnya. Pada tahapan ini, sistem memiliki dua puluh tiga tabel yang terdiri dari tabel *user*, admin, penjual, pembeli, kurir, umkm, kota, provinsi, *chat*, detail *chat*, produk, detail produk, *image* produk, kategori, kemasan, keranjang, pembelian, detail pembelian, ekspedisi, metode pembayaran, pembayaran, detail metode pembayaran, *history*. Berikut rancangan *database* aplikasi dengan model bisnis *marketplace* UMKM yang digambarkan dalam model ERD dapat dilihat pada Gambar 4.15



Gambar 4. 15 Perancangan Database

Berdasarkan Gambar 4.14 diatas, dapat dilihat bahwa setiap tabel memiliki atribut dan relasinya dengan tabel lain sesuai dengan kebutuhan sistem.

#### 4.2.2 Struktur Tabel Database

Struktur tabel *database* merupakan lanjutan dari ERD (*Entity Relationship Diagram*) dimana terdapat nama, atribut, tipe data, ukuran serta keterangan dari atribut tersebut. Struktur tabel *database* ini terdiri atas dua puluh tiga tabel sesuai dengan ERD diatas. Struktur tabel *database* yang ditampilkan pada sub bab ini sebanyak empat tabel, yaitu tabel pembeli, tabel UMKM, tabel prduk dan tabel pembelian. Struktur tabel *database* lainnya dapat dilihat pada Lampiran E.

Tabel 4. 5 Struktur Tabel Pembeli

Nama Atribut	Tipe Data	Ukuran	Keterangan
Pembeli_id	Bigint	20	Primary Key
User_id	Bigint	20	Foreign Key
Kota_id	Bigint	20	Foreign Key
Provinsi_id	Bigint	20	Foreign Key
Firstname	Varchar	255	
Lastname	Varchar	255	
Handphone	Varchar	13	
Alamat	Varchar	255	
Img	Varchar	255	

Tabel 4. 6 Struktur Tabel UMKM

Nama Atribut	Tipe Data	Ukuran	Keterangan
Umkm_id	Bigint	20	Primary Key
Penjual_id	Bigint	20	Foreign Key
Kota_id	Bigint	20	Foreign Key
Provinsi_id	Bigint	20	Foreign Key
Nama	Varchar	255	

Tabel 4.6 Struktur Tabel UMKM (Lanjutan)

Status	Integer	4	
Imb	Varchar	255	
Alamat	Varchar	255	Unique
Informasi	Varchar	255	
handphone	Varchar	13	
Latitude	Double	-	
Longitude	Double	-	
Reject_note	Varchar	255	

Tabel 4.7 Struktur Tabel Produk

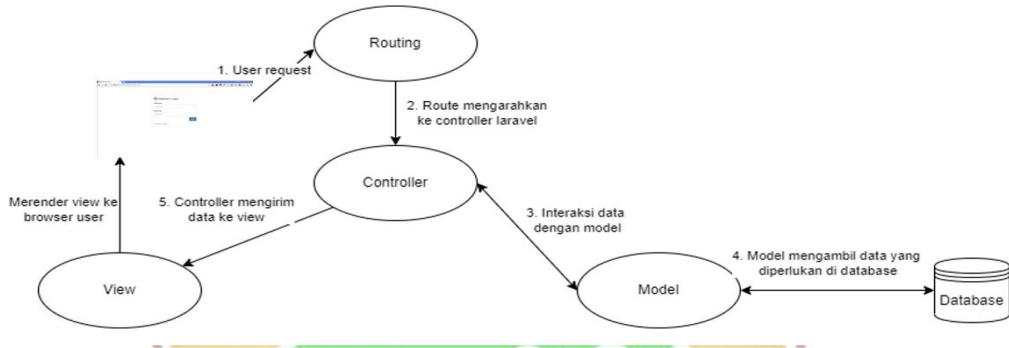
Nama Atribut	Tipe Data	Ukuran	Keterangan
Produk_id	Bigint	20	Primary Key
Umkm_id	Bigint	20	Foreign Key
Kategori_id	Bigint	20	Foreign Key
Nama	Varchar	10	
Deskripsi	Varchar	10	
No_p_irt	Varchar	255	Unique

Tabel 4.8 Struktur Tabel Pembelian

Nama Atribut	Tipe Data	Ukuran	Keterangan
Pembelian_id	Varchar	10	Primary Key
Ekspedisi_id	Varchar	10	Foreign Key
Token	Varchar	10	Unique
Nama_pembeli	varchar	255	
Status_pembayaran	Varchar	4	
Status_pembelian	Varchar	4	
Reject_note	Varchar	255	
Bukti_pengiriman	Varhcarr	255	
Noresi	Varchar	255	
Total_harga	Varchar	255	

#### 4.2.3 Arsitektur Aplikasi

Pada perancangan pembangunan aplikasi dengan model bisnis *marketplace* ini, arsitektur yang akan digunakan yaitu arsitektur aplikasi MVC (*Model View Controller*) dengan menggunakan *framework* laravel yang dapat dilihat pada Gambar 4.16 berikut ini.

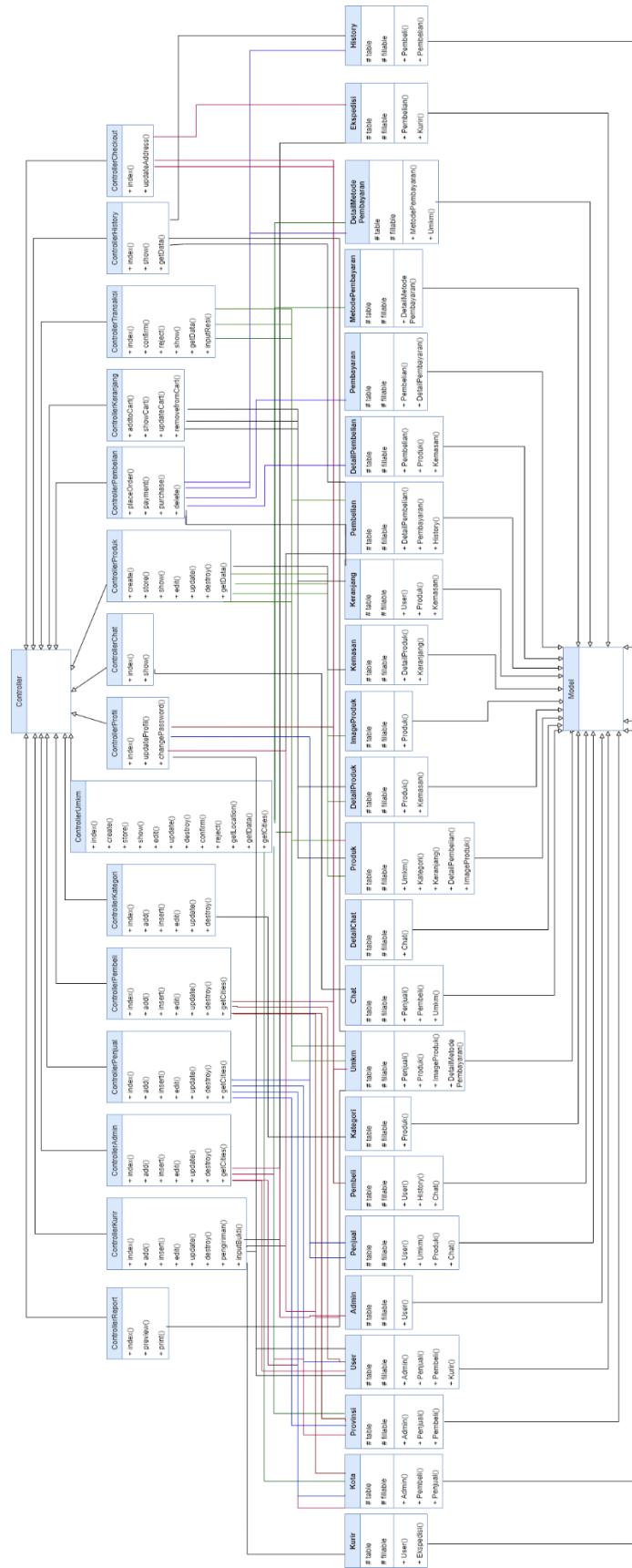


Gambar 4. 16 Arsitektur Aplikasi MVC

Berdasarkan Gambar 4.16 dapat dilihat bahwa dalam arsitektur aplikasi ini memisahkan antara tiga komponen yaitu *model*, *view* dan *controller*. Alur dari MVC ini yaitu ketika *user* melakukan *request* di *view*, *request* tersebut akan dilanjutkan ke *routing* laravel. *Router* akan memproses *request* dari *user* tersebut dan mengarahkannya ke *controller* berdasarkan dengan *class* dan *method* yang dibutuhkan. *Controller* melakukan komunikasi dengan *model* jika ada data yang dibutuhkan sesuai *request* dari *user*. Data yang berhasil ditemukan oleh *model* di *database* akan dikirim kembali ke *controller* dan *controller* akan melakukan *render* *view* terhadap data sehingga data tersebut akan dikonversi menjadi HTML dan dikirimkan kembali ke *view browser* sehingga *user* dapat melihat respon atas *request* yang telah dilakukan.

#### 4.2.4 Class Diagram

*Class diagram* berperan dalam menggambarkan proses yang berjalan pada aplikasi secara umum. Pada *class diagram* juga terdapat komponen *model*, *view* dan *controller*. *Class diagram* ini menggambarkan struktur sistem dengan menunjukkan sistem *class* beserta atribut, *method*, dan hubungan antar objeknya. *Class diagram* dari pembangunan aplikasi ini dapat dilihat pada Gambar 4.17 berikut ini.



Gambar 4. 17 *Class Diagram* Aplikasi

#### 4.2.5 Perancangan Antarmuka Aplikasi

Pada sub bab ini dijelaskan gambaran mengenai rancangan antarmuka sistem atau yang biasa dikenal dengan *user interface*. Design yang dibuat merupakan penghubung antara user dan sistem, user akan berkomunikasi dengan sistem melalui tampilan tersebut. Perancangan antarmuka aplikasi yang akan ditampilkan pada sub bab ini sebanyak tiga antarmuka aplikasi yaitu, perancangan antarmuka data UMKM, perancangan antarmuka home pembeli, perancangan antarmuka keranjang. Perancangan antarmuka lainnya dapat dilihat pada Lampiran F.

##### 4.2.5.1 Perancangan Antarmuka Data UMKM

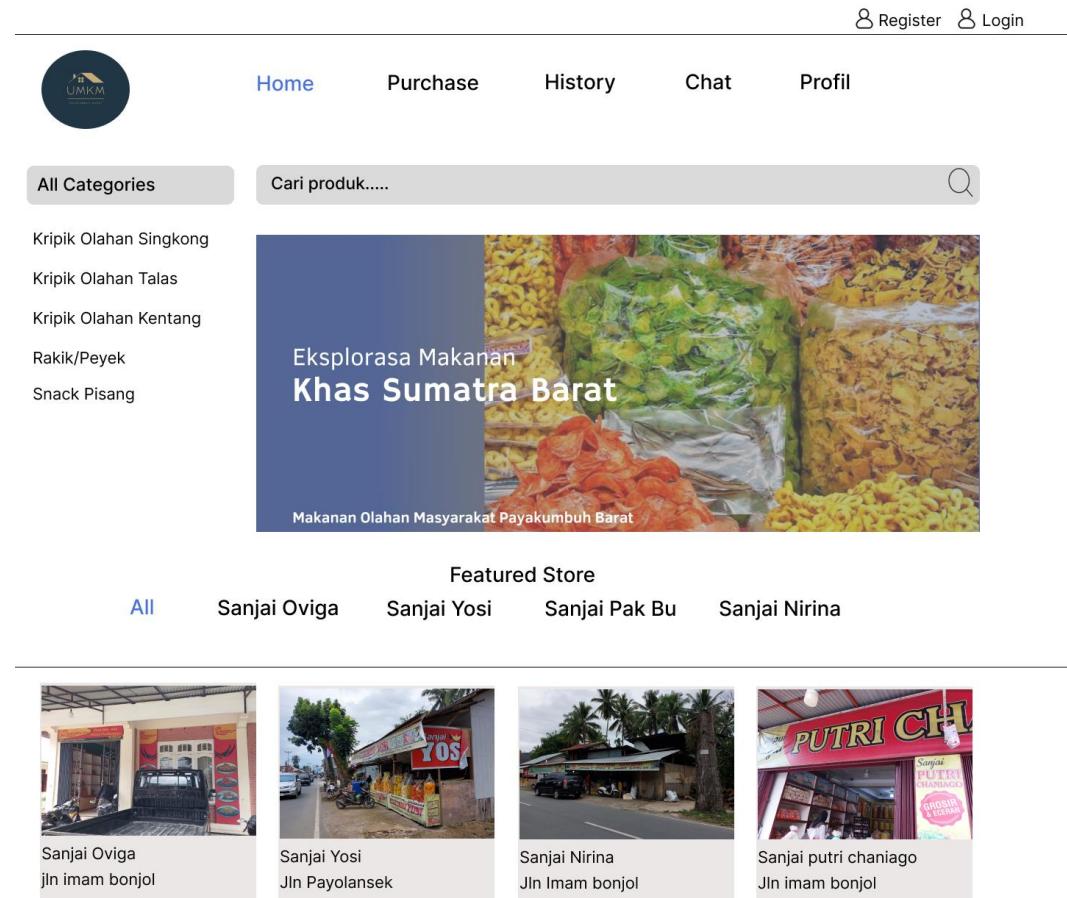
Perancangan antarmuka data UMKM ini dapat diakses oleh penjual. Halaman ini berisi daftar UMKM yang telah ditambahkan, terdapat tombol tambah, detail, hapus beserta informasi status dari UMKM tersebut. Tombol tambah untuk menambahkan data UMKM, tombol detail untuk menampilkan informasi detail mengenai UMKM dan tombol hapus untuk menghapus data UMKM. Perancangan antarmuka data UMKM dapat dilihat pada Gambar 4.18 dibawah ini.



Gambar 4. 18 Perancangan Antarmuka Data UMKM

#### 4.2.5.2 Perancangan Antarmuka *Home* Pembeli

Perancangan antarmuka *home* pembeli dapat diakses oleh pembeli untuk melihat daftar produk dari berbagai UMKM yang tersedia. Perancangan antarmuka *home* pembeli dapat dilihat pada Gambar 4.19 dibawah ini.



Gambar 4. 19 Perancangan Antarmuka Home

#### 4.2.5.3 Perancangan Antarmuka Keranjang

Perancangan antarmuka keranjang ini dapat diakses oleh pembeli untuk meletakkan produk-produk yang akan dibeli dan terdapat tombol *checkout* untuk melanjutkan pembelian. Perancangan antarmuka keranjang dapat dilihat pada Gambar 4.20 dibawah ini.

[Home](#)[Purchase](#)[History](#)[Chat](#)[Profil](#)

Product	Price	Quantity	Total
	Karak Kaliang Rp10.000,-	4	Rp40.000,-

[Continue Shopping](#)[Update Cart](#)

Cart Total  
Subtotal Rp40.000,-

[Proceed to Checkout](#)



Gambar 4. 20 Perancangan Antarmuka Keranjang

## BAB V

### IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini dijelaskan mengenai implementasi sistem berdasarkan perancangan yang telah dibuat sebelumnya, kemudian dilakukan pengujian terhadap aplikasi tersebut untuk memastikan bahwa aplikasi yang dibangun ini telah sesuai serta tidak ada kesalahan dan kecacatan pada fungsionalnya.

#### 5.1 Implementasi Sistem

Aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh ini dapat mempermudah para pemilik UMKM dalam mempromosikan dan memasarkan produknya. Dalam aplikasi ini dapat terjadi transaksi jual beli antara pembeli dengan penjual tanpa harus bertemu secara langsung serta dapat melakukan pembelian dari jarak jauh sehingga lebih efektif dari sisi waktu. Dengan adanya aplikasi ini juga dapat membantu pihak pemilik UMKM dalam mengelola transaksi penjualan secara *online* dengan lebih efisien. Pada aplikasi ini terdapat tiga aktor yang terlibat dan sembilan belas fungsional yang dapat dijalankan dan diharapkan dapat memberikan efektivitas dan efisiensi terhadap proses bisnis yang diharapkan.

Implementasi aplikasi ini dilakukan dengan menggunakan perangkat keras komputer dengan spesifikasi sebagai berikut.

1. Laptop dengan *processor* Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 1.99 GHz
2. *Random Access Memory* (RAM) 8 GB
3. *Solid State Drive* (SSD) laptop dengan kapasitas 120 GB

Spesifikasi perangkat lunak yang digunakan dalam implementasi sistem ini adalah sebagai berikut.

1. Sistem operasi Windows 10
2. *Web browser* google chrome versi 116.0.5845.96

Spesifikasi system requirements yang dibutuhkan adalah sebagai berikut.

1. *Web server* dengan modul PHP 8.0.18 atau setelahnya
2. *Web browser* yang mendukung HTML 5

Implementasi aplikasi ini menggunakan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan *framework* Laravel dengan versi 9.52.8. Sedangkan, *web server* yang digunakan adalah *web server apache* XAMPP versi 2.4.47 dan *database* yang digunakan yaitu *database* MySQL. *Web server* ini berfungsi untuk menjalankan bahasa pemrograman PHP pada *browser* dan mengolah data pada *database*, sedangkan *database* berfungsi untuk penyimpanan data. Pada sub bab ini terdiri dari pengkodean program serta implementasi antarmuka aplikasi.

### 5.1.1 Pengkodean Program

Pada bagian pengkodean program ini dijelaskan kode program yang dibuat berdasarkan perancangan aplikasi yang telah dibuat. Terdiri atas bagian *view*, *model* dan *controller* yang dijelaskan secara terpisah berdasarkan fungsinya masing-masing. Berikut diberikan contoh kode program implementasi berdasarkan *routing*, *view*, *model* dan *controller* dimana kode program implementasi lainnya dijelaskan pada lampiran G.

#### 5.1.1.1 Kode Program *Routing* Aplikasi

Pada bagian ini dijelaskan mengenai *routing* aplikasi dimana *routing* ini adalah sebuah proses yang mengatur arah perpindahan dari permintaan *user* pada aplikasi. *Route* ini menangani request *user* melalui url yang kemudian mengarahkannya ke sebuah *method* pada *controller* tertentu yang menghasilkan halaman sesuai *request user*. Kode program *routing* aplikasi dapat dilihat pada Gambar 5.1 berikut.

```
Route::get('/', function () {
    return view('landing.welcome');
});

Route::get('/', [LandingController::class, 'index'])->name('landing.index');
Route::get('/search', [LandingController::class, 'search'])->name('search');
Route::get('/searchproduk', [LandingController::class, 'searchproduk'])->name('searchproduk');
Route::get('/category/{id}', [LandingController::class, 'showCategory'])->name('category');
Route::get('/kategoriumkm/{umkmId}/{kategoriId}', [LandingController::class, 'showKategoriUkm'])->name('kategoriumkm');
Route::get('auth/google', [LoginWithGoogleController::class, 'redirectToGoogle']);
Route::get('auth/google/callback', [LoginWithGoogleController::class, 'HandleGoogleCallback']);
Route::get('/landing/produk/{id}', [LandingController::class, 'showDetail'])->name('detail-produk');
Route::get('/landing/umkm/{id}', [LandingController::class, 'showDetailUkm'])->name('detail-umkm');

Route::group(['middleware' => ['auth', 'level:admin']], function(){
    Route::get('/dashboardadmin', function () {
        return view('dashboardadmin');
    })->name('dashboardadmin');
    //kelola user
    Route::delete('/dataadmin/{id}/delete', [AdminController::class, 'destroy'])->name('dataadmin.destroy');
    Route::get('/dataadmin', [AdminController::class, 'index'])->name('dataadmin');
    Route::get('/dataadmin/create', [AdminController::class, 'create']);
    Route::get('/dataadmin/add', [AdminController::class, 'add']);
    Route::get('/provinsi/{provincisId}/city', [AdminController::class, 'getCities']);
    Route::post('/dataadmin/insert', [AdminController::class, 'insert']);

    Route::get('/dataadmin/edit/{id}', [AdminController::class, 'edit'])->name('admin.edit');
    Route::post('/dataadmin/update/{id}', [AdminController::class, 'update']);
    Route::delete('/datapenjual/{id}/delete', [PenjualController::class, 'destroy'])->name('datapenjual.destroy');
    Route::get('/datapenjual', [PenjualController::class, 'index'])->name('datapenjual');
    Route::get('/datapenjual/create', [PenjualController::class, 'create']);
});
```

Gambar 5. 1 Kode Program Routing Aplikasi

```

Route::get('/datapenjual/add', [PenjualController::class, 'add']);
Route::get('/province/{provinsiId}/cities', [PenjualController::class, 'getCities']);
Route::post('/datapenjual/insert', [PenjualController::class, 'insert']);

Route::get('/datapenjual/edit/{id}', [PenjualController::class, 'edit'])->name('penjual.edit');
Route::post('/datapenjual/update/{id}', [PenjualController::class, 'update']);
Route::get('/datapembeli', [PembeliController::class, 'index'])->name('datapembeli');
Route::get('/datapembeli/create', [PembeliController::class, 'create']);
Route::get('/datapembeli/add', [PembeliController::class, 'add']);
Route::get('/province/{provinsiId}/cities', [PembeliController::class, 'getCities']);
Route::post('/datapembeli/insert', [PembeliController::class, 'insert']);
Route::delete('/datapembeli/{id}/delete', [PembeliController::class, 'destroy'])->name('datapembeli.destroy');
Route::get('/datapembeli/edit/{id}', [PembeliController::class, 'edit'])->name('pembeli.edit');
Route::post('/datapembeli/update/{id}', [PembeliController::class, 'update']);

//kelola kategori
Route::get('/kategori', [KategoriController::class, 'index'])->name('kategori');
Route::get('/kategori/create', [KategoriController::class, 'create']);
Route::get('/kategori/add', [KategoriController::class, 'add']);
Route::post('/kategori/insert', [KategoriController::class, 'insert']);
Route::delete('/kategori/{id}/delete', [KategoriController::class, 'destroy'])->name('kategori.destroy');
Route::get('/kategori/edit/{id}', [KategoriController::class, 'edit']);
Route::post('/kategori/update/{id}', [KategoriController::class, 'update']);

//profil
Route::get('/profiladmin', [ProfilController::class, 'index'])->name('profiladmin');
Route::post('/profiladmin/update', [ProfilController::class, 'updateProfile'])->name('profiladmin.update');
Route::post('/profiladmin/change-password', [ProfilController::class, 'changePassword'])->name('change.password.admin');

//kelolaumkm
Route::get('/umkm/index-admin', [UmkmController::class, 'index_admin'])->name('umkm.index.admin');
Route::patch('/umkm/{id}/confirm', [UmkmController::class, 'confirm'])->name('admin.umkm.confirm');
Route::patch('/umkm/{id}/reject', [UmkmController::class, 'reject'])->name('admin.umkm.reject');
Route::get('/umkm/{id}/show-index', [UmkmController::class, 'show_index'])->name('umkm.show-index');

Route::get('/umkm/index1-admin', [UmkmController::class, 'index1_admin'])->name('umkm.index1.admin');
Route::get('/umkm/{id}/show1-index', [UmkmController::class, 'show1_index'])->name('umkm.show1-index');
Route::patch('/umkm/{id}/banned', [UmkmController::class, 'banned'])->name('admin.umkm.banned');

Route::get('/umkm/index2-admin', [UmkmController::class, 'index2_admin'])->name('umkm.index2.admin');
Route::get('/umkm/{id}/show2-index', [UmkmController::class, 'show2_index'])->name('umkm.show2-index');

Route::get('/umkm/index3-admin', [UmkmController::class, 'index3_admin'])->name('umkm.index3.admin');
Route::get('/umkm/{id}/show3-index', [UmkmController::class, 'show3_index'])->name('umkm.show3-index');

e::group(['middleware' => ['auth','level:penjual']], function(){
    Route::get('/dashboard', function () {
        return view('dashboard');
    })->name('dashboard');
    //profil
    Route::get('/province/{provinsiId}/cities', [PenjualController::class, 'getCities']);
    Route::get('/profilpenjual', [ProfilController::class, 'index'])->name('profilpenjual');
    Route::post('/profilpenjual/update', [ProfilController::class, 'updateProfile'])->name('profilpenjual.update');
    Route::post('/profilpenjual/changepassword', [ProfilController::class, 'changePassword'])->name('change.password');
    //umkm
    Route::resource('/umkm', UmkmController::class);
    Route::put('/umkm/{id}', [UmkmController::class, 'update'])->name('umkm.update');
    //produk
    Route::resource('/produk', ProdukController::class);
    Route::get('/umkm/{id}/product-create', [ProdukController::class, 'productCreate'])->name('umkm.product.create');
    Route::post('/umkm/{id}/product-store', [ProdukController::class, 'productStore'])->name('umkm.product.store');
    Route::put('/umkm/{id}/product-update', [ProdukController::class, 'productUpdate'])->name('umkm.product.update');
    Route::get('/umkm/{id}/product-show', [ProdukController::class, 'productShow'])->name('umkm.product.show');
    Route::post('/tambah-stok/{id}', [ProdukController::class, 'tambahStok'])->name('tambah-stok');
    Route::get('/api/detail-produk', [ProdukController::class, 'showDetail']);
    Route::get('/fetch-detail/{productId}/{kemasanId}', [ProdukController::class, 'fetchDetail']);
})

```

Gambar 5.1 Kode Program *Routing* Aplikasi (Lanjutan)

```

//transaksi
Route::resource('/transaksi', TransaksiController::class);
Route::get('/transaksi/{id}/show', [TransaksiController::class, 'show'])->name('transaksi.show');
Route::patch('/transaksi/{id}/finish', [TransaksiController::class, 'finish'])->name('penjual.transaksi.finish');
Route::patch('/transaksi/{id}/confirm', [TransaksiController::class, 'confirm'])->name('penjual.transaksi.confirm');
Route::patch('/transaksi/{id}/reject', [TransaksiController::class, 'reject'])->name('penjual.transaksi.reject');
Route::post('/input-noresi/{id}', [TransaksiController::class, 'inputResi'])->name('input-noresi');
//history
Route::resource('/history', HistoryController::class);
Route::get('/history/{id}/show', [HistoryController::class, 'show'])->name('history.show');
//chat
Route::get('/chat/index-penjual', [ChatController::class, 'index_penjual'])->name('chat.index.penjual');
Route::get('/chatch/{id}', [ChatController::class, 'show'])->name('chat.show.penjual');
//report
Route::get('/report/index', [ReportController::class, 'index'])->name('report.index');
Route::post('/report/preview', [ReportController::class, 'preview'])->name('report.preview');
Route::post('/report/print', [ReportController::class, 'print'])->name('report.print');

});

Route::group(['middleware' => ['auth','level:pembeli']], function(){
    Route::post('/add-to-cart', [KeranjangController::class, 'addToCart'])->name('add-to-cart');
    Route::get('/keranjang', [KeranjangController::class, 'showCart'])->name('keranjang');
    Route::delete('/remove-from-cart/{id}', [KeranjangController::class, 'removeFromCart'])->name('remove-from-cart');
    Route::patch('/update-cart', [KeranjangController::class, 'updateCart'])->name('update-cart');
    Route::get('/checkout', [CheckoutController::class, 'index'])->name('checkout');
    Route::post('/checkout/update', [CheckoutController::class, 'updateAddress'])->name('checkout.update');
    Route::resource('/chat', ChatController::class);
    Route::get('/chat/{id}', [ChatController::class, 'show'])->name('chat.show');
    Route::post('/place-order', [PembelianController::class, 'placeOrder'])->name('place.order');
    Route::get('/payment/{id}', [PembelianController::class, 'payment'])->name('payment');
    Route::post('/post-payment', [PembelianController::class, 'postPayment'])->name('post.payment');
    Route::get('/purchase', [PembelianController::class, 'purchase'])->name('purchase');

    Route::group(['middleware' => ['auth','level:kurir']], function(){
        Route::get('/dashboardkurir', function () {
            return view('dashboardkurir');
        })->name('dashboardkurir');

        Route::get('/pengiriman', [KurirController::class, 'pengiriman'])->name('pengiriman');
        Route::get('/pengiriman/{id}/show', [KurirController::class, 'show'])->name('pengiriman.show');
        Route::patch('/pengiriman/{id}/finish', [KurirController::class, 'finish'])->name('kurir.pengiriman.finish');
        Route::post('/input-bukti/{id}', [KurirController::class, 'inputBukti'])->name('input-bukti');

        Route::get('/historykurir', [KurirController::class, 'historyPengiriman'])->name('historykurir');
        Route::get('/historykurir/{id}/show', [KurirController::class, 'historyPengirimanShow'])->name('history.kurir.show');

    });

    Route::get('/payment/{id}', [PembelianController::class, 'payment'])->name('payment');
    Route::post('/post-payment', [PembelianController::class, 'postPayment'])->name('post.payment');
    Route::get('/purchase', [PembelianController::class, 'purchase'])->name('purchase');
    Route::delete('/purchase/{id}', [PembelianController::class, 'delete'])->name('pembelian.delete');

    //history
    Route::get('/pembeli/history', [HistoryController::class, 'index_pembeli'])->name('pembeli.history.index');
    Route::get('/pembeli/history/{id}', [HistoryController::class, 'show_pembeli'])->name('pembeli.show.history');

    //profil
    Route::get('/profilpembeli', [ProfilpembController::class, 'index'])->name('profilpembeli');
    Route::post('/profilpembeli/update', [ProfilpembController::class, 'updateProfile'])->name('profilpembeli.update');
    Route::post('/profilpembeli/changepassword', [ProfilpembController::class, 'changePassword'])->name('change.password');
    Route::get('/provinsi/{provinsiId}/cityy', [PembeliController::class, 'getCities']);

});

Route::get('logout', function ()
{
    auth()->logout();
    Session()->flush();

    return Redirect::to('/');
})->name('logout');

```

Gambar 5.1 Kode Program *Routing* Aplikasi (Lanjutan)

### 5.1.1.2 Kode Program *Controller* Aplikasi

Pada bagian ini dijelaskan mengenai *controller* dimana *controller* ini adalah sebuah fungsi di dalam laravel yang berguna untuk mengatur seluruh alur dan *method* pada aplikasi. *Controller* berisi logika dalam penyusunan dalam fungsi tertentu yang prosesnya diatur didalam *controller* tertentu. Salah satu *controller* yang dibangun adalah UmkmController. Kode program UmkmController dapat dilihat pada Gambar 5.2 berikut.

```
3  namespace App\Http\Controllers;
4
5  use App\Models\Detail_Metode_Pembayaran;
6  use App\Models\Umkm;
7  use App\Models\Provinsi;
8  use App\Models\Kota;
9  use App\Models\Metode_Pembayaran;
10 use Auth;
11 use Illuminate\Support\Facades\Log;
12 use Illuminate\Http\Request;
13 use App\Http\Resources\UmkmList;
14 use File;
15 use RealRashid\SweetAlert\Facades\Alert;
16 use Illuminate\Support\Facades\DB;
17
18 class UmkmController extends Controller
19 {
20     public function index()
21     {
22         try {
23             $umkms = Umkm::where('penjual_id', Auth::user()->penjual->id)
24                         ->orderBy('id', 'desc')
25                         ->get();
26
27             $umkmData = [];
28             foreach ($umkms as $umkm) {
29                 if ($request->hasFile('img')) {
30                     $filee = Request->file('img');
31                     $filename = Request->firstname . '.' . $filee->extension();
32                     $filee->move(public_path('img_umkm'), $filename);
33                     $umkm->img = $filename;
34                 }
35             }
36             $umkmData[] = $umkm;
37         }
38
39         Log::info("User " . Auth::user()->Penjual->firstname . " " . Auth::user()->Penjual->lastname . " Berhasil mengakses halaman Umkm");
40         return view('penjual.manage_umkm.index', compact('umkmData'));
41
42     } catch (\Exception $e) {
43         Log::error("User " . Auth::user()->Penjual->firstname . " " . Auth::user()->Penjual->lastname . " Gagal mengakses halaman Umkm");
44         return redirect()->back();
45     }
46
47     public function index_admin()
48     {
49         try{
50             return view('admin.manage_umkm.index');
51         } catch (\Exception $e){
52             return redirect()->back()->with('error', __('toats.index.failed.message'));
53         }
54     }
55
56     public function index1_admin()
57     {
58         try{
59             return view('admin.manage_umkm.index1');
60         } catch (\Exception $e){
61             return redirect()->back()->with('error', __('toats.index.failed.message'));
62         }
63     }
64
65     public function index2_admin()
66     {
67         try{
68             return view('admin.manage_umkm.index2');
69         } catch (\Exception $e){
70             return redirect()->back()->with('error', __('toats.index.failed.message'));
71         }
72     }
73
74     public function index3_admin()
75     {
76         try{
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
617
618
619
619
620
621
622
623
623
624
625
625
626
626
627
627
628
628
629
629
630
630
631
631
632
632
633
633
634
634
635
635
636
636
637
637
638
638
639
639
640
640
641
641
642
642
643
643
644
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
```

```

79     |         return view('admin.manage_umkm.index3');
80   }
81   catch (\Exception $e){
82   |       return redirect()->back()->with('error', __('toats.index.failed.message'));
83   }
84 }
85
86 public function create()
87 {
88
89     $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
90
91     $kotaId = old('kota');
92     $kotaName = Kota::where('city_id', $kotaId)->value('nama_kota');
93
94     $kotas = [];
95     if (!empty($kotaId)) {
96         $kotas = Kota::where('province_id', $kotaId)->pluck('nama_kota', 'city_id');
97     }
98     $metode_pembayaran = Metode_Pembayaran::all();
99     try{
100         return view('penjual.manage_umkm.add', compact('metode_pembayaran','provinsi', 'kotas'));
101     } catch (\Exception $e) {
102         return redirect()->back();
103     }
104 }
105
106 public function getCities($id)
107 {
108     $kota = Kota::where('province_id', $id)->pluck('nama_kota', 'city_id');
109     return response()->json($kota);
110 }
111
112 public function store(Request $request)
113 {
114     try {
115         $umkm = new Umkm;
116         $umkm->nama = $request->nama;
117         $umkm->province_id = $request->provinsi;
118         $umkm->province_id = $request->provinsi;
119         $umkm->city_id = $request->kota;
120         $umkm->alamat = $request->alamat;
121         $umkm->informasi = $request->informasi;
122         $umkm->handphone = $request->handphone;
123         $umkm->latitude = $request->latitude;
124         $umkm->longitude = $request->longitude;
125         $umkm->penjual_id = Auth::user()->penjual->id;
126
127         $dir = public_path().'/imb';
128         $file = $request->file('imb');
129         if ($file) {
130             $fileName = time() . '.' . $file->getClientOriginalName();
131             $file->move($dir, $fileName);
132             $umkm->imb = $fileName;
133         }
134
135         $dirr = public_path().'/img_umkm';
136         $filee = $request->file('img');
137         if ($filee) {
138             $fileNamee = time() . '.' . $filee->getClientOriginalName();
139             $filee->move($dirr, $fileNamee);
140             $umkm->img = $fileNamee;
141         }
142
143         $umkm->save();
144
145         //Payment method
146         if($request->metode_pembayaran){
147             for($i=0; $i < count($request->metode_pembayaran); $i++){
148                 $DetailMetodePembayaran = new Detail_Metode_Pembayaran;
149                 $DetailMetodePembayaran->umkm_id = $umkm->id;
150                 $DetailMetodePembayaran->metodepemb_id = $request->metode_pembayaran[$i];
151                 $DetailMetodePembayaran->norek = $request->norek[$i];
152                 $DetailMetodePembayaran->save();
153             }
154         }
155         Alert::success('Berhasil!', 'Data umkm berhasil ditambahkan!');
156         return redirect()->route('umkm.index')->with('success', __('toast.create.success.message')));
157     }
158 }
```

Gambar 5.2 Kode Program *Controller* Aplikasi (Lanjutan)

```

156 } catch (\Exception $e) {
157     // dd($e);
158     Alert::error('Gagal!', 'Data umkm gagal ditambahkan! Alamat sudah ada!');
159     return redirect()->back()->with('error', __('toast.create.failed.message'));
160 }
161 }
162 }
163
164 public function getData(Request $request)
165 {
166     $data = [];
167     if ($request->data == "all") {
168         $data = Umkm::where('status', $request->status)
169             ->orderby('id', 'desc')
170             ->get();
171     } elseif ($request->data == "id") {
172         $data = Umkm::find($request->id);
173     } elseif ($request->data == "select") {
174         $id = explode(',', $request->id);
175         $data = Umkm::wherenotin('id', $id)->get();
176     }
177
178     if ($data) {
179         Log::debug('Data berhasil ditemukan');
180         return response()->json(UmkmList::collection($data));
181     }
182     Log::error('Data tidak ditemukan');
183     return response()->json($data);
184 }
185
186
187 public function confirm($id)
188 {
189     try {
190         $umkm = Umkm::find($id);
191         $umkm->status = 1;
192         $umkm->save();
193
194         return redirect()->route('umkm.index.admin')->with('success', __('toast.confirm.success.message'));
195     } catch (\Exception $e) {
196         return redirect()->back()->with('error', __('toast.confirm.failed.message'));
197     }
198 }
199
200 public function reject(Request $request, $id)
201 {
202     try {
203         $umkm = Umkm::find($request->umkm_id);
204         $umkm->status = 2;
205         $umkm->reject_note = $request->reject_note;
206         $umkm->save();
207
208         return redirect()->route('umkm.index.admin')->with('success', __('toast.reject.success.message'));
209     } catch (\Exception $e) {
210         dd($e);
211         return redirect()->back()->with('error', __('toast.reject.failed.message'));
212     }
213 }
214
215 public function banned($id)
216 {
217     try {
218         $umkm = Umkm::find($id);
219         $umkm->status = 3;
220         $umkm->save();
221
222         return redirect()->route('umkm.index.admin')->with('success', __('toast.confirm.success.message'));
223     } catch (\Exception $e) {
224         return redirect()->back()->with('error', __('toast.confirm.failed.message'));
225     }
226 }
227
228 public function edit(Request $request, $id)
229 {
230     try{
231         $umkms = Umkm::find($id);
232         if($request->data){
233             $data = 'request';
234         } else{

```

Gambar 5.2 Kode Program *Controller* Aplikasi (Lanjutan)

```

235 |     $data = 'none';
236 |
237 |     $detail_metode_pembayaran = Detail_Metode_Pembayaran::where('umkm_id', $umkms->id)->get();
238 |     $metode_pembayaran = Metode_Pembayaran::all();
239 |     //dd($payment_method_details);
240 |
241 |     $provinis = Provinsi::pluck('nama_provinsi', 'province_id');
242 |     $kotas = Kota::where('province_id', $umkms->province_id)->pluck('nama_kota', 'city_id');
243 |
244 |     Log::info("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengakses halaman edit umkm");
245 |     return view('penjual.manage_umkm.edit',
246 |         compact('umkms', 'detail_metode_pembayaran','data','provinis', 'kotas'));
247 | } catch (\Exception $e) {
248 |
249 |     Log::error("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Gagal mengakses halaman edit umkm");
250 |     return redirect()->back();
251 | }
252 |
253 | public function update(Request $request, $id)
254 |
255 | {
256 |     try {
257 |         $umkm = Umkm::find($id);
258 |         $umkm->nama = $request->nama;
259 |
260 |         if($request->data == 'request'){
261 |             $umkm->status = 0;
262 |         }
263 |         $umkm->alamat = $request->alamat;
264 |         $umkm->province_id = $request->provinsi;
265 |         $umkm->city_id = $request->kota;
266 |         $umkm->alamat = $request->alamat;
267 |         $umkm->informasi = $request->informasi;
268 |         $umkm->handphone = $request->handphone;
269 |         $umkm->latitude = $request->latitude;
270 |         $umkm->longitude = $request->longitude;
271 |
272 |         if($request->file('imb')){
273 |             //Rule upload
274 |             if($umkm->imb){
275 |
276 |                 if(File::exists(public_path('imb/'.$umkm->imb))){
277 |                     File::delete(public_path('imb/'.$umkm->imb));
278 |
279 |                 }
280 |
281 |                 $dir = public_path().'/imb';
282 |                 $fileImbUpload = $request->file('imb');
283 |
284 |                 $fileName = Time().".". $fileImbUpload->getClientOriginalName();
285 |                 $fileImbUpload->move($dir, $fileName);
286 |
287 |                 if($fileImbUpload){
288 |                     $umkm->imb = $fileName;
289 |                 }
290 |
291 |             }else{
292 |                 $dir = public_path().'/imb';
293 |                 $fileImbUpload = $request->file('imb');
294 |
295 |                 $fileName = Time().".". $fileImbUpload->getClientOriginalName();
296 |                 $fileImbUpload->move($dir, $fileName);
297 |
298 |                 if($fileImbUpload){
299 |                     $umkm->imb = $fileName;
300 |                 }
301 |
302 |             }
303 |
304 |             if($request->file('img')){
305 |                 //Rule upload
306 |                 if($umkm->img){
307 |                     if(File::exists(public_path('img_umkm/'.$umkm->img))){
308 |                         File::delete(public_path('img_umkm/'.$umkm->img));
309 |
310 |                     }
311 |
312 |                     $dirr = public_path().'/img_umkm';
313 |                     $fileImgUpload = $request->file('img');
314 |
315 |                     $filenamee = Time().".". $fileImgUpload->getClientOriginalName();
316 |                     $fileImgUpload->move($dirr, $filenamee);
317 |
318 |                     if($fileImgUpload){
319 |                         $umkm->img = $filenamee;
320 |                     }
321 |                 }
322 |             }
323 |         }
324 |     }
325 | }

```

Gambar 5.2 Kode Program *Controller* Aplikasi (Lanjutan)

```

314     }
315     }else{
316         $dirr = public_path().'/img_umkm';
317         $fileImgUpload = $request->file('img');
318
319         $fileNameee = Time().".". $fileImgUpload->getClientOriginalName();
320         $fileImbUpload->move($dirr, $fileNameee);
321
322         if($fileImbUpload){
323             $umkm->img = $fileNameee;
324         }
325     }
326 }
327 $umkm->save();
328 //Payment method
329 $DetailMetodePembayarans = Detail_Metode_Pembayaran::where('umkm_id', $umkm->id)->get();
330 foreach($DetailMetodePembayarans as $DetailMetodePembayaran){
331     $DetailMetodePembayaran->delete();
332 }
333
334 if($request->metode_pembayaran){
335     for($i=0; $i < count($request->metode_pembayaran); $i++){
336         $DetailMetodePembayaran = new Detail_Metode_Pembayaran;
337         $DetailMetodePembayaran->umkm_id = $umkm->id;
338         $DetailMetodePembayaran->metodepemb_id = $request->metode_pembayaran[$i];
339         $DetailMetodePembayaran->nrek = $request->nrek[$i];
340         $DetailMetodePembayaran->save();
341     }
342 }
343 Log::notice("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengedit data umkm");
344 if($request->data == 'request'){
345     Alert::success('Berhasil!', 'Data umkm berhasil diubah!');
346     return redirect()->route('umkm.index')->with('success', __('toast.update.success.message'));
347 }
348 Alert::success('Berhasil!', 'Data umkm berhasil diubah!');
349 return redirect()->route('umkm.show', $umkm->id)->with('success', __('toast.update.success.message'));
350 } catch (\Exception $e) {
351     Log::error("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Gagal mengedit data umkm");
352     return redirect()->back()->with('error', __('toast.update.failed.message'));
353 }
354 }
355 public function show($id)
356 {
357     try{
358         $umkms = Umkm::find($id);
359
360         Log::info("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengakses halaman detail umkm");
361         return view('penjual.manage_umkm.show', compact('umkms','id'));
362     }
363     catch(\Exception $e){
364         Log::error("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Gagal mengakses halaman detail umkm");
365         return redirect()->back();
366     }
367 }
368 public function show_index($id)
369 {
370     try{
371         $umkms = Umkm::find($id);
372
373         return view('admin.manage_umkm.show', compact('umkms','id'));
374     }
375     catch (\Exception $e){
376         return redirect()->back()->with('error', __('toats.index.failed.message'));
377     }
378 }
379 public function show1_index($id)
380 {
381     try{
382         $umkms = Umkm::find($id);
383
384         return view('admin.manage_umkm.show1', compact('umkms','id'));
385     }
386     catch (\Exception $e){
387         return redirect()->back()->with('error', __('toats.index.failed.message'));
388     }
389 }
390 public function show2_index($id)
391 {
392     ...
393 }

```

Gambar 5.2 Kode Program *Controller* Aplikasi (Lanjutan)

```

393     $umkms = Umkm::find($id);
394
395     return view('admin.manage_umkm.show2', compact('umkms','id'));
396 }
397 catch (\Exception $e){
398     return redirect()->back()->with('error', __('toats.index.failed.message'));
399 }
400 }
401
402 public function show3_index($id)
403 {
404     try{
405         $umkms = Umkm::find($id);
406
407         return view('admin.manage_umkm.show3', compact('umkms','id'));
408     }
409     catch (\Exception $e){
410         return redirect()->back()->with('error', __('toats.index.failed.message'));
411     }
412 }
413
414 public function destroy($id)
415 {
416     try{
417         $umkms = Umkm::find($id);
418         $umkms->delete();
419         return redirect()->route('umkm.index')->with('success', __('toast.delete.success.message'));
420     } catch (\Exception $e) {
421         return redirect()->back()->with('error', __('toast.delete.failed.message'));
422     }
423 }
424 public function getLocation(Request $request)
425 {
426     $data = [];
427     $data = Umkm::select(['latitude', 'longitude'])
428     ->where('id', $request->id)
429     ->get();
430
431     return response()->json($data);

```

Gambar 5.2 Kode Program *Controller* Aplikasi (Lanjutan)

### 5.1.1.3 Kode Program *Model* Aplikasi

Pada bagian ini dijelaskan mengenai *model* dimana *model* adalah penghubung antara *controller* dengan *database*, didalam model ini terdapat relasi relasi yang menghubungkan antar tabel pada *database*. Sehingga nantinya kita bisa mengambil data dari tabel berelasi untuk diproses dan ditampilkan ke *view*. Kode program *model* umkm dapat dilihat pada Gambar 5.3 berikut.

```

1  <?php
2  namespace App\Models;
3  use Illuminate\Database\Eloquent\Factories\HasFactory;
4  use Illuminate\Database\Eloquent\Model;
5  use Illuminate\Support\Facades\DB;
6
7  class Umkm extends Model
8  {
9      use HasFactory;
10     protected $table = 'umkm';
11
12     protected $fillable = [
13         'nama',
14         'province_id',
15         'city_id',
16         'alamat',
17         'handphone',
18         'latitude',
19         'longitude',
20         'penjual_id',
21         'imb',
22         'img',
23     ];
24
25     public function Penjual()
26     {
27         return $this->hasOne(Penjual::class, 'id', 'penjual_id');
28     }
29     public function Detail_Metode_Pembayaran()
30     {
31         return $this->hasMany(Detail_Metode_Pembayaran::class, 'umkm_id', 'id');
32     }
33     public function Produk()
34     {
35         return $this->hasMany(Produk::class, 'umkm_id', 'id');
36     }
37
38     public function Datacity($cityId)
39     {
40         return DB::table('umkm')
41             ->select('provinsi.nama_provinsi', 'kota.nama_kota')
42             ->leftJoin('provinsi', 'provinsi.id', '=', 'umkm.province_id')
43             ->leftJoin('kota', 'kota.city_id', '=', 'umkm.city_id')
44             ->where('kota.city_id', '=', $cityId)
45             ->first();
46     }
47     public function chat()
48     {
49         return $this->hasMany(Chat::class, 'umkm_id', 'id');
50     }
51 }

```

Gambar 5. 3 Kode Program *Model* Aplikasi

#### 5.1.1.4 Kode Program *View* Aplikasi

Pada bagian ini dijelaskan mengenai *view* dimana *view* adalah bagian dari *user interface* yang akan ditampilkan ke *user*. Bagian ini merupakan bagian yang berinteraksi langsung dengan para *user*. *View* menampilkan data yang diterima *controller* dari *model*. Salah satu *view* yang dibangun adalah *view index umkm* pada *user penjual* yang berfungsi menampilkan halaman daftar umkm yang dimiliki oleh

penjual. Kode program *view index* umkm pada *user* penjual dapat dilihat pada Gambar 5.4 berikut.

```
<div class="container-fluid">
    <div class="row">
        <div class="col-12">
            <div class="card">
                <div class="card-header">
                    <h3 class="card-title">List Data UMKM</h3>
                    @if (Auth::user()->penjual->ktp)
                        <a style="float:right; margin-right:5px;" href="{{ route('umkm.create') }}>
                            <button class="btn btn-primary" type="button">Tambah UMKM</button>
                        </a>
                    @else
                        <a style="float:right; margin-right:5px;" href="#">
                            <button onclick="warning()" class="btn btn-primary" type="button">Tambah UMKM</button>
                        </a>
                    @endif
                </div>
                <div class="card-body">
                    <div class="row sortable-card">
                        @foreach ($umkmData as $umkm)
                            <div class="col-4 col-sm-6 col-md-4 d-flex align-items-stretch flex-column">
                                <div class="card bg-light d-flex flex-fill">
                                    <div class="card-header text-muted border-bottom-0">
                                        <h2 class="lead"><b>{$umkm->nama}</b></h2>
                                    </div>
                                    <div class="card-body pt-0">
                                        <div class="row">
                                            <div class="col-7">
                                                <p class="text-muted text-sm"><b></b></p>
                                                <ul class="list-group mb-0 fa-ul text-muted">
                                                    <li class="medium"><span class="fa-li"><i class="mdi mdi-map-marker"></i></span>
                                                        Alamat: {$umkm->alamat}</li>
                                                    <li class="medium"><span class="fa-li"><i class="mdi mdi-cellphone-basic"></i></span> Phone:
                                                        {$umkm->handphone}</li>
                                                </ul>
                                            </div>
                                            <div class="col-5 text-center">
                                                
                                                @if($umkm->status == 0)
                                                    <span class="badge badge-info" style="width:100%">Pending</span>
                                                @elseif ($umkm->status == 1)
                                                    <span class="badge badge-success" style="width:100%">Aktif</span>
                                                @else
                                                    <span class="badge badge-danger" style="width:100%">Ditolak</span>
                                                @endif
                                            </div>
                                        </div>
                                    </div>
                                    <div class="card-footer">
                                        <div class="text-right">
                                            <a class="btn btn-sm btn-primary" @if ($umkm->status ==
                                                1)
                                                href="{{ route('umkm.show', $umkm->id) }}"
                                            @elseif ($umkm->status == 2) onclick="edit_umkm({{$umkm->id}})">
                                                href="javascript:void(0)"
                                            @elseif ($umkm->status == 0) onclick="confirm_umkm()"
                                                href="javascript:void(0)"
                                            @endif>
                                                View UMKM &nbsp;<i class="fas fa-arrow-right"></i>
                                            </a>
                                        </div>
                                    </div>
                                </div>
                            @endforeach
                        <!-- /.card-body -->
                    <!-- /.card -->
                </div>
            <!-- /.col -->
        </div>
    </div>

```



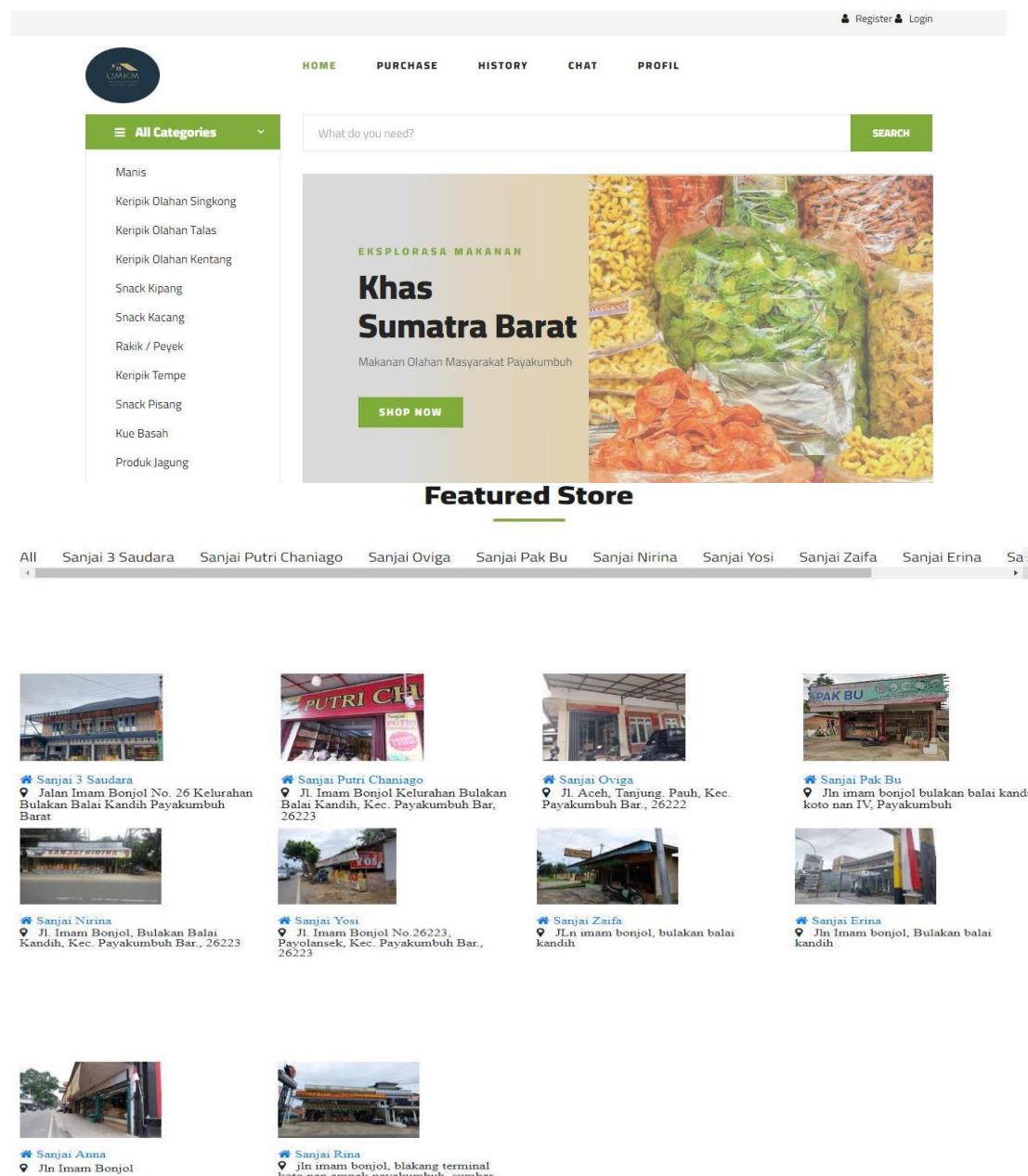
Gambar 5.4 Kode Program View Aplikasi

### 5.1.2 Implementasi Antarmuka Aplikasi

Implementasi antarmuka aplikasi adalah bagian yang menampilkan halaman depan suatu aplikasi yang dapat berinteraksi langsung dengan *user*. Implementasi antarmuka yang akan dijelaskan pada bagian ini yaitu halaman *landing page*, halaman *login*, halaman awal pembeli, halaman awal penjual dan halaman awal admin. Implementasi antarmuka lainnya akan dijelaskan pada Lampiran H.

### 5.1.2.1 Implementasi Antarmuka Halaman *Landing Page*

Implementasi antarmuka halaman *landing page* ini adalah implementasi halaman yang akan ditampilkan kepada *user* ketika mengakses aplikasi. Pada halaman ini terdapat informasi mengenai UMKM-UMKM yang tergabung dan juga mengenai produk-produk yang terdapat pada UMKM tersebut. Implementasi antarmuka halaman landing page dapat dilihat pada Gambar 5.5 berikut.



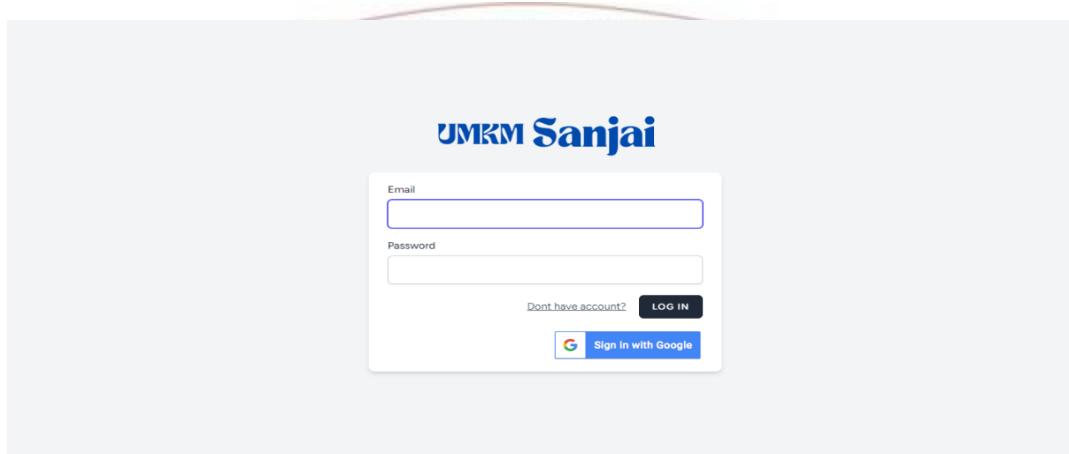
Gambar 5. 5 Implementasi Halaman Landing Page

Berdasarkan Gambar 5.5, halaman *landing page* ini menampilkan UMKM-UMKM yang tergabung dengan aplikasi, terdapat juga informasi kategori-kategori

produk yang berisikan produk-produk terkait dengan kategori terkait. Pada halaman ini, pembeli juga dapat melakukan pencarian berdasarkan nama produk dan nama UMKM.

#### **5.1.2.2 Implementasi Antarmuka Halaman *Login***

Implementasi antarmuka halaman *login* ini merupakan halaman yang dapat diakses user ketika mengklik tombol *login* pada halaman *landing page*. Halaman ini berisi inputan *email* dan *password* yang memungkinkan pengguna untuk masuk ke dalam sistem. Implementasi antarmuka halaman login dapat dilihat pada Gambar 5.6 berikut.

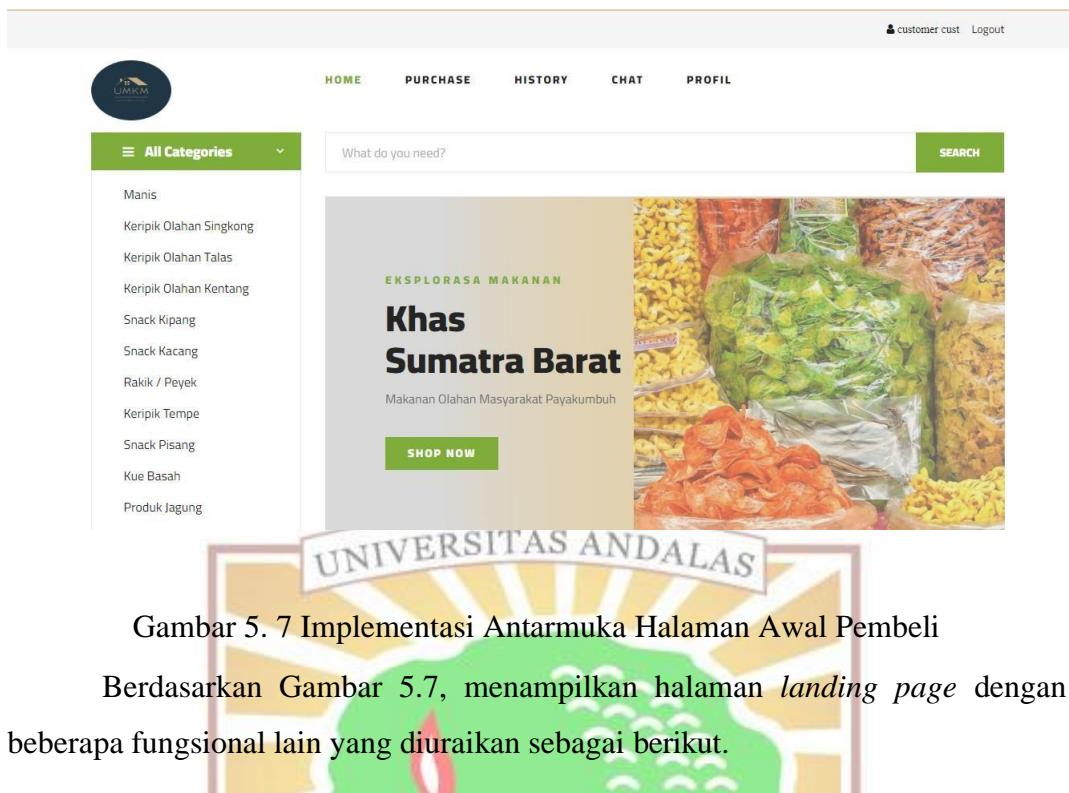


Gambar 5. 6 Implementasi Antarmuka Halaman Login

Berdasarkan Gambar 5.6, halaman ini menampilkan inputan berupa *email* dan *password* yang harus diisi oleh *user* ketika ingin masuk ke aplikasi, terdapat juga tombol “*Don't have account?*” untuk mengarahkan ke halaman registrasi jika belum memiliki akun.

#### **5.1.2.3 Implementasi Antarmuka Halaman Awal Pembeli**

Implementasi antarmuka halaman awal pembeli ini merupakan halaman yang menampilkan katalog UMKM yang bergabung dengan aplikasi ini. Halaman ini tampil setelah pembeli berhasil melakukan *login*. Implementasi antarmuka halaman awal pembeli dapat dilihat pada Gambar 5.7 berikut.



Gambar 5. 7 Implementasi Antarmuka Halaman Awal Pembeli

Berdasarkan Gambar 5.7, menampilkan halaman *landing page* dengan beberapa fungsional lain yang diuraikan sebagai berikut.

1. Menu *Home*

Menu ini adalah menu yang menampilkan halaman *landing page*, berupa informasi-informasi mengenai UMKM yang terdapat di aplikasi.

2. Menu *Purchase*

Menu ini adalah menu yang menampilkan daftar pesanan pembeli yang belum dibayar.

3. Menu *History*

Menu ini adalah menu yang menampilkan daftar pembelian yang telah dibayar oleh pembeli.

4. Menu *Chat*

Menu ini adalah menu yang menampilkan daftar *chat* pembeli dengan para penjual UMKM.

5. Menu *Profil*

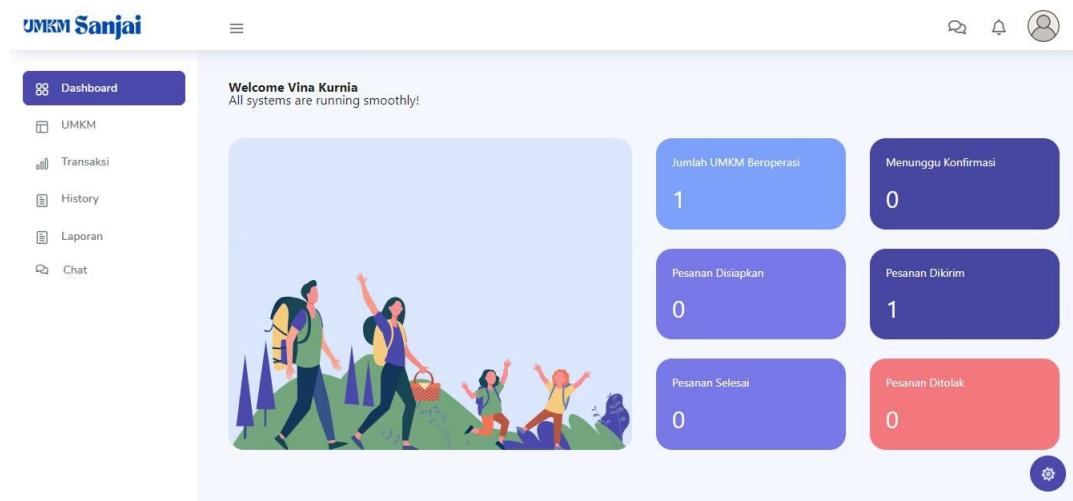
Menu yang menampilkan halaman profil serta halaman untuk mengubah *password user* pembeli.

6. Menu *Logout*

Menu *logout* ini adalah menu yang berfungsi ketika *user* pembeli ingin keluar dari aplikasi.

#### 5.1.2.4 Implementasi Antarmuka Halaman Awal Penjual

Implementasi antarmuka halaman awal penjual ini merupakan halaman yang menampilkan informasi mengenai UMKM yang dimiliki oleh penjual. Halaman ini dapat diakses ketika *user* penjual berhasil *login*. Implementasi antarmuka halaman awal penjual dapat dilihat pada Gambar 5. 8 berikut.



Gambar 5. 8 Implementasi Antarmuka Halaman Awal Penjual

Berdasarkan Gambar 5. 8, halaman awal penjual ini menampilkan data mengenai jumlah UMKM yang dimiliki serta menampilkan jumlah data transaksi pembelian produk. Pada *sidebar* terdapat beberapa fungsional yang diuraikan sebagai berikut.

##### 1. Menu *Dashboard*

Untuk menampilkan halaman awal penjual dengan berbagai informasi.

##### 2. Menu UMKM

Menu UMKM ini adalah menu yang menampilkan daftar UMKM yang dimiliki penjual tersebut, pada menu ini juga dapat melakukan pengelolaan menambah, mengedit serta menghapus UMKM tersebut.

##### 3. Menu Transaksi

Pada menu ini menampilkan daftar transaksi pembelian produk, penjual juga dapat melakukan pengelolaan melihat detail transaksi, konfirmasi pesanan, tolak pesanan serta menghapus transaksi.

##### 4. Menu *History*

Pada menu ini menampilkan halaman daftar pemesanan produk pada UMKM dan menampilkan detail pemesanannya.

#### 5. Menu Laporan

Pada menu ini menampilkan laporan transaksi berdasarkan rentang tanggal yang dipilih serta dapat mencetak laporan transaksinya.

#### 6. Menu Chat

Pada menu ini menampilkan daftar *chat* penjual dengan pembeli.

#### 7. Menu Profil

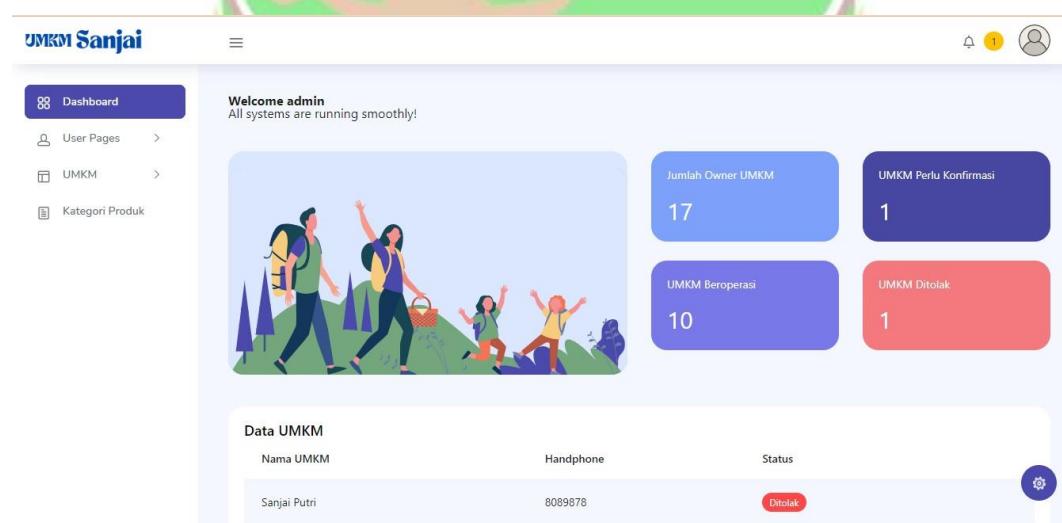
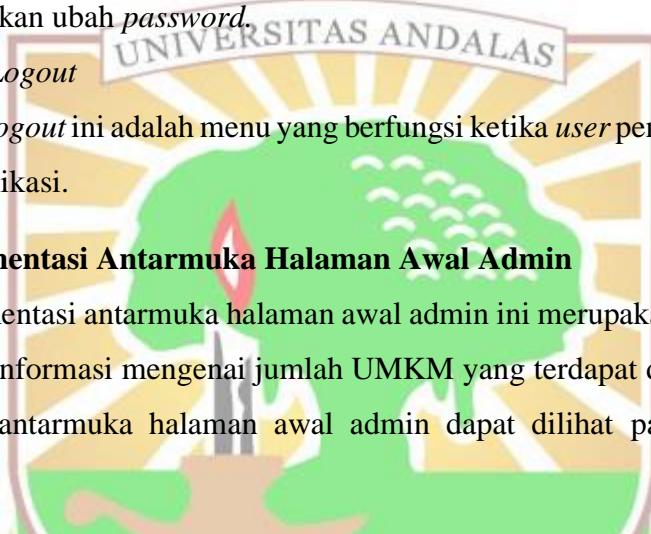
Menu yang menampilkan halaman profil *user* penjual serta dapat melakukan ubah *password*.

#### 8. Menu Logout

Menu *logout* ini adalah menu yang berfungsi ketika *user* penjual ingin keluar dari aplikasi.

#### 5.1.2.5 Implementasi Antarmuka Halaman Awal Admin

Implementasi antarmuka halaman awal admin ini merupakan halaman yang menampilkan informasi mengenai jumlah UMKM yang terdapat didalam aplikasi. Implementasi antarmuka halaman awal admin dapat dilihat pada Gambar 5.9 berikut.



Gambar 5. 9 Implementasi Antarmuka Halaman Awal Admin

Berdasarkan Gambar 5.9, halaman awal admin ini menampilkan informasi mengenai UMKM yang melakukan aktifitas pada aplikasi. Terdapat juga *sidebar* dengan fungsional yang diuraikan sebagai berikut.

1. Menu *Dashboard*

Menu *dashboard* adalah halaman utama yang pertama kali diakses admin ketika berhasil *login*. Berisi informasi-informasi mengenai UMKM.

2. Menu *User Pages*

Menu ini berisi pengelolaan *user* admin, pembeli dan penjual. Admin dapat melakukan pengelolaan tambah, edit dan hapus *user* pada halaman ini.

3. Menu UMKM

Menu UMKM ini adalah menu yang menampilkan daftar UMKM baik yang telah aktif, sedang proses konfirmasi, dan UMKM yang ditolak oleh admin.

4. Menu Kategori

Menu yang menampilkan halaman untuk melakukan pengelolaan kategori produk.

5. Menu Profil

Menu yang menampilkan halaman profil *user* admin serta dapat melakukan ubah *password*.

6. Menu *Logout*

Menu *logout* ini adalah menu yang berfungsi ketika *user* admin ingin keluar dari aplikasi.

## 5.2 Pengujian Sistem

Pengujian sistem ini adalah sebuah tahapan untuk memastikan bahwa aplikasi yang telah dibangun ini telah sesuai atau belum dengan perancangan yang diharapkan dan juga untuk mengecek apakah ada kesalahan pada fitur-fitur yang terdapat pada aplikasi ini. Pengujian sistem ini menggunakan metode *black box testing*, dimana metode ini berfokus pada pemeriksaan dari fungsional-fungsional

yang telah dibangun serta dapat memastikan bahwa aplikasi ini sudah memiliki fitur seperti yang diharapkan dan siap digunakan oleh *user* yang bersangkutan.

### 5.2.1 Fokus Pengujian

Pada bagian ini fokus pengujian dilakukan terhadap 21 item uji dari data yang telah didapatkan pada aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan. Fokus pengujian dapat dilihat pada Tabel 5.1 berikut.

Tabel 5. 1 Fokus Pengujian

No	Item Uji	User	Detail Pengujian
1.	<i>Login</i>	Admin, Penjual, Pembeli, Kurir	Verifikasi data <i>login</i>
2.	<i>Register</i>	Penjual, Pembeli	Tambah
3.	Kelola data <i>user</i>	Admin	Lihat, tambah, edit, hapus
4.	Ubah password	Admin, Penjual, Pembeli	Edit
5.	Kelola data profil	Admin, Penjual, Pembeli	Lihat, edit
6.	Kelola data kategori produk	Admin	Lihat, tambah, edit, hapus
7.	Kelola data UMKM	Penjual	Lihat, tambah, edit, hapus
8.	Kelola perizinan UMKM	Admin	Lihat, terima, tolak, <i>banned</i>
9.	Kelola data produk	Penjual	Lihat, tambah, edit, hapus
10.	Menambah stok produk	Penjual	Tambah
11.	Melakukan <i>chat</i>	Penjual, Pembeli	Tambah
12.	Melakukan penelusuran umkm dan produk	Pembeli	Lihat
13.	Melakukan pembelian produk	Pembeli	Lihat, tambah

Tabel 5.1 Fokus Pengujian (Lanjutan)

14.	Kelola data transaksi pembelian	Penjual	Lihat, terima, tolak
15.	Melihat laporan transaksi	Penjual	Lihat, cetak
16.	Melihat riwayat transaksi	Penjual, Pembeli	Lihat
17.	Menerima notifikasi pembayaran pesanan	Penjual	Lihat
18.	Menerima notifikasi UMKM perlu konfirmasi	Admin	Lihat
19.	Menginputkan bukti pengiriman	Kurir	Tambah
20.	Melihat riwayat pengiriman	Kurir	Lihat
21.	<i>Logout</i>	Admin, Penjual, Pembeli, Kurir	<i>Logout</i>

## 5.2.2 Kasus Hasil Pengujian

Pada kasus hasil pengujian dilakukan pengujian berdasarkan fungsional aplikasi. Kasus-kasus ini berpedoman pada fokus pengujian yang ditampilkan pada Tabel 5.1 diatas. Pada bagian ini dijelaskan tiga kasus pengujian yaitu menambahkan UMKM, menambahkan produk dan menambahkan stok produk. Kasus hasil pengujian lainnya akan dijelaskan pada Lampiran J.

### 5.2.2.1 Pengujian Menambahkan UMKM

Pengujian menambahkan UMKM adalah pengujian dengan melakukan input data informasi UMKM, alamat UMKM serta pembayaran UMKM. Hasil pengujian menambahkan UMKM dapat dilihat pada Tabel 5.2

Tabel 5. 2 Pengujian Menambahkan UMKM (Benar)

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan dalam halaman tambah UMKM
Yang diharapkan	Sistem menampilkan halaman daftar UMKM dan notifikasi berhasil menambahkan UMKM baru
Pengamatan	Dapat menginputkan data pada halaman tambah UMKM dan mengklik tombol “Tambah”
Hasil	Sesuai

Pengujian menambahkan UMKM ini dilakukan oleh penjual yang mengakses halaman “UMKM” yang berada pada *sidebar* aplikasi. Pengujian dilakukan dengan mengklik tombol “Tambah UMKM” dan sistem menampilkan halaman *form* tambah UMKM. Pada form tambah UMKM, penjual menginputkan data informasi umum, alamat dan pembayaran dari UMKM. Setelah mengisi inputan, *user* akan mengklik tombol “Submit” dan sistem menampilkan notifikasi berhasil menambahkan UMKM dan menampilkan halaman daftar UMKM yang dimiliki. Tampilan hasil pengujian menambahkan UMKM (benar) dapat dilihat pada Gambar 5.10 dan Gambar 5.11 berikut

Gambar 5. 10 Pengujian Masukan Penambahan UMKM (Benar)

UMKM Sanjai

- Transaksi
- History
- Laporan
- Chat

Informasi Umum

Provinsi: Sumatera Barat

Kota: Payakumbuh

Alamat: Jl. Aceh, Tanjung Pauh 26222

Coordinate: -0.243091, 100.628405

UMKM Sanjai

- Dashboard
- UMKM**
- Transaksi
- History
- Laporan
- Chat

### Menambahkan Data UMKM

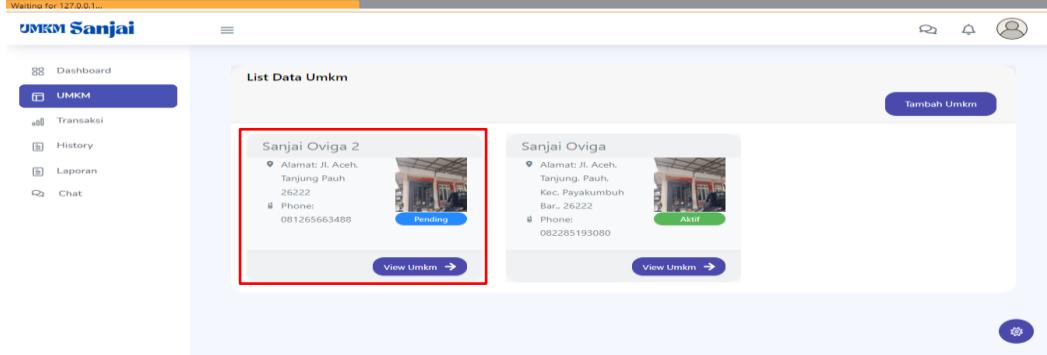
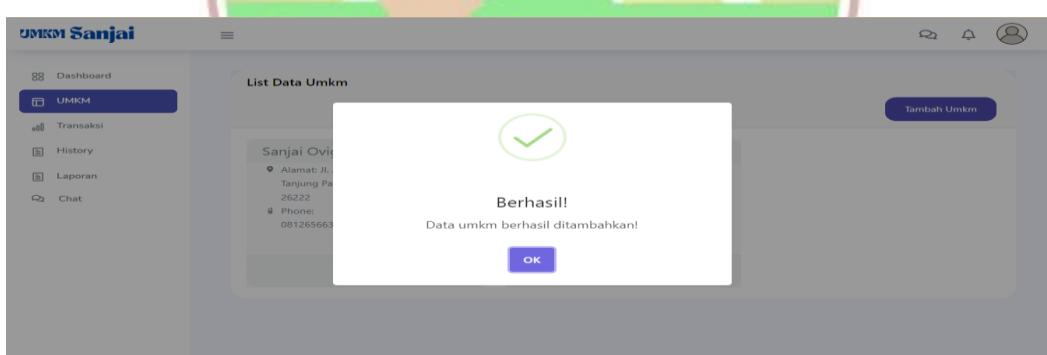
Pilih Metode Pembayaran

Bank BNI  Bank Mandiri  Bank BRI  Bank BCA  
 Bank Bukopin  Bank Permata  Bank BTN  Bank BJB  
 Bank CIMB  Bank Nagari

Bank Mandiri 08233343421 Bank Nagari 01223445332311

< Previous Submit

Gambar 5. 10 Pengujian Masukan Penambahan UMKM (Benar) Lanjutan



Gambar 5. 11 Pengujian Keluaran Penambahan UMKM (Benar)

Berdasarkan Gambar 5.10 dan Gambar 5.11, dapat dilihat bahwa inputan data berhasil dilakukan dan sistem menampilkan notifikasi data UMKM berhasil ditambahkan.

Untuk melakukan pembuktian bahwa input data UMKM sesuai dengan keinginan, maka pengujian dilanjutkan dengan mengakses langsung ke database dan melihat apakah data yang baru ditambahkan ada pada tabel umkm. Pembuktian pada database dapat dilihat pada Gambar 5.12 berikut.

	<input type="button" value="←"/>	<input type="button" value="→"/>	<a href="#">id</a>	<a href="#">province_id</a>	<a href="#">city_id</a>	<a href="#">penjual_id</a>	<a href="#">nama</a>	<a href="#">status</a>	<a href="#">imb</a>	<a href="#">alamat</a>	<a href="#">informasi</a>	<a href="#">img</a>	<a href="#">handphone</a>	<a href="#">latitude</a>	<a href="#">longitude</a>		
	<input type="checkbox"/>	<input type="button" value="Edit"/>	<input type="button" value="Copy"/>	<input type="button" value="Delete"/>	22	32	345	14	Sanjai Oviga 2	0	1692864462 slup.jpeg	Jl. Aceh, Tanjung Pahu 26222	Merupakan cabang dari Sanjai Oviga 1, menjual produk	1692864462 WhatsApp Image 2023-08-23 at 10.34.26 A...	081265663488	-0.243091	100.62840

Gambar 5. 12 Pengecekan Pengujian Penambahan UMKM (Benar)

Tabel 5. 3 Pengujian Menambahkan UMKM (Alternatif)

Kasus dan Hasil Uji (Alternatif)	
Data masukan	Penjual mengklik tombol “Submit” saat form tambah UMKM masih ada yang kosong
Yang diharapkan	Sistem menampilkan notifikasi kesalahan
Pengamatan	Penjual tidak bisa melakukan penambahan UMKM dan sistem menampilkan pesan kesalahan pada <i>form</i>
Hasil	Sesuai

Pengujian aplikasi ini dilakukan dengan mengklik tombol “Next” sebelum semua form diisi, kemudian sistem menampilkan pesan kesalahan. Hasil pengujian tambah UMKM (alternatif) dapat dilihat pada Gambar 5.13 berikut.

The screenshot shows the 'Menambahkan Data UMKM' (Add UMKM Data) form. It has two main sections: 'Informasi Umum' (Basic Information) and 'Informasi Tambahan' (Additional Information). The 'Informasi Umum' section contains fields for 'Nama Ukmn' (Name of Ukm) with value 'Sanjai Oviga', 'No Handphone' (Phone number) with an error message 'This value is required.', and 'Alamat' (Address). The 'Informasi Tambahan' section contains a text area for 'Input your umkm's information' and a file upload field for 'Upload Izin Usaha' (Business License) with an error message 'This value is required.' Below it is another file upload field for 'Upload Foto' (Photo) with an error message 'This value is required.' On the left sidebar, there are navigation links for 'UMKM', 'Transaksi', 'History', 'Laporan', and 'Chat'. The top right corner shows user profile icons.

Gambar 5. 13 Output Menambahkan UMKM (Alternatif)

### 5.2.2.2 Pengujian Menambahkan Produk

Pengujian menambahkan produk ini adalah pengujian dengan melakukan penginputan data berupa informasi umum produk, stok, harga, berat pengiriman serta gambar produk. Hasil pengujian menambahkan produk dapat dilihat pada Tabel 5.4 berikut.

Tabel 5. 4 Pengujian Menambahkan Produk (Benar)

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan dalam halaman tambah produk
Yang diharapkan	Sistem menampilkan halaman daftar produk dan notifikasi berhasil menambahkan produk baru
Pengamatan	Dapat menginputkan data pada halaman tambah produk dan mengklik tombol “Submit”
Hasil	Sesuai

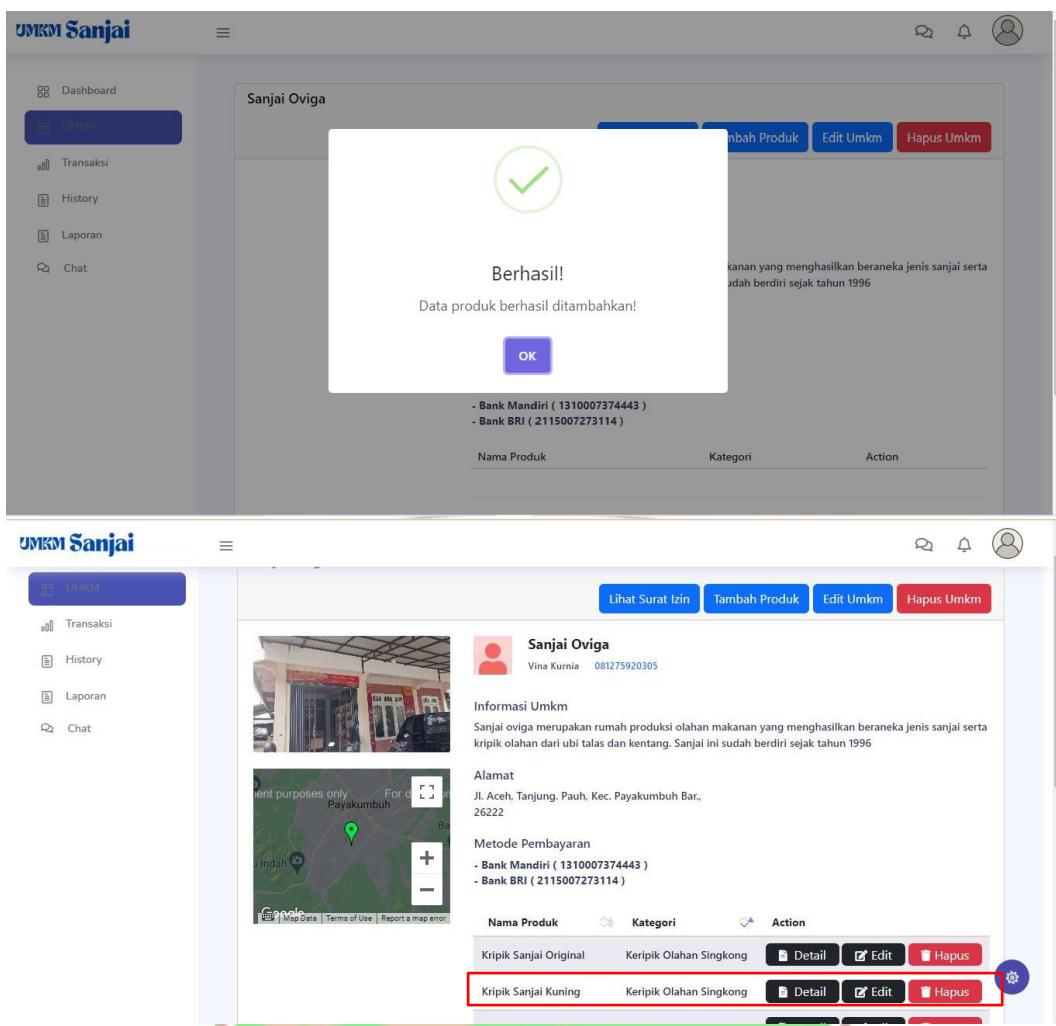
Pengujian menambahkan produk ini dilakukan oleh penjual yang mengakses halaman “UMKM” yang berada pada *sidebar* aplikasi, lalu mengklik “View UMKM” pada UMKM yang ingin ditambahkan produknya sehingga tampil halaman detail UMKM. Pengujian dilakukan dengan mengklik tombol “Tambah Produk” dan sistem menampilkan halaman *form* tambah produk. Pada form tambah produk, penjual menginputkan data informasi umum, data stok, kode produk, berat pengiriman dan gambar produk. Setelah mengisi inputan, *user* akan mengklik tombol “Submit” dan sistem menampilkan notifikasi berhasil menambahkan produk dan menampilkan halaman daftar produk pada UMKM di halaman detail UMKM. Tampilan hasil pengujian menambahkan produk (benar) dapat dilihat pada Gambar 5.14 dan Gambar 5.15 berikut.

**UMKM Sanjai**

The screenshots illustrate the steps of adding a product:

- Step 1: Product Information Form**  
The first screenshot shows the "Menambahkan Data Produk" (Adding Product Data) form. It has three tabs: "Informasi Umum Produk" (General Product Info), "Rincian Produk" (Product Details), and "Gambar Produk" (Product Images). The "Informasi Umum Produk" tab is active. It contains fields for "Nama Produk" (Product Name) with "Kripik Sanjai Kuning" entered, "Kategori Produk" (Product Category) with "Keripik Olahan Singkong" selected, "No Izin P-IRT" (P-IRT License Number) with "2.15.1376.09.0351.25" entered, and "Deskripsi" (Description) with "Terbuat dari singkong. Gurih dan renyah" (Made from cassava. Tasty and crispy) entered.
- Step 2: Product Detail Selection**  
The second screenshot shows the "Rincian Produk" tab. Under "Kemasan Produk" (Packaging), several options are listed with checkboxes: "kecil" (small), "sedang" (medium), "besar" (large), "100 gr", "200 gr", "250 gr" (selected), "300 gr", "500 gr", "1 kg", "3 kg", "5 kg", and "10 kg". Below this is a dropdown menu for "250 gr" showing values 1011, 20, 10000, and 1000, with 1000 selected. Navigation buttons "< Previous" and "Next >" are at the bottom.
- Step 3: Product Images**  
The third screenshot shows the "Gambar Produk" tab. It displays two images of the product: a bag of yellow crisps and some individual pieces. Each image has "Change", "Remove", and "Delete" buttons below it. Navigation buttons "< Previous" and "Next >" are at the bottom, along with a large green "Submit" button.

Gambar 5. 14 Pengujian Masukan Penambahan Produk (Benar)



Gambar 5. 15 Pengujian Keluaran Penambahan Produk (Benar)

Berdasarkan Gambar 5.14 dan Gambar 5.15, dapat dilihat bahwa input data berhasil dilakukan dan sistem menampilkan notifikasi data produk berhasil ditambahkan.

Untuk melakukan pembuktian bahwa input data produk sesuai dengan keinginan, maka pengujian dilanjutkan dengan mengakses langsung ke database dan melihat apakah data yang baru ditambahkan ada pada tabel produk. Pembuktian pada database dapat dilihat pada Gambar 5.16 berikut.

	<a href="#">Browse</a>	<a href="#">Structure</a>	<a href="#">SQL</a>	<a href="#">Search</a>	<a href="#">Insert</a>	<a href="#">Export</a>	<a href="#">Import</a>	<a href="#">Privileges</a>	<a href="#">Operations</a>	<a href="#">Tracking</a>	<a href="#">Triggers</a>
	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	49	14	15	Kue Dawang	2.06.1376.13.0351.25	Sangat gurih dan renyah	2023-08-23 21:25:26	2023-08-23 21:25:26
	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	50	14	4	Kripik Gula Aren	2.15.1376.14.0351.25	Olahan ubi talas yang diberi gula aren	2023-08-23 21:29:59	2023-08-23 21:29:59
	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	51	14	10	Kripik Pisang	2.14.1376.15.0351.25	Terbuat dari pisang manis sangat gurih dan renyah	2023-08-23 21:31:55	2023-08-23 21:31:55
	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	52	14	13	Kripik Pisang	3.16.1376.16.0351.25	Terbuat dari pisang manis sangat gurih dan renyah	2023-08-23 21:31:55	2023-08-23 21:31:55
	<a href="#">Edit</a>	<a href="#">Copy</a>	<a href="#">Delete</a>	54	14	3	Kripik Sanjai Kuning	2.15.1376.09.0351.25	Terbuat dari singkong. Gurih dan renyah	2023-08-24 15:57:11	2023-08-24 15:57:11

Gambar 5. 16 Pengecekan Pengujian Menambahkan Produk (Benar)

Tabel 5. 5 Pengujian Menambahkan Produk (Alternatif)

Kasus dan Hasil Uji (Alternatif)	
Data masukan	Penjual mengklik tombol “Submit” saat form tambah produk masih ada yang kosong
Yang diharapkan	Sistem menampilkan notifikasi kesalahan
Pengamatan	Penjual tidak bisa melakukan penambahan produk dan sistem menampilkan pesan kesalahan pada <i>form</i>
Hasil	Sesuai

Pengujian aplikasi ini dilakukan dengan mengklik tombol “Next” sebelum semua form diisi, kemudian sistem menampilkan pesan kesalahan. Hasil pengujian tambah produk (alternatif) dapat dilihat pada Gambar 5.17 berikut.

Gambar 5. 17 Output Menambahkan Produk (Alternatif)

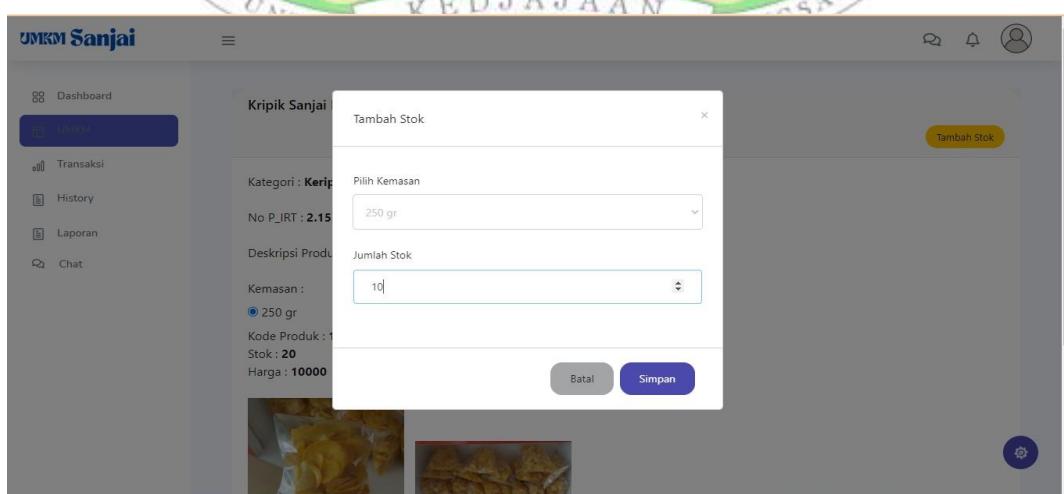
### 5.2.2.3 Pengujian Menambahkan Stok Produk

Pengujian menambahkan stok produk adalah pengujian dengan melakukan input jumlah stok pada produk yang ingin ditambahkan stoknya. Hasil pengujian menambahkan UMKM dapat dilihat pada Tabel 5.6

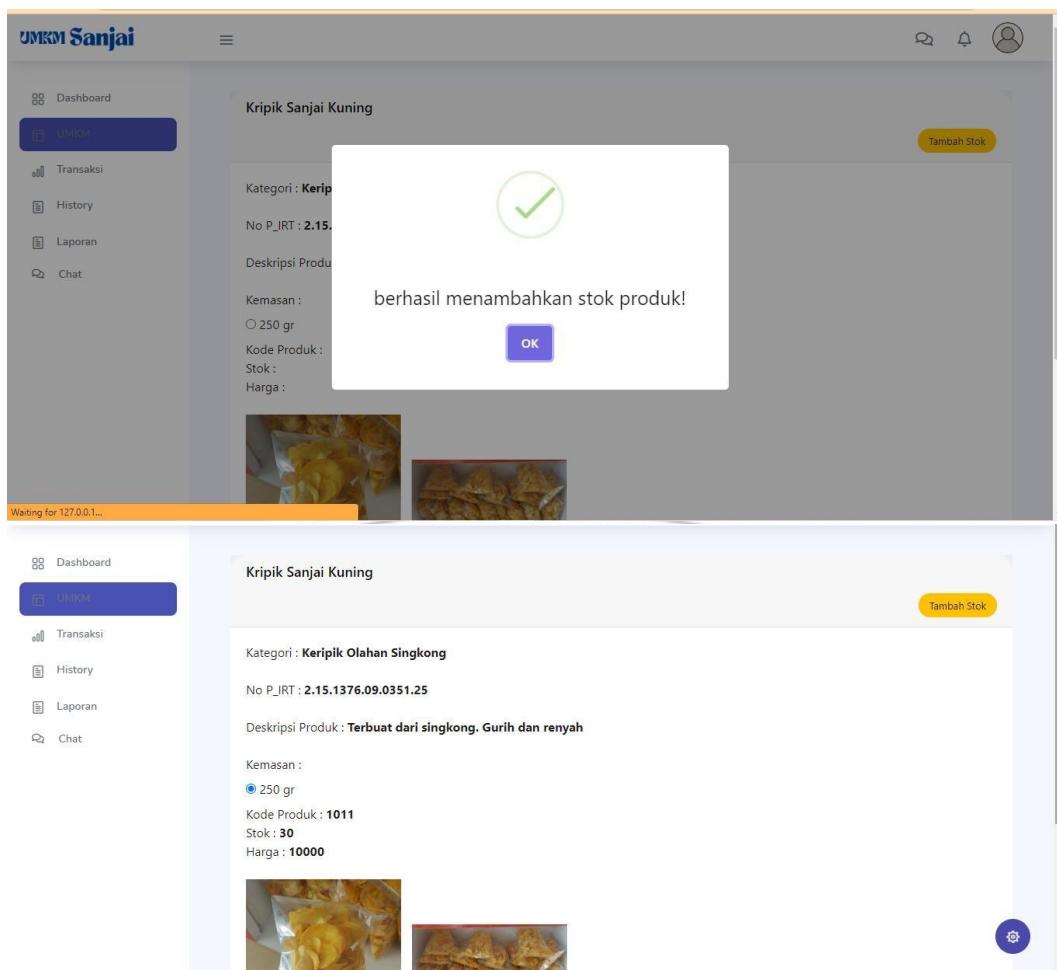
Tabel 5. 6 Pengujian Menambahkan Stok Produk (Benar)

Kasus dan Hasil Uji (Benar)	
Data masukan	Data id kemasan dan jumlah stok yang ingin ditambahkan
Yang diharapkan	Sistem menampilkan notifikasi berhasil menambahkan stok dan menampilkan halaman detail produk
Pengamatan	Dapat menginputkan data pada modal tambah stok dan megklik tombol “Tambah”
Hasil	Sesuai

Pengujian menambahkan stok produk dilakukan setelah penjual yang mengakses halaman “UMKM” yang berada pada sidebar aplikasi, lalu mengklik “View UMKM” pada UMKM yang ingin ditambahkan produknya sehingga tampil halaman detail UMKM. Pengujian dilakukan dengan mengklik tombol “Detail” pada produk yang ingin ditambahkan stoknya, dan sistem menampilkan halaman detail produk produk, lalu penjual mengklik tombol “tambah stok” dan muncul modal untuk melakukan penginputan variasi kemasan yang ingin ditambah stok nya serta inputan jumlah stok produk. Setelah mengisi inputan, user akan mengklik tombol “Simpan” dan sistem menampilkan notifikasi berhasil menambahkan stok produk dan menampilkan halaman detail produk. Tampilan hasil pengujian menambahkan stok produk (benar) dapat dilihat pada Gambar 5.18 dan Gambar 5.19 berikut



Gambar 5. 18 Pengujian Masukan Menambahkan Stok Produk (Benar)



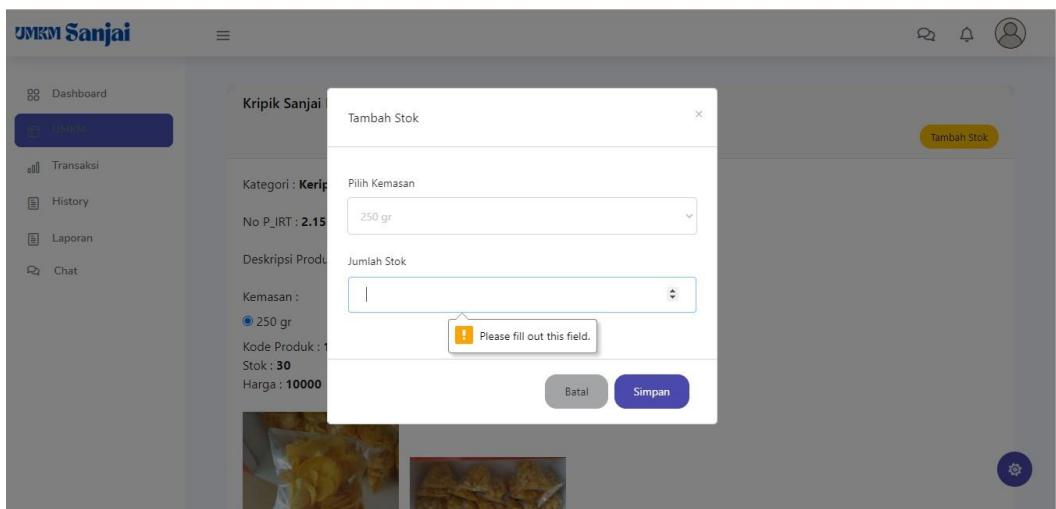
Gambar 5. 19 Pengujian Keluaran Menambahkan Stok Produk

Berdasarkan Gambar 5.18 dan Gambar 5.19, dapat dilihat bahwa inputan menambahkan stok berhasil dilakukan dan sistem menampilkan notifikasi data stok produk berhasil ditambahkan, dapat dilihat stok yang awalnya berjumlah 20 setelah ditambahkan 10 menjadi 30.

Tabel 5. 7 Pengujian Menambahkan Stok Produk (Alternatif)

Kasus dan Hasil Uji (Alternatif)	
Data masukan	Penjual mengklik tombol “Simpan” saat modal tambah stok produk masih ada yang kosong
Yang diharapkan	Sistem menampilkan pesan kesalahan
Pengamatan	Penjual tidak bisa melakukan penambahan stok produk dan sistem menampilkan pesan kesalahan
Hasil	Sesuai

Pengujian aplikasi ini dilakukan dengan mengklik tombol “Simpan” saat modal menambahkan stok produk belum terisi, kemudian sistem menampilkan pesan kesalahan. Hasil pengujian menambahkan stok produk (alternatif) dapat dilihat pada Gambar 5.20 berikut

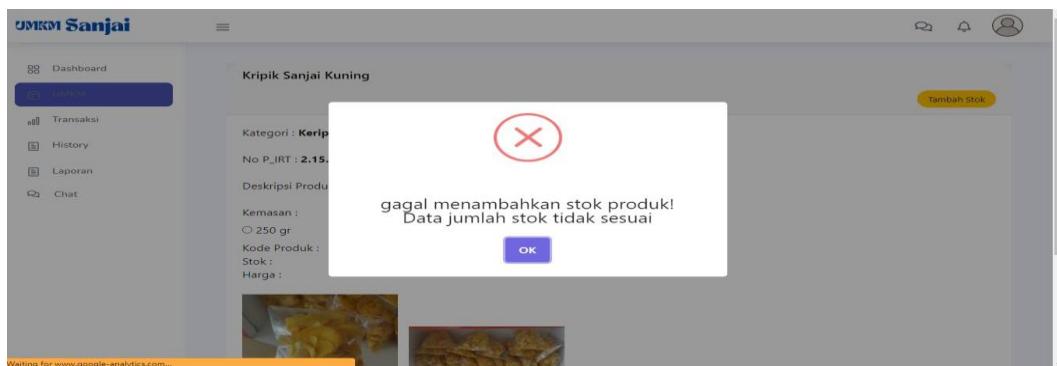


Gambar 5. 20 Output Menambahkan Stok Produk (Alternatif)

Tabel 5. 8 Pengujian Menambahkan Stok Produk (Alternatif)

Kasus dan Hasil Uji (Alternatif)	
Data masukan	Penjual mengklik tombol “Simpan” saat inputan jumlah stok minus
Yang diharapkan	Sistem menampilkan notifikasi kesalahan
Pengamatan	Penjual tidak bisa melakukan penambahan stok produk dan sistem menampilkan notifikasi kesalahan
Hasil	Sesuai

Pengujian aplikasi ini dilakukan dengan mengklik tombol “Simpan” saat inputan jumlah stok tidak valid seperti *minus*, kemudian sistem menampilkan notifikasi gagal menambahkan stok produk. Hasil pengujian menambahkan stok produk (alternatif) dapat dilihat pada Gambar 5.21 berikut.



Gambar 5. 21 *Output* Menambahkan Stok Produk (Alternatif)

### 5.3 Analisis Hasil Pengujian

Berdasarkan pengujian yang telah dilakukan, didapatkan kecocokan hasil antara perancangan sistem dengan keluaran sistem. Setelah dilakukan pengujian ini, tidak terdapat kegagalan dalam setiap proses fungsionalnya sehingga dapat diambil kesimpulan bahwa *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh telah berjalan sesuai fungsional yang dirancang. Analisis dari hasil pengujian dapat dilihat pada Tabel 5.9 berikut.

Tabel 5. 9 Analisis Hasil Pengujian

No	Item Uji	User	Detail Pengujian	Hasil
1.	<i>Login</i>	Admin, Penjual, Pembeli, Kurir	Verifikasi data <i>login</i>	Sesuai
2.	<i>Register</i>	Penjual, Pembeli	Tambah	Sesuai
3.	Kelola data <i>user</i>	Admin	Lihat, tambah, edit, hapus	Sesuai
4.	Ubah <i>password</i>	Admin, Penjual, Pembeli	Edit	Sesuai
5.	Kelola data profil	Admin, Penjual, Pembeli	Lihat, edit	Sesuai
6.	Kelola data kategori produk	Admin	Lihat, tambah, edit, hapus	Sesuai
7.	Kelola data UMKM	Penjual	Lihat, tambah, edit, hapus	Sesuai

Tabel 5.9 Analisis Hasil Pengujian (Lanjutan)

8.	Kelola perizinan UMKM	Admin	Lihat, terima, tolak, <i>banned</i>	Sesuai
9.	Kelola data produk	Penjual	Lihat, tambah, edit, hapus	Sesuai
10.	Menambah stok produk	Penjual	Tambah	Sesuai
11.	Melakukan <i>chat</i>	Penjual, Pembeli	Tambah	Sesuai
12.	Melakukan penelusuran umkm dan produk	Pembeli	Lihat	Sesuai
13.	Melakukan pembelian produk	Pembeli	Lihat, tambah	Sesuai
14.	Kelola data transaksi pembelian	Penjual	Lihat, terima, tolak	Sesuai
15.	Melihat laporan transaksi	Penjual	Lihat, cetak	Sesuai
16.	Melihat riwayat pembelian	Penjual, Pembeli	Lihat	Sesuai
17.	Menerima notifikasi pembelian produk	Penjual	Lihat	Sesuai
18.	Menerima notifikasi UMKM perlu konfirmasi	Admin	Lihat	Sesuai
19.	Menginputkan bukti pengiriman	Kurir	Tambah	Sesuai
20.	Melihat riwayat pengiriman	Kurir	Lihat	Sesuai
21.	<i>Logout</i>	Admin, Penjual, Pembeli	<i>Logout</i>	Sesuai

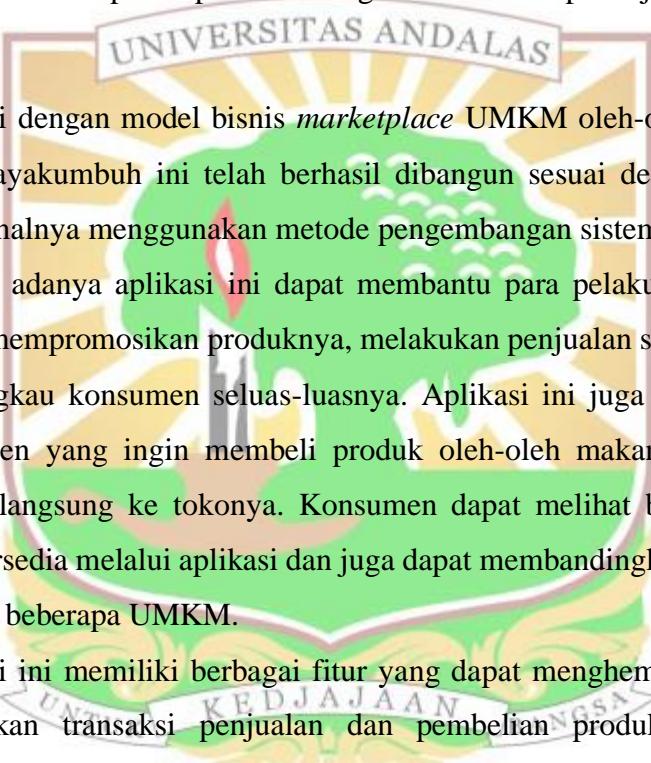
## BAB VI

### PENUTUP

Pada bab ini dijelaskan mengenai kesimpulan dan saran dari laporan tugas akhir ini. Pada kesimpulan terdapat hasil akhir dari penelitian ini berdasarkan tujuan yang telah ditetapkan sebelumnya. Saran berisi harapan yang diinginkan kedepannya terkait dengan penelitian ini.

#### 6.1 Kesimpulan

Kesimpulan dari laporan penelitian tugas akhir ini dapat dijabarkan sebagai berikut :

- 
1. Aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh ini telah berhasil dibangun sesuai dengan kebutuhan fungsionalnya menggunakan metode pengembangan sistem *waterfall*.
  2. Dengan adanya aplikasi ini dapat membantu para pelaku usaha UMKM dalam mempromosikan produknya, melakukan penjualan secara online dan menjangkau konsumen seluas-luasnya. Aplikasi ini juga membantu para konsumen yang ingin membeli produk oleh-oleh makanan tanpa harus datang langsung ke tokonya. Konsumen dapat melihat berbagai produk yang tersedia melalui aplikasi dan juga dapat membandingkan produk yang sama di beberapa UMKM.
  3. Aplikasi ini memiliki berbagai fitur yang dapat menghemat waktu dalam melakukan transaksi penjualan dan pembelian produk. Aplikasi ini dirancang untuk meminimalisir proses bisnis penjualan online yang sebelumnya kompleks dan tidak tercatat menjadi proses bisnis yang lebih efektif dan efisien.

#### 6.2 Saran

Aplikasi dengan model bisnis *marketplace* UMKM oleh-oleh makanan di Kota Payakumbuh ini masih membutuhkan pengembangan lebih lanjut yang sejalan dengan fungsional dan perkembangan aplikasi dengan model bisnis *marketplace* UMKM kedepannya. Diharapkan aplikasi ini dapat dikembangkan lebih luas dengan menambahkan fitur yang lebih lengkap daripada fitur yang sudah ada.

Selain itu diharapkan juga adanya pengembangan aplikasi bahwa aplikasi ini tidak hanya diperuntukkan untuk UMKM oleh-oleh makanan saja tetapi untuk keseluruhan UMKM secara umum pada daerah tertentu. Aplikasi ini juga diharapkan adanya pengembangan berbasis *mobile* sehingga pengguna lebih mudah mendapatkan update terbaru tentang produk tanpa harus membuka melalui *link website* terlebih dahulu.



## DAFTAR PUSTAKA

- Ahsyar, T. K., Syaifulah, S., & Ardiansyah, A. (2020). E-Marketplace Media Pengembangan Promosi Usaha Mikro Kecil Dan Menengah Dinas Koperasi Umkm Kota Pekanbaru. *Jurnal Ilmiah Rekayasa Dan Manajemen Sistem Informasi*, 6(1), 43. <https://doi.org/10.24014/rmsi.v6i1.8768>
- Artaya, I. P., & Purworusmiardi, T. (2019). Efektifitas Marketplace Dalam Meningkatkan Konsentrasi. *Ekonomi Dan Bisnis, Universitas Narotama Surabaya, April*, 1–10.
- Aulami, R., & Ariani, F. (2022). Aplikasi E-Marketplace Pada Usaha Mikro Kecil Menengah (UMKM) Berbasis Mobile. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, 3(1), 66–72. <http://jim.teknokrat.ac.id/index.php/informatika/article/view/1861>
- Bambang, S., & Abdur, R. (2019). Implementasi Pemasaran Digital Dalam Meningkatkan Pendapatan Pelaku UMKM Pamokolan Ciamis. *Jurnal PRODUKTIF*, 3(2), 254–261. <http://www.journal.umtas.ac.id/index.php/produktif/article/view/669>
- “Clone Worthy Script Features to Create P2P Handmade and Unique Goods Marketplace,” fatbit.com, [Online]. Available: <https://www.fatbit.com/fab/clone-worthy-script-features-create-unique-p2p-handmade-goods-marketplace/>. [Diakses 14 12 2022].
- Cv, K., & Berbasis, X. Y. Z. Purwanto (2014). Perancangan Sistem Informasi Administrasi Kepegawaian Cv. Xyz Berbasis Web. *Jurnal Sistem Informasi Universitas Suryadarma*, 5(2). <https://doi.org/10.35968/jsi.v5i2.241>
- D. Edi and S. Betshani, “Analisis Data Dengan Menggunakan ERD Dan Model Konseptual Data Warehouse,” *Informatika*, vol. 5, pp. 71–85, 2009
- Dewi, L. P., Indahyanti, U., & S, Y. H. (2017). Pemodelan Proses Bisnis Menggunakan Activity Diagram Uml Dan Bpmn ( Studi Kasus Frs Online ). *Informatika*, 1–9.

- Dr. H. Muhammad Yusuf Saleh, S. E. M. S., Dr. Miah Said, S. E. M. S., Sobirin, S. S. M. S., & Media, S. A. H. (2019). *Konsep dan Strategi Pemasaran: Marketing Concepts and Strategies*. SAH MEDIA. <https://books.google.co.id/books?id=pZu-DwAAQBAJ>
- Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2021). Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website. *EXPERT: Jurnal Manajemen Sistem Informasi Dan Teknologi*, 11(1), 48. <https://doi.org/10.36448/expert.v11i1.2012>
- Fauzan, R., & Jayanti, A. (2019). Strategi Pemasaran Untuk Meningkatkan Volume Pejualan Dengan Menggunakan Blue Ocean Strategy Model Pada Usaha Sanjai Nitta Bukittinggi. *Jurnal Studi Manajemen Dan Bisnis*, 6(2), 49–54.
- Fauzan, R., Lotte, L. N. A., Saeni, N., Sudirjo, F., Setiawan, Z., Pandowo, A., Titing, A. S., Murdana, I. M., Nasution, A. A., & others. (2023). *Manajemen Pemasaran Global Eksekutif Teknologi*. <https://books.google.co.id/books?id=28K7EAAAQBAJ>
- Hamza, L. M., & Agustien, D. (2019). Pengaruh Perkembangan Usaha Mikro, Kecil, dan Menengah Terhadap Pendapatan Nasional Pada Sektor UMKM di Indonesia. *Jurnal Ekonomi Pembangunan*, 8(2), 127–135. <https://doi.org/10.23960/jep.v8i2.45>
- Ismanto, Firman Hidayah, K. (2020). Pemodelan Proses Bisnis Menggunakan Business Process Modelling Notation (BPMN). *Jurnal Riset Dan Konseptual*, 5, 69–76.
- Lengkong, M., Pardanus, R. H. W., & Parinsi, M. T. (2022). Sistem Informasi Kelurahan Tataaran 1 Kecamatan Tondano Selatan. *Edutik : Jurnal Pendidikan Teknologi Informasi Dan Komunikasi*, 2(1), 36–43. <https://doi.org/10.53682/edutik.v2i1.3367>
- Luthfi, F. (2017). Penggunaan Framework Laravel dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID. *JISKA (Jurnal Informatika Sunan Kalijaga)*, 2(1), 34–41. <https://doi.org/10.14421/jiska.2017.21-05>

- Naovarat, S., & Juntongjin, P. (2015). *Factor that affecting success of E-Marketplace in Thailand*. 15–18. <https://doi.org/10.15242/iae.iae0415016>
- Natania, A. T., & Dwijayanti, R. (2023). *PEMANFAATAN PLATFORM DIGITAL*. 11(3).
- Pamungkas, C. A. (2017). *Pengantar dan Implementasi Basis Data*. Deepublish. <https://books.google.co.id/books?id=hKdADwAAQBAJ>
- Sadgotra, W. Y., & Saputra, E. H. (2013). Perancangan Online Marketplace Untuk Usaha Kecil Dan Menengah (Ukm) Di Kabupaten Purworejo. *Jurnal Ilmiah DASI*, 14(04), 54–58.
- Saputro, Haris.(2012). “Modul Pembelajaran Basis Data (MySQL)”, Halaman 6-7
- Seidl, M., Scholz, M., Huemer, C., & Kappel, G. (2015). The Sequence Diagram. In *UML @ Classroom: An Introduction to Object-Oriented Modeling* (pp. 107–140). Springer International Publishing. [https://doi.org/10.1007/978-3-319-12742-2\\_6](https://doi.org/10.1007/978-3-319-12742-2_6)
- Setiawati, I. (2017). Pengaruh Strategi Pemasaran Online Terhadap Peningkatan Laba Umkm. *Strategi Komunikasi Pemasaran*, 20, 1–5.
- Setiyani, L. (2021). Desain Sistem : Use Case Diagram Pendahuluan. *Prosiding Seminar Nasional : Inovasi & Adopsi Teknologi 2021*, 1(1), 246–260.
- Setiyawati, N. (2020). Marketplace Larisumkm.Com Sebagai Alat Peningkatan Promosi Ukm Di Kota Salatiga. *SELAPARANG Jurnal Pengabdian Masyarakat Berkemajuan*, 4(1), 696. <https://doi.org/10.31764/jpmb.v4i1.3376>
- Shah, C. (2020). MySQL. *A Hands-On Introduction to Data Science*, 187–206. <https://doi.org/10.1017/9781108560412.008>
- Sudarsono, H. (n.d.). *Manajemen Pemasaran*. Pustaka Abadi. <https://books.google.co.id/books?id=Z-fWDwAAQBAJ>
- Undang-Undang Republik Indonesia Nomor 20 Tahun. (2008). *Undang-Undang Republik Indonesia Nomor 20 Tahun 2008*.

Verma, A., Khatana, A., & Chaudhary, S. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304.  
<https://doi.org/10.26438/ijcse/v5i12.301304>

Wahid, A. A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, November, 1–5.

Wibowo, K. (2015). Analisa Konsep Object Oriented Programming Pada Bahasa Pemrograman Php. *Jurnal Khatulistiwa Informatika*, 3(9), 151–159.

Yanto, R. (2016). *Manajemen Basis Data Menggunakan MySQL*. Deepublish.  
<https://books.google.co.id/books?id=VMReDwAAQBAJ>

Yustiani, R., & Yunanto, R. (2017). Peran Marketplace Sebagai Alternatif Bisnis Di Era Teknologi Informasi. *Komputa: Jurnal Ilmiah Komputer Dan Informatika*, 6(2), 43–48. <https://doi.org/10.34010/komputa.v6i2.2476>





WAWANCARA UMKM

Profil usaha	<ul style="list-style-type: none"> <li>- Sanjai mai</li> <li>- Mbuk Mai.</li> <li>- Segar tahun 1998</li> <li>- Jln Imam Bonjol, Kelurahan bulakor, Kecamatan Kandis.</li> </ul>
Produk yang dijual beserta harga per bks/per kg	<ul style="list-style-type: none"> <li>- Sanjai</li> <li>- Karor kelingking</li> <li>- kerupuk halado</li> <li>- Jagung kelingking</li> <li>- gulamai</li> <li>- beras randuncing.</li> </ul>
Pemasaran nya dilakukan bagaimana	<p>Sama setali tidak menggunakan platform teknologi apapun.</p> <p>Dilakukan secara manual melalui spanduk di tetek saja.</p>
Pengiriman ke luar daerah bagaimana sistem nya	<ul style="list-style-type: none"> <li>- Pemesanan langsung lewat telepon</li> <li>- Merepotkan koran melalui burs, untuk konfirmasi harus telpon-telpon kurir dulu.</li> </ul>
Pembukuan pencatatan penjualannya bagaimana (kalau boleh difoto)	<ul style="list-style-type: none"> <li>- tidak ada pencatatan / pembukuan sama sekali</li> </ul>
Foto kwitansi penjualan	X

Apakah ada kasus sanjai hasil produksi tidak habis	Jika ada berarti dibayar.
Pengadaan stok	- Produk siap jual langsung.

Pewawancara

(Distifia Oktari)

Pihak Pelaku UMKM

( Sisda )

WAWANCARA UMKM

Profil usaha	<ul style="list-style-type: none"> <li>- Nama usaha → <b>Gudis</b></li> <li>- Pemilik <b>Ibu Kadi</b></li> <li>- Sudah berapa lama <b>L 2000.</b></li> <li>- Alamat</li> </ul> <ul style="list-style-type: none"> <li>- Sangan 3 saudara</li> <li>- <del>Gudu</del> ibuk Kadi</li> <li>- sebelum tahun 2000</li> <li>- Jln. Imam Bonjol No 26, kelurahan bulakan halai kandi</li> </ul>
Produk yang dijual beserta harga per bks/per kg	<ul style="list-style-type: none"> <li>- kripik ubi biasa</li> <li>- kerak kailang</li> <li>- kripik batu</li> <li>- kripik bayam</li> <li>- jagung keriting</li> <li>- </li> </ul> <p style="text-align: right;">} 10k / 1/4</p>
Pemasaran nya dilakukan bagaimana	<ul style="list-style-type: none"> <li>- Sudah ada dilakukan pemasaran online, tetapi pemilik kurang mengetahui platform apa yang digunakan</li> <li>- kebanyakan pengujal masih dilakukan secara langsung.</li> </ul>
Pengiriman ke luar daerah bagaimana sistem nya	<ul style="list-style-type: none"> <li>- pengiriman dilakukan via call dan dikirim melalui bus.</li> <li>- tidak ada aplikasi untuk melihat status pengiriman</li> </ul>
Pembukuan pencatatan penjualannya bagaimana (kalau boleh difoto)	<ul style="list-style-type: none"> <li>- pencatatan tidak ada, hanya dalam bentuk nota</li> <li>- tidak ada data yang disimpan secara terkomputerisasi</li> </ul>
Foto kwitansi penjualan	✓

Apakah ada kasus sanjai hasil produksi tidak habis	ada , produk juga dibakar sebelum manuk itu kadaluwarsa .
Pengadaan stok	Produksi sendiri dan telpan dari orang lain .

Pewawancara

(Distifia Oktari)

Pihak Pelaku UMKM

)

WAWANCARA UMKM

Profil usaha	<ul style="list-style-type: none"> <li>- Nama usaha</li> <li>- Pemilik</li> <li>- Sudah berapa lama</li> <li>- Alamat</li> </ul> <p><del>Putri</del> - Putri Chonidjojo Sariani - 25 tahun - Jln. Imam Bonjol</p>
Produk yang dijual beserta harga per bks/per kg	<ul style="list-style-type: none"> <li>- kripik ubi halado (tulasi)</li> <li>- kripik ubi kuning</li> <li>- pisang salai</li> <li>- kripik kacang</li> <li>- Harga grosir 30k / kg.</li> <li>- per kgs 10k .</li> </ul>
Pemasaran nya dilakukan bagaimana	<ul style="list-style-type: none"> <li>- Pernah melakukan pemasaran lewat online (shopee) tetapi sudah tidak jalan lagi.</li> <li>- Pemasaran melalui media sosial Instagram.</li> </ul>
Pengiriman ke luar daerah bagaimana sistem nya	<ul style="list-style-type: none"> <li>- Pengiriman manual melalui telepon dan dikirim melalui bus</li> <li>- tidak ada aplikasi untuk melihat status pengiriman</li> </ul>
Pembukuan pencatatan penjualannya bagaimana (kalau boleh difoto)	<ul style="list-style-type: none"> <li>- Pencatatan manual dengan kartas. → tetapi tidak teratur dan ada sekurang sudah tidak</li> </ul>
Foto kwitansi penjualan	✓

Apakah ada kasus sanjai hasil produksi tidak habis	ada, kejatuhan dibakar.
Pengadaan stok	produk jendiri, ada dapur / gedung penyaluran produk.

Pewawancara

(Distifia Oktari)

Pihak Pelaku UMKM

( Ratna )

WAWANCARA UMKM

Profil usaha	<ul style="list-style-type: none"> <li>- Jangai Oviga</li> <li>- Vina Kurnia</li> <li>- sejak tahun 1996</li> <li>- jalan Aceh, Tanjung Reuh</li> </ul>
Produk yang dijual beserta harga per bks/per kg	<ul style="list-style-type: none"> <li>- kripik ubi ori</li> <li>- kripik ubi pedas</li> <li>- kripik tempe ori</li> <li>- kripik tempe pedas</li> <li>- ds</li> </ul>
Pemasaran nya dilakukan bagaimana	<ul style="list-style-type: none"> <li>- Melalui organisasi tulapi , mulut ke mulut .</li> <li>- Melalui wa .</li> </ul>
Pengiriman ke luar daerah bagaimana sistem nya	<ul style="list-style-type: none"> <li>- Menggunakan Indah Cargo</li> <li>- Pengiriman melalui Paket .</li> </ul>
Pembukuan pencatatan penjualannya bagaimana (kalau boleh difoto)	<p><del>pencatatan</del> Pencatatan manual buku (dulu) sekarang sudah jarang .</p>
Foto kwitansi penjualan	-

Apakah ada kasus sanjai hasil produksi tidak habis	Jangan terjadi
Pengadaan stok	Produksi sendiri jika ada stok yang habis.

Pewawancara

(Distifia Oktari)

Pihak Pelaku UMKM

(Vina Kurnia )



## 1. Use Case Scenario Login

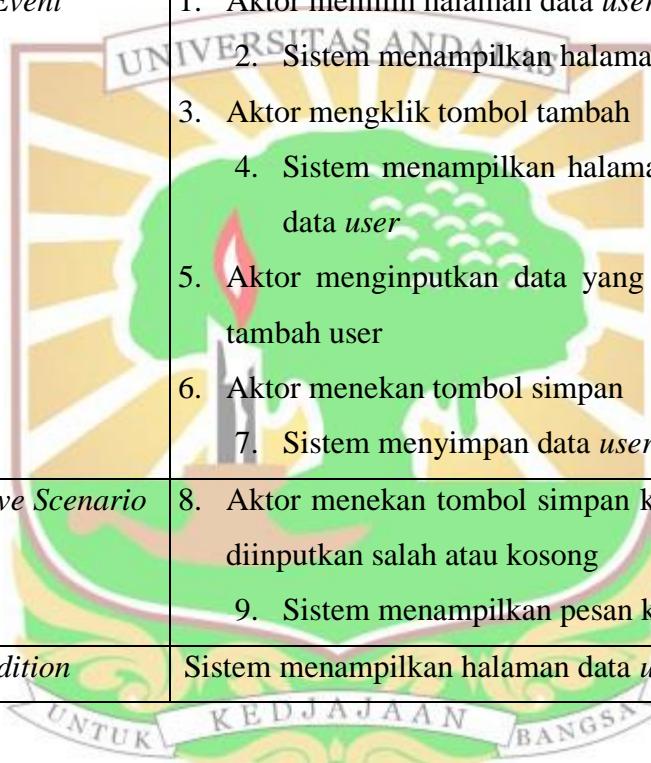
<i>Use case</i>	<i>Login</i>
<i>Actor</i>	Admin, Penjual, Pembeli, Kurir
<i>Entry Condition</i>	Aktor telah memiliki akun
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor membuka aplikasi UMKM</li> <li>2. Sistem menampilkan halaman <i>landing page</i></li> <li>3. Aktor mengklik tombol <i>login</i></li> <li>4. Sistem menampilkan halaman login</li> <li>5. Aktor menginputkan <i>email</i> dan <i>password</i> yang sesuai</li> <li>6. Aktor menekan tombol <i>login</i></li> <li>7. Sistem menampilkan halaman <i>dashboard</i></li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol <i>login</i> ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan gagal <i>login</i></li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman utama

## 2. Use Case Scenario Registrasi

<i>Use case</i>	Registrasi
<i>Actor</i>	Penjual, Pembeli
<i>Entry Condition</i>	Aktor belum memiliki akun
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor membuka aplikasi UMKM</li> <li>2. Sistem menampilkan halaman <i>landing page</i></li> <li>3. Aktor mengklik tombol registrasi</li> <li>4. Sistem menampilkan halaman form registrasi</li> <li>5. Aktor menginputkan data yang ada pada form</li> <li>6. Aktor menekan tombol register</li> <li>7. Sistem menampilkan halaman <i>dashboard</i></li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol register ketika data yang diinputkan kosong</li> </ol>

	9. Sistem menampilkan pesan “ <i>please fill this field</i> ”
<i>Exit Condition</i>	Sistem menampilkan halaman login

### 3. Use Case Scenario Menambah Data User



<i>Use case</i>	Menambah data <i>user</i>
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman data <i>user</i></li> <li>2. Sistem menampilkan halaman data <i>user</i></li> <li>3. Aktor mengklik tombol tambah</li> <li>4. Sistem menampilkan halaman <i>form</i> tambah data <i>user</i></li> <li>5. Aktor menginputkan data yang ada pada <i>form</i> tambah user</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data <i>user</i> baru</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data <i>user</i>

### 4. Use Case Scenario Mengedit Data User

<i>Use case</i>	Mengedit data <i>user</i>
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman data <i>user</i></li> <li>2. Sistem menampilkan halaman data <i>user</i></li> <li>3. Aktor mengklik tombol edit pada <i>field user</i> yang akan diedit</li> </ol>

	4. Sistem menampilkan halaman <i>form</i> edit data <i>user</i> 5. Aktor mengedit data yang ingin diupdate 6. Aktor menekan tombol simpan 7. Sistem menyimpan data <i>user</i> yang telah diedit
<i>Alternative Scenario</i>	8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong 9. Sistem menampilkan pesan kesalahan
<i>Exit Condition</i>	Sistem menampilkan halaman data <i>user</i>

### 5. Use Case Scenario Menghapus Data User

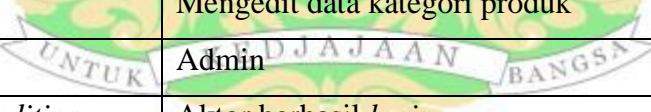
<i>Use case</i>	Menghapus data <i>user</i>
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	1. Aktor memilih halaman data <i>user</i> 2. Sistem menampilkan halaman data <i>user</i> 3. Aktor mengklik tombol hapus pada <i>field user</i> yang akan dihapus 4. Sistem menampilkan pesan konfirmasi untuk menghapus data 5. Aktor menekan tombol “Ya” 6. Sistem menghapus data <i>user</i>
<i>Alternative Scenario</i>	7. Aktor menekan tombol “Batal” 8. Sistem batal menghapus data <i>user</i>
<i>Exit Condition</i>	Sistem menampilkan halaman data <i>user</i>

### 6. Use Case Scenario Menambah Data Kategori Produk



<i>Use case</i>	Menambah data kategori produk
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman kategori produk</li> <li>2. Sistem menampilkan halaman kategori produk</li> <li>3. Aktor mengklik tombol tambah</li> <li>4. Sistem menampilkan halaman <i>form</i> tambah data kategori produk</li> <li>5. Aktor menginputkan data yang ada pada <i>form</i> tambah kategori produk</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data kategori produk baru</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data kategori produk

## 7. Use Case Scenario Mengedit Data Kategori Produk



<i>Use case</i>	Mengedit data kategori produk
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil login
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman data kategori produk</li> <li>2. Sistem menampilkan halaman kategori produk</li> <li>3. Aktor mengklik tombol edit pada <i>field</i> kategori produk yang akan diedit</li> <li>4. Sistem menampilkan halaman <i>form</i> edit data kategori produk</li> <li>5. Aktor mengedit data yang ingin diupdate</li> </ol>

	<p>6. Aktor menekan tombol simpan</p> <p>7. Sistem menyimpan data kategori produk yang telah diedit</p>
<i>Alternative Scenario</i>	<p>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</p> <p>9. Sistem menampilkan pesan kesalahan</p>
<i>Exit Condition</i>	Sistem menampilkan halaman data kategori produk

## 8. Use Case Scenario Menghapus Data Kategori Produk

<i>Use case</i>	Menghapus data kategori produk
<i>Actor</i>	Admin
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<p>1. Aktor memilih halaman kategori produk</p> <p>2. Sistem menampilkan halaman kategori produk</p> <p>3. Aktor mengklik tombol hapus pada <i>field</i> kategori produk yang akan dihapus</p> <p>4. Sistem menampilkan pesan konfirmasi untuk menghapus data</p> <p>5. Aktor menekan tombol “Ya”</p> <p>6. Sistem menghapus data kategori produk</p>
<i>Alternative Scenario</i>	<p>7. Aktor menekan tombol “Batal”</p> <p>8. Sistem batal menghapus data kategori produk</p>
<i>Exit Condition</i>	Sistem menampilkan halaman kategori produk

## 9. Use Case Scenario Mengubah Password

<i>Use case</i>	Mengubah password
<i>Actor</i>	Admin, Penjual, Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<p>1. Aktor memilih menu profil</p>

	<ol style="list-style-type: none"> <li>2. Sistem menampilkan profil yang mana terdapat section untuk mengubah <i>password</i></li> <li>3. Aktor menginputkan <i>password</i> lama dan <i>password</i> baru</li> <li>4. Aktor menekan tombol simpan</li> <li>5. Sistem menyimpan data perubahan <i>password</i></li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>6. Aktor menekan tombol simpan ketika data <i>password</i> yang diinputkan salah</li> <li>7. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Aktor berhasil mengubah <i>password</i>

#### 10. Use Case Scenario Mengubah Data Profil

<i>Use case</i>	Mengubah data profil
<i>Actor</i>	Admin, Penjual, Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman data profil</li> <li>2. Sistem menampilkan halaman profil</li> <li>3. Sistem menampilkan halaman form edit profil</li> <li>4. Aktor mengisi data profil yang ingin diubah</li> <li>5. Aktor menekan tombol simpan</li> <li>6. Sistem menyimpan data perubahan profil</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>7. Aktor menekan tombol simpan ketika data profil yang diinputkan salah</li> <li>8. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman profil

#### 11. Use Case Scenario Menambah Data Produk

<i>Use case</i>	Menambah data produk
<i>Actor</i>	Penjual

<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman <i>view UMKM</i></li> <li>2. Sistem menampilkan halaman detail UMKM</li> <li>3. Aktor mengklik tombol tambah produk</li> <li>4. Sistem menampilkan halaman form tambah data produk</li> <li>5. Aktor menginputkan data yang ada pada <i>form</i> tambah produk</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data produk baru</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data produk

## 12. Use Case Scenario Mengedit Data Produk

<i>Use case</i>	Mengedit data produk
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman <i>view UMKM</i></li> <li>2. Sistem menampilkan halaman detail UMKM</li> <li>3. Aktor mengklik tombol edit pada <i>field</i> produk yang akan diedit</li> <li>4. Sistem menampilkan halaman <i>form</i> edit data produk</li> <li>5. Aktor mengedit data yang ingin diupdate</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data produk yang telah diedit</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> </ol>

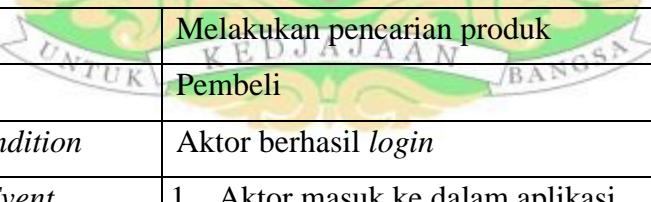
	9. Sistem menampilkan pesan kesalahan
<i>Exit Condition</i>	Sistem menampilkan halaman detail UMKM

### 13. Use Case Scenario Menghapus Data Produk



<i>Use case</i>	Menghapus data produk
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman <i>view UMKM</i></li> <li>2. Sistem menampilkan halaman detail UMKM</li> <li>3. Aktor mengklik tombol <i>hapus</i> pada <i>field produk</i> yang akan dihapus</li> <li>4. Sistem menampilkan pesan konfirmasi untuk menghapus data</li> <li>5. Aktor menekan tombol “Ya”</li> <li>6. Sistem menghapus data produk</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>7. Aktor menekan tombol “Batal”</li> <li>8. Sistem batal menghapus data produk</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman detail UMKM

### 14. Use Case Scenario Melakukan Pencarian Produk



<i>Use case</i>	Melakukan pencarian produk
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor masuk ke dalam aplikasi</li> <li>2. Sistem menampilkan halaman <i>dashboard</i> untuk <i>user</i> pembeli</li> <li>3. Aktor melakukan pencarian terhadap produk yang diinginkan</li> <li>4. Sistem menampilkan produk sesuai dengan pencarian aktor</li> </ol>

<i>Alternative Scenario</i>	<p>5. Aktor melakukan pencarian dengan kata kunci yang tidak sesuai</p> <p>6. Sistem menampilkan pesan “produk tidak tersedia”</p>
<i>Exit Condition</i>	Sistem menampilkan halaman <i>dashboard</i>

### 15. Use Case Scenario Melakukan Chat

<i>Use case</i>	Melakukan <i>chat</i>
<i>Actor</i>	Penjual, Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<p>1. Aktor memilih halaman <i>chat</i></p> <p>2. Sistem menampilkan halaman <i>chat</i></p> <p>3. Aktor memilih user untuk melakukan <i>chat</i></p> <p>4. Sistem menampilkan halaman <i>chat</i></p> <p>5. Aktor melakukan <i>chat</i></p>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Aktor berhasil melakukan <i>chat</i>

### 16. Use Case Scenario Mengedit Pesanan di Keranjang

<i>Use case</i>	Mengedit pesanan di keranjang
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<p>1. Aktor memilih halaman keranjang</p> <p>2. Sistem menampilkan halaman keranjang</p> <p>3. Aktor melakukan pengeditan terhadap jumlah produk yang diinginkan</p> <p>4. Sistem menampilkan jumlah hasil edit pembeli</p>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman keranjang

### 17. Use Case Scenario Menghapus Pesanan di Keranjang

<i>Use case</i>	Menghapus pesanan di keranjang
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman keranjang</li> <li>2. Sistem menampilkan halaman keranjang</li> <li>3. Aktor mengklik tombol hapus pada produk yang akan dihapus</li> <li>4. Sistem menampilkan pesan konfirmasi untuk menghapus data</li> <li>5. Aktor menekan tombol “Ya”</li> <li>6. Sistem menghapus data produk</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>7. Aktor menekan tombol “Batal”</li> <li>8. Sistem batal menghapus data produk</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman keranjang

### 18. Use Case Scenario Melihat Noresi Pengiriman Produk

<i>Use case</i>	Melihat noresi pengiriman produk
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman riwayat pembelian</li> <li>2. Sistem menampilkan halaman riwayat pembelian</li> <li>3. Aktor menekan tombol detail pada pesanan yang akan dicek</li> <li>4. Aktor melihat noresi pengiriman produk</li> </ol>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Aktor berhasil melihat noresi

### 19. Use Case Scenario Melihat Laporan Transaksi

<i>Use case</i>	Melihat laporan transaksi
-----------------	---------------------------

<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman laporan transaksi</li> <li>2. Sistem menampilkan halaman laporan transaksi</li> <li>3. Aktor memilih rentang tanggal transaksi</li> <li>4. Aktor menekan tombol <i>view</i></li> <li>5. Sistem menampilkan daftar transaksi sesuai rentang tanggal</li> <li>6. Aktor menekan tombol cetak laporan</li> <li>7. Sistem mencetak hasil laporan transaksi</li> </ol>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Sistem menampilkan laporan transaksi

## 20. Use Case Scenario Melihat Riwayat Transaksi

<i>Use case</i>	Melihat riwayat transaksi
<i>Actor</i>	Pembeli, Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor masuk ke dalam aplikasi</li> <li>2. Sistem menampilkan halaman <i>dashboard</i></li> <li>3. Aktor memilih menu riwayat pembelian</li> <li>4. Sistem menampilkan halaman riwayat pembelian</li> </ol>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Sistem menampilkan riwayat pembelian

## 21. Use Case Scenario Logout

<i>Use case</i>	<i>Logout</i>
<i>Actor</i>	Admin, Penjual, Pembeli, Kurir
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor masuk ke dalam aplikasi</li> </ol>

	<ol style="list-style-type: none"> <li>2. Sistem menampilkan halaman aplikasi</li> <li>3. Aktor menekan tombol <i>logout</i></li> <li>4. Sistem menampilkan halaman <i>landing page</i></li> </ol>
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Sistem menampilkan halaman <i>landing page</i>

## 22. Use Case Scenario Menambahkan Data UMKM

<i>Use case</i>	Menambah data UMKM
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman UMKM</li> <li>2. Sistem menampilkan halaman UMKM</li> <li>3. Aktor mengklik tombol tambah</li> <li>4. Sistem menampilkan halaman <i>form</i> tambah data UMKM</li> <li>5. Aktor menginputkan data yang ada pada <i>form</i> tambah UMKM</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data UMKM baru</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data UMKM

## 23. Use Case Scenario Mengedit Data UMKM

<i>Use case</i>	Mengedit data UMKM
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman data UMKM</li> <li>2. Sistem menampilkan halaman UMKM</li> </ol>

	<ol style="list-style-type: none"> <li>3. Aktor mengklik tombol edit pada <i>field</i> UMKM yang akan diedit</li> <li>4. Sistem menampilkan halaman form edit data UMKM</li> <li>5. Aktor mengedit data yang ingin diupdate</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data UMKM yang telah diedit</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data UMKM

#### 24. Use Case Scenario Menghapus Data UMKM

<i>Use case</i>	Menghapus data UMKM
<i>Actor</i>	Penjual
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman UMKM</li> <li>2. Sistem menampilkan halaman UMKM</li> <li>3. Aktor mengklik tombol hapus pada <i>field</i> UMKM yang akan dihapus</li> <li>4. Sistem menampilkan pesan konfirmasi untuk menghapus data</li> <li>5. Aktor menekan tombol “Ya”</li> <li>6. Sistem menghapus data UMKM</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>7. Aktor menekan tombol “Batal”</li> <li>8. Sistem batal menghapus data UMKM</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman data UMKM

## 25. Use Case Scenario Melihat List Pembelian yang Belum Dibayar

<i>Use case</i>	Melihat list pembelian yang belum dibayar
<i>Actor</i>	Pembeli
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor masuk ke dalam aplikasi</li> <li>2. Sistem menampilkan halaman <i>dashboard</i></li> <li>3. Aktor memilih menu <i>purchase</i></li> <li>4. Sistem menampilkan daftar pembelian yang belum dibayar</li> </ol>
<i>Alternative Scenario</i>	
<i>Exit Condition</i>	Sistem menampilkan halaman <i>purchase</i>

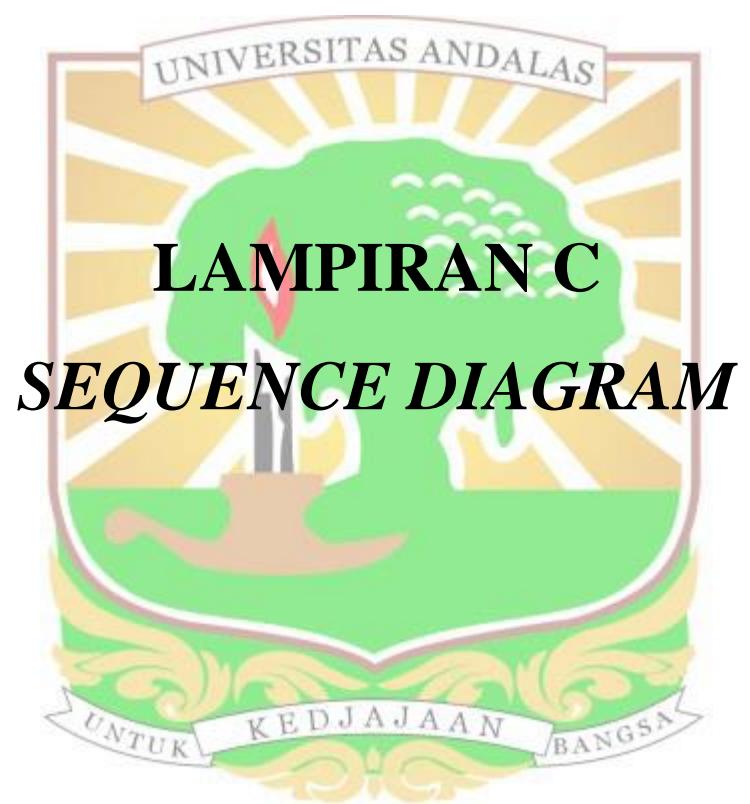
## 26. Use Case Scenario Menginputkan Bukti Pengiriman

<i>Use case</i>	Menginputkan bukti pengiriman
<i>Actor</i>	Kurir
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	<ol style="list-style-type: none"> <li>1. Aktor memilih halaman pengiriman</li> <li>2. Sistem menampilkan halaman pengiriman</li> <li>3. Aktor mengklik tombol detail</li> <li>4. Sistem menampilkan halaman detail pengiriman</li> <li>5. Aktor menginputkan bukti pengiriman</li> <li>6. Aktor menekan tombol simpan</li> <li>7. Sistem menyimpan data bukti pengiriman</li> </ol>
<i>Alternative Scenario</i>	<ol style="list-style-type: none"> <li>8. Aktor menekan tombol simpan ketika data yang diinputkan salah atau kosong</li> <li>9. Sistem menampilkan pesan kesalahan</li> </ol>
<i>Exit Condition</i>	Sistem menampilkan halaman pengiriman

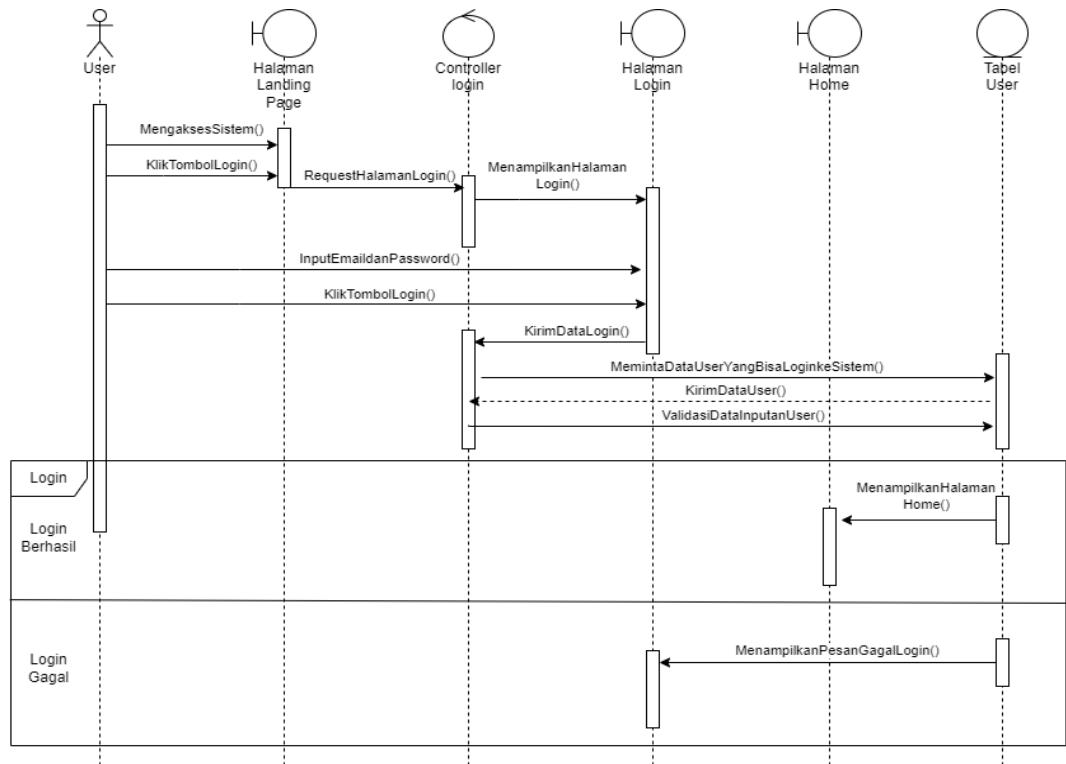
## 27. Use Case Scenario Melihat Riwayat Pengiriman

<i>Use case</i>	Melihat riwayat pengiriman
<i>Actor</i>	Kurir
<i>Entry Condition</i>	Aktor berhasil <i>login</i>
<i>Flow of Event</i>	5. Aktor masuk ke dalam aplikasi 6. Sistem menampilkan halaman dashboard 7. Aktor memilih menu riwayat pengirimna 8. Sistem menampilkan halaman riwayat pengiriman
<i>Alternative Scenario</i>	-
<i>Exit Condition</i>	Sistem menampilkan riwayat pengiriman

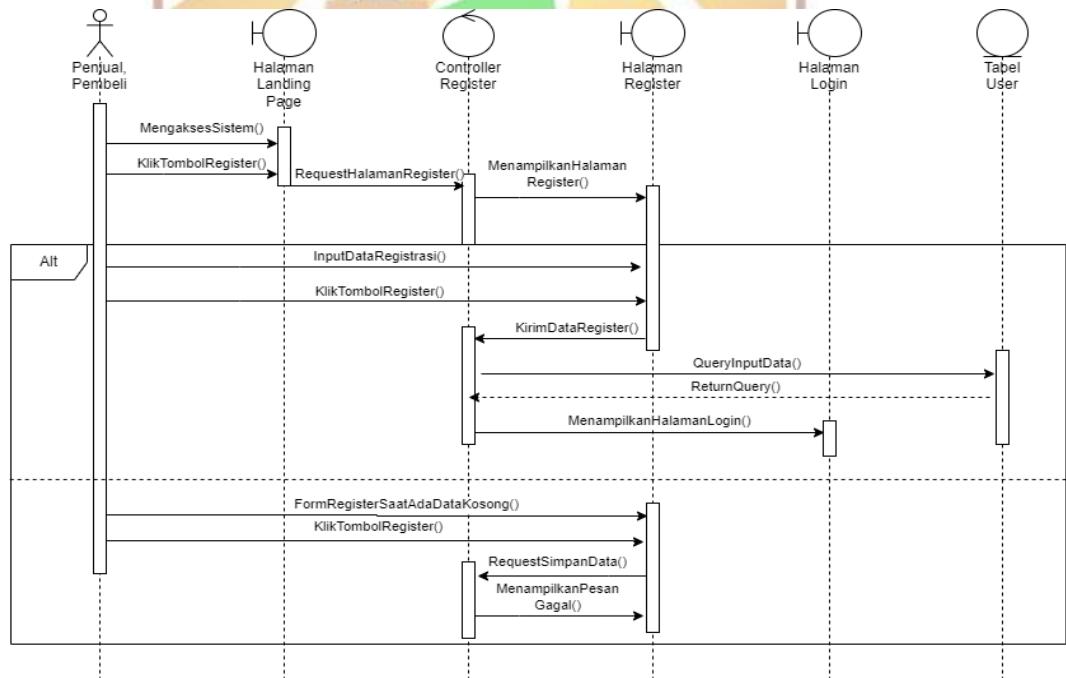




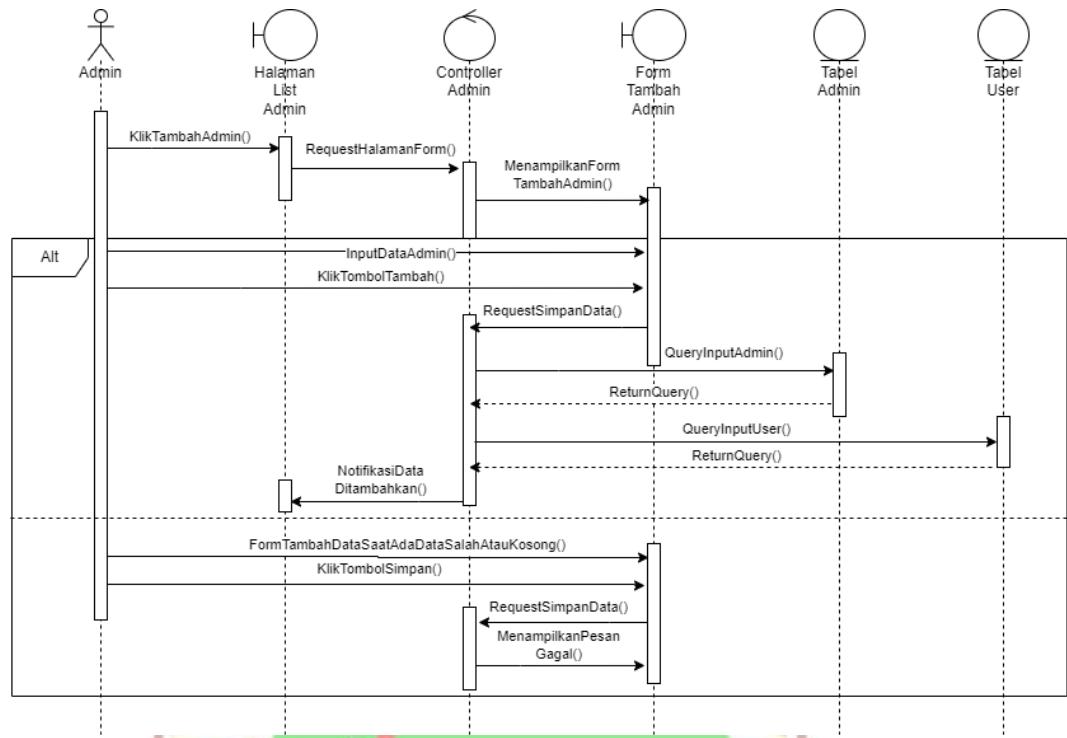
## 1. Sequence Diagram Login



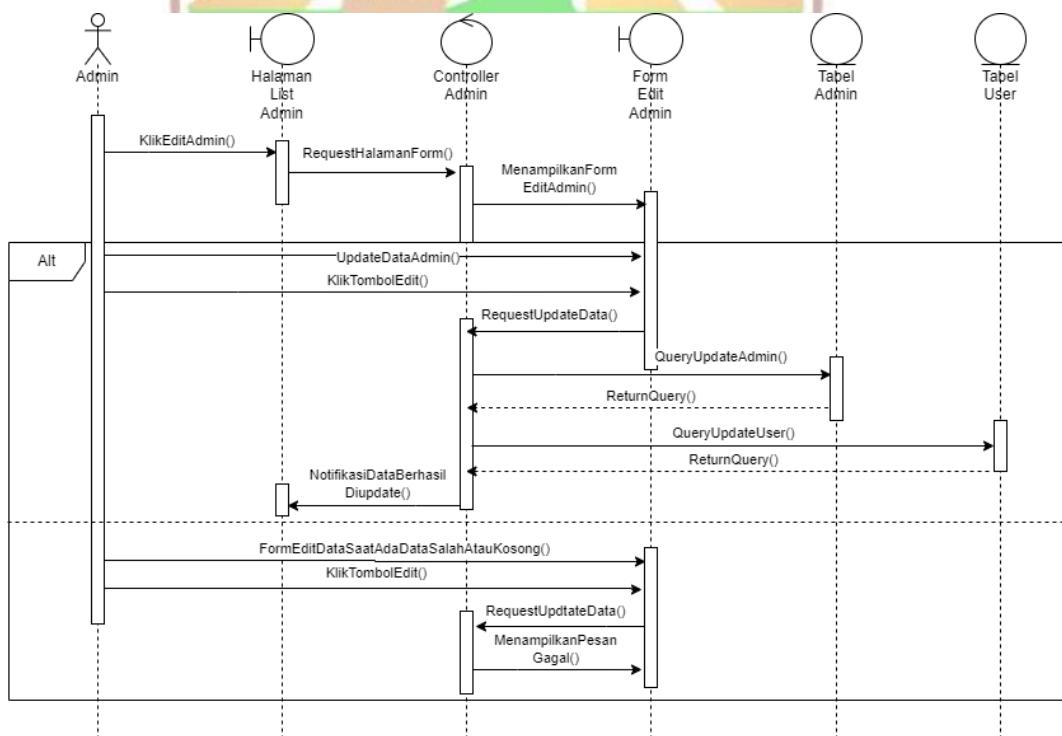
## 2. Sequence Diagram Registrasi



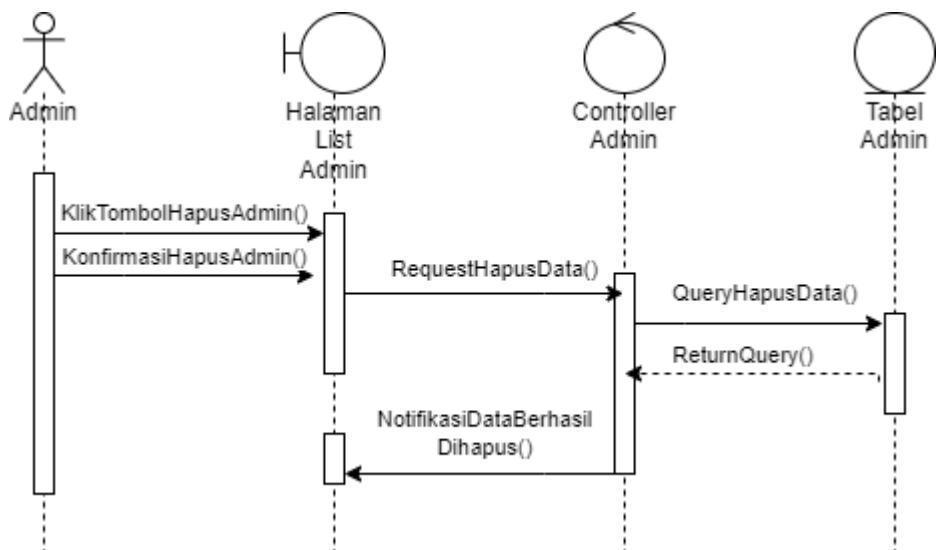
### 3. Sequence Diagram Menambah Data Admin



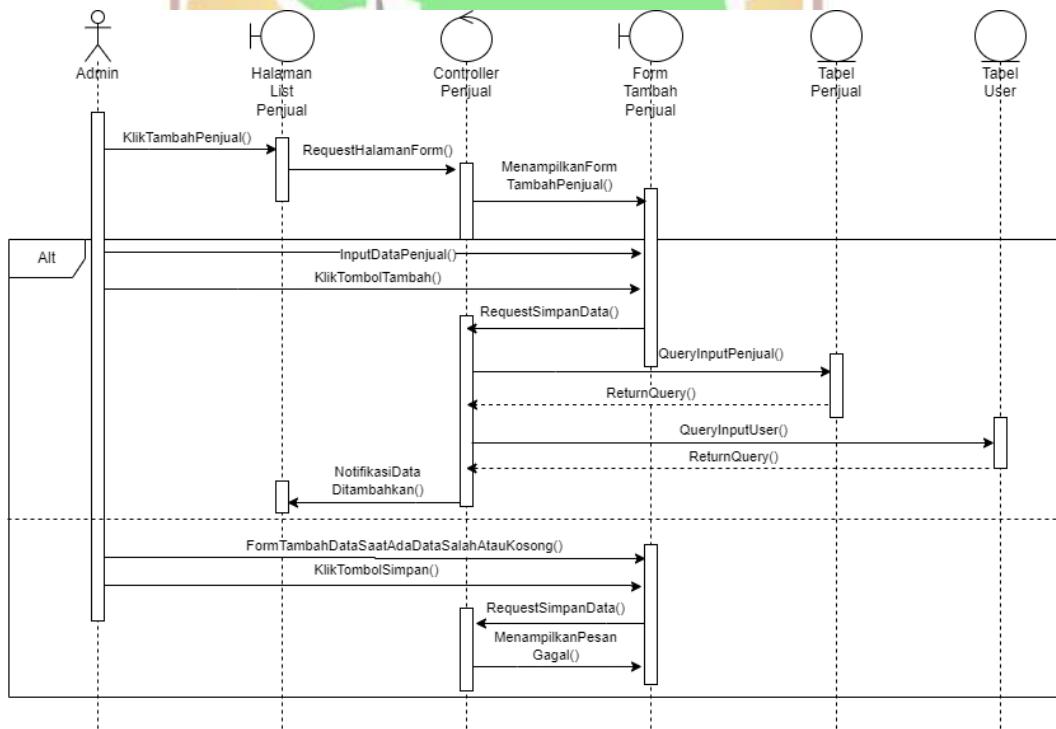
### 4. Sequence Diagram Mengedit Data Admin



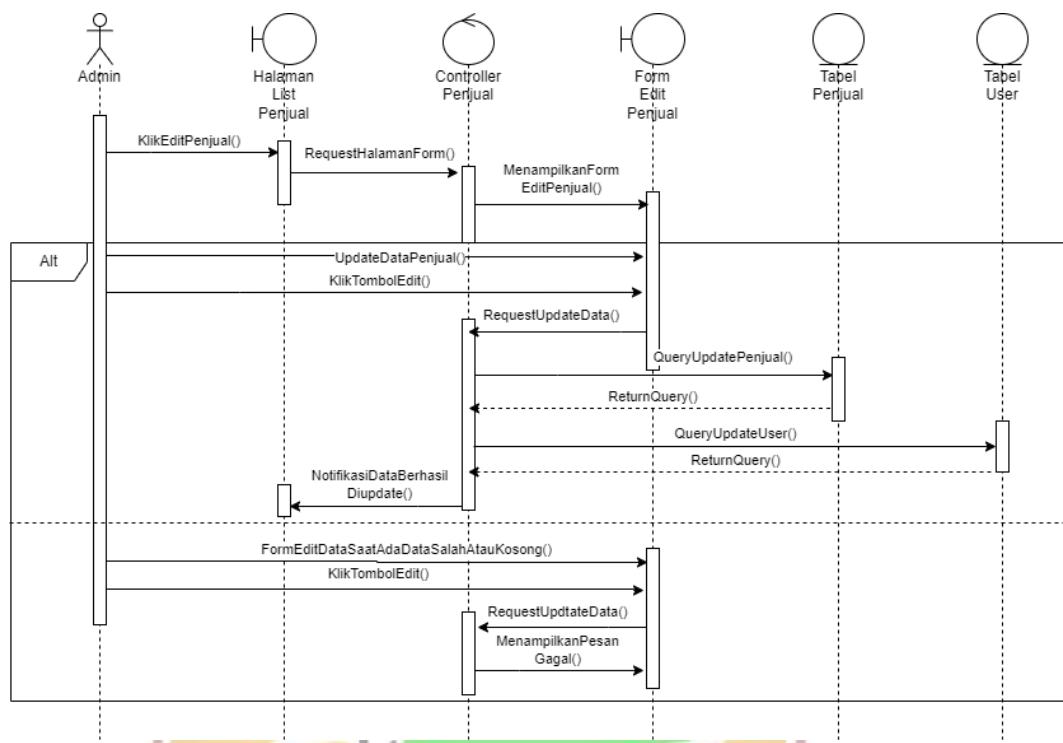
## 5. Sequence Diagram Menghapus Data Admin



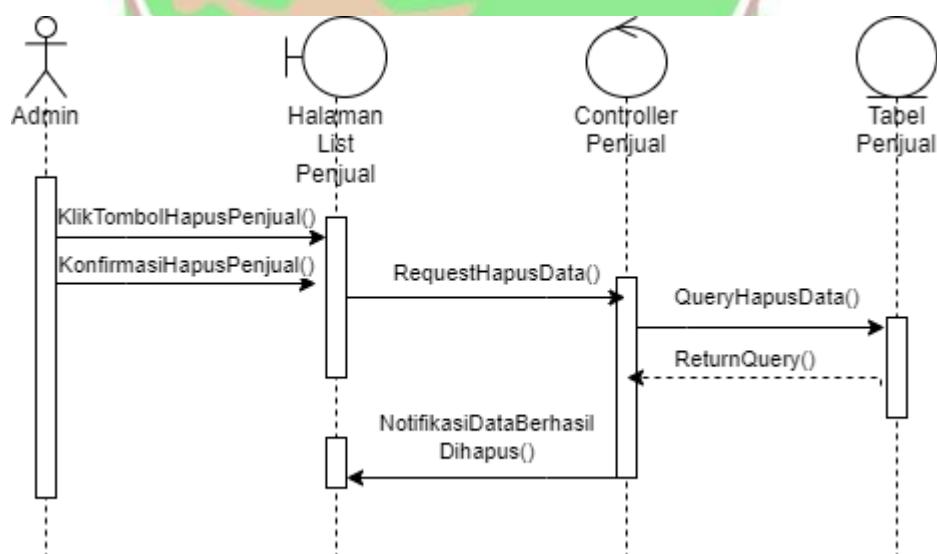
## 6. Sequence Diagram Menambah Data Penjual



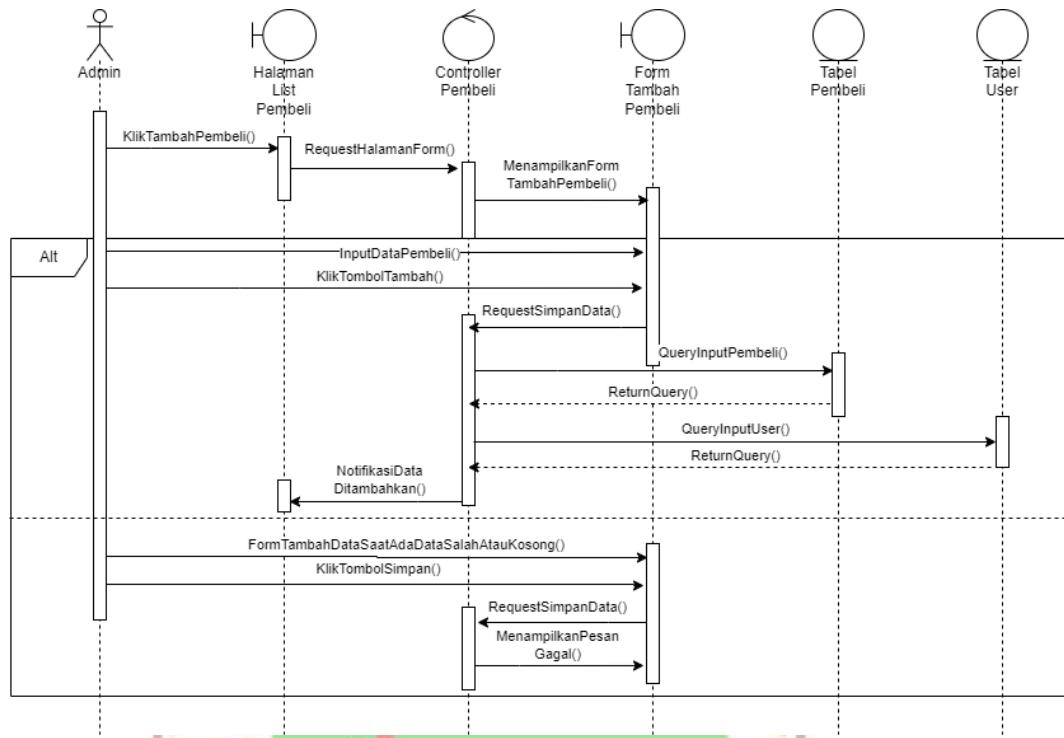
## 7. Sequence Diagram Mengedit Data Penjual



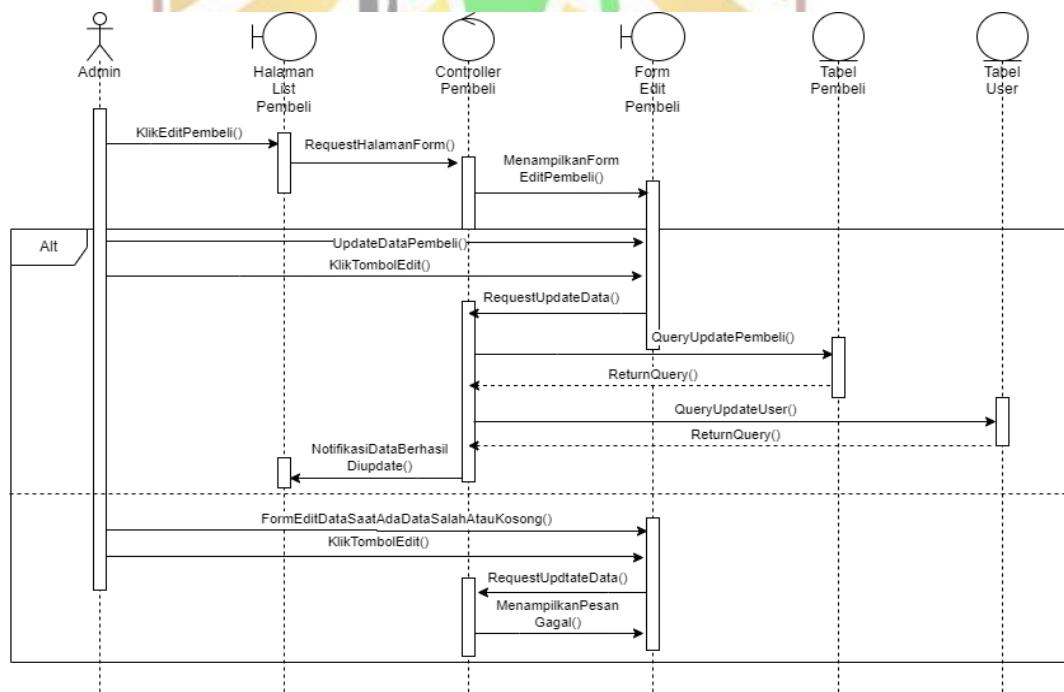
## 8. Sequence Diagram Menghapus Data Penjual



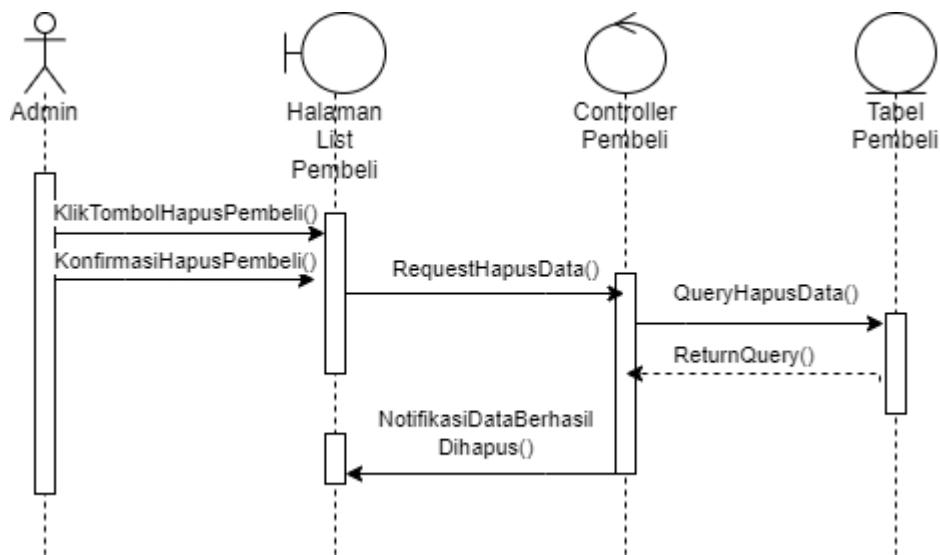
## 9. Sequence Diagram Menambah Data Pembeli



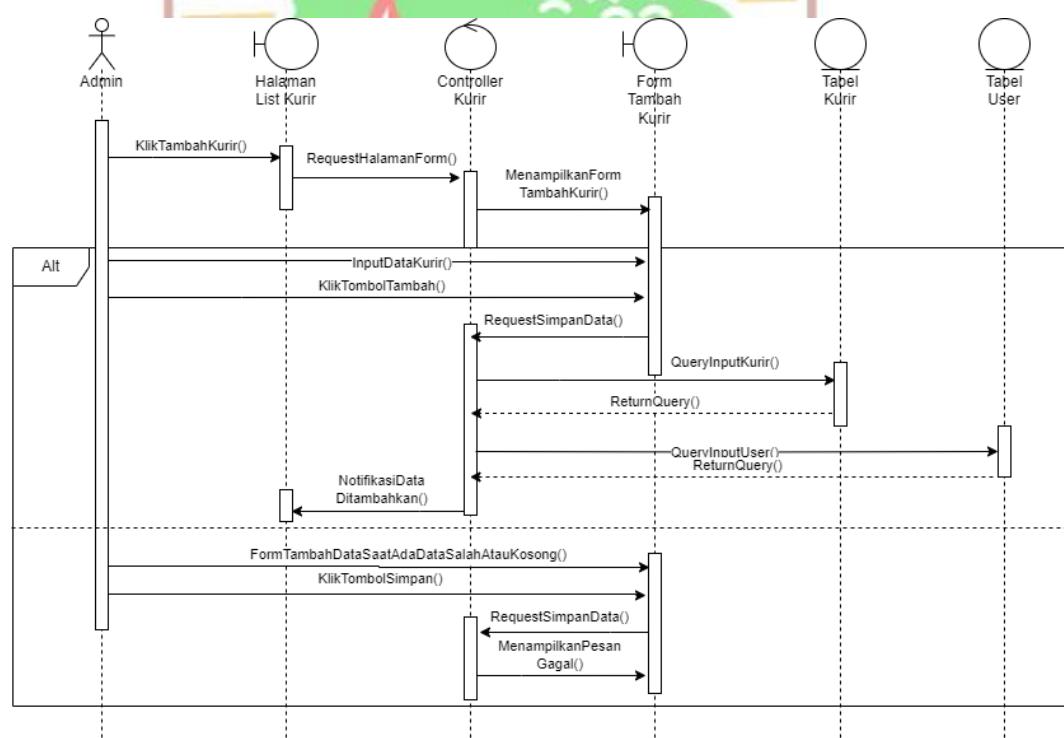
## 10. Sequence Diagram Mengedit Data Pembeli



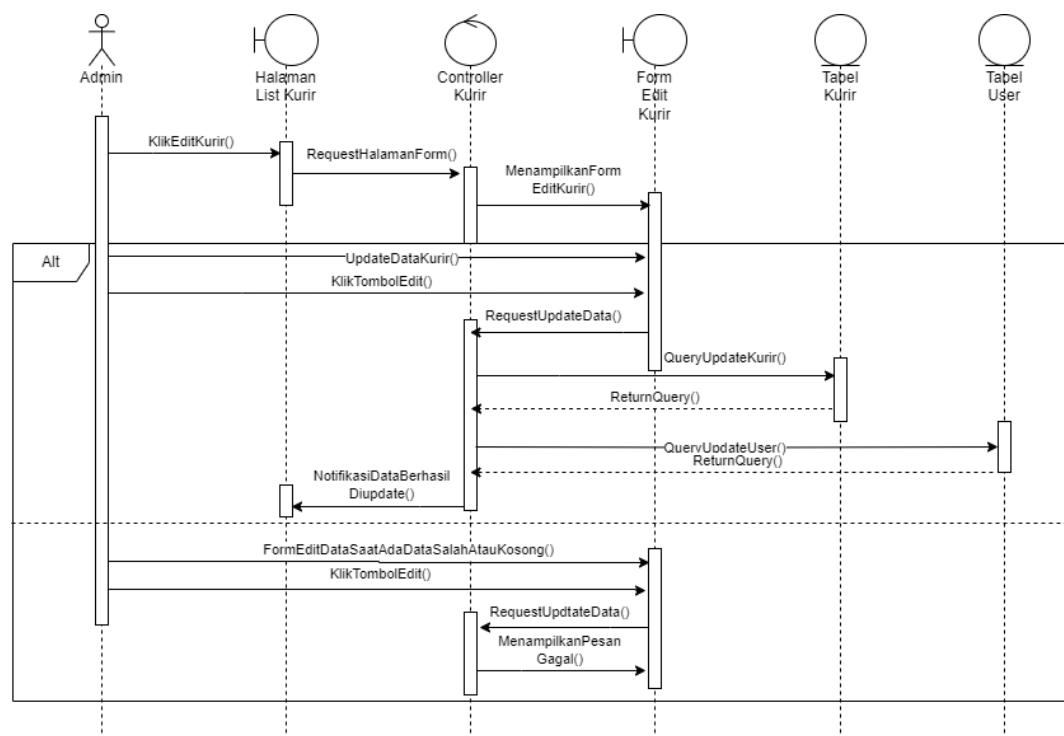
### 11. Sequence Diagram Menghapus Data Pembeli



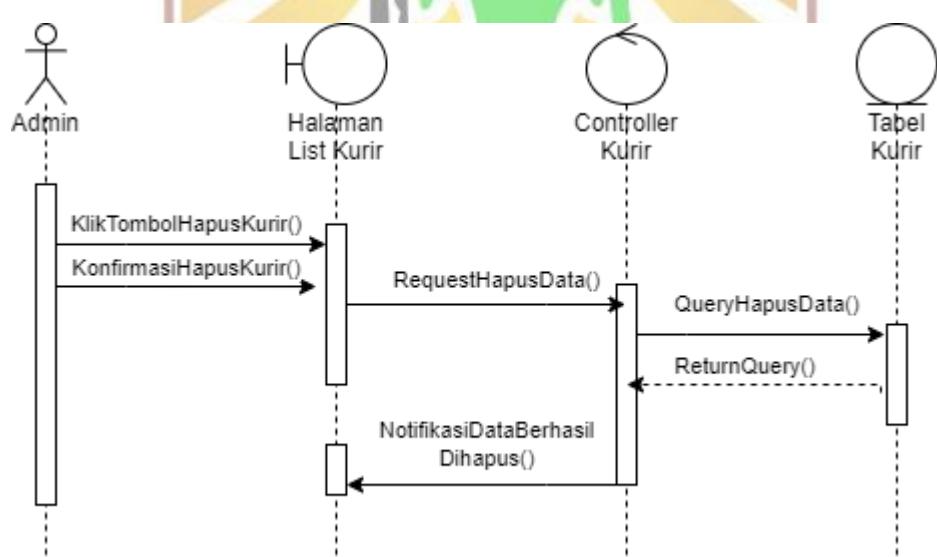
### 12. Sequence Diagram Menambah Data Kurir



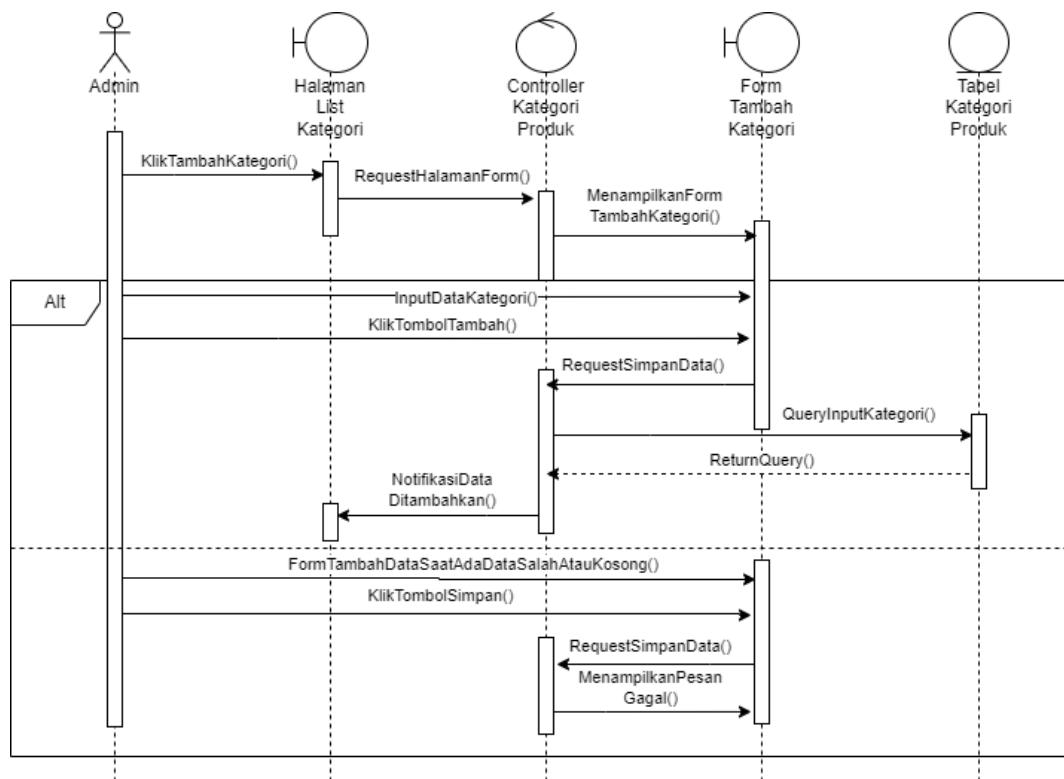
### 13. Sequence Diagram Mengubah Data Kurir



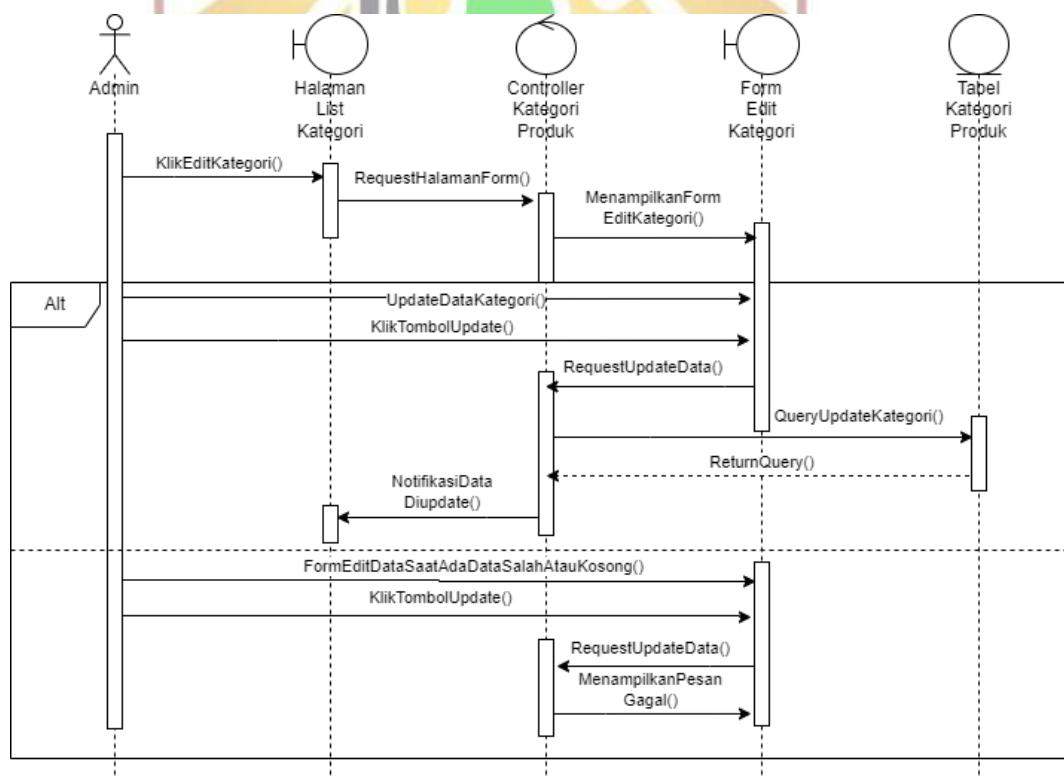
### 14. Sequence Diagram Menghapus Data Kurir



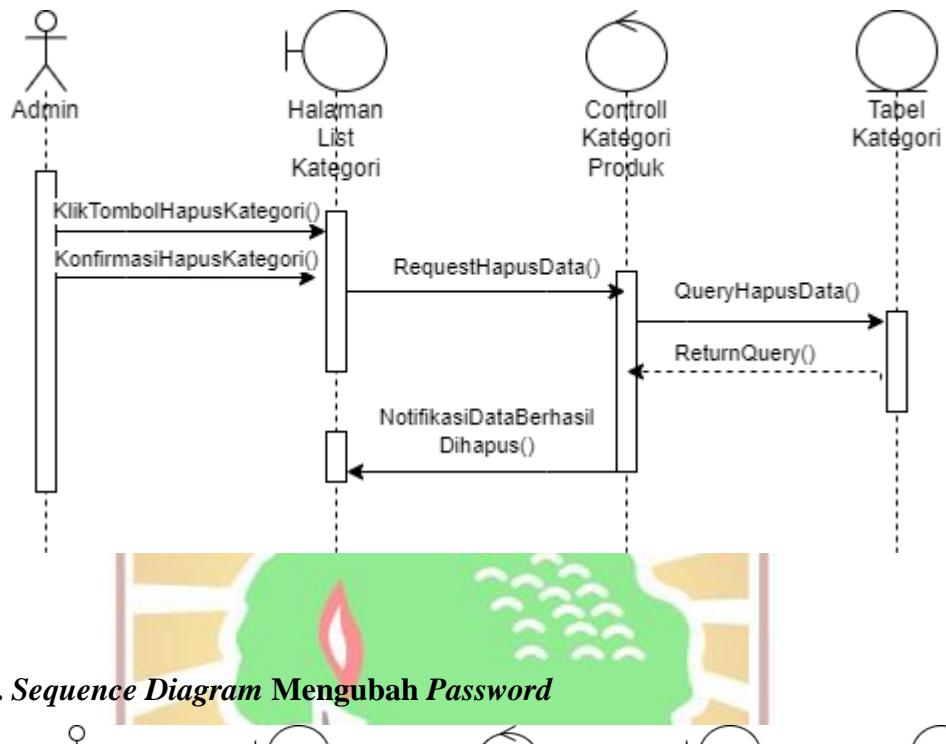
### 15. Sequence Diagram Menambah Data Kategori Produk



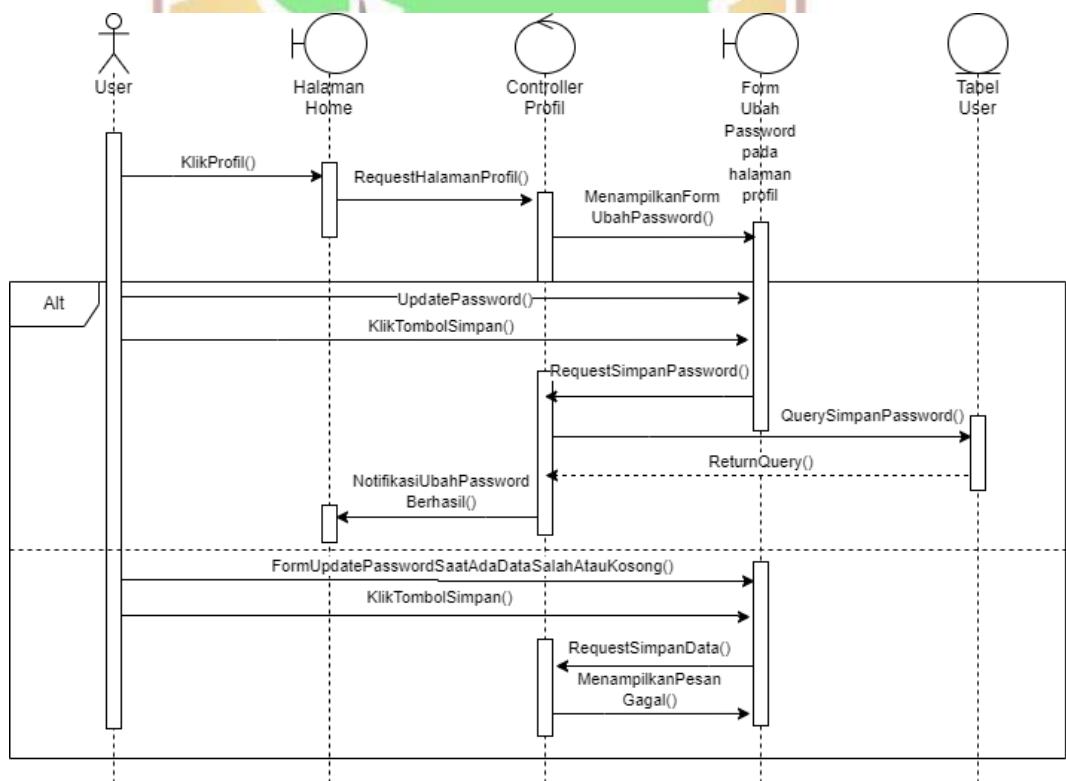
### 16. Sequence Diagram Mengedit Data Kategori Produk



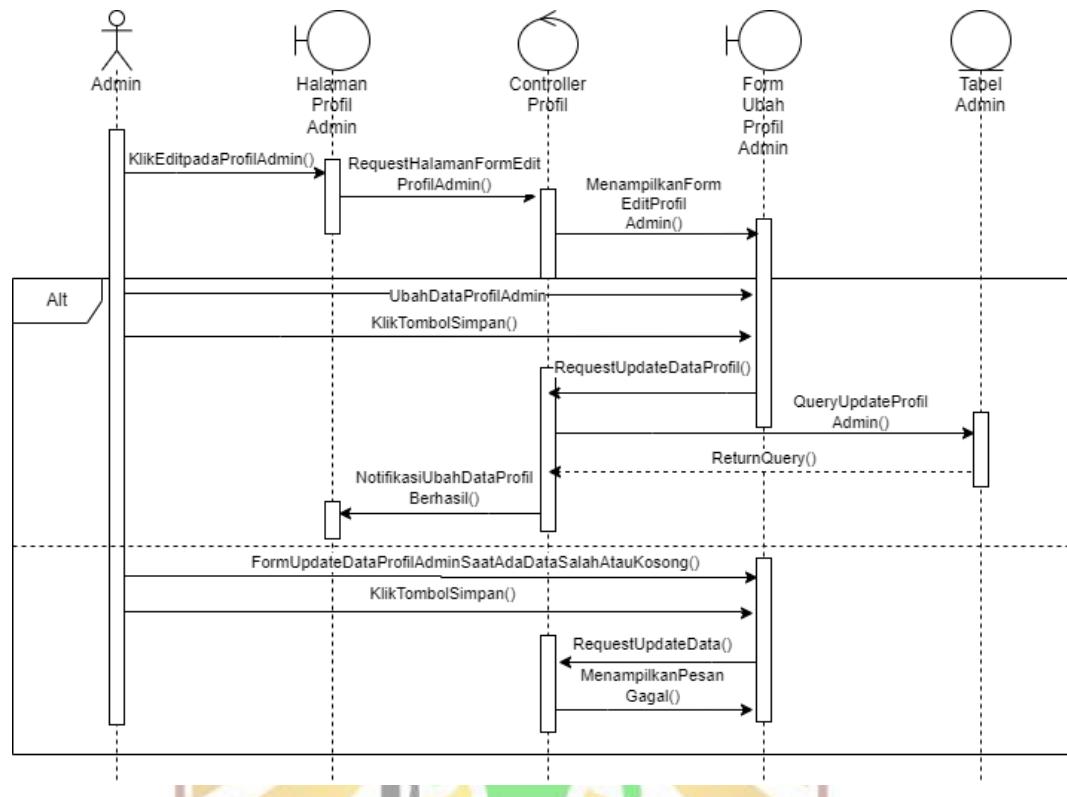
### 17. Sequence Diagram Menghapus Data Kategori Produk



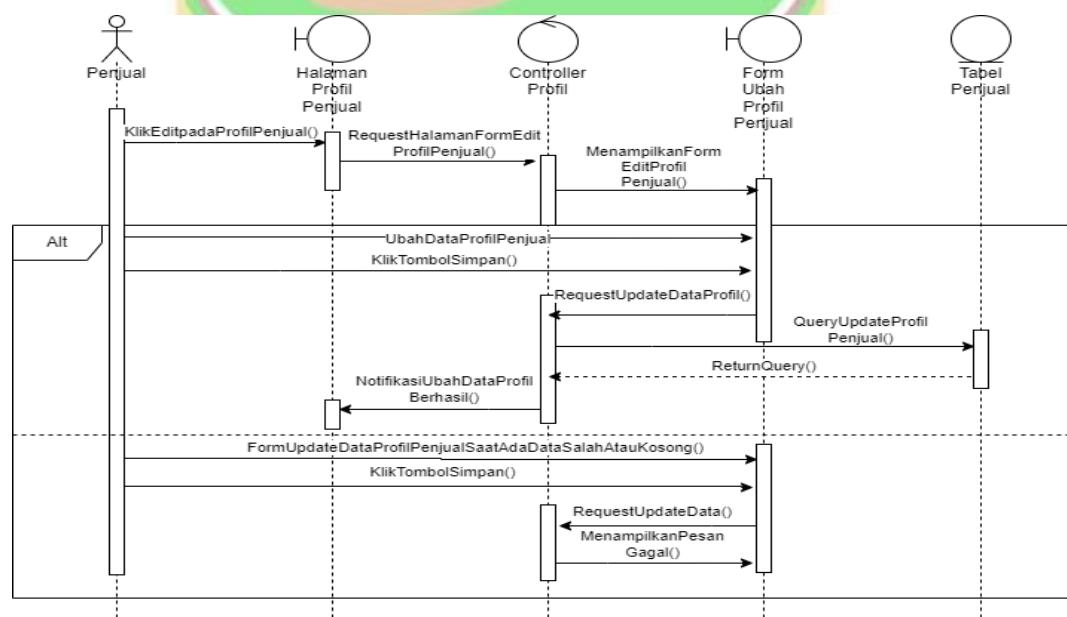
### 18. Sequence Diagram Mengubah Password



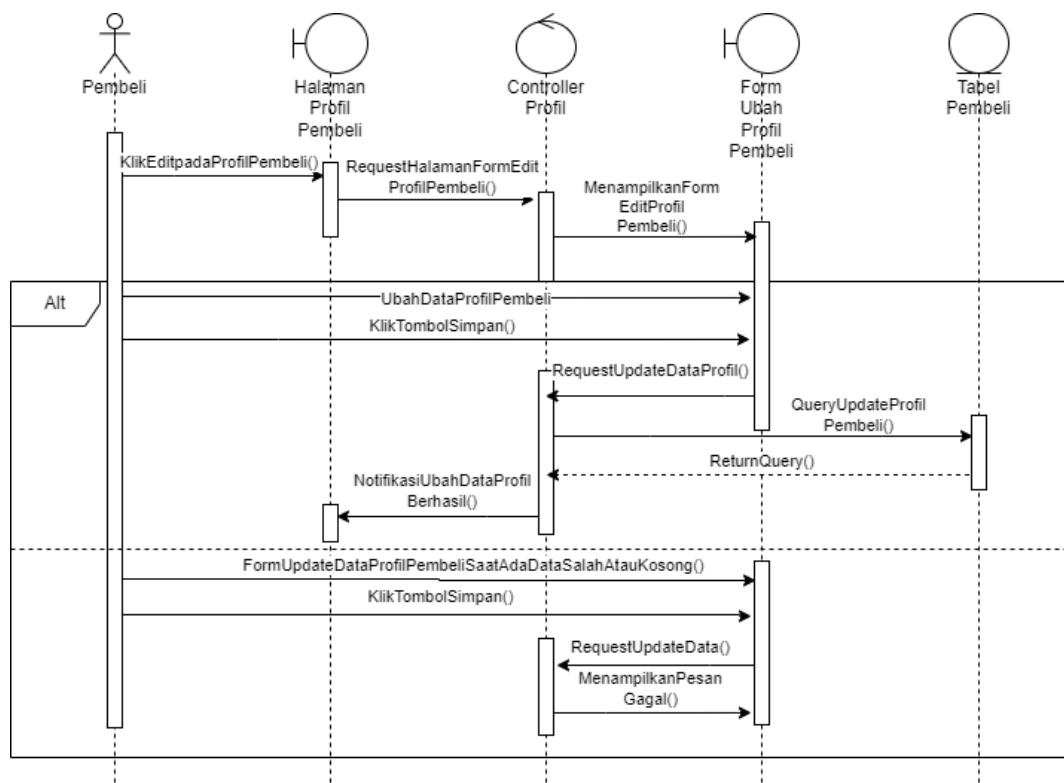
### 19. Sequence Diagram Mengubah Data Profil Admin



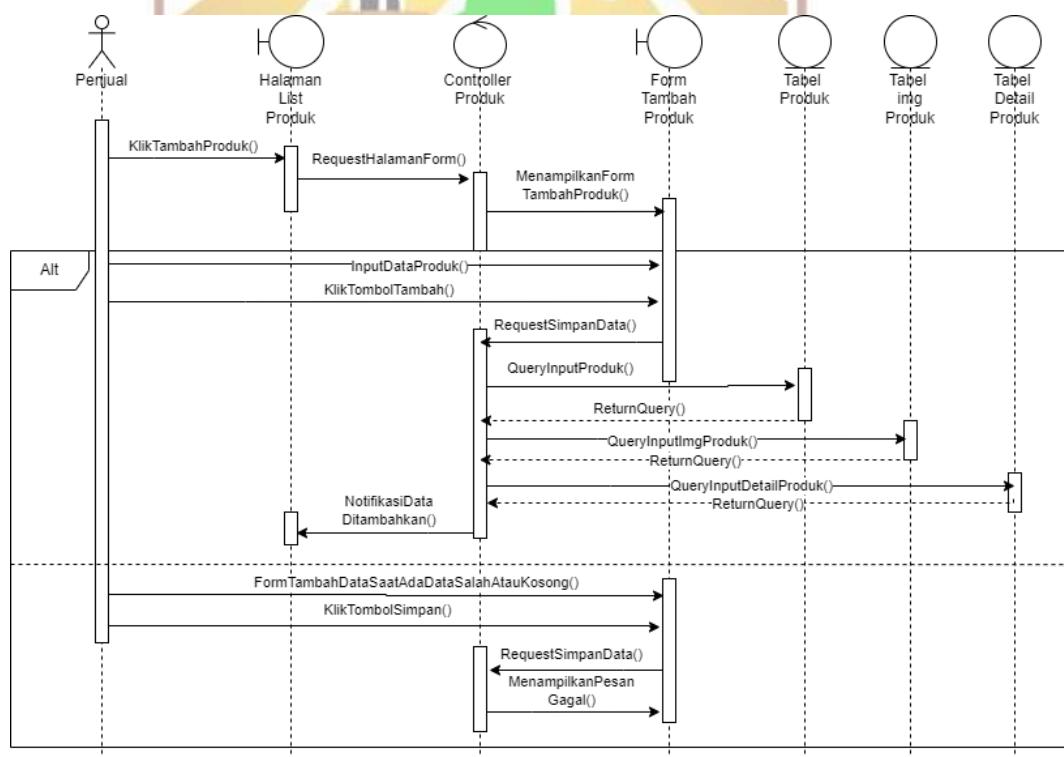
### 20. Sequence Diagram Mengubah Data Profil Penjual



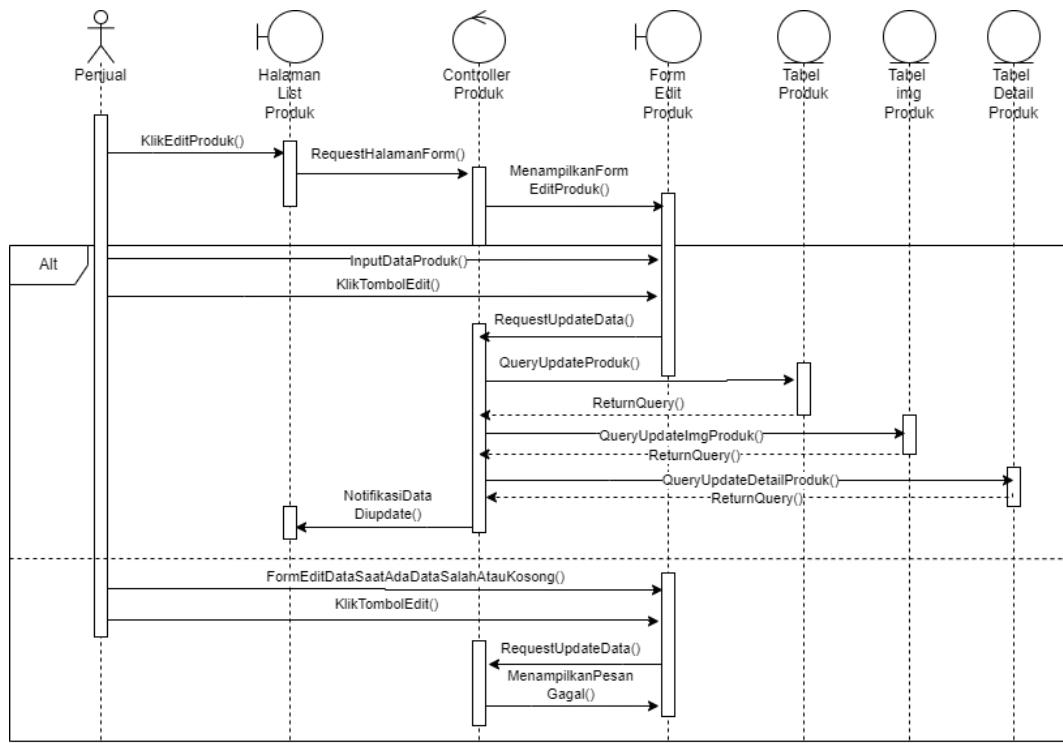
## 21. Sequence Diagram Mengubah Data Profil Pembeli



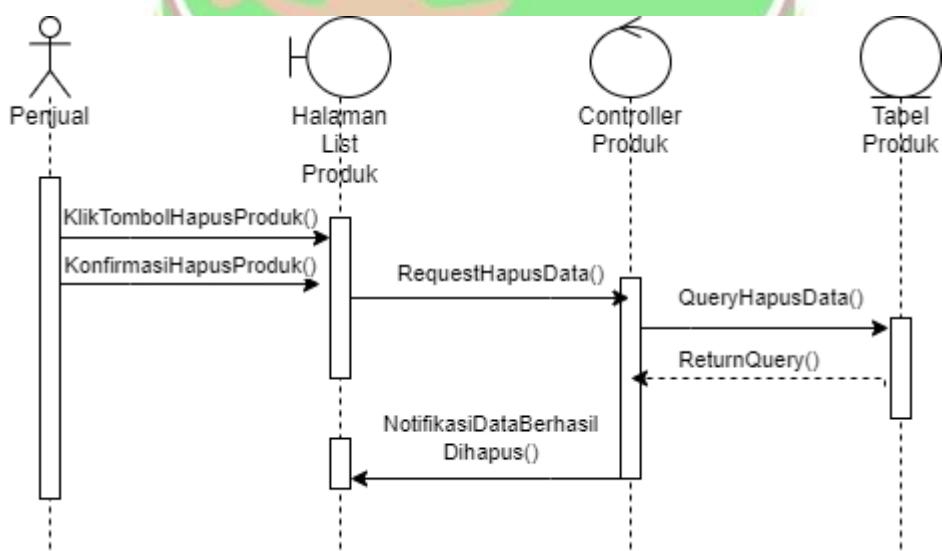
## 22. Sequence Diagram Menambah Data Produk



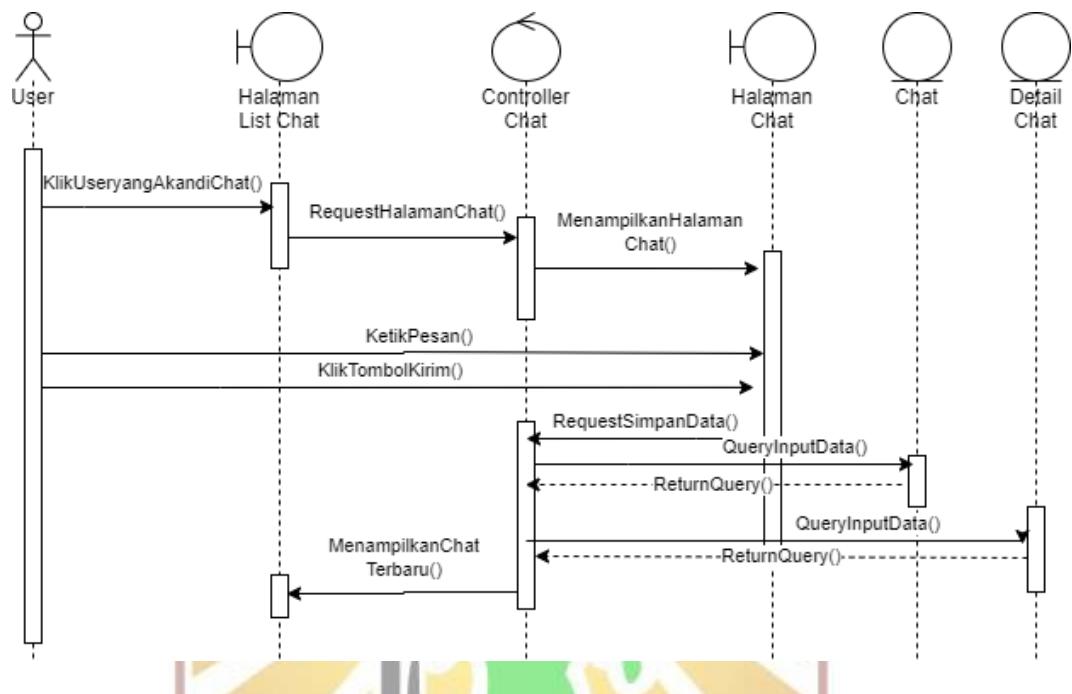
### 23. Sequence Diagram Mengedit Data Produk



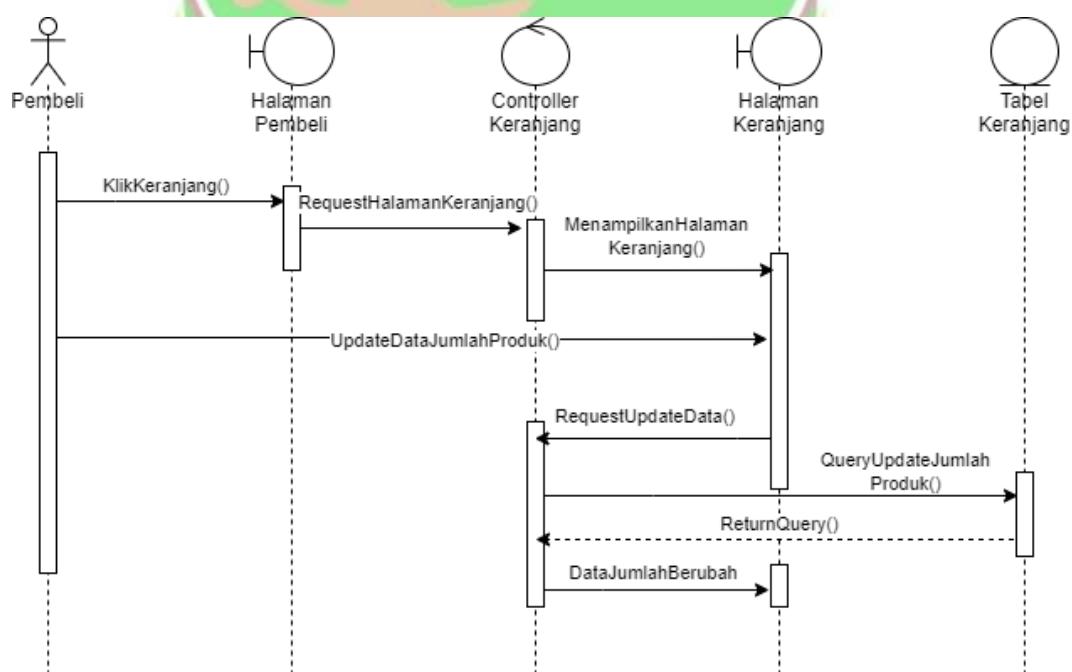
### 24. Sequence Diagram Menghapus Data Produk



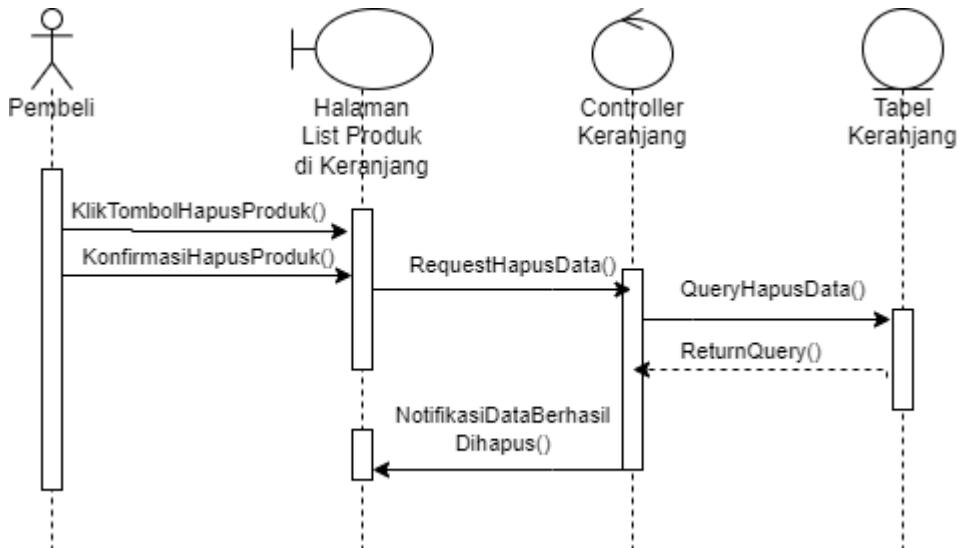
### 25. Sequence Diagram Melakukan Chat



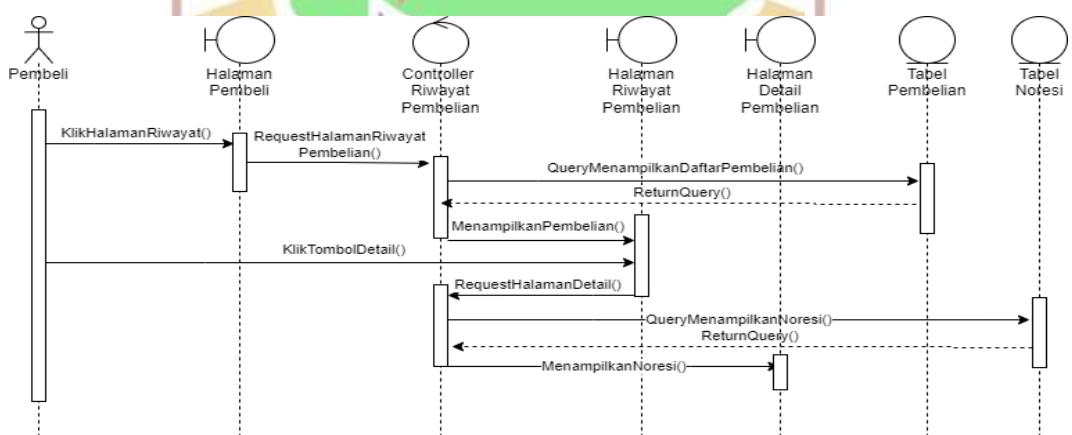
### 26. Sequence Diagram Mengedit Pesanan di Keranjang



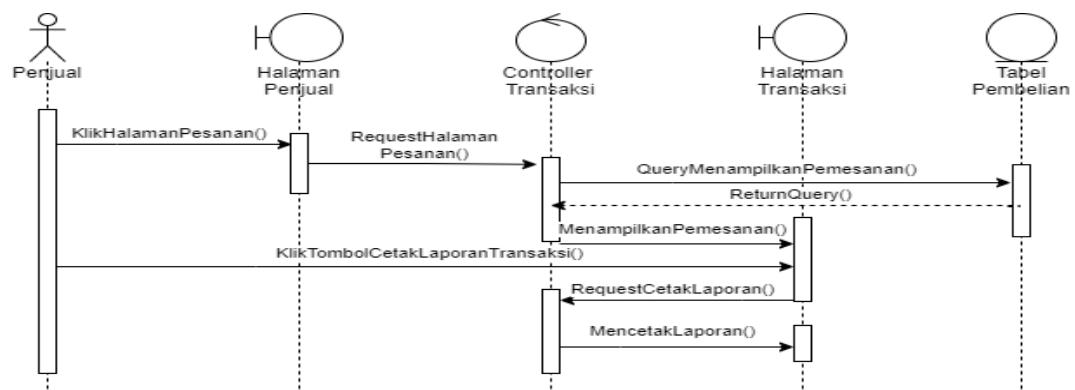
### 27. Sequence Diagram Menghapus Pesanan di Keranjang



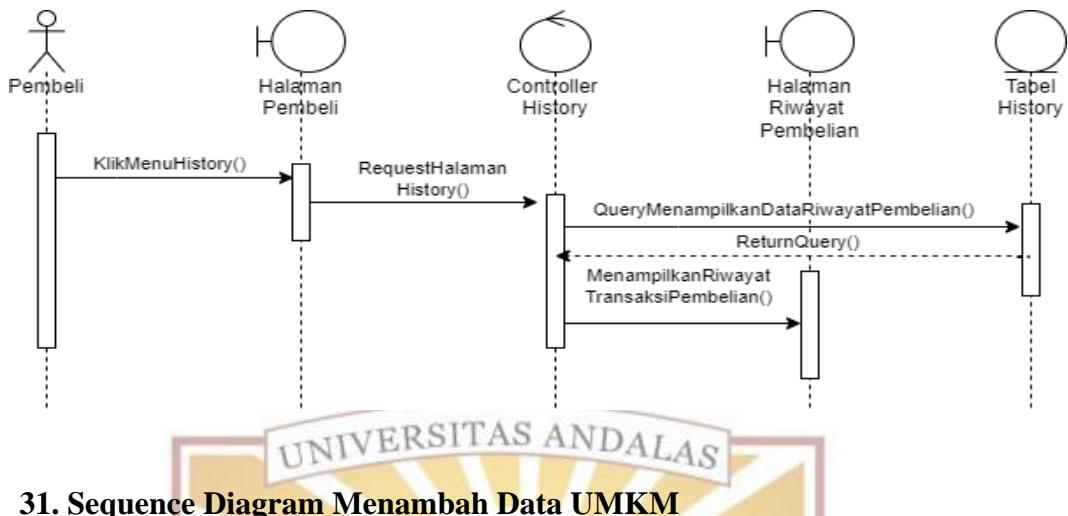
### 28. Sequence Diagram Melihat Noresi Pengiriman Produk



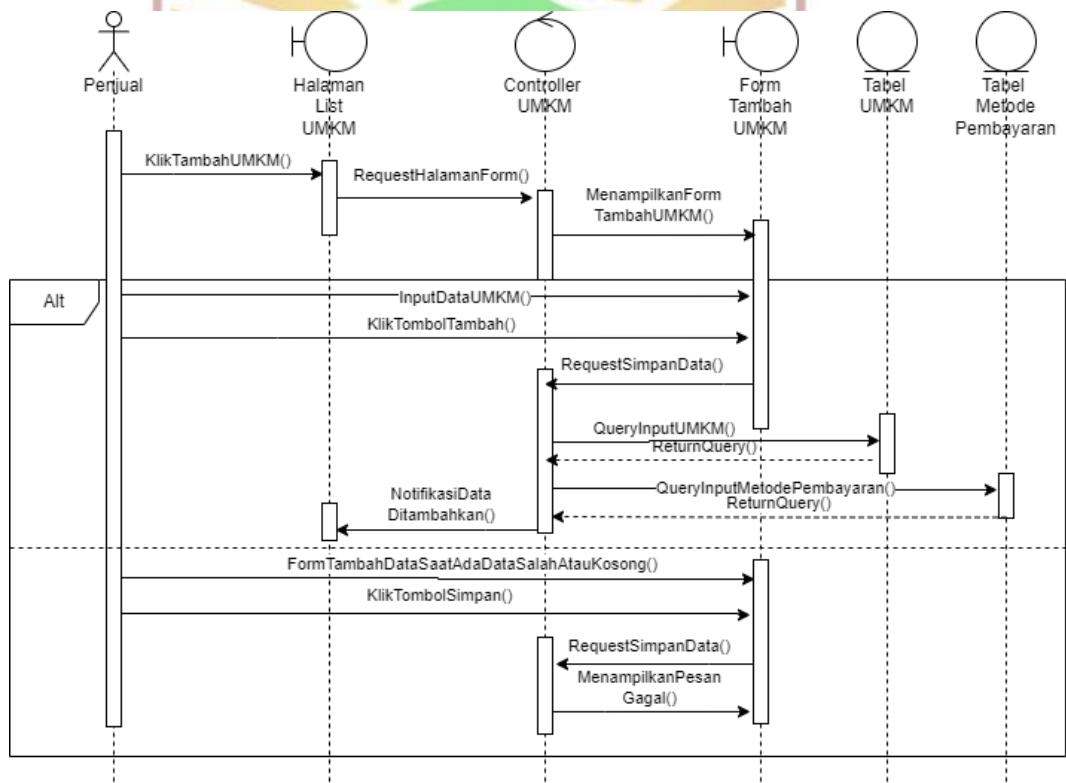
### 29. Sequence Diagram Melihat Laporan Transaksi



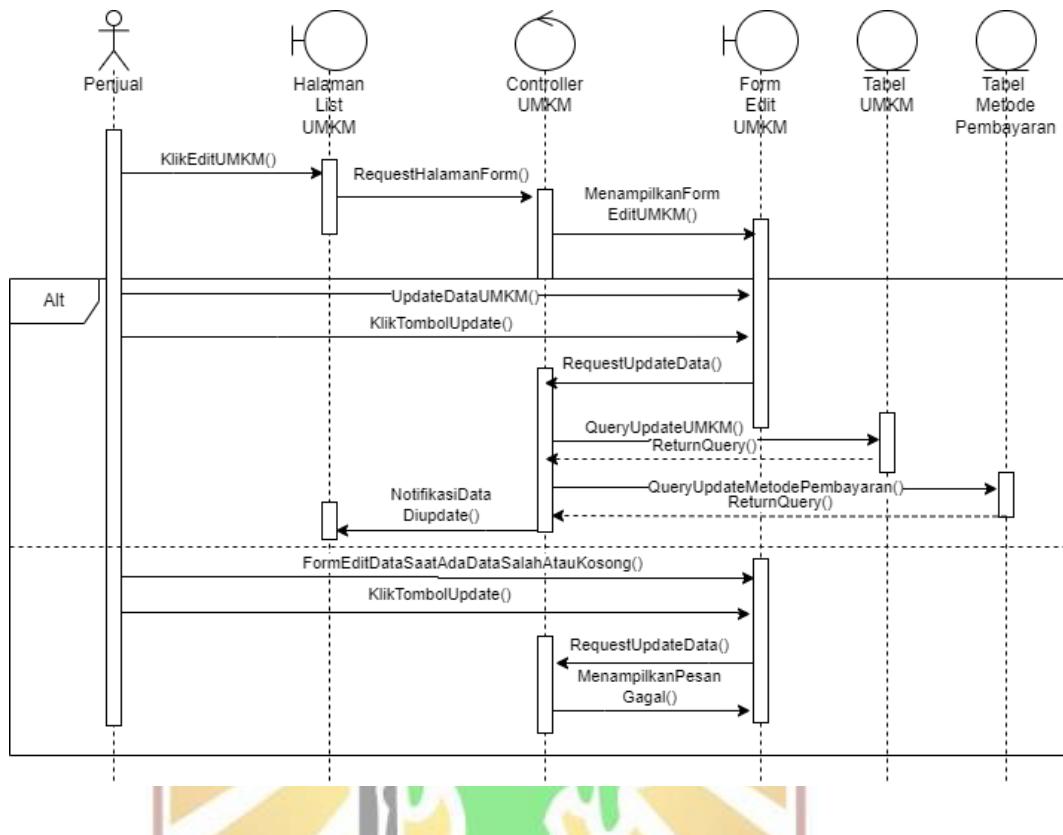
### 30. Sequence Diagram Melihat Riwayat Pembelian



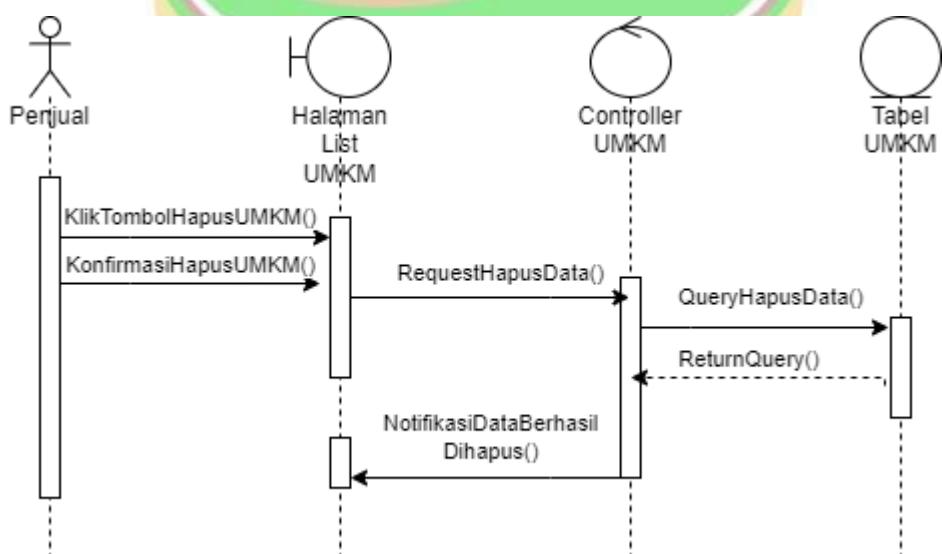
### 31. Sequence Diagram Menambah Data UMKM



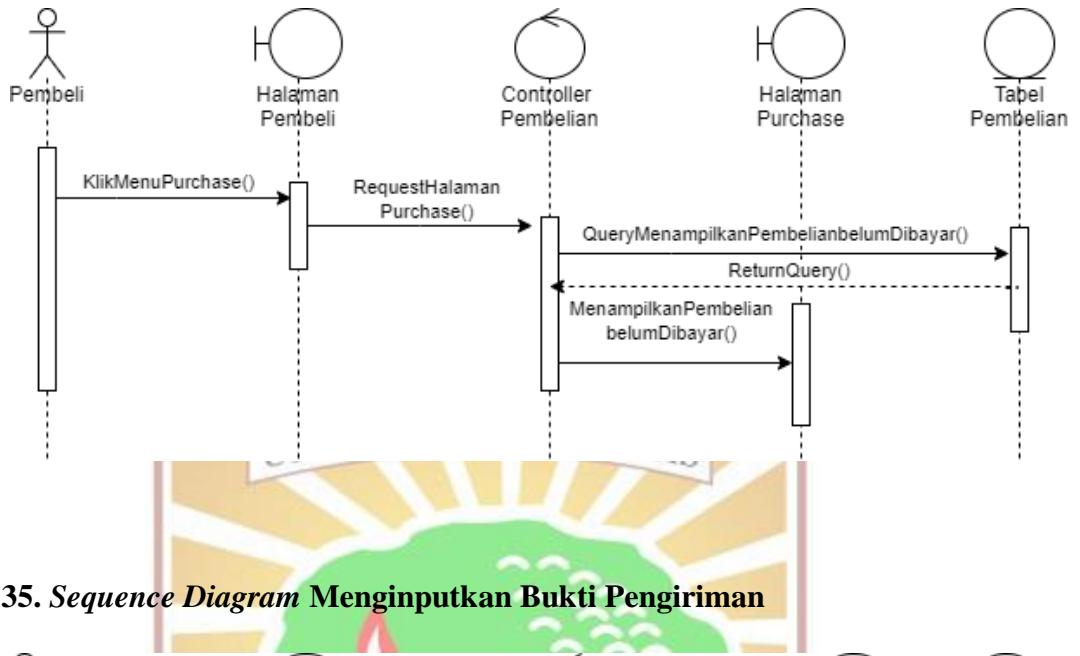
### 32. Sequence Diagram Mengedit Data UMKM



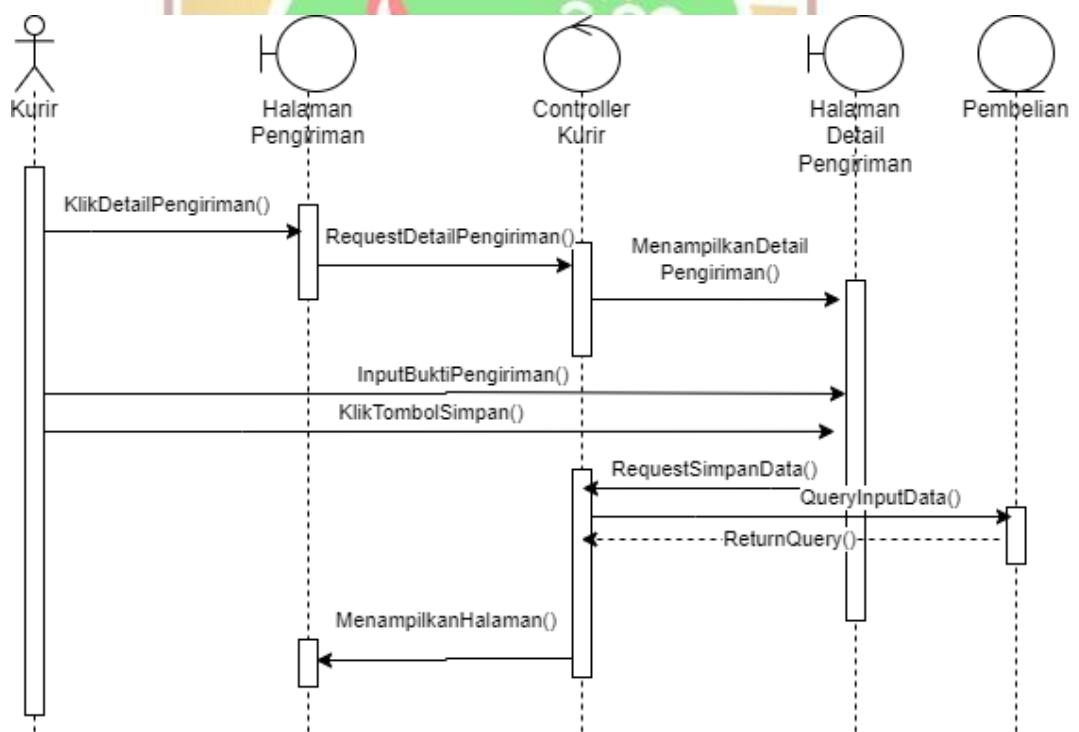
### 33. Sequence Diagram Menghapus Data UMKM



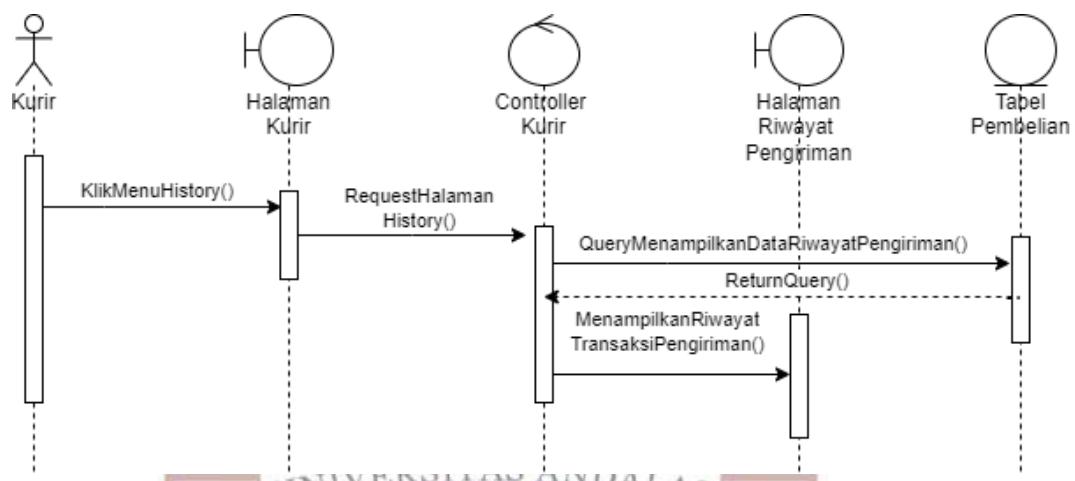
### 34. Sequence Diagram Melihat Pembelian Belum Dibayar



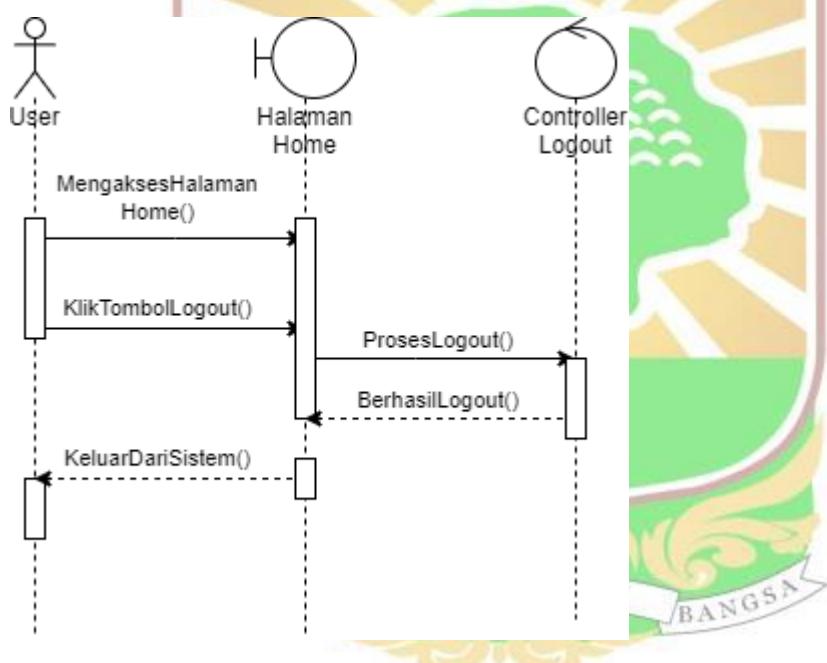
### 35. Sequence Diagram Menginputkan Bukti Pengiriman

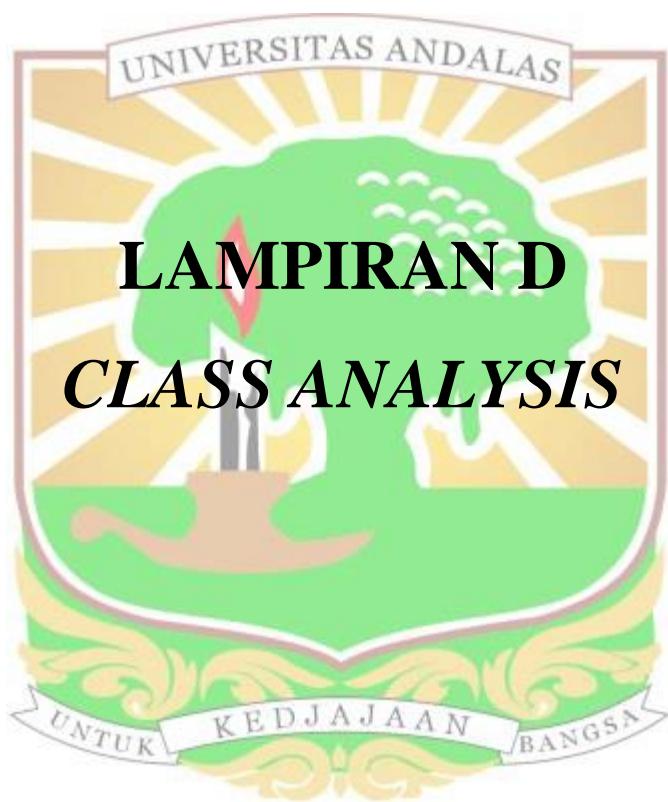


### 36. Sequence Diagram Melihat Riwayat Pengiriman

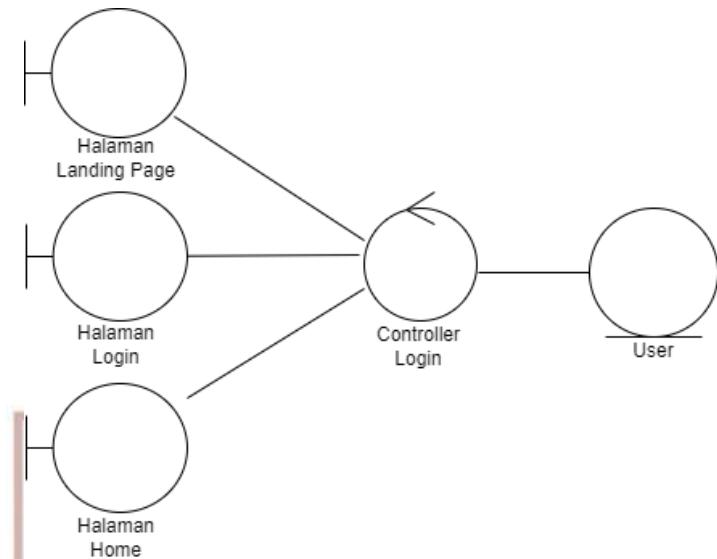


### 37. Sequence Diagram Logout

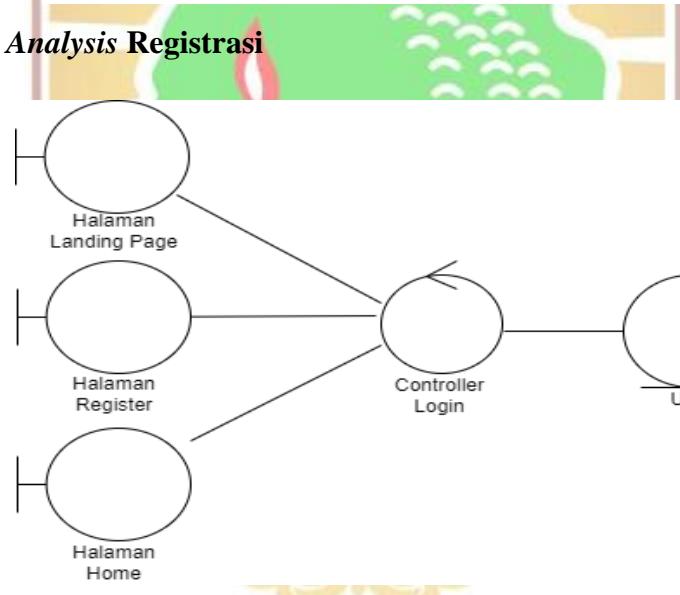




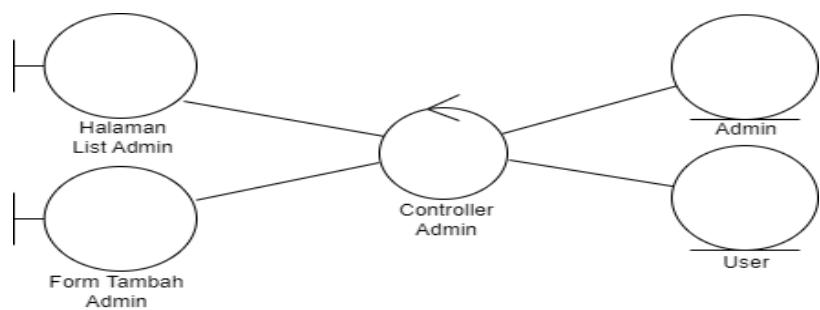
### **1. Class Analysis Login**



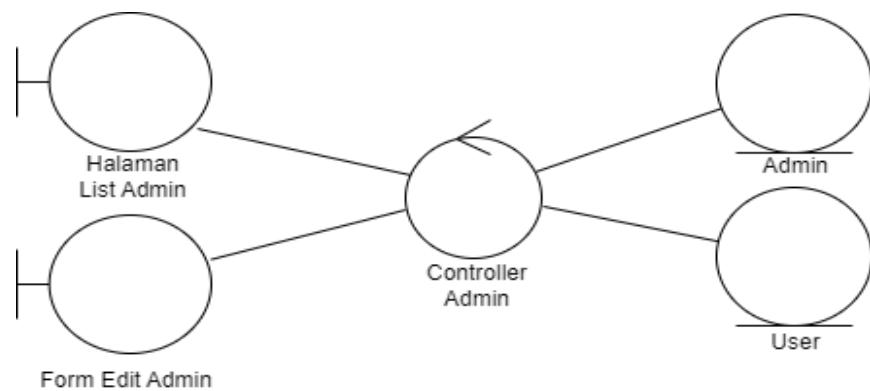
### **2. Class Analysis Registrasi**



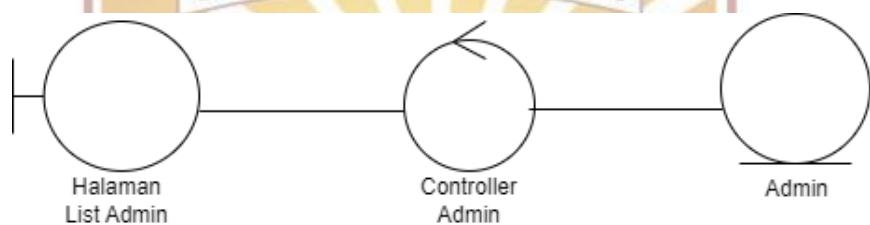
### **3. Class Analysis Menambah Data Admin**



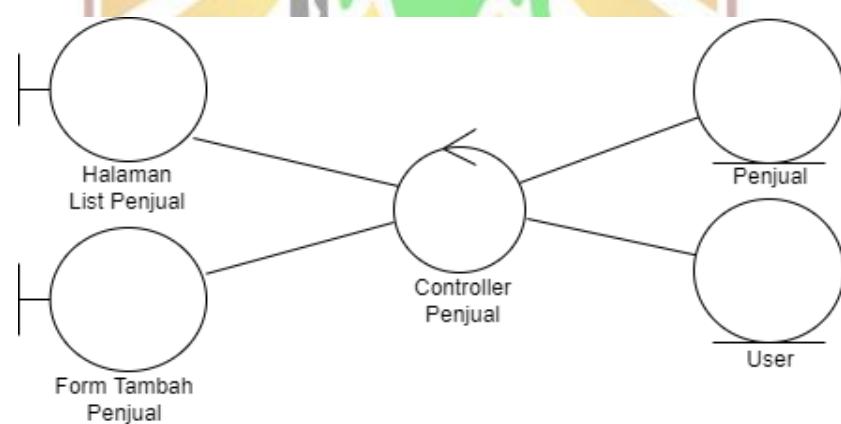
#### 4. Class Analysis Mengedit Data Admin



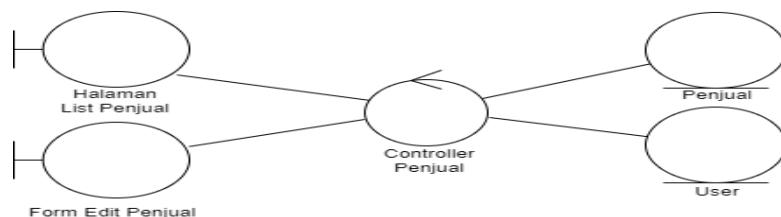
#### 5. Class Analysis Menghapus Data Admin



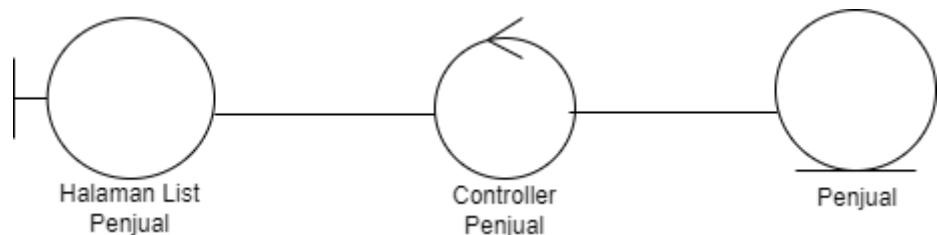
#### 6. Class Analysis Menambah Data Penjual



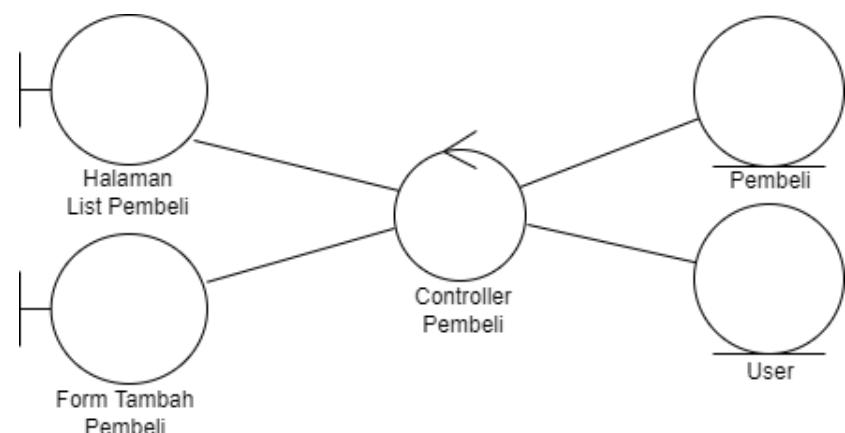
#### 7. Class Analysis Mengedit Data Penjual



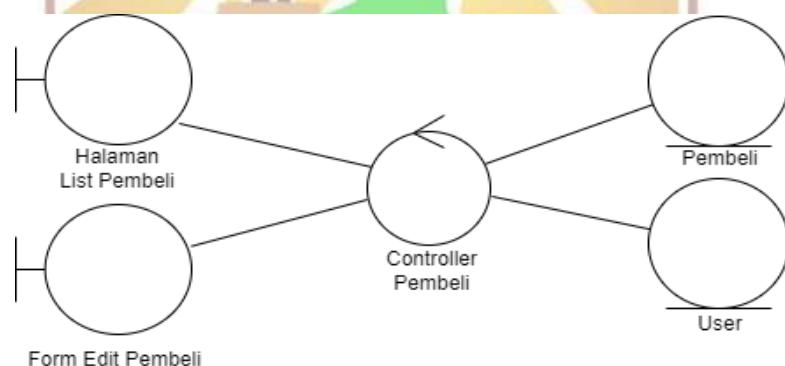
#### **8. Class Analysis Menghapus Data Penjual**



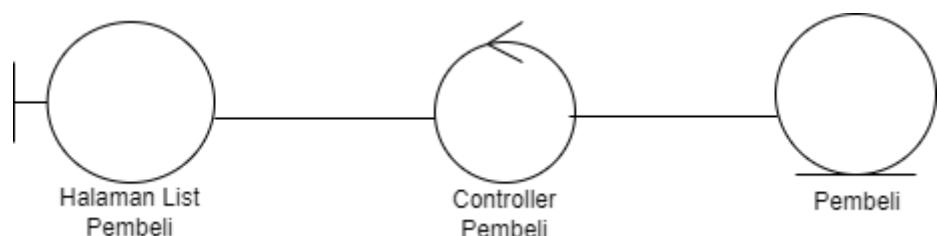
#### **9. Class Analysis Menambah Data Pembeli**



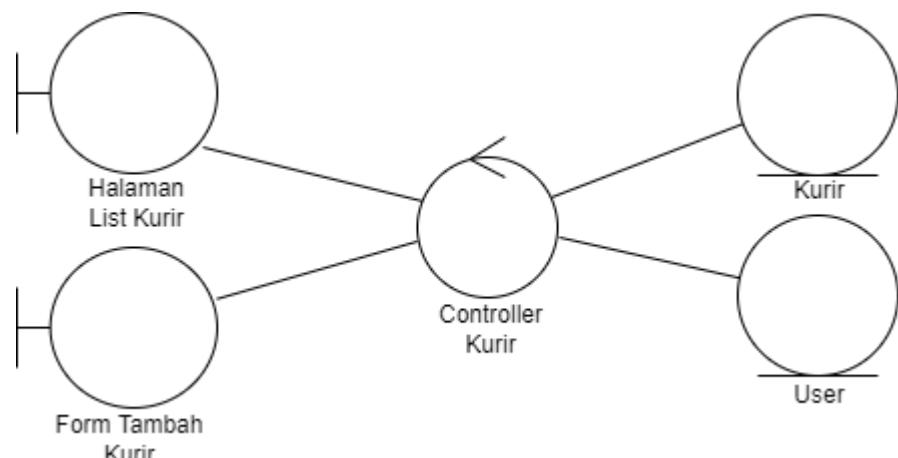
#### **10. Class Analysis Mengedit Data Pembeli**



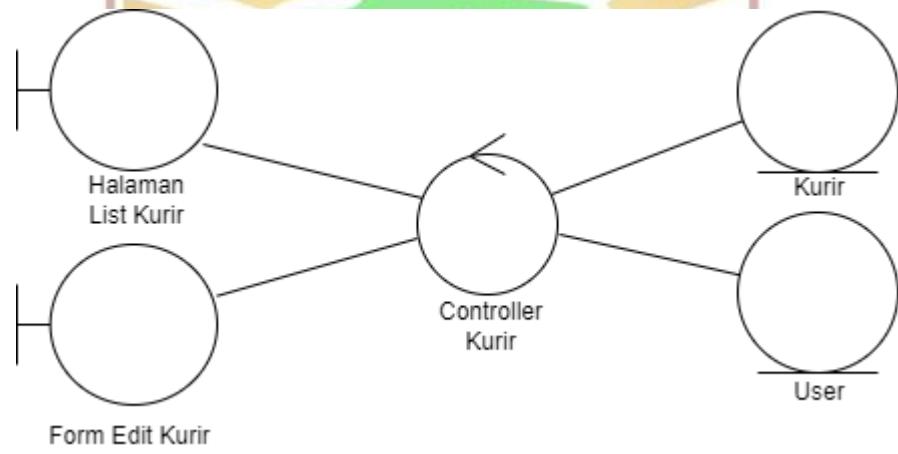
#### **11. Class Analysis Menghapus Data Pembeli**



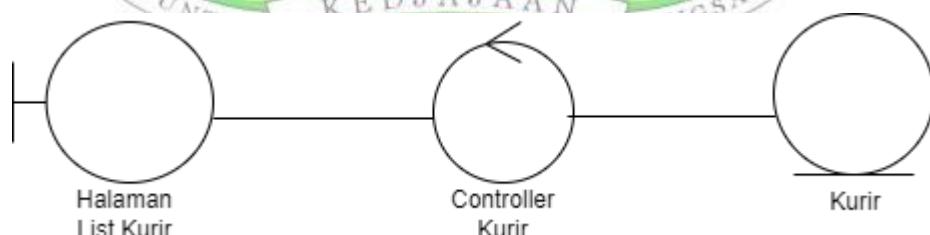
**12. Class Analysis Menambah Data Kurir**



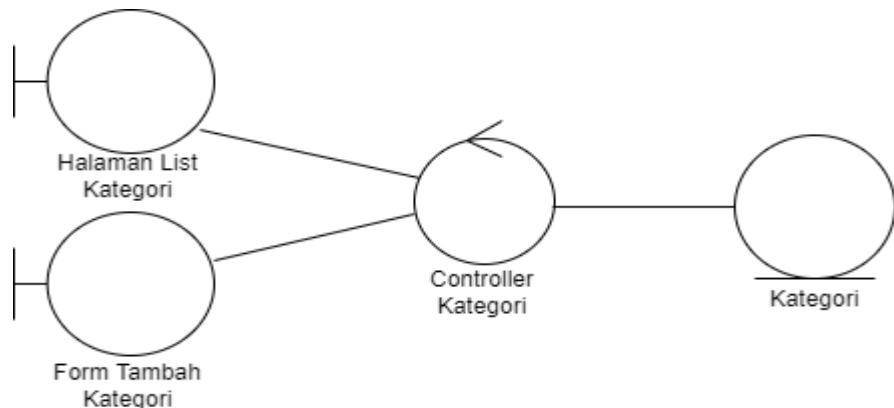
**13. Class Analysis Mengubah Data Kurir**



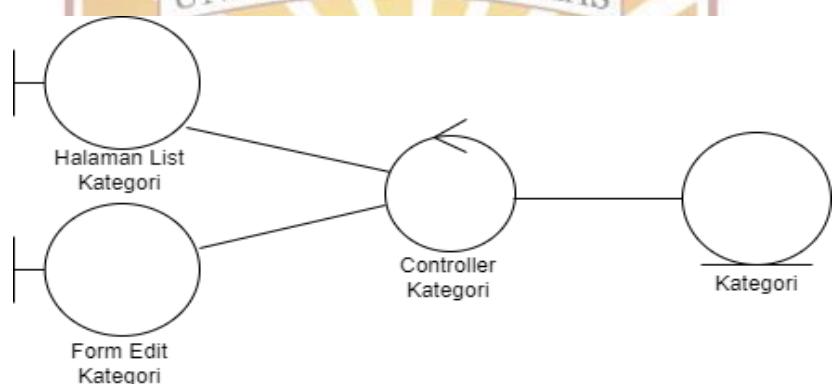
**14. Class Analysis Menghapus Data Kurir**



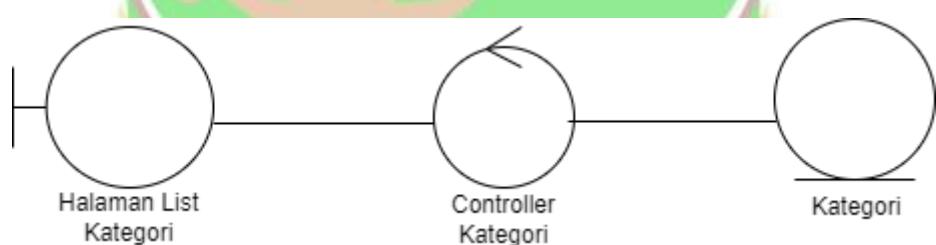
### **15. Class Analysis Menambah Data Kategori Produk**



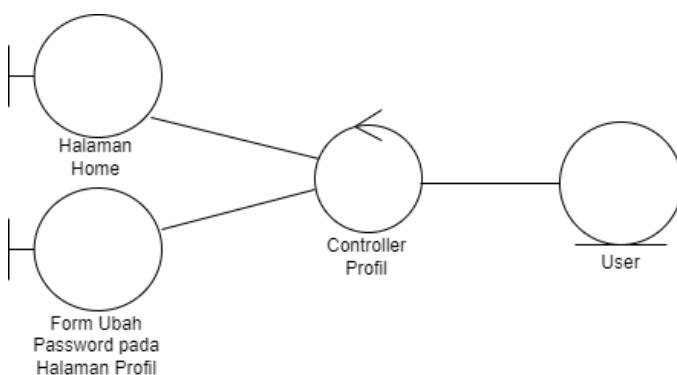
### **16. Class Analysis Mengedit Data Kategori Produk**



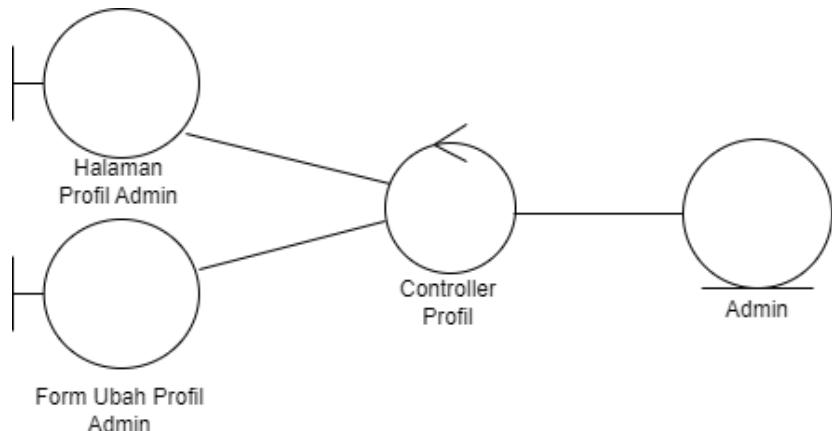
### **17. Class Analysis Menghapus Data Kategori Produk**



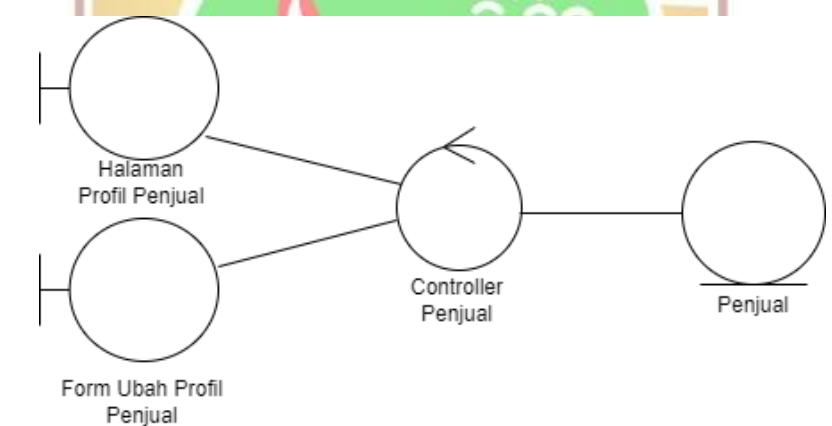
### **18. Class Analysis Mengubah Password Admin**



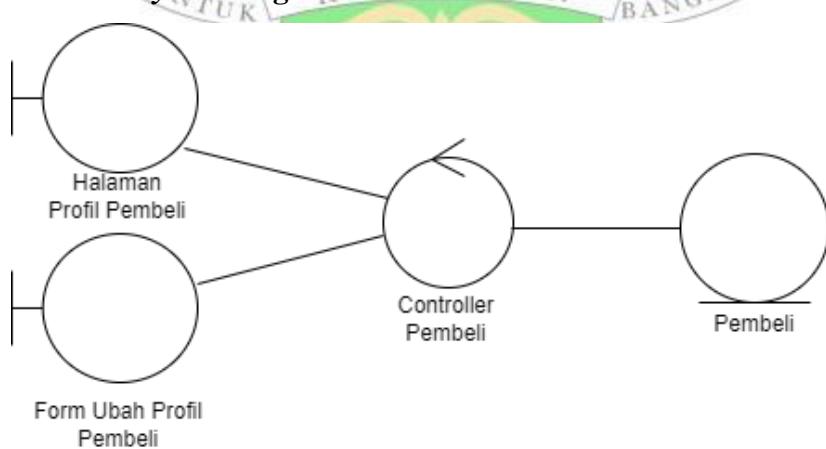
**19. Class Analysis Mengubah Data Profil Admin**



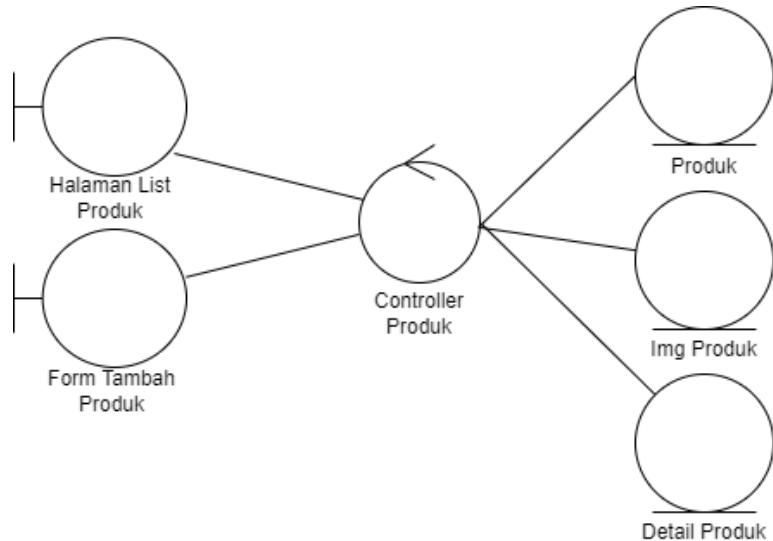
**20. Class Analysis Mengubah Data Profil Penjual**



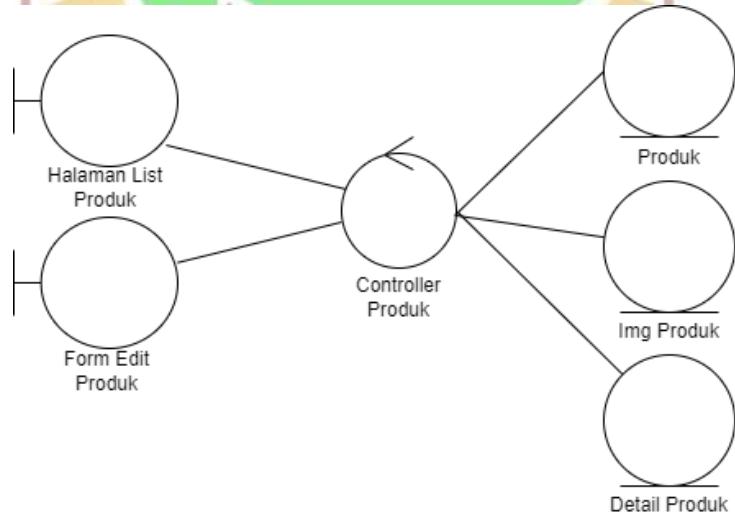
**21. Class Analysis Mengubah Data Profil Pembeli**



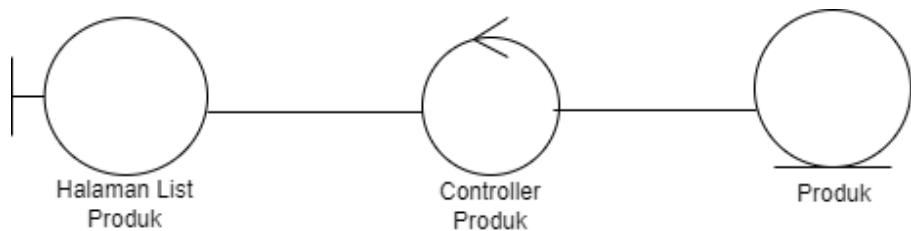
## 22. Class Analysis Menambah Data Produk



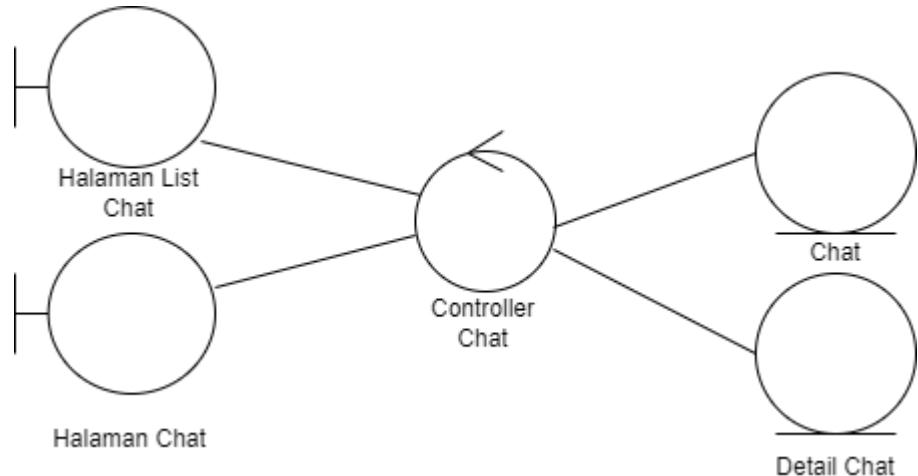
## 23. Class Analysis Mengedit Data Produk



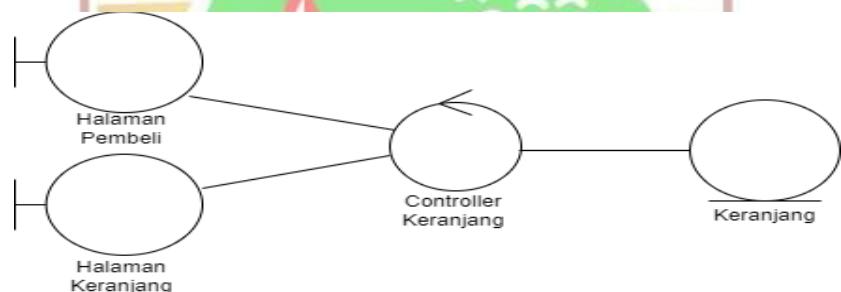
## 24. Class Analysis Menghapus Data Produk



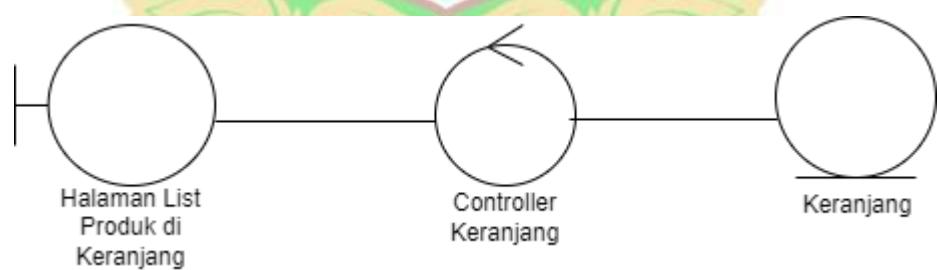
**25. Class Analysis Melakukan Chat**



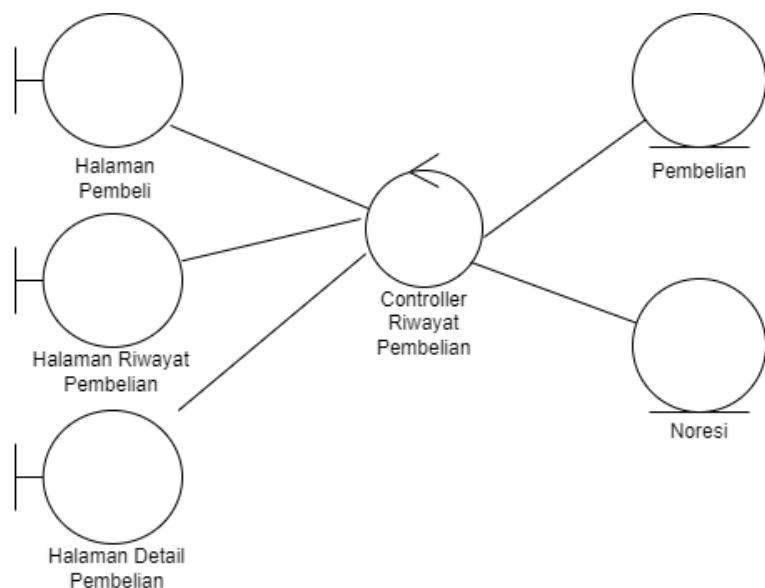
**26. Class Analysis Mengedit Pesanan di Keranjang**



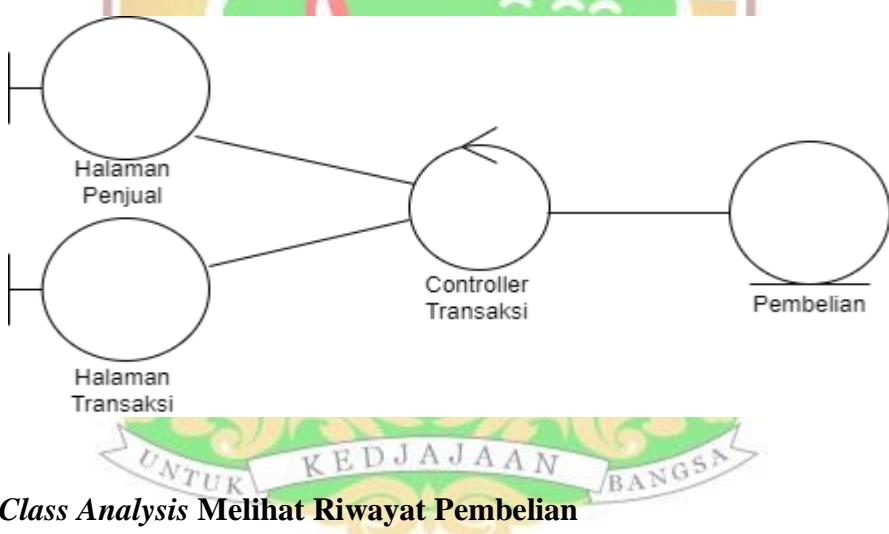
**27. Class Analysis Menghapus Pesanan di Keranjang**



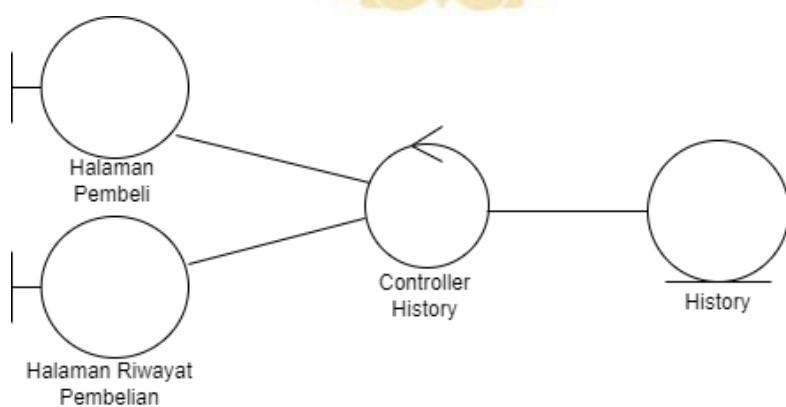
**28. Class Analysis Melihat Noresi Pengiriman Produk**



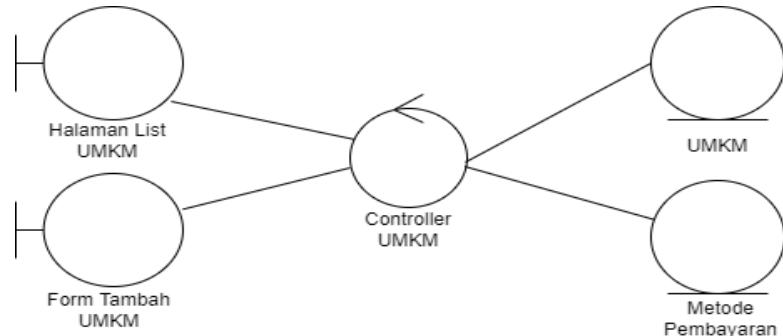
**29. Class Analysis Melihat Laporan Transaksi**



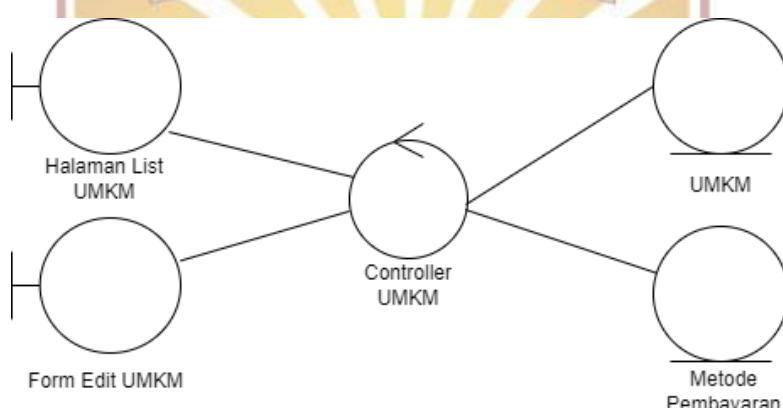
**30. Class Analysis Melihat Riwayat Pembelian**



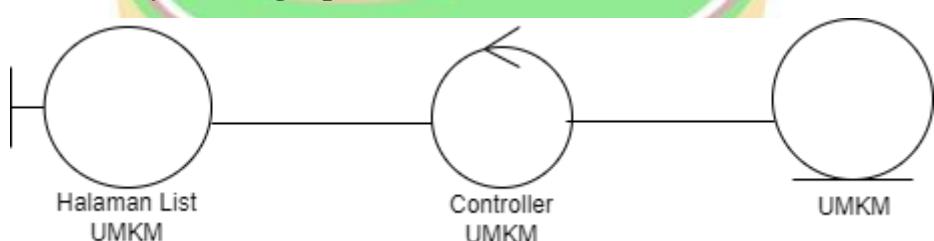
### **31. Class Analysis Menambah Data UMKM**



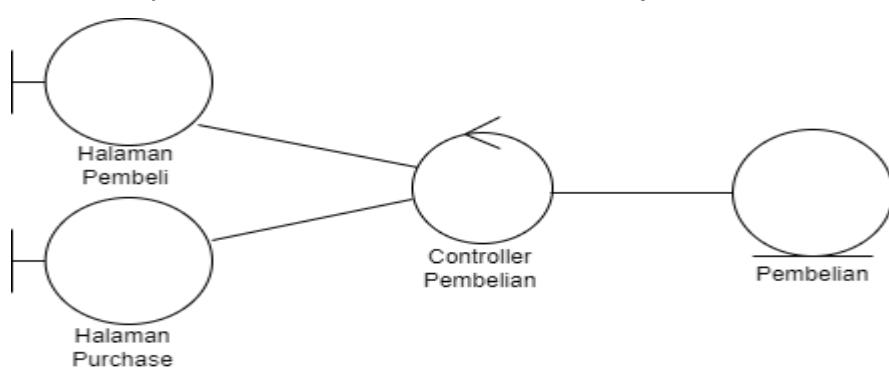
### **32. Class Analysis Mengedit Data UMKM**



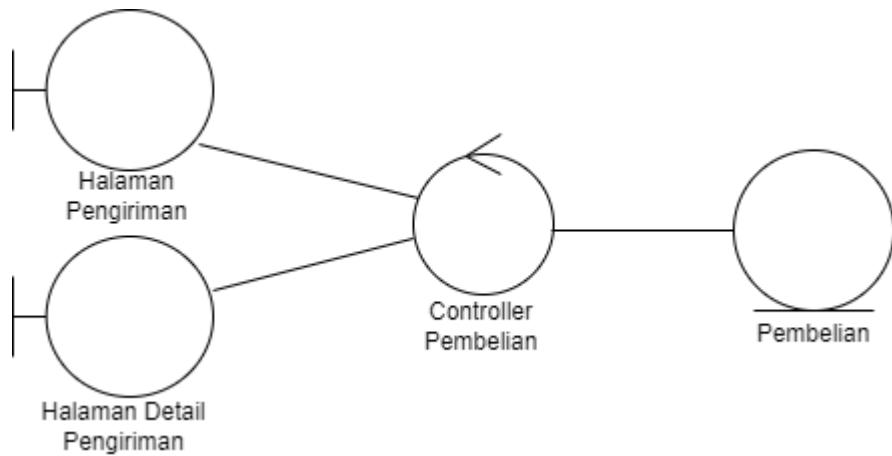
### **33. Class Analysis Menghapus Data UMKM**



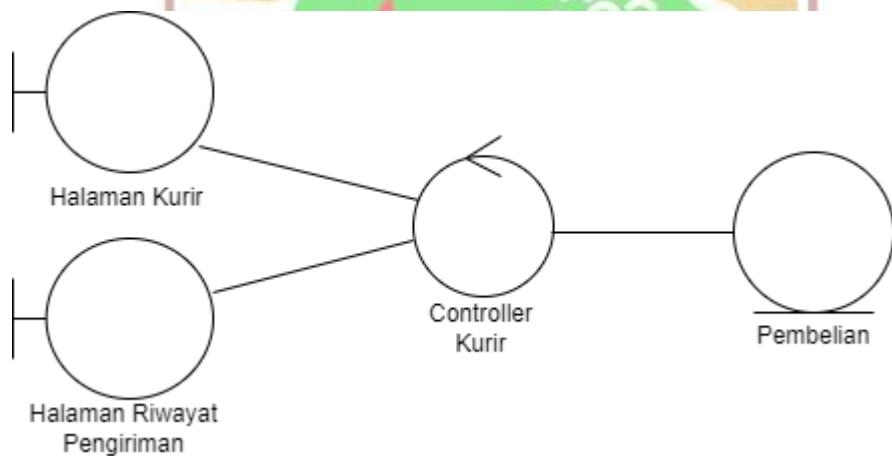
### **34. Class Analysis Melihat Pembelian Belum Dibayar**



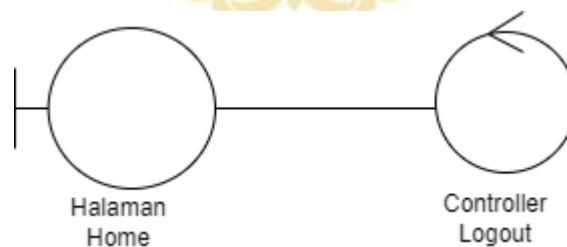
**35. Class Analysis Menginputkan Bukti Pengiriman**

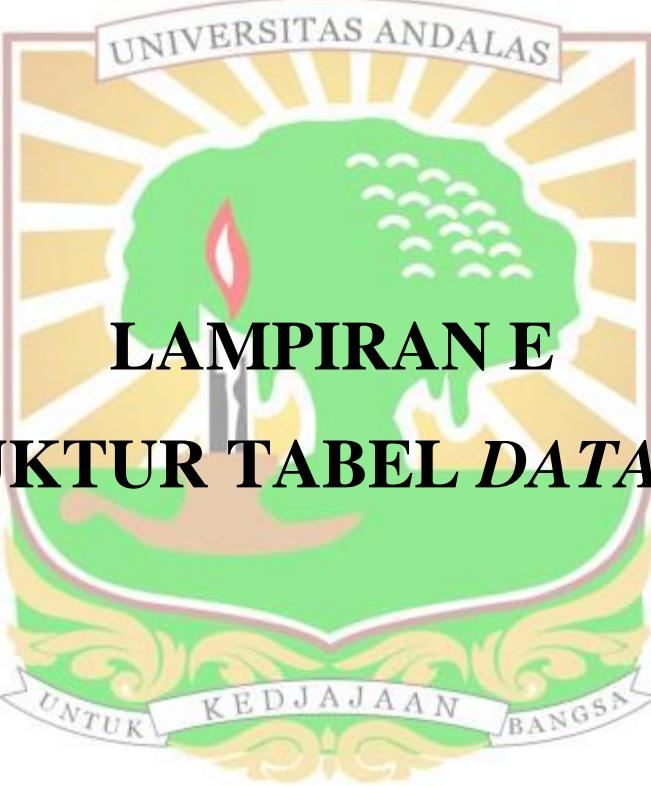


**36. Class Analysis Melihat Riwayat Pengiriman**



**37. Class Analysis Logout**





The logo of Universitas Andalas features a central green shield with a yellow sunburst pattern behind it. Inside the shield is a red candle flame on the left and a white bird's nest with eggs on the right. Below the shield, the text "LAMPIRAN E" is written in bold black capital letters. Underneath the shield, the words "STRUKTUR TABEL DATABASE" are also written in bold black capital letters.

UNIVERSITAS ANDALAS

LAMPIRAN E

STRUKTUR TABEL DATABASE

## 1. Struktur Tabel *User*

Nama Atribut	Tipe Data	Ukuran	Keterangan
User_id	Bigint	20	Primary Key
Email	Varchar	255	
Password	Varchar	255	
Level	Varchar	255	

## 2. Struktur Tabel Kota

Nama Atribut	Tipe Data	Ukuran	Keterangan
Kota_id	Bigint	20	Primary Key
Provinsi_id	Bigint	20	Foreign Key
Nama_kota	Varchar	255	

## 3. Struktur Tabel Provinsi

Nama Atribut	Tipe Data	Ukuran	Keterangan
Provinsi_id	Bigint	20	Primary Key
Nama_provinsi	Varchar	255	

## 4. Struktur Tabel Admin

Nama Atribut	Tipe Data	Ukuran	Keterangan
Admin_id	Bigint	20	Primary Key
User_id	Bigint	20	Foreign Key
Kota_id	Bigint	20	Foreign Key
Provinsi_id	Bigint	20	Foreign Key
Firstname	Varchar	255	
Lastname	Varchar	255	
Handphone	Varchar	13	
Alamat	Varchar	255	
Img	Varchar	255	

## 5. Struktur Tabel Penjual

Nama Atribut	Tipe Data	Ukuran	Keterangan
Penjual_id	Bigint	20	Primary Key
User_id	Bigint	20	Foreign Key
Kota_id	Bigint	20	Foreign Key
Provinsi_id	Bigint	20	Foreign Key
Firstname	Varchar	255	
Lastname	Varchar	255	
Handphone	Varchar	13	
Alamat	Varchar	255	
Img	Varchar	255	
ktp	Varchar	255	

## 6. Struktur Tabel Kurir

Nama Atribut	Tipe Data	Ukuran	Keterangan
Kurir_id	Bigint	20	Primary Key
User_id	Bigint	20	Foreign Key
Firstname	Varchar	255	
Lastname	Varchar	255	
Handphone	Varchar	13	
Alamat	Varchar	255	
Img	Varchar	255	

## 7. Struktur Tabel Kategori

Nama Atribut	Tipe Data	Ukuran	Keterangan
Kategori_id	Bigint	20	Primary Key
Nama_kategori	Varchar	255	

## 8. Struktur Tabel Image Produk

Nama Atribut	Tipe Data	Ukuran	Keterangan
Imageproduk_id	Bigint	20	Primary Key
Produk_id	Bigint	20	Foreign Key
Image	Varchar	255	

## 9. Struktur Tabel Kemasan

Nama Atribut	Tipe Data	Ukuran	Keterangan
Kemasan_id	Bigint	20	Primary Key
Ukuran	Integer	50	

## 10. Struktur Tabel Detail Produk

Nama Atribut	Tipe Data	Ukuran	Keterangan
Detailproduk_id	Bigint	20	Primary Key
Produk_id	Bigint	20	Foreign Key
Kemasan_id	Bigint	20	Foreign Key
Stok	Integer	50	
Harga	Integer	50	
Kode_produk	Varchar	255	
Berat_pengiriman	Varchar	255	

## 11. Struktur Tabel Keranjang

Nama Atribut	Tipe Data	Ukuran	Keterangan
Keranjang_id	Bigint	20	Primary Key
User_id	Bigint	20	Foreign Key
Produk_id	Bigint	20	Foreign Key
Kemasan_id	Bigint	20	Foreign Key
Jumlah	Integer	50	

12. Struktur Tabel Detail Pembelian

Nama Atribut	Tipe Data	Ukuran	Keterangan
Detailpembelian_id	Bigint	20	Primary Key
Pembelian_id	Bigint	20	Foreign Key
Produk_id	Bigint	20	Foreign Key
Kemasan_id	Bigint	20	Foreign Key
Jumlah	Integer	50	

13. Struktur Tabel Metode Pembayaran

Nama Atribut	Tipe Data	Ukuran	Keterangan
metode_id	Bigint	20	Primary Key
Nama_bank	Varchar	255	
Ikon	Varchar	255	

14. Struktur Tabel Detail Metode Pembayaran

Nama Atribut	Tipe Data	Ukuran	Keterangan
Detailmetodepembayaran_id	Bigint	20	Primary Key
Metode_id	Bigint	20	Foreign Key
Umkm_id	Bigint	20	Foreign Key
Norek	Varchar	255	

15. Struktur Tabel Pembayaran

Nama Atribut	Tipe Data	Ukuran	Keterangan
Pembayaran_id	Bigint	20	Primary Key
Pembelian_id	Bigint	20	Foreign Key
Detailmetodepembayaran_id	Bigint	20	Foreign Key
Pembayaran	Varchar	255	

16. Struktur Tabel Ekspedisi

Nama Atribut	Tipe Data	Ukuran	Keterangan
Ekspedisi_id	Bigint	20	Primary Key
Nama_ekspedisi	Varchar	255	

17. Struktur Tabel *History*

Nama Atribut	Tipe Data	Ukuran	Keterangan
History_id	Bigint	20	Primary Key
Pembelian_id	Bigint	20	Foreign Key
Pembeli_id	Bigint	20	Foreign Key

18. Struktur Tabel *Chat*

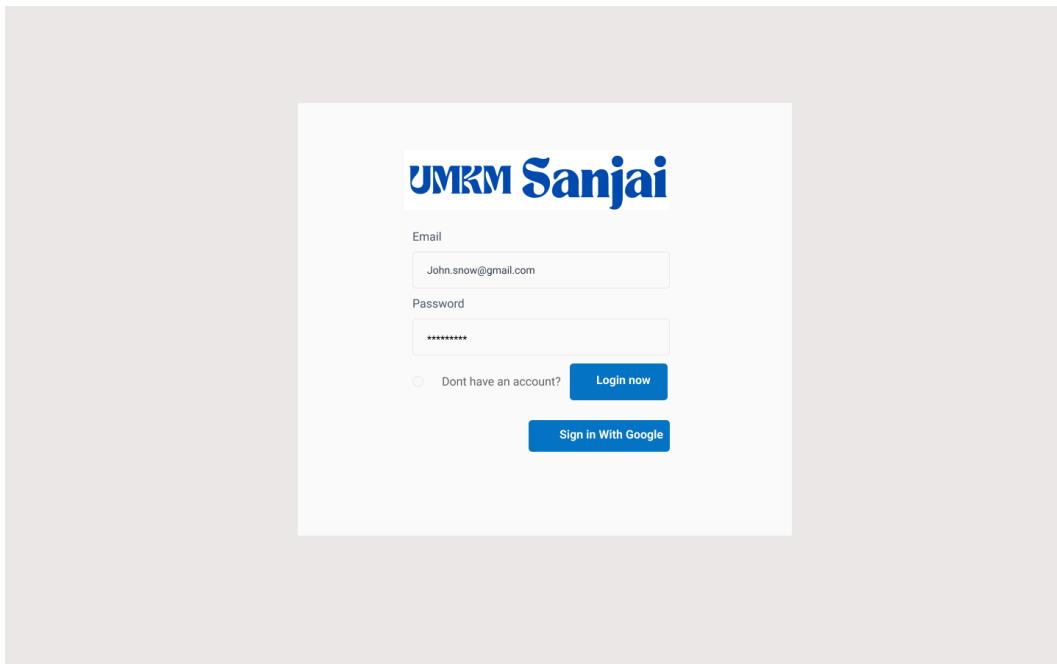
Nama Atribut	Tipe Data	Ukuran	Keterangan
Chat_id	Bigint	20	Primary Key
Pembeli_id	Bigint	20	Foreign Key
Penjual_id	Bigint	20	Foreign Key
Umkm_id	Bigint	20	Foreign Key
Penjual_status	Varchar	255	
Pembeli_status	Varchar	255	

19. Struktur Tabel Detail *Chat*

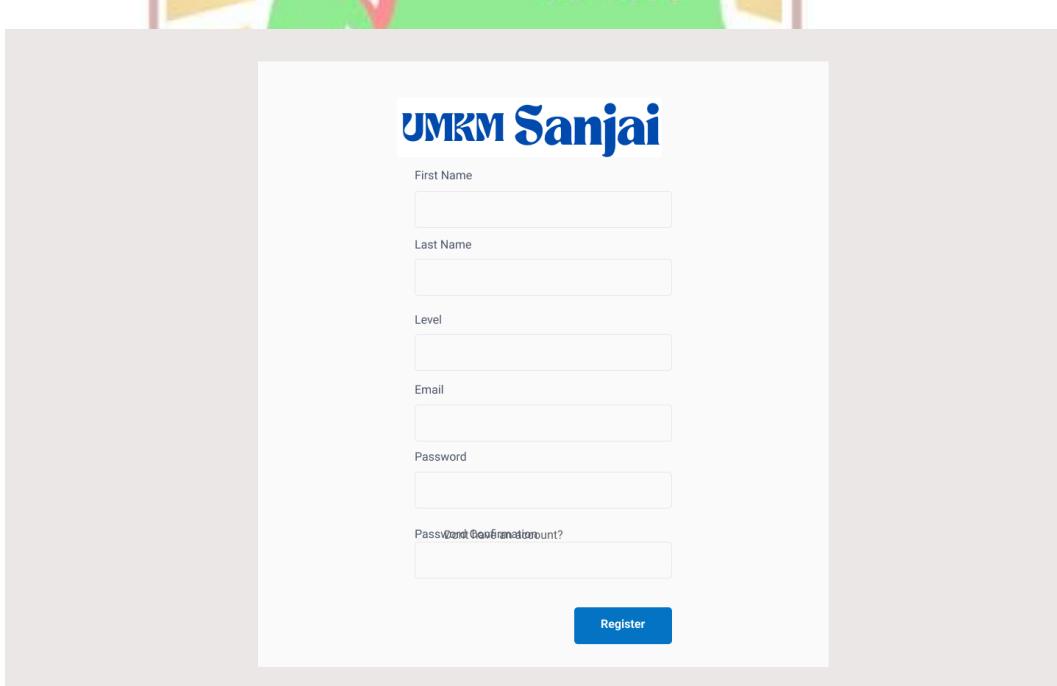
Nama Atribut	Tipe Data	Ukuran	Keterangan
Detailchat_id	Bigint	20	Primary Key
Chat_id	Bigint	20	Foreign Key
Sender	Varchar	255	
Receiver	Varchar	255	
Message	Varchar	255	



## 1. Perancangan Antarmuka Login



## 2. Perancangan Antarmuka Register



### 3. Perancangan Antarmuka Melihat Data User

The screenshot shows the 'List Data User' page. At the top right are icons for a message bubble, a bell, and a user profile. The main area has a header 'List Data User' and a 'Tambah' button. Below is a table with columns: NO, NAMA, NO HP, ALAMAT, and AKSI. Three rows of data are listed:

NO	NAMA	NO HP	ALAMAT	AKSI
1	Distifia Oktari	082285443515	Komplek polda balai baru	<button>Edit</button> <button>Hapus</button>
2	Distifia Oktari	082285443515	Komplek polda balai baru	<button>Edit</button> <button>Hapus</button>
3	Distifia Oktari	082285443515	Komplek polda balai baru	<button>Edit</button> <button>Hapus</button>

### 4. Perancangan Antarmuka Menambah Data User

The screenshot shows the 'Menambah Data User' form. At the top right are icons for a message bubble, a bell, and a user profile. The form fields are as follows:

- First Name: [Input field]
- Last Name: [Input field]
- Email Address: [Input field]
- Phone Number: [Input field]
- Password: [Input field]
- Provinsi: [Input field]
- Kota: [Input field]
- Alamat: [Input field]
- Foto Profil: [Input field]

A 'Save' button is located at the bottom right.

## 5. Perancangan Antarmuka Mengedit Data User

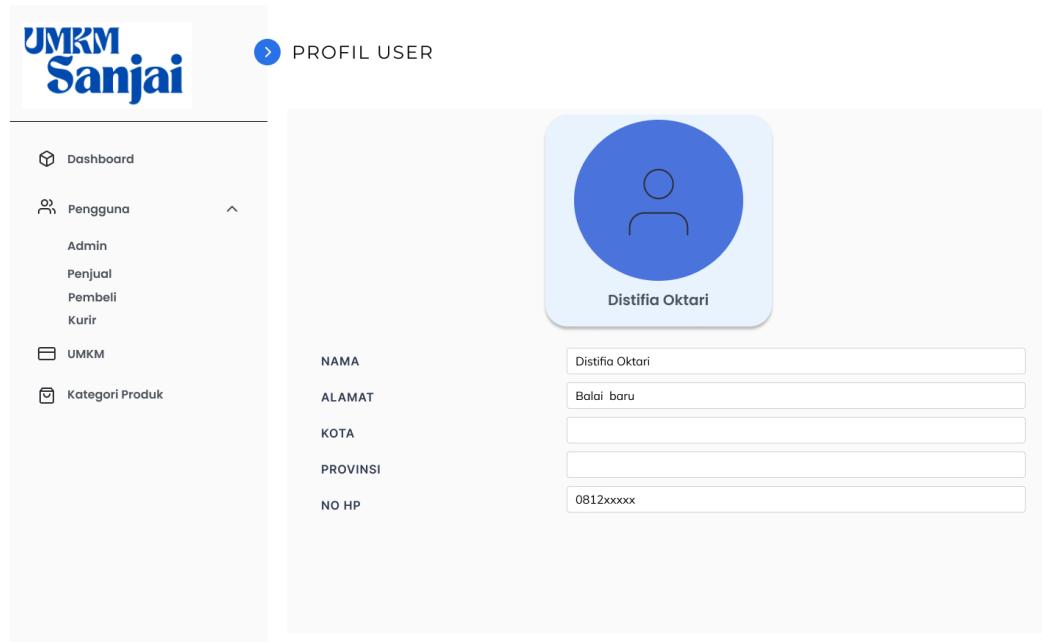
The screenshot shows the 'Edit Data User' form. On the left is a sidebar with navigation links: Dashboard, Pengguna (selected), Admin, Penjual, Pembeli, UMKM, and Kategori Produk. The main area has a title 'Mengedit Data User'. It contains fields for First Name (Distifia), Last Name (Oktari), Email Address (distifaoktari@gmail.com), Phone Number (082285443515), Alamat (Balai baru), Password (\*\*\*\*\*), Provinsi (Sumatra Barat), Kota (Padang), and Foto Profil (disti.png). A 'Save' button is at the bottom.

## 6. Perancangan Antarmuka Menghapus Data User

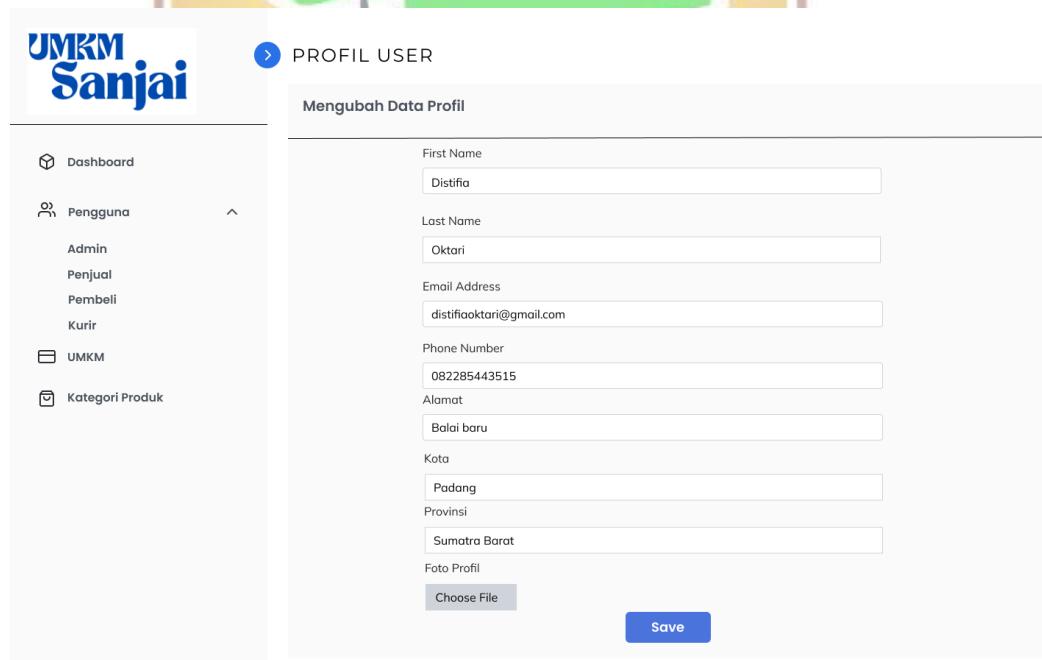
The screenshot shows a confirmation dialog titled 'Apakah Yakin Untuk Menghapus Data?'. It includes 'Tidak' and 'YA' buttons. In the background, the 'List Data User' table is visible, showing three rows of data with 'Edit' and 'Hapus' buttons. The table columns are ALAMAT and AKSI.

ALAMAT	AKSI
rek polda balai baru	<button>Edit</button> <button>Hapus</button>
rek polda balai baru	<button>Edit</button> <button>Hapus</button>
rek polda balai baru	<button>Edit</button> <button>Hapus</button>

## 7. Perancangan Antarmuka Profil



## 8. Perancangan Antarmuka Mengubah Profil User



## 9. Perancangan Antarmuka Mengubah Password User

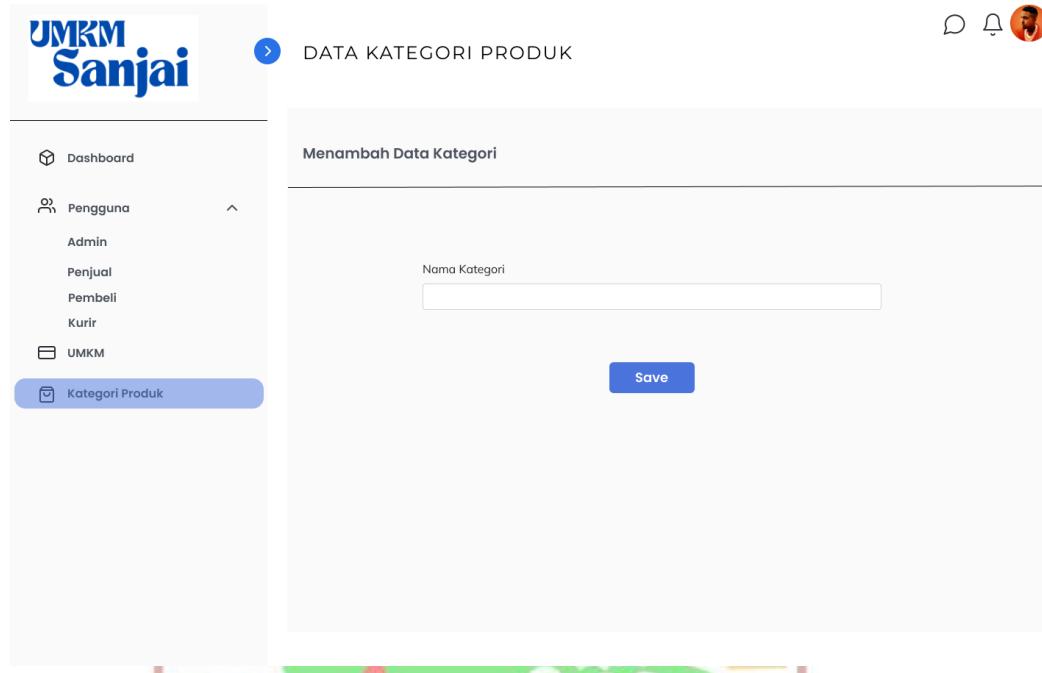
The screenshot shows a user interface titled 'PROFIL USER'. On the left sidebar, under 'Pengguna', 'Kategori Produk' is selected. The main content area is titled 'Mengubah Password' and contains three input fields: 'Current Password' (with placeholder '\*\*\*\*\*'), 'New Password', and 'Confirm Password'. A blue 'Save' button is located at the bottom right of the form.

## 10. Perancangan Antarmuka Melihat Kategori Produk

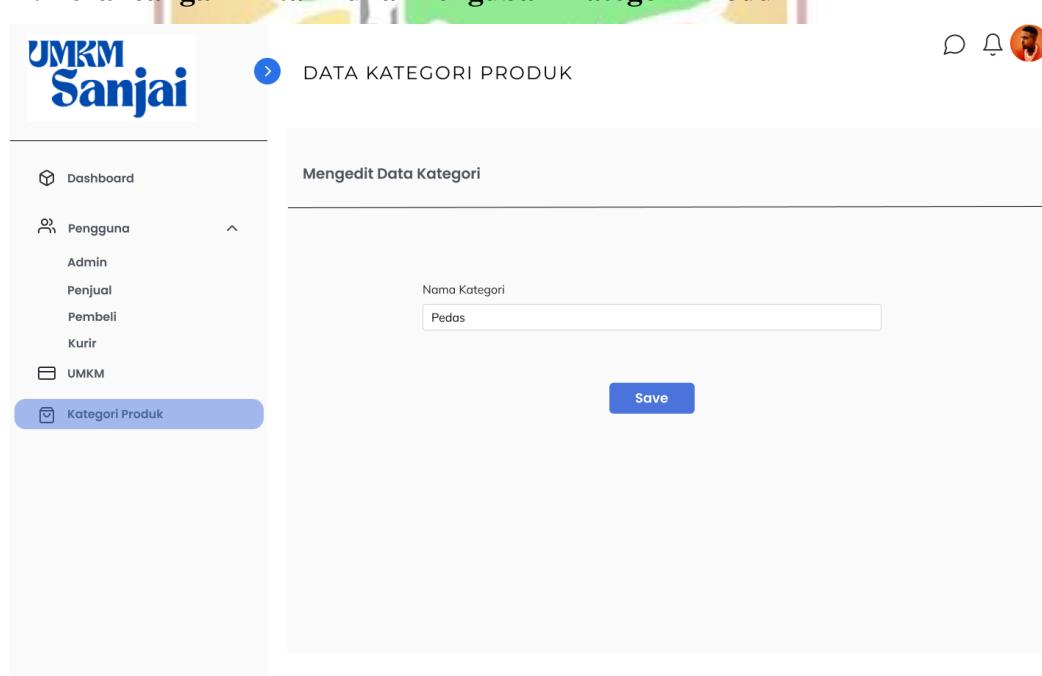
The screenshot shows a user interface titled 'DATA KATEGORI PRODUK'. On the left sidebar, 'Kategori Produk' is selected. The main content area is titled 'List Data Kategori' and displays a table of product categories:

NO	NAMA KATEGORI PRODUK	AKSI
1	Kripik Olahan Singkong	<button>Edit</button> <button>Hapus</button>
2	Kripik Pisang	<button>Edit</button> <button>Hapus</button>
3	Manis	<button>Edit</button> <button>Hapus</button>

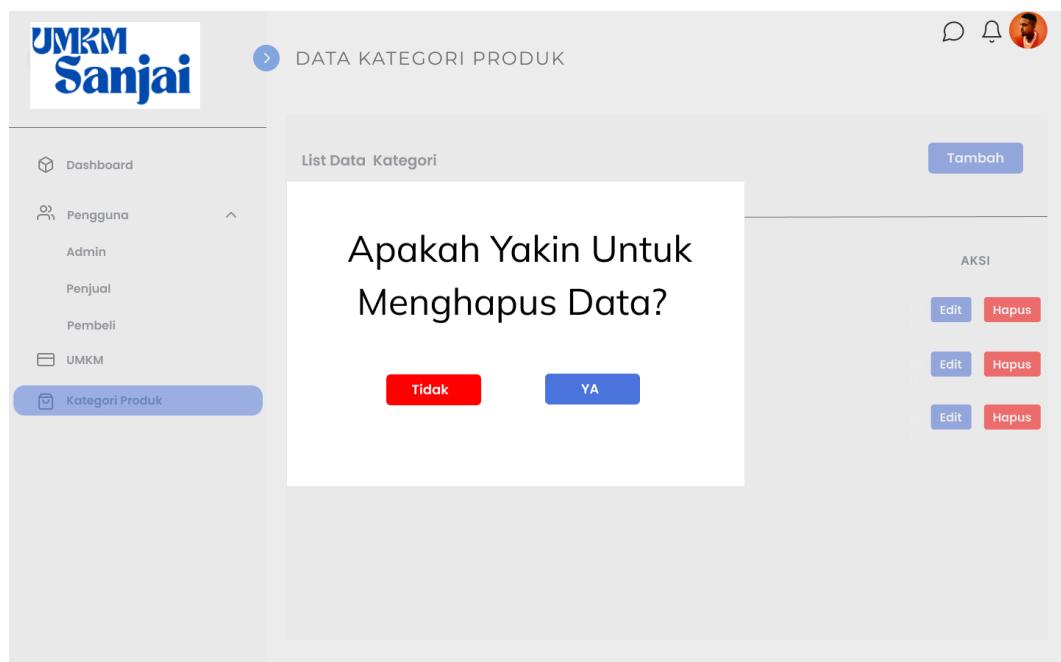
## 11. Perancangan Antarmuka Menambah Kategori Produk



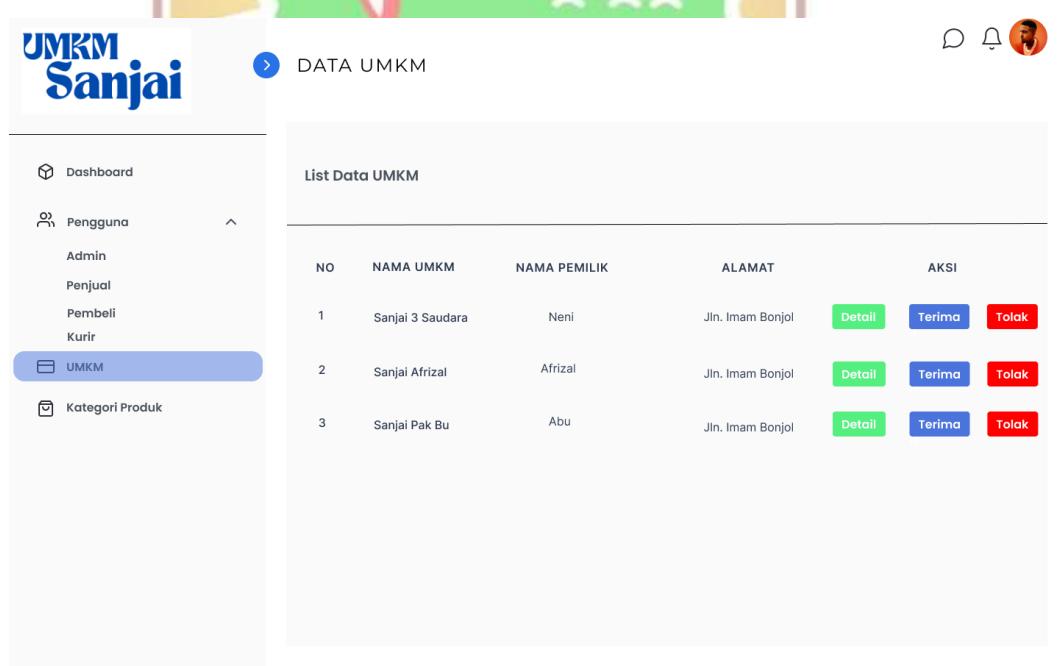
## 12. Perancangan Antarmuka Mengubah Kategori Produk



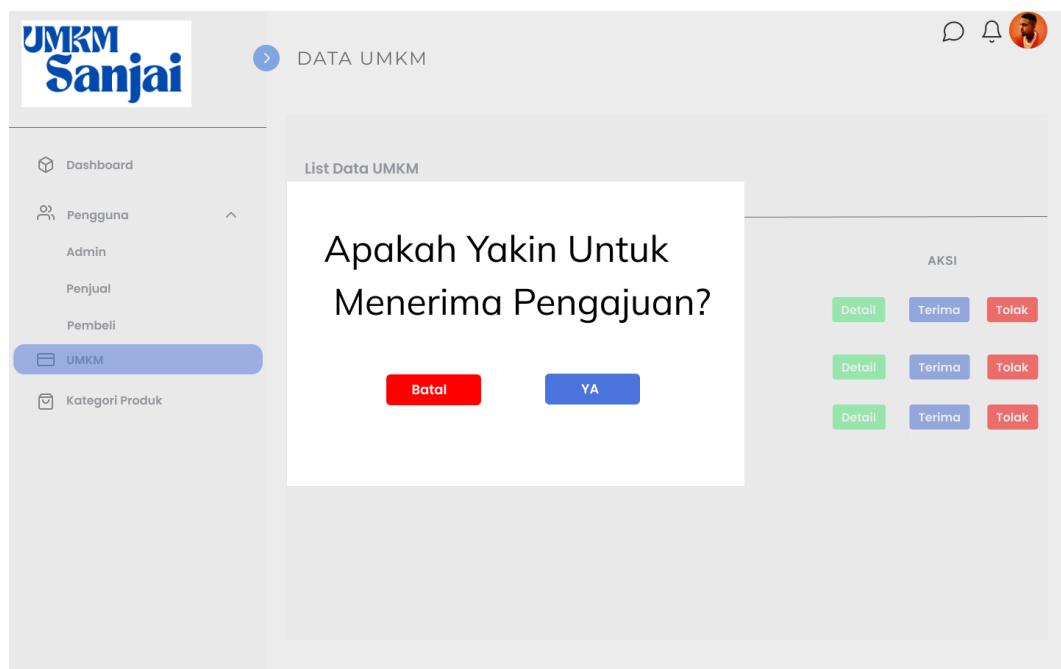
### 13. Perancangan Antarmuka Menghapus Kategori Produk



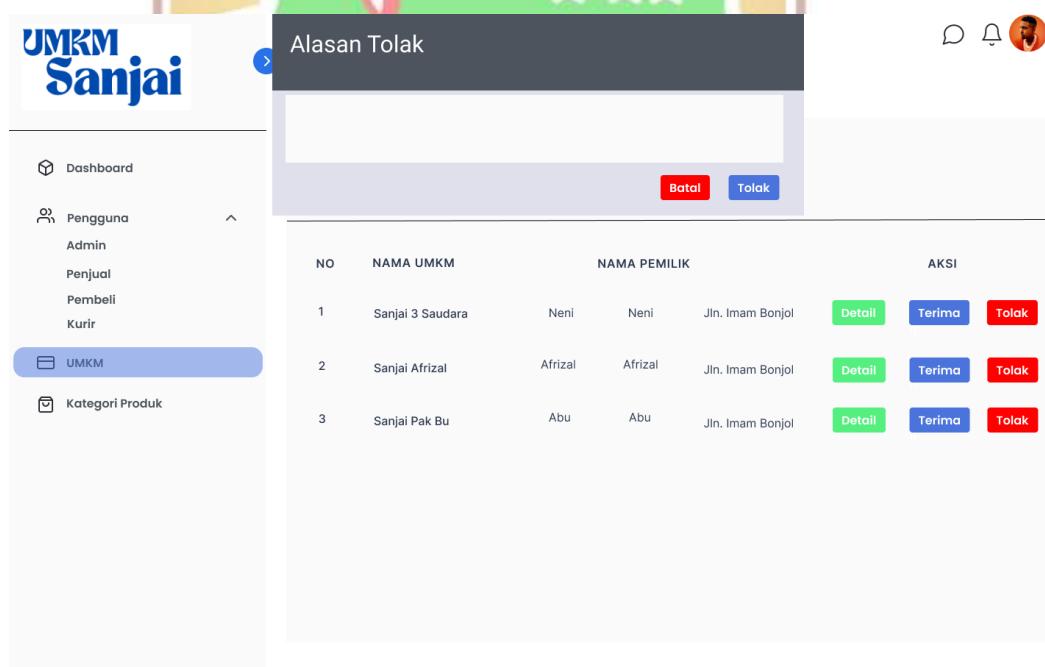
### 14. Perancangan Antarmuka Daftar Pengajuan UMKM



## 15. Perancangan Antarmuka Konfirmasi Terima Pengajuan



## 16. Perancangan Antarmuka Konfirmasi Tolak Pengajuan



## 17. Perancangan Antarmuka Detail Pengajuan

**DETAIL DATA UMKM**

**Sanjai 3 Saudara**

NAMA PENJUAL	Noni	IMB
ALAMAT	jln. Imam Bonjol no 32	File: imbsanjaitigasaudara.pdf
KOTA	Payakumbuh	Map showing the location of Sanjai 3 Saudara in Payakumbuh.
PROVINSI	Sumatra Barat	
NO HP	0812xxxx	
INFORMASI	Merupakan sanjai yang berdiri sejak tahun 2000, menjual berbagai jenis macam sanjai dan makanan kripik khas Sumatra Barat.	

## 18. Perancangan Antarmuka Menambah Data UMKM

**DATA UMKM**

**Menambah Data UMKM**

Nama UMKM	HKI/SIUP
Kota	Choose File
Provinsi	Metode Pembayaran
Alamat	Latitude
No Hp	Longitude
Informasi	Map showing the location of Sanjai 3 Saudara in Payakumbuh.

**Save**

## 19. Perancangan Antarmuka Mengubah Data UMKM

The screenshot shows the 'Edit Data UMKM' page. On the left, there's a sidebar with 'Dashboard', 'UMKM' (which is highlighted in blue), 'Transaksi', 'Laporan', and 'Chat'. The main area has a title 'Mengedit Data UMKM'. It contains fields for 'Nama UMKM' (Sanjai 3 Saudara), 'Kota' (Payakumbuh), 'Provinsi' (Sumatra Barat), 'Alamat' (Jln Imam Bonjol No 32), 'No Hp' (081261368371), and 'Informasi' (Menjual berbagai jenis sanjai khas Sumatra Barat). There's also a file input for 'Imb' with a chosen file named 'imbsanjaaitigasaudara.pdf', a dropdown for 'Metode Pembayaran' (BNI, BCA, Mandiri), and coordinates for 'Latitude' (1.122332) and 'Longitude' (0.122332). A map shows the location in Payakumbuh. A 'Save' button is at the bottom right.

## 20. Perancangan Antarmuka Menghapus Data UMKM

The screenshot shows a confirmation dialog box asking 'Apakah Yakin Untuk Menghapus Data?'. Below it, there are 'Tidak' and 'YA' buttons. In the background, the 'SANJAI OVIGA' dashboard is visible with tabs for 'Lihat Surat Izin', 'Tambah Produk', 'Edit UMKM', and 'Hapus'. The 'Edit UMKM' tab is active. The main area shows a list of products: 'Kirpik Singkong' and 'Olahan Singkong'. For 'Kirpik Singkong', there are 'Detail', 'Edit', and 'Hapus' buttons. The 'Hapus' button is red. A small note says 'yang telah beroperasi sejak tahun 1990...').' At the bottom right, there are 'KATEGORI' and 'AKSI' sections.

## 21. Perancangan Antarmuka Detail UMKM

The screenshot shows the 'UMKM Sanjai' application interface. On the left, a sidebar menu includes 'Dashboard', 'UMKM' (which is highlighted in blue), 'Transaksi', 'Laporan', and 'Chat'. The main content area is titled 'SANJAI OVIGA' and shows a profile picture of a person. Below the profile are details: 'Sanjai Oviga', 'Vina Kurnia', '081257687879', 'Informasi Umum', 'Rumah produksi sanjai yang telah beroperasi sejak tahun 1990...', 'Alamat', 'Jln Aceh Tanjung', 'Metode Pembayaran', and 'Bank Bri (195458438433)'. There is also a table with columns 'NAMA', 'KATEGORI', and 'AKSI' showing one item: 'Kiripik Singkong' under 'Olahan Singkong'. Buttons for 'Detail', 'Edit', and 'Hapus' are at the bottom right.

## 22. Perancangan Antarmuka Menambah Produk

The screenshot shows the 'UMKM Sanjai' application interface. On the left, a sidebar menu includes 'Dashboard', 'UMKM' (highlighted in blue), 'Transaksi', 'Laporan', and 'Chat'. The main content area is titled 'DATA PRODUK' and shows a sub-section titled 'Menambah Data Produk'. It features three tabs: 'Informasi Umum Produk' (selected), 'Rincian Produk', and 'Gambar Produk'. Under 'Informasi Umum Produk', there are fields for 'Nama Produk', 'Kategori Produk', 'No izin P.IRT', and 'Deskripsi'. A 'Next' button is located at the bottom right of the form.

### 23. Perancangan Antarmuka Mengubah Produk

The screenshot shows the 'UMKM Sanjai' application interface. On the left, there's a sidebar with 'Dashboard', 'UMKM' (which is highlighted in blue), 'Transaksi', 'Laporan', and 'Chat'. The main area is titled 'DATA PRODUK' and contains a sub-section 'Menambah Data Produk'. It has three tabs: 'Informasi Umum Produk' (selected), 'Rincian Produk', and 'Gambar Produk'. The 'Informasi Umum Produk' tab contains fields for 'Nama Produk' (Karak kaliang), 'Kategori Produk' (Olahan Singkong), 'No izin PIRT' (12343546544), and 'Deskripsi' (Gurih dan renyah). A 'Next' button is at the bottom right.

### 24. Perancangan Antarmuka Menghapus Produk

The screenshot shows the 'UMKM Sanjai' application interface. On the left, there's a sidebar with 'Dashboard', 'UMKM' (highlighted in blue), 'Transaksi', 'Laporan', and 'Chat'. The main area is titled 'SANJAI OVIGA' and shows a product card for 'Kirpik Singkong' (Category: Olahan Singkong) with ID 87879. Below the card is a confirmation dialog box asking 'Apakah Yakin Untuk Menghapus Data?'. It has 'Tidak' and 'YA' buttons. To the right of the dialog, there's a table with columns 'KATEGORI' and 'AKSI'. It lists 'Kirpik Singkong' under 'Olahan Singkong' with 'Detail', 'Edit', and 'Hapus' buttons.

## 25. Perancangan Antarmuka Detail Produk

The screenshot shows the 'DETAIL DATA PRODUK' (Product Data Detail) screen. At the top right are icons for profile, notifications, and search. On the left is a sidebar with 'Dashboard', 'UMKM' (selected), 'Transaksi', 'Laporan', and 'Chat'. The main area displays a circular image of krispy kreme and the product name 'Kripik Singkong'. Below it are input fields for product details:

NAMA PRODUK	Kripik Singkong	NO PIRT	122.76.454342
KATEGORI	Asin	DESKRIPSI	Kripik yang terbuat dari singkong
UKURAN KEMASAN	1/4 kg		
STOK	30		
HARGA	15.000		
KODE PRODUK	T02		

A green button at the bottom right says 'Tambah Stok' (Add Stock).

## 26. Perancangan Antarmuka Chat Penjual

The screenshot shows the 'CHAT' screen. The sidebar on the left has 'Chat' selected. The main area shows a conversation with a customer named 'Distifia Oktari' from March 9, 2022, at 15:10. The message content is redacted. At the bottom is a text input field with placeholder 'Ketikkan Pesan Anda' (Type your message) and a send arrow icon.

## 27. Perancangan Antarmuka Melihat Laporan Transaksi

LAPORAN TRANSAKSI

NO	PEMBELI	PRODUK	KEMASAN	JUMLAH	HARGA	ONGKIR	TOTAL TRANSAKSI
1	Distii	Karak kaliang	250 gr	3	15.000	10.000	55.000

Cetak



## 28. Perancangan Antarmuka Transaksi

TRANSAKSI

NO	TANGGAL	UMKM	TOTAL PEMBELIAN	STATUS	AKSI
1	1/03/23	Sanjai Oviga	55.000	Sedang Disiapkan	<button>Detail</button> <button>Terima</button> <button>Tolak</button>
2	24/05/23	Sanjai Oviga	30.000	Dikirim	<button>Detail</button> <button>Terima</button> <button>Tolak</button>

## 29. Perancangan Antarmuka Detail Transaksi

**TRANSAKSI**

**Detail Transaksi Pemesanan**

DISTIFIA 04/03/23 BANK MANDIRI MENUNGGU KONFRIMASI

Kode Pemesanan : 121325465767

Detail pemesanan :

NO	NAMA	KEMASAN	JUMLAH	HARGA	SUBTOTAL
1	Karak Kaliang	250 gr	2	10.000	20.000
2	Kripik Pisang	250 gr	2	10.000	20.000
				BIAYA PENGIRIMAN	20.000
				TOTAL	60.000

## 30. Perancangan Antarmuka Input Noresi

**Input Noresi**

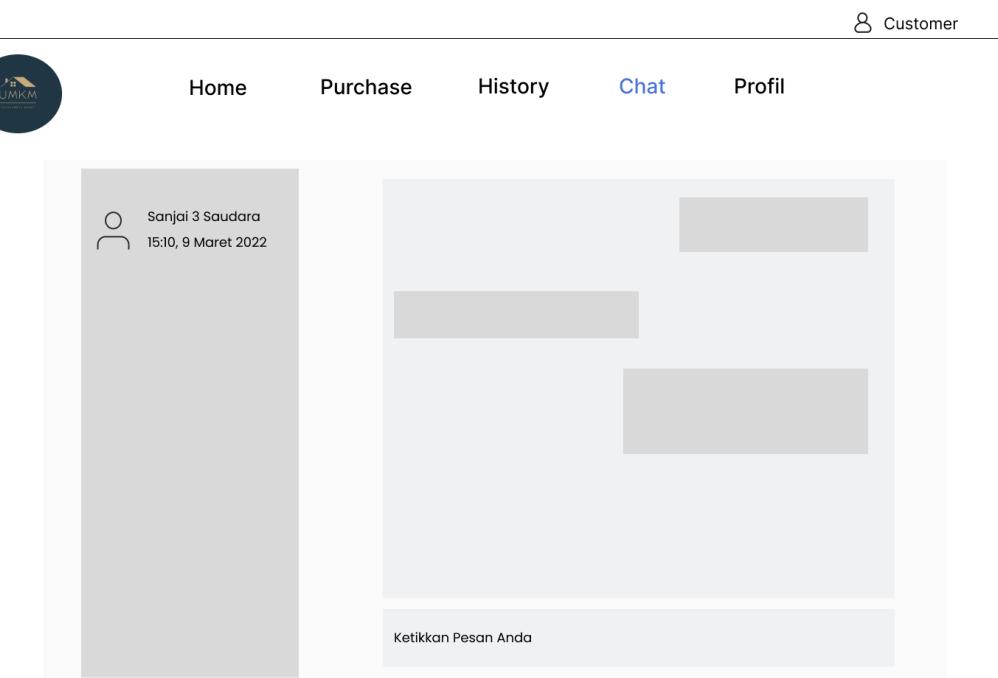
Batal Input

Kode Pemesanan : 121325465767

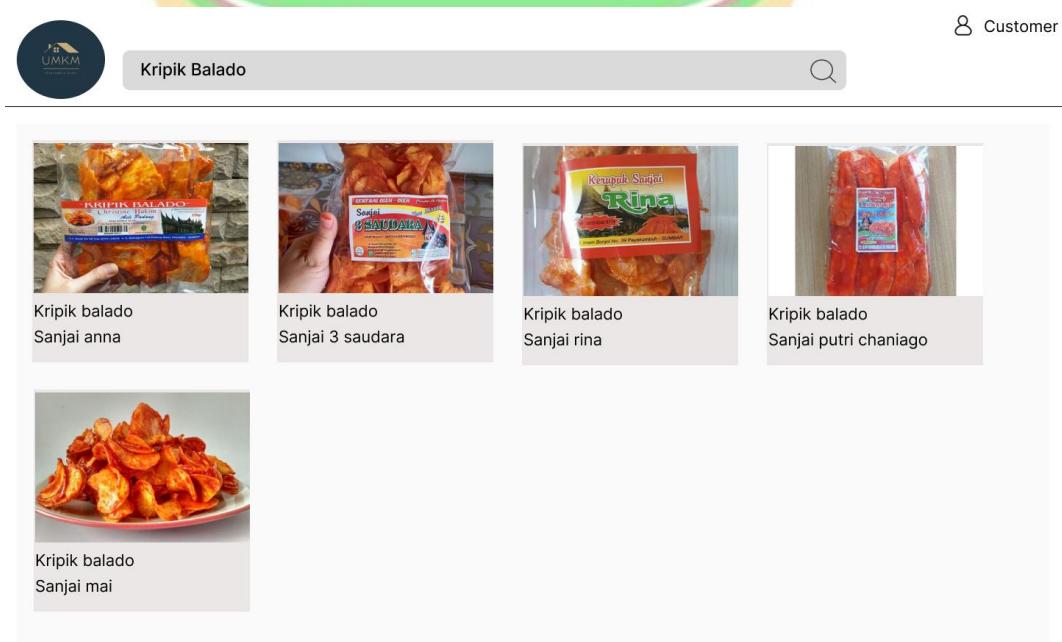
Detail pemesanan :

NO	NAMA	KEMASAN	JUMLAH	HARGA	SUBTOTAL
1	Karak Kaliang	250 gr	2	10.000	20.000
2	Kripik Pisang	250 gr	2	10.000	20.000
				BIAYA PENGIRIMAN	20.000
				TOTAL	60.000

### 31. Perancangan Antarmuka *Chat* Pembeli



### 32. Perancangan Antarmuka Melakukan Pencarian



### 33. Perancangan Antarmuka Detail Produk

The screenshot shows a mobile application interface for a product detail page. At the top, there is a navigation bar with icons for Home, Purchase, History, Chat, and Profil. A user icon labeled 'Customer' is also present. On the left, a sidebar titled 'All Categories' lists product types: Kripik Olahan Singkong, Kripik Olahan Talas, Kripik Olahan Kentang, Rakik/Peyek, and Snack Pisang. In the center, there is a large image of a pack of 'Kripik balado' (balado chips) with a red label featuring a traditional building illustration. To the right of the image, the product name 'Kripik balado' is displayed, along with 'Sanjai putri chaniago' and a price range 'Rp15.000,- ~ Rp100.000,-'. Below this, a section titled 'Deskripsi Produk' contains the text 'Informasi : Kripik yang terbuat dari ubi, diolah dengan menggunakan teknik khusus, halal dan tanpa pengawet buatan.' There is also a 'Pilih Ukuran' (Select Size) section with options for 1/4 Kg and 1/2 Kg, and a 'Jumlah' (Quantity) input field set to '1'. A blue 'Add To Cart' button is located at the bottom right.

### 34. Perancangan Antarmuka Checkout

The screenshot shows a mobile application interface for a checkout page. At the top, there is a navigation bar with icons for Home, Purchase, History, Chat, and Profil. A user icon labeled 'Customer' is also present. On the left, a form titled 'Details' asks for First Name (Distifia), Last Name (Oktari), Alamat (Balai baru), Provinsi (Sumatra Barat), Kota (Padang), and Ekspedisi (dropdown menu 'Pilih Ekspedisi'). On the right, a summary of the 'Your Order' is shown:

Products	Total
Jagung	Rp15.000,-
Kipang	Rp15.000,-
<b>Subtotal</b>	<b>Rp30.000,-</b>

Below this, a dropdown menu 'Pilih Layanan' is open, showing 'Pilih Layanan' and 'Metode Pembayaran' with options for Mandiri (123243546) and BRI (34365776756). The total amount is listed as 'Biaya Pengiriman Rp15.000,-' and 'Total Rp45.000,-'. A blue 'Place Order' button is located at the bottom right.

## 35. Perancangan Antarmuka Ringkasan Pembelian

Customer



Home Purchase History Chat Profil

Terima kasih telah melakukan pembelian.

Ringkasan Order

Product	Price	Quantity	Subtotal
 Karak Kaliang	Rp10.000,-	4	Rp40.000,-
			Biaya Pengiriman Rp20.000,-
			Total Harga Rp60.000,-

Silahkan melakukan transfer ke rekening berikut :

mandiri (1245657687)

Bukti transfer :

Choose File

Submit Payment



## 36. Perancangan Antarmuka Riwayat Pembelian

Customer



Home Purchase History Chat Profil

No	Nama	Total Harga	Status	Aksi
1	Sanjai Oviga	Rp40.000,-	Menunggu Konfirmasi	<a href="#">Detail Transaksi</a>

### 37. Perancangan Antarmuka Detail Pembelian

The screenshot shows a mobile application interface for a small business named "Sanjai". At the top, there is a circular profile picture with the text "UMKM Sanjai". To the right of the profile picture are navigation links: "Home", "Purchase", "History" (which is highlighted in blue), "Chat", "Profil", and a shopping cart icon. Below the navigation bar, a message says "Terima kasih telah melakukan pembelian." (Thank you for your purchase). Underneath this, it says "Ringkasan Order" (Order Summary) and lists the following details:  
Umkm : Sanjai Oviga  
Tanggal : 16/04/23  
Status : Menunggu Konfirmasi  
Noresi : Belum tersedia

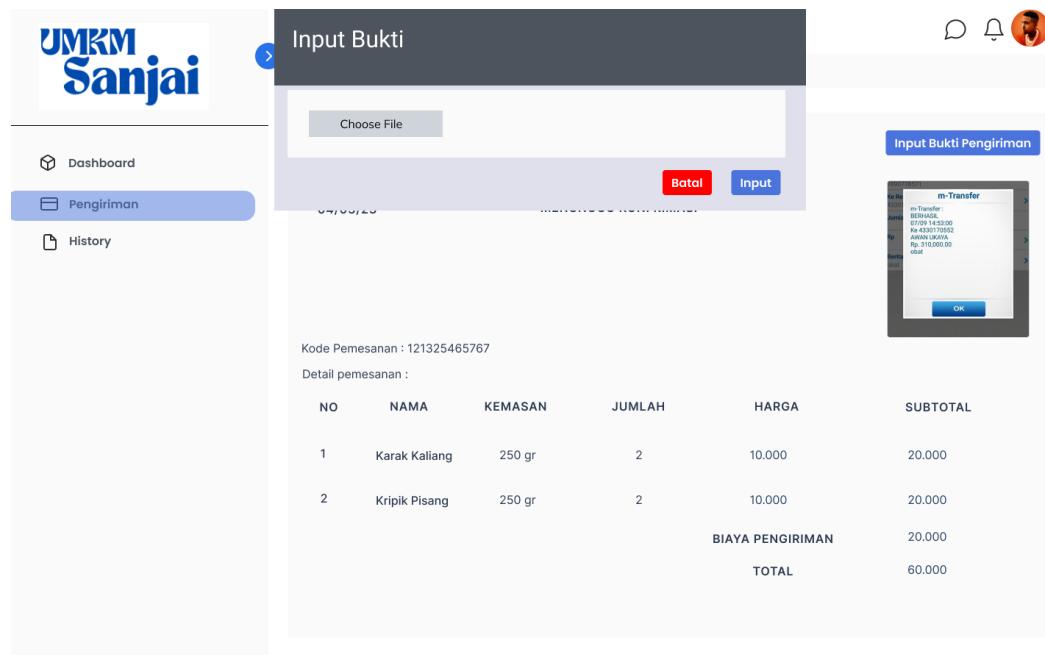
Product	Price	Quantity	Subtotal
 Karak Kaliang	Rp10.000,-	4	Rp40.000,-
			Biaya Pengiriman
			Rp20.000,-
			Total Harga
			Rp60.000,-

### 38. Perancangan Antarmuka Pengiriman

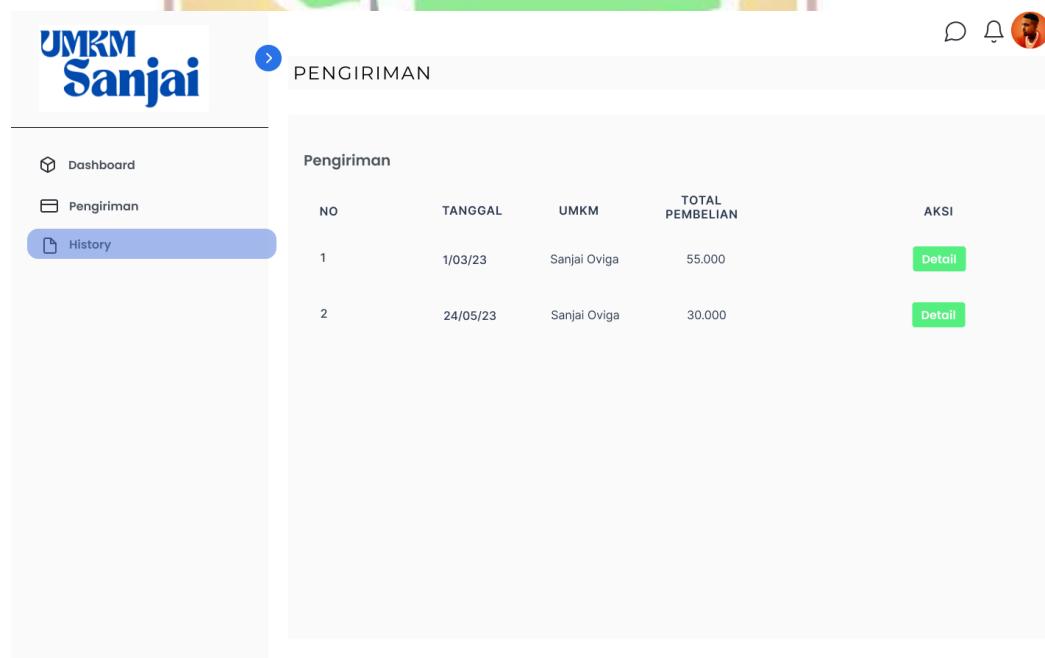
The screenshot shows a mobile application interface for a small business named "Sanjai". At the top, there is a logo for "UMKM Sanjai" and a navigation bar with icons for "Dashboard", "Pengiriman" (Delivery, highlighted in blue), and "History". To the right of the navigation bar are icons for a message bubble, a bell, and a user profile. The main content area is titled "PENGIRIMAN" and has a sub-section titled "Pengiriman". It displays a table of delivery history:

NO	TANGGAL	UMKM	TOTAL PEMBELIAN	AKSI
1	1/03/23	Sanjai Oviga	55.000	<button>Detail</button>
2	24/05/23	Sanjai Oviga	30.000	<button>Detail</button>

### 39. Perancangan Antarmuka Menambahkan Bukti Pengiriman



### 40. Perancangan Antarmuka History Pengiriman



#### 41. Perancangan Antarmuka Detail History

**HISTORY**

**Detail Transaksi Pemesanan**

DISTIFIA  
04/03/23  
BANK MANDIRI  
SELESAI

Kode Pemesanan : 121325465767  
Norec : 132325465768

Detail pemesanan :

NO	NAMA	KEMASAN	JUMLAH	HARGA	SUBTOTAL	
1	Karak Kaliang	250 gr	2	10.000	20.000	
2	Kripik Pisang	250 gr	2	10.000	20.000	
					BIAYA PENGIRIMAN	20.000
					TOTAL	60.000

Bukti pengiriman





**LAMPIRAN G**

**KODE PROGRAM APLIKASI**

## 1.Bagian Controller

### 1. LokasiController

```
1 <?php namespace App\Http\Controllers\api;
2
3 use Illuminate\Http\Request;
4 use App\Http\Controllers\Controller;
5 use App\Models\Umkm;
6 use App\Models\Provinsi;
7 use App\Models\Kota;
8
9 class LokasiController extends Controller
10 {
11     public function getCities(Request $request, $id)
12     {
13         $cities = Kota::where('province_id', $id)->pluck('nama_kota', 'city_id');
14         $selectedCityId = $request->input('kota_id');
15
16         // Mengambil nama kota berdasarkan selectedCityId
17         $kota = Kota::where('city_id', $selectedCityId)->first();
18         $kotaName = $kota ? $kota->nama_kota : null;
19
20         $response = [
21             'kotas' => $cities,
22             'kotaName' => $kotaName
23         ];
24
25         return json_encode($response);
26     }
27 }
28 }
```

### 2. MessageController

```
1 <?php namespace App\Http\Controllers\api;
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\Controller;
5 use Illuminate\Http\Request;
6 use App\Models\Chat;
7 use App\Models\ChatDetail;
8 use App\Models\Penjual;
9 use App\Models\Pembeli;
10 use Illuminate\Support\Facades\Auth;
11 use Carbon\Carbon;
12 use App\Helper;
13
14 class MessageController extends Controller {
15     public function sendMessage(Request $request) {
16         $chat=Chat::where('pembeli_id', $request->pembeli)
17             ->where('penjual_id', $request->penjual)
18             ->where('umkm_id', $request->id)->first();
19
20         $penjual = Penjual::find($request->penjual);
21         $pembeli = Pembeli::find($request->pembeli);
22         if($chat) {
23             $chatDetail=new ChatDetail;
24             $chatDetail->chat_id=$chat->id;
25             if($request->status) {
26                 $chatDetail->sender=$pembeli->User->id;
27                 $chatDetail->receiver=$penjual->User->id;
28             }else {
29                 $chatDetail->sender=$penjual->User->id;
30                 $chatDetail->receiver=$pembeli->User->id;
31             }
32             $chatDetail->message=$request->message;
33             $chatDetail->save();
34             $today=Carbon::now();
35             $chat->updated_at=$today->format('Y-m-d H:i:s');
36             $chat->save();
37         }
38         else {
39             $chat=new Chat;
40             $chat->penjual_id=$request->penjual;
41             $chat->pembeli_id=$request->pembeli;
42             $chat->umkm_id=$request->id;
43             $chat->status=$request->status;
44             $chat->message=$request->message;
45             $chat->save();
46         }
47     }
48 }
```

```

41     $chat->pembeli_id=$request->pembeli;
42     $chat->umkm_id=$request->id;
43     $chat->save();
44     $chatDetail=new ChatDetail;
45     $chatDetail->chat_id=$chat->id;
46     if($request->status) {
47         $chatDetail->sender=$pembeli->User->id;
48         $chatDetail->receiver=$penjual->User->id;
49     } else {
50         $chatDetail->sender=$penjual->User->id;
51         $chatDetail->receiver=$pembeli->User->id;
52     }
53     $chatDetail->message=$request->message;
54     $chatDetail->save();
55 }
56 $data=[ "name"=>$chatDetail->Pembeli->first_name." ".$chatDetail->Pembeli->last_name,
57 "avatar"=>$chatDetail->Pembeli->img,
58 "message"=>$request->message,
59 "created_at"=>$chatDetail->created_at->format('H:i A | d M Y'),
60 "created_atV2"=>$chatDetail->created_at->format('D M d Y - H:i:s'));
61 return response()->json($data);
62 }
63 }

```

### 3. MetodePembayaranController

```

1 <?php
2
3 namespace App\Http\Controllers\api;
4
5 use Illuminate\Http\Request;
6 use App\Http\Controllers\Controller;
7 use App\Models\Metode_Pembayaran;
8 use App\Models\Detail_Metode_Pembayaran;
9
10 class MetodePembayaranController extends Controller
11 {
12     public function addPaymentMethod(Request $request){
13         $data = Metode_Pembayaran::whereIn('id', $request->id)->get();
14         if($request->umkm_id){
15             foreach($datas as $data){
16                 $detail = Detail_Metode_Pembayaran::where('metodepemb_id', $data->id)
17                                     ->where('umkm_id', $request->umkm_id)
18                                     ->first();
19                 if($detail){
20                     $data->norek = $detail->norek;
21                 } else{
22                     $data->norek = 'null';
23                 }
24             }
25         }
26         return response()->json($data);
27     }
28 }

```

### 4. AdminController

```

1 <?php
2 namespace App\Http\Controllers;
3
4 use Illuminate\Http\Request;
5 use App\Models\Admin;
6 use App\Models\Ekspedisi;
7 use App\Models\Provinsi;
8 use App\Models\Kota;
9 use App\Models\User;
10 use Illuminate\Support\Facades\Hash;
11 use Illuminate\Support\Facades\DB;
12 use RealRashid\SweetAlert\Facades\Alert;
13 use Kavist\RajaOngkir\Facades\RajaOngkir;
14
15 class AdminController extends Controller
16 {
17     public function __construct()
18     {
19         $this->Admin = new Admin();
20     }
21
22     public function index()
23     {
24         $data = [
25             'admin' => $this->Admin->allData(),
26         ];
27         return view('admin/dataadmin', $data);
28     }
29     public function add()
30     {
31         $ekspedisi = Ekspedisi::pluck('nama', 'code');
32         $provinsis = Provinsi::pluck('nama_provinsi', 'province_id');
33
34         $kotaId = old('kota');
35         $kotaName = Kota::where('city_id', $kotaId)->value('nama_kota');
36         $kotas = [$kotaId => $kotaName];
37
38         return view('admin/add_admin', compact('ekspedisi', 'provinsis', 'kotas'));
39     }
40 }

```

```

41     public function getCities($provinsiId)
42     {
43         // dd($provinceId);
44         $city = Kota::Where('province_id', $provinsiId)->pluck('nama_kota', 'city_id');
45         return json_encode($city);
46     }
47     public function insert()
48     {
49         // Validate the required fields
50         Request()->validate([
51             'firstname' => 'required',
52             // 'lastname' => 'required',
53             'handphone' => 'required|numeric',
54             'alamat' => 'required',
55             'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
56             'password' => ['required', 'string', 'min:8'],
57             // 'img' => 'required | mimes:png,jpg, pdf,jpeg',
58             'provinsi' => 'required',
59             'kota' => 'required',
60         ], [
61             'firstname.required' => 'First name harus diisi!',
62             // 'lastname.required' => 'Last name harus diisi!',
63             'handphone.required' => 'Nomor HP harus diisi!',
64             'handphone.numeric' => 'Nomor HP harus berupa angka!',
65             'alamat.required' => 'Alamat harus diisi!',
66             'email.required' => 'Email harus diisi!',
67             'password.required' => 'Password harus diisi!',
68             // 'img.required' => 'Profil harus diisi!',
69             'provinsi.required' => 'Provinsi harus diisi!',
70             'kota.required' => 'Kota harus diisi!',
71         ]);
72
73         $fileName = null;
74         if (request()->hasFile('img')) {
75             $file = Request()->img;
76             $fileName = Request()->firstname . '.' . $file->extension();
77             $file->move(public_path('foto_profil'), $fileName);
78         }
79
80         // Create a new user record
81         $user = new User();
82         $user->email = Request()->email;
83         $user->password = Hash::make(Request()->password);
84         $user->name = Request()->firstname . ' ' . Request()->lastname;
85         $user->level = 'admin';
86         $user->save();
87
88         // Create a new admin record
89         $admin = new Admin();
90         $admin->province_id = Request()->provinsi;
91         $admin->city_id = Request()->kota;
92         $admin->users_id = $user->id;
93         $admin->firstname = Request()->firstname;
94         $admin->lastname = Request()->lastname;
95         $admin->handphone = Request()->handphone;
96         $admin->alamat = Request()->alamat;
97         $admin->img = $fileName;
98
99         // $admin->img = Request()->img;
100        $admin->save();
101        Alert::success('Berhasil!', 'Data admin berhasil disimpan!');
102        return redirect()->route('dataadmin');
103    }
104 }
105
106     public function destroy($id)
107     {
108         DB::table('admin')->where('id', $id)->delete();
109         Alert::success('Berhasil!', 'Data admin berhasil dihapus!');
110         return redirect('dataadmin');
111     }
112
113     public function edit($id)
114     {
115         $users = User::all();
116         $ekspedisi = Ekspedisi::pluck('nama', 'code');
117         $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
118
119         $admin = Admin::findOrFail($id); // Retrieve the admin using the Admin model
120         ...
121     }

```

```

119 | $admin = Admin::findOrFail($id); // Retrieve the admin using the Admin model
120 | $kotas = Kota::where('province_id', $admin->province_id)->pluck('nama_kota', 'city_id');
121 | // Retrieve the related user data for the admin
122 | $user = $admin->user; // Assuming the column name is 'id_users' in the 'admin' table
123 |
124 | return view('admin/edit_admin', compact('admin', 'user', 'ekspedisi', 'provinsi', 'kotas', 'users'));
125 |
126 |
127 | public function update(Request $request, $id)
128 | {
129 | // Validate the form data
130 | $request->validate([
131 |     'firstname' => 'required',
132 |     'handphone' => 'required|numeric',
133 |     'alamat' => 'required',
134 |     'provinsi' => 'required',
135 |     'kota' => 'required',
136 |     'email' => ['required', 'string', 'email', 'max:255'],
137 |     'password' => ['nullable', 'string', 'min:8'],
138 |     'img' => 'nullable|mimes:png,jpg,pdf,jpeg',
139 | ]);
140 |
141 | // Find the admin record
142 | $admin = Admin::findOrFail($id);
143 |
144 | // Update the admin data
145 | $admin->province_id = $request->provinsi;
146 | $admin->city_id = $request->kota;
147 | $admin->firstname = $request->firstname;
148 | $admin->lastname = $request->lastname;
149 | $admin->handphone = $request->handphone;
150 | $admin->alamat = $request->alamat;
151 |
152 | // Check if the image file is uploaded
153 | if ($request->hasFile('img')) {
154 |     $file = $request->file('img');
155 |     $fileName = $request->firstname . '.' . $file->extension();
156 |     $file->move(public_path('foto_profil'), $fileName);
157 |     $admin->img = $fileName;
158 | }
159 |
160 | // Save the updated admin data
161 | $admin->save();
162 |
163 | // Find the user record associated with the admin
164 | $user = $admin->user;
165 |
166 | // Update the user email
167 | $user->name = $admin->firstname . ' ' . $admin->lastname;
168 | $user->email = $request->input('email');
169 | $user->save();
170 |
171 | // Update the password if provided
172 | if ($request->filled('password')) {
173 |     $user->password = Hash::make($request->input('password'));
174 |     $user->save();
175 | }
176 |
177 | // Redirect to the admin list page
178 | Alert::success('Berhasil!', 'Data admin berhasil diupdate!');
179 | return redirect()->route('dataadmin');
180 |
181 | }
182 |

```

## 5. ChatController

```
1 <?php
2 namespace App\Http\Controllers;
3
4 use App\Models\Umkm;
5 use App\Models\Penjual;
6 use App\Models\Chat;
7 use App\Models\Keranjang;
8 use Illuminate\Support\Facades\Auth;
9 use Illuminate\Support\Facades\Log;
10 use RealRashid\SweetAlert\Facades\Alert;
11 use Illuminate\Http\Request;
12 use Carbon\Carbon;
13
14 class ChatController extends Controller
{
15 /**
16 * Display a listing of the resource.
17 *
18 * @return \Illuminate\Http\Response
19 */
20 public function index()
21 {
22 //
23 try{
24     $cartItems = Keranjang::where('users_id', auth()->user()->id)
25         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
26         ->get();
27     $chats = Chat::where('pembeli_id', Auth::user()->pembeli->id)
28         ->orderby('updated_at','desc')
29         ->get();
30     $firstChat = null; // Initialize $firstChat as null
31     if($chats->count() > 0){
32         $firstChat = Chat::where('pembeli_id', Auth::user()->pembeli->id)
33             ->orderby('updated_at','desc')
34             ->first();
35         if($firstChat){
36             $firstChat->pembeli_status = 1;
37             $firstChat->save();
38         }
39     }
40
41     Log::info("User ".Auth::user()->Pembeli->firstname." ".Auth::user()->Pembeli->lastname." Berhasil mengakses halaman index chat");
42
43     return view('pembeli.chat.index', compact('chats','firstChat','cartItems'));
44 }else{
45     $cartItems = Keranjang::where('users_id', auth()->user()->id)
46         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
47         ->get();
48     return view('pembeli.chat.noindex', compact('cartItems'));
49
50 }
51
52 }
53 catch(\Exception $e){
54
55     return redirect()->back();
56 }
57
58
59 public function index_penjual()
60 {
61 //
62 try{
63     $chats = Chat::where('penjual_id', Auth::user()->penjual->id)
64         ->orderby('updated_at','desc')
65         ->get();
66     if($chats->count() > 0){
67         $firstChat = Chat::where('penjual_id', Auth::user()->penjual->id)
68             ->orderby('updated_at','desc')
69             ->first();
70         if($firstChat){
71             $firstChat->penjual_status = 1;
72             $firstChat->save();
73         }
74
75         Log::info("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengakses halaman index chat");
76     return view('penjual.chat.index', compact('chats','firstChat'));
77 }else{
78 }
```

```

80     |         return redirect()->back();
81     |
82   }
83   catch(\Exception $e){
84     dd($e);
85     return redirect()->back()->with('error', __('toast.index.failed.message'));
86   }
87 }
88 /**
89 * Display the specified resource.
90 *
91 * @param int $id
92 * @return \Illuminate\Http\Response
93 */
94 public function show($id)
95 {
96   try{
97     if (Auth::user()->level === 'penjual'){
98       $chats = Chat::where('penjual_id', Auth::user()->penjual->id)
99           | ->orderby('updated_at','desc')
100          | ->get();
101        $firstChat = Chat::find($id);
102        if($firstChat){
103          $firstChat->penjual_status = 1;
104          $firstChat->save();
105        }
106
107        Log::info("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengakses halaman detail chat");
108        return view('penjual.chat.index', compact('chats','firstChat'));
109      }
110    elseif(Auth::user()->level === 'pembeli'){
111      $chats = Chat::where('pembeli_id', Auth::user()->pembeli->id)
112          | ->orderby('updated_at','desc')
113          | ->get();
114        $firstChat = Chat::find($id);
115        if($firstChat){
116          $firstChat->pembeli_status = 1;
117
118          $firstChat->pembeli_status = 1;
119          $firstChat->save();
120        }
121        $cartItems = Keranjang::where('users_id', auth()->user()->id)
122          ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
123          ->get();
124
125        Log::info("User ".Auth::user()->Pembeli->first_name." ".Auth::user()->Pembeli->last_name." Berhasil mengakses halaman detail chat");
126        return view('pembeli.chat.index', compact('chats','firstChat','cartItems'));
127      }
128    }
129  }
130  catch(\Exception $e){
131    dd($e);
132    return redirect()->back()->with('error', __('toast.index.failed.message'));
133  }
134 }
135
136 public function show_p($id)
137 {
138   try {
139     $chats = Chat::where('penjual_id', Auth::user()->penjual->id)
140           | ->orderby('updated_at','desc')
141           | ->get();
142     $firstChat = Chat::find($id);
143     if($firstChat){
144       $firstChat->penjual_status = 1;
145       $firstChat->save();
146     }
147
148     Log::info("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil mengakses halaman detail chat");
149
150     return view('penjual.chat.index', compact('chats', 'firstChat'));
151   } catch (\Exception $e) {
152     dd($e);
153     return redirect()->back()->with('error', __('toast.index.failed.message'));
154   }
155 }
156 }

```

## 6. CheckoutController

```

1  <?php
2  namespace App\Http\Controllers;
3
4  use Illuminate\Http\Request;
5  use App\Models\Keranjang;
6  use App\Models\Ekspedisi;
7  use App\Models\Provinsi;
8  use App\Models\Kota;
9  use App\Models\Ukm;
10 use Kavist\RajaOngkir\Facades\RajaOngkir;
11 use Illuminate\Support\Facades\Http;
12 use Illuminate\Support\Facades\Session;
13 use Kavist\RajaOngkir\RajaOngkir as RajaOngkirAPI;
14 use App\Models\Produk;
15 use App\Models\User;
16 use Auth;
17 use Illuminate\Support\Facades\Log;
18 use RealRashid\SweetAlert\Facades\Alert;
19 use Illuminate\Support\Facades\DB;
20 use Illuminate\Support\Facades\Hash;
21
22 class CheckoutController extends Controller
23 {
24     public function index(Request $request)
25     {
26         $user = Auth::user();
27         $pembeli = $user->pembeli;
28
29         if ($pembeli) {
30             $ekspedisis = $pembeli->pluck('nama', 'code');
31             $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
32             $kotas = Kota::where('province_id', $pembeli->province_id)->pluck('nama_kota', 'city_id');
33         } else {
34             $ekspedisis = [];
35             $provinsi = [];
36             $kotas = [];
37             $pembeli = (object) ['img' => null];
38         }
39
40         $cartItems = Keranjang::where('users_id', auth()->user()->id)
41             ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
42             ->get();
43
44         // Mengambil data produk dari keranjang
45         $produkItems = [];
46         $subtotal = 0;
47         $hargas = [];
48
49         foreach ($cartItems as $item) {
50             $produkItems[] = $item->produk->nama;
51             $hargaProduk = $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
52             $subtotal += $hargaProduk->jumlah;
53             $hargas[$item->id] = $hargaProduk;
54         }
55
56         $biayaPengiriman = 0; // Default value jika biaya pengiriman tidak tersedia
57
58         $produkIds = $cartItems->pluck('produk_id'); // Ambil semua produk_id dari cartItems
59         $produk = Produk::whereIn('id', $produkIds)->first(); // Ambil salah satu produk untuk mendapatkan umkm_id
60
61         $umkm = Umkm::find($produk->umkm_id); // Ambil data UMKM berdasarkan umkm_id
62         $provinsiAsal = Provinsi::find($umkm->province_id);
63         $kotaAsal = Kota::join('umkm', 'kota.city_id', '=', 'umkm.city_id')
64             ->where('umkm.id', $umkm->id)
65             ->select('kota.city_id')
66             ->first();
67
68         $provinsiTujuan = Provinsi::find($pembeli->province_id);
69         $kotaTujuan = Kota::join('pembeli', 'kota.city_id', '=', 'pembeli.city_id')
70             ->where('pembeli.id', $pembeli->id)
71             ->select('kota.city_id')
72             ->first();
73
74         $beratPengiriman = $cartItems->sum(function ($item) {
75             return $item->produk->Detail_Produk->first()->berat_pengiriman * $item->jumlah;
76         });
77
78         // Mengambil semua ID dari tabel Keranjang
79
80         ->pluck('id')
81         ->toArray();
82
83         // Mengumpulkan informasi umkm berdasarkan produk yang ada di keranjang
84         $umkmIds = Produk::whereIn('id', $produkIds)->pluck('umkm_id')->unique();
85
86         $umkmMetodePembayaran = [];
87
88         foreach ($umkmIds as $umkmId) {
89             $umkm = Umkm::find($umkmId);
90             $metodePembayaranUmkm = $umkm->Detail_Metode_Pembayaran;
91             $umkmMetodePembayaran[$umkmId] = $metodePembayaranUmkm;
92         }
93
94         $cekongkir = session('cekongkir');
95         // Ambil nilai ekspedisi_id dari Session
96         $ekspedisiId = session('ekspedisi_id');
97
98
99         return view('pembeli.pembelian.checkout', compact('cartItems', 'pembeli', 'user', 'ekspedisis', 'provinsi', 'kotas', 'produkItems', 'subtotal', 'hargas', 'biayaPeng'));
100    }
101
102    public function updateAddress(Request $request)
103    {
104        $user = Auth::user();
105        $pembeli = $user->pembeli;
106
107        // Validate the form inputs
108        $request->validate([
109            'provinsi' => 'required',
110            'kota' => 'required',
111            'alamat' => 'required',
112            'ekspedisi' => 'required',
113        ]);
114
115        $pembeli->province_id = $request->provinsi;
116        $pembeli->city_id = $request->kota;
117        $pembeli->alamat = $request->alamat;
118    }

```

```

119 $pembeli->save();
120
121 $cartItems = Keranjang::where('users_id', auth()->user->id)
122     ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
123     ->get();
124
125     // Mengambil data produk dari keranjang
126     $produkItems = [];
127     $subtotal = 0;
128     $hargaas = [];
129
130     foreach ($cartItems as $item) {
131         $produkItems[] = $item->produk->nama;
132         $hargaProduk = $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
133         $subtotal += $hargaProduk * $item->jumlah;
134         $hargaas[$item->id] = $hargaProduk;
135     }
136
137     $produkIds = $cartItems->pluck('produk_id'); // Ambil semua produk_id dari cartItems
138     $produk = Produk::whereIn('id', $produkIds)->first(); // Ambil salah satu produk untuk mendapatkan umkm_id
139
140     $umkm = Umkm::find($produk->umkm_id); // Ambil data UMKM berdasarkan umkm_id
141     $provinsiAsal = Provinsi::find($umkm->province_id);
142     $kotaAsal = Kota::join('umkm', 'kota.city_id', '=', 'umkm.city_id')
143         ->where('umkm.id', $umkm->id)
144         ->select('kota.city_id')
145         ->first();
146
147     $provinsiTujuan = Provinsi::find($pembeli->province_id);
148     $kotaTujuan = Kota::join('pembeli', 'kota.city_id', '=', 'pembeli.city_id')
149         ->where('pembeli.id', $pembeli->id)
150         ->select('kota.city_id')
151         ->first();
152
153     $beratPengiriman = $cartItems->sum(function ($item) {
154         return $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->berat_pengiriman * $item->jumlah;
155     });
156
157
158     $selectedEkspedisi = $request->ekspedisi; // Ambil nilai ekspedisi dari inputan
159     $selectedKota = $request->kota;
160
161
162     $response = Http::asForm()->withHeaders([
163         'key' => 'fa28c78a8d8a150e5800la1a0ce014cba'
164     ])->post("https://api.rajaongkir.com/starter/cost", [
165         'origin' => $kotaAsal->city_id,
166         'destination' => $selectedKota,
167         'weight' => $beratPengiriman,
168         'courier' => $selectedEkspedisi
169     ]);
170
171
172     $cekongkir = $response['rajaongkir'][0]['results'][0]['costs'];
173     // Simpan ekspedisi_id ke dalam session
174     $selectedEkspedisId = $request->input('ekspedisi');
175     session(['ekspedisi_id' => $selectedEkspedisId]);
176
177
178     // Alert:success('Berhasil!', 'Data alamat berhasil diupdate!');
179     return redirect()->route('checkout')->with('cekongkir', $cekongkir);
180 }
181
182
183

```

## 7. HistoryController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\Auth;
7 use App\Models\Pembelian;
8 use App\Models\Detail_Pembelian;
9 use App\Models\History;
10 use App\Models\Umkm;
11 use App\Models\Produk;
12 use App\Models\Keranjang;
13 use App\Http\Resources\Transaksilist;
14
15 class HistoryController extends Controller
16 {
17     /**
18      * Display a listing of the resource.
19      *
20      * @return \Illuminate\Http\Response
21     */
22     public function index()
23     {
24         try{
25             if(Auth::user()->level === 'pembeli')
26             {
27                 $histories = History::where('pembeli_id', Auth::user()->pembeli->id)
28                             ->orderBy('id','desc')
29                             ->get();
30
31                 $cartItems = Keranjang::where('users_id', auth()->user()->id)
32                     ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
33                     ->get();
34
35                 // Mengambil data pembelian yang belum dibayar untuk user yang sedang login
36                 $pembelian = Pembelian::all()
37                     ->whereHas('Detail_Pembelian', function ($query) {
38                         // Menambahkan kriteria untuk hanya mengambil detail pembelian dari produk yang terhubung dengan UMKM yang sesuai
39                         $query->whereHas('produk.umkm', function ($umkmQuery) {
40                             $umkmQuery->where('nama_pembeli', auth()->user()->name);
41                         });
42                     });
43
44             }
45         }
46     }
47 }
```

```

41     |    })
42     |    ->with("Detail_Pembelian.produk.umkm") // Menambahkan relasi dengan model Produk dan Umkm
43     |    ->get();
44
45     |    return view('pembeli.history.index', compact('histories', 'cartItems','pembelian' ));
46 }
47 elseif(Auth::user()->level === 'penjual')
48 {
49     $pembelian = Pembelian::where('status_pembelian',4)->get();
50     $umkm = Umkm::where('penjual_id', Auth::user()->penjual->id)->where('status',1)->get()->pluck('nama', 'id');
51     // $schedules = Schedule::where('')
52     $produk = Produk::pluck('nama', 'id');
53
54
55     return view('penjual.manage_history.index', compact('pembelian','umkm','produk'));
56 }
57 catch(\Exception $e){
58     |    return redirect()->back();
59 }
60
61
62 public function index_pembeli(){
63     try{
64         $histories = History::where('pembeli_id', Auth::user()->pembeli->id)
65         ->orderBy('id','desc')
66         ->get();
67         $cartItems = Keranjang::where('users_id', auth()->user()->id)
68         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
69         ->get();
70
71         // Mengambil data pembelian yang belum dibayar untuk user yang sedang login
72         $pembelian = Pembelian::where('status_pembayaran', 1)
73         ->whereHas('Detail_Pembelian', function ($query) {
74             // Menambahkan kriteria untuk hanya mengambil detail pembelian dari produk yang terhubung dengan Umkm yang sesuai
75             $query->whereHas('produk.umkm', function ($umkmQuery) {
76                 $umkmQuery->where('nama_pembeli', auth()->user()->name);
77             });
78         });
79     }
80
81     ->with('Detail_Pembelian.produk.umkm') // Menambahkan relasi dengan model Produk dan Umkm
82     ->get();
83
84     return view('pembeli.history.index', compact('histories', 'cartItems','pembelian' ));
85 }
86 catch(\Exception $e){
87     |    return redirect()->back();
88 }
89
90 public function getData(Request $request)
91 {
92     $data = [];
93     if($request->data=="all"){
94         $umkms = Umkm::where('penjual_id', $request->penjual_id)->get();
95         $umkm_id = collect([[]]);
96         foreach($umkms as $umkm){
97             $umkm_id->push($umkm->id);
98         }
99         $produk = Produk::whereIn('umkm_id', $umkm_id)->get();
100        $produk_id = collect([[]]);
101        foreach($produk as $produk){
102            $produk_id->push($produk->id);
103        }
104        $data = Pembelian::with(['Detail_Pembelian.Produk.Umkm'])
105        ->whereHas('Detail_Pembelian.Produk.Umkm', function ($query) use ($request) {
106            $query->where('penjual_id', $request->penjual_id);
107        })
108        ->where('id', 4)
109        ->orderBy('id', 'desc')
110        ->get();
111
112
113
114
115
116     elseif($request->data=="id"){
117         $data = Pembelian::find($request->id);
118     }
119

```

```

120     elseif($request->data=="select"){
121         $id = explode(',',$request->id);
122         $data = Pembelian::wherenotin('id', $id)->get();
123     }
124     // dd($data);
125     if($data) return response()->json(TransaksiList::collection($data));
126     return $data;
127 }
128
129 /**
130 * Display the specified resource.
131 *
132 * @param int $id
133 * @return \Illuminate\Http\Response
134 */
135 public function show($id)
{
    try{
        if(Auth::user()->level === 'pembeli')
        {
            $pembelian = Pembelian::find($id);
            return view('pembeli.manage_transaction.show', compact('pembelian'));
        }
        elseif(Auth::user()->level === 'penjual')
        {
            try{
                $pembelian = Pembelian::find($id);
                // dd($pembelian);
                return view('penjual.manage_history.show', compact('pembelian'));
            } catch (\Exception $e) {
                return redirect()->back();
            }
        }
    }
    catch(\Exception $e){
        dd($e);
        return redirect()->back();
    }
}
161
162 public function show_pembeli($id){
163     $cartItems = Keranjang::where('users_id', auth()->user()->id)
164     ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
165     ->get();
166
167     $detail_pembelian = Detail_Pembelian::with('produk')->find($id);
168     // Access the associated pembelian data using the relationship
169     $pembelian = $detail_pembelian->pembelian;
170
171     return view('pembeli.history.show', compact('cartItems', 'detail_pembelian', 'pembelian'));
172 }
173

```

## 8. KategoriController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Kategori;
7 use Illuminate\Support\Facades\DB;
8 use RealRashid\SweetAlert\Facades\Alert;
9
10 class KategoriController extends Controller
{
11     public function __construct()
12     {
13         $this->Kategori = new Kategori();
14     }
15
16     public function index()
17     {
18         $data = [
19             'kategori' => $this->Kategori->allData(),
20         ];
21         return view('kategori/kategori', $data);
22     }
23     public function add()
24     {
25         return view ('kategori/add_kategori');
26     }
27
28     public function insert()
29     {
30         Request()->validate([
31             'nama_kategori' => 'required',
32         ],[
33             'nama_kategori.required' => 'nama kategori harus diisi!',
34         ]);
35
36         $data = [
37             'nama_kategori' => Request()->nama_kategori,
38         ];
39     }
40

```

```

    ...
42     $this->kategori->addData($data);
43     Alert::success('Berhasil!', 'Data kategori berhasil disimpan!');
44     return redirect()->route('kategori');
45 }
46 public function destroy($id)
{
47     DB::table('kategori')->where('id', $id)->delete();
48     Alert::success('Berhasil!', 'Data kategori berhasil dihapus!');
49     return redirect('kategori');
50 }
51
52 public function edit($id)
{
53     $kategori = DB::table('kategori')->find($id);
54     return view('kategori/edit_kategori',[ 'kategori'=>$kategori]);
55 }
56 /**
57 * Update the specified resource in storage.
58 *
59 * @param \Illuminate\Http\Request $request
60 * @param int $id
61 * @return \Illuminate\Http\Response
62 */
63 public function update(Request $request, $id)
{
64     $affected = DB::table('kategori')
65         ->where('id', $id)
66         ->update(['nama_kategori' => $request->nama_kategori]);
67
68     Alert::success('Berhasil!', 'Data kategori berhasil diupdate!');
69     return redirect('/kategori');
70 }
71
72 }
73
74 }
75
76 }
77
78 }

```

## 9. KeranjangController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Keranjang;
7 use App\Models\Detail_Produk;
8 use App\Models\Produk;
9 use Illuminate\Support\Facades\Session;
10 use RealRashid\SweetAlert\Facades\Alert;
11
12 class KeranjangController extends Controller
13 {
14     public function addToCart(Request $request)
15     {
16         $validatedData = $request->validate([
17             'produk_id' => 'required|integer',
18             'jumlah' => 'required|integer|min:1',
19             'size' => 'required',
20         ], [
21             'jumlah.min' => 'Minimal pembelian adalah 1 produk.',
22         ]);
23
24         // Tambahkan validasi untuk memastikan kemasan_id dipilih
25         if (!$validatedData['size']) {
26             Session::flash('error', 'Pilih dulu variasi produk sebelum menambahkan ke keranjang.');
27             return redirect()->back();
28         }
29
30         $product = Produk::findOrFail($validatedData['produk_id']);
31         $umkmIdToAdd = $product->umkm_id;
32
33         // Check if there are products from a different UMKM already in the cart
34         $existingProducts = Keranjang::where('users_id', auth()->user()->id)->get();
35         foreach ($existingProducts as $existingProduct) {
36             $existingProductUmkmId = $existingProduct->produk->umkm_id;
37             if ($existingProductUmkmId !== $umkmIdToAdd) {
38                 Session::flash('error', 'Tidak bisa menambahkan produk ke keranjang. Selesaikan terlebih dahulu transaksi produk pada UMKM yang');
39             }
40         }
41     }
42 }

```

```

1-
43     $detailProduk = Detail_Produk::where('produk_id', $validatedData['produk_id'])
44         ->where('kemasan_id', $request->input('size'))
45         ->first();
46
47 if (!$detailProduk) {
48     Session::flash('error', 'Kemasan yang dipilih tidak tersedia untuk produk ini.');
49     return redirect()->back();
50 }
51
52 $keranjang = Keranjang::where('users_id', auth()->user()->id)
53     ->where('produk_id', $validatedData['produk_id'])
54     ->where('kemasan_id', $request->input('size'))
55     ->first();
56
57 if ($keranjang) {
58     $keranjang->jumlah += $validatedData['jumlah'];
59     $keranjang->save();
60 } else {
61     // Tambahkan entri baru ke keranjang
62     $keranjangBaru = new Keranjang();
63     $keranjangBaru->users_id = auth()->user()->id;
64     $keranjangBaru->produk_id = $validatedData['produk_id'];
65     $keranjangBaru->kemasan_id = $request->input('size');
66     $keranjangBaru->jumlah = $validatedData['jumlah'];
67     $keranjangBaru->save();
68 }
69
70 return redirect()->route('keranjang');
71 }
72 public function showCart()
73 {
74     $cartItems = Keranjang::where('users_id', auth()->user()->id)
75         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
76         ->get();
77
78 return view('pembeli.pembelian.keranjang', [
79     'cartItems' => $cartItems
80 ]);
81 }
82
83 public function updateCart(Request $request)
84 {
85     $validatedData = $request->validate([
86         'jumlah' => 'required|array',
87         'jumlah.*' => 'required|integer|min:1',
88     ]);
89
90     foreach ($validatedData['jumlah'] as $keranjangId => $jumlah) {
91         $keranjang = Keranjang::findOrFail($keranjangId);
92         $keranjang->jumlah = $jumlah;
93         $keranjang->save();
94     }
95
96     return redirect()->route('keranjang');
97 }
98 public function removeFromCart($id)
99 {
100     $keranjang = Keranjang::findOrFail($id);
101     $keranjang->delete();
102
103     return response()->json(['message' => 'Product removed from cart']);
104 }
105

```

## 10. LandingController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Kategori;
7 use App\Models\Produk;
8 use App\Models\Umkm;
9 use App\Models\Keranjang;
10 use App\Models\Chat;
11 use Illuminate\Support\Facades\Auth;
12 use App\Helper;
13
14 class LandingController extends Controller
15 {
16     public function index()
17     {
18         $kategori = new Kategori();
19         $datakategori = $kategori->allData();
20         $produk = Produk::with('Umkm', 'Imgproduk')->get();
21         $umkm = Ukm::where('status', 1)->get();
22         $cartItems = [];
23
24         if (auth()->check()) {
25             $cartItems = Keranjang::where('users_id', auth()->user()->id)
26             | ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
27             | ->get();
28         }
29
30         return view('landing.welcome', compact('datakategori', 'produk', 'cartItems', 'umkm'));
31     }
32
33     public function search(Request $request)
34     {
35         $kategori = new Kategori();
36         $datakategori = $kategori->allData();
37         $keyword = $request->input('keyword');
38
39         $produk = Produk::with('umkm', 'imgproduk')
40             | ->where('nama', 'like', '%'.$keyword.'%')
41
42             | ->orWhereHas('umkm', function ($query) use ($keyword) {
43                 $query->where('nama', 'like', '%'.$keyword.'%');
44             })
45             | ->get();
46
47         if ($produk->count() === 0) {
48             $message = 'No data found for the search keyword: '.$keyword;
49         } else {
50             $message = null;
51         }
52         $cartItems = [];
53
54         if (auth()->check()) {
55             $cartItems = Keranjang::where('users_id', auth()->user()->id)
56             | ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
57             | ->get();
58
59         return view('landing.search', compact('produk', 'datakategori', 'message', 'cartItems'));
60     }
61
62     public function searchproduk(Request $request)
63     {
64         $keyword = $request->input('keyword');
65
66         $sumkmId = $request->input('umkm_id');
67         $umkm = Ukm::findOrFail($sumkmId);
68         $produk = Produk::where('umkm_id', $sumkmId)
69             | ->where('nama', 'LIKE', '%'.$keyword.'%')
70             | ->get();
71
72         $datakategori = Kategori::join('produk', 'kategori.id', '=', 'produk.kategori_id')
73             | ->where('produk.umkm_id', $sumkmId)
74             | ->get();
75
76         if ($produk->count() === 0) {
77             $message = 'No data found for the search keyword: '.$keyword;
78         } else {
79             $message = null;
```

```

80      }
81      $data = [
82          'umkm' => $this->Umkm->Datacity($umkm->city_id),
83      ];
84      $cartItems = [];
85
86      if ($auth()->check()) {
87          $cartItems = Keranjang::where('users_id', $auth()->user()->id)
88              ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
89              ->get();
90      }
91
92      return view('landing.searchproduk', compact('produks', 'dataKategori', 'message', 'umkm', 'data', 'cartItems'));
93  }
94
95
96  public function showCategory($id)
97  {
98      $dataKategori = Kategori::all();
99      $kategori = Kategori::findOrFail($id);
100     $produks = Produk::with('umkm', 'imgproduk')
101         ->where('kategori_id', $kategori->id)
102         ->get();
103     $cartItems = [];
104
105     if ($auth()->check()) {
106         $cartItems = Keranjang::where('users_id', $auth()->user()->id)
107             ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
108             ->get();
109     }
110
111     return view('landing.kategori', compact('produks', 'dataKategori', 'kategori', 'cartItems'));
112 }
113
114 {
115 // Mengambil data umkm berdasarkan id yang diberikan
116 $umkm = Umkm::findOrFail($umkmId);
117
118 // Mengambil data kategori berdasarkan id yang diberikan
119 $kategori = Kategori::findOrFail($kategoriId);
120
121 $produks = Produk::with('umkm', 'imgproduk')
122     ->where('umkm_id', $umkmId)
123     ->when($kategoriId, function ($query, $kategoriId) {
124         return $query->where('kategori_id', $kategoriId);
125     })
126     ->get();
127
128
129
130     $dataKategori = Kategori::whereIn('id', function ($query) use ($umkmId) {
131         $query->select('kategori.id')
132             ->from('kategori')
133             ->join('produk', 'kategori.id', '=', 'produk.kategori_id')
134             ->where('produk.umkm_id', $umkmId)
135             ->groupBy('kategori.id');
136     })->get();
137
138     $data = [
139         'umkm' => $this->Umkm->Datacity($umkm->city_id),
140     ];
141
142     $cartItems = [];
143
144     if ($auth()->check()) {
145         $cartItems = Keranjang::where('users_id', $auth()->user()->id)
146             ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
147             ->get();
148     }
149
150     return view('landing.kategoriumkm', compact('produks', 'umkm', 'kategori', 'data', 'cartItems', 'dataKategori'));
151 }
152
153
154  public function showDetail($id)
155  {
156      $dataKategori = Kategori::all();
157      $product = Produk::with('Umkm', 'Imgproduk')->findOrFail($id);
158      $cartItems = [];
159

```

```

158     if (auth()->check()) {
159         $cartItems = Keranjang::where('users_id', auth()->user()->id)
160             ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
161             ->get();
162     }
163
164     return view('landing.produkdetail', compact('product', 'dataKategori', 'cartItems'));
165 }
166
167 public function __construct()
168 {
169     $this->Umkm = new Umkm();
170 }
171
172 public function showDetailUmkm($id)
173 {
174     $umkm = Umkm::findOrFail($id);
175     $produk = Produk::where('umkm_id', $id)->get();
176
177     // Mengambil kategori yang terkait dengan produk pada umkm
178     // $dataKategori = Kategori::join('produk', 'kategori.id', '=', 'produk.kategori_id')
179     //     ->where('produk.umkm_id', $id)
180     //     ->get();
181
182     $dataKategori = Kategori::whereIn('id', function ($query) use ($id) {
183         $query->select('kategori.id')
184             ->from('kategori')
185             ->join('produk', 'kategori.id', '=', 'produk.kategori_id')
186             ->where('produk.umkm_id', $id)
187             ->groupBy('kategori.id');
188     } )->get();
189
190     $data = [
191         'umkm' => $this->Umkm->Datacity($umkm->city_id),
192     ];
193
194     if (auth()->check())
195     {
196         $cartItems = Keranjang::where('users_id', auth()->user()->id)
197             ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
198             ->get();
199
200         $chat = [];
201         if(Auth::user()){
202             $chat = Chat::where('penjual_id', $umkm->Penjual->id)
203                 ->where('pembeli_id', Auth::user()->pembeli->id)
204                 ->where('umkm_id', $id)
205                 ->first();
206         }
207     }
208 }
209

```

## 11. LoginWithGoogleController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Laravel\Socialite\Facades\Socialite;
7 use App\Models\User;
8 use Illuminate\Support\Facades\Auth;
9 use Exception;
10
11 class LoginWithGoogleController extends Controller
12 {
13     public function redirectToGoogle(){
14         return Socialite::driver('google')->redirect();
15     }
16
17     public function handleGoogleCallback()
18     {
19         try {
20             $user = Socialite::driver('google')->user();
21             $finduser = User::where('google_id', $user->id)->first();
22
23             if($finduser){
24                 Auth::login($finduser);
25                 return redirect()->intended('dashboard');
26             }else{
27                 $newUser = User::create([
28                     'name' => $user->name,
29                     'email' => $user->email,
30                     'google_id' => $user->id,
31                     'level' => 'pembeli',
32                     'password' => encrypt('123456dummy')
33                 ]);
34
35                 Auth::login($newUser);
36                 return redirect()->intended('dashboard');
37             }
38         } catch (Exception $e) {
39             dd($e->getMessage());
40         }
41     }
42

```

```

41     }
42     public function __construct()
43     {
44         $this->middleware('guest')->except('logout');
45     }
46 }
47

```

## 12. PembelianController

```

1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use Carbon\Carbon;
7  use App\Models\Pembelian;
8  use App\Models\Pembeli;
9  use App\Models\Detail_Pembelian;
10 use App\Models\Keranjang;
11 use App\Models\Pembayaran;
12 use App\Models\History;
13 use App\Models\Metode_Pembayaran;
14 use App\Models\Detail_Metode_Pembayaran;
15 use Illuminate\Support\Facades\Session;
16 use Illuminate\Support\Facades\Auth;
17 use Illuminate\Support\Facades\DB;
18
19 class PembelianController extends Controller
20 {
21     public function placeOrder(Request $request)
22     {
23         $request->validate([
24             'layanan' => 'required',
25             'payment_method' => 'required',
26         ]);
27
28         // Ambil nilai ekspedisi_id dari Session
29         $ekspedisiId = session('ekspedisi_id');
30
31         $pembelian = new Pembelian;
32         $pembelian->token = Carbon::now()->addHours(3)->format('YmdHis');
33         $pembelian->nama_pembeli = Auth::user()->pembeli->firstname . ' ' . Auth::user()->pembeli->lastname;
34         $pembelian->total_harga = $request->input('total');
35         $pembelian->status_pembayaran = 0;
36         $pembelian->status_pembelian = 0;
37         $pembelian->ekspedisi_id = $ekspedisiId;
38         $pembelian->save();
39
40         // Set ID pembelian ke keranjang
41         $pembelian_id = $pembelian->id;
42
43         // Simpan data detail pembelian
44         $cartItems = Keranjang::where('users_id', auth()->user()->id)->get();
45
46         foreach ($cartItems as $cartItem) {
47             // Check if the cart item exists and is associated with a product
48             if ($cartItem && $cartItem->produk) {
49                 $detailPembelian = new Detail_Pembelian;
50                 $detailPembelian->pembelian_id = $pembelian_id;
51                 $detailPembelian->produk_id = $cartItem->produk->id;
52                 $detailPembelian->jumlah = $cartItem->jumlah;
53                 $detailPembelian->kemasan_id = $cartItem->kemasan->id;
54                 // Add other relevant data from the cart item or the related product
55                 $detailPembelian->save();
56             }
57         }
58         try {
59             DB::beginTransaction();
60             $pembayaran = new Pembayaran;
61             $pembayaran->detailmetpemb_id = $request->input('payment_method');
62             $pembayaran->pembelian_id = $pembelian_id;
63             $pembayaran->save();
64
65             Keranjang::where('users_id', auth()->user()->id)->delete();
66
67             DB::commit();
68
69             return redirect()->route('payment', ['id' => $detailPembelian])->with('success', 'Order has been successfully placed. Thank you for your order!');
70         } catch (\Exception $e) {
71             DB::rollback();
72             return redirect()->back()->with('error', 'An error occurred while processing the order. Please try again.');
73         }
74     }
75
76     public function payment($id){
77
78         $cartItems = Keranjang::where('users_id', auth()->user()->id)
79         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')

```

```

80     ->get();
81
82     $detail_pembelian = Detail_Pembelian::with('produk')->find($id);
83     $pembelian = $detail_pembelian->pembelian;
84
85     $pembayaran = $pembelian->pembayaran;
86
87     $metode_pembayaran = $pembayaran->Detail_Metode_Pembayaran;
88
89     $nama_bank = $metode_pembayaran->Metode_Pembayaran->nama_bank;
90
91     return view('pembeli.pembelian.payment', compact('cartItems', 'detail_pembelian', 'pembelian', 'metode_pembayaran', 'nama_bank'));
92 }
93
94
95 public function postPayment(Request $request)
96 {
97     $request->validate([
98         'bukti_transfer' => 'required|image|mimes:jpeg,png,jpg,gif|max:2048',
99     ]);
100
101    $buktiTransferPath = null;
102    if ($request->hasFile('bukti_transfer')) {
103        $image = $request->file('bukti_transfer');
104        $imageName = time() . '.' . $image->getClientOriginalExtension();
105        $image->move(public_path('/bukti_transfer'), $imageName);
106        $buktiTransferPath = $imageName;
107    }
108
109    $pembelianId = $request->input('pembelian_id');
110
111    Pembayaran::where('pembelian_id', $pembelianId)->update(['pembayaran' => $buktiTransferPath]);
112
113    Pembelian::where('id', $pembelianId)->update(['status_pembayaran' => 1]);
114
115    $pembelian = Pembelian::find($pembelianId);
116
117    if ($pembelian->status_pembayaran === 1) {
118        $pembeli_id = Pembeli::where('users_id', Auth::user()->id)->value('id');
119
120        $history = new History();
121        $history->pembeli_id = $pembeli_id;
122        $history->pembelian_id = $pembelianId; // Menggunakan $pembelianId yang telah diperbarui
123        $history->save();
124    }
125    return redirect()->back()->with('success', 'Payment has been submitted successfully. Thank you for your payment!');
126 }
127
128
129 public function purchase(){
130
131     $cartItems = Keranjang::where('users_id', auth()->user()->id)
132         ->with('produk', 'produk.Detail_Produk', 'produk.Imgproduk')
133         ->get();
134
135     $pembelian = Pembelian::where('status_pembayaran', 0)
136         ->whereHas('Detail_Pembelian', function ($query) {
137             $query->whereHas('produk.umkm', function ($umkmQuery) {
138                 $umkmQuery->where('nama_pembeli', auth()->user()->name);
139             });
140         })
141         ->with('Detail_Pembelian.produk.umkm') // Menambahkan relasi dengan model Produk dan Ukm
142         ->get();
143
144
145     return view('pembeli.pembelian.purchase', compact('cartItems', 'pembelian'));
146 }
147
148 public function delete($id)
149 {
150     $pembelian = Pembelian::find($id);
151
152     if ($pembelian) {
153         $pembelian->delete();
154         return response()->json(['message' => 'Pembelian berhasil dihapus'], 200);
155     }
156
157     return response()->json(['message' => 'Pembelian tidak ditemukan'], 404);
158 }

```

### 13. PembeliController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Pembeli;
7 use App\Models\Ekspedisi;
8 use App\Models\Provinsi;
9 use App\Models\Kota;
10 use App\Models\User;
11 use Illuminate\Support\Facades\Hash;
12 use Illuminate\Support\Facades\DB;
13 use RealRashid\SweetAlert\Facades\Alert;
14 use Kavist\RajaOngkir\Facades\RajaOngkir;
15
16 class PembeliController extends Controller
17 {
18     public function __construct()
19     {
20         $this->Pembeli = new Pembeli();
21     }
22
23     public function index()
24     {
25         $data = [
26             'pembeli' => $this->Pembeli->allData(),
27         ];
28
29         return view('pembeli/datapembeli', $data);
30     }
31
32     public function add()
33     {
34         $ekspedisi = Ekspedisi::pluck('nama', 'code');
35         $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
36
37         $kotaId = old('kota');
38         $kotaName = Kota::where('city_id', $kotaId)->value('nama_kota');
39         $kotas = [$kotaId => $kotaName];
40
41         return view('pembeli/add_pembeli', compact('ekspedisi', 'provinsi', 'kotas'));
42     }
43
44     public function getCities($provinsiId)
45     {
46         $kota = Kota::where('province_id', $provinsiId)->pluck('nama_kota', 'city_id');
47         return json_encode($kota);
48     }
49
50     public function insert()
51     {
52         // Validate the required fields
53         Request()->validate([
54             'firstname' => 'required',
55             'handphone' => 'required|numeric',
56             'alamat' => 'required',
57             'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
58             'password' => ['required', 'string', 'min:8'],
59             'provinsi' => 'required',
60             'kota' => 'required',
61         ], [
62             'firstname.required' => 'First name harus diisi!',
63             'handphone.required' => 'Nomor HP harus diisi!',
64             'handphone.numeric' => 'Nomor HP harus berupa angka!',
65             'alamat.required' => 'Alamat harus diisi!',
66             'email.required' => 'Email harus diisi!',
67             'password.required' => 'Password harus diisi!',
68             'provinsi.required' => 'Provinsi harus diisi!',
69             'kota.required' => 'Kota harus diisi!',
70         ]);
71
72         $fileName = null;
73         if (request()->hasFile('img')) {
74             $file = Request()->img;
75             $fileName = Request()->firstname . '.' . $file->extension();
76             $file->move(public_path('foto_profil'), $fileName);
77         }
78
79     }
```

```

82 // Create a new user record
83 $user = new User();
84 $user->email = Request()->email;
85 $user->password = Hash::make(Request()->password);
86 $user->name = Request()->firstname . ' ' . Request()->lastname;
87 $user->level = 'pembeli';
88 $user->save();
89
90 // Create a new pembeli record
91 $pembeli = new Pembeli();
92 $pembeli->province_id = Request()->provinsi;
93 $pembeli->city_id = Request()->kota;
94 $pembeli->users_id = $user->id;
95 $pembeli->firstname = Request()->firstname;
96 $pembeli->lastname = Request()->lastname;
97 $pembeli->handphone = Request()->handphone;
98 $pembeli->alamat = Request()->alamat;
99 $pembeli->img = $fileName;
100
101 // $pembeli->img = Request()->img;
102 $pembeli->save();
103 Alert::success('Berhasil!', 'Data pembeli berhasil disimpan!');
104 return redirect()->route('datapembeli');
105 }
106
107 public function destroy($id)
108 {
109     DB::table('pembeli')->where('id', $id)->delete();
110     Alert::success('Berhasil!', 'Data pembeli berhasil dihapus!');
111     return redirect('datapembeli');
112 }
113
114 public function edit($id)
115 {
116     $users = User::all();
117     $ekspedisi = Ekspedisi::pluck('nama', 'code');
118     $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
119
120     $pembeli = Pembeli::findOrFail($id); // Retrieve the pembeli using the pembeli model
121     $kotas = Kota::where('province_id', $pembeli->province_id)->pluck('nama_kota', 'city_id');
122
123     $user = $pembeli->user;
124
125     return view('pembeli/edit_pembeli', compact('pembeli', 'user', 'ekspedisi', 'provinsi', 'kotas', 'users'));
126 }
127
128 public function update(Request $request, $id)
129 {
130     // Validate the form data
131     $request->validate([
132         'firstname' => 'required',
133
134         'handphone' => 'required|numeric',
135         'alamat' => 'required',
136         'provinsi' => 'required',
137         'kota' => 'required',
138         'email' => ['required', 'string', 'email', 'max:255'],
139         'password' => ['nullable', 'string', 'min:8'],
140         'img' => 'nullable|mimes:png,jpg,pdf,jpeg',
141     ]);
142
143     // Find the pembeli record
144     $pembeli = Pembeli::findOrFail($id);
145
146     // Update the pembeli data
147     $pembeli->province_id = $request->provinsi;
148     $pembeli->city_id = $request->kota;
149     $pembeli->firstname = $request->firstname;
150     $pembeli->lastname = $request->lastname;
151     $pembeli->handphone = $request->handphone;
152     $pembeli->alamat = $request->alamat;
153
154     // Check if the image file is uploaded
155     if ($request->hasFile('img')) {
156         $file = $request->file('img');
157         $fileName = $request->firstname . '.' . $file->extension();
158         $file->move(public_path('foto_profil'), $fileName);
159         $pembeli->img = $fileName;
160     }
...

```

```

162 // Save the updated pembeli data
163 $pembeli->save();
164
165 // Find the user record associated with the pembeli
166 $user = $pembeli->user;
167
168 // Update the user email
169 $user->name = $pembeli->firstname . ' ' . $pembeli->lastname;
170 $user->email = $request->input('email');
171 $user->save();
172
173 // Update the password if provided
174 if ($request->filled('password')) {
175     $user->password = Hash::make($request->input('password'));
176     $user->save();
177 }
178
179 // Redirect to the pembeli list page
180 Alert::success('Berhasil', 'Data pembeli berhasil diupdate!');
181 return redirect()->route('datapembeli');
182 }
183 }
184

```

## 14. PenjualController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Penjual;
7 use App\Models\Ekspedisi;
8 use App\Models\Provinsi;
9 use App\Models\Kota;
10 use App\Models\User;
11 use Illuminate\Support\Facades\Hash;
12 use Illuminate\Support\Facades\DB;
13 use Auth;
14 use RealRashid\SweetAlert\Facades\Alert;
15 use Illuminate\Support\Facades\Log;
16
17 class PenjualController extends Controller
18 {
19     public function __construct()
20     {
21         $this->Penjual = new Penjual();
22     }
23
24     public function index()
25     {
26         $data = [
27             'penjual' => $this->Penjual->allData(),
28         ];
29
30         return view('penjual/datapenjual', $data);
31     }
32     public function add()
33     {
34         $ekspedisi = Ekspedisi::pluck('nama', 'code');
35         $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
36
37         $kotaId = old('kota');
38         $kotaName = Kota::where('city_id', $kotaId)->value('nama_kota');
39         $kotas = [$kotaId => $kotaName];
40

```



```

41     return view('penjual/add_penjual', compact('ekspedisi', 'provinsi', 'kotas'));
42 }
43 public function getCities($provinsiId)
44 {
45     $kota = Kota::Where('province_id', $provinsiId)->pluck('nama_kota', 'city_id');
46     return json_encode($kota);
47 }
48 public function insert()
49 {
50     try{
51         // Validate the required fields
52         Request()->validate([
53             'firstname' => 'required',
54             'handphone' => 'required|numeric',
55             'alamat' => 'required',
56             'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
57             'password' => ['required', 'string', 'min:8'],
58             'ktp' => 'required | mimes:png,jpg,pdf,jpeg',
59             'provinsi' => 'required',
60             'kota' => 'required',
61         ], [
62             'firstname.required' => 'First name harus diisi!',
63             'handphone.required' => 'Nomor HP harus diisi!',
64             'handphone.numeric' => 'Nomor HP harus berupa angka!',
65             'alamat.required' => 'Alamat harus diisi',
66             'email.required' => 'Email harus diisi',
67             'password.required' => 'Password harus diisi',
68             'ktp.required' => 'Ktp harus diisi',
69             'provinsi.required' => 'Provinsi harus diisi',
70             'kota.required' => 'Kota harus diisi',
71         ]);
72     ];
73 }
74
75
76
77
78
79
80     $fileName = null;
81     if (request()->hasFile('img')) {
82         $file = Request()->img;
83         $fileName = Request()->firstname . '.' . $file->extension();
84         $file->move(public_path('foto_profil'), $fileName);
85     }
86
87     if (request()->hasFile('ktp')) {
88         $filekt = Request()->ktp;
89         $fileNamekt = Request()->firstname . '.' . $filekt->extension();
90         $filekt->move(public_path('ktp'), $fileNamekt);
91     }
92
93     // Create a new user record
94     $user = new User();
95     $user->email = Request()->email;
96     $user->password = Hash::make(Request()->password);
97     $user->name = Request()->firstname . ' ' . Request()->lastname;
98     $user->level = 'penjual';
99     $user->save();
100
101    // Create a new penjual record
102    $penjual = new Penjual();
103    $penjual->province_id = Request()->provinsi;
104    $penjual->city_id = Request()->kota;
105    $penjual->users_id = $user->id;
106    $penjual->firstname = Request()->firstname;
107    $penjual->lastname = Request()->lastname;
108    $penjual->handphone = Request()->handphone;
109    $penjual->alamat = Request()->alamat;
110    $penjual->img = $fileName;
111    $penjual->ktp = $fileNamekt;
112
113    // $penjual->img = Request()->img;
114    $penjual->save();
115    Alert::success('Berhasil!', 'Data penjual berhasil disimpan!');
116    return redirect()->route('datapenjual');
117 } catch (\Exception $e) {
118     dd($e);

```

```

119     Log::error("User \"." Gagal menambahkan data");
120     return redirect()->back();
121 }
123
124 public function destroy($id)
125 {
126     DB::table('penjual')->where('id', $id)->delete();
127     Alert::success('Berhasil!', 'Data penjual berhasil dihapus!');
128     return redirect('datapenjual');
129 }
130
131 public function edit($id)
132 {
133     $users = User::all();
134     $ekspedisi = Ekspedisi::pluck('nama', 'code');
135     $provinsi = Provinsi::pluck('nama_provinsi', 'province_id');
136
137     $penjual = Penjual::findOrFail($id); // Retrieve the penjual using the penjual model
138     $kotas = Kota::where('province_id', $penjual->province_id)->pluck('nama_kota', 'city_id');
139     // Retrieve the related user data for the penjual
140     $user = $penjual->user; // Assuming the column name is 'id_users' in the 'penjual' table
141
142     return view('penjual/edit_penjual', compact('penjual', 'user', 'ekspedisi', 'provinsi', 'kotas', 'users'));
143 }
144
145 public function update(Request $request, $id)
146 {
147     // Validate the form data
148     $request->validate([
149         'firstname' => 'required',
150
151         'handphone' => 'required|numeric',
152         'alamat' => 'required',
153         'provinsi' => 'required',
154         'kota' => 'required',
155         'email' => ['required', 'string', 'email', 'max:255'],
156         'password' => ['nullable', 'string', 'min:8'],
157         'img' => 'nullable|mimes:png,jpg,pdf,jpeg',
158
159         'ktp' => 'nullable|mimes:png,jpg,pdf,jpeg',
160     ]);
161
162     // Find the penjual record
163     $penjual = Penjual::findOrFail($id);
164
165     // Update the penjual data
166     $penjual->province_id = $request->provinsi;
167     $penjual->city_id = $request->kota;
168     $penjual->firstname = $request->firstname;
169     $penjual->lastname = $request->lastname;
170     $penjual->handphone = $request->handphone;
171     $penjual->alamat = $request->alamat;
172
173     // Check if the image file is uploaded
174     if ($request->hasFile('img')) {
175         $file = $request->file('img');
176         $fileName = $request->firstname . '.' . $file->extension();
177         $file->move(public_path('foto_profil'), $fileName);
178         $penjual->img = $fileName;
179     }
180
181     if ($request->hasFile('ktp')) {
182         $filekt = $request->file('ktp');
183         $filenamekt = $request->firstname . '.' . $filekt->extension();
184         $filekt->move(public_path('ktp'), $filenamekt);
185         $penjual->ktp = $filenamekt;
186     }
187
188     // Save the updated penjual data
189     $penjual->save();
190
191     // Find the user record associated with the penjual
192     $user = $penjual->user;
193
194     // Update the user email
195     $user->name = $penjual->firstname . ' ' . $penjual->lastname;
196     $user->email = $request->input('email');
197     $user->save();
198
199     // Update the password if provided
200     if ($request->filled('password')) {
201         $user->password = Hash::make($request->input('password'));
202         $user->save();
203     }
204
205     // Redirect to the penjual list page
206     Alert::success('Berhasil!', 'Data penjual berhasil diupdate!');
207     return redirect()->route('datapenjual');
208 }
209

```

## 15. ProdukController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Umkm;
6 use App\Models\Produk;
7 use App\Models\Kategori;
8 use App\Models\Kemasan;
9 use App\Models\Detail_Produk;
10 use App\Models\Imgproduk;
11 use App\Http\Resources\ProdukList;
12 use Illuminate\Support\Facades\Auth;
13 use Illuminate\Support\Facades\Log;
14 use RealRashid\SweetAlert\Facades\Alert;
15 use File;
16 use Illuminate\Support\Facades\Session;
17
18 use Illuminate\Http\Request;
19
20 class ProdukController extends Controller
21 {
22     public function productCreate($id)
23     {
24         try{
25             $umkm = Umkm::find($id);
26             $kategori = Kategori::select('nama_kategori','id')->get();
27             $kemasan = Kemasan::select('ukuran','id')->get();
28
29             Log::info("User ".Auth::user()." Penjual->firstname. ".Auth::user()." Berhasil mengakses halaman tambah produk");
30
31             return view('penjual.manage_product.create', compact('umkm','kategori','kemasan'));
32         } catch (\Exception $e) {
33             return redirect()->back();
34         }
35     }
36
37     public function productStore(Request $request, $id)
38     {
39         try {
40             $validatedData = $request->validate([
41                 'kode_produk.*' => 'required',
42                 'stok.*' => 'required|integer|min:1',
43                 'harga.*' => 'required|integer|min:1000',
44                 'berat_pengiriman.*' => 'required|integer|min:1000',
45             ], [
46                 'stok.*.min' => 'Stok harus lebih dari 0.',
47                 'harga.*.min' => 'Harga harus lebih dari 1000.',
48                 'berat_pengiriman.*.min' => 'Berat pengiriman harus sama atau lebih dari 1000.',
49             ]);
50
51             if (!Produk::where('umkm_id', $id)->whereRaw("UPPER(nama) = '".strtoupper($request->nama)."")->first()) {
52                 $produk = new Produk;
53                 $produk->nama = $request->nama;
54                 $produk->kategori_id = $request->kategori;
55                 $produk->umkm_id = $id;
56                 $produk->no_p_irt = $request->no_p_irt;
57                 $produk->deskripsi = $request->deskripsi;
58
59                 $produk->save();
60
61                 $dir = public_path().'/images/produk';
62                 $files = $request->file('image_product');
63                 if ($files) {
64                     foreach ($files as $file) {
65                         $fileName = time() . '.' . $file->getClientOriginalName();
66                         $file->move($dir, $fileName);
67
68                         $imgproduk = new Imgproduk;
69                         $imgproduk->produk_id = $produk->id;
70                         $imgproduk->image = $fileName;
71                         $imgproduk->save();
72                     }
73                 }
74
75                 $kemasanIds = $request->input('kemasan_id');
76                 $kodeProduks = $request->input('kode_produk');
77                 $stoks = $request->input('stok');
78                 $hargas = $request->input('harga');
```



```

156     $produk = Produk::find($id);
157     // $detail_produk = Detail_Produk::where('produk_id', $produk->id)->first();
158     $kategori = Kategori::select('nama_kategori','id')->get();
159     $kemasan = Kemasan::all();
160     $img_produk = Imgproduk::where('produk_id', $produk->id)->get();
161     $selectedKemasanIds = $produk->Detail_Produk->pluck('kemasan_id')->toArray();
162
163
164     return view('penjual.manage_product.edit', compact('produk','kategori','id','kemasan','img_produk', 'selectedKemasanIds'));
165 } catch (\Exception $e) {
166     dd($e);
167     return redirect()->back();
168 }
169
170
171 public function productUpdate(Request $request, $id)
172 {
173     try {
174         $produk = Produk::find($id);
175         if (!$produk) {
176             return redirect()->back()->with('error', 'Product not found.');
177         }
178
179         // Validate the request data
180         $validatedData = $request->validate([
181             'nama' => 'required',
182             'kategori' => 'required',
183             'no_p_int' => 'required',
184             'deskripsi' => 'required',
185             'kode_produk.*' => 'required',
186             'stok.*' => 'required|integer|min:1',
187             'harga.*' => 'required|integer|min:1000',
188             'berat_pengiriman.*' => 'required|integer|min:1000',
189             // Add validation rules for other fields
190         ]);
191
192
193         // Update the product data
194         $produk->nama = $validatedData['nama'];
195         $produk->kategori_id = $validatedData['kategori'];
196         $produk->no_p_int = $validatedData['no_p_int'];
197         $produk->deskripsi = $validatedData['deskripsi'];
198         // Update other fields as needed
199
200         $produk->save();
201
202
203         // Update product images
204         $dir = public_path('images/produk');
205         $files = $request->file('image_produk');
206
207         if ($files) {
208             foreach ($files as $file) {
209                 $filename = time() . '_' . $file->getClientOriginalName();
210                 $file->move($dir, $filename);
211
212                 $imgproduk = new Imgproduk;
213                 $imgproduk->produk_id = $produk->id;
214                 $imgproduk->image = $filename;
215                 $imgproduk->save();
216             }
217         }
218
219         // Update detail produk
220         $kemasanIds = $request->input('kemasan_id');
221         $kodeProduks = $request->input('kode_produk');
222         $stoks = $request->input('stok');
223         $hargas = $request->input('harga');
224         $beratPengiriman = $request->input('berat_pengiriman');
225
226         if ($kemasanIds && $kodeProduks && $stoks && $hargas && $beratPengiriman) {
227             // Get the existing detail produk for the product
228             $existingDetailProduks = $produk->Detail_Produk->pluck('kemasan_id')->toArray();
229
230             // Delete detail produk that are not selected in the update
231             $deletedDetailProduks = array_diff($existingDetailProduks, $kemasanIds);
232             Detail_Produk::where('produk_id', $produk->id)->whereIn('kemasan_id', $deletedDetailProduks)->delete();

```

```
234 // Update or create detail produk for selected kemasan
235 foreach ($kemasanIds as $index => $kemasanId) {
236     $detailProduk = Detail_Produk::updateOrCreate(
237         ['produk_id' => $produk->id, 'Kemasan_id' => $kemasanId],
238         ['kode_produk' => $kodeProduks[$index], 'stok' => $stoks[$index], 'harga' => $hargas[$index], 'berat_pengiriman' => $beratP
239     );
240 }
241 }
242 Alert::success('Berhasil!', 'Data produk berhasil diubah');
243 return redirect()->route('umkm.show', $produk->umkm_id)->with('success', 'Product updated successfully.');
244 } catch (\Exception $e) {
245     // dd($e);
246     Alert::error('gagal mengedit data produk! Data detail produk tidak sesuai');
247     return redirect()->back()->with('error', 'An error occurred.');
248 }
249 }
250
251 public function destroy($id)
252 {
253     try{
254         $produk = Produk::find($id);
255         $produk->delete();
256
257         Log::notice("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Berhasil menghapus produk");
258         return redirect()->back()->with('success', __('toast.delete.success.message'));
259     } catch (\Exception $e) {
260
261         Log::error("User ".Auth::user()->Penjual->firstname." ".Auth::user()->Penjual->lastname." Gagal menghapus produk");
262         return redirect()->back()->with('success', __('toast.delete.failed.message'));
263     }
264 }
265
266 public function getImage(Request $request)
267 {
268     $data = [];
269     $data = Imgproduk::select('image')
270     ->where('produk_id', $request->produk_id)
271
272     ->where('produk_id', $request->produk_id)
273     ->get();
274     return response()->json($data);
275 }
276
277 public function deleteImage($id)
278 {
279     try {
280         $file = Imgproduk::find($id)
281         $file = Imgproduk::find($id);
282         if($file->exists('images/produk/'.$file->image)){
283             File::delete('images/produk/'.$file->image);
284         }
285         $file->delete();
286
287         return json_encode(array('statusCode'=>200));
288     } catch (\Exception $e) {
289         return redirect()->back()->with('error', __('toast.delete.failed.message'));
290     }
291 }
292
293 public function tambahStok(Request $request, $id)
294 {
295     try {
296         $validatedData = $request->validate([
297             'jumlah_stok' => 'required|integer|min:1',
298             'selected_kemasan' => 'required|exists:kemasan,id',
299         ]);
300
301         $produk = Produk::find($id);
302         $kemasan = Kemasan::find($request->selected_kemasan);
303
304         $detailProduk = Detail_Produk::where('produk_id', $produk->id)
305             ->where('kemasan_id', $request->selected_kemasan)
306             ->first();
307
308         if (!$detailProduk) {
309             // Jika detail produk dengan kemasan yang dipilih belum ada, buat baru
310             $detailProduk = new Detail_Produk();
311             $detailProduk->produk_id = $produk->id;
312
313             $detailProduk->update(['stok' => $validatedData['jumlah_stok']]);
314
315             $detailProduk->save();
316
317             $produk->update(['stok' => $produk->stok + $validatedData['jumlah_stok']]);
318
319             $produk->save();
320
321             Alert::success('Berhasil!', 'Data produk berhasil diubah');
322             return redirect()->route('umkm.show', $produk->umkm_id)->with('success', 'Product updated successfully.');
323         }
324     } catch (\Exception $e) {
325         // dd($e);
326         Alert::error('gagal mengambah stok produk! Data detail produk tidak sesuai');
327         return redirect()->back()->with('error', 'An error occurred.');
328     }
329 }
```

```

310     $detailProduk->kemasan_id = $kemasan->id;
311     $detailProduk->stok = 0;
312     $detailProduk->harga = $produk->harga; // Atur harga sesuai kebutuhan
313 }
314
315 $detailProduk->stok += $request->jumlah_stok;
316 $detailProduk->save();
317 Alert::success('berhasil menambahkan stok produk!');
318 return redirect()->back()->with('success', 'Stok berhasil ditambahkan.');
319 } catch (\Exception $e) {
320     Alert::error('gagal menambahkan stok produk! Data jumlah stok tidak sesuai');
321     return redirect()->back()->with('error', __('toast.delete.failed.message'));
322 }
323
324
325 public function showdetail(Request $request)
326 {
327     $produktId = $request->input('produk_id');
328     $kemasanId = $request->input('kemasan_id');
329
330     $detailProduk = Detail_Produk::where('produk_id', $produktId)
331         ->where('kemasan_id', $kemasanId)
332         ->first();
333
334     if (!$detailProduk) {
335         return response()->json(['error' => 'Detail produk not found'], 404);
336     }
337
338     return response()->json($detailProduk);
339 }
340
341 public function fetchDetail($produktId, $kemasanId)
342 {
343     $detailProduk = Detail_Produk::where('produk_id', $produktId)
344         ->where('kemasan_id', $kemasanId)
345         ->first();
346
347     if ($detailProduk) {
348         return response()->json([
349             'kode_produk' => $detailProduk->kode_produk,
350             'stok' => $detailProduk->stok,
351             'harga' => $detailProduk->harga,
352         ]);
353     }
354
355     return response()->json([]); // Return an empty object if no detail is found
356 }
357 }
358

```

## 16. ProfilController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Profil;
7 use App\Models\Admin;
8 use App\Models\Ekspedisi;
9 use App\Models\Provinsi;
10 use App\Models\Kota;
11 use App\Models\User;
12 use Auth;
13 use RealRashid\SweetAlert\Facades\Alert;
14 use Illuminate\Support\Facades\Hash;
15
16 class ProfilController extends Controller
17 {
18     public function index()
19     {
20         $user = Auth::user();
21         $admin = $user->admin;
22
23         if ($admin) {
24             $ekspedisi = Ekspedisi::pluck('nama', 'code');
25             $provinsis = Provinsi::pluck('nama_provinsi', 'province_id');
26
27             $kotas = Kota::where('province_id', $admin->province_id)->pluck('nama_kota', 'city_id');
28         } else {
29             $ekspedisi = [];
30             $provinsis = [];
31             $kotas = [];
32         }
33         if (request()->hasFile('img')) {
34             $file = request()->file('img');
35             $fileName = Request()->firstname . '.' . $file->extension();
36             $file->move(public_path('foto_profil'), $fileName);
37             $admin->img = $fileName;
38         }
39
40         return view('profil/profiladmin', compact('admin', 'user', 'ekspedisi', 'provinsis', 'kotas'));
41     }
42 }

```

```

41     }
42     public function updateProfile(Request $request)
43     {
44         $user = Auth::user();
45         $admin = $user->admin;
46
47         // Validate the form inputs
48         $request->validate([
49             'img' => 'image|mimes:jpeg,png,jpg|max:2048', // Add validation rules for the image file if required
50             'firstname' => 'required',
51
52             'handphone' => 'required',
53             'provinsi' => 'required',
54             'kota' => 'required',
55             'alamat' => 'required',
56             'email' => 'required|email',
57         ]);
58
59         if ($request->hasFile('img')) {
60             $file = $request->file('img');
61             $fileName = $request->firstname . '.' . $file->extension();
62             $file->move(public_path('foto_profil'), $fileName);
63             $admin->img = $fileName;
64         }
65
66         $admin->firstname = $request->firstname;
67         $admin->lastname = $request->lastname;
68         $admin->handphone = $request->handphone;
69         $admin->province_id = $request->provinsi;
70         $admin->city_id = $request->kota;
71         $admin->alamat = $request->alamat;
72         $user->email = $request->email;
73
74         $admin->save();
75         $user->save();
76
77         Alert::success('Berhasil!', 'Data profil berhasil diupdate!');
78         return redirect()->back();
79     }

80
81     public function changePassword(Request $request)
82     {
83         $request->validate([
84             'current_password' => 'required',
85             'new_password' => 'required|min:8',
86             'confirm_password' => 'required|same:new_password',
87         ], [
88             'current_password.required' => 'Current password is required.',
89             'new_password.required' => 'New password is required.',
90             'new_password.min' => 'New password must be at least 8 characters.',
91             'confirm_password.required' => 'Confirm password is required.',
92             'confirm_password.same' => 'Password confirmation does not match.',
93         ]);
94
95         $user = Auth::user();
96
97         if (Hash::check($request->current_password, $user->password)) {
98             $user->password = Hash::make($request->new_password);
99             $user->save();
100
101             Alert::success('Berhasil!', 'Password berhasil diupdate!');
102             return redirect()->back();
103         } else {
104             Alert::error('Gagal!', 'Current password is incorrect.');
105             return redirect()->back();
106         }
107     }
108 }

```

## 17. ReportController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Umkm;
7 use App\Models\Produk;
8 use App\Models\Pembelian;
9 use DB;
10 use PDF;
11 use Illuminate\Support\Facades\Auth;
12
13 class ReportController extends Controller
14 {
15     /**
16      * Display a listing of the resource.
17      *
18      * @return \Illuminate\Http\Response
19      */
20     public function index()
21     {
22         //
23         try{
24             $umkms = Umkm::where('penjual_id', Auth::user()->penjual->id)
25             ->orderby('id','desc')
26             ->get();
27             return view('penjual.manage_report.index', compact('umkms'));
28         }
29         catch(\Exception $e){
30             return redirect()->back();
31         }
32     }
33
34
35     /**
36      * Show the form for creating a new resource.
37      *
38      * @return \Illuminate\Http\Response
39      */
40     public function preview(Request $request)
41     {
42
43         $umkms = Umkm::whereIn('id', $request->umkm)->get();
44         $umkm_select = Umkm::whereIn('id', $request->umkm)->pluck('nama', 'id');
45         $date = explode("- ",$request->date_range);
46         $data = [
47             'umkms' => $umkms,
48             'date' => $date
49         ];
50
51         $umkm_ids = $request->umkm;
52         $date_range = $request->date_range;
53
54
55
56         return view('penjual.manage_report.view_print', compact('umkms', 'date', 'umkm_ids', 'date_range', 'umkm_select'));
57         // $pdf = PDF::loadView('backend.owner.manage_report.print', $data);
58         // return $pdf->download('itsolutionstuff.pdf');
59
60
61     }
62
63
64
65 }
66
67 public function print(Request $request)
68 {
69     try{
70         $umkms = Umkm::whereIn('id', $request->umkm)->get();
71         $date = explode("- ",$request->date_range);
72         $data = [
73             'umkms' => $umkms,
74             'date' => $date
75         ];
76
77         $pdf = PDF::loadView('penjual.manage_report.print', $data);
78
79         return $pdf->download('Laporan Transaksi'. $date[0]. '-' . $date[1] .'.pdf');
80     }
81     catch(\Exception $e) {
82         return redirect()->back();
83     }
84 }
85
86 }
```

## 18. TransaksiController

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Detail_Pembelian;
6 use Illuminate\Http\Request;
7 use Illuminate\Support\Facades\Auth;
8 use App\Models\Pembelian;
9 use App\Models\Umkm;
10 use App\Models\Produk;
11 use App\Http\Resources\Transaksilist;
12 use App\Models\Detail_Produk;
13
14 class TransaksiController extends Controller
15 {
16     /**
17      * Display a listing of the resource.
18      *
19      * @return \Illuminate\Http\Response
20      */
21     public function index()
22     {
23         //
24         try{
25             if(Auth::user()->level === 'penjual')
26             {
27                 $pembelians = Pembelian::all();
28                 $umkm = Umkm::where('penjual_id', Auth::user()->penjual->id)->where('status',1)->get()->pluck('nama', 'id');
29
30                 return view('penjual.manage_transaction.index', compact('pembelians', 'umkm'));
31             }
32             elseif(Auth::user()->level === 'pembeli')
33             {
34
35             }
36         }
37         catch(\Exception $e){
38             dd($e);
39             return redirect()->back();
40         }
41     }
42 }
43
44 /**
45 * Remove the specified resource from storage.
46 *
47 * @param int $id
48 * @return \Illuminate\Http\Response
49 */
50 public function destroy($id)
51 {
52     try{
53         $pembelian = Pembelian::find($id);
54         $pembelian->delete();
55         return redirect()->route('transaksi.index')->with('success', __('toast.delete.success.message'));
56     } catch (\Exception $e) {
57         return redirect()->back()->with('error', __('toast.delete.failed.message'));
58     }
59 }
60
61 public function getData(Request $request)
62 {
63     $data = [];
64     if($request->data=="all"){
65         $umkms = Umkm::where('penjual_id', $request->penjual_id)->get();
66         $umkm_id = collect([]);
67         foreach($umkms as $umkm){
68             $umkm_id->push($umkm->id);
69         }
70         $produk = Produk::whereIn('umkm_id', $umkm_id)->get();
71         $produk_id = collect([]);
72         foreach($produk as $produk){
73             $produk_id->push($produk->id);
74         }
75         $data = Pembelian::with(['Detail_Pembelian.Produk.Umkm'])
76             ->whereHas('Detail_Pembelian.Produk.Umkm', function ($query) use ($request) {
77                 $query->where('penjual_id', $request->penjual_id);
78             })
79             ->orderBy('created_at', 'desc')
```

```
84
85
86     }
87
88     elseif($request->data=="id"){
89         $data = Pembelian::find($request->id);
90     }
91
92     elseif($request->data=="select"){
93         $id = explode(',',$request->id);
94         $data = Pembelian::whereIn('id', $id)->get();
95     }
96     // dd($data);
97     if($data) return response()->json(Transaksilist::collection($data));
98     return $data;
99 }
100
101 public function confirm($id)
102 {
103     try {
104         $pembelian = Pembelian::find($id);
105         $pembelian->status_pembelian = 1;
106         $pembelian->save();
107
108         // Mengurangi stok produk berdasarkan detail pembelian
109         $detailPembelians = $pembelian->Detail_Pembelian;
110         foreach ($detailPembelians as $detailPembelian) {
111             $produkId = $detailPembelian->produk_id;
112             $jumlah = $detailPembelian->jumlah;
113             $kemasanId = $detailPembelian->kemasan_id;
114
115             // Mengurangi stok pada tabel detail_produk
116             $detailProduk = Detail_Produk::where('produk_id', $produkId)
117                             ->where('kemasan_id', $kemasanId)
118                             ->first();
119
120             if ($detailProduk) {
121                 $detailProduk->stok -= $jumlah;
122                 $detailProduk->save();
123             }
124         }
125
126         $time = strtotime($pembelian->created_at);
127         $created_at = date('d M Y', $time);
128
129         return redirect()->route('transaksi.index')->with('success', __('toast.confirm.success.message'));
130     } catch (\Exception $e) {
131         dd($e);
132         return redirect()->back()->with('error', __('toast.confirm.failed.message'));
133     }
134
135     public function reject(Request $request, $id)
136     {
137         try {
138             $pembelian = Pembelian::find($request->pembelian_id);
139             $pembelian->status_pembelian = 3;
140             $pembelian->reject_note = $request->reject_note;
141             $pembelian->save();
142
143             return redirect()->route('transaksi.index')->with('success', __('toast.reject.success.message'));
144         } catch (\Exception $e) {
145             dd($e);
146             return redirect()->back()->with('error', __('toast.reject.failed.message'));
147         }
148
149     /**
150      * Display the specified resource.
151      *
152      * @param int $id
153      * @return \Illuminate\Http\Response
154      */
155     public function show($id)
156     {
157         try{
158             if(Auth::user()->level === 'pembeli')
159             {
160                 $pembelian = Pembelian::find($id);
161                 return view('pembeli.manage_transaction.show', compact('pembelian'));
162             }
163         }
164     }
165 }
```

```

162     }
163     elseif(Auth::user()->level === 'penjual')
164     {
165         try{
166             $pembelian = Pembelian::find($id);
167             // dd($pembelian);
168
169             return view('penjual.manage_transaction.show', compact('pembelian'));
170         } catch (\Exception $e) {
171             return redirect()->back();
172         }
173     }
174 }
175 catch(\Exception $e){
176     dd($e);
177     return redirect()->back();
178 }
179
180 public function finish($id)
181 {
182     try {
183         $pembelian = Pembelian::find($id);
184         $pembelian->status_pembelian = 4;
185         $pembelian->save();
186
187
188         $time = strtotime($pembelian->created_at);
189         $created_at = date('d M Y', $time);
190
191         return redirect()->route('transaksi.index')->with('success', 'Pesanan telah berakhir');
192     } catch (\Exception $e) {
193         dd($e);
194         return redirect()->back()->with('error', __('toast.reject.failed.message'));
195     }
196 }
197
198 public function inputResi(Request $request, $id)
199 {
200
201     $validatedData = $request->validate([
202         'noresi' => 'required',
203     ]);
204
205
206     // Ambil produk berdasarkan ID
207     $pembelian = Pembelian::find($id);
208     $pembelian->status_pembelian = 2;
209     $pembelian->noresi = $request->noresi;
210     $pembelian->save();
211
212
213
214     return redirect()->back()->with('success', 'noresi berhasil ditambahkan.');
215 }
216
217

```

## 19. KurirController

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use App\Models\Kurir;
7 use App\Models\Ekspedisi;
8 use App\Models\User;
9 use App\Models\Pembelian;
10 use Illuminate\Support\Facades\Hash;
11 use Illuminate\Support\Facades\DB;
12 use Auth;
13 use RealRashid\SweetAlert\Facades\Alert;
14 use Illuminate\Support\Facades\Log;
15
16 class KurirController extends Controller
17 {
18     public function __construct()
19     {
20         $this->Kurir = new Kurir();
21     }
22
23     public function index()
24     {
25         $data = [
26             'kurir' => $this->Kurir->allData(),
27
28         ];
29
30         return view('kurir/datakurir', $data);
31     }
32
33     public function add()
34     {
35         $ekspedisi = Ekspedisi::select('nama','id')->get();
36
37         return view ('kurir/add_kurir', compact('ekspedisi'));
38     }
39

```

```

40 public function insert()
41 {
42     try{
43         // Validate the required fields
44         Request()->validate([
45             'firstname' => 'required',
46             'handphone' => 'required|numeric',
47             'alamat' => 'required',
48             'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
49             'password' => ['required', 'string', 'min:8'],
50             'ekspedisi' => 'required',
51         ],
52         [
53             'firstname.required' => 'First name harus diisi!',
54             'handphone.required' => 'Nomor HP harus diisi!',
55             'handphone.numeric' => 'Nomor HP harus berupa angka!',
56             'alamat.required' => 'Alamat harus diisi!',
57             'email.required' => 'Email harus diisi!',
58             'password.required' => 'Password harus diisi!',
59             'ekspedisi.required' => 'Ekspedisi harus diisi!',
60         ]);
61     );
62 };
63
64
65
66
67     $fileName = null;
68     if (request()->hasFile('img')) {
69         $file = Request()->file;
70         $fileName = Request()->firstname . '.' . $file->extension();
71         $file->move(public_path('foto_profil'), $fileName);
72     }
73
74
75     // Create a new user record
76     $user = new User();
77     $user->email = Request()->email;
78     $user->password = Hash::make(Request()->password);
79
80     $user->name = Request()->firstname . ' ' . Request()->lastname;
81     $user->level = 'kurir';
82     $user->save();
83
84     // Create a new penjual record
85     $kurir = new Kurir();
86     $kurir->ekspedisi_id = Request()->ekspedisi;
87     $kurir->users_id = $user->id;
88     $kurir->firstname = Request()->firstname;
89     $kurir->lastname = Request()->lastname;
90     $kurir->handphone = Request()->handphone;
91     $kurir->alamat = Request()->alamat;
92     $kurir->img = $fileName;
93
94     // $penjual->img = Request()->img;
95     $kurir->save();
96     Alert::success('Berhasil!', 'Data kurir berhasil disimpan!');
97     return redirect()->route('datakurir');
98     } catch (\Exception $e) {
99         dd($e);
100        Log::error("User " . " Gagal menambahkan data");
101        return redirect()->back();
102    }
103
104    public function destroy($id)
105    {
106        DB::table('kurir')->where('id', $id)->delete();
107        Alert::success('Berhasil!', 'Data kurir berhasil dihapus!');
108        return redirect('datakurir');
109    }
110
111    public function edit($id)
112    {
113        $users = User::all();
114        $ekspedisi = Ekspedisi::select('nama','id')->get();
115
116        $kurir = Kurir::findOrFail($id); // Retrieve the penjual using the penjual model
117
118        $user = $kurir->user; // Assuming the column name is 'id_users' in the 'penjual' table
119
120        return view('kurir/edit_kurir', compact('kurir', 'user', 'ekspedisi', 'users'));
121    }
122
123    public function update(Request $request, $id)
124    {
125        // Validate the form data
126        $request->validate([
127            'firstname' => 'required',
128            'handphone' => 'required|numeric',
129            'alamat' => 'required',
130            'ekspedisi' => 'required',
131            'email' => ['required', 'string', 'email', 'max:255'],
132            'password' => ['nullable', 'string', 'min:8'],
133            'img' => 'nullable|mimes:png,jpg,odf,jpeg',
134        ]);
135
136        // Find the penjual record
137        $kurir = Kurir::findOrFail($id);
138
139        // Update the penjual data
140        $kurir->ekspedisi_id = $request->ekspedisi;
141        $kurir->firstname = $request->firstname;
142        $kurir->lastname = $request->lastname;
143        $kurir->handphone = $request->handphone;
144        $kurir->alamat = $request->alamat;
145
146        // Check if the image file is uploaded
147        if ($request->hasFile('img')) {
148            $file = $request->file('img');
149            $fileName = $request->firstname . '.' . $file->extension();
150            $file->move(public_path('foto_profil'), $fileName);
151            $kurir->img = $fileName;
152        }
153
154    }

```

## 2.Bagian Model

### 1. Admin

```
app > Models > Admin.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Support\Facades\DB;
8  use Illuminate\Database\Eloquent\Relations\BelongsTo;
9
10 class Admin extends Model
11 {
12     use HasFactory;
13     protected $table = 'admin';
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array<int, string>
19      */
20     protected $fillable = [
21         'users_id',
22     ];
23
24     public function user(): BelongsTo
25     {
26         return $this->belongsTo(User::class, 'users_id', 'id');
27     }
28
29     public function allData()
30     {
31         return DB::table('admin')
32             ->select('admin.id', 'provinsi.nama_provinsi', 'kota.nama_kota', 'admin.firstname', 'admin.lastname', 'admin.handphone', 'admin.alamat',
33             ->leftJoin('provinsi', 'provinsi.id', '=', 'admin.province_id')
34             ->leftJoin('kota', 'kota.city_id', '=', 'admin.city_id')
35             ->leftJoin('users', 'users.id', '=', 'admin.users_id')
36             ->get();
37     }
38
39     public function addData($dataaa)
40     {
41         DB::table('admin')->insert($dataaa);
42     }
43 }
44
```

### 2. Chat

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Support\Facades\Auth;
8
9  class Chat extends Model
10 {
11     use HasFactory;
12     protected $table = 'chat';
13
14     public function Penjual()
15     {
16         return $this->hasOne(Penjual::class, 'id', 'penjual_id');
17     }
18
19     public function Pembeli()
20     {
21         return $this->hasOne(Pembeli::class, 'id', 'pembeli_id');
22     }
23
24     public function Umkm()
25     {
26         return $this->hasOne(Umkm::class, 'id', 'umkm_id');
27     }
28
29     public function chatDetail()
30     {
31         return $this->hasMany(ChatDetail::class, 'chat_id', 'id');
32     }
33
34     public function lastChat()
35     {
36         $result = ChatDetail::where('chat_id', $this->id)->orderBy('created_at', 'desc')->first();
37         return $result;
38     }
39
40     public function newMessage(){
```

```

41 |     $result = ChatDetail::where('chat_id', $this->id)
42 |             ->where('receiver', Auth::user()->id)
43 |             ->orderBy('created_at', 'desc')
44 |             ->first();
45 |
46 |     return $result;
47 |
48 }
49

```

### 3. ChatDetail

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Support\Facades\Auth;
8
9 class ChatDetail extends Model
10 {
11     use HasFactory;
12     protected $table = 'detailchat';
13
14     public function chat()
15     {
16         return $this->hasOne(Chat::class, 'id', 'chat_id');
17     }
18
19     public function Penjual()
20     {
21         return $this->hasOne(User::class, 'id', 'sender');
22     }
23
24     public function Pembeli()
25     {
26         return $this->hasOne(User::class, 'id', 'sender');
27     }
28
29 }
30
31

```

### 4. Detail\_Metode\_Pembayaran

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Detail_Metode_Pembayaran extends Model
9 {
10     use HasFactory;
11     protected $table = 'detailmetpemb';
12     public function Umkm()
13     {
14         return $this->hasOne(Umkm::class, 'id', 'umkm_id');
15     }
16
17     public function Metode_Pembayaran()
18     {
19         return $this->hasOne(Metode_Pembayaran::class, 'id', 'metodepemb_id');
20     }
21     public function Pembayaran()
22     {
23         return $this->hasMany(Pembayaran::class, 'id', 'pembelian_id', 'id');
24     }
25 }
26

```

## 5. Detail\_Pembelian

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Detail_Pembelian extends Model
9 {
10     use HasFactory;
11     protected $table = 'detailpemb';
12
13
14     public function Pembelian()
15     {
16         return $this->hasOne(Pembelian::class, 'id', 'pembelian_id');
17     }
18
19     public function Produk()
20     {
21         return $this->hasOne(Produk::class, 'id', 'produk_id');
22     }
23     public function Kemasan()
24     {
25         return $this->hasOne(Kemasan::class, 'id', 'kemasan_id');
26     }
27 }
28
```



```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Detail_Produk extends Model
9 {
10     use HasFactory;
11     protected $table = 'detailproduk';
12
13     protected $fillable = [
14         'produk_id',
15         'kemasan_id',
16         'kode_produk',
17         'stok',
18         'harga',
19         'berat_pengiriman',
20     ];
21
22     public function Produk()
23     {
24         return $this->hasOne(Produk::class, 'id', 'produk_id');
25     }
26
27     public function Kemasan()
28     {
29         return $this->hasOne(Kemasan::class, 'id', 'kemasan_id');
30     }
31 }
32
33
```

## 6. Detail\_Produk



## 7. Ekspedisi

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Ekspedisi extends Model
9 {
10     use HasFactory;
11     protected $table = 'ekspedisi';
12     protected $guarded = [];
13
14 }
15
16
```

## 8. History

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class History extends Model
9 {
10     use HasFactory;
11     protected $table = 'history';
12
13     public function Pembelian()
14     {
15         return $this->hasOne(Pembelian::class, 'id', 'pembelian_id');
16     }
17
18     public function Pembeli()
19     {
20         return $this->hasOne(Pembeli::class, 'id', 'pembeli_id');
21     }
22 }
23
```

## 9. Imgproduk

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Imgproduk extends Model
9 {
10     use HasFactory;
11     protected $table = 'imgproduk';
12
13     protected $fillable = ['img'];
14
15     public function Produk()
16     {
17         return $this->hasOne(Produk::class, 'id', 'produk_id');
18     }
19 }
```

## 10. Kategori

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Support\Facades\DB;
8 use Illuminate\Database\Eloquent\Relations\BelongsTo;
9
10 class Kategori extends Model
11 {
12     use HasFactory;
13     protected $table = 'kategori';
14
15     public function allData()
16     {
17         return DB::table('kategori')->get();
18     }
19
20     public function addData($data)
21     {
22         DB::table('kategori')->insert($data);
23     }
24     public function Produk()
25     {
26         return $this->hasMany(Produk::class, 'kategori_id', 'id');
27     }
28 }
29
30
```

## 11. Kemasan

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Kemasan extends Model
9 {
10     use HasFactory;
11     protected $table = 'kemasan';
12
13     public function Detail_Produk()
14     {
15         return $this->hasOne(Detail_Produk::class, 'id', 'kemasan_id');
16     }
17     public function Keranjang()
18     {
19         return $this->hasMany(Keranjang::class, 'produk_id', 'id');
20     }
21     public function Detail_Pembelian()
22     {
23         return $this->hasMany(Detail_Pembelian::class, 'kemasan_id', 'id');
24     }
25 }
```

## 12. Keranjang

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Database\Eloquent\Relations\BelongsTo;
8
9 class Keranjang extends Model
10 {
11     use HasFactory;
12     protected $table = 'keranjang';
13     public function Kemasan()
14     {
15         return $this->hasOne(Kemasan::class, 'id', 'kemasan_id');
16     }
17     public function Produk()
18     {
19         return $this->hasOne(Produk::class, 'id', 'produk_id');
20     }
21     public function user(): BelongsTo
22     {
23         return $this->belongsTo(User::class, 'users_id', 'id');
24     }
25     // Relasi ke tabel detail_produk berdasarkan 'produk_id' dan 'kemasan_id'
26     public function Detail_Produk()
27     {
28         return $this->hasOne(Detail_Produk::class, 'produk_id', 'produk_id')->where('kemasan_id', $this->kemasan_id);
29     }
30 }
31
32 
```

## 13. Kota

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Kota extends Model
9 {
10     use HasFactory;
11     protected $table = 'kota';
12     protected $guarded = [];
13 }
```

## 14. Metode\_Pembayaran

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Metode_Pembayaran extends Model
9 {
10     use HasFactory;
11     protected $table = 'metodepemb';
12     public function Detail_Metode_Pembayaran()
13     {
14         return $this->hasMany(Detail_Metode_Pembayaran::class, 'metodepemb_id', 'id');
15     }
16
17     public function isChecked($umkm_id){
18         $result = Detail_Metode_Pembayaran::where('metodepemb_id', $this->id)
19             ->where('umkm_id', $umkm_id)
20             ->first();
21
22         return $result;
23     }
24 }
25
```

## 15. Pembayaran

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Pembayaran extends Model
9 {
10     use HasFactory;
11     protected $table = 'pembayaran';
12
13     public function Pembelian()
14     {
15         return $this->hasOne(Pembelian::class, 'id', 'pembelian_id');
16     }
17
18     public function Detail_Metode_Pembayaran()
19     {
20         return $this->hasOne(Detail_Metode_Pembayaran::class, 'id', 'detailmetpemb_id');
21     }
22 }
```

## 16. Pembeli

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Support\Facades\DB;
8 use Illuminate\Database\Eloquent\Relations\BelongsTo;
9
10 class Pembeli extends Model
11 {
12     use HasFactory;
13     protected $table = 'pembeli';
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array<int, string>
19      */
20     protected $fillable = [
21         'users_id',
22         'firstname',
23         'lastname',
24     ];
25
26     public function user(): BelongsTo
27     {
28         return $this->belongsTo(User::class, 'users_id', 'id');
29     }
30
31     public function allData()
32     {
33         return DB::table('pembeli')
34             ->select('pembeli.id', 'provinsi.nama_provinsi', 'kota.nama_kota', 'pembeli.firstname', 'pembeli.lastname', 'pembeli.handphone', 'pembeli.province_id')
35             ->leftJoin('provinsi', 'provinsi.id', '=', 'pembeli.province_id')
36             ->leftJoin('kota', 'kota.city_id', '=', 'pembeli.city_id')
37             ->leftJoin('users', 'users.id', '=', 'pembeli.users_id')
38             ->get();
39     }
40 }
```

```

40     }
41     public function chat()
42     {
43         return $this->hasMany(Chat::class, 'pembeli_id', 'id');
44     }
45 }
46

```

## 17. Pembelian

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Pembelian extends Model
9 {
10     use HasFactory;
11     protected $table = 'pembelian';
12
13     public function Detail_Pembelian()
14     {
15         return $this->hasMany(Detail_Pembelian::class, 'pembelian_id', 'id');
16     }
17     public function Pembayaran()
18     {
19         return $this->hasOne(Pembayaran::class, 'pembelian_id', 'id');
20     }
21     public function History()
22     {
23         return $this->hasOne(History::class, 'pembelian_id', 'id');
24     }
25
26
27 }
28

```

## 18. Penjual

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Support\Facades\DB;
8 use Illuminate\Database\Eloquent\Relations\BelongsTo;
9
10 class Penjual extends Model
11 {
12     use HasFactory;
13     protected $table = 'penjual';
14
15     /**
16      * The attributes that are mass assignable.
17      *
18      * @var array<int, string>
19      */
20     protected $fillable = [
21         'users_id',
22         'firstname',
23         'lastname',
24     ];
25
26     public function user(): BelongsTo
27     {
28         return $this->belongsTo(User::class, 'users_id', 'id');
29     }
30
31     public function allData()
32     {
33         return DB::table('penjual')
34             ->select('penjual.id', 'provinsi.nama_provinsi', 'kota.nama_kota', 'penjual.firstname', 'penjual.lastname', 'penjual.handphone', 'penjual.address', 'penjual.city_id', 'penjual.province_id')
35             ->leftJoin('provinsi', 'provinsi.id', '=', 'penjual.province_id')
36             ->leftJoin('kota', 'kota.city_id', '=', 'penjual.city_id')
37             ->leftJoin('users', 'users.id', '=', 'penjual.users_id')
38             ->get();
39     }
40
41     public function Umkm()
42     {
43         return $this->hasMany(Umkm::class, 'penjual_id', 'id');
44     }
45     public function chat()
46     {
47         return $this->hasMany(Chat::class, 'penjual_id', 'id');
48     }
49 }
50

```

## 19. Produk

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Produk extends Model
9 {
10     use HasFactory;
11     protected $table = 'produk';
12
13     protected $fillable = [
14         'nama',
15         'no_p_irt',
16         'deskripsi',
17         'umkm_id',
18         'kategori_id'
19     ];
20
21     public function Umkm()
22     {
23         return $this->hasOne(Umkm::class, 'id', 'umkm_id');
24     }
25
26     public function Kategori()
27     {
28         return $this->belongsTo(Kategori::class, 'kategori_id');
29     }
30     public function Detail_Produk()
31     {
32         return $this->hasMany(Detail_Produk::class, 'produk_id', 'id');
33     }
34     public function Imgproduk()
35     {
36         return $this->hasMany(Imgproduk::class, 'produk_id', 'id');
37     }
38     public function Keranjang()
39     {
40         return $this->hasMany(Keranjang::class, 'produk_id', 'id');
41     }
42     public function Detail_Pembelian()
43     {
44         return $this->hasMany(Detail_Pembelian::class, 'produk_id', 'id');
45     }
46 }
47
```

## 20. Profil

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Profil extends Model
9 {
10     use HasFactory;
11 }
```

## 21. Provinsi

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Provinsi extends Model
9 {
10     use HasFactory;
11     protected $table = 'provinsi';
12     protected $guarded = [];
13 }
```

## 22. User

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Contracts\Auth\MustVerifyEmail;
6 use Illuminate\Database\Eloquent\Factories\HasFactory;
7 use Illuminate\Foundation\Auth\User as Authenticatable;
8 use Illuminate\Notifications\Notifiable;
9 use Laravel\Fortify\TwoFactorAuthenticatable;
10 use Laravel\Jetstream\HasProfilePhoto;
11 use Laravel\Sanctum\HasApiTokens;
12 // use Illuminate\Contracts\Auth\Authenticated;
13
14 class User extends Authenticatable
15 {
16     use HasApiTokens;
17     use HasFactory;
18     use HasProfilePhoto;
19     use Notifiable;
20     use TwoFactorAuthenticatable;
21
22     /**
23      * The attributes that are mass assignable.
24      *
25      * @var array<int, string>
26      */
27     protected $fillable = [
28         'name',
29         'email',
30         'password',
31         'google_id',
32         'level',
33     ];
34
35     /**
36      * The attributes that should be hidden for serialization.
37      *
38      * @var array<int, string>
39      */
40     protected $hidden = [
41         'password',
42         'remember_token',
43         'two_factor_recovery_codes',
44         'two_factor_secret',
45     ];
46
47     /**
48      * The attributes that should be cast.
49      *
50      * @var array<string, string>
51      */
52     protected $casts = [
53         'email_verified_at' => 'datetime',
54     ];
55
56     /**
57      * The accessors to append to the model's array form.
58      *
59      * @var array<int, string>
60      */
61     protected $appends = [
62         'profile_photo_url',
63     ];
64
65     public function admin(){
66         return $this->hasOne(Admin::class, 'users_id');
67     }
68
69     public function penjual(){
70         return $this->hasOne(Penjual::class, 'users_id');
71     }
72     public function pembeli(){
73         return $this->hasOne(Pembeli::class, 'users_id');
74     }
75     public function keranjang(){
76         return $this->hasOne(Keranjang::class, 'users_id');
77     }
78 }
```

## 23. Kurir

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Support\Facades\DB;
8 use Illuminate\Database\Eloquent\Relations\BelongsTo;
9
10 class Kurir extends Model
11 {
12     use HasFactory;
13     protected $table = 'kurir';
14
15     public function user(): BelongsTo
16     {
17         return $this->belongsTo(User::class, 'users_id', 'id');
18     }
19
20     public function Ekspedisi()
21     {
22         return $this->hasOne(Ekspedisi::class, 'id', 'ekspedisi_id');
23     }
24
25     public function allData()
26     {
27         return DB::table('kurir')
28             ->select('kurir.id', 'ekspedisi.nama', 'kurir.firstname', 'kurir.lastname', 'kurir.handphone', 'kurir.alamat', 'kurir.img')
29             ->leftJoin('ekspedisi', 'ekspedisi.id', '=', 'kurir.ekspedisi_id')
30             ->leftJoin('users', 'users.id', '=', 'kurir.users_id')
31             ->get();
32     }
33 }
```

## 3.Bagian View

### a. Bagian Auth

#### 1.Halaman Login



```
1 <x-guest-layout>
2     <x-authentication-card>
3         <x-slot name="logo">
4             
5         </x-slot>
6
7         <x-validation-errors class="mb-4" />
8
9         @if (session('status'))
10             <div class="mb-4 font-medium text-sm text-green-600">
11                 {{ session('status') }}
12             </div>
13         @endif
14
15         <form method="POST" action="{{ route('login') }}>
16             @csrf
17
18             <div>
19                 <x-label for="email" value="{{ __('Email') }}" />
20                 <x-input id="email" class="block mt-1 w-full" type="email" name="email" :value="old('email')" required autofocus autocomplete="off">
21             </div>
22
23             <div class="mt-4">
24                 <x-label for="password" value="{{ __('Password') }}" />
25                 <x-input id="password" class="block mt-1 w-full" type="password" name="password" required autocomplete="current-password" />
26             </div>
27
28             <div class="flex items-center justify-end mt-4">
29                 <a class="underline text-sm text-gray-600 hover:text-gray-900 rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-green-600 py-2 px-4 transition duration-150 ease-in-out">
30                     {{ __('Dont have account?') }}
31                 </a>
32
33                 <x-button class="ml-4">
34                     {{ __('Log in') }}
35                 </x-button>
36             </div>
37             <div class="flex items-center justify-end mt-4">
38                 <a href="{{ url('auth/google') }}>
39                     
40                 </a>
41             </div>
42         </form>
43     </x-authentication-card>
44 </x-guest-layout>
```

## 2. Halaman Register

```
1  <x-guest-layout>
2    <x-authentication-card>
3      <x-slot name="logo">
4        
5      </x-slot>
6
7      <x-validation-errors class="mb-4" />
8
9      <form method="POST" action="{{ route('register') }}>
10     @csrf
11
12     <div>
13       <x-label for="firstname" value="{{ __('First Name') }}" />
14       <x-input id="firstname" class="block mt-1 w-full" type="text" name="firstname" :value="old('firstname')" required autofocus />
15     </div>
16
17     <div class="mt-4">
18       <x-label for="lastname" value="{{ __('Last Name') }}" />
19       <x-input id="lastname" class="block mt-1 w-full" type="text" name="lastname" :value="old('lastname')" />
20     </div>
21
22     <div class="mt-4">
23       <x-label for="email" value="{{ __('Email') }}" />
24       <x-input id="email" class="block mt-1 w-full" type="email" name="email" :value="old('email')" required autocomplete="username" />
25     </div>
26
27
28     <div class="mt-4">
29       <x-label for="password" value="{{ __('Password') }}" />
30       <x-input id="password" class="block mt-1 w-full" type="password" name="password" required autocomplete="new-password" />
31     </div>
32
33
34     <div class="mt-4">
35       <x-label for="password_confirmation" value="{{ __('Confirm Password') }}" />
36       <x-input id="password_confirmation" class="block mt-1 w-full" type="password" name="password_confirmation" required autocomplete="new-password" />
37     </div>
38
39
40     <div class="mt-4">
41       <x-label for="level" value="{{ __('Level') }}" />
42       <select id="level" name="level" class="block mt-1 w-full" type="dropdown" :value="old('level')" required autofocus autocomplete="level">
43         <!-- <x-input id="level" class="block mt-1 w-full" type="text" name="level" :value="old('level')" required autofocus autocomplete="level" /-->
44         <option selected disabled>Pilih Level</option>
45         <option value="penjual">Penjual</option>
46         <option value="pembeli">pembeli</option>
47       </select>
48     </div>
49
50     @if (Laravel\Jetstream\Jetstream::hasTermsAndPrivacyPolicyFeature())
51       <div class="mt-4">
52         <x-label for="terms">
53           <div class="flex items-center">
54             <x-checkbox name="terms" id="terms" required />
55
56             <div class="ml-2">
57               {!! __('I agree to the :terms_of_service and :privacy_policy', [
58                 'terms_of_service' => '<a target="_blank" href="{{ route('terms.show') }}>' . class="underline text-sm text-gray-600 hover:text-gray-900 rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500" href="{{ route('terms.show') }}',
59                 'privacy_policy' => '<a target="_blank" href="{{ route('policy.show') }}>' . class="underline text-sm text-gray-600 hover:text-gray-900 rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500" href="{{ route('policy.show') }}"
60               ]) !!}
61             </div>
62           </div>
63         </x-label>
64       </div>
65     @endif
66
67     <div class="flex items-center justify-end mt-4">
68       <a class="underline text-sm text-gray-600 hover:text-gray-900 rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500" href="{{ route('register') }}>
69         {{ __('Already registered?') }}
70       </a>
71
72       <x-button class="ml-4">
73         {{ __('Register') }}
74       </x-button>
75     </div>
76   </form>
77 </x-authentication-card>
78 </x-guest-layout>
```

## b. Bagian Admin

### 1. Halaman add\_admin

```
1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4     <div class="row">
5         <div class="col-lg-12 grid-margin stretch-card">
6             <div class="card">
7                 <div class="card-body">
8                     <h4 class="card-title">Menambahkan Data Admin</h4>
9                     <form action="/dataadmin/insert" method="POST" class="forms-sample" enctype="multipart/form-data">
10                         @csrf
11                         <div class="form-group">
12                             <label for="exampleInputName1">Firstname</label>
13                             <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ old('firstname') }}>
14                             <div class="text-danger">
15                                 @error('firstname')
16                                     {{ $message }}
17                                 @enderror
18                             </div>
19                         </div>
20                         <div class="form-group">
21                             <label for="exampleInputEmail3">Lastname</label>
22                             <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ old('lastname') }}>
23                             <!-- <div class="text-danger">
24                                 @error('lastname')
25                                     {{ $message }}
26                                 @enderror
27                             </div> -->
28                         </div>
29                         <div class="form-group">
30                             <label for="exampleInputEmail3">No Handphone</label>
31                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ old('handphone') }}>
32                             <div class="text-danger">
33                                 @error('handphone')
34                                     {{ $message }}
35                                 @enderror
36                             </div>
37                         </div>
38                         <div class="form-group">
39                             <label for="">Provinsi</label>
40                             <select class="form-control" name="provinsi" id="provinsi" required>
41                                 <option value="">Provinsi</option>
42                                 @foreach ($provinsi as $provinsi => $value)
43                                     <option value="{{ $provinsi }}" {{ old('provinsi') == $provinsi ? 'selected' : '' }}>
44                                         {{ $value }}
45                                     </option>
46                                 @endforeach
47                             </select>
48                             <div class="text-danger">
49                                 @error('provinsi')
50                                     {{ $message }}
51                                 @enderror
52                             </div>
53                         </div>
54                         <div class="form-group">
55                             <label for="">Kota</label>
56                             <select class="form-control" name="kota" id="kota">
57                                 <option value="">Kota</option>
58                                 @foreach ($kotas as $kotaId => $kotaName)
59                                     <option value="{{ $kotaId }}" {{ old('kota') == $kotaId ? 'selected' : '' }}>
60                                         {{ $kotaName }}
61                                     </option>
62                                 @endforeach
63                             </select>
64                             <div class="text-danger">
65                                 @error('kota')
66                                     {{ $message }}
67                                 @enderror
68                             </div>
69                             <div class="form-group">
70                                 <label for="exampleTextareal">Alamat</label>
71                                 <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ old('alamat') }}>
72                                 <div class="text-danger">
73                                     @error('alamat')
74                                         {{ $message }}
75                                     @enderror
76                             </div>
77                             <div class="form-group">
78                                 <label for="exampleInputEmail3">Email</label>
79                                 <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ old('email') }}>
```

```

80     <div class="text-danger">
81       @error('email')
82         {{ $message }}
83       @enderror
84     </div>
85   </div>
86   <div class="form-group">
87     <label for="exampleInputPassword4">Password</label>
88     <input name="password" type="password" class="form-control" id="password" placeholder="Password" value="{{ old('password') }}"/>
89     <div class="text-danger">
90       @error('password')
91         {{ $message }}
92       @enderror
93     </div>
94   </div>
95   <div class="form-group">
96     <label>Foto Profil</label>
97     <div> </div>
98     <input type="file" name="img" class="form-control" value="{{ old('img') }}"/>
99   </div>
100
101   <button type="submit" class="btn btn-primary mr-2">Submit</button>
102 </form>
103 </div>
104 </div>
105 </div>
106 </div>
107 </div>
108 </div>
109 </div>
110 @endsection
111
112 @section('script')
113   <!-- Include the jQuery library -->
114   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
115   <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
116
117   <script>
118     $(document).ready(function() {
119       $('select[name="provinsi"]').on('change', function() {
120         let provinsiId = $(this).val();
121         if (provinsiId) {
122           jQuery.ajax({
123             url: '/provinsi/' + provinsiId + '/city',
124             type: "GET",
125             dataType: "json",
126             success: function(data) {
127               $('select[name="kota"]').empty();
128               $.each(data, function(key, value) {
129                 $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
130               });
131             },
132             error: function(xhr, status, error) {
133               console.error(error);
134             }
135           });
136         } else {
137           $('select[name="kota"]').empty();
138         }
139       });
140     });
141   </script>
142
143
144
145
146 @endsection

```



## 2. Halaman dataadmin

```
1 @extends('layouts.admin')
2 @section('content')
3     <div class="content-wrapper">
4         <div class="row">
5             <div class="col-lg-12 grid-margin stretch-card">
6                 <div class="card">
7                     <div class="card-body">
8                         <h4 class="card-title">Data Admin</h4>
9                         <a href="/dataadmin/add" type="button" class="btn btn-primary">Add</a>
10                        @if (session('message'))
11                            {{ session('message') }}
12                        @endif
13                        <!-- <p class="card-description">
14                            Add class <code>.table-striped</code>
15                        </p> -->
16                        <div class="table-responsive">
17                            <table class="table table-striped">
18                                <thead>
19                                    <tr>
20                                        <th>No</th>
21                                        <th>Nama</th>
22                                        <th>No Hp</th>
23                                        <th>Provinsi</th>
24                                        <th>Kota</th>
25                                        <th>Alamat</th>
26                                        <th>Foto Profil</th>
27                                        <th>Aksi</th>
28                                    </tr>
29                                </thead>
30                                <tbody>
31                                    @foreach ($admin as $ad)
32
33                                    <tr>
34                                        <td scope="row">{{ $loop->iteration }}</td>
35                                        <td>@php
36                                            echo $ad->firstname . ' ' . $ad->lastname;
37                                            @endphp</td>
38                                        <td>{{ $ad->handphone }}</td>
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
    @endforeach
        <td>{{ $ad->handphone }}</td>
        <td>{{ $ad->nama_provinsi }}</td>
        <td>{{ $ad->nama_kota }}</td>
        <td>{{ $ad->alamat }}</td>
        <!-- <td>
            <a href="{{ asset('foto_profil/' . $ad->img) }}" target="_blank" rel="noopener noreferrer"> {{ $ad->img }} </a>
        </td> -->
        <td>
            <div class="d-flex">
                <a href="/dataadmin/edit/{{ $ad->id }}" class="btn btn-sm btn-primary mr-2">Edit</a>
                <button type="button" class="btn btn-danger" data-toggle="modal" data-target="#deleteModal{{ $ad->id }}>Hapus</button>
            </div>
        </td>
    <div class="modal fade" id="deleteModal{{ $ad->id }}" tabindex="-1" role="dialog" aria-labelledby="deleteModalLabel{{ $ad->id }}"
        <div class="modal-dialog" role="document">
            <div class="modal-content">
                <div class="modal-header">
                    <h5 class="modal-title" id="deleteModalLabel{{ $ad->id }}>Konfirmasi Hapus</h5>
                    <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                        <span aria-hidden="true">&times;</span>
                    </button>
                </div>
                <div class="modal-body">
                    Apakah kamu yakin ingin menghapus data?
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
                    <form action="{{ route('dataadmin.destroy', $ad->id) }}" method="POST">
                        @csrf
                        @method('DELETE')
                        <button type="submit" class="btn btn-danger">Hapus</button>
                    </form>
                </div>
            </div>
        </div>
    </div>
    </div>
    </div>
    </tr>
    @endforeach
```

### 3. Halaman edit admin

```
1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4   <div class="row">
5     <div class="col-lg-12 grid-margin stretch-card">
6       <div class="card">
7         <div class="card-body">
8           <h4 class="card-title">Mengubah Data Admin</h4>
9           <form action="/dataadmin/update/{{ $admin->id }}" method="POST" class="forms-sample" enctype="multipart/form-data">
10             @csrf
11             <div class="form-group">
12               <label for="exampleInputName1">Firstname</label>
13               <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $admin->firstname }}">
14               <div class="text-danger">
15                 @error('firstname')
16                   {{ $message }}
17                 @enderror
18               </div>
19             </div>
20             <div class="form-group">
21               <label for="exampleInputEmail3">Lastname</label>
22               <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $admin->lastname }}">
23               <div class="text-danger">
24                 @error('lastname')
25                   {{ $message }}
26                 @enderror
27               </div>
28             </div>
29             <div class="form-group">
30               <label for="exampleInputEmail3">No Handphone</label>
31               <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $admin->handphone }}">
32               <div class="text-danger">
33                 @error('handphone')
34                   {{ $message }}
35                 @enderror
36               </div>
37             </div>
38             <div class="form-group">
39               <label for="Provinsi">Provinsi</label>
40               <select class="form-control" name="provinsi" id="provinsi" required>
```

```

41         <option value="">Provinsi</option>
42         @foreach ($provinsis as $provinsi => $value)
43             <option value="{{ $provinsi }}" {{ $admin->province_id == $provinsi ? 'selected' : '' }}> {{ $value }}</option>
44         @endforeach
45     </select>
46     <div class="text-danger">
47         @error('provinsi')
48             {{ $message }}
49         @enderror
50     </div>
51     </div>
52     <div class="form-group">
53         <label for="Kota">
54             <select class="form-control" name="kota" id="kota" >
55                 <option value="">Kota</option>
56                 @foreach ($kotas as $kota => $value)
57                     <option value="{{ $kota }}" {{ $admin->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
58                 @endforeach
59             </select>
60             <div class="text-danger">
61                 @error('kota')
62                     {{ $message }}
63                 @enderror
64             </div>
65             <div class="form-group">
66                 <label for="exampleTextareal">Alamat</label>
67                 <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $admin->alamat }}">
68                 <div class="text-danger">
69                     @error('alamat')
70                         {{ $message }}
71                     @enderror
72                 </div>
73             </div>
74             <div class="form-group">
75                 <label for="exampleInputEmail3">Email</label>
76                 <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}">
77                 <div class="text-danger">
78                     @error('email')
79                         {{ $message }}
80                     @enderror
81                 </div>
82             </div>
83             <div class="form-group">
84                 <label for="password">Password</label>
85                 <input type="password" class="form-control" name="password" id="password" value="">
86                 <small class="form-text text-muted">Leave it blank if you don't want to change the password.</small>
87             </div>
88             <div class="form-group">
89                 <label>Foto Profil</label>
90                 <div> </div>
91                 <input type="file" name="img" class="form-control" value="{{ $admin->img }}">
92                 <div>
93                     <a href="{{ asset('foto_profil/'. $admin->img )}}" target="_blank" rel="noopener noreferrer"> {{ $admin->img }} </a> </div>
94                     <div class="text-danger">
95                         @error('img')
96                             {{ $message }}
97                         @enderror
98                     </div>
99                 </div>
100                <button type="submit" class="btn btn-primary mr-2">Submit</button>
101            </form>
102        </div>
103    </div>
104    </div>
105 </div>
106 </div>
107 </div>
108
109 @endsection
110

```

```

111  <!-- Include the jQuery library -->
112  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
113  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
114
115  <script>
116  $(document).ready(function() {
117    $('select[name="provinsi"]').on('change', function() {
118      let provinceId = $(this).val();
119      if (provinceId) {
120        jQuery.ajax({
121          url: '/province/' + provinceId + '/cities',
122          type: "GET",
123          dataType: "json",
124          success: function(data) {
125            $('select[name="kota"]').empty();
126            $.each(data, function(key, value) {
127              $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
128            });
129          },
130          error: function(xhr, status, error) {
131            console.error(error);
132          }
133        });
134      } else {
135        $('select[name="kota"]').empty();
136      }
137    });
138  });
139 </script>

```

#### 4. Halaman index manage umkm

```

1 @extends('layouts.admin')
2
3 @section('content')
4 <div class="content-wrapper">
5 <section class="content">
6   <div class="container-fluid">
7     <div class="row">
8       @include('admin.manage_umkm.reject')
9       <div class="col-12">
10         <div class="card">
11           <div class="card-header">
12             <h3 class="card-title">List Data Ummk Perlu Konfirmasi</h3>
13           <!-- /.card-header -->
14           <div class="card-body">
15             <table class="table table-striped dataTables-venue-1" id="table-2">
16               <thead>
17                 <tr>
18                   <th style="width: 15%">Nama Ummk</th>
19                   <th style="width: 15%">Pemilik</th>
20                   <th style="width: 25%">Alamat</th>
21                   <th style="width: 20%">Tanggal Pembuatan</th>
22                   <th style="width: 25%">Action</th>
23                 </tr>
24               </thead>
25               <tfoot>
26                 <tr>
27                   <th style="width: 15%">Nama Ummk</th>
28                   <th style="width: 15%">Pemilik</th>
29                   <th style="width: 25%">Alamat</th>
30                   <th style="width: 20%">Tanggal Pembuatan</th>
31                   <th style="width: 25%">Action</th>
32                 </tr>
33               </tfoot>
34             </table>
35           </div>
36           <!-- /.card-body -->
37         </div>
38         <!-- /.card -->
39       </div>
40     </div>

```

```

41 |         </div>
42 |         <!-- /.col -->
43 |     </div>
44 |     <!-- /.row -->
45 |   <!-- /.container-fluid -->
46 | </section>
47 | </div>
48 |
49 |
50 |
51 {!! Form::open(['method'=>'PATCH', 'route' => ['admin.umkm.confirm', 0], 'style' =>
52 'display:none','id=>'>'confirm']) !!}
53 {!! Form::close() !!}
54 |
55 |
56 @endsection
57
58 @section('script')
59 <script>
60 | $(document).ready(function () {
61 |     $('#dataTables-venue-1').DataTable({
62 |
63 |         responsive: true,
64 |         searchDelay: 500,
65 |         processing: true,
66 |         // serverSide: true,
67 |         ajax: {
68 |             url: "{{url('api/umkm')}}?data=all&&status=0",
69 |             dataSrc: ''
70 |         },
71 |         columns: [
72 |             {
73 |                 data: 'nama'
74 |             },
75 |             {
76 |                 data: 'penjual'
77 |             },
78 |             {
79 |                 data: 'alamat'
80 |             },
81 |             {
82 |                 data: 'date'
83 |             },
84 |             {
85 |                 data: 'id',
86 |                 responsivePriority: -1
87 |             },
88 |         ],
89 |         columnDefs: [
90 |             {
91 |                 targets: -1,
92 |                 title: "Action",
93 |                 orderable: false,
94 |                 render: function (data, type, full, meta) {
95 |                     var base = "{{url('/')}}";
96 |                     return '' +
97 |                         ` <a class="btn btn-icon btn-sm btn-dark" href="/umkm/` +
98 |                         data + `/show-index"><i class="fa fa-info"></i></a>` +
99 |                         ` <a class="btn btn-icon btn-sm btn-success" onclick="confirm(` +
100 |                         data +
101 |                         `)` href="javascript:void(0)"><i class="fa fa-check"></i></a>` +
102 |                         ` <a class="btn btn-icon btn-sm btn-danger" onclick="reject(` +
103 |                         data +
104 |                         `)` href="javascript:void(0)"><i class="fa fa-times"></i></a>` +
105 |                         ``;
106 |                 },
107 |             },
108 |         ],
109 |     });
110 |     table.on('order.dt search.dt', function () {
111 |         table.column(0, {
112 |             search: 'applied',
113 |             order: 'applied'
114 |         }).nodes().each(function (cell, i) {
115 |             cell.innerHTML = i + 1;
116 |         });
117 |     }).draw();
118 |

```

```

119 });
120
121 function confirm(id) {
122     swal({
123         title: "Apakah kamu yakin?",
124         text: "Venue yang telah dikonfirmasi akan dibebaskan untuk beroperasi pada aplikasi ini",
125         type: "warning",
126         showCancelButton: true,
127         confirmButtonColor: "#5cc744",
128         confirmButtonText: "Ya, konfirmasi!",
129         cancelButtonText: "Tidak, batalkan!",
130         closeOnConfirm: false,
131         closeOnCancel: false
132
133     },
134     function (isConfirm) {
135         if (isConfirm) {
136             swal("Konfirmasi!", "Umkm telah dapat beroperasi pada aplikasi ini.", "success");
137             $('#confirm').attr('action', '/umkm/' + id + '/confirm');
138             $('#confirm').submit();
139         } else {
140             swal("Batal", "Kamu batal melakukan konfirmasi venue", "error");
141         }
142     });
143 }
144
145
146 function reject1(id) {
147     console.log(id);
148     swal({
149         title: "Apakah kamu yakin?",
150         text: "Permohonan venue ini akan ditolak dan pemilik venue akan diminta untuk melengkapi datanya kembali",
151         type: "warning",
152         showCancelButton: true,
153         confirmButtonColor: "#D06B55",
154         confirmButtonText: "Ya, Tolak!",
155         cancelButtonText: "Tidak, batalkan!",
156         closeOnConfirm: false,
157         closeOnCancel: false
158
159     },
160     function (isConfirm) {
161         if (isConfirm) {
162             swal("Ditolak!", "Pemilik venue akan diminta untuk melengkapi data venuenya kembali.", "success");
163             $('#reject').attr('action', '/umkm/' + id + '/reject');
164             $('#reject').submit();
165         } else {
166             swal("Batal", "Kamu batal melakukan penolakan konfirmasi venue", "error");
167         }
168     });
169 }
170
171 function reject(id) {
172     $('#reject-venue').toggle();
173     $('#umkm_id').val(id).change();
174 }
175
176 </script>
177 @endsection

```

## 5. Halaman *reject manage umkm*

```

1  {!! Form::open(['method'=>'PATCH', 'route' => ['admin.umkm.reject', 0], 'id'=>'reject']) !!}
2  <div class="col-12" animated fadeInRight" id="reject-venue" style="display:none">
3      <div class="card">
4          <div class="card-header">
5              <h3 class="card-title">Alasan penolakan umkm</h3>
6          </div>
7          <!-- /.card-header -->
8          <div class="card-body">
9              <div class="row">
10                 <div class="col-lg-12">
11                     <textarea name="reject_note" class="form-control" cols="200" rows="4" required></textarea>
12                     <input type="hidden" name="umkm_id" id="umkm_id">
13                 </div>
14             </div>
15             <div class="row">
16                 <div class="col-lg-12 d-flex justify-content-end">
17                     <button class="btn btn-primary" type="submit">Kirim alasan tolak</button>
18                 </div>
19             </div>
20         </div>
21         <!-- /.card-body -->
22     </div>
23     <!-- /.card -->
24 </div>
25 {!! Form::close() !!}

```

## 6. Halaman show umkm

```
1  @extends('layouts.admin')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQhgGwzxH7pPCaAQC
5  <link rel="stylesheet" href="{{ asset('templates/css/components.css') }}>
6  <link href="{{asset('templates/js/plugins/dataTables.datatables.bundle.css')}}" rel="stylesheet" type="text/css" />
7
8  <link href="https://cdn.datatables.net/1.11.4/css/jquery.dataTables.min.css" rel="stylesheet">
9  @endsection
10
11 @section('content')
12 <script type="text/javascript"
13 |   src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDQTpXj82d8UpC197wzo_nKXL7nYrd4G70"></script>
14 <div class="content-wrapper">
15
16 <section class="content">
17   <div class="container-fluid">
18     <div class="row">
19       <div class="col-12">
20         <div class="card">
21           <!-- /.card-header -->
22           <div class="card-body">
23             <div class="row">
24               <div class="col-md-12">
25                 <a href="#" class="btn btn-primary btn-icon icon-left btn-lg btn-block mb-4 d-md-none"
26                   data-toggle-slide="#ticket-items">
27               </a>
28               <div class="tickets">
29                 <div class="ticket-items" id="ticket-items">
30                   <div class="gallery gallery-fw" data-item-height="400">
31                     <strong>
32                       <center>Surat Izin Usaha</center>
33                     </strong>
34                     @if ($umkms->imb == NULL)
35                       <div class="gallery-item"
36                         | data-image="{{ asset('templates/img/no_image.jpg') }}">
37                     </div>
38                     @else
39                       <div class="gallery-item"
40                         | data-image="{{ asset('imb/'.$umkms->imb) }}">
41                         @endif
42                     </div>
43                   </div>
44                 </div>
45               <div class="ticket-content">
46                 <div class="ticket-header">
47                   <div class="ticket-sender-picture img-shadow">
48                     @if ($umkms->penjual->img == NULL)
49                       
50                     @else
51                       
52                     @endif
53                   </div>
54                   <div class="ticket-detail">
55                     <div class="ticket-title">
56                       <h4>{{$umkms->penjual->firstname}} {{$umkms->penjual->lastname}}</h4>
57                     </div>
58                     <div class="ticket-info">
59                       <div class="font-weight-600">Penjual {{$umkms->nama}}</div>
60                       <div class="bullet"></div>
61                     </div>
62                   </div>
63                 </div>
64               <div class="ticket-description">
65                 <h6><b>No Handphone:</b></h6>
66                 <p>{{($umkms->penjual->handphone)}}</p>
67                 <h6><b>Alamat:</b></h6>
68                 <p>{{($umkms->alamat)}}</p>
69                 <h6><b>Foto KTP:</b></h6>
70                 <p>
71                   <figure>
72                     </div>
77             </div>
78           </div>
79         </div>
80       </div>
81     </div>
82   </section>
83 </div>
```

```

80 |         <div class="ticket-form">
81 |             </div>
82 |         </div>
83 |     </div>
84 |     </div>
85 |     </div>
86 |     </div>
87 |     <!-- /.card-body -->
88 | </div>
89 | <!-- /.card -->
90 | </div>
91 | <!-- /.col -->
92 | </div>
93 | <div class="row">
94 |     <div class="col-12">
95 |         <div class="card">
96 |             <div class="card-header">
97 |                 <h2 class="card-title">{$umkms->nama}</h2>
98 |             </div>
99 |             <!-- /.card-header -->
100 |             <div class="card-body">
101 |                 <div class="row">
102 |                     <div class="col-md-12">
103 |                         <a href="#">
104 |                             class="btn btn-primary btn-icon icon-left btn-lg btn-block mb-4 d-md-none"
105 |                             data-toggle-slide="#ticket-items"
106 |                         </a>
107 |                         <div class="tickets">
108 |                             <div class="ticket-items" id="ticket-items">
109 |                                 <div class="gallery gallery-fw" data-item-height="150">
110 |                                     <div class="gallery-item"
111 |                                         data-image="{{ asset('img_umkm/'.$umkms->img) }}"
112 |                                         data-title="Profil Umkm"
113 |                                     </div>
114 |                                 </div>
115 |                                 <div class="google-map" id="map-show" style="height:200px"></div>
116 |                             </div>
117 |                         </div>
118 |                         <div class="ticket-content">
119 | 
120 | 
121 | 
122 | 
123 | 
124 | 
125 | 
126 | 
127 | 
128 | 
129 | 
130 | 
131 | 
132 | 
133 | 
134 | 
135 | 
136 | 
137 | 
138 | 
139 | 
140 | 
141 | 
142 | 
143 | 
144 | 
145 | 
146 | 
147 | 
148 | 
149 | 
150 | 
151 | 
152 | 
153 | 
154 | 
155 | 
156 | 
157 |

```

```

158
159
160
161
162
163
164
165
166
167
168     </table>
169     </div>
170     <div class="col-lg-6">
171
172         </div>
173
174     </div>
175
176     <div class="ticket-divider"></div>
177
178     <div class="ticket-form">
179
180         </div>
181
182         <!-- /.card-body -->
183
184         <!-- /.card -->
185
186         <!-- /.col -->
187
188     </div>
189
190     <!-- /.row -->
191
192     <!-- /.container-fluid -->
193
194 </section>
195
196 @endsection
197
198 @section('script')
199 <!-- General JS Scripts -->
200 <!-- Masukkan file jQuery -->
201 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
202
203 <!-- Masukkan file Bootstrap JavaScript -->
204 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL
205 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
206 <script src="https://code.jquery.com/jquery-3.3.1.min.js"
207     integrity="sha256-fGPcbKJQlEJMI3qpcR3I9pEYzuBfJIDqOxqaa8=" crossorigin="anonymous"></script>
208 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
209     integrity="sha384-UO2eT0CpHqdSJQ6nJty5kVphtPhzWj9W01c1HTNg3JDZwmQq4sF86dIHNDz0W1" crossorigin="anonymous">
210 </script>
211 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
212     integrity="sha384-JjSmVgyd0p3XB1RibZUAvIoIy6OrQ6VrjIaEfff/nJGzIxDsF4x0xIM+B07jRM" crossorigin="anonymous"></script>
213 <script src="https://cdn.jsdelivr.net/gh/afollestad/smooth-scroll@11.1.1/dist/smooth-scroll.min.js"></script>
214
215 <script src="{{ asset('templates/js/scripts.js') }}"></script>
216 <script type="text/javascript">
217
218     function deleteUmkm(id) {
219         swal({
220             title: "Apakah kamu yakin?",
221             text: "Data ini akan terhapus total",
222             type: "warning",
223             showCancelButton: true,
224             confirmButtonColor: "#ff2929",
225             confirmButtonText: "Ya, Hapus Data!",
226             cancelButtonText: "Tidak, batalkan!",
227             closeOnConfirm: false,
228             closeOnCancel: false
229
230         })
231     }

```

```

232     function (isConfirm) {
233         if (isConfirm) {
234             swal("Data Terhapus!", "Data tersebut telah terhapus", "success");
235             $('#deleted_umkm').attr('action', '{route('umkm.index')}/' + id);
236             $('#deleted_umkm').submit();
237         } else {
238             swal("Batal", "Anda batal menghapus data", "error");
239         }
240     });
241 }
242
243
244 let mapCreate;
245 let mapShow;
246 let markers = [];
247 // When the window has finished loading google map
248 google.maps.event.addDomListener(window, 'load', init);
249
250 function init() {
251     // More info see: https://developers.google.com/maps/documentation/javascript/reference#MapOptions
252     var mapOptions1 = {
253         zoom: 12,
254         center: new google.maps.LatLng(-0.911111111111111, 100.3497222222221),
255         // Style for Google Maps
256         styles: [
257             {
258                 "featureType": "water",
259                 "stylers": [
260                     "saturation": 43
261                 ],
262                 "lightness": -11
263             },
264             {
265                 "hue": "#0088ff"
266             }
267         ],
268         {
269             "featureType": "road",
270             "elementType": "geometry.fill",
271             "stylers": [
272                 "hue": "#ff0000"
273             ],
274             "saturation": -100
275         },
276         {
277             "lightness": 99
278         }
279     },
280     {
281         "featureType": "road",
282         "elementType": "geometry.stroke",
283         "stylers": [
284             "color": "#800000"
285         ],
286         "lightness": 54
287     },
288     {
289         "featureType": "landscape.man_made",
290         "elementType": "geometry.fill",
291         "stylers": [
292             "color": "#ece2d9"
293         ]
294     },
295     {
296         "featureType": "poi.park",
297         "elementType": "geometry.fill",
298         "stylers": [
299             "color": "#ccdcac"
300         ]
301     },
302     {
303         "featureType": "road",
304         "elementType": "labels.text.fill",
305         "stylers": [
306             "color": "#767676"
307         ]
308     },
309     {
310         "featureType": "road",
311         "elementType": "labels.text.stroke",
312         "stylers": [
313             "color": "#ffffff"
314         ]
315     },
316     {
317         "featureType": "poi",
318         "stylers": [
319             "visibility": "off"
320         ]
321     }
322 }

```

```

312     },
313     {
314       "featureType": "landscape.natural",
315       "elementType": "geometry.fill",
316       "stylers": [
317         {
318           "visibility": "on"
319         }
320       ],
321       "featureType": "poi.park",
322       "stylers": [
323         {
324           "visibility": "on"
325         }
326       ],
327       "featureType": "poi.sports_complex",
328       "stylers": [
329         {
330           "visibility": "on"
331         }
332       ],
333       "featureType": "poi.medical",
334       "stylers": [
335         {
336           "visibility": "on"
337         }
338       ],
339       "featureType": "poi.business",
340       "stylers": [
341         {
342           "visibility": "simplified"
343         }
344       ]
345     }
346   );
347
348   // Get all html elements for map
349   var mapElement3 = document.getElementById('map-show');
350
351   // Create the Google Map using elements
352   var map = new google.maps.Map(mapElement3, mapOptions1);
353
354   // Variabel untuk menyimpan batas kordinat
355   bounds = new google.maps.LatLngBounds();
356
357   $.ajax({
358     url: "{{url('api/umkm/get-location')}}?id={{$umkms->id}}",
359     dataType: 'json',
360     cache: false,
361     dataSrc: '',
362
363     success: function (data) {
364       var latitude = data.map(function (item) {
365         return item.latitude;
366       });
367       var longitude = data.map(function (item) {
368         return item.longitude;
369       });
370       var lating = new google.maps.LatLng(parseFloat(latitude), parseFloat(longitude));
371       console.log(latitude);
372       console.log(longitude);
373       for (i = 0; i < data.length; i++) {
374         var pos = {
375           lat: parseFloat(latitude[i]),
376           lng: parseFloat(longitude[i])
377         };
378         var marker = new google.maps.Marker({
379           position: pos,
380           map: map,
381           title: 'Lokasi Anda',
382           icon: 'http://maps.google.com/mapfiles/ms/icons/green-dot.png',
383           draggable: true,
384           animation: google.maps.Animation.DROP
385         });
386         marker.setMap(map);
387         map.setCenter(lating);
388       }
389     });
390   }
391   </script>
392 @endsection
393

```

## 7. Halaman add\_kategori

```
1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4   <div class="row">
5     <div class="col-lg-12 grid-margin stretch-card">
6       <div class="card">
7         <div class="card-body">
8           <h4 class="card-title">Menambahkan Data Kategori</h4>
9           <form action="/kategori/insert" method="POST" class="forms-sample" enctype="multipart/form-data">
10             @csrf
11             <div class="form-group">
12               <label for="exampleInputName1">Nama Kategori</label>
13               <input name="nama_kategori" type="text" class="form-control" id="nama_kategori" placeholder="nama kategori" value="{{ old('nama_kategori') }}"/>
14               <div class="text-danger">
15                 @error('firstname')
16                   {{ $message }}
17                 @enderror
18               </div>
19             </div>
20
21             <button type="submit" class="btn btn-primary mr-2">Submit</button>
22           </form>
23         </div>
24       </div>
25     </div>
26   </div>
27 </div>
28 </div>
29
30 @endsection
31
32 | | | | | <!-- Include the jQuery library -->
33 | | | | | <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
34 | | | | | <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
35
36
```

## 8. Halaman edit\_kategori

```
1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4   <div class="row">
5     <div class="col-lg-12 grid-margin stretch-card">
6       <div class="card">
7         <div class="card-body">
8           <h4 class="card-title">Mengubah Data Kategori</h4>
9           <form action="/kategori/update/{{ $kategori->id }}" method="POST" class="forms-sample" enctype="multipart/form-data">
10             @csrf
11             <div class="form-group">
12               <label for="exampleInputName1">Nama Kategori</label>
13               <input name="nama_kategori" type="text" class="form-control" id="nama_kategori" placeholder="nama kategori" value="{{ $kategori->nama_kategori }}"/>
14               <div class="text-danger">
15                 @error('firstname')
16                   {{ $message }}
17                 @enderror
18               </div>
19             </div>
20
21             <button type="submit" class="btn btn-primary mr-2">Submit</button>
22           </form>
23         </div>
24       </div>
25     </div>
26   </div>
27 </div>
28 </div>
29
30 @endsection
31
32 | | | | | <!-- Include the jQuery library -->
33 | | | | | <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
34 | | | | | <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
35
36
```

## 9. Halaman datakategori

```
1  @extends('layouts.admin')
2  @section('content')
3  <div class="content-wrapper">
4      <div class="row">
5          <div class="col-lg-12 grid-margin stretch-card">
6              <div class="card">
7                  <div class="card-body">
8                      <h4 class="card-title">Data Kategori</h4>
9                      <a href="/kategori/add" type="button" class="btn btn-primary">Add</a>
10                     @if (session('message'))
11                         <div class="alert alert-success">
12                             {{ session('message') }}
13                         </div>
14                     @endif
15                     <!-- <p class="card-description">
16                         Add class <code>.table-striped</code>
17                     </p> -->
18                     <div class="table-responsive">
19                         <table class="table table-striped">
20                             <thead>
21                                 <tr>
22                                     <th>No</th>
23                                     <th>Nama Kategori</th>
24                                     <th>aksi</th>
25                                 </tr>
26                             </thead>
27                             <tbody>
28                                 @foreach ($kategori as $kt)
29
30                                     <tr>
31                                         <td scope="row">{{ $loop->iteration }}</td>
32                                         <td>{{ $kt->nama_kategori }}</td>
33                                         <td>
34                                             <div class="d-flex">
35                                                 <a href="/kategori/edit/{{ $kt->id }}" class="btn btn-sm btn-primary mr-2">Edit</a>
36                                                 <button type="button" class="btn btn-danger" data-toggle="modal" data-target="#deleteModal{{ $kt->id }}">Hapus</button>
37                                             </div>
38                                         </td>
39                                     </tr>
40
41                                     <div class="modal fade" id="deleteModal{{ $kt->id }}" tabindex="-1" role="dialog" aria-labelledby="deleteModalLabel{{ $kt->id }}"
42                                         >
43                                         <div class="modal-dialog" role="document">
44                                             <div class="modal-content">
45                                                 <div class="modal-header">
46                                                     <h5 class="modal-title" id="deleteModalLabel{{ $kt->id }}">Konfirmasi Hapus</h5>
47                                                     <button type="button" class="close" data-dismiss="modal" aria-label="Close">
48                                                         <span aria-hidden="true">&times;
```

```

75 <script src="https://cdnjs.cloudflare.com/ajax/libs/sweetalert/2.1.0/sweetalert.min.js"></script>
76 <script>
77     $('.show_confirm').click(function(event) {
78         var form = $(this).closest("form");
79         var name = $(this).data("name");
80         event.preventDefault();
81         swal({
82             title: "Apakah Anda Yakin?",
83             text: "Setelah dihapus, Anda tidak akan dapat memulihkan data ini!",
84             icon: "warning",
85             buttons: true,
86             dangerMode: true,
87         }).then((willDelete) => {
88             if (willDelete) {
89                 form.submit();
90             }
91         });
92     });
93 </script>
94 @endpush
95 @push('notif')
96     @if (session('success'))
97         <div class="alert alert-success border-left-success alert-dismissible fade show" role="alert">
98             {{ session('success') }}
99             <button type="button" class="close" data-dismiss="alert" aria-label="Close">
100                | <span aria-hidden="true">&times;</span>
101            </button>
102        </div>
103    @endif
104
105    @if (session('status'))
106        <div class="alert alert-success border-left-success" role="alert">
107            {{ session('status') }}
108        </div>
109    @endif
110 @endpush

```

## 10. Halaman add\_penjual

```

1  @extends('layouts.admin')
2  @section('content')
3  <div class="content-wrapper">
4      <div class="row">
5          <div class="col-lg-12 grid-margin stretch-card">
6              <div class="card">
7                  <div class="card-body">
8                      <h4 class="card-title">Menambahkan Data Penjual</h4>
9                      <form action="datapenjual/insert" method="POST" class="forms-sample" enctype="multipart/form-data">
10                         @csrf
11                         <div class="form-group">
12                             <label for="exampleInputName1">Firstname</label>
13                             <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ old('firstname') }}"/>
14                             <div class="text-danger">
15                                 @error('firstname')
16                                     {{ $message }}
17                                 @enderror
18                             </div>
19                         </div>
20                         <div class="form-group">
21                             <label for="exampleInputEmail13">Lastname</label>
22                             <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ old('lastname') }}"/>
23                         </div>
24                         <div class="form-group">
25                             <label for="exampleInputEmail13">No Handphone</label>
26                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ old('handphone') }}"/>
27                             <div class="text-danger">
28                                 @error('handphone')
29                                     {{ $message }}
30                                 @enderror
31                             </div>
32                         </div>
33                         <div class="form-group">
34                             <label for="">Provinsi</label>
35                             <select class="form-control" name="provinsi" id="provinsi" required>
36                                 <option value="">Provinsi</option>
37                                 @foreach ($provinces as $provinci) => $value
38                                 <option value="{{ $provinci }}" {{ old('provinsi') == $provinci ? 'selected' : '' }}>
39                                     {{ $provinci }}
40                             </select>
41                         </div>
42                     </div>
43                     <div class="form-group">
44                         <button type="submit" class="btn btn-primary" style="width: 100%;">Simpan
45                     </div>
46                 </div>
47             </div>
48         </div>
49     </div>
50 </div>
51 
```

```

40         {{ $value }}
41     </option>
42     @endforeach
43     </select>
44     <div class="text-danger">
45     @error('provinsi')
46     {{ $message }}
47     @enderror
48
49   </div>
50
51   </div>
52   <div class="form-group">
53     <label for="">Kota</label>
54     <select class="form-control" name="kota" id="kota">
55       <option value="">Kota</option>
56       @foreach ($kotas as $kotaId => $kotaName)
57         <option value="{{ $kotaId }}" {{ old('kota') == $kotaId ? 'selected' : '' }}>
58           {{ $kotaName }}
59         </option>
60       @endforeach
61     </select>
62     <div class="text-danger">
63     </div>
64   </div>
65   <div class="form-group">
66     <label for="exampleTextareal">Alamat</label>
67     <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ old('alamat') }}>
68     <div class="text-danger">
69       @error('alamat')
70       {{ $message }}
71       @enderror
72     </div>
73   </div>
74   <div class="form-group">
75     <label for="exampleInputEmail3">Email</label>
76     <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ old('email') }}>
77     <div class="text-danger">
78       @error('email')
79       {{ $message }}
80       @enderror
81     </div>
82   </div>
83   <div class="form-group">
84     <label for="exampleInputPassword4">Password</label>
85     <input name="password" type="password" class="form-control" id="password" placeholder="Password" value="{{ old('password') }}>
86     <div class="text-danger">
87       @error('password')
88       {{ $message }}
89       @enderror
90     </div>
91   </div>
92
93   <!-- <div class="form-group">
94     <label>Foto Profil</label>
95     <input type="file" name="img[]" class="file-upload-default">
96     <div class="input-group col-xs-12">
97       <input type="text" class="form-control file-upload-info" disabled placeholder="Upload Image">
98       <span class="input-group-append">
99         <button class="file-upload-browse btn btn-primary" type="button">Upload</button>
100      </span>
101    </div>
102  </div> -->
103
104  <div class="form-group">
105    <label>Foto Profil</label>
106    <div> </div>
107    <input type='file' name='img' class="form-control" value="{{ old('img') }}>
108
109  </div>
110
111  <div class="form-group">
112    <label>Foto KTP</label>
113    <div> </div>
114    <input type='file' name='ktp' class="form-control" value="{{ old('ktp') }}>
115    <div class="text-danger">
116      @error('ktp')

```

```

117 |     | {{ $message }}
118 |     | @enderror
119 |     | </div>
120 |   </div>
121 |
122 |   | <button type="submit" class="btn btn-primary mr-2">Submit</button>
123 |   | </form>
124 |   | </div>
125 |   | </div>
126 |   | </div>
127 | </div>
128 | </div>
129 |
130 @endsection
131
132 | <!-- Include the jQuery library -->
133 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
134 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
135
136 <script>
137 $(document).ready(function() {
138   $('select[name="provinsi"]').on('change', function() {
139     let provinsiId = $(this).val();
140     if (provinsiId) {
141       jQuery.ajax({
142         url: '/provinsi/' + provinsiId + '/city',
143         type: "GET",
144         dataType: "json",
145         success: function(data) {
146           $('select[name="kota"]').empty();
147           $.each(data, function(key, value) {
148             $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
149           });
150         },
151         error: function(xhr, status, error) {
152           console.error(error);
153         }
154       });
155     } else {
156       $('select[name="kota"]').empty();
157     }
158   });
159 });
160 </script>

```

## 11. Halaman datapenjual

```

1 @extends('layouts.admin')
2 @section('content')
3
4 <div class="content-wrapper">
5 <div class="row">
6   <div class="col-md-12 grid-margin">
7     <!-- <div class="row">
8       <div class="col-12 col-xl-8 mb-4 mb-xl-0" -->
9       <div class="card">
10         <div class="card-body">
11           <h4 class="card-title">Data Penjual</h4>
12           <a href="/datapenjual/add" type="button" class="btn btn-primary">Add</a>
13           <div class="table-responsive">
14             <table class="table table-striped">
15               <thead>
16                 <tr>
17                   <th>No</th>
18                   <th>Nama</th>
19                   <th>No Hp</th>
20                   <th>Provinsi</th>
21                   <th>Kota</th>
22                   <th>Alamat</th>
23                   <!-- <th>Foto Profil</th> -->
24                   <th>KTP</th>
25                   <th>Aksi</th>
26               </tr>
27             </thead>
28             <tbody>
29               @foreach ($penjual as $pj)
30               <tr>
31                 <td>{{ $loop->iteration }}</td>
32                 <td>{{ $pj->firstname . ' ' . $pj->lastname; }}<br>
33                 @endphp
34                 <td>{{ $pj->handphone}}</td>
35                 <td>{{ $pj->nama_provinsi}}</td>
36                 <td>{{ $pj->nama_kota}}</td>
37                 <td>{{ $pj->alamat}}</td>
38                 <!-- <td>
39                   <a href="{{ asset('foto_profil/' . $pj->img )}}" target=" blank" rel="noopener noreferrer">{{ $pj->img }}</a>

```

```

41      </td> -->
42      <td>
43          <a href="{{ asset('ktp', $pj->ktp) }}" target="_blank" rel="noopener noreferrer">{{ $pj->ktp }}</a>
44      </td>
45      <td>
46          <div class="d-flex">
47              <a href="{{ route('datapenjual/edit', { $pj->id }) }}" class="btn btn-sm btn-primary mr-2">Edit</a>
48              <button type="button" class="btn btn-danger" data-toggle="modal" data-target="#deleteModal{{ $pj->id }}">Hapus</button>
49          </div>
50      </td>
51  <div class="modal fade" id="deleteModal{{ $pj->id }}" tabindex="-1" role="dialog" aria-labelledby="deleteModalLabel{{ $pj->id }}">
52      <div class="modal-dialog" role="document">
53          <div class="modal-content">
54              <div class="modal-header">
55                  <h5 class="modal-title" id="deleteModalLabel{{ $pj->id }}>Konfirmasi Hapus</h5>
56                  <button type="button" class="close" data-dismiss="modal" aria-label="Close">&times;</span>
57              </button>
58          </div>
59          <div class="modal-body">
60              Apakah kamu yakin ingin menghapus data?
61          </div>
62          <div class="modal-footer">
63              <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
64              <form action="{{ route('datapenjual.destroy', $pj->id) }}" method="POST">
65                  @csrf
66                  @method('DELETE')
67                  <button type="submit" class="btn btn-danger">Hapus</button>
68              </form>
69          </div>
70      </div>
71  </div>
72  </div>
73  </td>
74  </tr>
75  @endforeach
76  </tbody>
77  </table>
78  </div>
79  </div>

```

## 12. Halaman edit\_penjual

```

1  @extends('layouts.admin')
2  @section('content')
3  <div class="content-wrapper">
4      <div class="row">
5          <div class="col-lg-12 grid-margin stretch-card">
6              <div class="card">
7                  <div class="card-body">
8                      <h4 class="card-title">Mengubah Data Penjual</h4>
9                      <form action="/datapenjual/update/{{ $penjual->id }}" method="POST" class="forms-sample" enctype="multipart/form-data">
10                         @csrf
11                         <div class="form-group">
12                             <label for="exampleInputName1">Firstname</label>
13                             <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $penjual->firstname }}>
14                             <div class="text-danger">
15                                 @error('firstname')
16                                     {{ $message }}
17                                 @enderror
18                             </div>
19                         </div>
20                         <div class="form-group">
21                             <label for="exampleInputEmail13">Lastname</label>
22                             <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $penjual->lastname }}>
23                             <div class="text-danger">
24                                 @error('lastname')
25                                     {{ $message }}
26                                 @enderror
27                             </div>
28                         </div>
29                         <div class="form-group">
30                             <label for="exampleInputEmail13">No Handphone</label>
31                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $penjual->handphone }}>
32                             <div class="text-danger">
33                                 @error('handphone')
34                                     {{ $message }}
35                                 @enderror
36                             </div>
37                         </div>
38                         <div class="form-group">
39                             <label for="">Provinsi</label>
40                             <select class="form-control" name="provinsi" id="provinsi" required>

```

```

41         <option value="">Provinsi</option>
42         @foreach ($provinsi as $provinsi => $value)
43             <option value="{{ $provinsi }}" {{ $penjual->province_id == $provinsi ? 'selected' : '' }}> {{ $value }}</option>
44         @endforeach
45     </select>
46     <div class="text-danger">
47         @error('provinsi')
48             {{ $message }}
49         @enderror
50     </div>
51 </div>
52 </div>
53     <div class="form-group">
54         <label for="Kota"></label>
55         <select class="form-control" name="kota" id="kota" >
56             <option value="">Kota</option>
57             @foreach ($kotas as $kota => $value)
58                 <option value="{{ $kota }}" {{ $penjual->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
59             @endforeach
60         </select>
61         <div class="text-danger">
62             @error('alamat')
63                 {{ $message }}
64             @enderror
65         </div>
66     <div class="form-group">
67         <label for="exampleTextareal">Alamat</label>
68         <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $penjual->alamat }}>
69     <div class="text-danger">
70         @error('alamat')
71             {{ $message }}
72         @enderror
73     </div>
74     <div class="form-group">
75         <label for="exampleInputEmail3">Email</label>
76         <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}>
77     <div class="text-danger">
78         @error('email')
79             {{ $message }}
80         @enderror
81     </div>
82 </div>
83     <div class="form-group">
84         <label for="password">Password</label>
85         <input type="password" class="form-control" name="password" id="password" value="">
86         <small class="form-text text-muted">Leave it blank if you don't want to change the password.</small>
87     </div>
88     <div class="form-group">
89         <label for="Foto Profil">Foto Profil</label>
90         <div>
91             <input type='file' name='img' class="form-control" value="{{ $penjual->img }}>
92             <a href="{{ asset('foto_profil/' . $penjual->img) }}" target="_blank" rel="noopener noreferrer" >{{ $penjual->img }} </a>
93         <div class="text-danger">
94             @error('img')
95                 {{ $message }}
96             @enderror
97         </div>
98     </div>
99     <div class="form-group">
100        <label>Foto KTP</label>
101        <div>
102            <input type='file' name='ktp' class="form-control" value="{{ $penjual->ktp }}>
103            <a href="{{ asset('ktp/' . $penjual->ktp) }}" target="_blank" rel="noopener noreferrer" >{{ $penjual->ktp }} </a>
104        <div class="text-danger">
105            @error('img')
106                {{ $message }}
107            @enderror
108        </div>
109    </div>
110    <div>
111        <button type="submit" class="btn btn-primary mr-2">Submit</button>
112    </div>
113 </div>
114 </div>
115 </div>
116 </div>
117 </div>
118 </div>
119 </div>
```

```

121     @endsection
122
123     <!-- Include the jQuery library -->
124     <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
125     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
126
127     <script>
128         $(document).ready(function() {
129             $('select[name="provinsi"]').on('change', function() {
130                 let provinceId = $(this).val();
131                 if (provinceId) {
132                     jQuery.ajax({
133                         url: '/province/' + provinceId + '/cities',
134                         type: "GET",
135                         dataType: "json",
136                         success: function(data) {
137                             $('select[name="kota"]').empty();
138                             $.each(data, function(key, value) {
139                                 $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
140                             });
141                         },
142                         error: function(xhr, status, error) {
143                             console.error(error);
144                         }
145                     });
146                 } else {
147                     $('select[name="kota"]').empty();
148                 }
149             });
150         });
151     </script>

```

### 13. Halaman add\_pembeli

```

1     @extends('layouts.admin')
2     @section('content')
3     <div class="content-wrapper">
4         <div class="row">
5             <div class="col-lg-12 grid-margin stretch-card">
6                 <div class="card">
7                     <div class="card-body">
8                         <h4 class="card-title">Menambahkan Data Pembeli</h4>
9                         <form action="/datapembeli/insert" method="POST" class="forms-sample" enctype="multipart/form-data">
10                            @csrf
11                            <div class="form-group">
12                                <label for="exampleInputName1">Firstname</label>
13                                <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ old('firstname') }}"/>
14                                <div class="text-danger">
15                                    @error('firstname')
16                                        {{ $message }}
17                                    @enderror
18                                </div>
19                            </div>
20                            <div class="form-group">
21                                <label for="exampleInputEmail13">Lastname</label>
22                                <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ old('lastname') }}"/>
23
24                            </div>
25                            <div class="form-group">
26                                <label for="exampleInputEmail13">No Handphone</label>
27                                <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ old('handphone') }}"/>
28                                <div class="text-danger">
29                                    @error('handphone')
30                                        {{ $message }}
31                                    @enderror
32                                </div>
33                            </div>
34                            <div class="form-group">
35                                <label for="">Provinsi</label>
36                                <select class="form-control" name="provinsi" id="provinsi" required>
37                                    <option value="">Provinsi</option>
38                                    @foreach ($provinsis as $provinsi => $value)
39                                        <option value="{{ $provinsi }}" {{ old('provinsi') == $provinsi ? 'selected' : '' }}>
40                                            {{ $value }}

```

```

41         </option> ..
42     @endforeach
43     </select>
44     <div class="text-danger">
45         @error('provinsi')
46         {{ $message }}
47         @enderror
48     </div>
49
50     </div>
51     <div class="form-group">
52         <label for="">Kota</label>
53         <select class="form-control" name="kota" id="kota">
54             <option value="">Kota</option>
55             @foreach ($kotas as $kotaId => $kotaName)
56                 <option value="{{ $kotaId }}" {{ old('kota') == $kotaId ? 'selected' : '' }}>
57                     {{ $kotaName }}
58                 </option>
59             @endforeach
60         </select>
61         <div class="text-danger">
62             @error('alamat')
63             {{ $message }}
64             @enderror
65         </div>
66         <div class="form-group">
67             <label for="exampleTextareal">Alamat</label>
68             <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ old('alamat') }}">
69             <div class="text-danger">
70                 @error('alamat')
71                 {{ $message }}
72                 @enderror
73             </div>
74         </div>
75         <div class="form-group">
76             <label for="exampleInputEmail3">Email</label>
77             <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ old('email') }}">
78             <div class="text-danger">
79                 @error('email')
80                 {{ $message }}
81                 @enderror
82             </div>
83             <div class="form-group">
84                 <label for="exampleInputPassword4">Password</label>
85                 <input name="password" type="password" class="form-control" id="password" placeholder="Password" value="{{ old('password') }}">
86                 <div class="text-danger">
87                     @error('password')
88                     {{ $message }}
89                     @enderror
90                 </div>
91             </div>
92             <div class="form-group">
93                 <label for="img">Foto Profil</label>
94                 <div class="image-preview"></div>
95                 <input type="file" name="img" id="img" class="form-control">
96             </div>
97
98             <button type="submit" class="btn btn-primary mr-2">Submit</button>
99         </form>
100     </div>
101   </div>
102 </div>
103 </div>
104 </div>
105
106 @endsection
107
108     <!-- Include the jQuery library --&gt;
109     &lt;script src="https://code.jquery.com/jquery-3.6.0.min.js"&gt;&lt;/script&gt;
110     &lt;script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"&gt;&lt;/script&gt;
111
112     &lt;script&gt;
113     $(document).ready(function() {
114         $('select[name="provinsi"]').on('change', function() {
115             let provinsiId = $(this).val();
116             if (provinsiId) {
117                 // Fetch data from API here
118             }
119         });
120     });
</pre>

```

```

117     jQuery.ajax({
118         url: '/provinsi/' + provinsiId + '/city',
119         type: "GET",
120         dataType: "json",
121         success: function(data) {
122             $('select[name="kota"]').empty();
123             $.each(data, function(key, value) {
124                 $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
125             });
126         },
127         error: function(xhr, status, error) {
128             console.error(error);
129         }
130     });
131     } else {
132     $('select[name="kota"]').empty();
133     }
134   });
135 });
</script>
<style>
.image-preview {
| margin-top: 10px;
}
.selected-image {
| width: 200px;
| height: auto;
}
</style>

```

## 14. Halaman datapembeli

```

1 @extends('layouts.admin')
2 @section('content')
3     <div class="content-wrapper">
4         <div class="row">
5             <div class="col-lg-12 grid-margin stretch-card">
6                 <div class="card">
7                     <div class="card-body">
8                         <h4 class="card-title">Data Pembeli</h4>
9                         <a href="/datapembeli/add" type="button" class="btn btn-primary">Add</a>
10                        <!-- <p class="card-description">
11                            | pbd class <code>.table-striped</code>
12                        </p> -->
13                        <div class="table-responsive">
14                            <table class="table table-striped">
15                                <thead>
16                                    <tr>
17                                        <th>No</th>
18                                        <th>>Nama</th>
19                                        <th>No Hp</th>
20                                        <th>Provinsi</th>
21                                        <th>Kota</th>
22                                        <th>Alamat</th>
23                                        <!-- <th>Foto Profil</th> -->
24                                        <th>Aksi</th>
25                                </tr>
26                            </thead>
27                            <tbody>
28                                @foreach ($pembeli as $pb)
29
30                                <tr>
31                                    <td scope="row">{{ $loop->iteration }}</td>
32                                    <td>@php
33                                        echo $pb->firstname . ' ' . $pb->lastname;
34                                    @endphp</td>
35                                    <td>{{ $pb->handphone }}</td>
36                                    <td>{{ $pb->nama_provinsi }}</td>
37                                    <td>{{ $pb->nama_kota }}</td>
38                                    <td>{{ $pb->alamat }}</td>
39                                    <!-- <td>
40                                        | | <a href="{{ asset('foto_profil/' . $pb->img) }}" target="_blank" rel="noopener noreferrer"> {{ $pb->img }} </a>
41                                    </td> -->

```

```
41 <td>
42     <div class="d-flex">
43         <a href="{{ route('datapembeli/edit', $pb->id) }}" class="btn btn-sm btn-primary mr-2">Edit</a>
44         <button type="button" class="btn btn-danger" data-toggle="modal" data-target="#deleteModal{{ $pb->id }}>Hapus</button>
45     </div>
46 </td>
47 <div class="modal fade" id="deleteModal{{ $pb->id }}" tabindex="-1" role="dialog" aria-labelledby="deleteModalLabel{{ $pb->id }}"
48 <div class="modal-dialog" role="document">
49     <div class="modal-content">
50         <div class="modal-header">
51             <h5 class="modal-title" id="deleteModalLabel{{ $pb->id }}>Konfirmasi Hapus</h5>
52             <button type="button" class="close" data-dismiss="modal" aria-label="Close">
53                 <span aria-hidden="true">&times;</span>
54             </button>
55         </div>
56         <div class="modal-body">
57             Apakah kamu yakin ingin menghapus data?
58         </div>
59         <div class="modal-footer">
60             <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
61             <form action="{{ route('datapembeli.destroy', $pb->id) }}" method="POST">
62                 @csrf
63                 @method('DELETE')
64                 <button type="submit" class="btn btn-danger">Hapus</button>
65             </form>
66         </div>
67     </div>
68 </div>
69 </td>
70 </tr>
71 </tbody>
72 </table>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
```

## 15. Halaman edit\_pembeli

```
1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4     <div class="row">
5         <div class="col-lg-12 grid-margin stretch-card">
6             <div class="card">
7                 <div class="card-body">
8                     <h4 class="card-title">Mengubah Data Pembeli</h4>
9                     <form action="/datapembeli/update/{{ $pembeli->id }}" method="POST" class="forms-sample" enctype="multipart/form-data">
10                         @csrf
11                         <div class="form-group">
12                             <label for="exampleInputName1">Firstname</label>
13                             <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $pembeli->firstname }}">
14                             <div class="text-danger">
15                                 @error('firstname')
16                                     {{ $message }}
17                                 @enderror
18                             </div>
19                         </div>
20                         <div class="form-group">
21                             <label for="exampleInputEmail3">Lastname</label>
22                             <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $pembeli->lastname }}">
23                             <div class="text-danger">
24                                 @error('lastname')
25                                     {{ $message }}
26                                 @enderror
27                             </div>
28                         </div>
29                         <div class="form-group">
30                             <label for="exampleInputEmail3">No Handphone</label>
31                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $pembeli->handphone }}>
32                             <div class="text-danger">
33                                 @error('handphone')
34                                     {{ $message }}
35                                 @enderror
36                             </div>
37                         </div>
38                         <div class="form-group">
39                             <label for="">Provinsi</label>
40                             <select class="form-control" name="provinsi" id="provinsi" required>
```

```

41         <option value="">Provinsi</option>
42         @foreach ($provinsis as $provinsi => $value)
43             | <option value="{{ $provinsi }}{{ $pembeli->province_id == $provinsi ? 'selected' : '' }}> {{ $value }}</option>
44         @endforeach
45     </select>
46     <div class="text-danger">
47         @error('provinsi')
48             {{ $message }}
49         @enderror
50     </div>
51     </div>
52     <div class="form-group">
53         <label for="Kota">Kota</label>
54         <select class="form-control" name="kota" id="kota" >
55             <option value="">Kota</option>
56             @foreach ($kotas as $kota => $value)
57                 <option value="{{ $kota }}{{ $pembeli->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
58             @endforeach
59         </select>
60         <div class="text-danger">
61             @error('kota')
62                 {{ $message }}
63             @enderror
64         </div>
65         <div class="form-group">
66             <label for="exampleTextareal">Alamat</label>
67             <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $pembeli->alamat }}>
68             <div class="text-danger">
69                 @error('alamat')
70                     {{ $message }}
71                 @enderror
72             </div>
73         </div>
74         <div class="form-group">
75             <label for="exampleInputEmail3">Email</label>
76             <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}>
77             <div class="text-danger">
78                 @error('email')
79                     {{ $message }}
80                 @enderror
81             </div>
82             <div class="form-group">
83                 <label for="password">Password</label>
84                 <input type="password" class="form-control" name="password" id="password" value="">
85                 <small class="form-text text-muted">Leave it blank if you don't want to change the password.</small>
86             </div>
87             <div class="form-group">
88                 <label>Foto Profil</label>
89                 <div>
90                     <input type='file' name='img' class="form-control" value="{{ $pembeli->img }}>
91                     <div>
92                         <a href="{{ asset('foto_profil/'. $pembeli->img )}}> target="_blank" rel="noopener noreferrer"{{ $pembeli->img }} </a>
93                     <div class="text-danger">
94                         @error('img')
95                             {{ $message }}
96                         @enderror
97                     </div>
98                 </div>
99             </div>
100             <button type="submit" class="btn btn-primary mr-2">Submit</button>
101         </form>
102     </div>
103     </div>
104     </div>
105 </div>
106 </div>
107 </div>
108 @endsection

```

```

111    <!-- Include the jQuery library -->
112    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
113    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
114
115    <script>
116        $(document).ready(function() {
117            $('select[name="provinsi"]').on('change', function() {
118                let provinceId = $(this).val();
119                if (provinceId) {
120                    jQuery.ajax({
121                        url: '/province/' + provinceId + '/cities',
122                        type: "GET",
123                        dataType: "json",
124                        success: function(data) {
125                            $('select[name="kota"]').empty();
126                            $.each(data, function(key, value) {
127                                $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
128                            });
129                        },
130                        error: function(xhr, status, error) {
131                            console.error(error);
132                        }
133                    });
134                } else {
135                    $('select[name="kota"]').empty();
136                }
137            });
138        });
139    </script>

```

## 16. Halaman profil admin

```

1 @extends('layouts.admin')
2 @section('content')
3 <div class="content-wrapper">
4     <section class="section main-section">
5         <div class="grid grid-cols-1 gap-6 lg:grid-cols-2 mb-6">
6
7             <div class="card">
8                 <header class="card-header">
9                     <p class="card-header-title">
10                         <span class="icon"><i class="mdi mdi-account"></i></span>
11                         Profile
12                     </p>
13                 </header>
14                 <div class="card-content">
15                     <div class="image w-28 h-28 mx-auto">
16                         
17                     <hr>
18                         <@csrf>
19                         <div class="form-group">
20                             <label for="exampleInputName1">Name</label>
21                             <input name="name" type="text" class="form-control" id="name" placeholder="Name" value="@php echo $admin->firstname . ' ' . $admin->lastname @endphp" required>
22                         </div>
23                         <div class="form-group">
24                             <label for="exampleInputEmail13">No Handphone</label>
25                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $admin->handphone }}>
26                         </div>
27                         <div class="form-group">
28                             <label for="Provinsi">Provinsi</label>
29                             <select class="form-control" name="provinsi" id="provinsi" disabled>
30                                 <option value="">Provinsi</option>
31                                 @foreach ($provinsis as $provinsi => $value)
32                                     <option value="{{ $provinsi }}> {{ $admin->province_id == $provinsi ? 'selected' : '' }}> {{ $value }}</option>
33                                 @endforeach
34                             </select>
35                             <div class="text-danger"></div>
36                         </div>
37                         <div class="form-group">
38                         </div>
39                     </div>
40                 </div>

```

```

41 <label for="">Kota</label>
42 <select class="form-control" name="kota" id="kota" disabled>
43     <option value="">Kota</option>
44     @foreach ($kotas as $kota => $value)
45         <option value="{{ $kota }}{{ $admin->city_id == $kota ? ' selected' : '' }}> {{ $value }}</option>
46     @endforeach
47 </select>
48 <div class="text-danger">
49 </div>
50 </div>
51 <div class="form-group">
52     <label for="exampleTextareal">Alamat</label>
53     <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $admin->alamat }}" readonly>
54
55     </div>
56     <div class="form-group">
57         <label for="exampleInputEmail3">Email</label>
58         <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}" readonly>
59         <div class="text-danger">
60             @error('email')
61             {{ $message }}
62         @enderror
63     </div>
64 </div>
65 </div>
66 <div class="card">
67     <header class="card-header">
68         <p class="card-header-title">
69             <span class="icon"><i class="mdi mdi-account-circle"></i></span>
70             Edit Profile
71         </p>
72     </header>
73     <div class="card-content">
74         <form method="POST" action="{{ route('profiladmin.update') }}" enctype="multipart/form-data">
75             @csrf
76             <div class="form-group">
77                 <label>Foto Profil</label>
78                 <div>
79                     <input type="file" name='img' class="form-control" value="{{ $admin->img }}>
80
81                     <!-- <div>
82                         <a href="{{ asset('foto_profil/' . $admin->img )}}" target="_blank" rel="noopener noreferrer"> {{ $admin->img }} </a>
83                     </div>
84                     <div class="text-danger">
85                         @error('img')
86                         {{ $message }}
87                     @enderror
88                 </div>
89                 <br>
90                 <div class="form-group">
91                     <label for="exampleInputName1">Firstname</label>
92                     <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $admin->firstname }}>
93                     <div class="text-danger">
94                         @error('firstname')
95                         {{ $message }}
96                     @enderror
97                 </div>
98                 <div class="form-group">
99                     <label for="exampleInputEmail3">Lastname</label>
100                    <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $admin->lastname }}>
101
102                    </div>
103                    <div class="form-group">
104                        <label for="exampleInputEmail3">No Handphone</label>
105                        <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $admin->handphone }}>
106                        <div class="text-danger">
107                            @error('handphone')
108                            {{ $message }}
109                        @enderror
110                    </div>
111                </div>
112                <div class="form-group">
113                    <label for="">Provinsi</label>
114                    <select class="form-control" name="provinsi" id="provinsi">
115                        <option value="">Provinsi</option>
116                        @foreach ($provinsis as $provinsi => $value)
117                            <option value="{{ $provinsi }}{{ $admin->province_id == $provinsi ? ' selected' : '' }}> {{ $value }}</option>
118                        @endforeach
119                    </div>

```

```

120      </select>
121      <div class="text-danger"></div>
122  </div>
123      <div class="form-group">
124          <label for="">Kota</label>
125          <select class="form-control" name="kota" id="kota">
126              <option value="">Kota</option>
127              @foreach ($kotas as $kota => $value)
128                  <option value="{{ $kota }}"" {{ $admin->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
129              @endforeach
130      </select>
131      <div class="text-danger"></div>
132  </div>
133      <div class="form-group">
134          <label for="exampleTextareal">Alamat</label>
135          <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $admin->alamat }}">
136          <div class="text-danger">
137              @error('alamat')
138                  {{ $message }}
139              @enderror
140          </div>
141      </div>
142      <div class="form-group">
143          <label for="exampleInputEmail13">Email</label>
144          <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}">
145          <div class="text-danger">
146              @error('email')
147                  {{ $message }}
148              @enderror
149          </div>
150      </div>
151      <hr> <br>
152      <div class="control">
153          <div class="control">
154              <button type="submit" class="btn btn-primary mr-2">Submit</button>
155          </div>
156      </div>
157  </div>
158  </form>
159      </div>
160  </div>
161  </div>
162  <div class="card">
163      <header class="card-header">
164          <p class="card-header-title">
165              <span class="icon"><i class="mdi mdi-lock"></i></span>
166              Change Password
167          </p>
168      </header>
169      <div class="card-content">
170          <form action="{{ route('change.password.admin') }}" method="POST">
171              @csrf
172
173              <div class="form-group">
174                  <label for="current_password">Current Password</label>
175                  <input name="current_password" type="password" class="form-control" id="current_password" placeholder="Current Password">
176                  <div class="text-danger">
177                      @error('current_password')
178                          {{ $message }}
179                      @enderror
180                  </div>
181              </div>
182
183              <div class="form-group">
184                  <label for="new_password">New Password</label>
185                  <input name="new_password" type="password" class="form-control" id="new_password" placeholder="New Password">
186                  <div class="text-danger">
187                      @error('new_password')
188                          {{ $message }}
189                      @enderror
190                  </div>
191              </div>
192
193              <div class="form-group">
194                  <label for="confirm_password">Confirm Password</label>
195                  <input name="confirm_password" type="password" class="form-control" id="confirm_password" placeholder="Confirm Password">
196                  <div class="text-danger">

```

```

197     @error('confirm_password')
198     {{ $message }}
199     @enderror
200   
```

```

201   
```

```

202   
```

```

203   
```

```

204   
```

```

205   
```

```

206   
```

```

207   
```

```

208   
```

```

209   
```

```

210   
```

```

211   
```

```

212   
```

```

213 @endsection
214 <!-- Include the jQuery library -->
215 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
216 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
217 
218 <script>
219 $(document).ready(function() {
220   $('select[name="provinsi"]').on('change', function() {
221     let provinsiId = $(this).val();
222     if (provinsiId) {
223       jQuery.ajax({
224         url: '/provinsi/' + provinsiId + '/city',
225         type: "GET",
226         dataType: "json",
227         success: function(data) {
228           $('select[name="kota"]').empty();
229           $.each(data, function(key, value) {
230             $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
231           });
232         },
233         error: function(xhr, status, error) {
234           console.error(error);
235         }
236       });
237     } else {
238       $('select[name="kota"]').empty();
239     }
240   });
241 </script>
242 
243

```

## 17. Halaman dashboard admin

```

1  @extends('layouts.admin')
2  @section('content')
3  <!-- partial -->
4  <!-- <div class="main-panel" -->
5  <div class="content-wrapper">
6    <div class="row">
7      <div class="col-md-12 grid-margin">
8        <div class="row">
9          <div class="col-12 col-xl-8 mb-4 mb-xl-0">
10            <h3 class="font-weight-bold">Welcome {{Auth::user()->name}}
11            <h6 class="font-weight-normal mb-0">All systems are running smoothly!
12          </div>
13        </div>
14      </div>
15    </div>
16    <div class="row">
17      <div class="col-md-6 grid-margin stretch-card">
18        <div class="card tale-bg">
19          <div class="card-peole mt-auto">
20            
21            <div class="weather-info">
22              <div class="d-flex">
23                <div>
24                  </div>
25                </div>
26              </div>
27            </div>
28            <div class="col-md-6 grid-margin transparent">
29              <div class="row">
30                <div class="col-md-6 mb-4 stretch-card transparent">
31                  <div class="card card-tale">
32                    <div class="card-body">
33                      <?php
34                        $jumlahPenjual = DB::table('penjual')->count(); // Menghitung jumlah penjual
35                      ?>
36                      <p class="mb-4">Jumlah Owner UMKM</p>
37                      <p class="fs-30 mb-2"><?php echo $jumlahPenjual; ?></p>
38
39                    </div>
40                  </div>

```

```

42 <div class="col-md-6 mb-4 stretch-card transparent">
43   <div class="card card-dark-blue">
44     <div class="card-body">
45       <?php
46       $jumlahMenungguKonfirmasi = DB::table('umkm')->where('status', 0)->count(); // Menghitung jumlah UMKM
47     ?>
48       <p class="mb-4">UMKM Perlu Konfirmasi</p>
49       <p class="fs-30 mb-2"><?php echo $jumlahMenungguKonfirmasi; ?></p>
50       <!-- <p>22.00% (30 days)</p> -->
51     </div>
52   </div>
53 </div>
54 <div class="row">
55   <div class="col-md-6 mb-4 mb-lg-0 stretch-card transparent">
56     <div class="card card-light-blue">
57       <div class="card-body">
58         <?php
59
60       $jumlahDiakui = DB::table('umkm')->where('status', 1)->count(); // Menghitung jumlah UMKM dengan status 1
61     ?>
62       <p class="mb-4">UMKM Beroperasi</p>
63       <p class="fs-30 mb-2"><?php echo $jumlahDiakui; ?></p>
64
65     </div>
66   </div>
67 </div>
68 <div class="col-md-6 stretch-card transparent">
69   <div class="card card-light-danger">
70     <div class="card-body">
71       <?php
72
73       $jumlahDitolak = DB::table('umkm')->where('status', 2)->count(); // Menghitung jumlah UMKM dengan status 2
74     ?>
75       <p class="mb-4">UMKM Ditolak</p>
76       <p class="fs-30 mb-2"><?php echo $jumlahDitolak; ?></p>
77     </div>
78   </div>
79 </div>
80
81 </div>
82 </div>
83
84
85 <div class="row">
86   <div class="col-md-12 grid-margin stretch-card">
87     <div class="card">
88       <div class="card-body">
89         <p class="card-title mb-0">Data UMKM</p>
90         <div class="table-responsive">
91           <table class="table table-striped table-bordered">
92             <thead>
93               <tr>
94                 <th>Nama UMKM</th>
95                 <!-- <th>Alamat</th> -->
96                 <th>Handphone</th>
97                 <th>Status</th>
98               </tr>
99             </thead>
100            <tbody>
101              <?php
102                $umkmData = DB::table('umkm')->get(); // Mengambil data UMKM
103                foreach ($umkmData as $umkm) {
104                  $statusLabel = "";
105                  switch ($umkm->status) {
106                    case 0:
107                      $statusLabel = "Menunggu Konfirmasi";
108                      break;
109                    case 1:
110                      $statusLabel = "Dikonfirmasi";
111                      break;
112                    case 2:
113                      $statusLabel = "Ditolak";
114                      break;
115                    default:
116                      $statusLabel = "Tidak Diketahui";
117                  }
118                }
119              <?>
120              <tr>
121                <td><?php echo $umkm->nama; ?></td>

```

```

121           . . .
122           <td><?php echo $umkm->handphone; ?></td>
123           <td class="font-weight-medium">
124             <div class="badge "><?php echo ($umkm->status == 0) ? 'badge-warning' : (($umkm->status == 1) ? 'badge-su
125               | <?php echo $statusLabel; ?>
126             </div>
127           </td>
128         </tr>
129       </php
130     }
131   >;
132   </tbody>
133 </table>
134 </div>
135 </div>
136 </div>
137 </div>
138
139 </div>
140
141 <div class="row">
142 @endsection

```

### c. Bagian Pembeli

#### 1. Halaman *index chat*

```

1 <!DOCTYPE html>
2 <html lang="{{ str_replace('_', '-', app()->getLocale()) }}>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1">
6
7     <title>Marketplace UMKM Sanjai</title>
8
9     <!-- Google Font -->
10    <link href="https://fonts.googleapis.com/css2?family=Cairo:wght@200;300;400;600;900&display=swap" rel="stylesheet">
11
12    <!-- Css Styles -->
13    <link rel="stylesheet" href="{{ asset('frontend/css/bootstrap.min.css') }}" type="text/css">
14    <link rel="stylesheet" href="{{ asset('frontend/css/font-awesome.min.css') }}" type="text/css">
15    <link rel="stylesheet" href="{{ asset('frontend/css/elegant-icons.css') }}" type="text/css">
16    <link rel="stylesheet" href="{{ asset('frontend/css/nice-select.css') }}" type="text/css">
17    <link rel="stylesheet" href="{{ asset('frontend/css/jquery-ui.min.css') }}" type="text/css">
18    <link rel="stylesheet" href="{{ asset('frontend/css/owl.carousel.min.css') }}" type="text/css">
19    <link rel="stylesheet" href="{{ asset('frontend/css/slicknav.min.css') }}" type="text/css">
20    <link rel="stylesheet" href="{{ asset('frontend/css/style.css') }}" type="text/css">
21
22    <style>
23      .container {
24        max-width: 1170px;
25        margin: auto;
26      }
27
28      img {
29        max-width: 100%;
30      }
31
32      .inbox_people {
33        background: #f8f8f8 none repeat scroll 0 0;
34        float: left;
35        overflow: hidden;
36        width: 40%;
37        border-right: 1px solid #c4c4c4;
38      }
39
40      .inbox_msg {

```

```

41      border: 1px solid #c4c4c4;
42      clear: both;
43      overflow: hidden;
44  }
45
46 .top_spac {
47   margin: 20px 0 0;
48 }
49
50
51 .recent_heading {
52   float: left;
53   width: 40%;
54 }
55
56 .srch_bar {
57   display: inline-block;
58   text-align: right;
59   width: 60%;
60 }
61
62 .headind_srch {
63   padding: 10px 29px 10px 20px;
64   overflow: hidden;
65   border-bottom: 1px solid #c4c4c4;
66 }
67
68 .recent_heading h4 {
69   color: #05728f;
70   font-size: 21px;
71   margin: auto;
72 }
73
74 .srch_bar input {
75   border: 1px solid #cdcdcd;
76   border-width: 0 0 1px 0;
77   width: 80%;
78   padding: 2px 0 4px 6px;
79
80   background: none;
81 }
82
83 .srch_bar .input-group-addon button {
84   background: rgba(0, 0, 0, 0) none repeat scroll 0 0;
85   border: medium none;
86   padding: 0;
87   color: #707070;
88   font-size: 18px;
89 }
90
91 .srch_bar .input-group-addon {
92   margin: 0 0 0 -27px;
93 }
94
95 .chat_ib h5 {
96   font-size: 15px;
97   color: #464646;
98   margin: 0 0 8px 0;
99 }
100
101 .chat_ib h5 span {
102   font-size: 13px;
103   float: right;
104 }
105
106 .chat_ib p {
107   font-size: 14px;
108   color: #989898;
109   margin: auto
110 }
111
112 .chat_img {
113   float: left;
114   width: 11%;
115 }
116
117 .chat_ib {
118   float: left;
119   padding: 0 0 0 15px;

```

```
119 |     width: 88%;  
120 | }  
121 | .chat_people {  
122 |     overflow: hidden;  
123 |     clear: both;  
124 | }  
125 | .chat_list {  
126 |     border-bottom: 1px solid #c4c4c4;  
127 |     margin: 0;  
128 |     padding: 18px 16px 10px;  
129 | }  
130 | .inbox_chat {  
131 |     height: 550px;  
132 |     overflow-y: scroll;  
133 | }  
134 | .active_chat {  
135 |     background: #ebebeb;  
136 | }  
137 | .incoming_msg_img {  
138 |     display: inline-block;  
139 |     width: 6%;  
140 | }  
141 | .received_msg {  
142 |     display: inline-block;  
143 |     padding: 0 0 0 10px;  
144 |     vertical-align: top;  
145 |     width: 92%;  
146 | }  
147 | .received_withd_msg p {  
148 |     background: #ebebeb none repeat scroll 0 0;  
149 |     border-radius: 3px;  
150 |     color: #646464;  
151 | }  
152 | .received_withd_msg p {  
153 |     font-size: 14px;  
154 |     margin: 0;  
155 |     padding: 5px 10px 5px 12px;  
156 |     width: 100%;  
157 | }  
158 | .time_date {  
159 |     color: #747474;  
160 |     display: block;  
161 |     font-size: 12px;  
162 |     margin: 8px 0 0;  
163 | }  
164 | .received_withd_msg {  
165 |     width: 57%;  
166 | }  
167 | .msgs {  
168 |     float: left;  
169 |     padding: 30px 15px 0 25px;  
170 |     width: 60%;  
171 | }  
172 | .sent_msg p {  
173 |     background: #05728f none repeat scroll 0 0;  
174 |     border-radius: 3px;  
175 |     font-size: 14px;  
176 |     margin: 0;  
177 |     color: #fff;  
178 |     padding: 5px 10px 5px 12px;  
179 |     width: 100%;  
180 | }  
181 | .outgoing_msg {  
182 |     overflow: hidden;  
183 |     margin: 26px 0 26px;  
184 | }  
185 | .sent_msg {  
186 |     float: right;  
187 | }
```

```

198     width: 46%;
199 }
200
201 .input_msg_write input {
202     background: rgba(0, 0, 0, 0) none repeat scroll 0 0;
203     border: medium none;
204     color: #4c4c4c;
205     font-size: 15px;
206     min-height: 48px;
207     width: 100%;
208 }
209
210 .type_msg {
211     border-top: 1px solid #c4c4c4;
212     position: relative;
213 }
214
215 .msg_send_btn {
216     background: #05728f none repeat scroll 0 0;
217     border: medium none;
218     border-radius: 50%;
219     color: #fff;
220     cursor: pointer;
221     font-size: 17px;
222     height: 33px;
223     position: absolute;
224     right: 0;
225     top: 11px;
226     width: 33px;
227 }
228
229 .messaging {
230     padding: 0 0 50px 0;
231 }
232
233 .msg_history {
234     height: 516px;
235     overflow-y: auto;
236 }
237 
```

---

```

238 </style>
239
240 </head>
241 <body class="antialiased">
242     <!-- Page Preloader -->
243     <div id="preloader">
244         <div class="loader"></div>
245     </div>
246
247     <!-- Humberger Begin -->
248     <div class="humberger_menu_overlay"></div>
249     <div class="humberger_menu_wrapper">
250         <div class="humberger_menu_logo">
251             <a href="#"></a>
252         </div>
253         <div class="humberger_menu_cart">
254             <ul>
255                 <li><a href="#"><i class="fa fa-heart"></i> <span>1</span></a></li>
256                 <li><a href="#"><i class="fa fa-shopping-bag"></i> <span>3</span></a></li>
257             </ul>
258             <div class="header_cart_price">item: <span>$150.00</span></div>
259         </div>
260         <div class="humberger_menu_widget">
261             <div class="header_top_right_language">
262                 
263                 <div>English</div>
264                 <span class="arrow_carrot-down"></span>
265                 <ul>
266                     <li><a href="#">Spanish</a></li>
267                     <li><a href="#">English</a></li>
268                 </ul>
269             </div>
270             <div class="header_top_right_auth">
271                 <a href="{{ route('login') }}><i class="fa fa-user"></i> Login</a>
272             </div>
273         </div>
274         <nav class="humberger_menu_nav mobile-menu">
275             <ul>
276                 <li class="active"><a href=".index.html">Home</a></li>
277                 <li><a href=".shoes-grid.html">Shoes</a></li>

```

```

277     <li><a href="#">Pages</a>
278         <ul class="header__menu__dropdown">
279             <li><a href="/shop-details.html">Shop Details</a></li>
280             <li><a href="/shopping-cart.html">Shopping Cart</a></li>
281             <li><a href="/checkout.html">Check Out</a></li>
282             <li><a href="/blog-details.html">Blog Details</a></li>
283         </ul>
284     </li>
285     <li><a href="/blog.html">Blog</a></li>
286     <li><a href="/contact.html">Contact</a></li>
287 </ul>
288 </nav>
289 <div id="mobile-menu-wrap"></div>
290 <div class="header__top__right__social">
291     <a href="#"><i class="fa fa-facebook"></i></a>
292     <a href="#"><i class="fa fa-twitter"></i></a>
293     <a href="#"><i class="fa fa-linkedin"></i></a>
294     <a href="#"><i class="fa fa-pinterest-p"></i></a>
295 </div>
296 <div class="umberger__menu__contact">
297     <ul>
298         <li><i class="fa fa-envelope"></i> hello@colorlib.com</li>
299         <li>Free Shipping for all Order of $99</li>
300     </ul>
301 </div>
302 </div>
303 <!-- Humberger End -->
304
305 <!-- Header Section Begin -->
306 <header class="header">
307     <div class="header__top">
308         <div class="container">
309             <div class="row">
310                 <div class="col-lg-6 col-md-6">
311                     <!-- <div class="header__top__left">
312                         <ul>
313                             <li><i class="fa fa-envelope"></i> hello@colorlib.com</li>
314                             <li>Free Shipping for all Order of $99</li>
315                         </ul>
316                     </div> -->
317                 </div>
318                 <div class="col-lg-6 col-md-6">
319                     <div class="header__top__right">
320                         <!-- <div class="header__top__right__social">
321                             <a href="#"><i class="fa fa-facebook"></i></a>
322                             <a href="#"><i class="fa fa-twitter"></i></a>
323                             <a href="#"><i class="fa fa-linkedin"></i></a>
324                             <a href="#"><i class="fa fa-pinterest-p"></i></a>
325                         </div> -->
326                         <!-- <div class="header__top__right__language">
327                             
328                             <div>English</div>
329                             <span>arrow_carrot-down</span>
330                             <ul>
331                                 <li><a href="#">Spanish</a></li>
332                                 <li><a href="#">English</a></li>
333                             </ul>
334                         </div> -->
335
336                         @if(Auth::guest())
337                             <div class="header__top__right__auth">
338                                 <a href="{{ route('register') }}><i class="fa fa-user"></i> Register</a>
339                             </div>
340                             <div class="header__top__right__auth">
341                                 <a href="{{ route('login') }}><i class="fa fa-user"></i> Login</a>
342                             </div>
343                         @else
344                             <div class="header__top__right__auth">
345                                 <a href="/profilpembeli"><i class="fa fa-user"> {{Auth::user()->name}}</i></a>
346                             </div>
347                             <div class="header__top__right__auth">
348                                 <a href="{{ route('logout') }}><i class="ti-power-off text-primary"></i> Logout</a>
349                             </div>
350                         @endif
351                     </div>
352                 </div>
353             </div>
354         </div>
355     </div>

```

```

356 <div class="container">
357   <div class="row">
358     <div class="col-lg-3">
359       <div class="header__logo">
360         |   <a href=".index.html"></a>
361       </div>
362     </div>
363     <div class="col-lg-6">
364       <nav class="header__menu">
365         <ul>
366           <li><a href="/">Home</a></li>
367           <li @if(Route::currentRouteName() == "purchase") class="active" @endif><a href="{{ route('purchase') }}>Purchase</a>
368           <li @if(Route::currentRouteName() == "pembeli.history.index") class="active" @endif><a href="{{ route('pembeli.history') }}>History</a>
369           <li @if(Route::currentRouteName() == "chat.index" || Route::currentRouteName() == "chat.show") class="active" @endif><a href="{{ route('chat.show') }}>Chat</a>
370           <li @if(Route::currentRouteName() == "profilpembeli") class="active" @endif><a href="{{ route('profilpembeli') }}>Profile</a>
371         </ul>
372       </nav>
373     </div>
374     <div class="col-lg-3">
375       <div class="header__cart">
376         <ul>
377           <!-- <li><a href="#"><i class="fa fa-bell"></i> <span>1</span></a></li> -->
378           @if(count($cartItems) > 0)
379             <li><a href="{{ route('keranjang') }}><i class="fa fa-shopping-bag"></i> <span>{{ count($cartItems) }}</span></a></li>
380           @endif
381         </ul>
382         <!-- <div class="header__cart__price">item: <span>$150.00</span></div> -->
383       </div>
384     </div>
385   </div>
386   <div class="hamberger__open">
387     <i class="fa fa-bars"></i>
388   </div>
389 </div>
390 </header>
391 <!-- Header Section End -->
392
393 <nav aria-label="breadcrumb" class="breadcrumb-nav mb-3">
394   <div class="container">
395     <div><!-- End .container -->
396   </nav><!-- End .breadcrumb-nav -->
397   <div class="page-content">
398     <div class="container">
399       <div class="row">
400         <link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet" />
401
402         <div class="container">
403           <h3 class="text-center">{{$firstChat->Umkm->nama}}</h3>
404           <div class="messaging">
405             <div class="inbox_msg">
406               <div class="inbox_people">
407                 <div class="headind_srch">
408                   <div class="recent_heading">
409                     |   <h4>Recent</h4>
410                   </div>
411                 </div>
412                 <div class="inbox_chat">
413                   @foreach ($chats as $chat)
414                     @if ($chat->id == $firstChat->id)
415                       <div class="chat_list active_chat">
416                         <div class="chat_people">
417                           <div class="chat_img"> <img @if ($chat->Penjual->img == NULL)
418                             src="{{ asset('templates/img/user.png') }}"
419                           @else
420                             src="{{ asset('foto_profil/' . $chat->Penjual->img) }}"
421                           @endif alt="sunil" class="img-circle" width="100" height="50">
422                         </div>
423                         <div class="chat_ib">
424                           <a href="{{ route('chat.show', $chat->id) }}>
425                             <h3>{{ $chat->Penjual->firstname }} {{ $chat->Penjual->lastname }}</h3>
426                             <span class="chat_date">{{ $chat->updated_at->format('H:i A, d M Y') }}</span>
427                           </a>
428                           <p>{{ $chat->lastChat()->message }}</p>
429                         </div>
430                       </div>
431                     @else
432                     <div class="chat_list">
433

```

```

434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508

```

<div style="background-color: #ff9800; position

```

518     </div>
519   </footer>
520   <!-- Footer Section End -->
521   <!-- Js Plugins -->
522   <script src="{{ asset('frontend/js/jquery-3.3.1.min.js') }}"></script>
523   <script src="{{ asset('frontend/js/bootstrap.min.js') }}"></script>
524   <script src="{{ asset('frontend/js/jquery.nice-select.min.js') }}"></script>
525   <script src="{{ asset('frontend/js/jquery-ui.min.js') }}"></script>
526   <script src="{{ asset('frontend/js/slicknav.js') }}"></script>
527   <script src="{{ asset('frontend/js/mixitup.min.js') }}"></script>
528   <script src="{{ asset('frontend/js/owl.carousel.min.js') }}"></script>
529   <script src="{{ asset('frontend/js/main.js') }}"></script>
530
531   <script>
532     function sendMessage(id, pembeli, penjual, status) {
533       let message = $('.write_msg').val();
534       console.log(pembeli)
535       let base_url = "{{URL('api/message/send_message')}}";
536       $.ajax({
537
538         url: base_url + "?pembeli=" + pembeli + "&penjual=" + penjual + "&id=" + id + "&message=" + message + "&status=" + status,
539         dataType: 'json',
540         cache: false,
541         dataSrc: '',
542         success: function (data) {
543           var div = $(`<div class="outgoing_msg">
544             <div class="sent_msg">
545               <p>+message+</p>
546               <span class="time_date">+data.created_at+</span>
547             </div>
548           </div>`);
549
550           $('.msg_history').append(div);
551         }
552       });
553     }
554   </script>
555   </body>
556 </html>

```

## 2. Halaman *checkout*

```

202   <!-- Checkout Section Begin -->
203   <section class="checkout spad">
204     <div class="container">
205       <div class="row">
206       </div>
207       <div class="checkout__form">
208         <h4>Billing Details</h4>
209         <form method="POST" action="{{ route('checkout.update') }}" enctype="multipart/form-data">
210           @csrf
211           <div class="row">
212             <div class="col-lg-8 col-md-6">
213               <div class="row">
214                 <div class="col-lg-6">
215                   <div class="checkout__input">
216                     <p>First Name<span></span></p>
217                     <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ :>
218                     @error('firstname')
219                     {{ $message }}
220                     @enderror
221                   </div>
222                 </div>
223                 <div class="col-lg-6">
224                   <div class="checkout__input">
225                     <p>Last Name<span></span></p>
226                     <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ :>
227                     @error('lastname')
228                     {{ $message }}
229                     @enderror
230                   </div>
231                 </div>
232               </div>
233             </div>
234           </div>
235         </div>
236       <div class="row">
237         <div class="col-lg-6">
238           <div class="checkout__input">
239             <p>No Handphone<span>*</span></p>

```

```

241         <input name="handphone" type="text" class="form-control" required="required"/>
242         <div class="text-danger">
243             @error('handphone')
244                 {{ $message }}
245             @enderror
246         </div>
247     </div>
248     <div class="col-lg-6">
249         <div class="checkout__input">
250             <p>Email<span>*</span></p>
251             <input name="email" type="email" class="form-control" required="required"/>
252             <div class="text-danger">
253                 @error('email')
254                     {{ $message }}
255                 @enderror
256             </div>
257         </div>
258     </div>
259     <div class="checkout__input">
260         <p>Province<span>*</span></p>
261         <select class="form-control" name="provinsi" id="provinsi">
262             <option value="">Provinsi</option>
263             @foreach ($provinsis as $provinsi => $value)
264                 <option value="{{ $provinsi }}>{{ $pembeli->province }}</option>
265             @endforeach
266         </select>
267         <div class="text-danger"></div>
268     </div>
269     <div class="checkout__input">
270         <p>City<span>*</span></p>
271         <select class="form-control" name="kota" id="kota">
272             </div>
273             <div class="text-danger"></div>
274         </div>
275         <div class="checkout__input">
276             <p>Address<span>*</span></p>
277             <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $pembeli->alamat }}"/>
278             <div class="text-danger">
279                 @error('alamat')
280                     {{ $message }}
281                 @enderror
282             </div>
283         </div>
284         <div class="checkout__input">
285             <p>Pilih Ekspedisi<span>*</span></p>
286             <select class="form-control" name="ekspedisi" id="ekspedisi" required="required">
287                 <option value="" disabled selected>Pilih Ekspedisi</option>
288                 @foreach ($ekspedisis as $id => $nama)
289                     <option value="{{ $id }}>{{ $nama }}</option>
290                 @endforeach
291             </select>
292             <div class="text-danger">
293                 @error('ekspedisi')
294                     {{ $message }}
295                 @enderror
296             </div>
297         </div>
298         <div class="form-group">
299             <div class="control">
300                 <button type="submit" class="btn btn-primary mr-2">Submit</button>
301             </div>
302         </div>
303     </div>
304     </div>
305     </div>
306     </div>
307     </div>
308     </div>
309     </div>
310     </div>
311     </div>
312 </form>
313     <div class="col-lg-4 col-md-6">
314         <div class="checkout__order">
315             <form method="POST" action="{{ route('place.order') }}" enctype="multipart/form-data">
316                 @csrf
317                 <!-- Loop through the $keranjangIds -->
318                 @foreach ($keranjangIds as $keranjangId)
319                     <input type="hidden" name="keranjang_ids[]" value="{{ $keranjangId }}>
320                 @endforeach

```

```

321 | <h4>Your Order</h4>
322 | <div class="checkout__order_products">Products <span>Total</span></div>
323 |
324 |
325 | @foreach ($produkItems as $key => $produk)
326 |
327 |     <li>
328 |         {{ $produk }}
329 |         <span>Rp{{ $hargas[$cartItems[$key]->id] * $cartItems[$key]->jumlah }},00</span>
330 |     </li>
331 | @endforeach
332 | </ul>
333 | </ul>
334 | <div class="checkout__order_subtotal">Subtotal <span>Rp({ $subtotal }},00</span></div>
335 | <div class="checkout__input">
336 |     <p>Pilih Layanan<span></span></p>
337 |     <select class="form-control" name="layanan" id="layanan">
338 |         <option value="" disabled selected>Pilih Layanan</option>
339 |         @if ($cekongkir)
340 |             @foreach ($cekongkir as $result)
341 |                 @if (isset($result['service']) && isset($result['cost'][0]['etd']) && isset($result['cost'][0]['value']))
342 |                     <option value="{{ $result['service'] }} | {{ $result['cost'][0]['etd'] }} | {{ $result['cost'][0]['value'] }}"
343 |                         | {{ $result['service'] }} | {{ $result['cost'][0]['etd'] }} | {{ $result['cost'][0]['value'] }}>
344 |                 @endif
345 |             @endforeach
346 |         @endif
347 |     </select>
348 |     <div class="text-danger">
349 |         @error('layanan')
350 |             {{ $message }}
351 |         @enderror
352 |     </div>
353 | </div>
354 | <div class="checkout__order_total">Biaya Pengiriman <span id="biayaPengiriman">Rp0,-</span></div>
355 | <div class="checkout__order_total">Total <span id="total">Rp({ $subtotal }},00</span></div>
356 | <!-- Hidden input field for the 'total' value -->
357 | <input type="hidden" name="total" id="totalValue" value="{{ $subtotal }}">
358 | <button type="submit" class="site-btn">PLACE ORDER</button>
359 | </div>

392 | <script>
393 |
394 | // Listen for changes in the select element (form pertama)
395 | document.getElementById('ekspedisi').addEventListener('change', function () {
396 |     // Get the selected value from the select element
397 |     var selectedEkspedisiId = this.value;
398 |
399 |     // Update the hidden input field for ekspedisi_id (form kedua)
400 |     document.querySelector('input[name="ekspedisi_id"]').value = selectedEkspedisiId;
401 | });
402 |
403 | document.getElementById('layanan').addEventListener('change', function() {
404 |     var selectedLayanan = this.value;
405 |     var harga = selectedLayanan.split('|')[2].trim();
406 |     var biayaPengiriman = document.getElementById('biayaPengiriman');
407 |     biayaPengiriman.textContent = 'Rp.' + harga + ',-' ;
408 |
409 |     // Mengambil nilai subtotal
410 |     var subtotal = parseInt('{{ $subtotal }}');
411 |
412 |     // Menghitung total
413 |     var total = subtotal + parseInt(harga);
414 |
415 |     // Memperbarui nilai total
416 |     var totalElement = document.getElementById('total');
417 |     totalElement.textContent = 'Rp' + total + ',00';
418 |
419 |     // Set the calculated total to the hidden input field
420 |     document.getElementById('totalValue').value = total;
421 | });
422 |
423 | $(document).ready(function() {
424 |     $('select[name="provinsi"]').on('change', function() {
425 |         let provinsiId = $(this).val();
426 |         if (provinsiId) {
427 |             jQuery.ajax({
428 |                 url: '/provinsi/' + provinsiId + '/cityy',
429 |                 type: "GET",
430 |                 dataType: "json",
431 |                 success: function(data) {
432 |                     $('select[name="kota"]').empty();
433 |                     $.each(data, function(key, value) {
434 |                         $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
435 |                     });
436 |                 },
437 |                 error: function(xhr, status, error) {
438 |                     console.error(error);
439 |                 }
440 |             });
441 |         } else {
442 |             $('select[name="kota"]').empty();
443 |         }
444 |     });
445 | });

446 |
447 |
448 |
449 | </script>

```

### 3. Halaman keranjang

```
200 <!-- Shoping Cart Section Begin -->
201 <section class="shoping-cart spad">
202   <div class="container">
203     <div class="row">
204       <div class="col-lg-12">
205         <div class="shoping_cart_table">
206           <form action="{{ route('update-cart') }}" method="POST">
207             @csrf
208             @method('PATCH')
209             <table>
210               <thead>
211                 <tr>
212                   <th class="shoping_product">Products</th>
213                   <th>Price</th>
214                   <th>Quantity</th>
215                   <th>Total</th>
216                   <th></th>
217                 </tr>
218               </thead>
219               <tbody>
220                 @php
221                   $subtotal = 0;
222                 @endphp
223                 @foreach($cartItems as $item)
224                   <tr id="row-{{ $item->id }}">
225                     <td class="shoping_cart_item">
226                       
227                       <h5>{{ $item->produk->nama }}</h5>
228                     </td>
229                     <td class="shoping_cart_price">
230                       @php
231                         // Fetch the 'harga' from the 'produk' relationship based on the selected 'kemasan_id'
232                         $harga = $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
233                       @endphp
234                       Rp{{ $harga }},00
235                     </td>
236                     <td class="shoping_cart_quantity">
237                       <div class="quantity">
238                         <div class="pro-qty">
239                           <input type="number" name="jumlah[{{ $item->id }}]" value="{{ $item->jumlah }}"/>
240                         </div>
241                       </div>
242                     </td>
243                     <td class="shoping_cart_total">
244                       Rp{{ $harga * $item->jumlah }},00
245                     </td>
246                     <td class="shoping_cart_item_close">
247                       <a href="javascript:void(0)" class="remove-from-cart" data-id="{{ $item->id }}">
248                         <span class="icon_close"></span>
249                       </a>
250                     </td>
251                   </tr>
252                 @php
253                   $subtotal += $harga * $item->jumlah;
254                 @endphp
255               @endforeach
256             @endforeach
257           </tbody>
258         </table>
259       </div>
260     </div>
261   </div>
262 </div>
263 <div class="row">
264   <div class="col-lg-12">
265     <div class="shoping_cart_btns">
266       <button>
267         <a href="/" type="submit" class="primary-btn cart-btn">CONTINUE SHOPPING</a></button>
268         <button class="primary-btn cart-btn cart-btn-right"><span class="icon_loading"></span> Update Cart</button>
269       </div>
270     </div>
271   </div>
272 </form>
273 <div class="col-lg-6">
274   <div class="shoping_continue">
275     <div class="shoping_discount">
276       </div>
277     </div>
278   </div>
279 </div>
```

```

276           </div>
277       </div>
278   </div>
279   <div class="col-lg-6">
280     <div class="shopping_checkout">
281       <h5>Cart Total</h5>
282       <ul>
283         <li>Subtotal <span>Rp{{ $subtotal }},00</span></li>
284       </ul>
285
286       <a href="{{ route('checkout') }}" class="primary-btn">PROCEED TO CHECKOUT</a>
287     </div>
288   </div>
289 </div>
290 </div>
291
292 </div>
293 </div>
294 <!-- Shopping Cart Section End -->
295
296 <!-- Footer Section Begin -->
297 <footer class="footer spad">
298   <div class="container">
299     <div class="row">
300       <div class="col-lg-12">
301         <div class="footer_copyright">
302           <div class="footer_copyright_text"><p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --></p><br>Marketplace UMKM, Makanan Khas Olahan Sumatra Barat.</div>
303           <div class="footer_copyright_payment"><p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --></p><br>This app made by Distifia Oktari</div>
304           <div class="footer_copyright_payment"><p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --></p></div>
305         </div>
306       </div>
307     </div>
308   </div>
309 </div>
310 </div>
311 </div>
312 </div>
313 </div>
314 </div>
315 </div>
316 </div>
317 </div>
318 </div>
319 </div>
320 </div>
321 </div>
322 </div>
323 </div>
324 </div>
325 </div>
326 </div>
327 </div>
328 <script>
329   document.addEventListener('DOMContentLoaded', function() {
330     // Menghitung subtotal berdasarkan produk yang dicentang
331     function calculateSubtotal() {
332       var checkboxes = document.querySelectorAll('input[name="selected_products[]"]');
333       var subtotal = 0;
334       checkboxes.forEach(function(checkbox) {
335         if (checkbox.checked) {
336           var row = checkbox.closest('tr');
337           var harga = parseFloat(row.querySelector('.shopping_cart_price').textContent);
338           var jumlah = parseFloat(row.querySelector('.pro-qty input').value);
339           subtotal += harga * jumlah;
340         }
341       });
342       return subtotal.toFixed(2);
343     }
344
345     // Memperbarui subtotal saat checkbox diubah
346     var checkboxes = document.querySelectorAll('input[name="selected_products[]"]');
347     checkboxes.forEach(function(checkbox) {
348       checkbox.addEventListener('change', function() {
349         var subtotal = calculateSubtotal();
350         var subtotalFormatted = 'Rp.' + subtotal.toLocaleString('id-ID', { style: 'currency', currency: 'IDR' });
351         document.querySelector('#subtotal').textContent = subtotalFormatted;
352       });
353     });
354   });
355   // Mengirim permintaan DELETE ke rute remove-from-cart/{id}
356   function removeFromCart(id) {
357     axios.delete('/remove-from-cart/' + id)
358       .then(function(response) {
359         // Hapus baris produk dari tabel
360         var row = document.getElementById('row-' + id);
361         if (row) {
362           row.remove();
363         }
364       })
365       .catch(function(error) {
366         console.error(error);
367       });
368     }
369
370   // Menambahkan event listener untuk menghapus produk dari keranjang
371   document.querySelectorAll('.remove-from-cart').forEach(function(element) {
372     element.addEventListener('click', function() {
373       var id = this.getAttribute('data-id');
374       removeFromCart(id);
375     });
376   });
377
378 </script>

```

#### 4. Halaman payment

```

201 <!-- Payment Section Begin -->
202 <section class="shoping-cart spad">
203   <div class="container">
204     <div class="row">
205       <div class="col-lg-12">
206         @if ($pembelian->status_pembayaran === 0)
207           <form method="POST" action="{{ route('post.payment') }}" enctype="multipart/form-data">
208             @csrf
209           @endif
210         </div>
211       </div>
212       <h2>Terima kasih telah melakukan pembelian.</h2>
213       @if ($pembelian->status_pembayaran === 0)
214         <h5 class="summary-title" id="remaining" style="color:red;text-align:left">
215           | Waktu yang tersisa untuk melakukan pembayaran
216         </h5>
217         <br>
218         <h3>Ringkasan order</h3> <br>
219         <table class="table">
220           <thead>
221             <tr>
222               <th>No</th>
223               <th>Gambar</th>
224               <th>Nama Produk</th>
225               <th>Jumlah</th>
226               <th>Harga Produk</th>
227               <th>Subtotal</th>
228             </tr>
229           </thead>
230           <tbody>
231             @php
232               $index = 1;
233               $total_harga = 0;
234             @endphp
235             @foreach($pembelian->detail_pembelian as $item)
236               <tr>
237                 <td>{{ $index }}</td>
238                 <td class="shoping_cart_item">
239                   
281         <label for="bukti_transfer">Bukti Transfer</label>
282         <input type="file" class="form-control" name="bukti_transfer" id="bukti_transfer">
283     </div>
284
285     <!-- Add hidden input field for the pembelian_id --&gt;
286     &lt;input type="hidden" name="pembelian_id" value="{{ $detail_pembelian-&gt;pembelian_id }}&gt;
287
288     &lt;!-- Close the form and add a submit button --&gt;
289     &lt;div class="form-group"&gt;
290         &lt;input type="submit" class="btn btn-primary" value="Submit Payment" id="submitBtn"&gt;
291     &lt;/div&gt;
292     &lt;/form&gt;
293     &lt;!-- End of the form --&gt;
294     @endif
295
296     @if ($pembelian-&gt;status_pembayaran === 1)
297         &lt;p&gt;Pembayaran mu berhasil. Tunggu konfirmasi pesanan dari toko.&lt;/p&gt;
298     @endif
299     &lt;/div&gt;
300   &lt;/div&gt;
301 &lt;/section&gt;
302 &lt;!-- Shopping Cart Section End --&gt;
303 &lt;!-- Footer Section Begin --&gt;
304 &lt;footer class="footer spad"&gt;
305     &lt;div class="container"&gt;
306         &lt;div class="row"&gt;
307             &lt;div class="col-lg-12"&gt;
308                 &lt;div class="footer__copyright"&gt;
309                     &lt;div class="footer__copyright__text"&gt;&lt;p&gt;!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. Marketplace UMKM, Makanan Khas Olahan Sumatra Barat. --&lt;/p&gt;&lt;/div&gt;
310                     &lt;div class="footer__copyright__payment"&gt;&lt;p&gt;!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. This app made by Distifia Oktari --&lt;/p&gt;&lt;/div&gt;
311                     &lt;div class="footer__copyright__payment"&gt;&lt;p&gt;!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --&lt;/p&gt;&lt;/div&gt;
312                 &lt;/div&gt;
313             &lt;/div&gt;
314         &lt;/div&gt;
315     &lt;/div&gt;
316 &lt;/div&gt;
317 &lt;/div&gt;
318 &lt;/div&gt;
319 &lt;/div&gt;
320
321 &lt;script&gt;
322 var date = new Date('{{ $detail_pembelian-&gt;created_at }}').getTime();
323 var countDownDate = new Date(date);
324 console.log(countDownDate); // penentuan selama 10 menit
325 countDownDate = new Date(countDownDate.getFullYear(), countDownDate.getMonth(), countDownDate.getDate(),
326     countDownDate.getHours(), countDownDate.getMinutes() + 90, countDownDate.getSeconds());
327 countDownDate.setDate(countDownDate.getDate());
328 // Hitungan Mundur Waktu Dilakukan Setiap Satu Detik
329 var x = setInterval(function () {
330     // Mendapatkan Tanggal dan waktu Pada Hari ini
331     var now = new Date().getTime();
332     // Jarak Waktu Antara Hitungan Mundur
333     var distance = countDownDate - now;
334     // Perhitungan waktu hari, jam, menit dan detik
335     // perhitungan komputer diubah ke time (hari, jam , menit, detik)
336     var days = Math.floor((distance % (1000 * 60 * 60 * 24)) / (1000 * 60 * 60));
337     var hours = Math.floor((distance % (1000 * 60 * 60)) / (1000 * 60));
338     var minutes = Math.floor((distance % (1000 * 60)) / 1000);
339     var seconds = Math.floor((distance % 1000) / 1000);
340     // Tampilkan hasilnya di elemen id = "carasingkat"
341     document.getElementById("remaining").innerHTML = "Waktu yang tersisa untuk melakukan pembayaran " +
342         hours + "h " +
343         minutes + "m " + seconds + "s ";
344     // Jika hitungan mundur selesai,
345     if (distance &lt; 0) {
346         clearInterval(x);
347         document.getElementById("remaining").innerHTML = "EXPIRED";
348         $('#submitBtn').prop('disabled', true); // Menonaktifkan tombol submit
349         $('#bukti_transfer').prop('disabled', true); // Menonaktifkan input file
350     }
351 }, 1000);
352 &lt;/script&gt;
</pre>

```

## 5. Halaman *purchase*

```

203 <!-- Purchase Section Begin -->
204 <section class="shoping-cart spad">
205   <div class="container">
206     <div class="row">
207       <div class="col-1-1g-12">
208         <div class="shoping__cart__table">
209           <table>
210             <thead>
211               <tr>
212                 <th>No</th>
213                 <th>Nama UMKM</th>
214                 <th>Total Harga</th>
215                 <th>Status</th>
216                 <th>Aksi</th>
217               </tr>
218             </thead>
219             <tbody>
220               @foreach ($pembelian as $index => $pembelianItem)
221                 @php
222                   $shownPembelianIds = []; // Inisialisasi array untuk menyimpan pembelian_id yang telah ditampilkan
223                 @endphp
224
225               @foreach ($pembelianItem->detail_Pembelian as $detailPembelianItem)
226                 @if (!in_array($detailPembelianItem->pembelian_id, $shownPembelianIds))
227                   @php
228                     $shownPembelianIds[] = $detailPembelianItem->pembelian_id; // Tambahkan pembelian_id ke dalam array
229                   @endphp
230
231                   <tr>
232                     <td>{{ $index + 1 }}</td>
233                     <td>
234                       @if ($detailPembelianItem->produk && $detailPembelianItem->produk->umkm)
235                         {{ $detailPembelianItem->produk->umkm->nama }}
236                       @else
237                         Nama UMKM Tidak Tersedia
238                       @endif
239                     </td>
240
241                     <td>
242                       <td>{{ $pembelianItem->total_harga }}</td>
243                     <td>
244                       @if ($pembelianItem->status_pembayaran == 0 && Carbon\Carbon::now() >= Carbon\Carbon::parse($pembelianItem->created_at)->addMinutes(90))
245                         <span class="badge badge-danger" >Expired</span>
246                       @elseif ($pembelianItem->status_pembayaran == 0)
247                         <span class="badge badge-info" >Menunggu Pembayaran</span>
248                       @endif
249                     <td>
250                       <a href="{{ route('payment', ['id' => $detailPembelianItem->id]) }}" class="btn btn-primary">Bayar</a>
251                       <button class="btn btn-danger" onclick="deletePurchase({{ $pembelianItem->id }})">Delete</button>
252                     </td>
253                   </tr>
254                 @endif
255               @endforeach
256             @endforeach
257           </tbody>
258         </table>
259       </div>
260     </div>
261   </div>
262 </div>
263 </div>
264 </div>
265 </div>
266 </div>
267 </div>
268 </div>
269 <script>
270   function deletePurchase(pembelianId) {
271     // Konfirmasi terlebih dahulu sebelum menghapus
272     if (confirm("Apakah Anda yakin ingin menghapus pembelian ini?")) {
273       // Lakukan permintaan penghapusan menggunakan Axios
274       axios.delete('/purchase/' + pembelianId)
275         .then(response => {
276           // Berhasil dihapus, lakukan tindakan tambahan jika perlu
277           console.log('Pembelian berhasil dihapus');
278           // Memuat ulang halaman untuk memperbarui tampilan data
279           window.location.reload();
280         })
281         .catch(error => {
282           // Tangani kesalahan jika ada
283           console.error('Gagal menghapus pembelian', error);
284         });
285     }
286   }
287 </script>

```

## 6. Halaman *index history*

```

203 <!-- History Section Begin -->
204 <section class="shoping-cart spad">
205   <div class="container">
206     <div class="row">
207       <div class="col-lg-12">
208         <div class="shoping__cart_table">
209           <table>
210             <thead>
211               <tr>
212                 <th>No</th>
213                 <th>Nama UMKM</th>
214                 <th>Total Harga</th>
215                 <th>Status</th>
216                 <th>Aksi</th>
217               </tr>
218             </thead>
219             <tbody>
220               @foreach ($pembelian as $index => $pembelianItem)
221                 @php
222                   $shownPembelianIds = []; // Inisialisasi array untuk menyimpan pembelian_id yang telah ditampilkan
223                 @endphp
224
225               @foreach ($pembelianItem->Detail_Pembelian as $detailPembelianItem)
226                 @if (!in_array($detailPembelianItem->pembelian_id, $shownPembelianIds))
227                   @php
228                     $shownPembelianIds[] = $detailPembelianItem->pembelian_id; // Tambahkan pembelian_id ke dalam array
229                   @endphp
230                 <tr>
231                   <td>{{ $index + 1 }}</td>
232                   <td>
233                     @if ($detailPembelianItem->produk && $detailPembelianItem->produk->umkm)
234                       {{ $detailPembelianItem->produk->umkm->nama }}
235                     @else
236                       Nama UMKM Tidak Tersedia
237                     @endif
238                   </td>
239
240                   <td>Rp{{ $pembelianItem->total_harga }},00</td>
241                   <td>
242                     @if ($pembelianItem->status_pembelian == 1)
243                       <span class="badge badge-primary">Pesanan Disiapkan</span>
244                     @elseif ($pembelianItem->status_pembelian == 0 && $pembelianItem->status_pembayaran == 0 && Carbon\Carbon::now() >= Carbon\Carbon::parse($pembelianItem->created_at))
245                       <span class="badge badge-danger">Expired</span>
246                     @elseif ($pembelianItem->status_pembelian == 0 && $pembelianItem->status_pembayaran == 0)
247                       <span class="badge badge-info">Menunggu Pembayaran</span>
248                     @elseif ($pembelianItem->status_pembelian == 0)
249                       <span class="badge badge-default">Menunggu Konfirmasi</span>
250                     @elseif ($pembelianItem->status_pembelian == 2)
251                       <span class="badge badge-primary">Pesanan Dikirim</span>
252                     @elseif ($pembelianItem->status_pembelian == 3)
253                       <span class="badge badge-danger">Pesanan Ditolak</span>
254                     @elseif ($pembelianItem->status_pembelian == 4)
255                       <span class="badge badge-info">Selesai</span>
256                     @endif
257                   </td>
258                   <td>
259                     <a href="{{ route('pembeli.show.history', ['id' => $detailPembelianItem->id]) }}" class="btn btn-primary">Detail Transaksi</a>
260                   </td>
261                 </tr>
262               @endif
263             @endforeach
264           @endfor
265         </tbody>
266       </table>
267     </div>
268   </div>
269   </div>
270 </section>
271
272 <script>
273   function deletePurchase(pembelianId) {
274     // Konfirmasi terlebih dahulu sebelum menghapus
275     if (confirm("Apakah Anda yakin ingin menghapus pembelian ini?")) {
276       // Lakukan permintaan penghapusan menggunakan Axios
277       axios.delete('/purchase/' + pembelianId)
278         .then(response => {
279           // Berhasil dihapus, lakukan tindakan tambahan jika perlu
280           console.log('Pembelian berhasil dihapus');
281           // Memuat ulang halaman untuk memperbarui tampilan data
282           window.location.reload();
283         })
284         .catch(error => {
285           // Tangani kesalahan jika ada
286           console.error('Gagal menghapus pembelian', error);
287         });
288     }
289   }
290 </script>
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
627
628
629
629
630
631
632
633
634
635
635
636
637
637
638
639
639
640
641
642
642
643
643
644
645
645
646
646
647
647
648
648
649
649
650
650
651
651
652
652
653
653
654
654
655
655
656
656
657
657
658
658
659
659
660
660
661
661
662
662
663
663
664
664
665
665
666
666
667
667
668
668
669
669
670
670
671
671
672
672
673
673
674
674
675
675
676
676
677
677
678
678
679
679
680
680
681
681
682
682
683
683
684
684
685
685
686
686
687
687
688
688
689
689
690
690
691
691
692
692
693
693
694
694
695
695
696
696
697
697
698
698
699
699
700
700
701
701
702
702
703
703
704
704
705
705
706
706
707
707
708
708
709
709
710
710
711
711
712
712
713
713
714
714
715
715
716
716
717
717
718
718
719
719
720
720
721
721
722
722
723
723
724
724
725
725
726
726
727
727
728
728
729
729
730
730
731
731
732
732
733
733
734
734
735
735
736
736
737
737
738
738
739
739
740
740
741
741
742
742
743
743
744
744
745
745
746
746
747
747
748
748
749
749
750
750
751
751
752
752
753
753
754
754
755
755
756
756
757
757
758
758
759
759
760
760
761
761
762
762
763
763
764
764
765
765
766
766
767
767
768
768
769
769
770
770
771
771
772
772
773
773
774
774
775
775
776
776
777
777
778
778
779
779
780
780
781
781
782
782
783
783
784
784
785
785
786
786
787
787
788
788
789
789
790
790
791
791
792
792
793
793
794
794
795
795
796
796
797
797
798
798
799
799
800
800
801
801
802
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
810
811
811
812
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
820
821
821
822
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
830
831
831
832
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
840
841
841
842
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
850
851
851
852
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
860
861
861
862
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
870
871
871
872
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
880
881
881
882
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
890
891
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
900
901
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
910
911
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538

```

## 7. Halaman *show history*

```

180 <!-- Show History Section Begin -->
181 <section class="shoping-cart spad">
182   <div class="container">
183     <div class="row">
184       <div class="col-lg-12">
185         </div>
186       </div>
187     <h2>Terima kasih telah melakukan pembelian.</h2>
188
189     <br>
190     <!-- Add the product table -->
191     <h3>Ringkasan order</h3> <br>
192     <h6>Umkm : {{ $detail_pembelian->produk->umkm->nama }}</h6> <br>
193     <h6>Tanggal Pembelian : {{ date('d-m-Y', strtotime($pembelian->created_at)) }}</h6> <br>
194     <h6> status :
195       @if($pembelian->status_pembelian == 0)
196         <span class="badge badge-info" > Menunggu Konfirmasi </span>
197         @elseif($pembelian->status_pembelian == 1)
198           <span class="badge badge-primary" > Pesanan Disediakan </span>
199         @elseif($pembelian->status_pembelian == 2)
200           <span class="badge badge-primary" > Pesanan Dikirim </span>
201         @elseif ($pembelian->status_pembelian == 3)
202           <span class="badge badge-danger" > Pesanan Ditolak </span>
203         @elseif ($pembelian->status_pembelian == 4)
204           <span class="badge badge-info" > Transaksi Selesai </span>
205         @endif
206       </h6>
207
208       <h6>
209         @if($pembelian->status_pembelian == 0)
210
211         @elseif($pembelian->status_pembelian == 1)
212
213           <br>
214           Noresi : Belum Tersedia
215         @elseif($pembelian->status_pembelian == 2)
216
217           <br>
218
219           Noresi : {{ $pembelian->noresi }}
220         @elseif ($pembelian->status_pembelian == 4)
221
222           <br>
223           Noresi : {{ $pembelian->noresi }}
224         @endif
225       </h6> <br>
226
227       @if($pembelian->status_pembelian == 3 && $pembelian->reject_note)
228         <h6>Alasan Ditolak : {{ $pembelian->reject_note }}</h6>
229       @endif <br>
230       <table class="table">
231         <thead>
232           <tr>
233             <th>No</th>
234             <th>Gambar</th>
235             <th>Nama Produk</th>
236             <th>Jumlah</th>
237             <th>Harga Produk</th>
238             <th>Subtotal</th> <!-- Tambahkan kolom Subtotal -->
239           </tr>
240         </thead>
241         <tbody>
242           @php
243             $index = 1;
244             $total_harga = 0;
245           @endphp
246           @foreach($pembelian->detail_pembelian as $item)
247             <tr>
248               <td>{{ $index }}</td>
249               <td class="shoping_cart_item">
250                 
251               </td>
252               <td>{{ $item->produk->nama }}</td>
253               <td>{{ $item->jumlah }}</td>
254               <td>Rp{{ number_format($item->produk->detail_produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0, ',', '.') }}</td>
255               <td>Rp{{ number_format($item->jumlah * $item->produk->detail_produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0,
256

```

```

257     </tr>
258     @php
259         $index++;
260         // Menghitung total harga dengan menambahkan subtotal setiap produk
261         $total_harga += $item->jumlah * $item->produk->detail_produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
262     @endphp
263     @endforeach
264   </tbody>
265   <tfoot>
266     <tr>
267       <th colspan="5" style="text-align:right">Biaya Pengiriman:</th>
268       <td>Rp{{ number_format($detail_pembelian->pembelian->total_harga - $total_harga, 0, ',', '.') }},00</td>
269     </tr>
270     <tr>
271       <th colspan="5" style="text-align:right">Total Harga:</th>
272       <td>Rp{{ number_format($detail_pembelian->pembelian->total_harga, 0, ',', '.') }},00</td>
273     </tr>
274   </tfoot>
275 </table>
276
277
278
279
280
281
282 </div>
283 </div>
284 </section>

```

## 8. Halaman profil pembeli

```

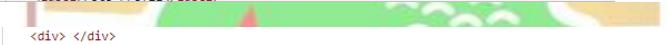
1  @extends('layouts.pembeli')
2  @section('content')
3  <div class="content-wrapper">
4      <section class="section main-section">
5          <div class="grid grid-cols-1 gap-6 lg:grid-cols-2 mb-6">
6
7              <div class="card">
8                  <header class="card-header">
9                      <p class="card-header-title">
10                         <span class="icon"><i class="mdi mdi-account"></i></span>
11                         Profile
12                     </p>
13                 </header>
14                 <div class="card-content">
15                     <div class="image w-28 h-28 mx-auto">
16                         
17                     <hr>
18                         @csrf
19                         <div class="form-group">
20                             <label for="exampleInputName1">Name</label>
21                             <input name="name" type="text" class="form-control" id="name" placeholder="Name" value="@php echo $pembeli->firstname . ' ' . $pembeli->lastname @endphp" />
22                         </div>
23                         <div class="form-group">
24                             <label for="exampleInputEmail1">Email</label>
25                             <input name="email" type="text" class="form-control" id="email" placeholder="Email" value="{{ $pembeli->email }}"/>
26                         </div>
27                         <div class="form-group">
28                             <label for="exampleInputHandphone1">No Handphone</label>
29                             <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $pembeli->handphone }}"/>
30                         </div>
31                         <div class="form-group">
32                             <label for="">Provinsi</label>
33                             <select class="form-control" name="provinsi" id="provinsi" disabled>
34                                 <option value="">Provinsi</option>
35                                 @foreach ($provinsis as $provinsi => $value)
36                                     <option value="{{ $provinsi }}"{!! $pembeli->province_id == $provinsi ? 'selected' : '' !!}>{!! $value !!}
37                                 @endforeach
38                             </select>
39                         <div class="text-danger"></div>
40                     </div>
41                     <div class="form-group">

```

```

~ 41 <div class="form-group">
~ 42 <label for="">Kota</label>
~ 43 <select class="form-control" name="kota" id="kota" disabled>
~ 44   <option value="">Kota</option>
~ 45   <foreach ($kotas as $kota => $value)
~ 46     <option value="{{ $kota }}{{ $pembeli->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
~ 47   </foreach>
~ 48 </select>
~ 49 <div class="text-danger">
~ 50   </div>
~ 51 <div class="form-group">
~ 52   <label for="exampleTextareal">Alamat</label>
~ 53   <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $pembeli->alamat }}" readonly>
~ 54
~ 55 </div>
~ 56 <div class="form-group">
~ 57   <label for="exampleInputEmail3">Email</label>
~ 58   <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}" readonly>
~ 59   <div class="text-danger">
~ 60     @error('email')
~ 61       {{ $message }}
~ 62     @enderror
~ 63   </div>
~ 64 </div>
~ 65 <hr>
~ 66 |   </div>
~ 67 </div>
~ 68 <div class="card">
~ 69   <header class="card-header">
~ 70     <p class="card-header-title">
~ 71       <span class="icon"><i class="mdi mdi-account-circle"></i></span>
~ 72       Edit Profile
~ 73     </p>
~ 74   </header>
~ 75   <div class="card-content">
~ 76     <form method="POST" action="{{ route('profilpembeli.update') }}" enctype="multipart/form-data">
~ 77       @csrf
~ 78       <div class="form-group">
~ 79         <label>Foto Profil</label>

```



```

80       <div> </div>
81       <input type="file" name="img" class="form-control" value="{{ $pembeli->img }}>
82       <!-- <div>
83       <a href="{{ asset('foto_profil/' . $pembeli->img )}}" target="_blank" rel="noopener noreferrer"> {{ $pembeli->img }} </a> </c
84       <div class="text-danger">
85         @error('img')
86           {{ $message }}
87         @enderror
88       </div>
89     </div>
90     <hr>
91     <div class="form-group">
92       <label for="exampleInputName1">Firstname</label>
93       <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $pembeli->firstna
94       <div class="text-danger">
95         @error('firstname')
96           {{ $message }}
97         @enderror
98       </div>
99     </div>
100    <div class="form-group">
101      <label for="exampleInputEmail3">Lastname</label>
102      <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $pembeli->lastname
103
104    </div>
105    <div class="form-group">
106      <label for="exampleInputEmail3">No Handphone</label>
107      <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $pembeli->hanc
108      <div class="text-danger">
109        @error('handphone')
110          {{ $message }}
111        @enderror
112      </div>
113    </div>
114    <div class="form-group">
115      <label for="">Provinsi</label>
116      <select class="form-control" name="provinsi" id="provinsi">
117        <option value="">Provinsi</option>
118        <foreach ($provinces as $provinsi => $value)

```

```

120      @endforeach
121      </select>
122      <div class="text-danger"></div>
123  </div>
124      <div class="form-group">
125          <label for="">Kota</label>
126          <select class="form-control" name="kota" id="kota">
127              <option value="">Kota</option>
128              @foreach ($kotas as $kota => $value)
129                  <option value="{{ $kota }}> {{ $pembeli->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
130              @endforeach
131          </select>
132          <div class="text-danger">
133          </div>
134      </div>
135      <div class="form-group">
136          <label for="exampleTextareal">Alamat</label>
137          <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $pembeli->alamat }}>
138      <div class="text-danger">
139          @error('alamat')
140              {{ $message }}
141          @enderror
142      </div>
143  </div>
144      <div class="form-group">
145          <label for="exampleInputEmail3">Email</label>
146          <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}>
147          <div class="text-danger">
148              @error('email')
149                  {{ $message }}
150              @enderror
151          </div>
152  </div>
153      <hr> <br>
154      <div class="form-group">
155          <div class="control">
156              <button type="submit" class="btn btn-primary mr-2">Submit</button>
157          </div>
158      </div>
159  </form>
160  </div>
161  </div>
162  </div>
163  <div class="card">
164      <header class="card-header">
165          <p class="card-header-title">
166              <span class="icon"><i class="mdi mdi-lock"></i></span>
167              Change Password
168          </p>
169      </header>
170      <div class="card-content">
171          <form action="{{ route('change.password') }}" method="POST">
172              @csrf
173
174          <div class="form-group">
175              <label for="current_password">Current Password</label>
176              <input name="current_password" type="password" class="form-control" id="current_password" placeholder="Current Password">
177          </div>
178
179          <div class="form-group">
180              <label for="new_password">New Password</label>
181              <input name="new_password" type="password" class="form-control" id="new_password" placeholder="New Password">
182          </div>
183
184          <div class="form-group">
185              <label for="confirm_password">Confirm Password</label>
186              <input name="confirm_password" type="password" class="form-control" id="confirm_password" placeholder="Confirm Password">
187          </div>
188
189          <div class="form-group">
190              <div class="control">
191                  <button type="submit" class="btn btn-primary mr-2">Submit</button>
192              </div>
193          </div>
194      </div>
195  </div>
196  </div>
197

```

```

198 </div>
199 @endsection
200
201 @section('script')
202 <!-- Include the jQuery library -->
203 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
204 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
205
206 <script>
207 $(document).ready(function() {
208     $('select[name="provinsi"]').on('change', function() {
209         let provinsiId = $(this).val();
210         if (provinsiId) {
211             jQuery.ajax({
212                 url: '/provinsi/' + provinsiId + '/cityy',
213                 type: "GET",
214                 dataType: "json",
215                 success: function(data) {
216                     $('select[name="kota"]').empty();
217                     $.each(data, function(key, value) {
218                         $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
219                     });
220                 },
221                 error: function(xhr, status, error) {
222                     console.error(error);
223                 }
224             });
225         } else {
226             $('select[name="kota"]').empty();
227         }
228     });
229 });
230 </script>

```

#### d. Bagian Penjual

##### 1. Halaman *index chat*

```

1 @extends('layouts.penjual')
2 @section('css')
3 <style>
4     .container {
5         max-width: 1170px;
6         margin: auto;
7     }
8
9     img {
10         max-width: 100%;
11     }
12
13     .inbox_people {
14         background: #f8f8f8 none repeat scroll 0 0;
15         float: left;
16         overflow: hidden;
17         width: 40%;
18         border-right: 1px solid #c4c4c4;
19     }
20
21     .inbox_msg {
22         border: 1px solid #c4c4c4;
23         clear: both;
24         overflow: hidden;
25     }
26
27     .top_spac {
28         margin: 20px 0 0;
29     }
30
31     .recent_heading {
32         float: left;
33         width: 40%;
34     }
35
36     .srch_bar {
37         display: inline-block;
38         text-align: right;
39     }

```

```

41 }
42
43 .headind_srch {
44   padding: 10px 29px 10px 20px;
45   overflow: hidden;
46   border-bottom: 1px solid #c4c4c4;
47 }
48
49 .recent_heading h4 {
50   color: #05728f;
51   font-size: 21px;
52   margin: auto;
53 }
54
55 .srch_bar input {
56   border: 1px solid #cdcdcd;
57   border-width: 0 0 1px 0;
58   width: 80%;
59   padding: 2px 0 4px 6px;
60   background: none;
61 }
62
63 .srch_bar .input-group-addon button {
64   background: rgba(0, 0, 0, 0) none repeat scroll 0 0;
65   border: medium none;
66   padding: 0;
67   color: #707070;
68   font-size: 18px;
69 }
70
71 .srch_bar .input-group-addon {
72   margin: 0 0 0 -27px;
73 }
74
75 .chat_ib h5 {
76   font-size: 15px;
77   color: #464646;
78   margin: 0 0 8px 0;
79 }
80
81 .chat_ib h5 span {
82   font-size: 13px;
83   float: right;
84 }
85
86 .chat_ib p {
87   font-size: 14px;
88   color: #989898;
89   margin: auto
90 }
91
92 .chat_img {
93   float: left;
94   width: 11%;
95 }
96
97 .chat_ib {
98   float: left;
99   padding: 0 0 0 15px;
100  width: 88%;
101 }
102
103 .chat_people {
104   overflow: hidden;
105   clear: both;
106 }
107
108 .chat_list {
109   border-bottom: 1px solid #c4c4c4;
110   margin: 0;
111   padding: 18px 16px 10px;
112 }
113
114 .inbox_chat {
115   height: 550px;
116   overflow-y: scroll;
117 }
118
119 .active_chat {

```

```

120 |     background: #ebebeb;
121 | }
122 |
123 .incoming_msg {
124 |     display: inline-block;
125 |     width: 6%;
126 | }
127 |
128 .received_msg {
129 |     display: inline-block;
130 |     padding: 0 0 0 10px;
131 |     vertical-align: top;
132 |     width: 92%;
133 | }
134 |
135 .received_withd_msg p {
136 |     background: #ebebeb none repeat scroll 0 0;
137 |     border-radius: 3px;
138 |     color: #646464;
139 |     font-size: 14px;
140 |     margin: 0;
141 |     padding: 5px 10px 5px 12px;
142 |     width: 100%;
143 | }
144 |
145 .time_date {
146 |     color: #747474;
147 |     display: block;
148 |     font-size: 12px;
149 |     margin: 8px 0 0;
150 | }
151 |
152 .received_withd_msg {
153 |     width: 57%;
154 | }
155 |
156 .msgs {
157 |     float: left;
158 |     padding: 30px 15px 0 25px;
159 |     width: 60%;
160 | }
161 |
162 .sent_msg p {
163 |     background: #05728f none repeat scroll 0 0;
164 |     border-radius: 3px;
165 |     font-size: 14px;
166 |     margin: 0;
167 |     color: #fff;
168 |     padding: 5px 10px 5px 12px;
169 |     width: 100%;
170 | }
171 |
172 .outgoing_msg {
173 |     overflow: hidden;
174 |     margin: 26px 0 26px;
175 | }
176 |
177 .sent_msg {
178 |     float: right;
179 |     width: 46%;
180 | }
181 |
182 .input_msg_write input {
183 |     background: rgba(0, 0, 0, 0) none repeat scroll 0 0;
184 |     border: medium none;
185 |     color: #4c4c4c;
186 |     font-size: 15px;
187 |     min-height: 48px;
188 |     width: 100%;
189 | }
190 |
191 .type_msg {
192 |     border-top: 1px solid #c4c4c4;
193 |     position: relative;
194 | }
195 |
196 .msg_send_btn {
197 |     background: #05728f none repeat scroll 0 0;
198 |     border: medium none;

```

```

199     border-radius: 50%;  
200     color: #ffff;  
201     cursor: pointer;  
202     font-size: 17px;  
203     height: 33px;  
204     position: absolute;  
205     right: 0;  
206     top: 11px;  
207     width: 33px;  
208   }  
209  
210   .messaging {  
211     padding: 0 0 50px 0;  
212   }  
213  
214   .msg_history {  
215     height: 516px;  
216     overflow-y: auto;  
217   }  
218 </style>  
219 @endsection  
220  
221 @section('content')  
222 <div class="content-wrapper">  
223 <nav aria-label="breadcrumb" class="breadcrumb-nav mb-3">  
224   <div class="container">  
225     </div><!-- End .container -->  
226 </nav><!-- End .breadcrumb-nav -->  
227 <div class="page-content">  
228   <div class="container">  
229     <div class="row">  
230       <link href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" rel="stylesheet" />  
231  
232     <div class="container">  
233       <h3 class="text-center">{{ $firstChat->Pembeli->firstname }}</h3>  
234       <div class="messaging">  
235         <div class="inbox_msg">  
236           <div class="inbox_people">  
237             <div class="headind_srch">  
238               <div class="recent_heading">  
239                 <h4>Recent</h4>  
240               </div>  
241             </div>  
242             <div class="inbox_chat">  
243               @foreach ($chats as $chat)  
244                 @if ($chat->id == $firstChat->id)  
245                   <div class="chat_list active_chat">  
246                     <div class="chat_people">  
247                       <div class="chat_img"> <img @if ($chat->Pembeli->img == NULL) src="{{ asset('templates/img/user.png') }}" @else src="{{ asset('foto_profil/'.$chat->Pembeli->img) }}" @endif alt="sunil" class="img-circle" width="100" height="50">  
248                     </div>  
249                     <div class="chat_ib">  
250                       <a href="{{ route('chat.show.penjual', $chat->id) }}">  
251                         <h5>{{ $chat->Pembeli->firstname }} {{ $chat->Pembeli->lastname }}</h5>  
252                         <span class="chat_date">{{ $chat->updated_at->format('H:i A, d M Y') }}</span>  
253                       </a>  
254                       <p>{{ $chat->lastChat()->message }}</p>  
255                     </div>  
256                   </div>  
257                 </div>  
258               @else  
259                 <div class="chat_list">  
260                   <div class="chat_people">  
261                     <div class="chat_img"> <img @if ($chat->Pembeli->img == NULL) src="{{ asset('templates/img/user.png') }}" @else src="{{ asset('foto_profil/'.$chat->Pembeli->img) }}" @endif alt="sunil" class="img-circle" width="100" height="50">  
262                     </div>  
263                     <div class="chat_ib">  
264                       <a href="{{ route('chat.show.penjual', $chat->id) }}">  
265                         <h5>{{ $chat->Pembeli->firstname }} {{ $chat->Pembeli->lastname }}</h5>  
266                         <span class="chat_date">{{ $chat->updated_at->format('H:i A, d M Y') }}</span>  
267                       </a>  
268                     </div>  
269                   </div>  
270                 </div>  
271               @endif  
272             </div>  
273             <div class="headind_srch">  
274               <div class="recent_heading">  
275                 <h4>Recent</h4>  
276               </div>  
277             </div>

```

```
278 |         |         |     </a>
279 |         |         |     |     <p>{{ $chat->lastChat()->message }}</p>
280 |         |         |     </div>
281 |         |     </div>
282 |     @endif
283 |
284 |     @endforeach
285 |   </div>
286 |   </div>
287 | </div class="msg_history">
288 | @foreach ($firstChat->chatDetail as $chatDetail)
289 |     @if ($chatDetail->sender == Auth::user()->id)
290 |         <div class="outgoing_msg">
291 |             <div class="sent_msg">
292 |                 <p>{{ $chatDetail->message }}</p>
293 |                 <span class="time_date">{{ $chatDetail->created_at->format('H:i A | d M Y') }}</span>
294 |             </div>
295 |         </div>
296 |     @else
297 |     @else
298 |         <div class="incoming_msg">
299 |             <div class="incoming_msg_img"> <img @if ($chatDetail->Pembeli->Pembeli->img == NULL)
300 |                 |     src="{{ asset('templates/img/user.png') }}"
301 |             @else
302 |                 |     src="{{ asset('foto_profil/' . $chatDetail->Pembeli->Pembeli->img) }}"
303 |             @endif
304 |                 |     alt="sunil" class="img-circle" width="100" height="50"
305 |             </div>
306 |             <div class="received_msg">
307 |                 <div class="received_withd_msg">
308 |                     <p>{{ $chatDetail->message }}</p>
309 |                     <span class="time_date">{{ $chatDetail->created_at->format('H:i A | d M Y') }}</span>
310 |                 </div>
311 |             </div>
312 |         @endif
313 |     @endif
314 |     @endforeach
315 |   </div>
316 | </div class="msg_history">
317 |
318 | <div class="type_msg">
319 |     <div class="input_msg_write">
320 |         <input type="text" class="write_msg" placeholder="Type a message" />
321 |         <button class="msg_send_btn" type="button" onclick="sendMessage({{$firstChat->Umkm->id}}, {{$firstChat->Pembeli->Pembeli->id}})">
322 |             |     <i class="fa fa-paper-plane-o" aria-hidden="true"></i>
323 |         </button>
324 |     </div>
325 |     </div>
326 |     </div>
327 |     </div>
328 | </div><!-- End .row --&gt;
329 | &lt;/div&gt;<!-- End .container --&gt;
330 | &lt;/div&gt;
331 | &lt;/div&gt;
332 | &lt;/div&gt;
333 | @endsection
334 |
335 | @section('script')
336 | &lt;script&gt;
337 |     function sendMessage(id, pembeli, penjual) {
338 |         let message = $('.write_msg').val();
339 |
340 |         let base_url = "{{URL('api/message/send_message')}}";
341 |         $.ajax({
342 |
343 |             url: base_url + "?pembeli=" + pembeli + "&amp;penjual=" + penjual + "&amp;id=" + id + "&amp;message=" + message,
344 |             dataType: 'json',
345 |             cache: false,
346 |             dataSrc: '',
347 |             success: function (data) {
348 |                 var div = $(&lt;div class="outgoing_msg"&gt;
349 |                             |         &lt;div class="sent_msg"&gt;
350 |                             |             |         &lt;p&gt;+message+&lt;/p&gt;
351 |                             |             |         &lt;span class="time_date"&gt;+data.created_at+&lt;/span&gt;
352 |                             |         &lt;/div&gt;
353 |                             &lt;/div&gt;)
354 |
355 |                 $('.msg_history').append(div);
356 |             }
357 |         });
358 |     }
359 | &lt;/script&gt;
360 | @endsection</pre>
```

## 2. Halaman *index* umkm

```
1  @extends('layouts.penjual')
2  @section('content')
3  <div class="content-wrapper">
4  <section class="content">
5
6      @if (count($umkmData) == 0)
7          <div class="container-fluid">
8              <div class="row">
9                  <div class="col-12">
10                     <div class="card">
11                         <div class="card-header">
12                             <h3 class="card-title">List Data Ukm</h3>
13                         @if (Auth::user()->penjual->ktp)
14                             <a style="float:right; margin-right:5px;" href="{{ route('umkm.create') }}>
15                                 <button class="btn btn-primary" type="button">Tambah Ukm</button>
16                             </a>
17                         @else
18                             <a style="float:right; margin-right:5px;" href="#">
19                                 <button onclick="warning()" class="btn btn-primary" type="button">Tambah Ukm</button>
20                             </a>
21                         @endif
22
23                     </div>
24                     <div class="card-body">
25                         <h3><center>Data Ukm Masih Kosong !</center></h3>
26                         <p><center>Ukm kamu telah terkonfirmasi oleh admin, mari isi data umkm anda dengan lengkap dan benar</center></p>
27                     </div>
28                     <!-- /.card -->
29                 </div>
30             <!-- /.col -->
31         </div>
32     </div>
33     @else
34         <div class="container-fluid">
35             <div class="row">
36                 <div class="col-12">
37                     <div class="card">
38                         <div class="card-header">
39                             <h3 class="card-title">List Data Ukm</h3>
40
41                             </h3>
42                             @if (Auth::user()->penjual->ktp)
43                                 <a style="float:right; margin-right:5px;" href="{{ route('umkm.create') }}>
44                                     <button class="btn btn-primary" type="button">Tambah Ukm</button>
45                                 </a>
46                             @else
47                                 <a style="float:right; margin-right:5px;" href="#">
48                                     <button onclick="warning()" class="btn btn-primary" type="button">Tambah Ukm</button>
49                                 </a>
50                             @endif
51                         </div>
52                         <div class="card-body">
53                             <div class="row sortable-card">
54                                 @foreach ($umkmData as $umkm)
55                                     <div class="col-4 col-sm-6 col-md-4 d-flex align-items-stretch flex-column">
56                                         <div class="card bg-light d-flex flex-fill">
57                                             <div class="card-header text-muted border-bottom-0">
58                                                 <h2 class="lead"><b>{$umkm->nama}</b></h2>
59                                             </div>
60                                             <div class="card-body pt-0">
61                                                 <div class="row">
62                                                     <div class="col-7">
63
64                                         <p class="text-muted text-sm"><b></b></p>
65                                         <ul class="list-group mb-0 fa-ul text-muted">
66                                             <li class="medium"><span class="fa-li"><i class="mdi mdi-map-marker"></i></span>
67                                                 Alamat: {$umkm->alamat}</li>
68                                             <li class="medium"><span class="fa-li"><i class="mdi mdi-cellphone-basic"></i></span> Phone:
69                                                 {$umkm->handphone}</li>
70                                         </ul>
71                                     </div>
72                                     <div class="col-5 text-center">
73                                         
74                                         | alt="umkm" class="img-circle img-fluid" style="width:500px; height:100px;">
75                                         @if($umkm->status == 0)
76                                             <span class="badge badge-info" style="width:100%">Pending</span>
77                                         @elseif ($umkm->status == 1)
78                                             <span class="badge badge-success" style="width:100%">Aktif</span>
79                                         </div>
80                                     </div>
81                                 </div>
82                             </div>
83                         </div>
84                     </div>
85                 </div>
86             </div>
87         </div>
88     </div>
89 
```

```

80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158

```

---

```

        @else
            <span class="badge badge-danger" style="width:100%">Ditolak</span>
        @endif
    </div>
</div>
<div class="card-footer">
    <div class="text-right">
        <a class="btn btn-sm btn-primary" @if ($umkm->status ==
            1)
            href="{{ route('umkm.show', $umkm->id) }}"
        @elseif ($umkm->status == 2) onclick="edit_umkm({{$umkm->id}})"
            href="javascript:void(0)"
        @elseif ($umkm->status == 0) onclick="confirm_umkm()"
            href="javascript:void(0)"
        @endif>
            View Umkm &nbsp;<i class="fas fa-arrow-right"></i>
        </a>
    </div>
</div>
</div>
</div>
@endforeach
</div>
<!-- /.card-body -->
</div>
<!-- /.card -->
</div>
<!-- /.col -->
</div>
</div>
@endif
</div>
</section>
</div>
{!! Form::open(['method'=>'PATCH', 'route' => ['umkm.edit', 0], 'style' =>'display:none','id'=>'edit_umkm']) !!}
{!! Form::close() !!}
@endsection

120 @section('script')
121 <script type="text/javascript">
122     function edit(id, name, address, latitude, longitude) {
123         if ($("#edit").is(":visible")) {
124             $("#edit").hide('500');
125         } else {
126             console.log(id);
127             $("#edit").show('500');
128             $("#_name").val(name);
129             $("#_address").val(address);
130             $("#_latitude").val(latitude);
131             $("#_longitude").val(longitude);
132             $("#form-update").attr('action', "{{route('umkm.index')}}/" + id);
133         }
134     }
135
136     function show(id, name, address, latitude, longitude) {
137         $("#_name").val(name);
138         $("#_address").val(address);
139         $("#_latitude").val(latitude);
140         $("#_longitude").val(longitude);
141     }
142
143     function confirm_umkm(id) {
144         swal({
145             title: "Umkm kamu belum dikonfirmasi !",
146             text: "Harap menunggu admin untuk mengkonfirmasi umkm kamu",
147             type: "warning",
148             confirmButtonColor: "#f02b2b",
149             confirmButtonText: "OK",
150             closeOnConfirm: false,
151             closeOnCancel: false
152         });
153     }
154
155     function edit_umkm(id) {
156         swal({
157             title: "Apakah kamu ingin melengkapi data umkm?",
158             text: "Data umkm yang telah dilengkapi akan diajukan lagi untuk dikonfirmasi oleh admin!",

```

```

159     type: "warning",
160     showCancelButton: true,
161     confirmButtonColor: "#5cc744",
162     confirmButtonText: "Ya, konfirmasi!",
163     cancelButtonText: "Tidak, batalkan!",
164     closeOnConfirm: false,
165     closeOnCancel: false
166   },
167   function (isConfirm) {
168     if (isConfirm) {
169       swal("Lengkapi data!", "Kamu akan dipindahkan ke halaman untuk melengkapi data umkm.", "success");
170       location.href = "{{route('umkm.index')}}/" + id + "/edit?data=request";
171     } else {
172       swal("Dibatalkan", "Kamu batal melengkapi data umkm", "error");
173     }
174   });
175 }
176
177 function warning() {
178   swal({
179     title: "Apakah kamu ingin melengkapi data ktp anda?",
180     text: "Sebelum menambahkan umkm, anda harus menginputkan ktp anda !",
181     type: "warning",
182     showCancelButton: true,
183     confirmButtonColor: "#5cc744",
184     confirmButtonText: "Ya, lengkapi!",
185     cancelButtonText: "Tidak, batalkan!",
186     closeOnConfirm: false,
187     closeOnCancel: false
188   },
189   function (isConfirm) {
190     if (isConfirm) {
191       swal("Lengkapi data!", "Kamu akan dipindahkan ke halaman untuk melengkapi data ktp.", "success");
192       location.href = "/profilpenjual";
193     } else {
194       swal("Dibatalkan", "Kamu batal melengkapi data ktp", "error");
195     }
196   });
197 }
198

```

### 3. Halaman add umkm

```

1 @extends('layouts.penjual')
2
3 <section('css')
4   <!-- multi step form -->
5   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css" integrity="sha384-TX8t2EcRE3e/ihLQ3fR2Ew5cBkQxVZ8TF27EcRYUH4Eo7h4kQXrEwv9oFQhQRh5" crossorigin="anonymous">
6   <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js" integrity="sha512-894YE6QD5I59HgZOGReFYm4dnWc1Qt5NtvYSnlvGJXtRi2P3qKss1EJQRnqeu2PFEIhCaH1+QuV2o5PnQJw==" crossorigin="anonymous">
7   <style>
8     .form-section{
9       display: none;
10    }
11
12    .form-section.current{
13      display: inline;
14    }
15    .parsley-errors-list{
16      color:red;
17    }
18
19    </style>
20    <!-- Latest compiled and minified CSS -->
21    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-select@1.13.14/dist/css/bootstrap-select.min.css">
22    <link href="{{asset('templatesAdminLTE/plugins/iCheck/custom.css')}}" rel="stylesheet">
23    <link href="{{asset('templatesAdminLTE/plugins/steps/jquery.steps.css')}}" rel="stylesheet">
24
25
26    <link href="{{ asset('templates/css/jasny/jasny-bootstrap.min.css') }}" rel="stylesheet">
27    <!-- Telephone Input CSS -->
28    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/intl-tel-input/12.1.2/css/intlTelInput.css">
29    <!-- Icons CSS -->
30    <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
31    <!-- Nice Select CSS -->
32    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/jquery-nice-select/1.1.0/css/nice-select.min.css">
33
34  @endsection
35
36  @section('content')
37  <script type="text/javascript"
38    src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDQTpXj82d8UpCi97wzo_nKXL7nYrd4G70"></script>
39
40  <div class="content-wrapper">
41    <div class="container-fluid " >

```

```

41 <div class="row justify-content-md-center">
42   <div class="col-md-9">
43     <div class="card px-1 py-1 mt-1 shadow">
44       <h2 class="text-danger text-center mt-3 mb-4">Menambahkan Data UMKM</h2>
45
46       <div class="nav nav-fill my-3">
47         <label class="nav-link shadow-sm step0 border ml-2 ">Informasi Umum</label>
48         <label class="nav-link shadow-sm step1 border ml-2 ">Alamat</label>
49         <label class="nav-link shadow-sm step2 border ml-2 ">Pembayaran</label>
50       </div>
51
52   <form action="{{ route('umkm.store') }}" method="post" class="employee-form" enctype="multipart/form-data">
53     @csrf
54     <div class="form-section">
55       <div class="row">
56
57         <div class="col-4">
58           <label for="field_name">Nama UMKM</label>
59         </div>
60         <div class="col-8">
61           <input name="nama" type="text" class="form-control" id="c_name"
62             placeholder="Input your umkm's name" required>
63           @error('nama')
64             <div class="invalid-feedback">
65               {{$message}}
66             </div>
67           @enderror
68         </div>
69       </div><br>
70       <div class="row">
71         <div class="col-4">
72           <label for="field_name">No Handphone</label>
73         </div>
74         <div class="col-8">
75           <input name="handphone" type="text" class="form-control"
76             id="c_phone_number" placeholder="Input your umkm's phone number"
77             required pattern="[0-9]+" title="Only numbers are allowed">
78           @error('handphone')
79             <div class="invalid-feedback">
80               {{$message}}
81             </div>
82           @enderror
83         </div>
84       </div><br>
85       <div class="row" style="display:none;">
86         <div class="col-4">
87           <label for="name">Status</label>
88         </div>
89         <div class="col-8">
90           <input name="status" type="text" class="form-control" id="c_status"
91             placeholder="Masukkan nama umkm anda">
92         </div>
93       </div><br>
94       <div class="row">
95         <div class="col-4">
96           <label for="name">Informasi Tambahan</label>
97         </div>
98         <div class="col-8">
99           <textarea id="c_information"
100             placeholder="Input your umkm's information" class="form-control"
101               name="informasi"></textarea>
102           @error('informasi')
103             <div class="invalid-feedback">
104               {{$message}}
105             </div>
106           @enderror
107         </div>
108       </div><br>
109       <div class="row">
110         <div class="col-4">
111           <label for="name">Upload Surat Izin Usaha *boleh HKI Merek atau SIUP</label>
112         </div>
113         <div class="col-8">
114           <div class="fileinput fileinput-new" data-provides="fileinput">
115             <span class="btn btn-outline-dark btn-file"><span
116               class="fileinput-new">Upload Izin Usaha</span>
117               <span class="fileinput-exists">Change</span><input required type="file"
118                 id="imb" name="imb"
119                 onchange="return fileValidation()" /></span>
120             <span class="fileinput-filename"></span>
121           </div>
122         </div>
123       </div>
124     </div>
125   </div>
126 </form>

```

```

120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
    <a href="#" class="close fileinput-exists" data-dismiss="fileinput"
        style="float: none" onclick="removeImage()">&times;</a>
    </div>
    <!-- Image preview -->
    <div id="imagePreview"></div>
</div>
</div>
<div class="row">
    <div class="col-4">
        <label for="name">Upload Foto Ummkm</label>
    </div>
    <div class="col-8">
        <div class="fileinput fileinput-new" data-provides="fileinput">
            <span class="btn btn-outline-dark btn-file"><span
                class="fileinput-new">Upload Foto</span>
                <span class="fileinput-exists">Change</span><input required type="file"
                id="img" name="img"
                onchange="return imgfileValidation()" /></span>
            <span class="fileinput-filename"></span>
            <a href="#" class="close fileinput-exists" data-dismiss="fileinput"
                style="float: none" onclick="removeImg()">&times;</a>
        </div>
        <!-- Image preview -->
        <div id="imgPreview"></div>
    </div>
</div>
</div>
<div class="form-section">
<div class="form-group">
    <label for="">Provinsi</label>
    <select class="form-control" name="provinsi" id="provinsi" required>
        <option value="">Provinsi</option>
        @foreach ($provinsis as $provinsi => $value)
        <option value="{{ $provinsi }}" {{ old('provinsi') == $provinsi ? 'selected' : '' }}>
            {{ $value }}
        </option>
        @endforeach
    </select>
    @error('provinsi')
        <div class="invalid-feedback">
            {{ $message }}
        </div>
    @enderror
    </div>
<div class="form-group">
    <label for="">Kota</label>
    <select class="form-control" name="kota" id="kota" required>
        <option value="">Kota</option>
        @foreach ($kotas as $kotaId => $kotaName)
        <option value="{{ $kotaId }}" {{ old('kota') == $kotaId ? 'selected' : '' }}>
            {{ $kotaName }}
        </option>
        @endforeach
    </select>
    @error('kota')
        <div class="invalid-feedback">
            {{ $message }}
        </div>
    @enderror
    </div>
    <div class="row">
        <div class="col-lg-12">
            <div class="row">
                <div class="col-4">
                    <label for="field_name">Alamat</label>
                </div>
                <div class="col-8">
                    <textarea class="form-control" placeholder="Input your umkm's address"
                        name="alamat" required></textarea>
                    @error('alamat')
                        <div class="invalid-feedback">
                            {{ $message }}
                        </div>
                    @enderror
                </div>
            </div>
        </div><br>
        <div class="row">
            <div class="col-4">

```

```

199 |         <label for="field_name">Coordinate</label>
200 |     </div>
201 |     <div class="col-4">
202 |         <input name="latitude" class="form-control" id="c_latitude"
203 |             placeholder="Latitude" required>
204 |     </div>
205 |     <div class="col-4">
206 |         <input name="longitude" class="form-control" id="c_longitude"
207 |             placeholder="Longitude" required>
208 |     </div>
209 |     </div><br>
210 |     <div class="row">
211 |         <!-- <div class="col-4">
212 |             <Button class="btn btn-outline-primary" id="get-location" type="button"
213 |                 style="margin-top:5px;">Use your location
214 |             </Button>
215 |         </div> -->
216 |         <div class="col-8">
217 |             <div class="google-map" id="map-marker" style="height:200px"></div>
218 |         </div>
219 |     </div>
220 | </div>
221 |
222 |
223 | </div>
224 | <div class="form-section">
225 |     <div class="row">
226 |         <div class="col-lg-12">
227 |             <div class="row">
228 |                 <div class="col-4">
229 |                     <label for="field_name">Pilih Metode Pembayaran</label>
230 |                 </div>
231 |                 <div class="col-8">
232 |                     <div class="selectgroup selectgroup-pills">
233 |                         @foreach($metode_pembayaran as $mp)
234 |                         <label class="selectgroup-item">
235 |                             <input type="checkbox"
236 |                                 name="mp[{$mp->id}]"
237 |                                 value="{$mp->id}">
238 |
239 |                         <span class="selectgroup-input checkbox-paymentMethod"
240 |                               onclick="addPaymentMethod()">
241 |                             <span class="selectgroup-button">{$mp->nama_bank}</span>
242 |                         </label>
243 |                         @endforeach
244 |                     </div>
245 |                 </div>
246 |             </div>
247 |             <div class="form-row" id="row-paymentMethod">
248 |                 <!-- Add Payment Method -->
249 |             </div>
250 |         </div>
251 |     </div>
252 |
253 |     </div>
254 |     <div class="form-navigation mt-3">
255 |         <button type="button" class="previous btn btn-primary float-left">&lt; PreviousNext &gt;Submit

```

```

278     $sections.removeClass('current').eq(index).addClass('current');
279
280     $('.form-navigation .previous').toggle(index>0);
281     var atTheEnd = index >= $sections.length - 1;
282     $('.form-navigation .next').toggle(!atTheEnd);
283     $('.form-navigation [Type=submit]').toggle(atTheEnd);
284
285
286     const step= document.querySelector('.step'+index);
287     step.style.backgroundColor="#17a2b8";
288     step.style.color="white";
289
290
291 }
292
293 function curIndex(){
294
295     return $sections.index($sections.filter('.current'));
296 }
297
298 $('.form-navigation .previous').click(function(){
299     | navigateTo(curIndex() - 1);
300 });
301
302 $('.form-navigation .next').click(function(){
303     | jQuery('.employee-form').parsley().whenValidate({
304         group:'block-'+curIndex()
305     }).done(function(){
306         | navigateTo(curIndex()+1);
307     });
308 });
309
310 });
311
312 $sections.each(function(index,section){
313     | $(section).find(':input').attr('data-parsley-group','block-'+index);
314 });
315
316
317 navigateTo(0);
318
319
320
321 });
322
323 $(document).ready(function () {
324     function getCities() {
325         let provinceId = $('select[name="provinsi"]').val();
326         if (provinceId) {
327             $.ajax({
328                 url: '/api/provinsi/' + provinceId + '/kota',
329                 type: "GET",
330                 dataType: "json",
331                 success: function(data) {
332                     var selectedCityId = "{{ old('kota') }}"; // Get the selected city ID
333                     var cityDropdown = $('select[name="kota"]'); // Select the city dropdown
334
335                     // Empty the dropdown options
336                     cityDropdown.empty();
337
338                     // Iterate over the cities data
339                     $.each(data.kotas, function(key, value) {
340                         // Create the option element
341                         var option = $('').val(key).text(value);
342
343                         // Check if the current city ID matches the selected city
344                         if (key == selectedCityId) {
345                             | option.attr('selected', 'selected'); // Set it as selected
346                         }
347
348                         // Append the option to the dropdown
349                         cityDropdown.append(option);
350                     });
351
352                     // Enable and show the city dropdown
353                     cityDropdown.prop('disabled', false);
354                     cityDropdown.show();
355                 },

```

```

356     |         error: function(xhr, status, error) {
357     |             console.error(error);
358     |         });
359     } else {
360         |             $('select[name="kota"]').empty();
361     }
362 }
363
364 // Event listener untuk perubahan dropdown provinsi
365 $('select[name="provinsi"]').on('change', function() {
366     |             getCities(); // Panggil fungsi getCities() saat perubahan dropdown provinsi
367 });
368
369 getCities(); // Panggil fungsi getCities() saat halaman dimuat
370
371
372 $("#wizard").steps();
373 $("#form").steps({
374     bodyTag: "fieldset",
375     onStepChanging: function (event, currentIndex, newIndex) {
376         // Always allow going backward even if the current step contains invalid fields!
377         if (currentIndex > newIndex) {
378             |             return true;
379         }
380
381         // Forbid suppressing "Warning" step if the user is too young
382         if (newIndex === 3 && Number($("#age").val()) < 18) {
383             |             return false;
384         }
385
386         var form = $(this);
387
388         // Clean up if user went backward before
389         if (currentIndex < newIndex) {
390             // To remove error styles
391             $(".body:eq(" + newIndex + ") label.error", form).remove();
392             $(".body:eq(" + newIndex + ") .error", form).removeClass("error");
393
394             // Disable validation on fields that are disabled or hidden.
395             form.validate().settings.ignore = ":disabled,:hidden";
396
397             // Start validation; Prevent going forward if false
398             return form.valid();
399         },
400         onStepChanged: function (event, currentIndex, priorIndex) {
401             // Suppress (skip) "Warning" step if the user is old enough.
402             if (currentIndex === 2 && Number($("#age").val()) >= 18) {
403                 |                 $(this).steps("next");
404             }
405
406             // Suppress (skip) "Warning" step if the user is old enough and wants to the previous step.
407             if (currentIndex === 2 && priorIndex === 3) {
408                 |                 $(this).steps("previous");
409             }
410         },
411         onFinishing: function (event, currentIndex) {
412             var form = $(this);
413
414             // Disable validation on fields that are disabled.
415             // At this point, it's recommended to do an overall check (meaning ignoring only disabled fields)
416             form.validate().settings.ignore = ":disabled";
417
418             // Start validation; Prevent form submission if false
419             return form.valid();
420         },
421         onFinished: function (event, currentIndex) {
422             var form = $(this);
423
424             // Submit form input
425             form.submit();
426         }
427     }).validate({
428         errorPlacement: function (error, element) {
429             |                 element.before(error);
430         },
431         rules: {
432

```

```

-->    });
433     |     confirm: {
434     |       equalTo: "#password"
435     |     }
436   });
437 });
438 });
439 });
440 });
441 });
442 function fileValidation() {
443   var fileInput = document.getElementById('imb');
444   var filePath = fileInput.value;
445   var allowedExtensions = /(.(jpg|jpeg|png|pdf)$)/i;
446   if (!allowedExtensions.exec(filePath)) {
447     alert('Please upload file having extensions .jpeg/.jpg/.png/.pdf only.');
448     fileInput.value = '';
449     return false;
450   } else {
451     //Image preview
452     if (fileInput.files && fileInput.files[0]) {
453       var reader = new FileReader();
454       reader.onload = function (e) {
455         document.getElementById('imagePreview').innerHTML = '<iframe id="image" src="' + e.target.result +
456         '" style="width: 100%; height: 100%;"></iframe>';
457       };
458       reader.readAsDataURL(fileInput.files[0]);
459     }
460   }
461 }
462 function imgfileValidation() {
463   var fileInput = document.getElementById('img');
464   var filePath = fileInput.value;
465   var allowedExtensions = /(.(jpg|jpeg|png|pdf)$)/i;
466   if (!allowedExtensions.exec(filePath)) {
467     alert('Please upload file having extensions .jpeg/.jpg/.png/.pdf only.');
468     fileInput.value = '';
469     return false;
470   } else {
471     //Image preview
472     if (fileInput.files && fileInput.files[0]) {
473       var reader = new FileReader();
474       reader.onload = function (e) {
475         document.getElementById('imgPreview').innerHTML = '<iframe id="image" src="' + e.target.result +
476         '" style="width: 100%; height: 100%;"></iframe>';
477       };
478       reader.readAsDataURL(fileInput.files[0]);
479     }
480   }
481 }
482 function removeImage() {
483   var image = document.getElementById('image');
484   var preview = document.getElementById('imagePreview');
485   preview.removeChild(image);
486 }
487 function removeImg() {
488   var preview = document.getElementById('imgPreview');
489   while (preview.firstChild) {
490     preview.removeChild(preview.firstChild);
491   }
492 }
493 }
494
495 function addPaymentMethod() {
496   let checked = [];
497   for (var i = 0; i < $('.checkbox-paymentMethod').length; i++) {
498     if ($('.checkbox-paymentMethod').eq(i).prop('checked') == true) {
499       checked.push($('.checkbox-paymentMethod').eq(i).val());
500     }
501   }
502   console.log(checked);
503   $.post(
504     "{{url('api/payment/add-paymentMethod'))}}",
505     {
506       "_token": "{{ csrf_token() }}",
507       id: checked
508     },
509     function (result) {
510       console.log(result.length)
511       $('.method').remove();
512       for (var i = 0; i < result.length; i++) {
513

```

```

512     var div = $('<div class="form-row method">
513         <div class="form-group col-md-4">
514             <label>' + result[i].nama_bank + '</label>
515         </div>
516         <div class="form-group col-md-8">
517             <input name="norek[]" type="text"
518                 placeholder="Nomor rekening"
519                 class="form-control" required>
520         </div>
521         <div class="form-group col-md-8">
522             <input name="metode_pembayaran[]" type="hidden"
523                 value="" + result[i].id + ""
524                 class="form-control">
525         </div>
526     </div>');
527
528     $('#row-paymentMethod').append(div);
529 }
530
531 }
532
533 }
534
535
536
537 function check() {
538     $("#row-dp").toggle(this.checked);
539 }
540
541 function cloneImage() {
542     $("#clone-1")
543         .eq(0)
544         .clone()
545         .find("input").val("").end() // ***
546         .show()
547         .insertAfter(".clone-1:last");
548 }
549
550 function delClone(data) {
551
552     console.log(data.closest('.clone-1'));
553     if ($('.clone-1').length > 1) {
554         data.closest('.clone-1').remove()
555     } else {
556         swal({
557             title: "Warning!",
558             text: "Select at least one image",
559             type: "warning",
560             confirmButtonColor: "#D0E855",
561             confirmButtonText: "OK",
562             closeOnConfirm: false,
563             closeOnCancel: false
564         });
565     }
566
567     // Proses membuat marker
568     let mapCreate;
569     let markers = [];
570     // When the window has finished loading google map
571     google.maps.event.addDomListener(window, 'load', init);
572
573     function init() {
574         // More info see: https://developers.google.com/maps/documentation/javascript/reference#MapOptions
575         var mapOptions1 = {
576             zoom: 13,
577             center: new google.maps.LatLng(-0.23976776742668077, 100.60986119473249),
578             // Style for Google Maps
579             styles: [
580                 {
581                     "featureType": "water",
582                     "stylers": [
583                         {
584                             "saturation": 43
585                         },
586                         {
587                             "lightness": -11
588                         },
589                         {
590                             "hue": "#0088ff"
591                         }
592                     ],
593                     "elementType": "geometry.fill",
594                     "stylers": [
595                         {
596                             "color": "#0088ff"
597                         }
598                     ]
599                 }
600             ]
601         }
602         mapCreate = new google.maps.Map(document.getElementById('map'), mapOptions1);
603
604         // Add markers
605         markers.push(new google.maps.Marker({
606             position: mapCreate.getCenter(),
607             map: mapCreate
608         }));
609     }
610 }

```

```

591     "stylers": [
592         "hue": "#ff0000"
593     ], {
594         "saturation": -100
595     }, {
596         "lightness": 99
597     }
598 }, {
599     "featureType": "road",
600     "elementType": "geometry.stroke",
601     "stylers": [
602         "color": "#808080"
603     ], {
604         "lightness": 54
605     }
606 }, {
607     "featureType": "landscape.man_made",
608     "elementType": "geometry.fill",
609     "stylers": [
610         "color": "#ece2d9"
611     ]
612 }, {
613     "featureType": "poi.park",
614     "elementType": "geometry.fill",
615     "stylers": [
616         "color": "#ccdcac"
617     ]
618 }, {
619     "featureType": "road",
620     "elementType": "labels.text.fill",
621     "stylers": [
622         "color": "#767676"
623     ]
624 }, {
625     "featureType": "road",
626     "elementType": "labels.text.stroke",
627     "stylers": [
628         "color": "#ffffff"
629     ]
630 },
631     "featureType": "poi",
632     "stylers": [
633         "visibility": "off"
634     ]
635 }, {
636     "featureType": "landscape.natural",
637     "elementType": "geometry.fill",
638     "stylers": [
639         "visibility": "on"
640     ], {
641         "color": "#b8cb93"
642     }
643 }, {
644     "featureType": "poi.park",
645     "stylers": [
646         "visibility": "on"
647     ]
648 }, {
649     "featureType": "poi.sports_complex",
650     "stylers": [
651         "visibility": "on"
652     ]
653 }, {
654     "featureType": "poi.medical",
655     "stylers": [
656         "visibility": "on"
657     ]
658 }, {
659     "featureType": "poi.business",
660     "stylers": [
661         "visibility": "simplified"
662     ]
663     }
664 };
665 var mapElement2 = document.getElementById('map-marker');
666 var mapCreate = new google.maps.Map(mapElement2, mapOptions1);

```

```

670     google.maps.event.addListener(mapCreate, 'click', function (e) {
671       if (markers.length > 0) {
672         for (let i = 0; i < markers.length; i++) {
673           markers[i].setMap(null);
674         }
675       }
676
677       $('#c_latitude').val(e.latLng.lat);
678       $('#c_longitude').val(e.latLng.lng);
679
680       var marker = new google.maps.Marker({
681         position: e.latLng,
682         title: "Hello world!"
683       });
684       markers.push(marker);
685       marker.setMap(mapCreate);
686     });
687
688 // Variabel untuk menyimpan batas kordinat
689 bounds = new google.maps.LatLngBounds();
690
691
692 document
693   .getElementById("get-location")
694   .addEventListener("click", function () {
695     getLocation(mapCreate);
696   });
697 }
698
699 function getLocation(mapCreate) {
700   // Geolokasi / lokasi sendiri
701   console.log('sampai sini');
702   var infowindow = new google.maps.InfoWindow({
703     map: mapCreate
704   });
705
706   if (navigator.geolocation) {
707     navigator.geolocation.getCurrentPosition(function (position) {
708       var pos = {
709
710         lat: position.coords.latitude,
711         lng: position.coords.longitude
712       };
713
714       $('#c_latitude').val(position.coords.latitude);
715       $('#c_longitude').val(position.coords.longitude);
716
717       if (marker.length > 0) {
718         for (let i = 0; i < markers.length; i++) {
719           markers[i].setMap(null);
720         }
721
722         var marker = new google.maps.Marker({
723           position: pos,
724           map: mapCreate,
725           title: 'Lokasi Anda',
726           icon: 'http://maps.google.com/mapfiles/ms/icons/green-dot.png',
727           draggable: true,
728           animation: google.maps.Animation.DROP
729         });
730         markers.push(marker);
731
732         marker.addListener('click', toggleBounce);
733
734         function toggleBounce() {
735           if (marker.getAnimation() != null) {
736             marker.setAnimation(null);
737           } else {
738             marker.setAnimation(google.maps.Animation.BOUNCE);
739           }
740         }
741
742         infowindow.setPosition(pos);
743         infowindow.setContent('Anda Disini');
744         map.setCenter(pos);
745       }, function () {
746         handleLocationError(true, infowindow, map.getCenter());
747       });
748     } else {
749       // Jika browser tidak mendukung geolokasi pindah ke lokasi ketetapan diatas (center)
750       handleLocationError(false, infowindow, map.getCenter());
751     }
752   }
753 }
754
755 function handleLocationError(browserHasGeolocation, infowindow, pos) {
756   infowindow.setPosition(pos);
757   infowindow.setContent('Error: Kesalahan layanan geolokasi.' +
758   'Error: <i>Browser</i> anda tidak mendukung geolokasi.');
759 }
760
761 </script>
762
763 @endsection

```

#### 4. Halaman edit umkm

```
37 @section('content')
38 <script type="text/javascript"
39   src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDQTpxj82d8UpCi97wzo_nKXL7nYrd4G70"></script>
40
41 <div class="content-wrapper">
42 <div class="container-fluid ">
43   <div class="row justify-content-md-center">
44     <div class="col-md-9 ">
45       <div class="card px-1 py-1 mt-1 shadow">
46
47         <h2 class="text-danger text-center mt-3 mb-4">Mengubah Data UMKM</h2>
48         @if ($umkms->status == 2)
49           <div class="row">
50             <div class="col-lg-12">
51               <h5 style="color:red">
52                 <center>
53                   <b>Umkm anda ditolak !</b><br>
54                   <b>({$umkms->reject_note})</b>
55                 </center>
56               </h5>
57             </div>
58           </div>
59         @endif
60
61         <div class="nav nav-fill my-3">
62           <label class="nav-link shadow-sm step0 border ml-2 ">Informasi Umum</label>
63           <label class="nav-link shadow-sm step1 border ml-2 ">Alamat</label>
64           <label class="nav-link shadow-sm step2 border ml-2 ">Pembayaran</label>
65         </div>
66         <form action="{{ route('umkm.update', $umkms->id) }}" method="post" class="employee-form" enctype="multipart/form-data">
67           @method('PUT')
68           @csrf
69             <div class="form-section">
70               <div class="row">
71                 <div class="col-4">
72                   <label for="field_name">Nama UMKM</label>
73                 </div>
74                 <div class="col-8">
75                   <input name="nama" type="text" class="form-control" id="c_name"
76
76                     value="{{ $umkms->nama }}" placeholder="Input your umkm's name" required>
77                     @error('nama')
78                       <div class="invalid-feedback">
79                         {{ $message }}
80                       </div>
81                     @enderror
82                   </div>
83                   </div><br>
84                   <div class="row">
85                     <div class="col-4">
86                       <label for="field_name">No Handphone</label>
87                     </div>
88                     <div class="col-8">
89                       <input name="handphone" type="phone" class="form-control"
90                         id="c_phone_number" value="{{ $umkms->handphone }}" required pattern="[0-9]+" title="Only numbers required">
91                         @error('handphone')
92                           <div class="invalid-feedback">
93                             {{ $message }}
94                           </div>
95                         @enderror
96                       </div>
97                     </div><br>
98                     <div class="row">
99                       <div class="col-4">
100                         <label for="name">Informasi Tambahan</label>
101                       </div>
102                       <div class="col-8">
103                         <textarea id="c_information" placeholder="Input your umkm's information" class="form-control"
104                           name="informasi">{{ $umkms->informasi }}
```

```
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
<div class="row">
    <div class="col-4">
        <label for="name">Upload Surat Izin Usaha *boleh HKI Merek atau SIUP</label>
    </div>
    <div class="col-8">
        <div @if ($umkms->imb)
            class="fileinput fileinput-exists"
        @else
            class="fileinput fileinput-new"
        @endif data-provides="fileinput">
            <span class="btn btn-outline-dark btn-file"><span
                class="fileinput-new">Upload Izin Usaha</span>
                <span class="fileinput-exists">Change</span><input type="file"
                    id="imb" name="imb" onchange="return fileValidation()">
            @if($umkms->imb)
                value="{{($umkms->imb)}}" file="{{($umkms->imb)}}"
            @endif</span>
            <span class="fileinput-filename">
                {{($umkms->imb)}}
            </span>
            <a href="#" class="close fileinput-exists" data-dismiss="fileinput"
                style="float: none" onclick="removeImage()">X</a>
        </div>
        <!-- Image preview -->
        <div id="imagePreview">
            @if ($umkms->imb)
                
            @endif
        </div>
    </div>
</div>
<div class="row">
    <div class="col-4">
        <label for="name">Upload Foto Umkm</label>
    </div>
    <div class="col-8">
        <div @if ($umkms->img)
            class="fileinput fileinput-exists"
        @else
            class="fileinput fileinput-new"
        @endif data-provides="fileinput">
            <span class="btn btn-outline-dark btn-file"><span
                class="fileinput-new">Upload Foto Umkm</span>
                <span class="fileinput-exists">Change</span><input type="file"
                    id="img" name="img" onchange="return imgfileValidation()">
            @if($umkms->img)
                value="{{($umkms->img)}}" file="{{($umkms->img)}}"
            @endif</span>
            <span class="fileinput-filename">
                {{($umkms->img)}}
            </span>
            <a href="#" class="close fileinput-exists" data-dismiss="fileinput"
                style="float: none" onclick="removeImage()">X</a>
        </div>
        <!-- Image preview -->
        <div id="imgPreview">
            @if ($umkms->img)
                
            @endif
        </div>
    </div>
</div>
<div class="form-section">
<div class="form-group">
    <label for="">Provinsi</label>
    <select class="form-control" name="provinsi" id="provinsi">
        <option value="">Provinsi</option>
        @foreach ($provinsis as $provinsi => $value)
            <option value="{{ $provinsi }}" {{ $umkms->province_id == $provinsi ? 'selected' : '' }}>{{ $value }}</option>
        @endforeach
    </select>
    <div class="text-danger"></div>
</div>
<div class="form-group">
    <label for="">Kota</label>
```

```

192
193     <select class="form-control" name="kota" id="kota">
194         <option value="">Kota</option>
195         @foreach ($kotas as $kota => $value)
196             <option value="{{ $kota }}" {{ $umkms->city_id == $kota ? 'selected' : '' }}>{{ $value }}</option>
197         @endforeach
198     </select>
199     <div class="text-danger"></div>
200     </div>
201     <div class="row">
202         <div class="col-lg-12">
203             <div class="row">
204                 <div class="col-4">
205                     <label for="field_name">Alamat</label>
206                 </div>
207                 <div class="col-8">
208                     <textarea class="form-control" value="{{($umkms->alamat)}}" placeholder="Input your umkm's address"
209                         name="alamat" required={{($umkms->alamat)}}></textarea>
210                     @error('alamat')
211                         <div class="invalid-feedback">
212                             {{($message)}}
213                         </div>
214                     @enderror
215                 </div>
216             </div>
217             <br>
218             <div class="row">
219                 <div class="col-4">
220                     <label for="field_name">Coordinate</label>
221                 </div>
222                 <div class="col-4">
223                     <input name="latitude" class="form-control" id="c_latitude"
224                         value="{{($umkms->latitude)}}" placeholder="Latitude" required>
225                 </div>
226                 <div class="col-4">
227                     <input name="longitude" class="form-control" id="c_longitude"
228                         value="{{($umkms->longitude)}}" placeholder="Longitude" required>
229                 </div>
230             </div>
231             <div class="row">
232                 <div class="col-4">
233                     <button type="button" style="margin-top:5px;">Use your location
234                     </button> -->
235                 </div>
236                 <div class="col-8">
237                     <div class="google-map" id="map-marker" style="height:200px"></div>
238                 </div>
239             </div>
240         </div>
241     </div>
242     <div class="form-section">
243         <div class="row">
244             <div class="col-lg-12">
245                 <div class="row">
246                     <div class="col-4">
247                         <label for="field_name">Pilih Metode Pembayaran</label>
248                     </div>
249                     <div class="col-8">
250                         <div class="selectgroup selectgroup-pills">
251                             @foreach($metode_pembayaran as $mp)
252                                 <label class="selectgroup-item">
253                                     <input type="checkbox"
254                                         name="mp[{{($mp->id)}}]"
255                                         value="{{($mp->id)}}"
256                                         class="selectgroup-input checkbox-paymentMethod"
257                                         onclick="addPaymentMethod()" @if ($mp->isChecked($umkms->id))
258                                             checked
259                                         @endif>
260                                     <span class="selectgroup-button">{{($mp->nama_bank)}}</span>
261                                 </label>
262                             @endforeach
263                         </div>
264                     </div>
265                 </div>
266                 <div class="form-row" id="row-paymentMethod">
267                     <!-- Add Payment Method -->
268                     @if($umkms->detail_metode_pembayaran)
269                         @foreach($umkms->detail_metode_pembayaran as $dmp)

```

```

271         <div class="form-group row method">
272             <label class="col-4 col-form-label">{{ $dmp->metode_pembayaran->nama_bank }}</label>
273             <div class="col-7">
274                 <input name="norek[]" type="text" placeholder="Nomor rekening"
275                     value="{{ $dmp->norek }}" class="form-control">
276             </div>
277             <div class="col-1">
278                 <input name="metode_pembayaran[]" type="hidden"
279                     value="{{ $dmp->metodepemb_id }}"
280                     class="form-control">
281             </div>
282         </div>
283     @endforeach
284     @endif
285     </div>
286     </div>
287     </div>
288     </div>
289     </div>
290     <div class="form-navigation mt-3">
291         <button type="button" class="previous btn btn-primary float-left">&lt; Previous</button>
292         <button type="button" class="next btn btn-primary float-right">Next &gt;</button>
293         <button type="submit" class="btn btn-success float-right">Submit</button>
294     </div>
295     <input hidden value="{{ $data }}" name="data">
296   </form>
297 </div>
298 </div>
299 </div>
300 </div>
301 </div>
302 </div>
303 @endsection

```

## 5. Halaman *show umkm*

```

1  @extends('layouts.penjual')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQhg/wzxH7pPCaAqCgJzJLWZGvXwDfPjyOOGVQ" crossorigin="anonymous">
5  <link rel="stylesheet" href="{{ asset('templates/css/components.css') }}>
6  <link href="{{ asset('templates/js/plugins/dataTables/datatables.bundle.css') }}" rel="stylesheet" type="text/css" />
7
8  <link href="https://cdn.datatables.net/1.11.4/css/jquery.dataTables.min.css" rel="stylesheet">
9  @endsection
10
11 @section('content')
12 <script type="text/javascript"
13     src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDQTpxj82d8UpC197wzo_nKXL7nyrd4670"></script>
14 <div class="content-wrapper">
15 <section class="content">
16     <div class="container-fluid">
17         <div class="row">
18             <div class="col-12">
19                 <div class="card">
20                     <div class="card-header">
21                         <h2 class="card-title">{{ $umkms->nama }}</h2>
22                         <a style="float:right;" href="#" onclick="deleteUmkm('{{ $umkms->id }}')">
23                             <button class="btn btn-danger">Hapus UMKM</button>
24                         </a>
25                         <a style="float:right; margin-right:5px;" href="{{ route('umkm.edit', $umkms->id) }}>
26                             <button class="btn btn-primary">Edit UMKM</button>
27                         </a>
28                         <a style="float:right; margin-right:5px;" href="{{ route('umkm.product.create', $umkms->id) }}>
29                             <button class="btn btn-primary">Tambah Produk</button>
30                         </a>
31                         <button style="float:right; margin-right:5px;" type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#lihatSuratIzin">
32                             Lihat Surat Izin
33                         </button>
34                     </div>
35                     <!-- /.card-header -->
36                     <!-- Modal -->
37                     <div class="modal fade" id="imb" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
38                         <div class="modal-dialog">
39                             <div class="modal-content">

```

```

41 |         <div class="modal-header">
42 |             <h1 class="modal-title fs-5" id="exampleModalLabel">Surat IMB</h1>
43 |             <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
44 |         </div>
45 |         <div class="modal-body">
46 |             <figure>
47 |                 
48 |             </figure>
49 |         </div>
50 |         <div class="modal-footer">
51 |             <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
52 |         </div>
53 |     </div>
54 | 
```

---

```

55 |     <div class="card-body">
56 |         <div class="row">
57 |             <div class="col-md-12">
58 |                 <a href="#" class="btn btn-primary btn-icon icon-left btn-lg btn-block mb-4 d-md-none" data-toggle="slide" #ticket-items>
59 |                     <i class="fas fa-list"></i> All Tickets
60 |                 </a>
61 |                 <div class="tickets">
62 |                     <div class="ticket-items" id="ticket-items">
63 |                         <div class="gallery gallery-fw" data-item-height="150">
64 |                             <div class="gallery-item" data-image="{{ asset('img_umkm.'/'.$Sumkms->img) }}"
65 |                                 data-title="Profil Umkm"
66 |                             </div>
67 |                         </div>
68 |                         <div class="google-map" id="map-show" style="height:200px"></div>
69 |                     </div>
70 |                 </div>
71 |                 <div class="ticket-content">
72 |                     <div class="ticket-header">
73 |                         <div class="ticket-sender-picture img-shadow">
74 |                             
75 |                         </div>
76 |                         <div class="ticket-detail">
77 |                             <div class="ticket-title">
78 | 
```

---

```

79 | 
80 |             <h4>{{$Sumkms->nama}}</h4>
81 |             </div>
82 |             <div class="ticket-info">
83 |                 <div class="font-weight-600">{{$Sumkms->penjual->firstname}}
84 |                     {{$Sumkms->penjual->lastname}}</div>
85 |                 <div class="bullet"></div>
86 |                 <div class="text-primary font-weight-600">
87 |                     {{$Sumkms->penjual->handphone}}
88 |                 </div>
89 |             </div>
90 |         </div>
91 |         <div class="ticket-description">
92 |             <h6>Informasi Umkm</h6>
93 |             <p>{{ $Sumkms->informasi }}</p>
94 |             <div class="row">
95 |                 <div class="col-6">
96 |                     <h6>Alamat</h6>
97 |                     <p>{{ $Sumkms->alamat }}</p>
98 |                 </div>
99 |             </div>
100 | 
101 |             <div class="row">
102 |                 <div class="col-6">
103 |                     <h6>Metode Pembayaran</h6>
104 |                     <p>
105 |                         @foreach($Sumkms->Detail_Metode_Pembayaran as $payment)
106 |                             <b> {{ $payment->Metode_Pembayaran->nama_bank}} ( {{ $payment->norek}} )</b>
107 |                             <br>
108 |                         @endforeach
109 |                     </p>
110 |                 </div>
111 |             </div>
112 | 
113 |             <div class="row">
114 |                 <div class="col-lg-12">
115 |                     <table class="table table-striped dataTables-produk" id="table-2">
116 |                         <thead>
117 |                             <tr>
118 |                                 <th>Nama Produk</th>
119 | 
```

```
120 <th>Kategor</th>
121 <th>Action</th>
122 </tr>
123 </thead>
124 </table>
125 </div>
126 <div class="col-lg-6">
127 </div>
128 </div>
129 </div>
130 </div>
131 </div>
132 </div>
133 <div class="ticket-divider"></div>
134 <div class="ticket-form">
135 </div>
136 </div>
137 </div>
138 </div>
139 </div>
140 </div>
141 </div>
142 </div>
143 </div>
144 </div>
145 <!-- /.card-body -->
146 </div>
147 <!-- /.card -->
148 </div>
149 <!-- /.col -->
150 </div>
151 </div>
152 <!-- /.row -->
153 </div>
154 <!-- /.container-fluid -->
155 </section>
156 </div>
157 </div>
158 {!! Form::open(['method'=>'PATCH', 'route' => ['umkm.edit', 0], 'style' =>'display:none','id'=>'edit_umkm']) !!}
159 {!! Form::close() !!}
160
161 {!! Form::open(['method'=>'DELETE','route'=>['produk.destroy',0],
162 'style'=>'display.none','id'=>'deleted_product']) !!}
163 {!! Form::close() !!}
164
165 {!! Form::open(['method'=>'DELETE','route'=>['umkm.destroy',0],
166 'style'=>'display.none','id'=>'deleted_umkm']) !!}
167 {!! Form::close() !!}
168
169 @endsection
170 @section('script')
171 <!-- General JS Scripts -->
172 <!-- Masukkan file Jquery -->
173 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
174
175 <!-- Masukkan file Bootstrap JavaScript -->
176 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7j"
177 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
178 <script src="https://code.jquery.com/jquery-3.3.1.min.js"
179 | integrity="sha256-fkpCb/KJQ1LNU0u9Ita32o/WfZxtwRo80tmkRdAu8=" crossorigin="anonymous"></script>
180 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
181 | integrity="sha384-U02e70CphQdSJQ6nJty5KvhPtPhzVj9W0c1lHTNGa3JDZwrNq4sF86gIHNDz0WI" crossorigin="anonymous">
182 </script>
183 <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
184 | integrity="sha384-J35mVgyd0p3pxBlrRibZUAYoIIy60rQ6rJiEaFf/nJGzIxDsF4x0xIM+B07jRM" crossorigin="anonymous">
185 </script>
186 <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery.nicescroll/3.7.6/jquery.nicescroll.min.js"></script>
187 <script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.24.0/moment.min.js"></script>
188 <!-- Data tables -->
189 <script src="{{asset('templates/js/plugins/dataTables.datatables.bundle.js')}}" type="text/javascript"></script>
190
191 <script src="https://cdn.datatables.net/1.11.4/js/jquery.dataTables.min.js"></script>
192
193 <script src="{{ asset('templates/js/scripts.js') }}"></script>
194 <script type="text/javascript">
```

```

196  $(document).ready(function () {
197      $('.dataTables-produk').DataTable({
198
199          responsive: true,
200          searching: false,
201          lengthChange: false,
202          searchDelay: 500,
203          processing: true,
204          // serverSide: true,
205          ajax: {
206              url: "{{url('api/produk')}}?data=all&&id={{$umkm->id}}",
207              dataSrc: ''
208          },
209          columns: [
210              { data: 'nama' },
211              { data: 'kategori' },
212              {
213                  data: 'id',
214                  responsivePriority: -1
215              }
216          ],
217          columnDefs: [
218              {
219                  targets: -1,
220                  title: "Action",
221                  orderable: false,
222                  render: function (data, type, full, meta) {
223                      var base = "{{url('/')}}";
224                      return ` +
225                          `<a class="btn btn-dark" href="/umkm/${data}/` +
226                          `/product-show">i class="fas fa-file-alt"></i> Detail</a>` +
227                          `<a class="btn btn-dark" href="/produk/${data}/` +
228                          `/edit" data-produk-id="${data}">i class="fas fa-edit"></i>Edit</a>` +
229                          `<a class="btn btn-danger" onclick="confirmdelete(` +
230                          `data + ` +
231                          `)` + href="javascript:void(0)">i class="fas fa-trash"></i>Hapus</a>` +
232                          ``;
233                  }
234              },
235          ],
236      });
237  });
238  });
239  });
240  });
241  });
242  });
243  });
244  });
245  });
246  });
247  });
248  });
249  });
250  });
251  });
252  });
253  });
254  });
255  });
256  });
257  });
258  });
259  });
260  });
261  });
262  });
263  });
264  });
265  });
266  });
267  });
268  });
269  });
270  });
271  });
272  });
273  });
274  });

```

```

275     showCancelButton: true,
276     confirmButtonColor: "#ff2929",
277     confirmButtonText: "Ya, Hapus Data!",
278     cancelButtonText: "Tidak, batalkan!",
279     closeOnConfirm: false,
280     closeOnCancel: false
281   },
282   function (isConfirm) {
283     if (isConfirm) {
284       swal("Data Terhapus!", "Data tersebut telah terhapus", "success");
285       $('#deleted_umkm').attr('action', "{{route('umkm.index')}}/" + id);
286       $('#deleted_umkm').submit();
287     } else {
288       swal("Batal", "Anda batal menghapus data", "error");
289     }
290   });
291 });
292
293
294
295
296 let mapCreate;
297 let mapShow;
298 let markers = [];
299 // When the window has finished loading google map
300 google.maps.event.addDomListener(window, 'load', init);
301
302 function init() {
303   // More info see: https://developers.google.com/maps/documentation/javascript/reference#MapOptions
304   var mapOptions1 = {
305     zoom: 12,
306     center: new google.maps.LatLng(-0.9111111111111111, 100.34972222222221),
307     // Style for Google Maps
308     styles: [
309       {
310         "featureType": "water",
311         "stylers": [
312           {
313             "saturation": 43
314           },
315           {
316             "hue": "#0088ff"
317           }
318         ],
319         "featureType": "road",
320         "elementType": "geometry.fill",
321         "stylers": [
322           {
323             "hue": "#ff0000"
324           },
325           {
326             "saturation": -100
327           },
328           {
329             "lightness": 99
330           }
331         ],
332         "featureType": "road",
333         "elementType": "geometry.stroke",
334         "stylers": [
335           {
336             "color": "#808080"
337           },
338           {
339             "lightness": 54
340           }
341         ],
342         "featureType": "landscape.man_made",
343         "elementType": "geometry.fill",
344         "stylers": [
345           {
346             "color": "#ece2d9"
347           }
348         ],
349         "featureType": "poi.park",
350         "elementType": "geometry.fill",
351         "stylers": [
352           {
353             "color": "#ccdcac"
354           }
355         ],
356         "featureType": "road",
357         "elementType": "labels.text.fill",
358         "stylers": [
359           {
360             "color": "#767676"
361           }
362         ]
363       }
364     ]
365   }
366 }

```

```

354     "featureType": "road",
355     "elementType": "labels.text.stroke",
356     "stylers": [
357       {
358         "color": "#ffffff"
359       }
360     ],
361     "featureType": "poi",
362     "stylers": [
363       {
364         "visibility": "off"
365       }
366     ],
367     "featureType": "landscape.natural",
368     "elementType": "geometry.fill",
369     "stylers": [
370       {
371         "visibility": "on"
372       },
373       {
374         "color": "#b8cb93"
375       }
376     ],
377     "featureType": "poi.park",
378     "stylers": [
379       {
380         "visibility": "on"
381       }
382     ],
383     "featureType": "poi.sports_complex",
384     "stylers": [
385       {
386         "visibility": "on"
387       }
388     ],
389     "featureType": "poi.business",
390     "stylers": [
391       {
392         "visibility": "simplified"
393       }
394     ]
395   };
396
397   // Get all html elements for map
398   var mapElement3 = document.getElementById('map-show');
399
400   // Create the Google Map using elements
401   var map = new google.maps.Map(mapElement3, mapOptions1);
402
403   // Variabel untuk menyimpan batas kordinat
404   bounds = new google.maps.LatLngBounds();
405
406
407   $.ajax({
408     url: "{{url('api/umkm/get-location')}}?id={{$umkms->id}}",
409     dataType: 'json',
410     cache: false,
411     dataSrc: '',
412
413     success: function (data) {
414       var latitude = data.map(function (item) {
415         return item.latitude;
416       });
417       var longitude = data.map(function (item) {
418         return item.longitude;
419       });
420       var latlng = new google.maps.LatLng(parseFloat(latitude), parseFloat(longitude));
421       console.log(latitude);
422       console.log(longitude);
423       for (i = 0; i < data.length; i++) {
424         var pos = {
425           lat: parseFloat(latitude[i]),
426           lng: parseFloat(longitude[i])
427         };
428         var marker = new google.maps.Marker({
429           position: pos,
430           map: map,
431           title: 'Lokasi Anda',
432           icon: 'http://maps.google.com/mapfiles/ms/icons/green-dot.png',
433             draggable: true,
434             animation: google.maps.Animation.DROP
435           });
436           marker.setMap(map);
437           map.setCenter(latlng);
438         }
439       }
440     );
441   });
442
443 </script>
444 @endsection
445

```

## 6. Halaman *create* produk

```
1  @extends('layouts.penjual')
2
3  @section('css')
4  <!-- multi step form -->
5  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css" integrity="sha384-TX8t27EcRE3e/lnU7zQxVnCDA5uIKz4rEkeIxeMed4WjlIDPvg6uq
6  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.6.0/jquery.min.js" integrity="sha512-894Y6QW6QD5159HgZOGReFym4dn/c1Qt5NtvYSAicOP+u1T9qYvdinZ0PPSiqn+/3e7J04EaG7Tub
7
8
9  <style>
10 .form-section{
11     display: none;
12 }
13
14 .form-section.current{
15     display: inline;
16 }
17 .parsley-errors-list{
18     color:red;
19 }
20 </style>
21 <!-- Latest compiled and minified CSS -->
22 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-select@1.13.14/dist/css/bootstrap-select.min.css"
23 <link href="{{asset('templatesAdminLTE/plugins/iCheck/custom.css')}}" rel="stylesheet">
24 <!-- <link href="{{asset('templatesAdminLTE/plugins/steps/jquery.steps.css')}}" rel="stylesheet"> -->
25
26
27 <link href="{{ asset('templates/css/jasny-bootstrap.min.css') }}" rel="stylesheet">
28 <!-- Telephone Input CSS -->
29 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/intl-tel-input/12.1.2/css/intlTelInput.css">
30 <!-- Icons CSS -->
31 <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">
32 <!-- Nice Select CSS -->
33 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/jquery-nice-select/1.1.0/css/nice-select.min.css">
34
35
36 @endsection
37
38 @section('content')
39     <div class="content-wrapper">
40         <div class="container-fluid">
41
42             <div class="row justify-content-md-center">
43                 <div class="col-md-9 ">
44                     <div class="card px-1 py-1 mt-1 shadow">
45                         <h2 class="text-danger text-center mt-3 mb-4">Menambahkan Data Produk</h2>
46
47                         <div class="nav nav-fill my-3">
48                             <label class="nav-link shadow-sm step0 border ml-2">Informasi Umum Produk</label>
49                             <label class="nav-link shadow-sm step1 border ml-2">Rincian Produk</label>
50                             <label class="nav-link shadow-sm step2 border ml-2">Gambar Produk</label>
51                         </div>
52                         <form action="{{ route('umkm.product.store', $umkm->id) }}" method="post" class="employee-form" enctype="multipart/form-data">
53                             @csrf
54                             <div class="form-section">
55                                 <div class="row">
56                                     <div class="col-4">
57                                         <label for="field_name">Nama Produk</label>
58                                         <div class="col-8">
59                                             <input name="nama" type="text" class="form-control" id="c_name" placeholder="Input your produk name" required>
60                                             @error('nama')
61                                                 <div class="invalid-feedback">
62                                                     {{ $message }}
63                                                 </div>
64                                             @enderror
65                                         </div>
66                                     </div>
67                                     <br>
68                                     <div class="row">
69                                         <div class="col-4">
70                                             <label for="kategori">Kategori Produk</label>
71                                         </div>
72                                         <div class="col-8">
73                                             <select name="kategori" id="s_kategori_create" class="form-control" data-live-search="true" onchange="this.className = ''">
74                                                 @foreach($kategori as $kt)
75                                                     <option value="{{ $kt->id }}>{{ $kt->nama_kategori }}</option>
76                                                 @endforeach
77                                             </select>
78                                             @error('kategori')
79                                                 <div class="invalid-feedback">
```



```

80           |   {{ $message }}  

81           |   </div>  

82           |   @enderror  

83           |   </div>  

84           |   </div> <br>  

85           |   <div class="row">  

86           |       <div class="col-4">  

87           |           <label for="field_name">No Izin P-IRT</label>  

88           |       </div>  

89           |       <div class="col-8">  

90           |           <input name="no_p_irt" type="text" class="form-control"  

91           |               id="c_no_p_irt" placeholder="Input your p_irt"  

92           |               required  

93           |               @error('no_p_irt')  

94           |                   <div class="invalid-feedback">  

95           |                       {{ $message }}  

96           |                   </div>  

97           |               @enderror  

98           |           </div>  

99           |       </div><br>  

100          |       <div class="row">  

101          |           <div class="col-4">  

102          |               <label for="name">Deskripsi</label>  

103          |           </div>  

104          |           <div class="col-8">  

105          |               <textarea id="c_information"  

106          |                   placeholder="Input your product's description" class="form-control"  

107          |                   name="deskripsi"></textarea>  

108          |               @error('deskripsi')  

109          |                   <div class="invalid-feedback">  

110          |                       {{ $message }}  

111          |                   </div>  

112          |               @enderror  

113          |           </div>  

114          |       </div><br>  

115          |   </div>  

116          |   <div class="form-section">  

117          |       <div class="row">  

118          |           <div class="col-4">  

119          |               <label for="kemasan">Kemasan Produk</label>  

120          |           </div>  

121          |           <div class="col-8">  

122          |               @foreach($kemasan as $km)  

123          |                   <div class="form-check">  

124          |                       <input class="form-check-input" type="checkbox" name="kemasan_id[]" value="{{ $km->id }}" id="kemasan_({$km->id})">  

125          |                       <label class="form-check-label" for="kemasan_({$km->id})">  

126          |                           {{ ($km->ukuran)}  

127          |                       </label>  

128          |                   </div>  

129          |               @endforeach  

130          |               @error('kemasan_id')  

131          |                   <div class="invalid-feedback">  

132          |                       {{ $message }}  

133          |                   </div>  

134          |               @enderror  

135          |           </div>  

136          |       </div><br>  

137          |       <div id="detail_produk_form">  

138          |           <!-- Form untuk detail produk akan muncul di sini -->  

139          |       </div>  

140          |   </div>  

141          |   <div class="form-section">  

142          |       <div class="row">  

143          |           <div class="col-lg-12">  

144          |               <a href="javascript:;" onclick="cloneImage()">  

145          |                   class="btn btn-outline-secondary btn-sm">  

146          |                       <i class="fa fa-plus-square"></i> Tambah gambar  

147          |               </a>  

148          |               <div class="form-group row">  

149          |                   <div class="col-4 clone-1" id="clone-1">  

150          |                       <div class="fileinput fileinput-new" data-provides="fileinput">  

151          |                           <div class="fileinput-new img-thumbnail"  

152          |                               style="width: 200px; height: 150px;">  

153          |                                 

154          |                           </div>  

155          |                           <div class="fileinput-preview fileinput-exists img-thumbnail" style="margin-top: 5px;">

```



```

277     inputkodeProduk.classList.add('form-control');
278     inputkodeProduk.placeholder = 'Input your product code';
279     detailProdukDiv.appendChild(inputkodeProduk);
280
281     var inputStok = document.createElement('input');
282     inputStok.type = 'number';
283     inputStok.name = 'stok[]';
284     inputStok.classList.add('form-control');
285     inputStok.placeholder = 'Stok';
286     detailProdukDiv.appendChild(inputStok);
287
288     var inputHarga = document.createElement('input');
289     inputHarga.type = 'number';
290     inputHarga.name = 'harga[]';
291     inputHarga.classList.add('form-control');
292     inputHarga.placeholder = 'Harga';
293     detailProdukDiv.appendChild(inputHarga);
294
295     var inputBerat = document.createElement('input');
296     inputBerat.type = 'number';
297     inputBerat.name = 'berat_pengiriman[]';
298     inputBerat.classList.add('form-control');
299     inputBerat.placeholder = 'Berat Pengiriman (dalam gram)';
300     detailProdukDiv.appendChild(inputBerat);
301
302     // Tambahkan form detail produk ke dalam div container
303     detailProdukForm.appendChild(detailProdukDiv);
304   });
305 });
306 });
307 });
308
309 $(document).ready(function () {
310   function getCities() {
311     let provinceId = $('select[name="provinsi"]').val();
312     if (provinceId) {
313       $.ajax({
314         url: '/api/provinsi/' + provinceId + '/kota',
315         type: "GET",
316         dataType: "json",
317         success: function(data) {
318           var selectedCityId = "{{ old('kota') }}"; // Get the selected city ID
319           var cityDropdown = $('select[name="kota"]'); // Select the city dropdown
320
321           // Empty the dropdown options
322           cityDropdown.empty();
323
324           // Iterate over the cities data
325           $.each(data.kotas, function(key, value) {
326             // Create the option element
327             var option = $(<option></option>).val(key).text(value);
328
329             // Check if the current city ID matches the selected city
330             if (key == selectedCityId) {
331               option.attr('selected', 'selected'); // Set it as selected
332             }
333
334             // Append the option to the dropdown
335             cityDropdown.append(option);
336           });
337
338           // Enable and show the city dropdown
339           cityDropdown.prop('disabled', false);
340           cityDropdown.show();
341         },
342         error: function(xhr, status, error) {
343           console.error(error);
344         }
345       });
346     } else {
347       $('select[name="kota"]').empty();
348     }
349   }
350
351   // Event listener untuk perubahan dropdown provinsi
352   $('select[name="provinsi"]').on('change', function() {
353     getCities(); // Panggil fungsi getCities() saat perubahan dropdown provinsi
354   });
355
356   getCities(); // Panggil fungsi getCities() saat halaman dimuat
357
358
359 });
360
361 function fileValidation() {
362   var fileInput = document.getElementById('imb');
363   var filePath = fileInput.value;
364   var allowedExtensions = /(.(jpg|jpeg|png|pdf)$)/i;
365   if (!allowedExtensions.exec(filePath)) {
366     alert("Please upload file having extensions .jpeg/.jpg/.png/.pdf only.");
367     fileInput.value = '';
368     return false;
369   } else {
370     //Image preview
371     if (fileInput.files && fileInput.files[0]) {
372       var reader = new FileReader();
373       reader.onload = function (e) {
374         document.getElementById('imagePreview').innerHTML = '<iframe id="image" src=' + e.target.result +
375         ' style="width: 100%; height: 100%;></iframe>';
376       };
377       reader.readAsDataURL(fileInput.files[0]);
378     }
379   }
380 }
381
382 function imgfileValidation() {
383   var fileInput = document.getElementById('img');
384   var filePath = fileInput.value;
385   var allowedExtensions = /(.(jpg|jpeg|png|pdf)$)/i;
386   if (!allowedExtensions.exec(filePath)) {
387     alert("Please upload file having extensions .jpeg/.jpg/.png/.pdf only.");
388     fileInput.value = '';
389     return false;
390   } else {
391     //Image preview
392     if (fileInput.files && fileInput.files[0]) {
393       var reader = new FileReader();
394       reader.onload = function (e) {
395         document.getElementById('imgPreview').innerHTML = '<iframe id="image" src=' + e.target.result +

```

```

395         " style='width: 100%; height: 100%;'>
396     );
397     reader.readAsDataURL(fileInput.files[0]);
398   }
399 }
400
401 function removeImage() {
402   var image = document.getElementById('image');
403   var preview = document.getElementById('imagePreview');
404   preview.removeChild(image);
405 }
406
407 function removeImg() {
408   var preview = document.getElementById('imgPreview');
409   while(preview.firstChild) {
410     preview.removeChild(preview.firstChild);
411   }
412 }
413
414 function addPaymentMethod() {
415   let checked = [];
416   for (var i = 0; i < $('.checkbox-paymentMethod').length; i++) {
417     if($('.checkbox-paymentMethod').eq(i).prop('checked') == true) {
418       checked.push($('.checkbox-paymentMethod').eq(i).val());
419     }
420   }
421   console.log(checked);
422   $.post(
423     "{{url('api/payment/add-paymentMethod'))}}",
424     {_token: "{{ csrf_token() }}",
425      id: checked
426    },
427     function (result) {
428       console.log(result.length)
429       $('.method').remove();
430       for (var i = 0; i < result.length; i++) {
431         var div = $(<div class="form-row method">
432           <div class="form-group col-md-4">
433             <label> + result[i].nama_bank + </label>
434           </div>
435           <div class="form-group col-md-8">
436             <input name="norek[]>" type="text"
437               placeholder="Nomor rekening"
438               class="form-control" required>
439           </div>
440           <div class="form-group col-md-8">
441             <input name="metode_pembayaran[]>" type="hidden"
442               value="" + result[i].id + ""
443               class="form-control">
444           </div>
445           </div>;
446
447         $('#row-paymentMethod').append(div);
448       }
449     }
450   )
451 }
452
453
454
455
456 function check() {
457   $("#row-dp").toggle(this.checked);
458 }
459
460 function cloneImage() {
461   $("#clone-1")
462     .eq(0)
463     .clone()
464     .find("input").val("").end() // ***
465     .show()
466     .insertAfter(".clone-1:last");
467 }
468
469 function delClone(data) {
470   console.log(data.closest('.clone-1'));
471
472   if($('.clone-1').length > 1) {
473     data.closest('.clone-1').remove()
474   } else {
475     swal({
476       title: "Warning!",
477       text: "Select at least one image",
478       type: "warning",
479       confirmButtonColor: "#D06855",
480       confirmButtonText: "OK",
481       closedOnConfirm: false,
482       closedOnCancel: false
483     });
484   }
485
486 
```

## 7. Halaman edit produk

```
48 @section('content')
49     <div class="content-wrapper">
50         <div class="container-fluid">
51             <div class="row justify-content-md-center">
52                 <div class="col-md-9 ">
53                     <div class="card px-1 py-1 mt-1 shadow">
54                         <h2 class="text-danger text-center mt-3 mb-4">Menambahkan Data Produk</h2>
55
56                         <div class="nav nav-fill my-3">
57                             <label class="nav-link shadow-sm step0 border ml-2 ">Informasi Umum Produk</label>
58                             <label class="nav-link shadow-sm step1 border ml-2 ">Rincian Produk</label>
59                             <label class="nav-link shadow-sm step2 border ml-2 ">Gambar Produk</label>
60                         </div>
61
62                         <form action="{{ route('umkm.product.update', $produk->id) }}" method="post" class="employee-form" enctype="multipart/form-data">
63                             @method('PUT')
64                             @csrf
65                             <div class="form-section">
66                                 <div class="row">
67                                     <div class="col-4">
68                                         <label for="field_name">Nama Produk</label>
69                                     </div>
70                                     <div class="col-8">
71                                         <input name="nama" type="text" class="form-control" id="c_name"
72                                         placeholder="Input your produk name" value="{{$produk->nama}}>
73                                         @error('nama')
74                                         <div class="invalid-feedback">
75                                         {{ $message }}</div>
76                                         @enderror
77                                     </div>
78                                     </div><br>
79                                     <div class="row">
80                                         <div class="col-4">
81                                             <label for="kategori">Kategori Produk</label>
82                                         </div>
83                                         <div class="col-8">
84                                             <select name="kategori" id="s_kategori_create" class="form-control" data-live-search="true" onchange="this.className = ''">
85                                                 @foreach($kategori as $kt)
86                                                 <option value="{{ $kt->id }}> {{ $kt->id == $produk->kategori_id ? 'selected' : '' }}>{{ $kt->nama_kategori }}</option>
87                                                 @endif
88                                             </select>
89                                             @error('kategori')
90                                                 <div class="invalid-feedback">
91                                                 {{ $message }}</div>
92                                             @enderror
93                                         </div>
94                                         <br>
95                                         <div class="row">
96                                             <div class="col-4">
97                                                 <label for="field_name">No Izin P-IRT</label>
98                                             </div>
99                                             <div class="col-8">
100                                                 <input name="no_p_int" type="text" class="form-control"
101                                                 id="c_no_p_int" placeholder="Input your p_int"
102                                                 value="{{$produk->no_p_int}}>
103                                                 @error('no_p_int')
104                                                 <div class="invalid-feedback">
105                                                 {{ $message }}</div>
106                                                 @enderror
107                                             </div>
108                                         </div>
109                                         <br>
110                                         <div class="row">
111                                             <div class="col-4">
112                                                 <label for="name">Deskripsi</label>
113                                             </div>
114                                             <div class="col-8">
115                                                 <textarea id="c_information"
116                                                     placeholder="Input your product's description" class="form-control"
117                                                     name="deskripsi">{{ $produk->deskripsi }}</textarea>
118                                                 @error('deskripsi')
119                                                 <div class="invalid-feedback">
120                                                 {{ $message }}</div>
121                                                 @enderror
122                                             </div>
123                                         </div>
124                                         <br>
125                                         <div>
126                                         </div>
127                                         </div>
128                                         </div>
129                                         </div>
130                                         </div>
131                                         </div>
132                                         </div>
133                                         </div>
134                                         </div>
135                                         </div>
136                                         </div>
137                                         </div>
138                                         </div>
139                                         </div>
140                                         </div>
141                                         </div>
142                                         </div>
143                                         </div>
144                                         </div>
145                                         </div>
146                                         </div>
147                                         </div>
148                                         </div>
149                                         </div>
150                                         </div>
151                                         </div>
152                                         </div>
153                                         </div>
154                                         </div>
155                                         </div>
156                                         </div>
157                                         </div>
158                                         </div>
159                                         </div>
160                                         </div>
161                                         </div>
162                                         </div>
163                                         </div>
164                                         </div>
165                                         </div>
166                                         </div>
167                                         </div>
168                                         </div>
169                                         </div>
170                                         </div>
171                                         </div>
172                                         </div>
173                                         </div>
174                                         </div>
175                                         </div>
176                                         </div>
177                                         </div>
178                                         </div>
179                                         </div>
180                                         </div>
181                                         </div>
182                                         </div>
183                                         </div>
184                                         </div>
185                                         </div>
186                                         </div>
187                                         </div>
188                                         </div>
189                                         </div>
190                                         </div>
191                                         </div>
192                                         </div>
193                                         </div>
194                                         </div>
195                                         </div>
196                                         </div>
197                                         </div>
198                                         </div>
199                                         </div>
200                                         </div>
201                                         </div>
202                                         </div>
203                                         </div>
204                                         </div>
205                                         </div>
206                                         </div>
207                                         </div>
208                                         </div>
209                                         </div>
210                                         </div>
211                                         </div>
212                                         </div>
213                                         </div>
214                                         </div>
215                                         </div>
216                                         </div>
217                                         </div>
218                                         </div>
219                                         </div>
220                                         </div>
221                                         </div>
222                                         </div>
223                                         </div>
224                                         </div>
225                                         </div>
226                                         </div>
227                                         </div>
228                                         </div>
229                                         </div>
230                                         </div>
231                                         </div>
232                                         </div>
233                                         </div>
234                                         </div>
235                                         </div>
236                                         </div>
237                                         </div>
238                                         </div>
239                                         </div>
240                                         </div>
241                                         </div>
242                                         </div>
243                                         </div>
244                                         </div>
245                                         </div>
246                                         </div>
247                                         </div>
248                                         </div>
249                                         </div>
250                                         </div>
251                                         </div>
252                                         </div>
253                                         </div>
254                                         </div>
255                                         </div>
256                                         </div>
257                                         </div>
258                                         </div>
259                                         </div>
260                                         </div>
261                                         </div>
262                                         </div>
263                                         </div>
264                                         </div>
265                                         </div>
266                                         </div>
267                                         </div>
268                                         </div>
269                                         </div>
270                                         </div>
271                                         </div>
272                                         </div>
273                                         </div>
274                                         </div>
275                                         </div>
276                                         </div>
277                                         </div>
278                                         </div>
279                                         </div>
280                                         </div>
281                                         </div>
282                                         </div>
283                                         </div>
284                                         </div>
285                                         </div>
286                                         </div>
287                                         </div>
288                                         </div>
289                                         </div>
290                                         </div>
291                                         </div>
292                                         </div>
293                                         </div>
294                                         </div>
295                                         </div>
296                                         </div>
297                                         </div>
298                                         </div>
299                                         </div>
300                                         </div>
301                                         </div>
302                                         </div>
303                                         </div>
304                                         </div>
305                                         </div>
306                                         </div>
307                                         </div>
308                                         </div>
309                                         </div>
310                                         </div>
311                                         </div>
312                                         </div>
313                                         </div>
314                                         </div>
315                                         </div>
316                                         </div>
317                                         </div>
318                                         </div>
319                                         </div>
320                                         </div>
321                                         </div>
322                                         </div>
323                                         </div>
324                                         </div>
325                                         </div>
326                                         </div>
327                                         </div>
328                                         </div>
329                                         </div>
330                                         </div>
331                                         </div>
332                                         </div>
333                                         </div>
334                                         </div>
335                                         </div>
336                                         </div>
337                                         </div>
338                                         </div>
339                                         </div>
340                                         </div>
341                                         </div>
342                                         </div>
343                                         </div>
344                                         </div>
345                                         </div>
346                                         </div>
347                                         </div>
348                                         </div>
349                                         </div>
350                                         </div>
351                                         </div>
352                                         </div>
353                                         </div>
354                                         </div>
355                                         </div>
356                                         </div>
357                                         </div>
358                                         </div>
359                                         </div>
360                                         </div>
361                                         </div>
362                                         </div>
363                                         </div>
364                                         </div>
365                                         </div>
366                                         </div>
367                                         </div>
368                                         </div>
369                                         </div>
370                                         </div>
371                                         </div>
372                                         </div>
373                                         </div>
374                                         </div>
375                                         </div>
376                                         </div>
377                                         </div>
378                                         </div>
379                                         </div>
380                                         </div>
381                                         </div>
382                                         </div>
383                                         </div>
384                                         </div>
385                                         </div>
386                                         </div>
387                                         </div>
388                                         </div>
389                                         </div>
390                                         </div>
391                                         </div>
392                                         </div>
393                                         </div>
394                                         </div>
395                                         </div>
396                                         </div>
397                                         </div>
398                                         </div>
399                                         </div>
400                                         </div>
401                                         </div>
402                                         </div>
403                                         </div>
404                                         </div>
405                                         </div>
406                                         </div>
407                                         </div>
408                                         </div>
409                                         </div>
410                                         </div>
411                                         </div>
412                                         </div>
413                                         </div>
414                                         </div>
415                                         </div>
416                                         </div>
417                                         </div>
418                                         </div>
419                                         </div>
420                                         </div>
421                                         </div>
422                                         </div>
423                                         </div>
424                                         </div>
425                                         </div>
426                                         </div>
427                                         </div>
428                                         </div>
429                                         </div>
430                                         </div>
431                                         </div>
432                                         </div>
433                                         </div>
434                                         </div>
435                                         </div>
436                                         </div>
437                                         </div>
438                                         </div>
439                                         </div>
440                                         </div>
441                                         </div>
442                                         </div>
443                                         </div>
444                                         </div>
445                                         </div>
446                                         </div>
447                                         </div>
448                                         </div>
449                                         </div>
450                                         </div>
451                                         </div>
452                                         </div>
453                                         </div>
454                                         </div>
455                                         </div>
456                                         </div>
457                                         </div>
458                                         </div>
459                                         </div>
460                                         </div>
461                                         </div>
462                                         </div>
463                                         </div>
464                                         </div>
465                                         </div>
466                                         </div>
467                                         </div>
468                                         </div>
469                                         </div>
470                                         </div>
471                                         </div>
472                                         </div>
473                                         </div>
474                                         </div>
475                                         </div>
476                                         </div>
477                                         </div>
478                                         </div>
479                                         </div>
480                                         </div>
481                                         </div>
482                                         </div>
483                                         </div>
484                                         </div>
485                                         </div>
486                                         </div>
487                                         </div>
488                                         </div>
489                                         </div>
490                                         </div>
491                                         </div>
492                                         </div>
493                                         </div>
494                                         </div>
495                                         </div>
496                                         </div>
497                                         </div>
498                                         </div>
499                                         </div>
500                                         </div>
501                                         </div>
502                                         </div>
503                                         </div>
504                                         </div>
505                                         </div>
506                                         </div>
507                                         </div>
508                                         </div>
509                                         </div>
510                                         </div>
511                                         </div>
512                                         </div>
513                                         </div>
514                                         </div>
515                                         </div>
516                                         </div>
517                                         </div>
518                                         </div>
519                                         </div>
520                                         </div>
521                                         </div>
522                                         </div>
523                                         </div>
524                                         </div>
525                                         </div>
526                                         </div>
527                                         </div>
528                                         </div>
529                                         </div>
530                                         </div>
531                                         </div>
532                                         </div>
533                                         </div>
534                                         </div>
535                                         </div>
536                                         </div>
537                                         </div>
538                                         </div>
539                                         </div>
540                                         </div>
541                                         </div>
542                                         </div>
543                                         </div>
544                                         </div>
545                                         </div>
546                                         </div>
547                                         </div>
548                                         </div>
549                                         </div>
550                                         </div>
551                                         </div>
552                                         </div>
553                                         </div>
554                                         </div>
555                                         </div>
556                                         </div>
557                                         </div>
558                                         </div>
559                                         </div>
560                                         </div>
561                                         </div>
562                                         </div>
563                                         </div>
564                                         </div>
565                                         </div>
566                                         </div>
567                                         </div>
568                                         </div>
569                                         </div>
570                                         </div>
571                                         </div>
572                                         </div>
573                                         </div>
574                                         </div>
575                                         </div>
576                                         </div>
577                                         </div>
578                                         </div>
579                                         </div>
580                                         </div>
581                                         </div>
582                                         </div>
583                                         </div>
584                                         </div>
585                                         </div>
586                                         </div>
587                                         </div>
588                                         </div>
589                                         </div>
590                                         </div>
591                                         </div>
592                                         </div>
593                                         </div>
594                                         </div>
595                                         </div>
596                                         </div>
597                                         </div>
598                                         </div>
599                                         </div>
600                                         </div>
601                                         </div>
602                                         </div>
603                                         </div>
604                                         </div>
605                                         </div>
606                                         </div>
607                                         </div>
608                                         </div>
609                                         </div>
610                                         </div>
611                                         </div>
612                                         </div>
613                                         </div>
614                                         </div>
615                                         </div>
616                                         </div>
617                                         </div>
618                                         </div>
619                                         </div>
620                                         </div>
621                                         </div>
622                                         </div>
623                                         </div>
624                                         </div>
625                                         </div>
626                                         </div>
627                                         </div>
628                                         </div>
629                                         </div>
630                                         </div>
631                                         </div>
632                                         </div>
633                                         </div>
634                                         </div>
635                                         </div>
636                                         </div>
637                                         </div>
638                                         </div>
639                                         </div>
640                                         </div>
641                                         </div>
642                                         </div>
643                                         </div>
644                                         </div>
645                                         </div>
646                                         </div>
647                                         </div>
648                                         </div>
649                                         </div>
650                                         </div>
```

```

127     </div>
128     <div class="form-section">
129       <div class="row">
130         <div class="col-4">
131           <label for="kemasan">Kemasan Produk</label>
132         </div>
133         <div class="col-8">
134           @foreach($kemasan as $km)
135             <div class="selectgroup selectgroup-pills">
136               <input class="form-check-input" type="checkbox" name="kemasan_id[]" value="{{($km->id)}" id="kemasan_{{$km->id}}">
137               <label class="form-check-label" for="kemasan_{{$km->id}}">
138                 {{($km->ukuran)}}
139               </label>
140             </div>
141           @endforeach
142           @error('kemasan_id')
143             <div class="invalid-feedback">
144               {{($message)}}
145             </div>
146           @enderror
147         </div>
148       </div><br>
149       <div id="detail_produk_form">
150         <div style="display: grid; grid-template-columns: repeat(auto-fill, minmax(auto, 1fr)); grid-gap: 20px;">
151           @foreach($selectedKemasanIds as $kemasanId)
152             @php
153               $detailProduk = $produks->Detail_Produk->where('kemasan_id', $kemasanId)->first();
154             @endphp
155             <div class="detail-produk">
156               <label>Kemasan: {{($detailProduk->kemasan->ukuran)}}</label><br>
157               <label>Kode Produk: &nbsp;</label><input type="text" id="kode_produku_{{$kemasanId}}" name="kode_produku[] value="{{($detailProduk->kode_produku)}}"><br>
158               <label>Stok: &nbsp;</label><input type="number" id="stok_{{$kemasanId}}" name="stok[] value="{{($detailProduk->stok)}}"><br>
159               <label>Harga: &nbsp;</label><input type="number" id="harga_{{$kemasanId}}" name="harga[] value="{{($detailProduk->harga)}}"><br>
160               <label>Berat Pengiriman: &nbsp;</label><input type="number" id="berat_pengiriman_{{$kemasanId}}" name="berat_pengiriman[] value="{{($detailProduk->berat_pengiriman)}}"><br>
161             </div>
162           @endforeach
163         </div>
164       </div>
165     </div>

166     <div class="form-section">
167       <div class="row">
168         <div class="col-lg-12">
169           <a href="javascript:;" onclick="cloneImage()">
170             <div class="btn btn-outline-secondary btn-sm">
171               <i class="fa fa-plus-square"></i> Tambah gambar
172             </div>
173           </div>
174           <div class="row">
175             @foreach ($img_produk as $img_produks)
176               <div class="col-2 clone-1" id="clone-1">
177                 <div class="fileinput fileinput-exist" data-provides="fileinput">
178                   <div class="fileinput-new img-thumbnail"
179                     style="width: 200px; height: 150px;">
180                     
181                   </div>
182                   <div class="fileinput-preview fileinput-exists img-thumbnail"
183                     style="max-width: 200px; max-height: 150px;"></div>
184                   <div>
185                     <a href="javascript:void(0);"
186                       onclick="delImage(this,{{$img_produks->id}})"
187                       class="btn btn-outline-secondary">
188                         Delete
189                       </a>
190                     </div>
191                   </div>
192                 </div>
193               @endforeach
194               <div class="col-4 clone-1" id="clone-1">
195                 <div class="fileinput fileinput-new" data-provides="fileinput">
196                   <div class="fileinput-new img-thumbnail"
197                     style="width: 200px; height: 150px;">
198                     
199                   </div>
200                   <div class="fileinput-preview fileinput-exists img-thumbnail"
201                     style="max-width: 200px; max-height: 150px;"></div>
202                 </div>
203               </div>
204             </div>
205           </div>
206           <span class="btn btn-outline-secondary btn-file"><span>
207             <input type="file" new>Select
208             image</span><span>
209             <input type="file" exists="">Change</span><input type="file"
210             name="image_product[]"/></span>
211             <a href="#" class="btn btn-outline-secondary fileinput-exists"
212               data-dismiss="fileinput">Remove</a>
213             <a href="javascript:;" onclick="delClone(this)">
214               <span>Delete
215               </span>
216             </a>
217           </div>
218         </div>
219       </div>
220     </div>
221   </div>
222 </div>
223 </div>
224 </div>
225 <div class="form-navigation mt-3">
226   <button type="button" class="previous btn btn-primary float-left">&lt; Previous</button>
227   <button type="button" class="next btn btn-primary float-right">Next &gt;</button>
228   <button type="submit" class="btn btn-success float-right">Submit</button>
229 </div>
230 </div>
231 </form>
232 </div>
233 </div>
234 </div>
235 </div>
236 </div>
237 </div>
238 @endsection

```

## 8. Halaman show produk

```
1  @extends('layouts.penjual')
2
3  @section('css')
4  <style>
5  .image-container {
6      text-align: center; /* Optional: Centers the images horizontally */
7  }
8
9  .custom-image {
10     display: inline-block;
11     width: 200px;
12     height: auto;
13     margin-right: 10px; /* Adds spacing between images */
14 }
15 </style>
16
17 @endsection
18
19 @section('content')
20 <div class="content-wrapper">
21 <section class="content">
22 <div class="container-fluid">
23     <div class="row">
24         <div class="col-md-12 col-md-12 col-lg-12">
25             <div class="row">
26                 <div class="col-12">
27                     <div class="card">
28                         <div class="card-header">
29                             <h4 class="card-title">{{ $produk->nama }}</h4>
30                             <a href="#" class="btn btn-warning btn-sm" style="float:right;margin-left:5px;" data-toggle="modal" data-target="#tambahStokModal">Tambah Stok</a>
31                         </div>
32                         <!-- Modal Tambah Stok -->
33                         <div class="modal fade" id="tambahStokModal" tabindex="-1" role="dialog" aria-labelledby="tambahStokModalLabel" aria-hidden="true">
34                             <div class="modal-dialog" role="document">
35                                 <div class="modal-content">
36                                     <div class="modal-header">
37                                         <h5 class="modal-title" id="tambahStokModalLabel">Tambah Stok</h5>
38                                         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
39                                             <span aria-hidden="true">&amptimes</span>
40                                         </button>
41                                     </div>
42                                     <form action="{{ route('tambah-stok', ['id' => $produk->id]) }}" method="POST">
43                                         @csrf
44                                         <div class="modal-body">
45                                             <div class="form-group">
46                                                 <label for="selected_kemasan">Pilih Kemasan</label>
47                                                 <select name="selected_kemasan" id="selected_kemasan" class="form-control" required>
48                                                     <option value="" disabled selected>Pilih Kemasan</option>
49                                                     @foreach ($kemasans as $kemasan)
50                                                         <option value="{{ $kemasan->id }}>{{ $kemasan->ukuran }}</option>
51                                                     @endforeach
52                                                 </select>
53                                             </div>
54                                             <div class="form-group">
55                                                 <label for="jumlah_stok">Jumlah Stok</label>
56                                                 <input type="number" name="jumlah_stok" id="jumlah_stok" class="form-control" required>
57                                             </div>
58                                         </div>
59                                         <div class="modal-footer">
60                                             <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
61                                             <button type="submit" class="btn btn-primary">Simpan</button>
62                                         </div>
63                                         @if ($errors->any())
64                                             <div class="alert alert-danger">
65                                                 <ul>
66                                                     @foreach ($errors->all() as $error)
67                                                         <li>{{ $error }}</li>
68                                                     @endforeach
69                                                 </ul>
70                                             </div>
71                                         @endif
72                                         </div>
73                                     </form>
74                                 </div>
75                             </div>
76                         <div class="card-body">
77                             <span>Kategori : <b>{{ $produk->kategori->nama_kategori }}</b><br>
78                             </span><br>
79                         </div>
80                     </div>
81                 </div>
82             </div>
83         </div>
84     </div>
85 </section>
86 </div>
```

```

80     <span>No_PIRT : <b>{{ $produk->no_pirt }}</b><br>
81     </span><br>
82     <span>Deskripsi Produk : <b>{{ $produk->deskripsi }}</b><br>
83     </span><br>
84
85     <span>
86         <label>kemasan :</label><br>
87         @foreach ($kemasans as $kemasan)
88             <label style="display: inline-block; margin-right: 10px;">
89                 <input type="radio" name="selected_kemasan" value="{{ $kemasan->id }}> {{ $kemasan->ukuran }}</label>
90             @endforeach
91         </span>
92     </span>
93     <div id="selected_kemasan_info">
94         <span>Kode Produk : <b id="kode_produk"></b><br></span>
95         <span>Stok : <b id="stok"></b><br></span>
96         <span>Harga : <b id="harga"></b><br></span>
97     </div> <br>
98     <div data-crop-image="285">
99         @foreach ($img_produk as $img)
100             
101         @endforeach
102     </div>
103     </div>
104     </div>
105     </div>
106     </div>
107 </div>
108
109     </div>
110     </div>
111 </div>
112 </section>
113 </div>
114 @endsection
115
116 @section('script')
117 <script>
118
119 var kemasans = @json($kemasans);
120 $(document).ready(function() {
121     $('input[name="selected_kemasan"]').change(function() {
122         var selectedKemasanId = $(this).val();
123         var produkId = '{{ $produk->id }}'; // Make sure to replace this with the actual produk_id
124
125         // Make an AJAX request to fetch detail_produk data
126         $.ajax({
127             url: '/fetch-detail/' + produkId + '/' + selectedKemasanId,
128             method: 'GET',
129             success: function(response) {
130                 $('#kode_produk').text(response.kode_produk);
131                 $('#stok').text(response.stok);
132                 $('#harga').text(response.harga);
133             },
134             error: function() {
135                 console.log('Error fetching detail_produk data.');
136             }
137         });
138     });
139 });
140
141 </script>
142
143 @endsection

```

## 9. Halaman index transaksi

```

1  @extends('layouts.penjual')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5  <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}"/>
6  @endsection
7
8  @section('content')
9  <div class="content-wrapper">
10    <section class="content">
11      <div class="container-fluid">
12        <div class="row">
13          @include('penjual.manage_transaction.reject')
14          <div class="col-12">
15            <div class="card">
16              <div class="card-header">
17                <h3 class="card-title">Transaksi Pembelian</h3>
18              </div>
19              <div class="card-body">
20                <table id="example1" class="table table-bordered table-striped dataTables-booking">
21                  <thead>
22                    <tr>
23                      <th>Tanggal</th>
24                      <th>Umkm</th>
25                      <th>Total Pembelian</th>
26                      <th>Status</th>
27                      <th>Action</th>
28                  </tr>
29                </thead>
30                <tbody>
31                  <tr>
32                    <td>2023-09-15</td>
33                    <td>Umkm A</td>
34                    <td>Rp 100,000</td>
35                    <td>Pending</td>
36                    <td><a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a></td>
37                  </tr>
38                </tbody>
39              </table>
40            </div>
41          </div>
42        </div>
43      </div>
44    </section>
45  </div>

```

```

41     <!-- /.row -->
42   </div>
43   <!-- /.container-fluid -->
44
45 <section>
46 </div>
47 {!! Form::open(['method'=>'PATCH', 'route' => ['penjual.transaksi.confirm', 0], 'style' =>
48 'display:none','id'=>'confirm']) !!}
49 {!! Form::close() !!}
50 {!! Form::open(['method'=>'DELETE', 'route' => ['transaksi.destroy',0],
51 'style'=>'display:none','id'=>'deleted_transaksi']) !!}
52 {!! Form::close() !!}
53
54 @endsection
55
56 @section('script')
57 <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
58 <script>
59   $(document).ready(function () {
60     $('.dataTables-bookings').DataTable({
61       responsive: true,
62       searchDelay: 500,
63       processing: true,
64       buttons: [
65         {
66           extend: 'print',
67           className: 'btn-inverse',
68           exportOptions: {
69             columns: [0, 1]
70           }
71         },
72         'copy',
73         {
74           extend: 'excel',
75           className: 'btn-inverse',
76           exportOptions: {
77             columns: [0, 1]
78           }
79         },
80         'csvHtml5',
81         {
82           extend: 'pdf',
83           className: 'btn-inverse',
84           exportOptions: {
85             columns: [0, 1]
86           }
87         }
88       ],
89       // serverSide: true,
90       ajax: {
91         url: "{{url('/api/transaksi')}}?data=all&penjual_id={{Auth::user()->penjual->id}},
92         dataSrc: ''
93       },
94       columns: [
95         {
96           data: 'date'
97         },
98         {
99           data: 'umkm'
100        },
101        {
102           data: 'total_harga'
103        },
104        {
105           data: 'p_status'
106        },
107        {
108           data: 'id',
109           responsivePriority: -1
110        },
111      ],
112      columnDefs: [
113        {
114          targets: -1,
115          title: "Action",
116          orderable: false,
117          render: function (data, type, full, meta) {
118            var base = "{{url('/')}}"
119
120            if (full.status_pembelian != 0 || full.expired == 1) {
121              return '<a href=' + base + 'index.php?r=transaksi/confirm/' + full.id + '>Confirm</a> &nbsp; <a href=' + base + 'index.php?r=transaksi/delete/' + full.id + '>Delete</a>';
122            } else {
123              return '<a href=' + base + 'index.php?r=transaksi/confirm/' + full.id + '>Confirm</a> &nbsp; <a href=' + base + 'index.php?r=transaksi/edit/' + full.id + '>Edit</a> &nbsp; <a href=' + base + 'index.php?r=transaksi/delete/' + full.id + '>Delete</a>';
124            }
125          }
126        }
127      ]
128    })
129  })
130 </script>

```

```
120
121     return `` +
122       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-light" onclick="show(` + data + `)" href="#/transaksi/` + data + `/show"><i class="fa fa-info">/</i>` +
123
124       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-danger" onclick="confirm_delete(` + data +
125         `)`)" href="javascript:void(0)"><i class="fa fa-trash">/</i>` </a>`;
126   } else if (full_status_pembelian == 0 && full_status_pembayaran == 0) {
127     return `` +
128       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-light" onclick="show(` + data + `)" href="#/transaksi/` + data + `/show"><i class="fa fa-info">/</i>` +
129
130       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-danger" onclick="confirm_delete(` + data +
131         `)`)" href="javascript:void(0)"><i class="fa fa-trash">/</i>` </a>`;
132   } else {
133     return `` +
134       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-light" onclick="show(` + data + `)" href="#/transaksi/` + data + `/show"><i class="fa fa-info">/</i>` +
135
136       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-success" onclick="confirm(` + data +
137         `)`)" href="javascript:void(0)"><i class="fa fa-check">/</i>` </a> +
138
139       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-danger" onclick="reject(` + data +
140         `)`)" href="javascript:void(0)"><i class="fa fa-times">/</i>` </a> +
141
142       `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-danger" onclick="confirm_delete(` + data +
143         `)`)" href="javascript:void(0)"><i class="fa fa-trash">/</i>` </a>`;
144   }
145
146   |
147   |
148   |
149   |
150   |
151   |
152   |
153   |
154   |
155   |   ],
156   ],
157 });
158
159   table.on('order.dt search.dt', function () {
160
161     table.column(0, {
162       search: 'applied',
163       order: 'applied'
164     }).nodes().each(function (cell, i) {
165       cell.innerHTML = i + 1;
166     });
167   }).draw();
168   $('#btn-print').on('click', function (e) {
169     e.preventDefault();
170     table.button(0).trigger();
171   });
172   $('#btn-copy').on('click', function (e) {
173     e.preventDefault();
174     table.button(1).trigger();
175   });
176   $('#btn-excel').on('click', function (e) {
177     e.preventDefault();
178     table.button(2).trigger();
179   });
180   $('#btn-csv').on('click', function (e) {
181     e.preventDefault();
182     table.button(3).trigger();
183   });
184   $('#btn-pdf').on('click', function (e) {
185     e.preventDefault();
186     table.button(4).trigger();
187   });
188
189   $('#c_field').prop("disabled", true);
190   $('#c_venue').prop("disabled", true);
191
192   $(".myselect2").select2({
193     width: '100%',
194     allowClear: true
195   });
196   $('.myselect2').val(0).change();
197
198   Function.dateChange() {
```

```

199     $('#c_venue').prop('disabled', false);
200     $('.myselect2').val(0).change();
201     $('.selectgroup-item').remove();
202   }
203
204   $('#c_venue').on('select2:select', function (e) {
205     let venue = $('#c_venue').val();
206     if (venue) {
207       $('#c_field').val(0).change();
208       $('.selectgroup-item').remove();
209       fetchDataField(venue);
210       $('#c_field').prop('disabled', false);
211     } else {
212       $('#c_field').prop('disabled', true);
213     }
214   });
215
216
217
218
219
220   function confirm(id) {
221     swal({
222       title: "Apakah permintaan pemesanan ini diterima?",
223       text: "Pemesanan yang diterima akan tercatat dalam transaksi",
224       type: "warning",
225       showCancelButton: true,
226       confirmButtonColor: "#5cc744",
227       confirmButtonText: "Ya, terima!",
228       cancelButtonText: "Tidak, batalkan!",
229       closeOnConfirm: false,
230       closeOnCancel: false
231
232     },
233     function (isConfirm) {
234       if (isConfirm) {
235         swal("Konfirmasi!", "Pemesanan Diterima", "success");
236         $('#confirm').attr('action', "/transaksi/" + id + "/confirm");
237
238         $('#confirm').submit();
239       } else {
240         swal("Batal", "Kamu batal melakukan penerimaan pemesanan", "error");
241       }
242     });
243   }
244
245   function confirm_delete(id) {
246     swal({
247       title: "Apakah kamu yakin?",
248       text: "Data ini akan terhapus total",
249       type: "warning",
250       showCancelButton: true,
251       confirmButtonColor: "#ff2929",
252       confirmButtonText: "Ya, Hapus Data!",
253       cancelButtonText: "Tidak, batalkan!",
254       closeOnConfirm: false,
255       closeOnCancel: false
256
257     },
258     function (isConfirm) {
259       if (isConfirm) {
260         swal("Data Terhapus!", "Data tersebut telah terhapus", "success");
261         $('#deleted_transaksi').attr('action', "{!! route('transaksi.index') !!}" + id);
262         $('#deleted_transaksi').submit();
263       } else {
264         swal("Batal", "Anda batal menghapus data", "error");
265       }
266     });
267   }
268
269
270   function reject1(id) {
271     swal({
272       title: "Apakah permintaan pesanan ini ditolak?",
273       text: "Pesanan yang ditolak tidak dapat diproses kembali",
274       type: "warning",
275       showCancelButton: true,
276       confirmButtonColor: "#d9534f",
277       confirmButtonText: "Ya, Tolak",
278       cancelButtonText: "Tidak, batalkan!",
279       closeOnConfirm: false,
280       closeOnCancel: false
281     },
282     function (isConfirm) {
283       if (isConfirm) {
284         if (isConfirm) {
285           swal("Ditolak!", "Pesanan ini ditolak", "success");
286           $('#reject').attr('action', "/transaksi/" + id + "/reject");
287           $('#reject').submit();
288         } else {
289           swal("Batal", "Kamu batal melakukan penolakan pesanan", "error");
290         }
291       });
292     }
293
294   function reject(id) {
295     $('#reject-transaksi').toggle();
296     $('#pembelian_id').val(id);
297   }
298 
299 @endsection

```

## 10. Halaman *reject* transaksi

```

1  {!! Form::open(['method'=>'PATCH', 'route' => ['penjual.transaksi.reject', 0], 'id'=>'reject-transaksi']) !!}
2  <div class="col-12 animated fadeInRight" id="reject-transaksi" style="display:none">
3      <div class="card">
4          <div class="card-header">
5              <h3 class="card-title">Alasan penolakan pemesanan</h3>
6          </div>
7          <!-- /.card-header -->
8          <div class="card-body">
9              <div class="row">
10                 <div class="col-lg-12">
11                     <textarea name="reject_note" class="form-control" cols="200" rows="4" required></textarea>
12                     <input type="hidden" name="pembelian_id" id="pembelian_id">
13                 </div>
14             </div>
15             <div class="row">
16                 <div class="col-lg-12 d-flex justify-content-end">
17                     <button class="btn btn-primary" type="submit">Tolak Pesanan</button>
18                 </div>
19             </div>
20         </div>
21         <!-- /.card-body -->
22     </div>
23     <!-- /.card -->
24 </div>
25 {!! Form::close() !!}

```

## 11. Halaman *show* transaksi

```

1  @extends('layouts.penjual')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5  <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}"/>
6  @endsection
7
8  @section('content')
9  <div class="content-wrapper">
10    <section class="content">
11        <div class="container-fluid">
12            <div class="row">
13                <div class="col-12">
14                    <div class="card">
15                        <div class="card-header">
16                            <h3 class="card-title">Detail Transaksi Pemesanan</h3>
17                        </div>
18                        @if ($pembelian->status_pembelian == 1)
19
20                            <a type="button" href="#" class="btn btn-primary" style="float:right;margin-left:5px;" data-toggle="modal" data-target="#tambahStokModal">Input Noresi</a>
21
22                            <!-- Modal Tambah Stok -->
23                            <div class="modal fade" id="tambahStokModal" tabindex="-1" role="dialog" aria-labelledby="tambahStokModalLabel" aria-hidden="true">
24                                <div class="modal-dialog" role="document">
25                                    <div class="modal-content">
26                                        <div class="modal-header">
27                                            <h5 class="modal-title" id="tambahStokModalLabel">Input Noresi</h5>
28                                            <button type="button" class="close" data-dismiss="modal" aria-label="Close">
29                                                <span aria-hidden="true">&times;</span>
30                                            </button>
31                                        </div>
32                                        <form action="{{ route('input-noresi', ['id' => $pembelian->id]) }}" method="POST">
33                                            @csrf
34                                            <div class="modal-body">
35                                                <div class="form-group">
36                                                    <label for="jumlah_stok">Noresi</label>
37                                                    <input type="text" name="noresi" id="noresi" class="form-control" required>
38                                                </div>
39                                            <div class="modal-footer">
40                                                <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
41
42                                                <button type="submit" class="btn btn-primary">Simpan</button>
43                                            </div>
44
45                                        </div>
46
47                                        @elseif ($pembelian->status_pembelian == 0 && $pembelian->status_pembayaran == 0 && Carbon\Carbon::now() >= Carbon\Carbon::parse($pembelian->created_at)->span style="float:right;margin-left:5px;" class="badge badge-danger">Expired</span>
48
49                                        @elseif ($pembelian->status_pembelian == 0 && $pembelian->status_pembayaran == 0)
50
51                                            <span style="float:right;margin-left:5px;" class="badge badge-info">Menunggu Pembayaran</span>
52
53
54                                        @elseif ($pembelian->status_pembelian == 0)
55
56                                            <span style="float:right;margin-left:5px;" class="badge badge-primary">Menunggu Konfirmasi</span>
57
58
59                                        @elseif ($pembelian->status_pembelian == 2)
60
61                                            <a type="button" style="float:right;margin-left:5px;" class="btn btn-primary"
62                                                onclick="finish('{{ $pembelian->id }}')> href="javascript:void(0)">
63                                                Transaksi Selesai</a>
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79

```



```

200  @section('script')
201  <script>
202
203
204      function selectCheckbox(data){
205          var index = $(data).index(".selectgroup-itemInput");
206          console.log($(data));
207          var ini = $(".selectgroup-itemInput").eq(index+1);
208          console.log($(".ini"));
209          $(ini).prop("disabled", false);
210          $(ini).attr('style','background-color:red');
211      }
212
213
214      function finish(id){
215          swal({
216              title: "Apakah pesanan telah selesai?",
217              text: "",
218              type: "warning",
219              showCancelButton: true,
220              confirmButtonColor: "#F9C99B",
221              confirmButtonText: "Ya, Selesaikan",
222              cancelButtonText: "Tidak, batalkan!",
223              closeOnConfirm: false,
224              closeOnCancel: false
225
226
227          },
228          function (isConfirm) {
229              if (isConfirm) {
230                  swal("Pesanan Selesai!", "Pesanan telah diselesaikan", "success");
231                  $('#finish').attr('action', "{{route('transaksi.index')}}/" + id + "/finish");
232                  $('#finish').submit();
233              } else {
234                  swal("Batal", "Anda batal menyelesaikan pesanan", "error");
235              }
236          });
237      }

```

## 12. Halaman *index history*

```

1  @extends('layouts.penjual')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5  <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}>
6  @endsection
7
8  @section('content')
9  <div class="content-wrapper">
10 <section class="content">
11     <div class="container-fluid">
12         <div class="row">
13             <div class="col-12">
14                 <div class="card">
15                     <div class="card-header">
16                         <h3 class="card-title">History Transaksi Pembelian
17                         </h3>
18
19                     </div>
20                     <!-- /.card-header -->
21                     <div class="card-body">
22                         <table id="example1" class="table table-bordered table-striped dataTables-booking">
23                             <thead>
24                                 <tr>
25                                     <th>Tanggal</th>
26                                     <th>Umkm</th>
27                                     <th>Nama Pembeli</th>
28                                     <th>Total Pembelian</th>
29                                     <th>Action</th>
30                                 </tr>
31                             </thead>
32                             <tbody>
33                                 <tr>
34                                     <td></td>
35                                     <td></td>
36                                     <td></td>
37                                     <td></td>
38                                     <td></td>
39                                 </tr>
40                             </tbody>
41                         </table>
42                     <!-- /.card-body -->
43                 </div>
44                 <!-- /.card -->
45             </div>
46             <!-- /.col -->
47         </div>
48         <!-- /.row -->

```

```

41     </div>
42     <!-- /.container-fluid -->
43
44 </section>
45 </div>
46
47 @endsection
48
49 @section('script')
50 <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
51 <script>
52     $(document).ready(function () {
53         $('.dataTables-booking').DataTable({
54
55             responsive: true,
56             searchDelay: 500,
57             processing: true,
58             buttons: [
59                 {
60                     extend: 'print',
61                     className: 'btn-inverse',
62                     exportOptions: {
63                         columns: [0, 1]
64                     }
65                 },
66                 'copy',
67                 {
68                     extend: 'excel',
69                     className: 'btn-inverse',
70                     exportOptions: {
71                         columns: [0, 1]
72                     }
73                 },
74                 'csvHtml5',
75                 {
76                     extend: 'pdf',
77                     className: 'btn-inverse',
78                     exportOptions: {
79                         columns: [0, 1]
80                     }
81                 },
82             ],
83             // serverSide: true,
84             ajax: {
85                 url: "{{url('api/transaksi')}}?data=all&penjual_id={{Auth::user()->penjual->id}},
86                 dataSrc: ''
87             },
88             columns: [
89                 {
90                     data: 'date'
91                 },
92                 {
93                     data: 'umkm'
94                 },
95                 {
96                     data: 'name'
97                 },
98                 {
99                     data: 'total_harga'
100                },
101                {
102                    data: 'id',
103                    responsivePriority: -1
104                },
105            ],
106            columnDefs: [
107                {
108                    targets: -1,
109                    title: "Action",
110                    orderable: false,
111                    render: function (data, type, full, meta) {
112                        var base = "{{url('/')}}"
113                        return `` +
114                            `<a type="button" style="margin-left:5px;" class="btn btn-icon btn-sm btn-light" onclick="show(`+data+`)"
115                            href="/history/" + data + `/show`><i class="fa fa-info"></i> Detail</a>` +
116                            ``;
117                    },
118                },
119            ],
120        });
121        table.on('order.dt search.dt', function () {

```

```

120     table.column(0, {
121         search: 'applied',
122         order: 'applied'
123     }).nodes().each(function (cell, i) {
124         cell.innerHTML = i + 1;
125     });
126     ).draw();
127     $('#btn-print').on('click', function (e) {
128         e.preventDefault();
129         table.button(0).trigger();
130     });
131     $('#btn-copy').on('click', function (e) {
132         e.preventDefault();
133         table.button(1).trigger();
134     });
135     $('#btn-excel').on('click', function (e) {
136         e.preventDefault();
137         table.button(2).trigger();
138     });
139     $('#btn-csv').on('click', function (e) {
140         e.preventDefault();
141         table.button(3).trigger();
142     });
143     $('#btn-pdf').on('click', function (e) {
144         e.preventDefault();
145         table.button(4).trigger();
146     });
147 });
148 });
149
150
151
152
153 </script>
154 @endsection

```

### 13. Halaman *show history*

```

1 @extends('layouts.penjual')
2
3 @section('css')
4 <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5 <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}"/>
6 @endsection
7
8 @section('content')
9 <div class="content-wrapper">
10 <section class="content">
11     <div class="container-fluid">
12         <div class="row">
13             <div class="col-12">
14                 <div class="card">
15                     <div class="card-header">
16                         <h3 class="card-title">Detail History Transaksi Pemesanan
17                         </h3>
18                         @if ($pembelian->status_pembelian == 1)
19                             <span style="float:right; margin-left:5px;" class="badge badge-primary" >Pesanan Disediakan</span>
20                         @elseif ($pembelian->status_pembayaran == 0 && Carbon\Carbon::now() >= Carbon\Carbon::parse($pembelian->created_at)-1)
21                             <span style="float:right; margin-left:5px;" class="badge badge-danger" >Expired</span>
22                         @elseif ($pembelian->status_pembelian == 0 && $pembelian->status_pembayaran == 0)
23                             <span style="float:right; margin-left:5px;" class="badge badge-info" >Menunggu Pembayaran</span>
24                         @elseif ($pembelian->status_pembelian == 0)
25                             <span style="float:right; margin-left:5px;" class="badge badge-default" >Menunggu Konfirmasi</span>
26                         @elseif ($pembelian->status_pembelian == 2)
27                             <span style="float:right; margin-left:5px;" class="badge badge-primary" >Pesanan Dikirim</span>
28                         @elseif ($pembelian->status_pembelian == 3)
29                             <span style="float:right; margin-left:5px;" class="badge badge-danger" >Pesanan Ditolak</span>
30                         @elseif ($pembelian->status_pembelian == 4)
31                             <span style="float:right; margin-left:5px;" class="badge badge-info" >Selesai</span>
32                         @endif
33
34                     </div>
35                     <!-- /.card-header -->
36                     <div class="card-body">
37                         <div class="row">
38                             <div class="col-sm-4">
39                                 <!-- text input -->
40                                 <div class="form-group">

```

```

41 |         <label><i class="fas fa-user-alt" data-toggle="tooltip" data-title="Nama Pembeli"></i>
42 |         data-placement="bottom"></i> {{ $pembelian->nama_pembeli }}</label>
43 |     <br>
44 |     <label><i class="fas fa-calendar-alt" data-toggle="tooltip" data-title="Tanggal Transaksi"></i> {{ date('d-m-Y', strtotime($pembelian->created_at)) }}</label>
45 |     <br>
46 |     <br>
47 |     <br>
48 |     <br>
49 |     <br>
50 |     </div>
51 |     <div class="col-sm-4">
52 |         <!-- text input -->
53 |         <div class="form-group">
54 |             @if ($pembelian->pembayaran)
55 |                 <label><i class="far fa-credit-card" data-toggle="tooltip" data-title="Metode Pembayaran"></i>
56 |                     data-placement="bottom"></i> {{ ($pembelian->Pembayaran->Detail_Metode_Pembayaran->Metode_Pembayaran
57 |                         )->name }}</label>
58 |             @endif
59 |             <label><i class="fas fa-tasks" data-toggle="tooltip" data-title="Status Pembelian"></i>
60 |                 data-placement="bottom"></i>
61 |                 @if($pembelian->status_pembelian == 0)
62 |                     Menggu Konfirmasi
63 |                 @elseif($pembelian->status_pembelian == 1)
64 |                     Pesanan Disiapkan
65 |                 @elseif($pembelian->status_pembelian == 2)
66 |                     Pesanan Dikirir
67 |                 @elseif ($pembelian->status_pembelian == 3)
68 |                     Pesanan Ditolak
69 |                 @elseif ($pembelian->status_pembelian == 4)
70 |                     Transaksi Selesai
71 |                 @endif
72 |             </label>
73 |         @endif
74 |
75 |
76 |
77 |     </div>
78 |     <div class="col-sm-4">
79 |
80 |         @if ($pembelian->pembayaran)
81 |             <div class="form-group">
82 |                 <label>Bukti Pembayaran </label><br>
83 |                 
85 |             </div>
86 |         @endif
87 |     </div>
88 |     <div class="row">
89 |         <div class="col-sm-12">
90 |             <!-- select -->
91 |             <h2>Kode Pemesanan : {{ $pembelian->token }}</h2>
92 |             <div class="form-group">
93 |                 <label>Detail Pemesanan </label>
94 |                 <div class="form-group">
95 |                     <table class="table table-striped">
96 |                         <thead>
97 |                             <tr>
98 |                                 <th style="width: 10px">No</th>
99 |                                 <th>Nama Produk</th>
100 |                                 <th>Kemasan</th>
101 |                                 <th>Jumlah</th>
102 |                                 <th>Harga Produk</th>
103 |                                 <th>Subtotal</th>
104 |                             </tr>
105 |                         </thead>
106 |                         <tbody>
107 |                         @php
108 |                         // Inisialisasi variabel
109 |                         $index = 1;
110 |                         $total_harga = 0;
111 |                         $subtotal_produk = 0;
112 | @endphp
113 | @foreach($pembelian->Detail_Pembelian as $item)
114 |     <tr>
115 |         <td>{{ $index }}</td>
116 |         <td>{{ $item->produk->nama }}</td>
117 |         <td>{{ $item->kemasan->kuran }}</td>
118 |         <td>{{ $item->jumlah }}</td>
119 |
120 |         <td>Rp {{ number_format($item->Produk->Detail_Produk->here('kemasan_id', $item->kemasan_id)->first()->harga, 0, ',', '.') }}</td>
121 |         <td>Rp {{ number_format($item->Produk->Detail_Produk->here('kemasan_id', $item->kemasan_id)->first()->harga, 0,
122 |                         ',.') }}</td>
123 |     @endphp
124 |     // Perhitungan total harga dan subtotal produk
125 |     $total_harga += $item->jumlah * $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
126 |     $subtotal_produk += $item->jumlah * $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
127 | @endforeach
128 |     </tbody>
129 |     <tfoot>
130 |         <tr>
131 |             <td colspan="5" style="text-align: right;">Subtotal Produk:</td>
132 |             <td>Rp {{ number_format($subtotal_produk, 0, ',', '.') }}</td>
133 |         </tr>
134 |         <tr>
135 |             <td colspan="5" style="text-align: right;">Biaya Pengiriman:</td>
136 |             <td>Rp {{ number_format($pembelian->total_harga - $subtotal_produk, 0, ',', '.') }}</td>
137 |         </tr>
138 |         <tr>
139 |             <td colspan="5" style="text-align: right;">Total Harga:</td>
140 |             <td>Rp {{ number_format($pembelian->total_harga, 0, ',', '.') }}</td>
141 |         </tr>
142 |     </tfoot>
143 |     <table>
144 |         <tr>
145 |             <td></td>
146 |         </tr>
147 |     </table>
148 |     </div>
149 |     </div>
150 |     <!-- /.card-body -->
151 |     <!-- /.card -->
152 |     <!-- /.col -->
153 |     </div>
154 |     <!-- /.row -->
155 | 
```

```

158     <!-- /.container-fluid -->
159
160 </section>
161 </div>
162
163
164 @endsection
165
166

```

## 14. Halaman *index report*

```

1  @extends('layouts.penjual')
2
3  @section('css')
4
5  <!-- daterangepicker -->
6  <link rel="stylesheet" href="{{ asset('templatesAdminLTE/plugins/daterangepicker/daterangepicker.css') }}"/>
7  <!-- Select2 -->
8  <link rel="stylesheet" href="{{ asset('templatesAdminLTE/plugins/select2/css/select2.min.css') }}"
9   href="{{ asset('templatesAdminLTE/plugins/select2-bootstrap4-theme/select2-bootstrap4.min.css') }}"/>
10
11
12 @endsection
13
14 @section('content')
15 <div class="content-wrapper">
16 <section class="content">
17   <div class="container-fluid">
18     <div class="row">
19       <div class="col-12">
20         <!-- /.card -->
21         {{ Form::open(array('method'=>'POST', 'url' => route('report.preview')) )}}
22         <div class="card">
23           <div class="card-header">
24             <h3 class="card-title">Cetak Laporan Transaksi
25             </h3>
26           </div>
27           <div class="card-body">
28             <div class="form-group">
29               <label>Laporan transaksi pada umkm :</label>
30               <select class="select2" name="umkm[]" multiple="multiple" data-placeholder="Select a State"
31                 style="width: 100%; required">
32                 @foreach ($umkms as $umkm)
33                   @if ($umkm->status == 1)
34                     <option value="{{ $umkm->id }}>{{ $umkm->name }}</option>
35                   @endif
36                 @endforeach
37               </select>
38             </div>
39             <div class="form-group">
40               <label>Laporan transaksi pada tanggal :</label>
41
42               <div class="input-group">
43                 <div class="input-group-prepend">
44                   <span class="input-group-text">
45                     | <i class="far fa-calendar-alt"></i>
46                   </span>
47                 </div>
48                 <input type="text" name="date_range" class="form-control float-right" id="reservation" required>
49               <!-- /.input group -->
50             </div>
51             <!-- <button type="submit" class="btn btn-primary">i class="fas fa-eye" style='font-size:12px'>/i View</button> -->
52             <button type="submit" class="btn btn-primary">i class="fas fa-download" style='font-size:12px'>/i View</button>
53
54           </div>
55           {!! Form::close() !!}
56           <!-- /.card-body -->
57         </div>
58       <!-- /.col -->
59     </div>
60     <!-- /.row -->
61   </div>
62   <!-- /.container-fluid -->
63
64 </section>
65 </div>
66 @endsection
67
68 @section('script')
69 <!-- date-range-picker -->
70 <script src="{{asset('templatesAdminLTE/plugins/moment/moment.min.js')}}"></script>
71 <script src="{{ asset('templatesAdminLTE/plugins/daterangepicker/daterangepicker.js') }}"/></script>
72 <!-- Bootstrap 4 -->
73 <script src="{{ asset('templatesAdminLTE/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"/></script>
74 <!-- Select2 -->
75 <script src="{{ asset('templatesAdminLTE/plugins/select2/js/select2.full.min.js') }}"/></script>
76 <script>
77   $(function () {
78     //Initialize Select2 Elements
79
80     $('.select2').select2()
81
82     //Initialize Select2 Elements
83     $('.select2bs4').select2({
84       theme: 'bootstrap4'
85     })
86     //Date range picker
87     $('#reservation').daterangepicker()
88   })
89 </script>
90 @endsection

```

## 15. Halaman *view\_print report*

```
1  @extends('layouts.penjual')
2
3  @section('css')
4
5  <!-- daterangepicker -->
6  <link rel="stylesheet" href="{{ asset('templatesAdminLTE/plugins/daterangepicker/daterangepicker.css') }}">
7  <!-- Select2 -->
8  <link rel="stylesheet" href="{{ asset('templatesAdminLTE/plugins/select2/css/select2.min.css') }}>
9  <link rel="stylesheet"
10   href="{{ asset('templatesAdminLTE/plugins/select2-bootstrap4-theme/select2-bootstrap4.min.css') }}>
11 <style>
12   @media print {
13     @page {
14       size: landscape
15     }
16   }
17   @page {
18     size: A4 landscape;
19   }
20
21 table {
22   border-collapse: collapse;
24   width: 100%;
25 }
26
27 .table th {
28   padding: 8px 8px;
29   border: 1px solid #000000;
30   font-weight: bold;
31   text-align: center;
32   color: rgb(3, 10, 103);
33   background-color: rgb(255, 252, 252);
34 }
35
36 .table td {
37   padding: 3px 3px;
38   border: 1px solid #000000;
39   text-align: center;
40   background-color: white;
41 }
42
43 .text-center {
44   text-align: center;
45 }
46 </style>
47 <style type="text/css" media="print">
48   .page {
49     -webkit-transform: rotate(-90deg);
50     -moz-transform: rotate(-90deg);
51     filter: progid:DXImageTransform.Microsoft.BasicImage(rotation=3);
52   }
53 </style>
54
55 @endsection
56
57 @section('content')
58 <div class="content-wrapper">
59 <section class="content">
60   <div class="container-fluid">
61     {{ Form::open(array('method'=>'POST', 'url' => route('report.print')) )}}
62     <div class="card">
63       <div class="card-header">
64         <h3 class="card-title">Cetak Laporan Transaksi
65         </h3>
66       </div>
67       <div class="card-body">
68         @foreach ($umkms as $umkm)
69           <div class="row">
70             <div class="col-12">
71               <p><strong>Nama Umkm : {{ $umkm->nama }}</strong></p>
72               <p><strong>Transaksi pada : {{ $date[0] }} - {{ $date[1] }}</strong></p>
73             </div>
74           </div>
75         @endphp
76         $produk = App\Models\Produk::where('umkm_id', $umkm->id)->get();
77         $produk_ids = $produk->pluck('id'); // Ambil ID produk yang terkait dengan umkm
78         $pembelian = App\Models\Pembelian::with(['Detail_Pembelian.Produk', 'Detail_Pembelian.Kemasan'])
79         ->WhereHas('Detail_Pembelian.Produk', function ($query) use ($produk_ids) {
```

```

80     $query->whereIn('produk_id', $produk_ids);
81
82     //>where(Db::raw("date_format(created_at, '%m/%d/%Y')", '>=', $date[0])
83     //>where(Db::raw("date_format(created_at, '%m/%d/%Y')", '<=', $date[1])
84     //>where('status_pembelian', 4)
85     //>get();
86   >>
87   <table class="table">
88     <thead>
89       <tr>
90         <th>No</th>
91         <th>Tanggal pembelian</th>
92         <th>Pembeli</th>
93         <th>Produk</th>
94         <th>Kemasan</th>
95         <th>Jumlah</th>
96         <th>Harga</th>
97         <th>Subtotal</th>
98       </tr>
99     </thead>
100    <tbody>
101      @foreach ($pembelians as $pembelian)
102        @php
103          // Reset subtotal_produk untuk setiap pembelian
104          $subtotal_produk = 0;
105        @endphp
106        <tr>
107          <td>{{ $loop->iteration }}</td>
108          <td>{{ $pembelian->created_at->format('d/m/Y') }}</td>
109          <td>{{ $pembelian->nama_pembeli }}</td>
110          <td>
111            @foreach ($pembelian->Detail_Pembelian as $detail)
112              {{ $detail->Produk->nama }} <br>
113            @endforeach
114          </td>
115          <td>
116            @foreach ($pembelian->Detail_Pembelian as $detail)
117              {{ $detail->Kemasan->ukuran }} <br>
118            @endforeach
119          </td>
120          <td>
121            @foreach ($pembelian->Detail_Pembelian as $detail)
122              {{ $detail->jumlah }} <br>
123            @endforeach
124          </td>
125          <td>
126            @foreach ($pembelian->Detail_Pembelian as $detail)
127              Rp {{ number_format($detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga, 0, ',', '.') }} <br>
128            @endforeach
129          </td>
130          <td>
131            @foreach ($pembelian->Detail_Pembelian as $detail)
132              Rp {{ number_format($detail->jumlah * $detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga, 0, ',', '.') }} <br>
133            @endphp
134            // Perhitungan subtotal produk untuk setiap detail
135            $subtotal_produk += $detail->jumlah * $detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga;
136            @endphp
137            @endforeach
138          </td>
139        </tr>
140
141        <tr>
142          <td colspan="7" style="text-align: right;"><strong>Biaya Pengiriman:</strong></td>
143          <td><strong>Rp {{ number_format($pembelian->total_harga - $subtotal_produk, 0, ',', '.') }}</strong></td>
144        </tr>
145
146        <tr>
147          <td colspan="7" style="text-align: right;"><strong>Total Harga</strong></td>
148          <td><strong>Rp {{ number_format($pembelian->total_harga, 0, ',', '.') }}</strong></td>
149        </tr>
150
151        <tr>
152          <th colspan="7" style="text-align: center;">Total Pemasukan</th>
153          <th>{{ Helper::rupiah($pembelians->sum('total_harga')) }}</th>
154        </tr>
155      </tbody>
156    </table>
157  </endforeach>
158
159  <div class="form-group" style="display:none">
160    <label>Laporan transaksi pada umkm :</label>
161   {!! Form::select('umkm[]', $umkm_select, $umkm_ids, ['title'=>'Nothing Selected',
162      'class' =>
163      'form-control',
164      'select2',
165      'data-live-search'=>'true', 'required'=>'required', 'id'=>'s_day',
166      'multiple' =>
167      'multiple', 'data-placeholder' => 'Select a State']) !!}
168  </div>
169  <div class="form-group" style="display:none">
170    <label>Laporan transaksi pada tanggal :</label>
171    <div class="input-group">
172      <div class="input-group-prepend">
173        <span class="input-group-text">
174          <i class="far fa-calendar-alt"></i>
175        </span>
176      </div>
177      <input type="text" name="date_range" value="{{ $date_range }}" class="form-control float-right" id="reservation" required>
178      </div>
179      <!-- /input group -->
180    </div>
181    <br>
182    <button type="submit" class="btn btn-primary"><i class="fas fa-download" style='font-size:12px'></i>
183    | Cetak</button>
184  </div>
185  {!! Form::close() !!}
186 </div>
187 <!-- /.row -->
188 </div>
189 <!-- /.container-fluid -->
190
191 </section>
192 </div>
193 @endsection

```

```

196 @section('script')
197 <!-- date-range-picker -->
198 <script src="{{ asset('templatesAdminLTE/plugins/daterangepicker/daterangepicker.js') }}"></script>
199 <!-- Bootstrap 4 -->
200 <script src="{{ asset('templatesAdminLTE/plugins/bootstrap/js/bootstrap.bundle.min.js') }}"></script>
201 <!-- Select2 -->
202 <script src="{{ asset('templatesAdminLTE/plugins/select2/js/select2.full.min.js') }}"></script>
203 <script>
204     $(function () {
205         //Initialize Select2 Elements
206         $('.select2').select2()
207
208         //Initialize Select2 Elements
209         $('.select2bs4').select2({
210             theme: 'bootstrap4'
211         })
212         //Date range picker
213         $('#reservation').daterangepicker()
214     })
215 </script>
216 @endsection

```

## 16. Halaman *print report*

```

1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="utf-8">
6      <title>Laporan Transaksi Penjualan Produk</title>
7      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/paper-css/0.4.1/paper.css">
8      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/paper-css/0.4.1/paper.css">
9      <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
|       integrity="sha384-Gn5z84+Qlae0XAA+050RXPxPg6fy4IwvTh0E263XmfCjISAw1GfAW/dA1S6JXm" crossorigin="anonymous">
10     <style>
11         @media print {
12             @page {
13                 size: landscape
14             }
15         }
16
17         @page {
18             size: A4 landscape;
19         }
20
21
22         h1 {
23             font-weight: bold;
24             font-size: 20pt;
25             text-align: center;
26         }
27
28         table {
29             border-collapse: collapse;
30             width: 100%;
31         }
32
33         .table th {
34             padding: 8px 8px;
35             border: 1px solid #000000;
36             font-weight: bold;
37             text-align: center;
38             color: rgb(3, 10, 103);
39             background-color: rgb(255, 252, 252);
40
41         }
42
43         .table td {
44             padding: 3px 3px;
45             border: 1px solid #000000;
46             text-align: center;
47             background-color: white;
48         }
49
50         .text-center {
51             text-align: center;
52         }
53     </style>
54     <style type="text/css" media="print">
55         .page {
56             -webkit-transform: rotate(-90deg);
57             -moz-transform: rotate(-90deg);
58             filter: progid:DXImageTransform.Microsoft.BasicImage(rotation=3);
59         }
60     </style>
61 </head>
62
63 <body>
64     <section class="sheet padding-10mm">
65         <div class="row">
66             <div class="col-12">
67             </div>
68             <div class="col-12">
69                 <center>
70                     <h4><b>Laporan Transaksi Penjualan Produk</b></h4>
71                 </center>
72             </div>
73             <br>
74             <br>
75             <hr>
76             @foreach ($umkms as $umkm)
77                 <div class="row">
78                     <div class="col-12">
79                         <p><strong>Nama Umkm : {{ $umkm->nama }}</strong></p>

```

```

80           <p><strong>Transaksi pada : {{ $date[0] }} - {{ $date[1] }}</strong></p>
81       </div>
82   </div>
83   <?php
84       $produk = App\Models\Produk::where('umkm_id', $umkm->id)->get();
85       $produk_ids = $produk->pluck('id'); // Ambil ID produk yang terkait dengan umkm
86       $pembelians = App\Models\Pembelian::with(['Detail_Pembelian.Produk', 'Detail_Pembelian.Kemasan'])
87           ->whereHas('Detail_Pembelian.Produk', function ($query) use ($produk_ids) {
88               $query->whereIn('produk_id', $produk_ids);
89           })
90           ->where(DB::raw('date_format(created_at, "%m/%d/%Y")'), '>', $date[0])
91           ->where(DB::raw('date_format(created_at, "%m/%d/%Y")'), '<', $date[1])
92           ->where('status_pembelian', 4)
93           ->get();
94   ?>
95   <table class="table">
96       <thead>
97           <tr>
98               <th>No</th>
99               <th>Tanggal pembelian</th>
100              <th>Pembeli</th>
101              <th>Produk</th>
102              <th>Kemasan</th>
103              <th>Jumlah</th>
104              <th>Harga</th>
105              <th>Subtotal</th>
106          </tr>
107      </thead>
108      <tbody>
109          @foreach ($pembelians as $pembelian)
110          @php
111              // Reset subtotal_produk untuk setiap pembelian
112              $subtotal_produk = 0;
113          @endphp
114          <tr>
115              <td>{{ $loop->iteration }}</td>
116              <td>{{ $pembelian->created_at->format('d/m/Y') }}</td>
117              <td>{{ $pembelian->nama_pembeli }}</td>
118              <td>
119                  @foreach ($pembelian->Detail_Pembelian as $detail)
120
121                      {{ $detail->Produk->nama }} <br>
122                  @endforeach
123              </td>
124              <td>
125                  @foreach ($pembelian->Detail_Pembelian as $detail)
126                      {{ $detail->kemasan->ukuran }} <br>
127                  @endforeach
128              </td>
129              <td>
130                  @foreach ($pembelian->Detail_Pembelian as $detail)
131                      {{ $detail->jumlah }} <br>
132                  @endforeach
133              </td>
134              <td>
135                  @foreach ($pembelian->Detail_Pembelian as $detail)
136                      Rp {{ number_format($detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga, 0, ',', '.') }} <br>
137                  @endforeach
138              </td>
139              <td>
140                  @foreach ($pembelian->Detail_Pembelian as $detail)
141                      Rp {{ number_format($detail->jumlah * $detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga, 0, ',', '.') }} <br>
142                  @php
143                      // Perhitungan subtotal produk untuk setiap detail
144                      $subtotal_produk += $detail->jumlah * $detail->Produk->Detail_Produk->where('kemasan_id', $detail->kemasan_id)->first()->harga;
145                  @endphp
146              </td>
147          </tr>
148
149          <tr>
150              <td colspan="7" style="text-align: right;"><strong>Biaya Pengiriman:</strong></td>
151              <td><strong>Rp {{ number_format($pembelian->total_harga - $subtotal_produk, 0, ',', '.') }}</strong></td>
152          </tr>
153
154          <tr>
155              <td colspan="7" style="text-align: right;"><strong>Total Harga:</strong></td>
156              <td><strong>Rp {{ number_format($pembelian->total_harga, 0, ',', '.') }}</strong></td>
157          </tr>
158      @endforeach
159
160      <tr>
161          <th colspan="7" style="text-align: center;">Total Pemasukan

```

## 17. Halaman profil penjual

```
1 @extends('layouts.penjual')
2 @section('content')
3 <div class="content-wrapper">
4     <section class="section main-section">
5         <div class="grid grid-cols-1 gap-6 lg:grid-cols-2 mb-6">
6             <div class="card">
7                 <header class="card-header">
8                     <p class="card-header-title">
9                         <span class="icon"><i class="mdi mdi-account"></i></span>
10                        Profile
11                     </p>
12                 </header>
13                 <div class="card-content">
14                     <div class="image w-28 h-28 mx-auto">
15                         
17                     <hr>
18                     <@csrf>
19                     <div class="form-group">
20                         <label for="exampleInputName1">Name</label>
21                         <input name="name" type="text" class="form-control" id="name" placeholder="Name" value="@php echo $penjual->firstname . ' ' . $penjual->lastname @endphp" readonly>
22                     </div>
23                     <div class="form-group">
24                         <label for="exampleInputEmail13">No Handphone</label>
25                         <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $penjual->handphone }}" readonly>
26                     </div>
27                     <div class="form-group">
28                         <label for="Provinsi">Provinsi</label>
29                         <select class="form-control" name="provinsi" id="provinsi" disabled>
30                             <option value="">Provinsi</option>
31                             @foreach ($provinsi as $provinsi => $value)
32                                 <option value="{{ $provinsi }}"{!! $penjual->province_id == $provinsi ? 'selected' : '' !!}>{{ $value }}</option>
33                             @endforeach
34                         </select>
35                         <div class="text-danger"></div>
36                     </div>
37                     <div class="form-group">
38                         <label for="Kota">Kota</label>
39                         <select class="form-control" name="kota" id="kota" disabled>
40                             <option value="">Kota</option>
41                             @foreach ($kota as $kota => $value)
42                                 <option value="{{ $kota }}"{!! $penjual->city_id == $kota ? 'selected' : '' !!}>{{ $value }}</option>
43                             @endforeach
44                         </select>
45                         <div class="text-danger"></div>
46                     </div>
47                     <div class="form-group">
48                         <label for="exampleTextareal">Alamat</label>
49                         <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $penjual->alamat }}" readonly>
50                     </div>
51                     <div class="form-group">
52                         <label for="exampleInputEmail13">Email</label>
53                         <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}" readonly>
54                     <div class="text-danger">
55                         @error('email')
56                             {{ $message }}
57                         @enderror
58                     </div>
59                     </div>
60                     <hr>
61                     <div class="image w-48 h-48 mx-auto">
62                         <label for="exampleInputEmail13">XTP</label>
63                         
65                     </div>
66                     <div class="card">
67                         <header class="card-header">
68                             <p class="card-header-title">
69                                 <span class="icon"><i class="mdi mdi-account-circle"></i></span>
70                                 Edit Profil
71                             </p>
72                         </header>
73                     </div>
74                 </div>
75             </div>
76         </div>
77     </div>
78 </div>
```

```

80      <div class="card-content">
81          <form method="POST" action="{{ route('profilpenjual.update') }}" enctype="multipart/form-data">
82              @csrf
83              <div class="form-group">
84                  <label>Foto Profil</label>
85                  <div></div>
86                  <input type="file" name="img" class="form-control" value="{{ $penjual->img }}>
87                  <!-- <div>
88                      <a href="{{ asset('foto_profil/'. $penjual->img )}}" target="_blank" rel="noopener noreferrer"{{ ($penjual->img) }}> </a> </div> -->
89                  <div class="text-danger">
90                      @error('img')
91                          {{ $message }}
92                      @enderror
93                  </div>
94              </div>
95              <div class="form-group">
96                  <label>KTP</label>
97                  <div></div>
98                  <input type="file" name="ktp" class="form-control" value="{{ $penjual->ktp }}>
99                  <!-- <div>
100                      <a href="{{ asset('foto_profil/'. $penjual->img )}}" target="_blank" rel="noopener noreferrer"{{ ($penjual->img) }}> </a> </div> -->
101                  <div class="text-danger">
102                      @error('ktp')
103                          {{ $message }}
104                      @enderror
105                  </div>
106              </div>
107              <hr>
108              <div class="form-group">
109                  <label for="exampleInputName1">Firstname</label>
110                  <input name="firstname" type="text" class="form-control" id="firstname" placeholder="Firstname" value="{{ $penjual->firstname }}>
111                  <div class="text-danger">
112                      @error('firstname')
113                          {{ $message }}
114                      @enderror
115                  </div>
116              </div>
117              <div class="form-group">
118                  <label for="exampleInputEmail13">Lastname</label>
119                  <input name="lastname" type="text" class="form-control" id="lastname" placeholder="Lastname" value="{{ $penjual->lastname }}>
120
121                  </div>
122                  <div class="form-group">
123                      <label for="exampleInputEmail3">No Handphone</label>
124                      <input name="handphone" type="text" class="form-control" id="handphone" placeholder="No Handphone" value="{{ $penjual->handphone }}>
125                      <div class="text-danger">
126                          @error('handphone')
127                              {{ $message }}
128                          @enderror
129                      </div>
130                  </div>
131                  <div class="form-group">
132                      <label for="">Provinsi</label>
133                      <select class="form-control" name="provinsi" id="provinsi">
134                          <option value="">Provinsi</option>
135                          @foreach ($provinsiss as $provinsi => $value)
136                              <option value="{{ $provinsi }}> {{ $penjual->province_id == $provinsi ? 'selected' : '' }}> {{ $value }}</option>
137                          @endforeach
138                      </select>
139                      <div class="text-danger"></div>
140                  </div>
141                  <div class="form-group">
142                      <label for="">Kota</label>
143                      <select class="form-control" name="kota" id="kota">
144                          <option value="">Kota</option>
145                          @foreach ($kotass as $kota => $value)
146                              <option value="{{ $kota }}> {{ $penjual->city_id == $kota ? 'selected' : '' }}> {{ $value }}</option>
147                          @endforeach
148                      </select>
149                      <div class="text-danger"></div>
150                  </div>
151                  <div class="form-group">
152                      <label for="exampleTextareal1">Alamat</label>
153                      <input name="alamat" type="text" class="form-control" id="alamat" placeholder="Alamat" value="{{ $penjual->alamat }}>
154                      <div class="text-danger">
155                          @error('alamat')
156                          {{ $message }}
157                          @enderror
158                  </div>

```

```

159 </div>
160 </div>
161 <div class="form-group">
162   <label for="exampleInputEmail3">Email</label>
163   <input name="email" type="email" class="form-control" id="email" placeholder="Email" value="{{ $user->email }}>
164   <div class="text-danger">
165     @error('email')
166       {{ $message }}
167     @enderror
168   </div>
169 </div>
170 <hr> <br>
171 <div class="form-group">
172   <div class="control">
173     <button type="submit" class="btn btn-primary mr-2">Submit</button>
174   </div>
175 </div>
176 </form>
177 </div>
178 </div>
179 <div class="card">
180   <header class="card-header">
181     <p class="card-header-title">
182       <span class="icon"><i class="mdi mdi-lock"></i></span>
183       Change Password
184     </p>
185   </header>
186   <div class="card-content">
187     <form action="{{ route('change.password') }}" method="POST">
188       @csrf
189
190       <div class="form-group">
191         <label for="current_password">Current Password</label>
192         <input name="current_password" type="password" class="form-control" id="current_password" placeholder="Current Password">
193       </div>
194
195       <div class="form-group">
196         <label for="new_password">New Password</label>
197
198         <label for="new_password">New Password</label>
199         <input name="new_password" type="password" class="form-control" id="new_password" placeholder="New Password">
200       </div>
201
202       <div class="form-group">
203         <label for="confirm_password">Confirm Password</label>
204         <input name="confirm_password" type="password" class="form-control" id="confirm_password" placeholder="Confirm Password">
205       </div>
206
207       <div class="form-group">
208         <div class="control">
209           <button type="submit" class="btn btn-primary mr-2">Submit</button>
210         </div>
211       </div>
212     </form>
213   </div>
214 </div>
215 </div>
216 @endsection
217 <!-- Include the jQuery library -->
218 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
219 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
220
221 <script>
222 $(document).ready(function() {
223   $('select[name="provinsi"]').on('change', function() {
224     let provinceId = $(this).val();
225     if (provinceId) {
226       jQuery.ajax({
227         url: '/provincie/' + provinceId + '/cities',
228         type: "GET",
229         dataType: "json",
230         success: function(data) {
231           $('select[name="kota"]').empty();
232           $.each(data, function(key, value) {
233             $('select[name="kota"]').append('<option value="' + key + '">' + value + '</option>');
234           });
235         },
236         error: function(xhr, status, error) {
237           | console.error(error);
238         }
239       });
240     } else {
241       $('select[name="kota"]').empty();
242     }
243   });
244 });
245 </script>
246

```

## 18. Halaman *dashboard* penjual

```
1  @extends('layouts.penjual')
2  @section('content')
3  <!-- partial -->
4  <!-- <div class="main-panel"> -->
5  |   <div class="content-wrapper">
6  |     <div class="row">
7  |       <div class="col-md-12 grid-margin">
8  |         <div class="row">
9  |           <div class="col-12 col-xl-8 mb-4 mb-xl-0">
10 |             <h3 class="font-weight-bold">Welcome {{Auth::user()->name}}
11 |             <h6 class="font-weight-normal mb-0">All systems are running smoothly!
12 |           </div>
13 |
14 |         </div>
15 |       </div>
16 |     </div>
17 |     <div class="row">
18 |       <div class="col-md-6 grid-margin stretch-card">
19 |         <div class="card tale-bg">
20 |           <div class="card-body">
21 |             
22 |             <div class="weather-info">
23 |               <div class="d-flex">
24 |
25 |               </div>
26 |             </div>
27 |           </div>
28 |         </div>
29 |       </div>
30 |       <div class="col-md-6 grid-margin transparent">
31 |         <div class="row">
32 |           <div class="col-md-6 mb-4 stretch-card transparent">
33 |             <div class="card card-tale">
34 |               <div class="card-body">
35 |                 <?php
36 |                 $user_id = Auth::id();
37 |
38 |                 // Ambil penjual_id dari tabel penjual berdasarkan users_id
39 |                 $penjual_id = DB::table('penjual')
40 |                   ->where('users_id', $user_id)
41 |                   ->value('id');
42 |
43 |                 // Hitung jumlah UMKM berdasarkan penjual_id dan status operasi
44 |                 $jumlahUmkm = DB::table('umkm')
45 |                   ->where('penjual_id', $penjual_id)
46 |                   ->where('status', 1)
47 |                   ->count();
48 |
49 |                 // Anda dapat menambahkan kueri lain untuk menghitung jumlah pembelian sesuai statusnya
50 |
51 |                 // Misalnya:
52 |                 $jumlahPembelianMenungguKonfirmasi = DB::table('pembelian')
53 |                   ->join('detailpemb', 'pembelian.id', '=', 'detailpemb.pembelian_id')
54 |                   ->join('produk', 'detailpemb.produk_id', '=', 'produk.id')
55 |                   ->join('umkm', 'produk.umkm_id', '=', 'umkm.id')
56 |                   ->select('pembelian.*')
57 |                   ->where('umkm.penjual_id', $penjual_id)
58 |                     ->where('status_pembayaran', 1)
59 |                     ->where('status_pembelian', 0)
60 |                     ->count();
61 |
62 |
63 |
64 |                 $pesanAndaSiapKirim = DB::table('pembelian')
65 |                   ->join('detailpemb', 'pembelian.id', '=', 'detailpemb.pembelian_id')
66 |                   ->join('produk', 'detailpemb.produk_id', '=', 'produk.id')
67 |                   ->join('umkm', 'produk.umkm_id', '=', 'umkm.id')
68 |                   ->select('pembelian.*')
69 |                   ->where('umkm.penjual_id', $penjual_id)
70 |                     ->where('status_pembelian', 1)
71 |                     ->count();
72 |                 $pesanAndaDikirim = DB::table('pembelian')
73 |                   ->join('detailpemb', 'pembelian.id', '=', 'detailpemb.pembelian_id')
74 |                   ->join('produk', 'detailpemb.produk_id', '=', 'produk.id')
75 |                   ->join('umkm', 'produk.umkm_id', '=', 'umkm.id')
76 |                   ->select('pembelian.*')
77 |                   ->where('umkm.penjual_id', $penjual_id)
78 |                     ->where('status_pembelian', 2)
79 |                     ->count();
80 |               
```

```

80           $pesananditolak = DB::table('pembelian')
81             ->join('detailpemb', 'pembelian.id', '=', 'detailpemb.pembelian_id')
82             ->join('produk', 'detailpemb.produk_id', '=', 'produk.id')
83             ->join('umkm', 'produk.umkm_id', '=', 'umkm.id')
84             ->select('pembelian.*')
85             ->where('umkm.penjual_id', $penjual_id)
86             ->where('status_pembelian', 3)
87             ->count();
88           $pesananslesai = DB::table('pembelian')
89             ->join('detailpemb', 'pembelian.id', '=', 'detailpemb.pembelian_id')
90             ->join('produk', 'detailpemb.produk_id', '=', 'produk.id')
91             ->join('umkm', 'produk.umkm_id', '=', 'umkm.id')
92             ->select('pembelian.*')
93             ->where('umkm.penjual_id', $penjual_id)
94             ->where('status_pembelian', 4)
95             ->count();
96         ?>
97         <p class="mb-4">Jumlah UMKM Beroperasi</p>
98         <p class="fs-30 mb-2"><?php echo $jumlahUmkm; ?></p>
99       
```

---

```

100      </div>
101    </div>
102  </div>
103  <div class="col-md-6 mb-4 stretch-card transparent">
104    <div class="card card-dark-blue">
105      <div class="card-body">
106        <p class="mb-4">Menunggu Konfirmasi</p>
107        <p class="fs-30 mb-2"><?php echo $jumlahPembelianMenungguKonfirmasi; ?></p>
108      </div>
109    </div>
110  </div>
111  </div>
112  </div>
113  </div>
114  <div class="row">
115    <div class="col-md-6 mb-4 stretch-card transparent">
116      <div class="card card-light-blue">
117        <div class="card-body">
118          <p class="mb-4">Pesanan Disiapkan</p>
119          <p class="fs-30 mb-2"><?php echo $pesanansiapkan; ?></p>
120        </div>
121      </div>
122    </div>
123  </div>
124  <div class="col-md-6 mb-4 stretch-card transparent">
125    <div class="card card-dark-blue">
126      <div class="card-body">
127        <p class="mb-4">Pesanan Dikirim</p>
128        <p class="fs-30 mb-2"><?php echo $pesanandikirim; ?></p>
129      </div>
130    </div>
131  </div>
132  </div>
133  </div>
134  <div class="row">
135    <div class="col-md-6 mb-4 mb-lg-0 stretch-card transparent">
136      <div class="card card-light-blue">
137        <div class="card-body">
138          <p class="mb-4">Pesanan Selesai</p>
139          <p class="fs-30 mb-2"><?php echo $pesananslesai; ?></p>
140        </div>
141      </div>
142    </div>
143  </div>
144  <div class="col-md-6 stretch-card transparent">
145    <div class="card card-light-danger">
146      <div class="card-body">
147        <p class="mb-4">Pesanan Ditolak</p>
148        <p class="fs-30 mb-2"><?php echo $pesananditolak; ?></p>
149      </div>
150    </div>
151  </div>
152  </div>
153  </div>
154  </div>
155  </div>
156  </div>
157 @endsection

```

## e. Bagian Landing

### 1. Halaman landing page

```
1 <!DOCTYPE html>
2 <html lang="{{ str_replace('_', '-', app()->getLocale()) }}>
3   <head>
4     <meta charset="utf-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1">
6     <title>Marketplace UNIKM Sanjai</title>
7     <!-- Google Font -->
8     <link href="https://fonts.googleapis.com/css2?family=Cairo:wght@200;300;400;600;900&display=swap" rel="stylesheet">
9
10    <!-- Css Styles -->
11    <link rel="stylesheet" href="{{ asset('Frontend/css/bootstrap.min.css') }}" type="text/css">
12    <link rel="stylesheet" href="{{ asset('Frontend/css/font-awesome.min.css') }}" type="text/css">
13    <link rel="stylesheet" href="{{ asset('Frontend/css/elegant-icons.css') }}" type="text/css">
14    <link rel="stylesheet" href="{{ asset('Frontend/css/nice-select.css') }}" type="text/css">
15    <link rel="stylesheet" href="{{ asset('Frontend/css/jquery-ui.min.css') }}" type="text/css">
16    <link rel="stylesheet" href="{{ asset('Frontend/css/owl.carousel.min.css') }}" type="text/css">
17    <link rel="stylesheet" href="{{ asset('Frontend/css/slicknav.min.css') }}" type="text/css">
18    <link rel="stylesheet" href="{{ asset('Frontend/css/style.css') }}" type="text/css">
19
20  </head>
21  <body class="antialiased">
22    | <!-- Page Preloader -->
23    | <div id="preloader">
24    |   <div class="loader"></div>
25    | </div>
26
27
28  <!-- Header Section Begin -->
29  <header class="header">
30    | <div class="header__top">
31    |   <div class="container">
32    |     <div class="row">
33    |       <div class="col-lg-6 col-md-6">
34    |       </div>
35    |       <div class="col-lg-6 col-md-6">
36    |         <div class="header__top_right">
37    |           &if(Auth::guest())
38    |             <div class="header__top_right_auth">
39    |               <a href="{{ route('register') }}><i class="fa fa-user"></i> Register</a>
40    |             </div>
41    |             <div class="header__top_right_auth">
42    |               <a href="{{ route('login') }}><i class="fa fa-user"></i> Login</a>
43    |             </div>
44    |             @else
45    |               <div class="header__top_right_auth">
46    |                 <a href="/profilpembeli"><i class="fa fa-user"> {{Auth::user()->name}}</i></a>
47    |               </div>
48    |               <div class="header__top_right_auth">
49    |                 <a href="{{ route('logout') }}><i class="ti-power-off text-primary"></i> Logout</a>
50    |               </div>
51    |             @endif
52    |           </div>
53    |         </div>
54    |       </div>
55    |     </div>
56    |   </div>
57    | </div>
58  </div>
59  <div class="container">
60    <div class="row">
61      <div class="col-lg-3">
62        <div class="header__logo">
63          <a href="/"></a>
64        </div>
65      </div>
66      <div class="col-lg-6">
67        <nav class="header__menu">
68          <ul>
69            <li class="active"><a href="/">Home</a></li>
70            <li @if(Route::currentRouteName() == "purchase" ) class="active" @endif><a href="{{ route('purchase') }}>Purchase</a></li>
71            <li @if(Route::currentRouteName() == "pembeli.history.index" ) class="active" @endif><a href="{{ route('pembeli.history.index') }}>History</a></li>
72            <li @if(Route::currentRouteName() == "chat.index" ) class="active" @endif><a href="{{ route('chat.index') }}>Chat</a></li>
73            <li @if(Route::currentRouteName() == "profilpembeli" ) class="active" @endif><a href="{{ route('profilpembeli') }}>Profil</a></li>
74          </ul>
75        </nav>
76      </div>
77      <div class="col-lg-3">
78        <div class="header__cart">
```

```

80
81         <ul>
82             <!-- <li><a href="#">i class="fa fa-bell"</a></li> -->
83             @if(count($cartItems) > 0)
84                 <i class="fa fa-shopping-bag"></i> <span>{{ count($cartItems) }}</span></a></li>
85             @endif
86         </ul>
87         <!-- <div class="header__cart__price">item: <span>$150.00</span></div> -->
88     </div>
89     <div class="humberger__open">
90         <i class="fa fa-bars"></i>
91     </div>
92 </div>
93 </div>
94 </header>
95 <!-- Header Section End -->
96
97 <!-- Hero Section Begin -->
98 <section class="hero">
99     <div class="container">
100        <div class="row">
101            <div class="col-lg-3">
102                <div class="hero__categories">
103                    <div class="hero__categories__all">
104                        <i class="fa fa-bars"></i>
105                        <span>All Categories</span>
106                    </div>
107                    <ul>
108                        @foreach($dataKategori as $kategori)
109                            <li><a href="{{ route('category', $kategori->id) }}>{{ $kategori->nama_kategori }}</a></li>
110                        @endforeach
111                    </ul>
112                </div>
113            </div>
114            <div class="col-lg-9">
115                <div class="hero__search">
116                    <div class="hero__search_form" style="width: 100%;>
117
118                        <form action="{{ route('search') }}" method="GET">
119                            <input type="text" name="keyword" placeholder="What do you need?">
120                            <button type="submit" class="site-btn">SEARCH</button>
121                        </form>
122                    </div>
123                    <div class="hero__item set-bg" data-setbg="{{ asset('frontend/img/bannerfix.png') }}>
124                        <div class="hero__text">
125                            <p>Kesepresiaan Makanan</p>
126                            <h2>Khas <br />Sekret Barat</h2>
127                            <p>Makanan Olahan Masyarakat Payakumbuh</p>
128                            <a href="#" class="primary-btn">SHOP NOW</a>
129                        </div>
130                    </div>
131                </div>
132            </div>
133        </div>
134    </section>
135 <!-- Hero Section End -->
136
137 <!-- Featured Section Begin -->
138 <section class="featured spad">
139     <div class="container">
140        <div class="row">
141            <div class="col-lg-12">
142                <div class="section-title">
143                    <h2>Featured Store</h2>
144                </div>
145                <div class="featured__controls" style="overflow-x: auto; white-space: nowrap;">
146                    <ul style="display: inline-block;">
147                        @foreach($sumkm as $umkm)
148                            <li class="active" data-filter="*">>All</li>
149                            <li data-filter=".{{ strToLower(str_replace(' ', '-', $umkm->nama)) }}>{{ $umkm->nama }}</li>
150                        @endforeach
151                    </ul>
152                </div>
153            </div>
154        </div>
155
156        <div class="products mb-3">
157            <div class="row featured__filter">
158                @foreach ($sumkm as $umkm)
159                    <div class="col-lg-3 col-md-4 col-sm-6 mix {{ strToLower(str_replace(' ', '-', $umkm->nama)) }}>
160
161                        <div class="product">
162                            <figure class="product-media">
163                                <a href="{{ route('detail-umkm', $umkm->id) }}>
164                                    
166                                </a>
167                            </figure><!-- End .product-media -->
168
169                            <div class="product-body">
170                                <div class="product-cat">
171                                    <div><!-- End .product-cat -->
172                                    <div class="product-title"><a class="fa fa-home"
173                                        href="{{ route('detail-umkm', $umkm->id) }}>
174                                            {{ $umkm->nama }}</a></div>
175                                    <!-- End .product-title -->
176                                    <div class="product-price">
177                                        <div><!-- End .product-price -->
178
179                                        <span class="fa fa-map-marker" &nbsp; {{ $umkm->alamat }}</span>
180
181                                    </div><!-- End .product-body -->
182                                </div><!-- End .product -->
183                            </div><!-- End .col-sm-6 col-md-4 col-xl-3 -->
184
185                            @endforeach
186                        </div><!-- End .row -->
187                    </div><!-- End .products -->
188
189            </div>
190        </section>

```

```

193     <!-- Footer Section Begin -->
194     <Footer class="footer spad">
195         <div class="container">
196             <div class="row">
197                 <div class="col-lg-12">
198                     <div class="footer__copyright">
199                         <div class="footer__copyright_text"><p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. -->
200                         Marketplace UMKM, Makanan Khas Olahan Sumatra Barat.
201                         </p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --></div>
202                         <div class="footer__copyright_payment"><p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. -->
203                         This app made by Distifia Oktari
204                         </p><!-- Link back to Colorlib can't be removed. Template is licensed under CC BY 3.0. --></div>
205                     </div>
206                 </div>
207             </div>
208         </div>
209     </Footer>
210     <!-- Footer Section End -->
211     <!-- Js Plugins -->
212     <script src="{{ asset('frontend/js/jquery-3.3.1.min.js') }}"></script>
213     <script src="{{ asset('frontend/js/bootstrap.min.js') }}"></script>
214     <script src="{{ asset('frontend/js/jquery.nice-select.min.js') }}"></script>
215     <script src="{{ asset('frontend/js/jquery-ui.min.js') }}"></script>
216     <script src="{{ asset('frontend/js/jquery.slicknav.js') }}"></script>
217     <script src="{{ asset('frontend/js/mixitup.min.js') }}"></script>
218     <script src="{{ asset('frontend/js/owl.carousel.min.js') }}"></script>
219     <script src="{{ asset('frontend/js/main.js') }}"></script>
220   </body>
221 </html>

```

## 2. Halaman search

```

176     <!-- Hero Section Begin -->
177     <section class="hero">
178         <div class="container">
179             <div class="row">
180                 <div class="col-lg-3">
181                     <div class="hero__categories">
182                         <div class="hero__categories__all">
183                             <i class="fa fa-bars"></i>
184                             <span>All Categories</span>
185                         </div>
186                         <ul>
187                             @foreach($dataKategori as $kategori)
188                             <li><a href="{{ route('category', $kategori->id) }}>{{ $kategori->nama_kategori }}</a></li>
189                             @endforeach
190                         </ul>
191                     </div>
192                 </div>
193                 <div class="col-lg-9">
194                     <div class="hero__search">
195                         <div class="hero__search__form" style="width: 100%;>
196                             <form action="{{ route('search') }}" method="GET">
197                                 <input type="text" name="keyword" placeholder="What do you need?">
198                                 <button type="submit" class="site-btn">SEARCH</button>
199                             </form>
200                         </div>
201                     </div>
202                     <div class="row featured__filter">
203                         @if ($message)
204                             <div class="col-lg-12">
205                                 <div class="alert alert-info" role="alert">
206                                     | {{ $message }}
207                                 </div>
208                             </div>
209                         @endif
210                         @foreach($produk as $item)
211                             @if ($item->Imgproduk)
212                                 <div class="col-lg-3 col-md-4 col-sm-6 mix {{ strToLower(str_replace(' ', '-', $item->kategori->nama_kategori)) }}>
213                                     <div class="featured__item">
214                                         <div class="featured__item_pic set-bg" data-setbg="{{ asset('images/produk' . $item->Imgproduk->first()->
215                                             <div class="featured__item_pic set-bg" data-setbg="{{ asset('images/produk' . $item->Imgproduk->first()->
216                                                 <ul class="featured__item_pic_hover">
217                                                     <li><a href="{{ route('detail-produk', ['id' => $item->id]) }}><i class="fa fa-info"></i></a></li>
218                                                 </ul>
219                                         <div class="featured__item_text">
220                                             <h6><a href="{{ route('detail-produk', ['id' => $item->id]) }}>{{ $item->nama }}</a></h6>
221                                             <p><a class="fa fa-home" href="{{ route('detail-umkm', ['id' => $item->umkm_id]) }}>{{ $item->Umkm->n
222                                                 @php
223                                                 $hargaTerendahProduk = $item->Detail_Produk->min('harga');
224                                                 $hargaTertinggiProduk = $item->Detail_Produk->max('harga');
225                                                 @endphp
226                                                 <h5>Rp{{ number_format($hargaTerendahProduk, 0, ',', '.') }},00 - Rp{{ number_format($hargaTertinggiPro
227                                         </div>
228                                     </div>
229                                 @endif
230                             @endforeach
231                         </div>
232                     </div>
233                 </div>
234             </div>
235         </div>
236     </section>
237     <!-- Hero Section End -->

```

### 3. Halaman detail umkm

```

1011     <!-- Hero Section Begin -->
1012     <section class="hero">
1013         <div class="container">
1014             <div class="row">
1015                 <div class="col-lg-3">
1016                     <div class="hero_categories">
1017                         <div class="hero_categories_all">
1018                             <i class="fa fa-bars"></i>
1019                             <span>All Categories</span>
1020                         </div>
1021                         <ul>
1022                             @foreach($dataKategori as $kategori)
1023                             <li>
1024                                 <a href="{{ route('kategoriumkm', ['umkmId' => $umkm->id, 'kategoriId' => $kategori->id]) }}">
1025                                     {{ $kategori->nama_kategori }}
1026                                 </a>
1027                             </li>
1028                         @endforeach
1029                     </ul>
1030                 </div>
1031             </div>
1032             <div class="col-lg-9">
1033                 <div class="row">
1034                     <div class="col-lg-6 col-md-6">
1035                         <div class="product_details_pic">
1036                             <div class="product_details_pic_item">
1037                                 
1038                             </div>
1039                             <div class="google-map" id="map-show" style="height:200px"></div>
1040                         </div>
1041                     <div class="col-lg-6 col-md-6">
1042                         <div class="product_details_text">
1043                             <h2>{{ $umkm->nama }}</h2>
1044                             <ul>
1045                                 <li><b>Alamat</b></li>
1046                                 <li><span class="fa fa-map-marker"> &nbsp; {{ $data['umkm']->nama_kota }}</span></li>
1047                                 <li><span>{{ $umkm->alamat }}</span></li>
1048                                 <li><b>Informasi</b><br> <span>{{ $umkm->informasi }}</span></li>
1049                             </ul>
1050                         <li><b>Phone</b> <span>{{ $umkm->handphone }}</span></li>
1051                         </ul>
1052                     </div>
1053                 @if (Auth::user())
1054                     <div class="fabs">
1055                         <div class="chat">
1056                             <div class="chat_header">
1057                                 <div class="chat_option">
1058                                     <div class="header_img">
1059                                         @if ($umkm->Penjual->img != NULL)
1060                                             
1062                                         @else
1063                                             
1064                                         @endif
1065                                     </div>
1066                                     <span id="chat_head">{{ $umkm->Penjual->first_name }}</span>
1067                                     {{ $umkm->Penjual->last_name }}<br> <span
1068                                         class="agent">{{ $umkm->name }}</span> <span
1069                                         class="online">{{ Carbon\Carbon::now() }}</span>
1070                                     <span id="chat_fullscreen_loader">
1071                                         <div class="chat_fullscreen_loader">
1072                                             <img class="fullscreen zmdi zmdi-window-maximize"/>
1073                                         </div>
1074                                     </span>
1075                                 </div>
1076                             </div>
1077                             <div id="chat_fullscreen" class="chat_conversion chat_converse">
1078                                 @if($chat)
1079                                     @foreach ($chat->chatDetail as $chatDetail)
1080                                         @if ($chatDetail->sender == Auth::user()->id)
1081                                             <span class="chat_msg_item chat_msg_item_user">
1082                                                 {{($chatDetail->message)}}
1083                                             <div class="status">{{ Helper::timeago($chatDetail->created_at) }}</div>
1084                                         @else
1085                                             <span class="chat_msg_item chat_msg_item_admin">
1086                                                 <div class="chat_avatar">
1087                                                     @if ($chat->Penjual->avatar != NULL)
1088                                                         
1089                                                     @else
1090                                                         
1091                                                     @endif
1092
1093                                                     </div>
1094                                                     <div class="chat_message">
1095                                                         @if ($chat->Penjual->avatar != NULL)
1096                                                             
1097                                                         @else
1098                                                             
1099                                                         @endif
1100
1101                                                     </div>>{{ $chatDetail->message }}</span>
1102                                         @endif
1103                                         @endforeach
1104                                         @endif
1105                                     </div>
1106                                     <div class="fab_field">
1107                                         <a id="fab_send" onclick="sendMessage([{$umkm->id}], {{Auth::user()}->pembeli->id}), {{($umkm->Penjual->id)}, {{Auth::user()}->Penjual() ? 1 : 0}}>
1108                                         <div class="fa fa-paper-plane-o"></div></a>
1109                                         <textaree id="chatSend" name="chat_message" placeholder="Send a message" class="chat_field chat_message"></textaree>
1110                                         </div>
1111                                         <a id="prime" class="fab"><i class="fa fa-commenting-o"></i></a>
1112                                     </div>
1113                                 @endif
1114                             </div>
1115                         </div>
1116                     </div>
1117                 </div>
1118             </section>
1119         <!-- Hero Section End -->

```

#### 4. Halaman kategori

```

176 <!-- Hero Section Begin -->
177 <section class="hero">
178     <div class="container">
179         <div class="row">
180             <div class="col-lg-3">
181                 <div class="hero__categories">
182                     <div class="hero__categories__all">
183                         <i class="fa fa-bars"></i>
184                         <span>All Categories</span>
185                     </div>
186                     <ul>
187                         @foreach($dataKategori as $kategori)
188                             <li><a href="{{ route('category', $kategori->id) }}>{{ $kategori->nama_kategori }}</a></li>
189                         @endforeach
190                     </ul>
191                 </div>
192             </div>
193             <div class="col-lg-9">
194                 <div class="hero__search">
195
196                     <div class="row featured__filter">
197                         @if ($produk->isEmpty())
198                             <div class="col-lg-12">
199                                 <p>No product in this category.</p>
200                             </div>
201                         @endif
202                         @foreach($produk as $item)
203                             @if ($item->Imgproduk)
204                                 <div class="col-lg-3 col-md-4 col-sm-6 mix {{ strToLower(str_replace(' ', '-', $item->kategori->nama_kategori)) }}>
205                                     <div class="featured__item">
206                                         <div class="featured__item_pic set-bg" data-setbg="{{ asset('images/produk/' . $item->Imgproduk->first()->image) }}>
207                                             <ul class="featured__item_pic_hover">
208                                                 <li><a href="{{ route('detail-produk', ['id' => $item->id]) }}><i class="fa fa-info"></i></a></li>
209                                             </ul>
210                                         </div>
211                                         <div class="featured__item_text">
212                                             <h6><a href="{{ route('detail-produk', ['id' => $item->id]) }}>{{ $item->nama }}</a></h6>
213                                             <p><a class="fa fa-home" href="{{ route('detail-umkm', ['id' => $item->umkm_id]) }}>{{ $item->Umkm->nama }}</a></p>
214                                         @php
215                                             $hargaTerendahProduk = $item->Detail_Produk->min('harga');
216                                             $hargaTertinggiProduk = $item->Detail_Produk->max('harga');
217                                         @endphp
218                                             <h5>Rp{{ number_format($hargaTerendahProduk, 0, ',', '.') }},00 - Rp{{ number_format($hargaTertinggiProduk, 0, ',', '.') }},00</h5>
219                                         </div>
220                                     </div>
221                             @endif
222                         @endforeach
223                     </div>
224                 </div>
225             </div>
226         </div>
227     </div>
228 </section>
229 <!-- Hero Section End -->

```

## f. Bagian Kurir

### 1. Halaman *index* pengiriman

```
1  @extends('layouts.kurir')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5  <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}">
6  @endsection
7
8  @section('content')
9  <div class="content-wrapper">
10 <section class="content">
11 <div class="container-fluid">
12     <div class="row">
13         <div class="col-12">
14             <div class="card">
15                 <div class="card-header">
16                     <h3 class="card-title">Transaksi Pengiriman</h3>
17                 </div>
18                 <!-- /.card-header -->
19                 <div class="card-body">
20                     <table id="example1" class="table table-bordered table-striped dataTables-booking">
21                         <thead>
22                             <tr>
23                                 <th>Tanggal</th>
24                                 <th>Umkm</th>
25                                 <th>Total Pembelian</th>
26                                 <th>Noresi</th>
27                                 <th>Action</th>
28                             </tr>
29                         </thead>
30                         <tbody>
31                             @foreach($pembelians as $pembelian)
32                             <tr>
33                                 <td>{{ $pembelian->updated_at->format('Y-m-d') }}</td>
34                                 <td>{{ $pembelian->Detail_Pembelian->first()->Produk->Umkm->nama }}</td>
35                                 <td>{{ $pembelian->total_harga }}</td>
36                                 <td>{{ $pembelian->noresi }}</td>
37                                 <td>
38                                     <a href="{{ route('pengiriman.show', $pembelian->id) }}" class="btn btn-info">Detail</a>
39                                 </td>
40                             @endforeach
41                         </tbody>
42                     </table>
43                 </div>
44                 <!-- /.card-body -->
45             </div>
46             <!-- /.card -->
47         </div>
48         <!-- /.col -->
49     </div>
50     <!-- /.row -->
51 </div>
52 <!-- /.container-fluid -->
53
54 </section>
55 </div>
56
57 @endsection
58
59 @section('script')
60 <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
61 <script>
62     $(document).ready(function () {
63         $('.dataTables-booking').DataTable({
64
65             responsive: true,
66             searchDelay: 500,
67             processing: true,
68             buttons: [
69                 {
70                     extend: 'print',
71                     className: 'btn-inverse',
72                     exportOptions: {
73                         columns: [0, 1]
74                     }
75                 },
76                 'copy',
77                 {
78                     extend: 'excel',
79                     className: 'btn-inverse',
```

```
80         exportOptions: {
81             columns: [0, 1]
82         }
83     },
84     'csvHtml5',
85     {
86         extend: 'pdf',
87         className: 'btn-inverse',
88         exportOptions: {
89             columns: [0, 1]
90         }
91     }
92 ],
93
94 });
95 table.on('order.dt search.dt', function () {
96     table.column(0, {
97         search: 'applied',
98         order: 'applied'
99     }).nodes().each(function (cell, i) {
100         cell.innerHTML = i + 1;
101     });
102 }).draw();
103 $('#btn-print').on('click', function (e) {
104     e.preventDefault();
105     table.button(0).trigger();
106 });
107 $('#btn-copy').on('click', function (e) {
108     e.preventDefault();
109     table.button(1).trigger();
110 });
111 $('#btn-excel').on('click', function (e) {
112     e.preventDefault();
113     table.button(2).trigger();
114 });
115 $('#btn-csv').on('click', function (e) {
116     e.preventDefault();
117     table.button(3).trigger();
118 });
119
120 $('#btn-pdf').on('click', function (e) {
121     e.preventDefault();
122     table.button(4).trigger();
123 });
124
125 $('#c_field').prop("disabled", true);
126 $('#c_venue').prop("disabled", true);
127
128 $(".myselect2").select2({
129     width: '100%',
130     allowClear: true
131 });
132 $('.myselect2').val(0).change();
133
134 function dateChange() {
135     $('#c_venue').prop("disabled", false);
136     $('.myselect2').val(0).change();
137     $(".selectgroup-item").remove();
138 }
139
140 $('#c_venue').on('select2:select', function (e) {
141     let venue = $('#c_venue').val();
142     if (venue) {
143         $('#c_field').val(0).change();
144         $(".selectgroup-item").remove();
145         fethDatafield(venue);
146         $('.myselect2').prop("disabled", false);
147     } else {
148         $('#c_field').prop("disabled", true);
149     }
150 });
151
152
153 </script>
154 @endsection
```

## 2. Halaman *show* pengiriman

```
1 @extends('layouts.kurir')
2
3 @section('css')
4 <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5 <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}"/>
6 @endsection
7
8 @section('content')
9 <div class="content-wrapper">
10 <section class="content">
11   <div class="container-fluid">
12     <div class="row">
13       <div class="col-12">
14         <div class="card">
15           <div class="card-header">
16             <h3 class="card-title">Detail Transaksi Pengiriman
17             </h3>
18             @if ($spembelian->status_pembelian == 2)
19
20               <a type="button" href="#" class="btn btn-primary" style="float:right;margin-left:5px;" data-toggle="modal" data-target="#tambahStokModal">Input Bukti
21               <!-- Modal bukti pengiriman -->
22             <div class="modal fade" id="tambahStokModal" tabindex="-1" role="dialog" aria-labelledby="tambahStokModalLabel" aria-hidden="true">
23               <div class="modal-dialog" role="document">
24                 <div class="modal-content">
25                   <div class="modal-header">
26                     <h5 class="modal-title" id="tambahStokModalLabel">Input Bukti Pengiriman</h5>
27                     <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28                       <span aria-hidden="true">&times;</span>
29                     </button>
30                   </div>
31                   <form action="{{ route('input-bukti', ['id' => $spembelian->id]) }}" method="POST" enctype="multipart/form-data">
32                     @csrf
33                     <div class="modal-body">
34                       <div class="form-group">
35                         <label for="bukti">Bukti Pengiriman</label>
36                         <input type="file" name="bukti_pengiriman" id="bukti_pengiriman" class="form-control-file" required>
37                       </div>
38                     <div class="modal-footer">
39
40                       <button type="button" class="btn btn-secondary" data-dismiss="modal">Batal</button>
41                       <button type="submit" class="btn btn-primary">Simpan</button>
42                     </div>
43                   </form>
44                 </div>
45               </div>
46             @else ($spembelian->status_pembelian == 0 && $spembelian->status_pembayaran == 0 && Carbon\Carbon::now() >= Carbon\Carbon::parse($spembelian->created_at) - 15)
47               <span style="float:right;margin-left:5px;" class="badge badge-danger" >Expired</span>
48             @elseif ($spembelian->status_pembelian == 0 && $spembelian->status_pembayaran == 0)
49               <span style="float:right;margin-left:5px;" class="badge badge-info" >Menunggu Pembayaran</span>
50             @elseif ($spembelian->status_pembelian == 0)
51               <span style="float:right;margin-left:5px;" class="badge badge-primary" >Menunggu Konfirmasi</span>
52
53             @elseif ($spembelian->status_pembelian == 2)
54               <a type="button" style="float:right;margin-left:5px;" class="btn btn-primary"
55                 onclick="finish('{{ $spembelian->id }}')" href="javascript:void(0)">
56                 Transaksi Selesai
57               </a>
58               <!-- <span style="float:right;margin-left:5px;" class="badge badge-primary" >Pesanan Dikirim</span -->
59             @elseif ($spembelian->status_pembelian == 3)
60               <span style="float:right;margin-left:5px;" class="badge badge-danger" >Transaksi Ditolak</span>
61             @elseif ($spembelian->status_pembelian == 4)
62               <span style="float:right;margin-left:5px;" class="badge badge-info" >Selesai</span>
63             @endif
64
65           </div>
66         <!-- ./card-header -->
67         <div class="card-body">
68           <div class="row">
69             <div class="col-sm-4">
70               <!-- text input -->
71               <div class="form-group">
72                 <label><i class="fas fa-user-alt" data-toggle="tooltip" data-title="Nama Pembeli" data-placement="bottom"></i> {{ $spembelian->nama_pembeli }}</label>
73                 <br>
74                 <label><i class="fas fa-calendar-alt" data-toggle="tooltip" data-title="Tanggal Transaksi" data-placement="bottom"></i> {{ date('d-m-Y', strtotime($spembelian->updated_at)) }}</label>
75                 <br>
76               </div>
77             </div>
78           </div>
79         </div>
80       </div>
81     </div>
82   </div>
83 </section>
84 </div>
85 </div>
```

```

    <br>
    <br>
</div>
<div class="col-sm-4">
    <!-- text input -->
    <div class="form-group">
@if ($pembelian->pembayaran)
    <label><i class="far fa-credit-card" data-toggle="tooltip"
        data-title="Metode Pembayaran" data-placement="bottom"></i> {{($pembelian->Pembayaran->Detail_Metode_Pembayaran->Metode_Pembayaran
    <br>
    <label><i class="fas fa-tasks" data-toggle="tooltip" data-title="Status Pembelian"
        data-placement="bottom"></i>
@if($pembelian->status_pembelian == 0)
    Menunggu Konfirmasi
@elseif($pembelian->status_pembelian == 1)
    Pesanan Disediakan
@elseif($pembelian->status_pembelian == 2)
    Pesanan Dikirim
@elseif ($pembelian->status_pembelian == 3)
    Pesanan Ditolak
@elseif ($pembelian->status_pembelian == 4)
    Transaksi Selesai
@endif
</label>
@endif
</div>
</div>
<div class="col-sm-4">
@if ($pembelian->pembayaran)
    <div class="form-group">
        <label>Bukti Pembayaran </label><br>
        
    </div>
</div>
</div>
<div class="row">
<div class="col-sm-12">
    <!-- select -->
    <h2><i>Kode Pemesanan : {{($pembelian->token)}}</i></h2>
    <h2><i>Nonesi : {{($pembelian->nonesi)}}</i></h2> <br>
    <div class="form-group">
        <h2>Detail Pemesanan </h2>
        <div class="form-group">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th style="width: 10px">No</th>
                        <th>Nama Produk</th>
                        <th>Kemasan</th>
                        <th>Jumlah</th>
                        <th>Harga Produk</th>
                        <th>Subtotal</th>
                    </tr>
                </thead>
                <tbody>
                    @php
                        // Inisialisasi variabel
                        $index = 1;
                        $total_harga = 0;
                        $subtotal_produk = 0;
                    @endphp
                    @foreach($pembelian->Detail_Pembelian as $item)
                        <tr>
                            <td>{{ $index }}</td>
                            <td>{{ $item->produk->nama }}</td>
                            <td>{{ $item->kemasan->ukuran }}</td>
                            <td>{{ $item->jumlah }}</td>
                            <td>Rp {{ number_format($item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0, ',', '.') }}</td>
                            <td>Rp {{ number_format($item->jumlah * $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0,
                        @endphp
                    @endphp
                    // Perhitungan total harga dan subtotal produk
                    <td>{{ $total_harga + $item->jumlah * $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
                        $subtotal_produk += $item->jumlah * $item->produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
                    @endphp
                @endforeach
            </tbody>
            <tfoot>
                <tr>
                    <td colspan="5" style="text-align: right;"><strong>Subtotal Produk:</strong></td>
                    <td>Rp {{ number_format($subtotal_produk, 0, ',', '.') }}</strong></td>
                </tr>
                <tr>
                    <td colspan="5" style="text-align: right;"><strong>Biaya Pengiriman:</strong></td>
                    <td>Rp {{ number_format($pembelian->total_harga - $subtotal_produk, 0, ',', '.') }}</strong></td>
                </tr>
                <tr>
                    <td colspan="5" style="text-align: right;"><strong>Total Harga:</strong></td>
                    <td>Rp {{ number_format($pembelian->total_harga, 0, ',', '.') }}</strong></td>
                </tr>
            </tfoot>
        </table>
    </div>
</div>
<div class="card-body" -->
</div>
<!-- /.card -->
<!-- /.col1 -->
</div>
<!-- /.row -->
</div>
<!-- /.container-fluid -->
</section>
</div>
{!! Form::open(['method'=>'PATCH', 'route' => ['kurir.pengiriman.finish', 0], 'style' =>
    'display: flex; justify-content: space-between; align-items: center; margin-bottom: 10px;'])

```

```

200
201 @section('script')
202 <script>
203
204
205     function selectCheckbox(data){
206         var index = $(data).index(".selectgroup-itemInput");
207         console.log($(data));
208         var ini = $(".selectgroup-itemInput").eq(index+1);
209         console.log($(ini));
210         $(ini).prop("disabled", false);
211         $(ini).attr('style','background-color:red');
212     }
213
214
215     function finish(id){
216         swal({
217             title: "Apakah pesanan telah selesai?",
218             text: "",
219             type: "warning",
220             showCancelButton: true,
221             confirmButtonColor: "#f0c998",
222             confirmButtonText: "Ya, Selesaikan!",
223             cancelButtonText: "Tidak, batalkan!",
224             closeOnConfirm: false,
225             closeOnCancel: false
226         },
227         function (isConfirm) {
228             if (isConfirm) {
229                 swal("Pesanan Selesai!", "Pesanan telah diselesaikan", "success");
230                 $('#finish').attr('action', '{{ route('pengiriman') }}/' + id + '/finish');
231                 $('#finish').submit();
232             } else {
233                 swal("Batal", "Anda batal menyelesaikan pesanan", "error");
234             }
235         });
236     }
237 });
238 
```

### 3. Halaman *index history*

```

1 @extends('layouts.kurir')
2
3 @section('css')
4 <link href="https://cdn.jsdelivr.net/npm/select2@4.0.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5 <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}"/>
6 @endsection
7
8 @section('content')
9 <div class="content-wrapper">
10 <section class="content">
11 <div class="container-fluid">
12     <div class="row">
13         <div class="col-12">
14             <div class="card">
15                 <div class="card-header">
16                     <h3 class="card-title">History Pengiriman</h3>
17                 <!-- /.card-header -->
18                 <div class="card-body">
19                     <table id="example1" class="table table-bordered table-striped dataTables-booking">
20                         <thead>
21                             <tr>
22                                 <th>Tanggal</th>
23                                 <th>Umkm</th>
24                                 <th>Total Pembelian</th>
25                                 <th>Noreksi</th>
26                                 <th>Action</th>
27                             </tr>
28                         </thead>
29                         <tbody>
30                             @foreach($pembelians as $pembelian)
31                             <tr>
32                                 <td>{{ $pembelian->updated_at->format('Y-m-d') }}</td>
33                                 <td>{{ $pembelian->Detail_Pembelian->first()->Produk->Umkm->name }}</td>
34                                 <td>{{ $pembelian->total_harga }}</td>
35                                 <td>{{ $pembelian->noreksi }}</td>
36                                 <td>
37                                     | <a href="{{ route('history.kurir|show', $pembelian->id) }}" class="btn btn-info">Detail</a>
38                                 </td>
39                             </tr>
40                         </tbody>
41                     </table>
42                 </div>
43             </div>
44         </div>
45     </div>
46 </div>
47 </section>
48 </div>
49 
```

```

41 |     @foreach
42 |     </tbody>
43 |   </table>
44 |   <!-- /.card-body -->
45 | </div>
46 | <!-- /.card -->
47 | </div>
48 | <!-- /.col -->
49 | </div>
50 | <!-- /.row -->
51 | </div>
52 | <!-- /.container-fluid -->
53 |
54 </section>
55 </div>
56
57 @endsection
58
59 @section('script')
60 <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
61 <script>
62     $(document).ready(function () {
63         $('.dataTables-booking').DataTable({
64             responsive: true,
65             searchDelay: 500,
66             processing: true,
67             buttons: [
68                 { extend: 'print',
69                  className: 'btn-inverse',
70                  exportOptions: {
71                      columns: [0, 1]
72                  }
73                },
74                'copy',
75                {
76                    extend: 'excel',
77                    className: 'btn-inverse',
78                    columns: [0, 1]
79                }
80            ],
81            'csvHtml5',
82            {
83                extend: 'pdf',
84                className: 'btn-inverse',
85                exportOptions: {
86                    columns: [0, 1]
87                }
88            }
89        ],
90    });
91
92     table.on('order.dt search.dt', function () {
93         table.column(0, {
94             search: 'applied',
95             order: 'applied'
96         }).nodes().each(function (cell, i) {
97             cell.innerHTML = i + 1;
98         });
99     }).draw();
100    $('#btn-print').on('click', function (e) {
101        e.preventDefault();
102        table.button(0).trigger();
103    });
104    $('#btn-copy').on('click', function (e) {
105        e.preventDefault();
106        table.button(1).trigger();
107    });
108    $('#btn-excel').on('click', function (e) {
109        e.preventDefault();
110        table.button(2).trigger();
111    });
112    $('#btn-csv').on('click', function (e) {
113        e.preventDefault();
114        table.button(3).trigger();
115    });
116    $('#btn-pdf').on('click', function (e) {
117
118        e.preventDefault();
119        table.button(4).trigger();
120    });
121
122 });
123
124 });
125
126 $('#c_field').prop("disabled", true);
127 $('#c_venue').prop("disabled", true);
128
129 $('.myselect2').select2({
130     width: '100%',
131     allowClear: true
132 });
133 $('.myselect2').val(0).change();
134
135 function dateChange() {
136     $('#c_venue').prop("disabled", false);
137     $('.myselect2').val(0).change();
138     $(".selectgroup-item").remove();
139 }
140
141 $('#c_venue').on('select2:select', function (e) {
142     let venue = $('#c_venue').val();
143     if (venue) {
144         $('#c_field').val(0).change();
145         $(".selectgroup-item").remove();
146         fetchDataField(venue);
147         $('#c_field').prop("disabled", false);
148     } else {
149         $('#c_field').prop("disabled", true);
150     }
151 });
152
153 </script>
154 @endsection

```

#### 4. Halaman show history

```

1  @extends('layouts.kurir')
2
3  @section('css')
4  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
5  <link rel="stylesheet" href="{{ asset('templates/css/custom/selectgroup.css') }}">
6  @endsection
7
8  @section('content')
9  <div class="content-wrapper">
10 <section class="content">
11   <div class="container-fluid">
12     <div class="row">
13       <div class="col-12">
14         <div class="card">
15           <div class="card-header">
16             <h3 class="card-title">Detail Transaksi Pengiriman
17             </h3>
18             @if ($spembelian->status_pembelian == 2)
19               <a type="button" href="#" class="btn btn-primary" style="float:right; margin-left:5px;" data-toggle="modal" data-target="#tambahStokModal">Input Bukti
20               <!-- Modal bukti pengiriman -->
21               <div class="modal fade" id="tambahStokModal" tabindex="-1" role="dialog" aria-labelledby="tambahStokModalLabel" aria-hidden="true">
22                 <div class="modal-dialog" role="document">
23                   <div class="modal-content">
24                     <div class="modal-header">
25                       <h5 class="modal-title" id="tambahStokModalLabel">Input Bukti Pengiriman</h5>
26                       <button type="button" class="close" data-dismiss="modal" aria-label="Close">
27                         <span aria-hidden="true">&times;

```

```

80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
<br>
<br>
</div>
</div>
<div class="col-sm-4">
    <!-- text input -->
    <div class="form-group">
        @if ($pembelian->Pembayaran)
            <label><i class="far fa-credit-card" data-toggle="tooltip" data-title="Metode Pembayaran" data-placement="bottom"></i> {{ $pembelian->Pembayaran->Detail_Pembayaran->Metode_Pembayaran }}<br>
            <label><i class="fas fa-tasks" data-toggle="tooltip" data-title="Status Pembelian" data-placement="bottom"></i>
                @if($pembelian->status_pembelian == 0)
                    Menunggu Konfirmasi
                @elseif($pembelian->status_pembelian == 1)
                    Pesanan Dipisahkan
                @elseif($pembelian->status_pembelian == 2)
                    Pesanan Dikirim
                @elseif ($pembelian->status_pembelian == 3)
                    Pesanan Ditolak
                @elseif ($pembelian->status_pembelian == 4)
                    Transaksi Selesai
                @endif
            </label>
        @endif

        </div>
    </div>
<div class="col-sm-4">
    <div class="form-group">
        <label>Bukti Pengiriman </label><br>
        
    </div>
</div>
</div>
</div>
<div class="row">
    <div class="col-sm-12">
        <!-- select -->
        <h2>Kode Pemesanan : {{ $pembelian->token }}</h2>
        <h2>Noreksi : {{ $pembelian->noreksi }}</h2> <br>
        <div class="form-group">
            <h2>Detail Pemesanan </h2>
            <div class="table">
                <tbl_struct version="1.1" x="1" y="1">
                    <tbl_header version="1.1" x="1" y="1">
                        <tbl_info cols="6" rindex="1" cindex="1" ix="1" maxcspan="1" maxrspan="1" usedcols="6" usedrows="1" />
                        <tbl_r cells="6" ix="1" maxcspan="1" maxrspan="1" usedcols="6" usedrows="1" />
                    </tbl_header>
                    <tbl_info cols="6" rindex="2" cindex="1" ix="2" maxcspan="1" maxrspan="1" usedcols="6" usedrows="1" />
                    <tbl_r cells="6" ix="2" maxcspan="1" maxrspan="1" usedcols="6" usedrows="1" />
                </tbl_struct>
                <table class="table table-striped">
                    <thead>
                        <tr>
                            <th style="width: 10px">No</th>
                            <th>Nama Produk</th>
                            <th>Kemasan</th>
                            <th>Jumlah</th>
                            <th>Harga Produk</th>
                            <th>Subtotal</th>
                        </tr>
                    </thead>
                    <tbody>
                        @php
                            // Inisialisasi variabel
                            $index = 1;
                            $total_harga = 0;
                            $subtotal_produk = 0;
                        @endphp
                        @foreach ($pembelian->Detail_Pembelian as $item)
                            <tr>
                                <td>{{ $index }}</td>
                                <td>{{ $item->produk->nama }}</td>
                                <td>{{ $item->kemasan->ukuran }}</td>
                                <td>{{ $item->jumlah }}</td>
                                <td>Rp {{ number_format($item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0, ',', '.') }}</td>
                                <td>Rp {{ number_format($item->jumlah * $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga, 0, ',', '.') }}</td>
                            </tr>
                        @php
                            // Perhitungan total harga dan subtotal produk
                            $total_harga += $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga * $item->jumlah;
                            $subtotal_produk += $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga * $item->jumlah;
                        @endphp
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>

```

```
156
157         // Perhitungan total harga dan subtotal produk
158         $total_harga += $item->jumlah * $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
159         $subtotal_produk += $item->jumlah * $item->Produk->Detail_Produk->where('kemasan_id', $item->kemasan_id)->first()->harga;
160     @endphp
161     @endforeach
162     </tbody>
163     <tfoot>
164         <tr>
165             <td colspan="5" style="text-align: right;"><strong>Subtotal Produk:</strong></td>
166             <td><strong>Rp {{ number_format($subtotal_produk, 0, ',', '.') }}</strong></td>
167         </tr>
168         <tr>
169             <td colspan="5" style="text-align: right;"><strong>Biaya Pengiriman:</strong></td>
170             <td><strong>Rp {{ number_format($pembelian->total_harga - $subtotal_produk, 0, ',', '.') }}</strong></td>
171         </tr>
172         <tr>
173             <td colspan="5" style="text-align: right;"><strong>Total Harga:</strong></td>
174             <td><strong>Rp {{ number_format($pembelian->total_harga, 0, ',', '.') }}</strong></td>
175         </tr>
176     </tfoot>
177     </table>
178     </div>
179     </div>
180     </div>
181     </div>
182     </div>
183     <!-- /.card-body -->
184   </div>
185   <!-- /.card -->
186 </div>
187   <!-- /.col -->
188 </div>
189   <!-- /.row -->
190 </div>
191 <!-- /.container-fluid -->
192
193 </section>
194 <!-- Main Content -->
```





### 1. Implementasi Antarmuka Register

The screenshot shows a registration form titled "UMKM Sanjai". The form fields include: First Name, Last Name, Email, Password, Confirm Password, and Level (with a dropdown menu showing "Pilih Level"). Below the form are links for "Already registered?" and a "REGISTER" button.

### 2. Implementasi Antarmuka Melihat Data User

The screenshot shows a table titled "Data Admin" with two entries. The columns are: No, Nama, No Hp, Provinsi, Kota, Alamat, and Aksi (Edit and Hapus buttons). The data is as follows:

No	Nama	No Hp	Provinsi	Kota	Alamat	Aksi
1	distifiaa oktari	082285443515	Sumatera Barat	Padang	Komplek polda balai baru blok G/26	<button>Edit</button> <button>Hapus</button>
2	admin	089617778650	Sumatera Barat	Payakumbuh	Jln. bulakan balai kandi	<button>Edit</button> <button>Hapus</button>

### 3. Implementasi Antarmuka Menambah Data User

The screenshot shows a form titled "Menambahkan Data Admin" with fields for: Firstname, Lastname, No Handphone, Provinsi (dropdown menu), and Kota (dropdown menu). There is also a "Provinces" dropdown menu at the bottom right.

Alamat

Email

Password

Foto Profil

**Submit**

#### 4. Implementasi Antarmuka Mengedit Data User

UMKM Sanji



Mengubah Data Admin

Firstname

Lastname

No Handphone

Provinsi

Kota

Alamat

Email

Password

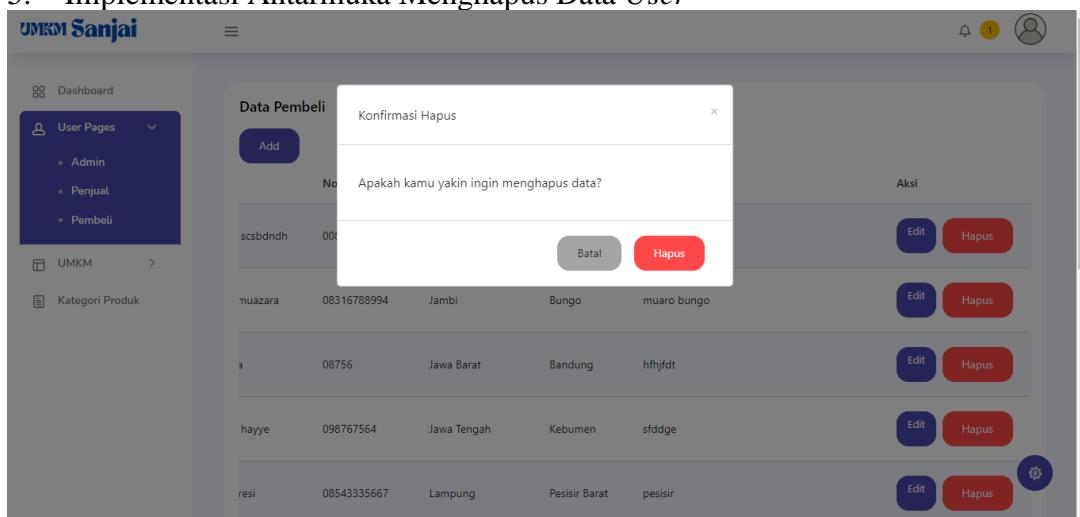
Leave it blank if you don't want to change the password.

Foto Profil

distifiaa.jpg

**Submit**

## 5. Implementasi Antarmuka Menghapus Data User



## 6. Implementasi Antarmuka Profil

Profile

Name: admin

No Handphone: 089617778650

Provinsi: Sumatera Barat

Kota: Payakumbuh

Alamat: jln. bulakan balai kandi

Email: admin@gmail.com

Edit Profile

Foto Profil: Choose File (No file chosen)

Fisrtname: admin

Lastname: Lastname

No Handphone: 089617778650

Provinsi: Sumatera Barat

Kota: Payakumbuh

Alamat: jln. bulakan balai kandi

Email: admin@gmail.com

Submit

Change Password

Current Password

New Password

Confirm Password

Submit

Copyright © 2021. Premium Bootstrap admin template from BootstrapDash. All rights reserved.  
Distributed by Themewagon

Hand-crafted & made with

## 7. Implementasi Antarmuka Melihat Kategori Produk

**Data Kategori**

No	Nama Kategori	Aksi
1	Manis	<button>Edit</button> <button>Hapus</button>
2	Keripik Olahan Singkong	<button>Edit</button> <button>Hapus</button>
3	Keripik Olahan Talas	<button>Edit</button> <button>Hapus</button>
4	Keripik Olahan Kentang	<button>Edit</button> <button>Hapus</button>
5	Snack Kipang	<button>Edit</button> <button>Hapus</button>

## 8. Implementasi Antarmuka Menambah Kategori Produk

**Menambahkan Data Kategori**

Nama Kategori

nama kategori

Submit

## 9. Implementasi Antarmuka Mengedit Kategori Produk

The screenshot shows a modal window titled "Mengubah Data Kategori". Inside the modal, there is a text input field labeled "Nama Kategori" containing the value "Keripik Olahan Singkong". Below the input field is a blue "Submit" button. The background of the main interface shows a sidebar with navigation items: Dashboard, User Pages, UMKM, and Kategori Produk, where Kategori Produk is highlighted.

## 10. Implementasi Antarmuka Menghapus Kategori Produk

The screenshot shows a modal window titled "Konfirmasi Hapus" asking "Apakah kamu yakin ingin menghapus data?". It contains two buttons: "Batal" (Cancel) and "Hapus" (Delete). In the background, there is a table titled "Data Kategori" listing five categories: 1. Keripik Olahan Singkong, 2. Keripik Olahan Talas, 3. Keripik Olahan Kentang, and 4. Snack Kipang. Each row has "Edit" and "Hapus" buttons. The sidebar on the left includes a "Data Kategori" section with "Add" and "No" buttons.

## 11. Implementasi Antarmuka UMKM Perlu Konfirmasi

The screenshot shows a table titled "List Data UMKM Perlu Konfirmasi" with one entry. The columns are: Nama Umkm, Pemilik, Alamat, Tanggal Pembuatan, and Action. The data row is: Sanjai Oviga 2, Vina Kurnia, Jl. Aceh, Tanjung Pauh 26222, 24-08-2023 / 15:07, with three icons in the Action column (info, checkmark, and delete). The sidebar on the left has a "UMKM" section with "Perlu Konfirmasi", "Ditolak", and "Terkonfirmasi" options, and a "Kategori Produk" section.

## 12. Implementasi Antarmuka UMKM Ditolak

The screenshot shows a list of rejected businesses. The columns are Nama Umkm, Pemilik, Alamat, Tanggal Pembuatan, and Action. The first entry is 'rfgfdg' owned by 'owner owner2' at 'hfghfr' with a creation date of '01-09-2023 / 17:05'. The second entry is 'Sanjai Putri' owned by 'owner owner2' at 'jin bulakan balai kandi no 10' with a creation date of '08-08-2023 / 16:16'. A search bar and a 'Show 10 entries' button are at the top.

Nama Umkm	Pemilik	Alamat	Tanggal Pembuatan	Action
rfgfdg	owner owner2	hfghfr	01-09-2023 / 17:05	
Sanjai Putri	owner owner2	jin bulakan balai kandi no 10	08-08-2023 / 16:16	

## 13. Implementasi Antarmuka UMKM Terkonfirmasi

The screenshot shows a list of confirmed active businesses. The columns are Nama Umkm, Pemilik, Alamat, and Action. The entries are: 'Sanjai 3 Saudara' owned by 'Afifah' at 'Jalan Imam Bonjol No. 26 Kelurahan Bulakan Balai Kandih Payakumbuh Barat'; 'Sanjai Anna' owned by 'Annisa Herina' at 'Jln Imam Bonjol'; 'Sanjai Erina' owned by 'Ningsih Eka Putri' at 'Jln Imam bonjol. Bulakan balai kandih'; 'Sanjai Nirina' owned by 'Nirina' at 'Jl. Imam Bonjol, Bulakan Balai Kandih, Kec. Payakumbuh Bar., 26223'; and 'Sanjai Oviosa' owned by 'Vina Kurnia' at 'Jl. Aceh, Taniuno. Pauh. Kec. Pavakumbuh Bar., 26222'. A search bar and a 'Show 10 entries' button are at the top.

Nama Umkm	Pemilik	Alamat	Action
Sanjai 3 Saudara	Afifah	Jalan Imam Bonjol No. 26 Kelurahan Bulakan Balai Kandih Payakumbuh Barat	
Sanjai Anna	Annisa Herina	Jln Imam Bonjol	
Sanjai Erina	Ningsih Eka Putri	Jln Imam bonjol. Bulakan balai kandih	
Sanjai Nirina	Nirina	Jl. Imam Bonjol, Bulakan Balai Kandih, Kec. Payakumbuh Bar., 26223	
Sanjai Oviosa	Vina Kurnia	Jl. Aceh, Taniuno. Pauh. Kec. Pavakumbuh Bar., 26222	

## 14. Implementasi Antarmuka UMKM di Banned

The screenshot shows a list of banned businesses. The columns are Nama Umkm, Pemilik, Alamat, Tanggal Pembuatan, and Action. The single entry is 'sanjai putri 2' owned by 'fadila annisa' at 'jln imam bonjol no 30' with a creation date of '21-10-2023 / 16:39'. A search bar and a 'Show 10 entries' button are at the top.

Nama Umkm	Pemilik	Alamat	Tanggal Pembuatan	Action
sanjai putri 2	fadila annisa	jln imam bonjol no 30	21-10-2023 / 16:39	

## 15. Implementasi Antarmuka Admin Menerima Notifikasi



## 16. Implementasi Antarmuka Halaman *Detail UMKM* pada Pembeli

The screenshot shows a user profile for 'Sanjai Oviga' with a photo of a shop building. It includes an address ('Payakumbuh, Jl. Aceh, Tanjung Pauh, Kec. Payakumbuh Bar., 26222'), information ('Sanjai oviga merupakan rumah produksi olahan makanan yang menghasilkan beraneka jenis sanjai serta kripik olahan dari ubi talas dan ketang. Sanjai ini sudah berdiri sejak tahun 1996'), and a phone number ('082285193080'). Below this is a 'Featured Product' section with four items: 'Sanjai Kuning Bumbu' (Rp 10,000,00 - Rp 10,000,00), 'Kripik Sanjai Balado' (Rp 10,000,00 - Rp 10,000,00), 'Kripik Lidi Balado' (Rp 10,000,00 - Rp 10,000,00), and 'Kripik Sanjai Balado Hijau' (Rp 10,000,00 - Rp 10,000,00). A search bar at the top of the page contains the placeholder 'What do you need?'.

## 17. Implementasi Antarmuka Halaman *Detail Produk*

The screenshot shows a product detail page for 'Jagung' (Corn). It features a large image of a bag of corn. The product title is 'Jagung', from 'Sanjai Oviga', with a price range of 'Rp 8.000,00 - Rp 12.000,00'. A note says 'Terbuat dari jagung'. Below this are sections for 'Kategori' (Produk Jagung) and 'Stok' (13 - 15). An 'Available Sizes' section offers 'kecil' and 'sedang' sizes. A quantity selector shows '1' with minus and plus buttons. A green 'ADD TO CART' button is at the bottom.

## 18. Implementasi Antarmuka Halaman Keranjang

The screenshot shows a shopping cart interface with a header featuring a logo and navigation links: HOME, PURCHASE, HISTORY, CHAT, PROFIL, and a shopping cart icon with a notification count of 2.

Products	Price	Quantity	Total
Jagung	Rp8000,00	- 2 +	Rp16000,00
Kripik Lidi Balado	Rp10000,00	- 1 +	Rp10000,00

Buttons at the bottom include "CONTINUE SHOPPING" and "UPDATE CART". A "Cart Total" summary box shows Subtotal Rp26000,00 and a green "PROCEED TO CHECKOUT" button.

## 19. Implementasi Antarmuka Halaman *Checkout*

The screenshot shows a checkout interface with a header featuring a logo and navigation links: HOME, PURCHASE, HISTORY, CHAT, PROFIL, and a user profile icon for Fadhillah Aini.

**Billing Details**

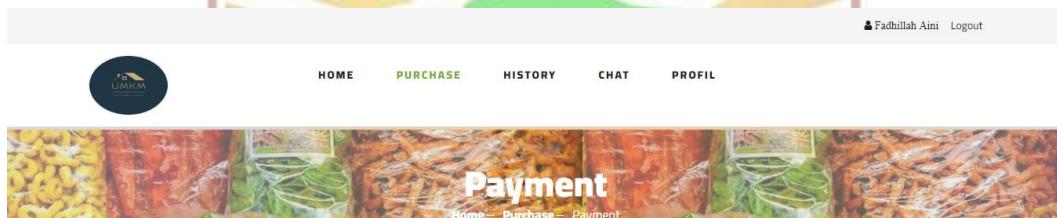
First Name*	Last Name*
Fadhillah	Aini
No Handphone*	Email*

**Your Order**

Products	Total
Jagung	Rp16000,00

081365913466	yxojxodxx@gmail.com																		
Province*	Sumatera Barat																		
City*	Padang																		
Address*	Jln Minahasa I, Jati, Padang timur																		
Pilih Ekspedisi*	Pilih Ekspedisi																		
<input type="button" value="Submit"/>																			
<table border="1"> <tr> <td>Kripik Lidi Balado</td> <td>Rp10000,00</td> </tr> <tr> <td><b>Subtotal</b></td> <td><b>Rp26000,00</b></td> </tr> <tr> <td colspan="2">Pilih Layanan*</td> </tr> <tr> <td colspan="2">REG   2-3   33000</td> </tr> <tr> <td colspan="2">Pilih Metode Pembayaran*</td> </tr> <tr> <td colspan="2"> <input checked="" type="radio"/> Bank Mandiri (1310007374443)  <input type="radio"/> Bank BRI (2115007273114)         </td> </tr> <tr> <td colspan="2"><b>Biaya Pengiriman</b> Rp.33000,-</td> </tr> <tr> <td colspan="2"><b>Total</b> Rp59000,00</td> </tr> <tr> <td colspan="2"> <input type="button" value="PLACE ORDER"/> </td> </tr> </table>		Kripik Lidi Balado	Rp10000,00	<b>Subtotal</b>	<b>Rp26000,00</b>	Pilih Layanan*		REG   2-3   33000		Pilih Metode Pembayaran*		<input checked="" type="radio"/> Bank Mandiri (1310007374443) <input type="radio"/> Bank BRI (2115007273114)		<b>Biaya Pengiriman</b> Rp.33000,-		<b>Total</b> Rp59000,00		<input type="button" value="PLACE ORDER"/>	
Kripik Lidi Balado	Rp10000,00																		
<b>Subtotal</b>	<b>Rp26000,00</b>																		
Pilih Layanan*																			
REG   2-3   33000																			
Pilih Metode Pembayaran*																			
<input checked="" type="radio"/> Bank Mandiri (1310007374443) <input type="radio"/> Bank BRI (2115007273114)																			
<b>Biaya Pengiriman</b> Rp.33000,-																			
<b>Total</b> Rp59000,00																			
<input type="button" value="PLACE ORDER"/>																			

## 20. Implementasi Antarmuka Halaman Pembayaran



UNIVERSITAS ANDALAS

Fadhillah Aini Logout

HOME PURCHASE HISTORY CHAT PROFIL

Payment

Home Purchase Payment

Terima kasih telah melakukan pembelian.

Waktu yang tersisa untuk melakukan pembayaran 1h 29m 54s

Ringkasan order

No	Gambar	Nama Produk	Jumlah	Harga Produk	Subtotal
1		Jagung	2	Rp8.000,00	Rp16.000,00
2		Kripik Lidi Balado	1	Rp10.000,00	Rp10.000,00

Biaya Pengiriman: Rp33.000,00

Total Harga: Rp59.000,00

Silakan melakukan pembayaran ke rekening berikut:

Bank Mandiri (1310007374443)

Bukti Transfer

Choose File No file chosen

## 21. Implementasi Antarmuka Halaman *Purchase*

The screenshot shows a web application interface for managing purchases. At the top, there is a navigation bar with links for HOME, PURCHASE, HISTORY, CHAT, and PROFIL. The PURCHASE link is highlighted. Below the navigation is a decorative banner featuring various food items like fruits and vegetables. The main content area is titled "Purchase" and shows a table with the following data:

No	Nama Ukm	Total Harga	Status	Aksi
1	Sanjai Oviga	59000	Menunggu Pembayaran	<a href="#">Bayar</a> <a href="#">Delete</a>

## 22. Implementasi Antarmuka Halaman *History*

The screenshot shows a web application interface for managing history. At the top, there is a navigation bar with links for HOME, PURCHASE, HISTORY, CHAT, and PROFIL. The HISTORY link is highlighted. Below the navigation is a decorative banner featuring various food items like fruits and vegetables. The main content area is titled "History" and shows a table with the following data:

No	Nama Ukm	Total Harga	Status	Aksi
1	Sanjai Oviga	Rp59000,00	Menunggu Konfirmasi	<a href="#">Detail Transaksi</a>

## 23. Implementasi Antarmuka Halaman *Detail* Transaksi Pembeli

Terima kasih telah melakukan pembelian.

Ringkasan order

Ukm : Sanjai Oviga

Tanggal Pembelian : 01-09-2023

status : Menunggu Konfirmasi

No	Gambar	Nama Produk	Jumlah	Harga Produk	Subtotal
1		Jagung	2	Rp8.000,00	Rp16.000,00
2		Kripik Lidi Balado	1	Rp10.000,00	Rp10.000,00

## 24. Implementasi Antarmuka Halaman *Chat* Pembeli

The screenshot displays a user interface for managing small and medium enterprises (UMKM). At the top, there are navigation tabs: HOME, PURCHASE, HISTORY, CHAT, and PROFIL. A sidebar on the left lists various product categories such as Keripik Olahan Singkong, Keripik Olahan Talas, Keripik Tempe, Snack Pisang, Produk Jagung, Kripik Olahan Ubi Jalar, and Snack Olahan Tepung. The main content area shows a profile for "Sanjai Oviga" with a photo, address (Payakumbuh, Jl. Aceh, Tanjung, Pa), and bio mentioning she started in 1996. Below this is a map from Google Maps showing the location. A chat window is open with a message "hallo". The bottom part of the interface shows a recent messages section with a message from "Vina Kurnia" at 17:41 PM on Sep 01, 2023.

## 25. Implementasi Antarmuka Halaman Daftar UMKM

This screenshot shows a dashboard for managing UMKM data. On the left, a sidebar includes links for Dashboard, UMKM (which is selected and highlighted in blue), Transaksi, History, Laporan, and Chat. The main content area is titled "List Data Ukmkm" and shows two entries for "Sanjai Oviga". The first entry is labeled "Pending" and the second is labeled "Aktif". Both entries provide contact information (Address: Jl. Aceh, Tanjung Pauh, 26222; Phone: 081265663488 and 082285193080) and a "View Ukmkm" button. A "Tambah Ukmkm" (Add Ukmkm) button is located in the top right corner of the list area.

## 26. Implementasi Antarmuka Halaman Tambah UMKM

**Menambahkan Data UMKM**

**Informasi Umum**

Nama Ukmk  
Input your umkm's name

No Handphone  
Input your umkm's phone number

Informasi Tambahan  
Input your umkm's information

Upload Surat Izin Usaha  
\*boleh HKI Merek atau SIUP  
Upload Foto Ukmk

Upload Izin Usaha  
Upload Foto

Next >

## 27. Implementasi Antarmuka Halaman *Detail UMKM*

**Sanjai Oviga**

Lihat Surat Izin | Tambah Produk | Edit Ukmk | Hapus Ukmk

**Sanjai Oviga**  
Vina Kurnia 081275920305

Informasi Ukmk  
Sanjai oviga merupakan rumah produksi olahan makanan yang menghasilkan beraneka jenis sanjai serta kripik olahan dari ubi talas dan kentang. Sanjai ini sudah berdiri sejak tahun 1996

Alamat  
Jl. Aceh, Tanjung. Pauh, Kec. Payakumbuh Bar., 26222

Metode Pembayaran  
- Bank Mandiri ( 1310007374443 )  
- Bank BRI ( 2115007273114 )

Nama Produk	Kategori	Action
Ganepo	Keripik Olahan Singkong	
Jagung	Produk Jagung	
Karak Kaliang	Keripik Olahan Singkong	
Kripik Bunga Durian	Snack Olahan Tepung	
Kripik Gula Aren	Keripik Olahan Talas	
Kripik Lidi Balado	Keripik Olahan Singkong	
Kripik Pisang	Snack Pisang	
Kripik Sanjai Balado Hijau	Keripik Olahan Singkong	
Kripik Sanjai Kuning	Keripik Olahan Singkong	

Showing 1 to 10 of 16 entries

Previous | 1 | 2 | Next

## 28. Implementasi Antarmuka Halaman Edit UMKM

The screenshot shows a user interface for managing small businesses. On the left is a sidebar with navigation links: Dashboard, UMKM (selected), Transaksi, History, Laporan, and Chat. The main area has a title 'Mengubah Data UMKM'. A modal window is open, divided into three tabs: 'Informasi Umkm' (selected), 'Alamat', and 'Pembayaran'. The 'Informasi Umkm' tab contains fields for 'Nama Ukm' (Sanjai Oviga) and 'No Handphone' (082285193080). The 'Alamat' tab contains a text area with a note about Sanjai Oviga being a food production company that produces various types of sanjai and kripik. The 'Pembayaran' tab shows a file upload section with a preview of a document titled 'SURAT IZIN USAHA PERDAGANGAN (SIUP) KECIL' from the 'PEMERINTAH KABUPATEN BADAN PERIZINAN TERPADU' dated 1323. There is also a 'Change' button and a delete icon.

## 29. Implementasi Antarmuka Halaman Hapus UMKM

The screenshot shows a confirmation dialog box with a large orange exclamation mark icon. The text reads 'Apakah kamu yakin?' (Are you sure?). Below it, a message states 'Data ini akan terhapus total' (This data will be deleted entirely). To the right, there is a note: 'Sanjai Oviga merupakan rumah produksi olahan makanan yang menghasilkan beraneka jenis sanjai serta kripik olahan yang sudah berdiri sejak tahun 1996'. At the bottom are two buttons: 'Tidak, batalkan!' (No, cancel!) and a red 'Ya, Hapus Data!' (Yes, delete data!).

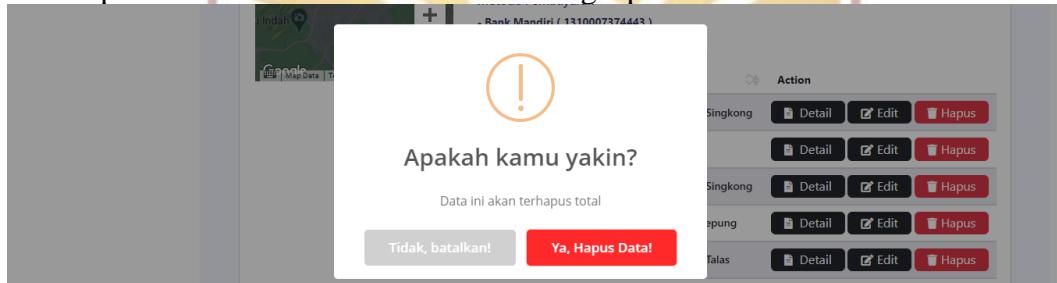
## 30. Implementasi Antarmuka Halaman Tambah Produk

The screenshot shows a user interface for adding products. On the left is a sidebar with navigation links: Dashboard, UMKM (selected), Transaksi, History, Laporan, and Chat. The main area has a title 'Menambahkan Data Produk'. A modal window is open, divided into three tabs: 'Informasi Umum Produk' (selected), 'Rincian Produk', and 'Gambar Produk'. The 'Informasi Umum Produk' tab contains fields for 'Nama Produk' (Input your produk name), 'Kategori Produk' (Manis, a dropdown menu), 'No Izin P-IRT' (Input your p\_irt), and 'Deskripsi' (Input your product's description). A 'Next >' button is at the bottom right of the modal.

### 31. Implementasi Antarmuka Halaman Mengedit Produk

The screenshot shows a modal window titled 'Mengedit Data Produk'. The window is divided into three tabs: 'Informasi Umum Produk' (selected), 'Rincian Produk', and 'Gambar Produk'. The 'Informasi Umum Produk' tab contains fields for 'Nama Produk' (Ganepo), 'Kategori Produk' (Keripik Olahan Singkong), 'No Izin P-IRT' (2.15.1376.07.0351.25), and 'Deskripsi' (Terbuat dari singkong. Gurah dan renyah.). A 'Next >' button is visible at the bottom right of the modal.

### 32. Implementasi Antarmuka Halaman Menghapus Produk



### 33. Implementasi Antarmuka Halaman Detail Produk

The screenshot shows a product detail page for 'Jagung'. The top navigation bar includes 'UMKM Sanji' and a search bar. The main content area displays the product name 'Jagung', category 'Produk Jagung', P-IRT number '2.15.1376.16.0351.25', description 'Terbuat dari jagung', packaging type ('kecil' is selected), product code '1018', stock '15', and price '8000'. A small image of a bag of corn is shown. A 'Tambah Stok' (Add Stock) button is located at the top right of the content area.

### 34. Implementasi Antarmuka Halaman Tambah Stok

Jagung

Tambah Stok

Pilih Kemasan

sedang

Jumlah Stok

25

Batal Simpan

### 35. Implementasi Antarmuka Halaman Transaksi

Transaksi Pembelian

Show 10 entries	Search:			
Tanggal	Umkm	Total Pembelian	Status	Action
02-09-2023	Sanjai Oviga	18000	Selesai	<i>Detail</i>
06-09-2023	Sanjai Oviga	59000	Selesai	<i>Detail</i>
08-09-2023	Sanjai Oviga	49500	Selesai	<i>Detail</i>
09-09-2023	Sanjai Oviga	21000	Expired	<i>Detail</i>
10-09-2023	Sanjai Oviga	26000	Selesai	<i>Detail</i>
10-09-2023	Sanjai Oviga	42000	Selesai	<i>Detail</i>

### 36. Implementasi Antarmuka Halaman Detail Transaksi Penjual

Detail Transaksi Pemesanan

Fadhillah Aini  
10-09-2023

Bank BRI  
Transaksi Selesai

Bukti Pembayaran

Kode Pemesanan : 20230910172412  
Norek : jp9667564  
Lihat Bukti Pengiriman : .png

Detail Pemesanan

No	Nama Produk	Kemasan	Jumlah	Harga Produk	Subtotal
1	Kue Bawang	250 gr	1	Rp 15.000	Rp 15.000

### 37. Implementasi Antarmuka Halaman History

The screenshot shows a web application interface for managing small business (UMKM) transactions. On the left sidebar, there are several menu items: Dashboard, UMKM, Transaksi, History (which is highlighted in blue), Laporan, and Chat. The main content area is titled "History Transaksi Pembelian". It displays a table with three entries, each representing a purchase transaction. The columns in the table are Tanggal (Date), Umkm (Business), Nama Pembeli (Buyer Name), Total Pembelian (Total Purchase), and Action (Detail). The transactions listed are from 01-09-2023, 24-08-2023, and 25-08-2023.

### 38. Implementasi Antarmuka Halaman Detail History

This screenshot shows the detailed view of a specific transaction. The title is "Detail History Transaksi Pemesanan". It includes a header with the buyer's name (Dhea Irma Putri), date (25-08-2023), payment method (Bank BRI), and status (Menunggu Konfirmasi). Below this, there is a section titled "Bukti Pembayaran" showing a screenshot of a payment confirmation screen. The main content area shows the "Detail Pemesanan" for the transaction. It lists one item: "Stik Ubi Jalar" (250 gr) at a price of Rp 12.000. The total amount is Rp 12.000, and the shipping cost is Rp 42.000.

### 39. Implementasi Antarmuka Halaman Laporan

This screenshot shows the "Cetak Laporan Transaksi" (Print Transaction Report) page. It has two main input fields: "Select a State" (likely referring to a dropdown menu) and a date range selector labeled "09/01/2023 - 09/01/2023". Below these fields is a blue "View" button. The rest of the page is mostly empty, indicating the report is yet to be generated or displayed.

#### 40. Implementasi Antarmuka Halaman View Laporan

**Cetak Laporan Transaksi**

Nama Umkm : Sanjai Oviga  
Transaksi pada : 08/07/2023 - 08/31/2023

No	Tanggal pembelian	Pembeli	Produk	Kemasan	Jumlah	Harga	Subtotal	
1	24/08/2023	customer cust	Jagung	sedang	2	Rp 12.000	Rp 24.000	
							Biaya Pengiriman:	Rp 33.000
							Total Harga	Rp 57.000
2	25/08/2023	Dhea Irma Putri	Stik Ubi Jalar	250 gr	1	Rp 12.000	Rp 12.000	
							Biaya Pengiriman:	Rp 42.000
							Total Harga	Rp 54.000
							<b>Total Pemasukan</b>	<b>Rp. 111.000,00</b>

**Cetak**

#### 41. Implementasi Antarmuka Halaman Cetak Laporan

**Laporan Transaksi Penjualan Produk**

Nama Umkm : Sanjai Oviga  
Transaksi pada : 08/07/2023 - 08/31/2023

No	Tanggal pembelian	Pembeli	Produk	Kemasan	Jumlah	Harga	Subtotal	
1	24/08/2023	customer cust	Jagung	sedang	2	Rp 12.000	Rp 24.000	
							Biaya Pengiriman:	Rp 33.000
							Total Harga	Rp 57.000
2	25/08/2023	Dhea Irma Putri	Stik Ubi Jalar	250 gr	1	Rp 12.000	Rp 12.000	
							Biaya Pengiriman:	Rp 42.000
							Total Harga	Rp 54.000
							<b>Total Pemasukan</b>	<b>Rp. 111.000,00</b>

#### 42. Implementasi Antarmuka Halaman Chat

**Fadhillah**

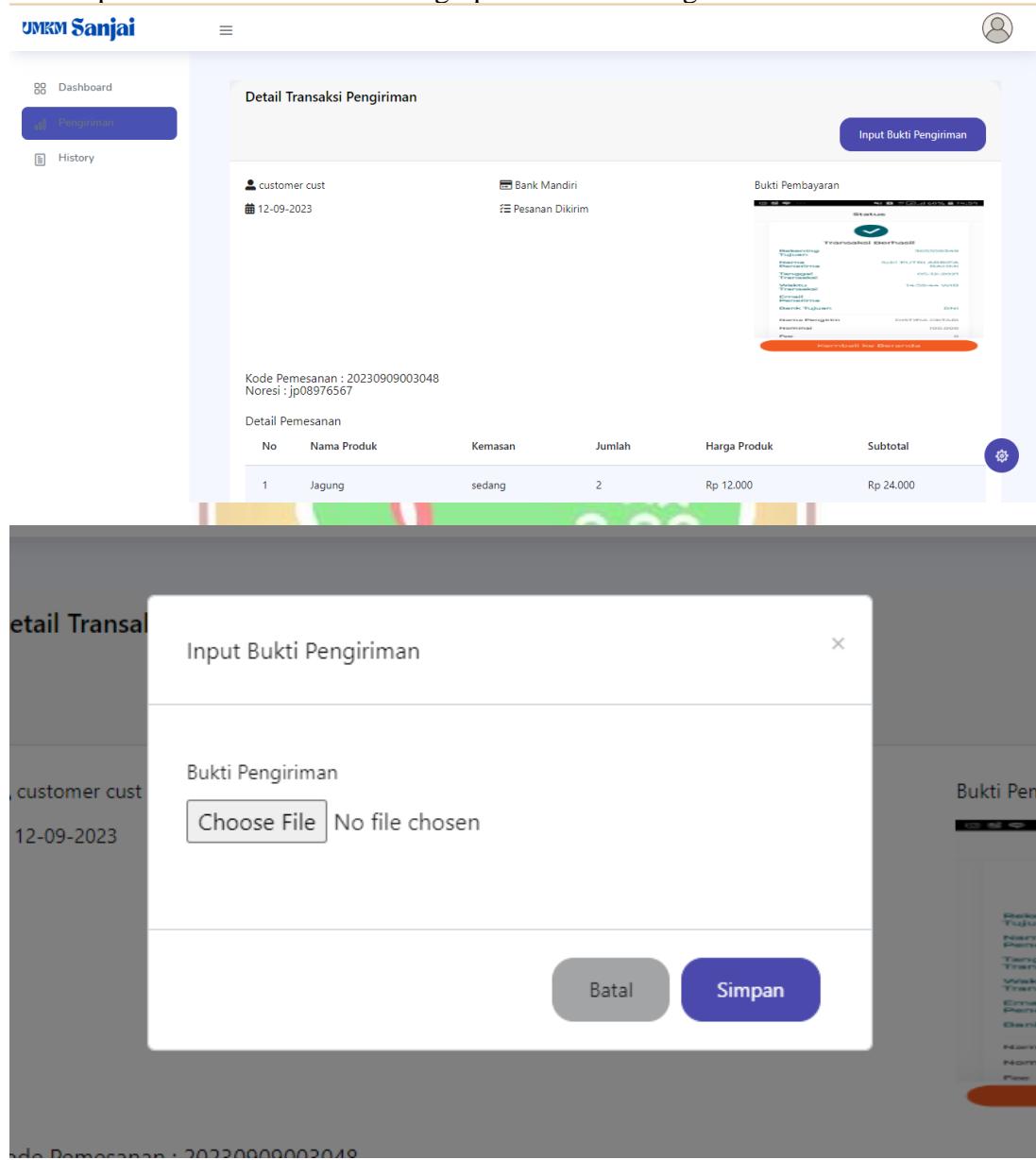
Recent

	Fadhillah Aini	17:41 PM, 01 Sep 2023
	hallo	
	Dhea Irma Putri	12:03 PM, 25 Aug 2023
	sanjainya kakak	

#### 43. Implementasi Antarmuka Menampilkan Notifikasi



#### 44. Implementasi Antarmuka Menginputkan Bukti Pengiriman



#### 45. Implementasi Antarmuka *History* Kurir

UMKM Sanjai

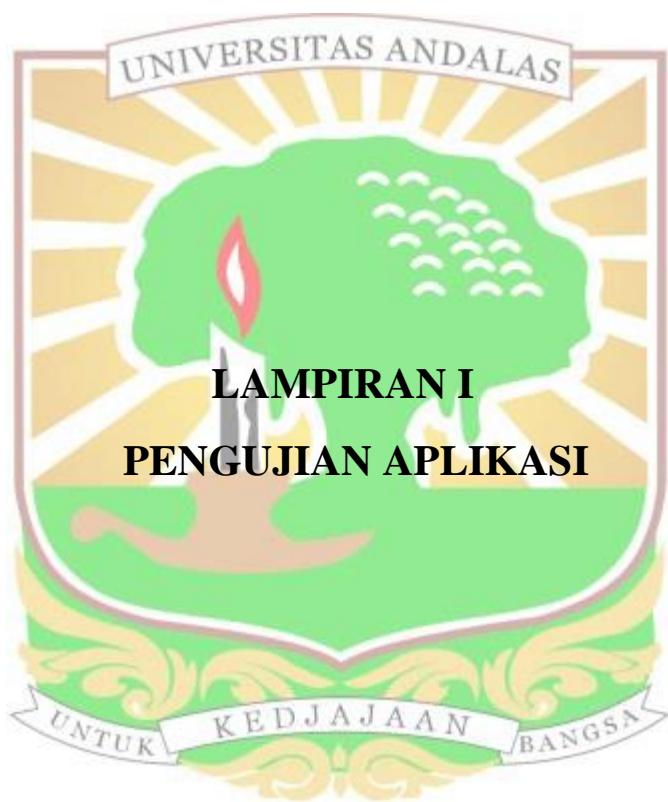
History Pengiriman

Show	10	entries	Search:						
Tanggal	2023-09-12	Umkm	Sanjai Oviga	Total Pembelian	49500	Noresi	jp08976567	Action	Detail

Showing 1 to 1 of 1 entries

Previous 1 Next





### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Dwi Astuti Octavia

Hak Akses / Jabatan : Admin

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama admin	✓	
2.	Kelola data user	Melihat data user	Aplikasi menampilkan halaman list data user	✓	
		Menambah data user	Data user berhasil ditambah dan aplikasi menampilkan halaman list data user	✓	
		Mengedit data user	Data user berhasil diedit dan aplikasi menampilkan halaman list data user	✓	
		Menghapus data user	Data user berhasil dihapus dan aplikasi menampilkan halaman list data user	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Kelola data kategori produk	Melihat list data kategori produk dengan mengklik menu "kategori"	Aplikasi menampilkan halaman list data kategori	✓	
		Menambah data kategori produk	Data kategori berhasil ditambahkan dan aplikasi menampilkan halaman list kategori	✓	
		Mengedit data	Data kategori berhasil	✓	

		kategori	diedit dan aplikasi menampilkan halaman list data kategori	✓	
		Menghapus data kategori	Data kategori berhasil dihapus dan aplikasi menampilkan halaman list data kategori	✓	
6.	Kelola perizinan umkm	Melihat list umkm yang mengajukan permohonan izin	Aplikasi menampilkan halaman list umkm yang telah mendaftar pada aplikasi.	✓	
		Menerima permohonan umkm	Permohonan pendaftaran umkm telah diterima dan umkm telah aktif	✓	
		Menolak permohonan umkm	Permohonan pendaftaran umkm ditolak dengan catatan alasan tolak dari admin	✓	
7.	Menerima notifikasi	Mengklik ikon notifikasi	Menampilkan notifikasi umkm yang mendaftar dan memerlukan konfirmasi	✓	
8.	Logout	Mengklik tombol logout	Admin berhasil keluar dari aplikasi	✓	

Padang, 20 Agustus 2023

Penguji



( Dhistiaq Octari )

### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Janan (Putri Chantika)

Hak Akses / Jabatan : Pihak UMKM (Penjual)

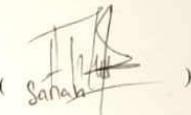
No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama penjual	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama penjual	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Kelola data umkm	Melihat data umkm	Aplikasi menampilkan halaman list data umkm	✓	
		Menambah data atau melakukan registrasi umkm pada aplikasi	Data umkm berhasil ditambah dan aplikasi menampilkan halaman list data umkm serta menunggu konfirmasi perizinan umkm dari admin	✓	
		Mengedit data umkm	Data umkm berhasil diedit dan aplikasi menampilkan halaman detail umkm	✓	
		Menghapus data umkm	Data umkm berhasil dihapus dan aplikasi menampilkan halaman list data umkm	✓	
6.	Kelola data produk	Melihat data produk	Aplikasi menampilkan halaman list data	✓	

			produk		
		Menambah data produk	Data produk berhasil ditambah dan aplikasi menampilkan halaman list data produk	✓	
		Mengedit data produk	Data produk berhasil diedit dan aplikasi menampilkan halaman list data produk	✓	
		Menghapus data produk	Data produk berhasil dihapus dan aplikasi menampilkan halaman list data produk	✓	
7.	Menambah stok produk	Melakukan penambahan stok produk	Stok produk berhasil ditambahkan	✓	
8.	Kelola transaksi pembelian	Melihat list pembelian produk	Aplikasi menampilkan halaman list pembelian produk	✓	
		Konfirmasi terima pembelian produk	Berhasil mengkonfirmasi pesanan dan aplikasi menampilkan list pembelian produk	✓	
		Tolak pembelian produk	Berhasil menolak pembelian dengan menginputkan alasan penolakan	✓	
		Menginputkan noresi	Berhasil menginputkan noresi pembelian yang telah diterima pembeliannya dan menampilkan halaman detail transaksi	✓	
		Menghapus pembelian	Pembelian berhasil dihapus dan aplikasi menampilkan halaman list pembelian produk	✓	
9.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pemilik umkm dan pembeli	✓	
10.	Melihat laporan transaksi pembelian	Melihat laporan transaksi pembelian	Aplikasi menampilkan halaman laporan transaksi pembelian	✓	
		Mencetak laporan transaksi pembelian dengan mengklik tombol cetak	Cetak laporan berhasil dan tersimpan dalam bentuk pdf	✓	
11.	Melihat riwayat	Melihat riwayat	Aplikasi berhasil		

	pembelian	pembelian produk	menampilkan halaman riwayat pembelian produk	✓	
12.	Menerima notifikasi pembelian produk	Mengklik ikon notifikasi	Menampilkan notifikasi pembelian produk dan memerlukan konfirmasi	✓	
13.	Logout	Mengklik tombol logout	Penjual berhasil keluar dari aplikasi	✓	

Padang, 2 Agustus 2023

Pengujii

(  )

### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Vina Firda (Sanjai Dviga)

Hak Akses / Jabatan : Pihak UMKM (Penjual)

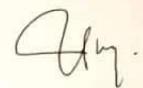
No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama penjual	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama penjual	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Kelola data umkm	Melihat data umkm	Aplikasi menampilkan halaman list data umkm	✓	
		Menambah data atau melakukan registrasi umkm pada aplikasi	Data umkm berhasil ditambah dan aplikasi menampilkan halaman list data umkm serta menunggu konfirmasi perizinan umkm dari admin	✓	
		Mengedit data umkm	Data umkm berhasil diedit dan aplikasi menampilkan halaman detail umkm	✓	
		Menghapus data umkm	Data umkm berhasil dihapus dan aplikasi menampilkan halaman list data umkm	✓	
6.	Kelola data produk	Melihat data produk	Aplikasi menampilkan halaman list data	✓	

		produk		
	Menambah data produk	Data produk berhasil ditambah dan aplikasi menampilkan halaman list data produk	✓	
	Mengedit data produk	Data produk berhasil diedit dan aplikasi menampilkan halaman list data produk	✓	
	Menghapus data produk	Data produk berhasil dihapus dan aplikasi menampilkan halaman list data produk	✓	
7.	Menambah stok produk	Melakukan penambahan stok produk	Stok produk berhasil ditambahkan	✓
8.	Kelola transaksi pembelian	Melihat list pembelian produk	Aplikasi menampilkan halaman list pembelian produk	✓
		Konfirmasi terima pembelian produk	Berhasil mengkonfirmasi pesanan dan aplikasi menampilkan list pembelian produk	✓
		Tolak pembelian produk	Berhasil menolak pembelian dengan menginputkan alasan penolakan	✓
		Menginputkan noresi	Berhasil menginputkan noresi pembelian yang telah diterima pembeliannya dan menampilkan halaman detail transaksi	✓
		Menghapus pembelian	Pembelian berhasil dihapus dan aplikasi menampilkan halaman list pembelian produk	✓
9.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pemilik umkm dan pembeli	✓
10.	Melihat laporan transaksi pembelian	Melihat laporan transaksi pembelian	Aplikasi menampilkan halaman laporan transaksi pembelian	✓
		Mencetak laporan transaksi pembelian dengan mengklik tombol cetak	Cetak laporan berhasil dan tersimpan dalam bentuk pdf	✓
11.	Melihat riwayat	Melihat riwayat	Aplikasi berhasil	

	pembelian	pembelian produk	menampilkan halaman riwayat pembelian produk	<input checked="" type="checkbox"/>	
12.	Menerima notifikasi pembelian produk	Mengklik ikon notifikasi	Menampilkan notifikasi pembelian produk dan memerlukan konfirmasi	<input checked="" type="checkbox"/>	
13.	Logout	Mengklik tombol logout	Penjual berhasil keluar dari aplikasi	<input checked="" type="checkbox"/>	

Padang, 21 Agustus 2023

Pengujji



(Vina Eufenia )

**LEMBARAN PENGUJIAN APLIKASI**

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Fadhillah Aini

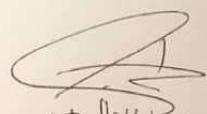
Hak Akses / Jabatan : Pembeli

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama pembeli	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama pembeli	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Melakukan penelusuran	Melakukan pencarian umkm	Aplikasi berhasil menampilkan list produk umkm berdasarkan pencarian	✓	
		Melakukan pencarian produk	Aplikasi berhasil menampilkan list produk berdasarkan pencarian	✓	
		Melakukan penyortiran kategori	Aplikasi berhasil menampilkan list produk berdasarkan kategori yang dipilih	✓	
6.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pembeli dan pemilik umkm	✓	
7.	Melakukan pembelian produk	Melihat detail informasi produk	Aplikasi menampilkan halaman detail informasi produk	✓	

		Menambahkan produk ke keranjang	Produk berhasil ditambahkan ke keranjang dengan menginputkan varian produk dengan stok yang tersedia	✓	
		Melakukan checkout	Aplikasi menampilkan halaman checkout	✓	
		Melakukan penginputan data alamat (jika ingin diubah) dan memilih ekspedisi pengiriman	Aplikasi berhasil menampilkan ringkasan pesanan dengan total harga	✓	
		Melakukan pembelian	Pembelian berhasil dilakukan dan aplikasi menampilkan halaman pembayaran	✓	
		Melakukan penginputan bukti transfer pembayaran	Pembayaran berhasil dan aplikasi menampilkan halaman ringkasan pesanan	✓	
8.	Melihat list pembelian yang belum dibayar	Melihat detail pembelian yang belum dibayar	Aplikasi menampilkan halaman ringkasan pesanan dan form penginputan bukti transfer	✓	
		Menghapus pembelian yang belum dibayar	Aplikasi berhasil menghapus pembelian yang belum dibayar	✓	
9.	Melihat riwayat pembelian	Melihat riwayat pembelian produk	Aplikasi berhasil menampilkan halaman riwayat pembelian produk	✓	
10.	Logout	Mengklik tombol logout	Pembeli berhasil keluar dari aplikasi	✓	

Padang, 1 Agustus 2023

Penguji



(Fadillah Azni)

### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Dhea Irma Putri

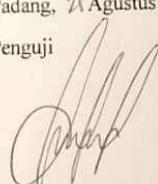
Hak Akses / Jabatan : Pembeli

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama pembeli	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama pembeli	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Melakukan penelusuran	Melakukan pencarian umkm	Aplikasi berhasil menampilkan list produk umkm berdasarkan pencarian	✓	
		Melakukan pencarian produk	Aplikasi berhasil menampilkan list produk berdasarkan pencarian	✓	
		Melakukan penyortiran kategori	Aplikasi berhasil menampilkan list produk berdasarkan kategori yang dipilih	✓	
6.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pembeli dan pemilik umkm	✓	
7.	Melakukan pembelian produk	Melihat detail informasi produk	Aplikasi menampilkan halaman detail informasi produk	✓	

		Menambahkan produk ke keranjang	Produk berhasil ditambahkan ke keranjang dengan menginputkan varian produk dengan stok yang tersedia	✓	
		Melakukan checkout	Aplikasi menampilkan halaman checkout	✓	
		Melakukan penginputan data alamat (jika ingin diubah) dan memilih ekspedisi pengiriman	Aplikasi berhasil menampilkan ringkasan pesanan dengan total harga	✓	
		Melakukan pembelian	Pembelian berhasil dilakukan dan aplikasi menampilkan halaman pembayaran	✓	
		Melakukan penginputan bukti transfer pembayaran	Pembayaran berhasil dan aplikasi menampilkan halaman ringkasan pesanan	✓	
8.	Melihat list pembelian yang belum dibayar	Melihat detail pembelian yang belum dibayar	Aplikasi menampilkan halaman ringkasan pesanan dan form penginputan bukti transfer	✓	
		Menghapus pembelian yang belum dibayar	Aplikasi berhasil menghapus pembelian yang belum dibayar	✓	
9.	Melihat riwayat pembelian	Melihat riwayat pembelian produk	Aplikasi berhasil menampilkan halaman riwayat pembelian produk	✓	
10.	Logout	Mengklik tombol logout	Pembeli berhasil keluar dari aplikasi	✓	

Padang, 26 Agustus 2023

Pengujii



( Dhea Irma Putri )

### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Annisha Apriyayd

Hak Akses / Jabatan : Pembeli

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama pembeli	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama pembeli	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Melakukan penelusuran	Melakukan pencarian umkm	Aplikasi berhasil menampilkan list <del>produk</del> umkm berdasarkan pencarian	✓	
		Melakukan pencarian produk	Aplikasi berhasil menampilkan list produk berdasarkan pencarian	✓	
		Melakukan penyortiran kategori	Aplikasi berhasil menampilkan list produk berdasarkan kategori yang dipilih	✓	
6.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pembeli dan pemilik umkm	✓	
7.	Melakukan pembelian produk	Melihat detail informasi produk	Aplikasi menampilkan halaman detail informasi produk	✓	

		Menambahkan produk ke keranjang	Produk berhasil ditambahkan ke keranjang dengan menginputkan varian produk dengan stok yang tersedia	✓	
		Melakukan checkout	Aplikasi menampilkan halaman checkout	✓	
		Melakukan penginputan data alamat (jika ingin diubah) dan memilih ekspedisi pengiriman	Aplikasi berhasil menampilkan ringkasan pesanan dengan total harga	✓	
		Melakukan pembelian	Pembelian berhasil dilakukan dan aplikasi menampilkan halaman pembayaran	✓	
		Melakukan penginputan bukti transfer pembayaran	Pembayaran berhasil dan aplikasi menampilkan halaman ringkasan pesanan	✓	
8.	Melihat list pembelian yang belum dibayar	Melihat detail pembelian yang belum dibayar	Aplikasi menampilkan halaman ringkasan pesanan dan form penginputan bukti transfer	✓	
		Menghapus pembelian yang belum dibayar	Aplikasi berhasil menghapus pembelian yang belum dibayar	✓	
9.	Melihat riwayat pembelian	Melihat riwayat pembelian produk	Aplikasi berhasil menampilkan halaman riwayat pembelian produk	✓	
10.	Logout	Mengklik tombol logout	Pembeli berhasil keluar dari aplikasi	✓	

Padang, 25 Agustus 2023

Penguji

(ANNISHA APRILLYA)

### LEMBARAN PENGUJIAN APLIKASI

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Lathif Nur Isyad

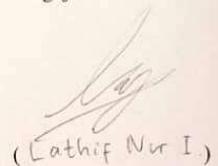
Hak Akses / Jabatan : Pembeli

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama pembeli	✓	
2.	Register	Menginputkan data yang diperlukan untuk registrasi	Pembeli berhasil register dan aplikasi menampilkan halaman utama pembeli	✓	
3.	Kelola data profil	Melihat profil	Aplikasi menampilkan halaman profil	✓	
		Mengubah / memperbarui data profil	Data profil berhasil diubah dan aplikasi menampilkan halaman profil	✓	
4.	Ubah password	Mengubah / memperbarui password	Data password berhasil diubah dan aplikasi menampilkan halaman profil	✓	
5.	Melakukan penelusuran	Melakukan pencarian umkm	Aplikasi berhasil menampilkan list produk umkm berdasarkan pencarian	✓	
		Melakukan pencarian produk	Aplikasi berhasil menampilkan list produk berdasarkan pencarian	✓	
		Melakukan penyortiran kategori	Aplikasi berhasil menampilkan list produk berdasarkan kategori yang dipilih	✓	
6.	Melakukan chat	Melakukan percakapan dengan pemilik umkm	Aplikasi menampilkan list daftar chat antara pembeli dan pemilik umkm	✓	
7.	Melakukan pembelian produk	Melihat detail informasi produk	Aplikasi menampilkan halaman detail informasi produk	✓	

		Menambahkan produk ke keranjang	Produk berhasil ditambahkan ke keranjang dengan menginputkan varian produk dengan stok yang tersedia	✓	
		Melakukan checkout	Aplikasi menampilkan halaman checkout	✓	
		Melakukan penginputan data alamat (jika ingin diubah) dan memilih ekspedisi pengiriman	Aplikasi berhasil menampilkan ringkasan pesanan dengan total harga	✓	
		Melakukan pembelian	Pembelian berhasil dilakukan dan aplikasi menampilkan halaman pembayaran	✓	
		Melakukan penginputan bukti transfer pembayaran	Pembayaran berhasil dan aplikasi menampilkan halaman ringkasan pesanan	✓	
8.	Melihat list pembelian yang belum dibayar	Melihat detail pembelian yang belum dibayar	Aplikasi menampilkan halaman ringkasan pesanan dan form penginputan bukti transfer	✓	
		Menghapus pembelian yang belum dibayar	Aplikasi berhasil menghapus pembelian yang belum dibayar	✓	
9.	Melihat riwayat pembelian	Melihat riwayat pembelian produk	Aplikasi berhasil menampilkan halaman riwayat pembelian produk	✓	
10.	Logout	Mengklik tombol logout	Pembeli berhasil keluar dari aplikasi	✓	

Padang, 13 Agustus 2023

Pengujii



(Lathif Nur I.)

**LEMBARAN PENGUJIAN APLIKASI**

Nama Aplikasi : Marketplace UMKM Sanjai di daerah Payakumbuh Barat

Nama Penguji : Heru Herianto

Hak Akses / Jabatan : Kurir

No	Fungsional	Skenario	Hasil yang Diharapkan	Hasil	
				Sesuai	Tidak Sesuai
1.	Login	Mengisi email dan password pada halaman login dengan benar	Aplikasi menampilkan halaman utama kurir	✓	
2.	Kelola pengiriman	Melihat data pengiriman	Aplikasi menampilkan halaman list data pengiriman	✓	
		Menginputkan bukti pengiriman	Data bukti pengiriman berhasil ditambahkan dan aplikasi menampilkan halaman detail pengiriman	✓	
3.	Melihat riwayat pengiriman	Melihat riwayat pengiriman	Aplikasi menampilkan halaman pengiriman yang telah selesai	✓	
		Melihat detail riwayat pengiriman	Aplikasi menampilkan halaman informasi detail pengiriman yang telah selesai	✓	
8.	Logout	Mengklik tombol logout	Kurir berhasil keluar dari aplikasi	✓	

Padang, 17 September 2023

Penguji



(Heru Herianto )

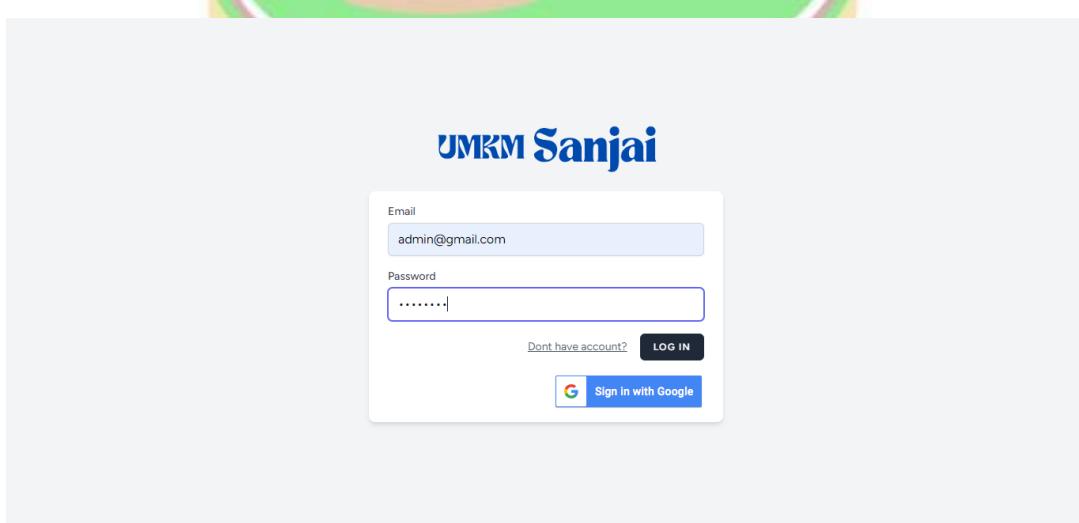


## 1. Login

Pengujian *login* ini dilakukan dengan menginputkan *email* dan *password* untuk masuk ke dalam aplikasi. Hasil pengujian *login* dapat dilihat pada tabel di bawah ini

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	<i>Email</i> dan <i>password</i>
Yang diharapkan	Sistem menampilkan halaman utama aplikasi sesuai dengan user yang sedang melakukan <i>login</i>
Pengamatan	Dapat menginput <i>email</i> dan <i>password</i> pada halaman <i>login</i> dan mengklik tombol “Login”
Hasil	Sesuai

Pengujian *login* ini dilakukan ketika *user* membuka aplikasi, lalu muncul halaman *landing page*, dan *user* mengklik tombol “*Login*” dan sistem menampilkan halaman *login*. Pada halaman ini *user* menginputkan *email* dan *password* dan mengklik tombol “*login*” lalu aplikasi akan menampilkan halaman utama sesuai dengan *role* dari *user* yang sedang *login*. Tampilan hasil pengujian *login* dapat dilihat pada gambar di bawah ini.



Tampilan Form *Login*

**UMKM Sanjai**

Dashboard

User Pages >

UMKM >

Kategori Produk

Welcome admin  
All systems are running smoothly!

Jumlah Owner UMKM  
17

UMKM Perlu Konfirmasi  
1

UMKM Beroperasi  
10

UMKM Ditolak  
2

Data UMKM

Nama UMKM	Handphone	Status
Sanjai Putri	8089878	Ditolak

UNIVERSITAS ANDALAS

Tampilan Halaman Utama Admin

**UMKM Sanjai**

Dashboard

UMKM

Transaksi

History

Laporan

Chat

Welcome Vina Kurnia  
All systems are running smoothly!

Jumlah UMKM Beroperasi  
1

Menunggu Konfirmasi  
2

Pesanan Disediakan  
0

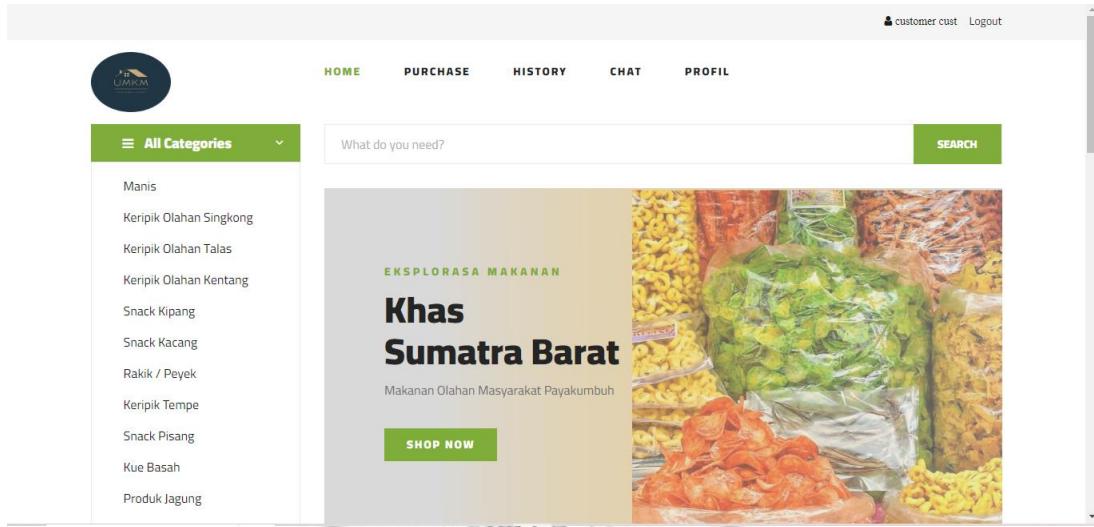
Pesanan Dikirim  
0

Pesanan Selesai  
2

Pesanan Ditolak  
0

UNTUK KEDAJAAN EGSA

Tampilan Halaman Utama Penjual



Tampilan Halaman Utama Pembeli

Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil melakukan *login* dan sistem menampilkan halaman utama sesuai *role user* yang melakukan *login*.

## 2. Register

Pengujian *register* ini dilakukan ketika *user* ingin mendaftarkan akunnya pada aplikasi. Hasil pengujian *register* dapat dilihat pada tabel di bawah ini.

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan dalam halaman halaman <i>register</i>
Yang diharapkan	Sistem menampilkan halaman utama aplikasi sesuai dengan <i>user</i> yang sedang <i>login</i>
Pengamatan	Dapat menginputkan data pada halaman <i>register</i> dan mengklik tombol “Register”
Hasil	Sesuai

Pengujian dilakukan dengan *user* membuka aplikasi sehingga tampil halaman *landing page*, kemudian user mengklik tombol “Register”. Pada halaman ini akan muncul form inputan data untuk melakukan *registrasi*, setelah itu *user* akan mengklik tombol “Register” dan aplikasi akan menampilkan halaman utama sesuai dengan *role*

*user* yang berhasil melakukan *register*. Tampilan hasil pengujian *register* dapat dilihat pada gambar di bawah ini.

The image shows two screenshots of a web application. The top screenshot displays a registration form titled "UMKM Sanjai". The fields filled in are: First Name (Hana), Last Name (Nisa), Email (ahana@gmail.com), Password (hidden), Confirm Password (hidden), and Level (Penjual). Below the form are links for "Already registered?" and a "REGISTER" button. The bottom screenshot shows the main dashboard for "UMKM Sanjai". It features a welcome message "Welcome Hana Nisa All systems are running smoothly!" above a central illustration of people walking outdoors. To the right are six cards with statistics: "Jumlah UMKM Beroperasi" (0), "Menunggu Konfirmasi" (0), "Pesanan Disiapkan" (0), "Pesanan Dikirim" (0), "Pesanan Selesai" (0), and "Pesanan Ditolak" (0). The left sidebar includes navigation links for Dashboard, UMKM, Transaksi, History, Laporan, and Chat.

Tampilan Halaman Utama Berhasil Registrasi

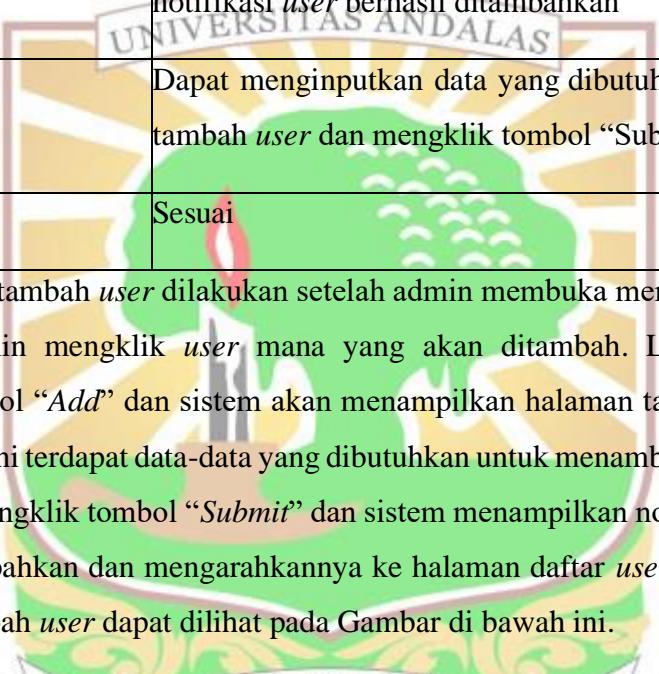
Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil melakukan *register* dan sistem menampilkan halaman utama sesuai dengan *user* yang berhasil melakukan *register*.

### 3. Menambah Data *User*

Pengujian menambah data *user* ini merupakan pengujian yang dilakukan ketika admin ingin menambahkan *user* pada aplikasi. Hasil pengujian tambah *user* dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam halaman tambah <i>user</i>
Yang diharapkan	Sistem menampilkan halaman daftar <i>user</i> dan muncul notifikasi <i>user</i> berhasil ditambahkan
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman tambah <i>user</i> dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian tambah *user* dilakukan setelah admin membuka menu “User Pages”, kemudian admin mengklik *user* mana yang akan ditambah. Lalu admin akan mengklik tombol “Add” dan sistem akan menampilkan halaman tambah pengguna. Pada halaman ini terdapat data-data yang dibutuhkan untuk menambahkan pengguna, admin akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data *user* berhasil ditambahkan dan mengarahkannya ke halaman daftar *user*. Tampilan hasil pengujian tambah *user* dapat dilihat pada Gambar di bawah ini.



The screenshot shows a user interface for adding administrative data. On the left, there's a sidebar with 'Dashboard', 'User Pages' (selected), 'UMKM', and 'Kategori Produk'. The main area has a title 'Menambahkan Data Admin'. It contains five input fields: 'Firstname' (Distifia), 'Lastname' (Oktari), 'No Handphone' (082285443515), 'Provinsi' (Sumatera Barat), and 'Kota' (Padang). A small gear icon is in the bottom right corner of the form area.

Alamat  
Komplek polda balai baru blok G/26

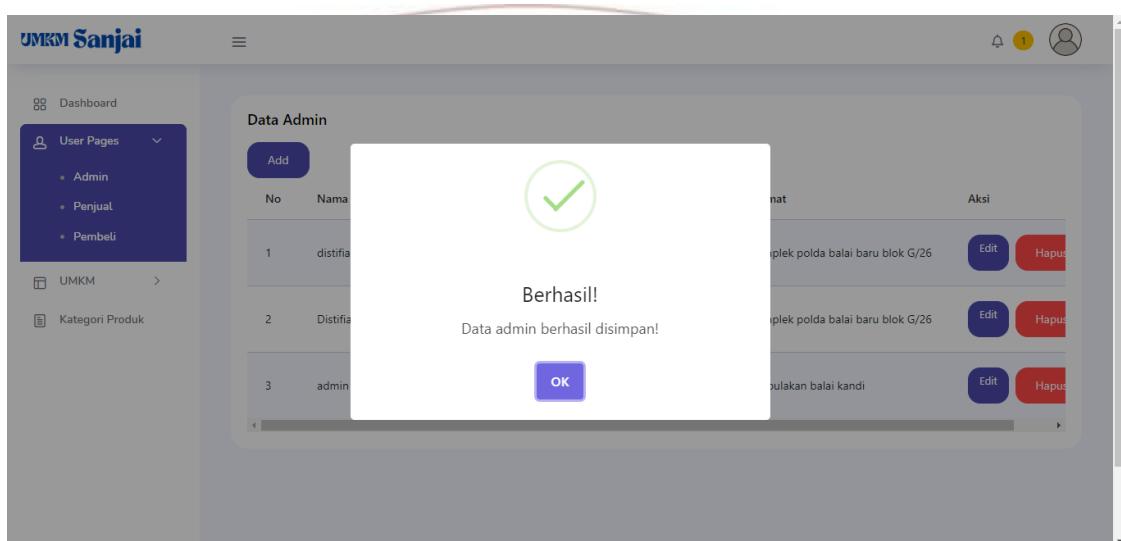
Email  
distifiaoktari@gmail.com

Password  
.....

Foto Profil  
 rsz\_r (1).jpg

Submit

Tampilan Form Tambah *User*



Tampilan Berhasil Tambah *User*

Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil ditambahkan dan sistem menampilkan halaman daftar *user*.

#### 4. Mengedit Data *User*

Pengujian mengedit data *user* ini merupakan pengujian yang dilakukan ketika admin ingin mengubah data *user* pada aplikasi. Hasil pengujian mengedit data *user* dapat dilihat pada tabel di bawah ini.

#### Kasus dan Hasil Uji (Benar)

Data masukan	Seluruh data yang dibutuhkan dalam halaman edit <i>user</i>
--------------	---

Yang diharapkan	Sistem menampilkan halaman daftar <i>user</i> dan muncul notifikasi data <i>user</i> berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman edit <i>user</i> dan mengklik tombol “Submit”
Hasil	Sesuai

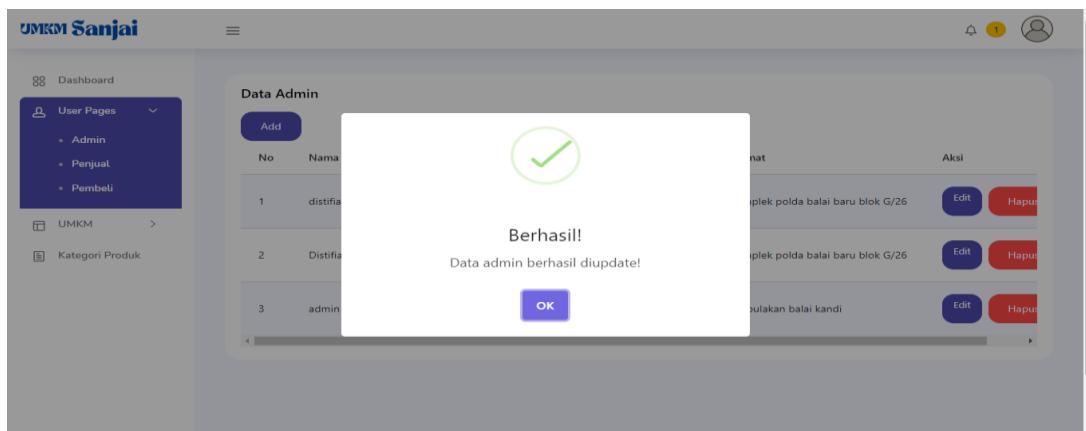
Pengujian edit data *user* dilakukan setelah admin membuka menu “User Pages”, kemudian admin mengklik *user* mana yang akan diedit. Lalu admin akan mengklik tombol “Edit” dan sistem akan menampilkan halaman edit pengguna. Pada halaman ini terdapat data-data yang dibutuhkan untuk mengubah data pengguna, admin akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data *user* berhasil diubah dan mengarahkannya ke halaman daftar *user*. Tampilan hasil pengujian megubah data *user* dapat dilihat pada Gambar di bawah ini.

The screenshot shows a user interface for editing a user profile. The top navigation bar includes 'UMKM Sanjai', 'Dashboard', 'User Pages' (selected), 'UMKM', and 'Kategori Produk'. The main content area is titled 'Mengubah Data Admin' (Change User Data). It contains the following input fields:

- Firstname: Distifia
- Lastname: Oktari
- No Handphone: 081261368371
- Provinsi: Sumatera Barat
- Kota: Padang

A blue 'Submit' button is located at the bottom right of the form.

Tampilan Form Edit User



Tampilan Berhasil Edit User

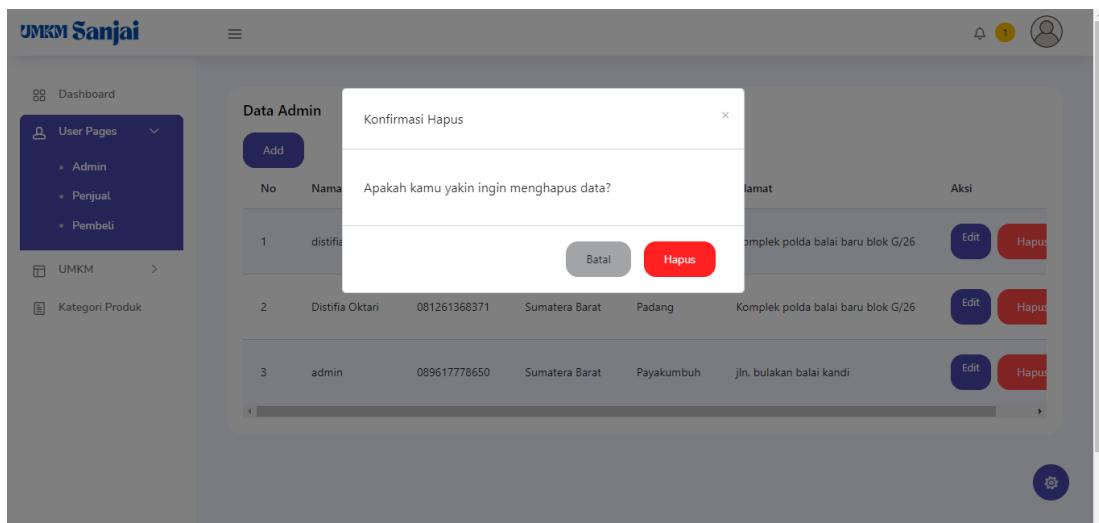
Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil diubah dan sistem menampilkan halaman daftar *user*.

##### 5. Menghapus Data User

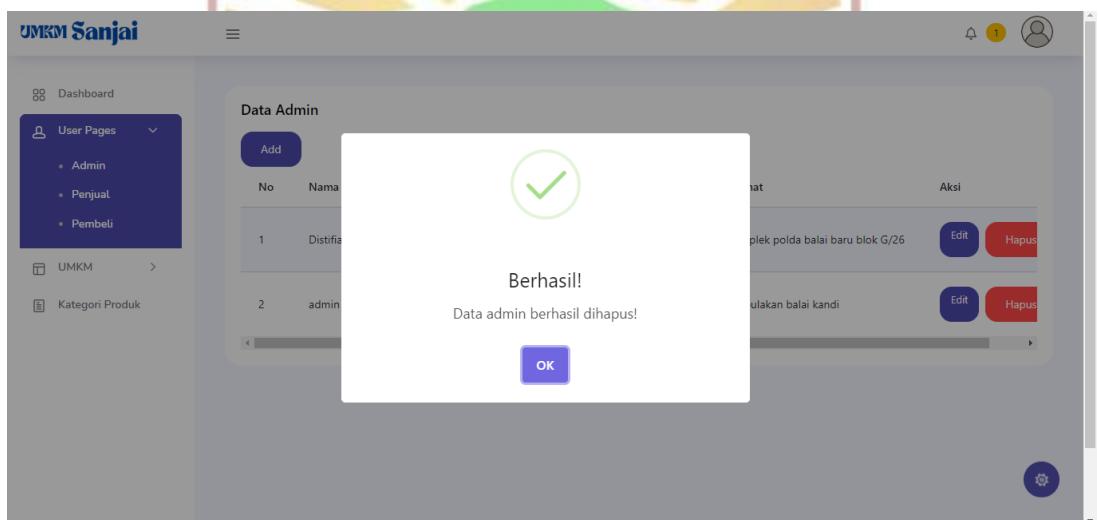
Pengujian menghapus data *user* ini merupakan pengujian yang dilakukan ketika admin ingin menghapus data *user* pada aplikasi. Hasil pengujian menghapus *user* dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Data yang akan dihapus
Yang diharapkan	Sistem menampilkan notifikasi berhasil menghapus data
Pengamatan	Dapat menghapus data <i>user</i>
Hasil	Sesuai

Pengujian hapus *user* dilakukan setelah admin membuka menu “User Pages”, kemudian admin mengklik *user* mana yang akan dihapus. Lalu admin akan mengklik tombol “Delete” dan sistem akan menampilkan pesan konfirmasi untuk menghapus data. Tampilan hasil pengujian hapus *user* dapat dilihat pada Gambar di bawah ini.



Tampilan Konfirmasi Hapus *User*



Tampilan Hapus Berhasil

Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil dihapus dan sistem menampilkan halaman daftar *user*.

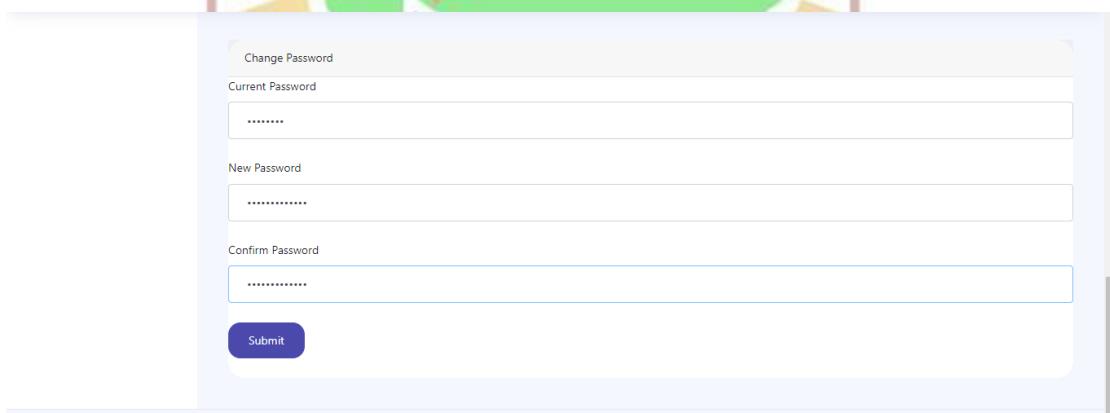
## 6. Mengubah *Password*

Pengujian mengubah *password* ini merupakan pengujian yang dilakukan ketika *user* ingin mengubah *password*. Hasil pengujian mengubah *password* dapat dilihat pada tabel di bawah ini.

### Kasus dan Hasil Uji (Benar)

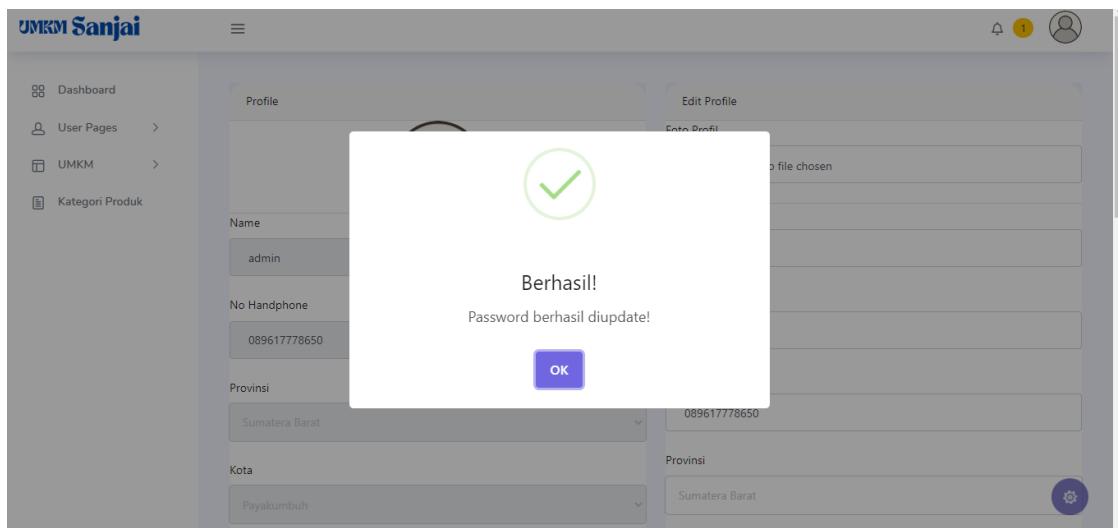
Data masukan	Seluruh data yang dibutuhkan dalam halaman ubah <i>password</i>
Yang diharapkan	Sistem menampilkan halaman profil dan muncul notifikasi data <i>password</i> berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halama ubah <i>password</i> dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian mengubah *password* dilakukan setelah *user* membuka menu profil, kemudian aplikasi akan menampilkan halaman profil. Pada halaman ini terdapat data-data yang dibutuhkan untuk mengubah data *password*, *user* akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data *password* berhasil diubah dan mengarahkannya ke halaman profil. Tampilan hasil pengujian megubah data profil dapat dilihat pada Gambar di bawah ini.



The screenshot shows a 'Change Password' form. At the top, it says 'Change Password'. There are three input fields: 'Current Password' (containing '.....'), 'New Password' (containing '.....'), and 'Confirm Password' (containing '.....'). Below these fields is a blue 'Submit' button.

Tampilan *Form* Ubah Password



Tampilan Ubah *Password* Berhasil

Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil mengubah *password* dan sistem menampilkan halaman profil.

## 7. Mengubah Data Profil

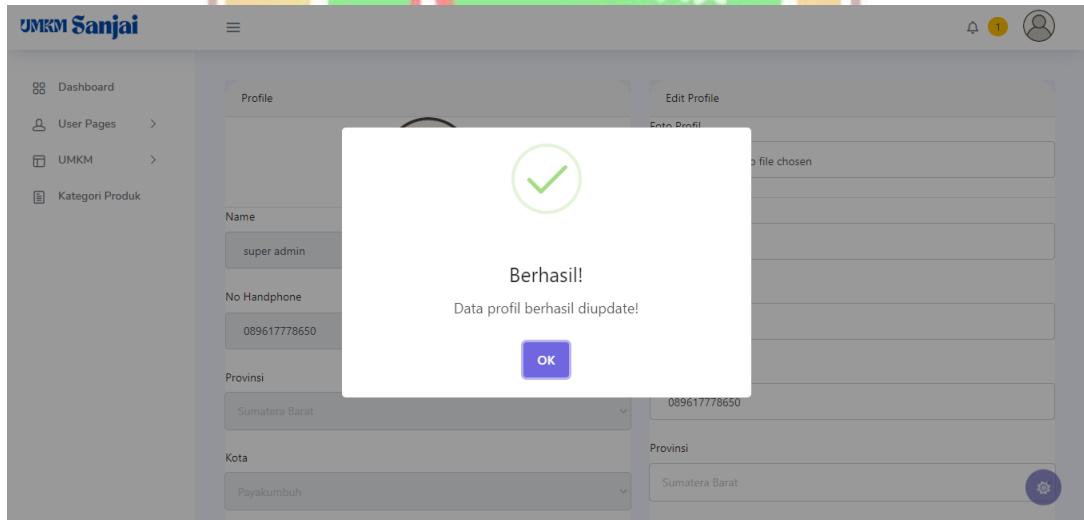
Pengujian mengubah data profil ini merupakan pengujian yang dilakukan ketika *user* ingin mengubah profil. Hasil pengujian mengubah data profil dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam halaman ubah profil
Yang diharapkan	Sistem menampilkan halaman profil dan muncul notifikasi data profil berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman ubah profil dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian mengubah profil dilakukan setelah *user* membuka menu profil, kemudian aplikasi akan menampilkan halaman profil. Pada halaman ini terdapat data-data yang dibutuhkan untuk mengubah data profil, *user* akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data profil berhasil diubah dan

mengarahkannya ke halaman profil. Tampilan hasil pengujian mengubah data profil dapat dilihat pada Gambar di bawah ini.

Tampilan Form Ubah Profil



Tampilan Ubah Profil Berhasil

Berdasarkan gambar diatas dapat dilihat bahwa *user* berhasil mengubah data profil dan sistem menampilkan halaman profil.

## 8. Menambah Data Kategori

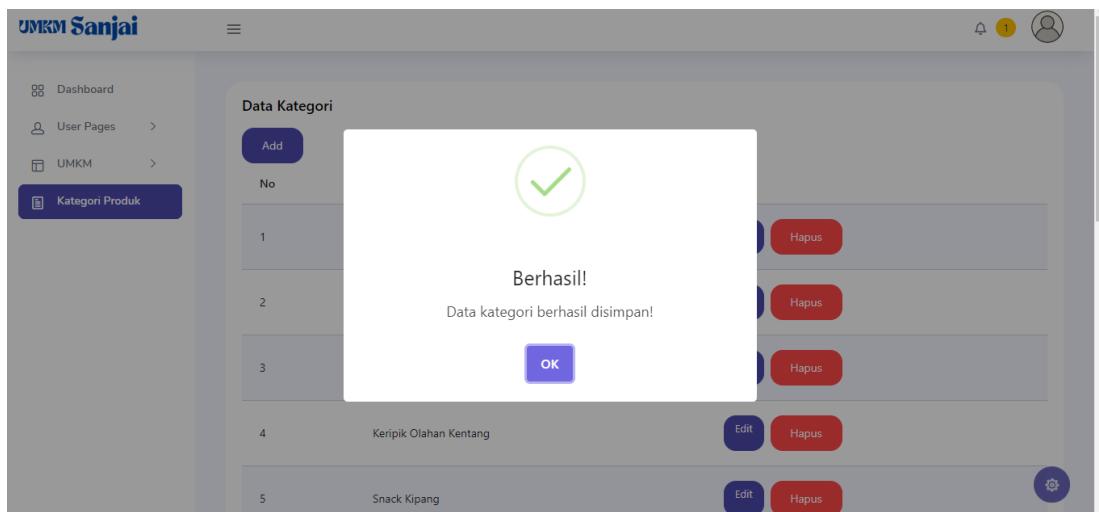
Pengujian menambah data kategori ini merupakan pengujian yang dilakukan ketika admin ingin menambahkan kategori produk pada aplikasi. Hasil pengujian tambah kategori dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam halaman tambah kategori
Yang diharapkan	Sistem menampilkan halaman daftar kategori dan muncul notifikasi kategori berhasil ditambahkan
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman tambah kategori dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian tambah kategori dilakukan setelah admin membuka menu “Kategori”, kemudian admin mengklik tombol “Add” dan sistem akan menampilkan halaman tambah kategori. Pada halaman ini terdapat data yang dibutuhkan untuk menambahkan kategori, admin akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data kategori berhasil ditambahkan dan mengarahkannya ke halaman daftar kategori. Tampilan hasil pengujian tambah kategori dapat dilihat pada Gambar di bawah ini.

The screenshot shows a web application interface for adding a new category. The header includes the logo 'UMKM Sanjai'. On the left, there is a sidebar with navigation links: 'Dashboard', 'User Pages', 'UMKM', and 'Kategori Produk' (which is highlighted in purple). The main content area has a title 'Menambahkan Data Kategori' and a form field labeled 'Nama Kategori' containing the text 'Kripik Olahan Ubi Ungu'. Below the input field is a blue 'Submit' button.

Tampilan *Form* Tambah Kategori



Tampilan Kategori Berhasil Ditambahkan

Berdasarkan gambar diatas dapat dilihat bahwa kategori berhasil ditambahkan dan sistem menampilkan halaman daftar kategori.

#### 9. Mengedit Data Kategori

Pengujian mengedit data kategori ini merupakan pengujian yang dilakukan ketika admin ingin mengubah data kategori pada aplikasi. Hasil pengujian mengedit data kategori dapat dilihat pada tabel di bawah ini.

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan dalam halaman edit kategori
Yang diharapkan	Sistem menampilkan halaman daftar kategori dan muncul notifikasi data kategori berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman edit kategori dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian edit data kategori dilakukan setelah admin membuka menu “Kategori”, kemudian admin mengklik tombol “Edit” dan sistem akan menampilkan halaman edit kategori. Pada halaman ini terdapat data yang dibutuhkan untuk mengubah data kategori, admin akan mengklik tombol “Submit” dan sistem

menampilkan notifikasi data kategori berhasil diubah dan mengarahkannya ke halaman daftar kategori. Tampilan hasil pengujian megubah data kategori dapat dilihat pada Gambar di bawah ini.

The image consists of two screenshots of the 'UMKM Sanjai' application interface. The top screenshot shows the 'Edit Category' form with the category name 'Kripik Ubi Ungu' entered in the input field, and a 'Submit' button below it. The bottom screenshot shows the 'Category Data' list with a modal dialog box displaying a green checkmark icon, the text 'Berhasil!', and the message 'Data kategori berhasil diupdate!' (Category data successfully updated!). The modal has an 'OK' button at the bottom. Both screenshots include a sidebar with navigation links: Dashboard, User Pages, UMKM, and Kategori Produk (which is highlighted in blue).

Tampilan Edit Kategori

Tampilan Kategori Berhasil Diubah

Berdasarkan gambar diatas dapat dilihat bahwa kategori berhasil diubah dan sistem menampilkan halaman daftar kategori.

## 10. Menghapus Data Kategori

Pengujian menghapus data kategori ini merupakan pengujian yang dilakukan ketika admin ingin menghapus data kategori pada aplikasi. Hasil pengujian menghapus kategori dapat dilihat pada tabel di bawah ini.

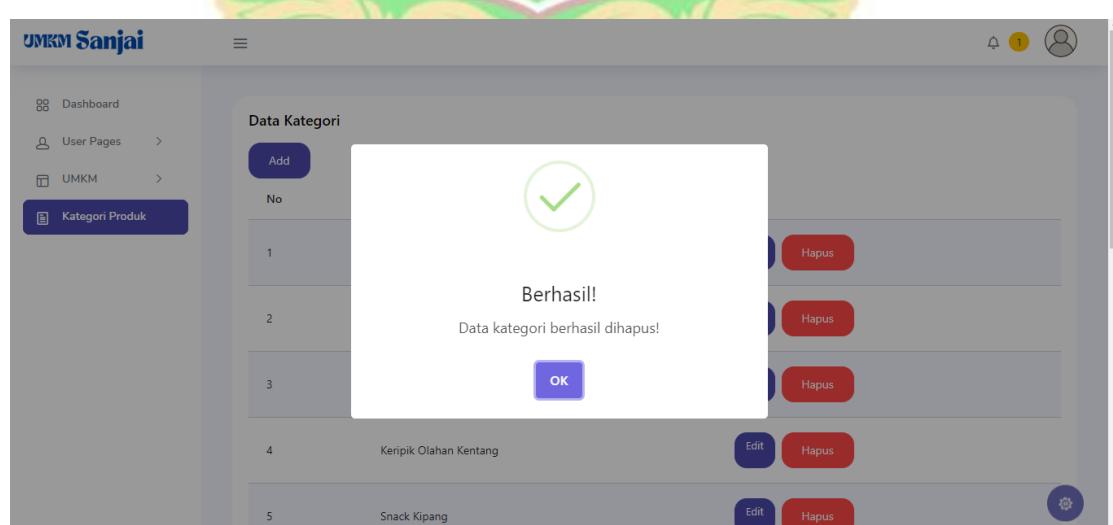
Kasus dan Hasil Uji (Benar)	
Data masukan	Data yang akan dihapus

Yang diharapkan	Sistem menampilkan notifikasi berhasil menghapus data
Pengamatan	Dapat menghapus data kategori
Hasil	Sesuai

Pengujian hapus kategori dilakukan setelah admin membuka menu “Kategori”, kemudian admin mengklik kategori mana yang akan dihapus. Lalu admin akan mengklik tombol “Delete” dan sistem akan menampilkan pesan konfirmasi untuk menghapus data. Tampilan hasil pengujian hapus kategori dapat dilihat pada Gambar di bawah ini.



Tampilan Konfirmasi Hapus Kategori



Tampilan Hapus Kategori Berhasil

Berdasarkan gambar diatas dapat dilihat bahwa kategori berhasil dihapus dan sistem menampilkan halaman daftar kategori.

## 11. Mengedit Data UMKM

Pengujian mengedit data UMKM ini merupakan pengujian yang dilakukan ketika penjual ingin mengubah data UMKM pada aplikasi. Hasil pengujian mengedit data UMKM dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam halaman edit UMKM
Yang diharapkan	Sistem menampilkan halaman daftar detail UMKM dan muncul notifikasi data UMKM berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halaman edit UMKM dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian edit data UMKM dilakukan setelah penjual membuka menu “UMKM”, kemudian penjual mengklik tombol “View Umkm” lalu mengklik tombol “Edit Umkm” dan sistem akan menampilkan halaman edit UMKM. Pada halaman ini terdapat data yang dibutuhkan untuk mengubah data UMKM, penjual akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data UMKM berhasil diubah dan mengarahkannya ke halaman detail UMKM. Tampilan hasil pengujian mengubah data UMKM dapat dilihat pada Gambar di bawah ini.

**UMKM Sanjai**

Dashboard UMKM Transaksi History Laporan Chat

### Sanjai Oviga

Lihat Surat Izin Tambah Produk Edit UMKM Hapus UMKM

**Sanjai Oviga**  
Vina Kurnia 081275920305

**Informasi Umkm**  
Sanjai oviga merupakan rumah produksi olahan makanan yang menghasilkan beraneka jenis sanjai serta kripik olahan dari ubi talas dan kentang. Sanjai ini sudah berdiri sejak tahun 1996.

**Alamat**  
Jl. Aceh, Tanjung. Pauh, Kec. Payakumbuh Bar., 26222

**Metode Pembayaran**  
- Bank Mandiri (1310007374443 )  
- Bank BRI ( 2115007273114 )

Nama Produk Kategori Action  
Ganepo Keripik Olahan Singkong Detail Edit Hapus

**Tampilan Detail UMKM**

**UMKM Sanjai**

Dashboard UMKM Transaksi History Laporan Chat

### Mengubah Data UMKM

Informasi Umum Alamat Pembayaran

Nama Umkm Sanjai Oviga

No Handphone 082285193081

Informasi Tambahan dari ubi talas dan kentang. Sanjai ini sudah berdiri sejak tahun 1996

Upload Surat Izin Usaha \*boleh HKI Merek atau SIUP

Change 1692777073.siup.jpeg X

**PEMERINTAH KABUPATEN BADAN PERIZINAN PERIZINAN TERPADU SURAT IZIN USAHA PERDAGANGAN (SIUP) KECIL Nomor : 1692777073**

**UMKM Sanjai**

History Laporan Chat

Informasi Umum Alamat Pembayaran

Provinsi Sumatera Barat

Kota Payakumbuh

Alamat Jl. Aceh, Tanjung. Pauh, Kec. Payakumbuh Bar., 26222

Coordinate -0.24292491117463 100.6285247659

**Google** Keyboard shortcuts Map data ©2023 Terms of Use Report a map error

**Tampilan Form Edit UMKM**

**Tampilan Edit UMKM Berhasil**

Berdasarkan gambar diatas dapat dilihat bahwa UMKM berhasil diubah dan sistem menampilkan halaman daftar detail UMKM.

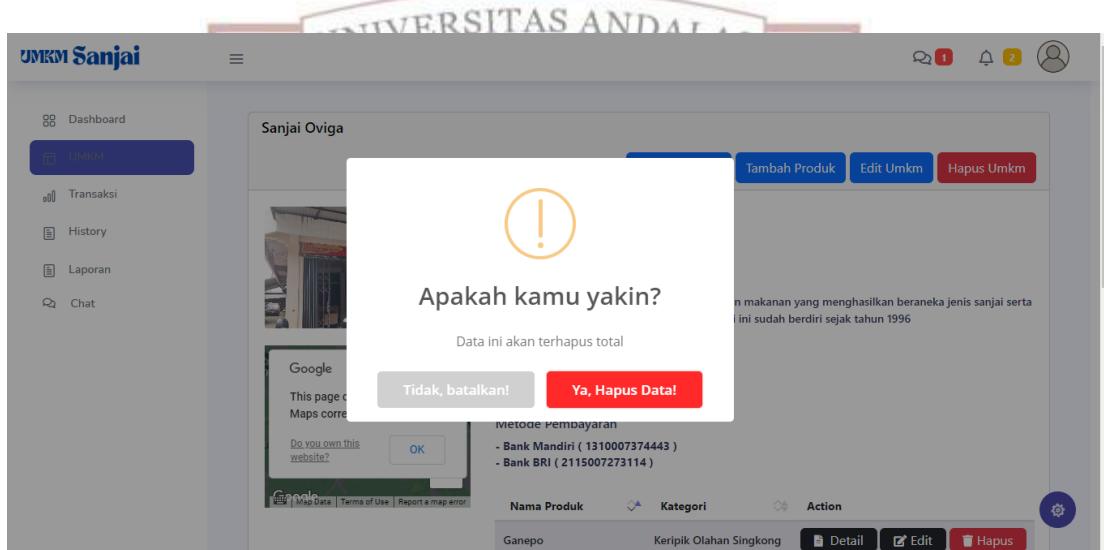
## 12. Menghapus Data UMKM

Pengujian menghapus data UMKM ini merupakan pengujian yang dilakukan ketika penjual ingin menghapus data UMKM pada aplikasi. Hasil pengujian menghapus UMKM dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Data yang akan dihapus

Yang diharapkan	Sistem menampilkan notifikasi berhasil menghapus data
Pengamatan	Dapat menghapus data UMKM
Hasil	Sesuai

Pengujian hapus UMKM dilakukan setelah penjual membuka menu “UMKM”, kemudian penjual mengakses halaman “View Umkm” lalu mengklik tombol “Hapus UMKM”. Lalu sistem akan menampilkan pesan konfirmasi untuk menghapus data. Tampilan hasil pengujian hapus UMKM dapat dilihat pada Gambar di bawah ini.



Tampilan Hapus UMKM

Berdasarkan gambar diatas dapat dilihat bahwa UMKM berhasil dihapus dan sistem menampilkan halaman daftar UMKM.

### 13. Mengedit Data Produk

Pengujian mengedit data produk ini merupakan pengujian yang dilakukan ketika penjual ingin mengubah data produk pada aplikasi. Hasil pengujian mengedit data produk dapat dilihat pada tabel di bawah ini.

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan dalam halaman edit produk

Yang diharapkan	Sistem menampilkan halaman daftar detail UMKM dan muncul notifikasi data produk berhasil diubah
Pengamatan	Dapat menginputkan data yang dibutuhkan pada halama edit produk dan mengklik tombol “Submit”
Hasil	Sesuai

Pengujian edit data produk dilakukan setelah penjual membuka menu “UMKM”, kemudian penjual mengklik tombol “View Umkm” lalu mengklik tombol “Edit” pada produk yang ingin diubah dan sistem akan menampilkan halaman edit produk. Pada halaman ini terdapat data yang dibutuhkan untuk mengubah data produk, penjual akan mengklik tombol “Submit” dan sistem menampilkan notifikasi data produk berhasil diubah dan mengarahkannya ke halaman detail UMKM. Tampilan hasil pengujian megubah data produk dapat dilihat pada Gambar di bawah ini.

The screenshot shows a user interface for editing product data. On the left is a sidebar with navigation links: Dashboard, UMKM, Transaksi, History, Laporan, and Chat. The main area has a title "Mengedit Data Produk". Below it is a tabbed section with "Informasi Umum Produk" selected. The form contains the following data:

Field	Value
Nama Produk	Ganepo
Kategori Produk	Keripik Olahan Singkong
No Izin P-IRT	2.15.1376.07.0351.25
Deskripsi	Terbuat dari singkong. Gurih dan renyah.

At the bottom right of the form is a blue "Next >" button.

**UMKM Sanjai**

☰

- Dashboard
- UMKM
- Transaksi
- History
- Laporan
- Chat

**Mengedit Data Produk**

Informasi Umum Produk      Rincian Produk      Gambar Produk

Kemasan Produk

kecil  
 sedang  
 besar  
 100 gr  
 200 gr  
 250 gr  
 300 gr  
 500 gr  
 1 kg  
 3 kg  
 5 kg  
 10 kg

Kemasan: kecil  
Kode Produk: 1007



☰

Stok: 20  
Harga: 7000  
Berat Pengiriman: 1000

Kemasan: sedang  
Kode Produk: 1008  
Stok: 20  
Harga: 10000  
Berat Pengiriman: 1000

Kemasan: besar  
Kode Produk: 1009  
Stok: 20  
Harga: 13000  
Berat Pengiriman: 1000

< Previous      Next >

**UMKM Sanjai**

☰

Dashboard      UMKM      Transaksi      History      Laporan      Chat

**Mengedit Data Produk**

Informasi Umum Produk      Rincian Produk      Gambar Produk

Tambah gambar



Upload

Delete      Select image      Delete

< Previous      Next >

**UMKM Sanjai**

☰

Dashboard      UMKM      Transaksi      History      Laporan      Chat

**Mengedit Data Produk**

Informasi Umum Produk      Rincian Produk      Gambar Produk

Tambah gambar

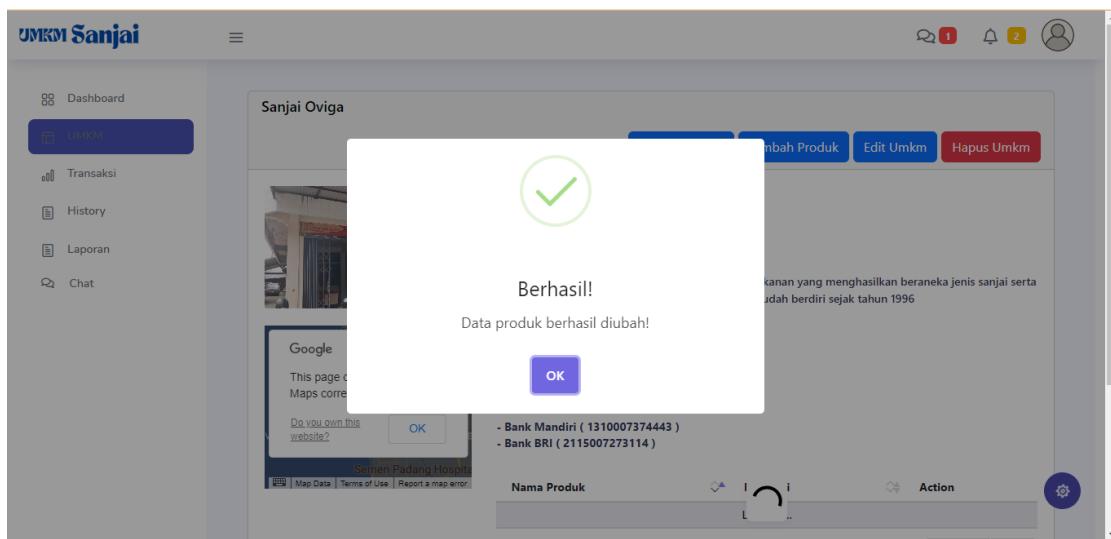


Upload

Delete      Select image      Delete

< Previous      Next >

Tampilan *Form Edit Produk*



Tampilan Edit Produk Berhasil

Berdasarkan gambar diatas dapat dilihat bahwa produk berhasil diubah dan sistem menampilkan halaman daftar detail UMKM.

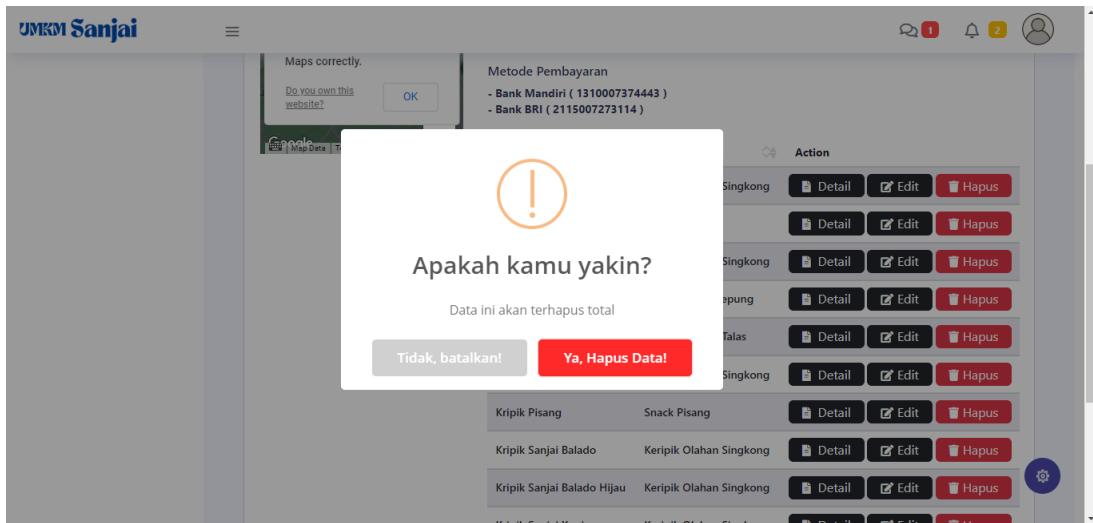
#### 14. Menghapus Data Produk

Pengujian menghapus data produk ini merupakan pengujian yang dilakukan ketika penjual ingin menghapus data produk pada UMKMnya. Hasil pengujian menghapus produk dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Data yang akan dihapus
Yang diharapkan	Sistem menampilkan notifikasi berhasil menghapus data
Pengamatan	Dapat menghapus data produk
Hasil	Sesuai

Pengujian hapus produk dilakukan setelah penjual membuka menu “UMKM”, kemudian penjual mengklik tombol “View umkm”. Lalu sistem akan menampilkan detail UMKM berserta produk-produk yang dimiliknya. Lalu penjual akan mengklik tombol “Delete” pada produk yang ingin dihapus dan sistem akan menampilkan

pesan konfirmasi untuk menghapus produk. Tampilan hasil pengujian hapus produk dapat dilihat pada Gambar di bawah ini.



Tampilan Menghapus Produk

Berdasarkan gambar diatas dapat dilihat bahwa produk berhasil dihapus dan sistem menampilkan halaman daftar produk.

### 15. Terima Perizinan UMKM

Pengujian terima perizinan UMKM ini merupakan pengujian yang dilakukan ketika admin ingin mengkonfirmasi pengajuan data UMKM pada aplikasi. Hasil pengujian menerima perizinan UMKM dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk menerima perizinan UMKM
Yang diharapkan	Sistem menampilkan daftar UMKM yang membutuhkan konfirmasi
Pengamatan	Dapat menerima perizinan UMKM
Hasil	Sesuai

Pengujian menerima perizinan UMKM dilakukan setelah admin membuka menu “UMKM”, kemudian admin mengakses halaman “Perlu Konfirmasi” lalu mengklik

tombol ceklis. Lalu sistem akan menampilkan pesan konfirmasi untuk mengkonfirmasi UMKM. Tampilan hasil pengujian menerima perizinan UMKM dapat dilihat pada Gambar di bawah ini.

Nama UMKM	Pemilik	Alamat	Tanggal Pembuatan	Action
Sanjai Oviga 2	Vina Kurnia	Jl. Aceh, Tanjung Pauh 26222	24-08-2023 / 15:07	<i>i</i> <span style="color: green;">✓</span> <span style="color: red;">✗</span>

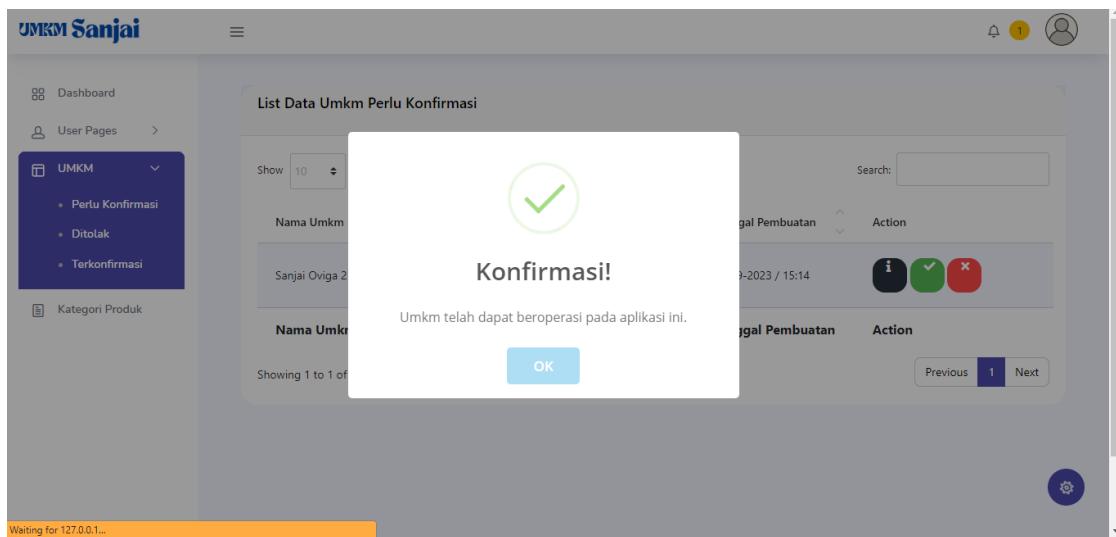
Tampilan UMKM Perlu Konfirmasi

Apakah kamu yakin?

Venue yang telah dikonfirmasi akan dibebaskan untuk beroperasi pada aplikasi ini

Tidak, batalkan!   Ya, konfirmasi!

Tampilan Konfirmasi Terima UMKM



### Tampilan Berhasil Konfirmasi UMKM

Berdasarkan gambar diatas dapat dilihat bahwa UMKM berhasil dikonfirmasi terima dan sistem menampilkan halaman daftar UMKM perlu konfirmasi.

#### 16. Tolak Perizinan UMKM

Pengujian tolak perizinan UMKM ini merupakan pengujian yang dilakukan ketika admin ingin mengkonfirmasi pengajuan data UMKM pada aplikasi. Hasil pengujian tolak perizinan UMKM dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk menolak perizinan UMKM
Yang diharapkan	Sistem menampilkan daftar UMKM yang membutuhkan konfirmasi
Pengamatan	Dapat menolak perizinan UMKM
Hasil	Sesuai

Pengujian menolak perizinan UMKM dilakukan setelah admin membuka menu “UMKM”, kemudian admin mengakses halaman “Perlu Konfirmasi” lalu mengklik

tombol silang. Lalu sistem akan menampilkan pesan konfirmasi untuk mengkonfirmasi UMKM. Tampilan hasil pengujian menolak perizinan UMKM dapat dilihat pada Gambar di bawah ini.

### Tampilan UMKM Ditolak

Berdasarkan gambar diatas dapat dilihat bahwa UMKM berhasil dikonfirmasi tolak dan sistem menampilkan halaman daftar UMKM perlu konfirmasi.

## 17. *Banned* UMKM

Pengujian *banned* UMKM ini merupakan pengujian yang dilakukan ketika admin ingin menutup akses UMKM yang sudah dikonfirmasi pada aplikasi. Hasil pengujian *banned* UMKM dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk melakukan <i>banned</i> UMKM
Yang diharapkan	Sistem menampilkan daftar UMKM yang telah dikonfirmasi dan menekan button <i>banned</i> UMKM
Pengamatan	Dapat menutup perizinan UMKM
Hasil	Sesuai

Pengujian menutup perizinan UMKM dilakukan setelah admin membuka menu “UMKM”, kemudian admin mengakses halaman “terkonfirmasi” lalu mengklik tombol silang. Lalu sistem akan menampilkan pesan konfirmasi untuk menutup akses UMKM. Tampilan hasil pengujian *banned* UMKM dapat dilihat pada Gambar di bawah ini.

Nama Ukm	Pemilik	Action
Sanjai 3 Saudara	Afifah	[Edit] [Delete]
Sanjai Anna	Annisa Herina	[Edit] [Delete]
Sanjai Erina	Ningsih Eka Putri	[Edit] [Delete]
Sanjai Nirina	Nirina	[Edit] [Delete]
Sanjai Oviga	Vina Kurnia	[Edit] [Delete]
Sanjai Pak Bu	Afrisal	[Edit] [Delete]

Tampilan Halaman Terkonfirmasi

### Tampilan UMKM Dibanned

Berdasarkan gambar diatas dapat dilihat bahwa UMKM berhasil ditutup aksesnya dan sistem menampilkan halaman daftar UMKM yang sudah di *banned*.

#### 18. Melakukan *Chat*

Pengujian melakukan chat ini merupakan pengujian yang merupakan percakapan antara penjual dan pembeli. Hasil pengujian melakukan chat dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam melakukan <i>chat</i>
Yang diharapkan	Sistem menampilkan halaman daftar <i>chat</i>
Pengamatan	Dapat melakukan <i>chat</i>
Hasil	Sesuai

Pengujian melakukan chat dilakukan setelah *user* membuka menu “*Chat*”, kemudian *user* dapat mengirim pesan dengan menekan tombol “*kirim*”. Tampilan hasil pengujian melakukan chat dapat dilihat pada Gambar di bawah ini.



## Featured Product

All   Keripik Olahan Talas   Keripik Olahan Singkong   Keripik Olahan Ubi Jalar   Keripik Tempe   Snack Olahan

Send a message  

**UNIVERSITAS ANDALAS**  
Tampilan Chat Pembeli

Fadhillah Aini   Logout

HOME   PURCHASE   HISTORY   CHAT   PROFIL

Recent

Vina Kurnia   17:54 PM, 01 Sep 2023  
hallo

hallo   17:41 PM | 01 Sep 2023

Sanjai Oviga

UNTUK RAKYAT BANGSA

UMKM Sanjai

Dashboard  

- UMKM
- Transaksi
- History
- Laporan
- Chat

Recent

Fadhillah Aini   17:54 PM, 01 Sep 2023  
hallo

Dhea Irma Putri   12:03 PM, 25 Aug 2023  
sanjainya kakak

Fadhillah

hallo   17:41 PM | 01 Sep 2023

Tampilan Chat Penjual

Berdasarkan gambar diatas dapat dilihat bahwa chat berhasil dilakukan dan sistem menampilkan halaman *list chat*.

## 19. Melakukan Penelusuran UMKM

Pengujian melakukan penelusuran ini merupakan pengujian dengan melakukan pencarian sesuai dengan nama UMKM yang dicari. Hasil pengujian melakukan penelusuran dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam penelusuran
Yang diharapkan	Sistem menampilkan halaman daftar produk sesuai dengan UMKM yang ditelusuri
Pengamatan	Dapat melakukan penelusuran
Hasil	Sesuai

Pengujian melakukan penelusuran dilakukan setelah *user* melakukan pencarian pada kolom pencarian, kemudian sistem akan menampilkan daftar produk sesuai dengan UMKM yang ditelusuri berdasarkan kata kunci yang dicari. Tampilan hasil pengujian melakukan penelusuran dapat dilihat pada Gambar di bawah ini.

Produk	Detail
Sanjai Kuning Bumbu	Sanjai Oviga Rp 10,000,00 - Rp 10,000,00
Kripik Sanjai Balado	Sanjai Oviga Rp 10,000,00 - Rp 10,000,00
Kripik Lidi Balado	Sanjai Oviga Rp 10,000,00 - Rp 10,000,00
Kripik Sanjai Balado Hijau	Sanjai Oviga Rp 10,000,00 - Rp 10,000,00

Tampilan Melakukan Pencarian

Berdasarkan gambar diatas dapat dilihat bahwa penelusuran berhasil dilakukan dan sistem menampilkan halaman list UMKM berdasarkan kata kunci.

## 20. Melakukan Pembelian produk

Pengujian melakukan pembelian produk ini merupakan pengujian yang dilakukan oleh pembeli ketika ingin melakukan transaksi. Hasil pengujian melakukan pembelian dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan dalam melakukan transaksi pembelian
Yang diharapkan	Sistem menampilkan halaman <i>purchase</i>
Pengamatan	Dapat melakukan pembelian
Hasil	Sesuai

Pengujian melakukan pembelian dilakukan pembeli ketika mengakses halaman detail produk, kemudian mengisikan inputan varian produk dan menambahkan ke keranjang, pada halaman keranjang pembeli mengklik tombol “*Proceed to Checkout*” dan sistem menampilkan halaman *checkout*, pada halaman ini pembeli menginputkan pilihan ekspedisi dan mengklik tombol purchase dan sistem menampilkan halaman pembayaran, pada halaman ini pembeli menginputkan bukti transfer. Tampilan hasil pengujian melakukan pembelian dapat dilihat pada Gambar di bawah ini.



**Ganepo**

Sanjai Oviga

**Rp 10.000,00**

Terbuat dari singkong. Gurih dan renyah.

---

**Kategori** Keripik Olahan Singkong

**Stok** 20

---

**Available Sizes :**

kecil  sedang  besar

- 1 +

**ADD TO CART**

### Tampilan Detail Produk

Products
Price
Quantity
Total



Ganepo

**Rp10000,00**

- 1 +

**Rp10000,00**

X

**CONTINUE SHOPPING**
**UPDATE CART**

**Cart Total**

Subtotal	<b>Rp10000,00</b>
<b>PROCEED TO CHECKOUT</b>	

### Tampilan Keranjang

Province\*
Subtotal **Rp10000,00**

Province\*

City\*

Address\*

Pilih Ekspedisi\*

Pilih Layanan\*

Pilih Metode Pembayaran\*

Bank Mandiri (1310007374443)

Bank BRI (2115007273114)

**Biaya Pengiriman Rp.8000,-**

**Total Rp18000,00**

**PLACE ORDER**

### Tampilan Checkout

Terima kasih telah melakukan pembelian.  
Waktu yang tersisa untuk melakukan pembayaran 1h 29m 43s

Ringkasan order

No	Gambar	Nama Produk	Jumlah	Harga Produk	Subtotal
1		Ganepo	1	Rp10.000,00	Rp10.000,00

Biaya Pengiriman: Rp8.000,00  
Total Harga: Rp18.000,00

Silakan melakukan pembayaran ke rekening berikut:  
Bank BRI (2115007273114)  
Bukti Transfer  
 Choose File No file chosen  
[Submit Payment](#)

**Tampilan Pembayaran**

UNIVERSITAS ANDALAS

HOME PURCHASE HISTORY CHAT PROFIL

**Payment**  
Home Purchase Payment



Terima kasih telah melakukan pembelian.  
Pembayaran mu berhasil. Tunggu konfirmasi pesanan dari toko.

## 21. Menerima Pesanan

Berdasarkan gambar diatas dapat dilihat bahwa pembelian produk berhasil dilakukan dan sistem menampilkan halaman daftar pembelian.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk menerima pesanan
Yang diharapkan	Sistem menampilkan daftar transaksi

Pengamatan	Dapat menerima transaksi pesanan
Hasil	Sesuai

Pengujian menerima pesanan dilakukan setelah penjual membuka menu “Transaksi”, kemudian penjual mengklik tombol ceklis. Lalu sistem akan menampilkan pesan konfirmasi untuk mengkonfirmasi pesanan akan diproses. Tampilan hasil pengujian menerima pesanan dapat dilihat pada Gambar di bawah ini.

The screenshot shows a table of transactions with columns for Date, Customer, Amount, Status, and Action. One transaction is highlighted with a yellow background and 'Expired' status. A modal window is open over the table, asking 'Apakah permintaan pemesanan ini diterima?' (Is this order request accepted?). It includes a large exclamation mark icon and two buttons: 'Tidak, batalkan!' (No, cancel!) and 'Ya, terima!' (Yes, accept!).

Date	Customer	Amount	Status	Action
08-09-2023	Sanjai Oviga	49500	Selesai	<i>i</i> Detail
09-09-2023	Sanjai Oviga	21000	Expired	<i>i</i> Detail
10-09-2023	Sanjai Oviga	26000	Selesai	<i>i</i> Detail
10-09-2023	Sanjai Oviga	42000	Selesai	<i>i</i> Detail
12-09-2023	Sanjai Oviga	21000	Menunggu Konfirmasi	<i>i</i> Detail <span style="color: green;">✓</span> <span style="color: red;">✗</span>
24-08-2023	Sanjai Oviga	57000	Selesai	<i>i</i> Detail
25-08-2023	Sanjai Oviga	54000	Selesai	<i>i</i> Detail

Showing 1 to 9 of 9 entries

Previous 1 Next

The screenshot shows a table of transactions with one row highlighted. A modal window is open, displaying a large exclamation mark icon and the text 'Apakah permintaan pemesanan ini diterima?' (Is this order request accepted?). Below the question, it says 'Pemesanan yang diterima akan tercatat dalam transaksi' (Accepted orders will be recorded in the transaction). At the bottom of the modal are two buttons: 'Tidak, batalkan!' (No, cancel!) and 'Ya, terima!' (Yes, accept!).

Date	Customer	Amount	Status	Action
02-09-2023	Sanjai Oviga	18000	Selesai	<i>i</i> Detail
06-09-2023	Sanjai Oviga	59000	Selesai	<i>i</i> Detail
08-09-2023				<i>i</i> Detail
09-09-2023				<i>i</i> Detail
10-09-2023				<i>i</i> Detail
10-09-2023				<i>i</i> Detail
12-09-2023				<i>i</i> Detail
24-08-2023				<i>i</i> Detail
25-08-2023	Sanjai Oviga	54000	Selesai	<i>i</i> Detail

Showing 1 to 9 of 9 entries

Previous 1 Next

Tampilan Konfirmasi Pesanan

09-09-2023	Sanjai Oviga	21000	Expired	<a href="#">Detail</a>
10-09-2023	Sanjai Oviga	26000	Selesai	<a href="#">Detail</a>
10-09-2023	Sanjai Oviga	42000	Selesai	<a href="#">Detail</a>
12-09-2023	Sanjai Oviga	21000	Pesanan Disiapkan	<a href="#">Detail</a>
24-08-2023	Sanjai Oviga	57000	Selesai	<a href="#">Detail</a>
25-08-2023	Sanjai Oviga	54000	Selesai	<a href="#">Detail</a>

Showing 1 to 9 of 9 entries

Previous [1](#) Next

Copyright © 2021. Premium Bootstrap admin template from BootstrapDash. All rights reserved.  
Distributed by Themewagon

Hand-crafted & made with

### Tampilan Berhasil Konfirmasi Pesanan

Berdasarkan gambar diatas dapat dilihat bahwa pesanan berhasil dikonfirmasi terima dan sistem menampilkan halaman daftar transaksi.

#### 22. Menolak Pesanan

Pengujian tolak transaksi pesanan ini merupakan pengujian yang dilakukan ketika penjual ingin mengkonfirmasi transaksi pesanan. Hasil pengujian tolak pesanan dapat dilihat pada tabel di bawah ini.

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan untuk menolak pesanan
Yang diharapkan	Sistem menampilkan daftar pesanan yang membutuhkan konfirmasi
Pengamatan	Dapat menolak pesanan
Hasil	Sesuai

Pengujian menolak transaksi pesanan dilakukan setelah penjual membuka menu “Transaksi”, kemudian penjual mengklik tombol silang. Lalu sistem akan menampilkan pesan konfirmasi untuk mengkonfirmasi pesanan. Tampilan hasil pengujian menolak transaksi pesanan dapat dilihat pada Gambar di bawah ini.

Alasan Penolakan Pemesanan

bukti transfer tidak valid

Tolak Pesanan

Transaksi Pembelian

Show 10 entries	Search:			
Tanggal	Umkm	Total Pembelian	Status	Action
02-09-2023	Sanjai Oviga	18000	Selesai	<i>D</i> Detail
06-09-2023	Sanjai Oviga	59000	Selesai	<i>D</i> Detail



09-09-2023	Sanjai Oviga	21000	Expired	<i>D</i> Detail
10-09-2023	Sanjai Oviga	26000	Selesai	<i>D</i> Detail
10-09-2023	Sanjai Oviga	42000	Selesai	<i>D</i> Detail
12-09-2023	Sanjai Oviga	21000	Pesanan Ditolak	<i>D</i> Detail
24-08-2023	Sanjai Oviga	57000	Selesai	<i>D</i> Detail
25-08-2023	Sanjai Oviga	54000	Selesai	<i>D</i> Detail

Showing 1 to 9 of 9 entries

Previous 1 Next

Copyright © 2021. Premium Bootstrap admin template from BootstrapDash. All rights reserved.  
Distributed by Themewagon

Hand-crafted & made with



Berdasarkan gambar diatas dapat dilihat bahwa pesanan berhasil dikonfirmasi tolak dan sistem menampilkan halaman daftar pesanan.

### 23. Menginputkan Noresi Pesanan

Pengujian menginputkan noresi ini merupakan pengujian yang dilakukan oleh penjual ketika telah memproses pesanan dan status pesanan disiapkan. Hasil pengujian menginputkan noresi dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Noresi

Yang diharapkan	Sistem menampilkan halaman detail transaksi dan noresi
Pengamatan	Dapat menginputkan noresi pesanan
Hasil	Sesuai

Pengujian menginputkan noresi dilakukan setelah penjual mengkonfirmasi terima pesanan dan memproses pesanan, kemudian penjual mengakses halaman detail transaksi dan mengklik tombol “Tambah Noresi” lalu menginputkan noresi dan mengklik tombol “Submit”. Tampilan hasil pengujian menginputkan noresi dapat dilihat pada Gambar di bawah ini.

The screenshot shows the 'Detail Transaksi Pemesanan' page. It includes a summary section with the customer's name (Fadhillah Aini), date (02-09-2023), payment method (Bank Mandiri), and note (Pesanan Disiapkan). Below this is a photo of a BNI bank slip. Further down, it shows the transaction code (Kode Pemesanan: 20230902182453) and a detailed table of the purchase items:

No	Nama Produk	Kemasan	Jumlah	Harga Produk	Subtotal
1	Ganepo	sedang	1	Rp 10.000	Rp 10.000

Tampilan Detail Transaksi

The screenshot shows the 'Input Noresi' modal window. It has a single input field labeled 'Noresi' containing the value 'JP0035545342'. At the bottom of the modal are two buttons: 'Batal' (Cancel) and 'Simpan' (Save).

## Tampilan Form Input Noresi

Terima kasih telah melakukan pembelian.

### Ringkasan order

Umkm : Sanjai Oviga  
Tanggal Pembelian : 02-09-2023  
status : Pesanan Dikirim  
Noresi : JP0035545342

No	Gambar	Nama Produk	Jumlah	Harga Produk	Subtotal
1		Ganepo	1	Rp10.000,00	Rp10.000,00
Biaya Pengiriman: Rp8.000,00					
Total Harga: Rp18.000,00					

## Tampilan Lihat Noresi

Berdasarkan gambar diatas dapat dilihat bahwa menginputkan noresi berhasil dilakukan dan sistem menampilkan halaman detail transaksi.

### 24. Menginputkan Bukti Pengiriman

Pengujian menginputkan bukti pengiriman ini dilakukan oleh kurir untuk menyelesaikan pesanan. Hasil pengujian menginputkan bukti pengiriman dapat dilihat pada tabel di bawah ini.

#### Kasus dan Hasil Uji (Benar)

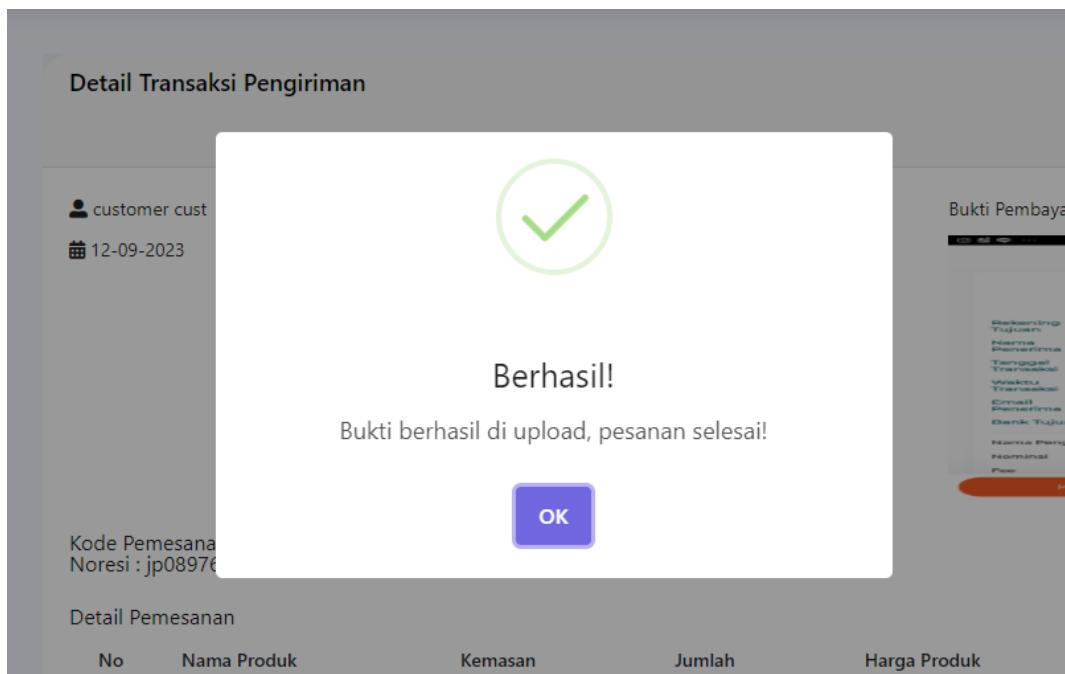
Data masukan	Foto bukti pengiriman
Yang diharapkan	Sistem menampilkan notifikasi bukti pengiriman berhasil ditambahkan dan pesanan selesai
Pengamatan	Dapat menginput bukti pengiriman
Hasil	Sesuai

Pengujian menginputkan bukti pengiriman dilakukan setelah penjual mengkonfirmasi terima pesanan dan memproses pesanan, menginputkan noresi kemudian kurir mengakses halaman detail pengiriman dan mengklik tombol “Input

Bukti Pengiriman” lalu sistem menampilkan form inputan. Tampilan hasil pengujian menginput bukti pengiriman dapat dilihat pada Gambar di bawah ini.

The screenshot shows the 'UMKM Sanjai' application interface. On the left, there's a sidebar with 'Dashboard', 'Pengiriman' (selected), and 'History'. The main area displays 'Detail Transaksi Pengiriman' for a customer named 'customer cust' on '12-09-2023'. It shows payment details ('Bank Mandiri', 'Pesanan Dikirim') and a QR code. Below this is a table for 'Detail Pemesanan' with one item: 'Jagung' (sedang) quantity 2 at Rp 12,000, totaling Rp 24,000. A blue button 'Input Bukti Pengiriman' is visible. A modal window titled 'Input Bukti Pengiriman' is open in the foreground, containing a file input field labeled 'Choose File' with 'No file chosen'. At the bottom of the modal are 'Batal' and 'Simpan' buttons. The background of the modal shows a blurred view of the transaction details.

Tampilan *Form* Inputan Bukti Pengiriman



### Tampilan Berhasil Menyelesaikan Pesanan

Berdasarkan gambar diatas dapat dilihat bahwa menginputkan buki pengiriman berhasil dilakukan dan sistem menampilkan halaman detail pengiriman.

#### 25. Mencetak Laporan Transaksi

Pengujian mencetak laporan transaksi ini merupakan pengujian yang dilakukan oleh penjual ketika ingin mencetak laporan. Hasil pengujian mencetak laporan dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk mencetak laporan transaksi
Yang diharapkan	Sistem menampilkan halaman cetak laporan
Pengamatan	Dapat melakukan pencetakan laporan
Hasil	Sesuai

Pengujian dilakukan ketika penjual mengakses menu “Laporan” lalu sistem menampilkan halaman form laporan, penjual menginputkan UMKM dan rentang tanggal transaksinya dan menekan tombol “Cetak”, maka laporan akan terdownload

pada perangkat komputer. Tampilan hasil pengujian mencetak laporan dapat dilihat pada Gambar di bawah ini.

**Screenshot 1: Cetak Laporan Transaksi**

Laporan transaksi pada umkm :

Laporan transaksi pada tanggal :

Aug 2023							Sep 2023						
Su	Mo	Tu	We	Th	Fr	Sa	Su	Mo	Tu	We	Th	Fr	Sa
30	31	1	2	3	4	5	27	28	29	30	31	1	2
6	7	8	9	10	11	12	3	4	5	6	7	8	9
13	14	15	16	17	18	19	10	11	12	13	14	15	16
20	21	22	23	24	25	26	17	18	19	20	21	22	23
27	28	29	30	31	1	2	24	25	26	27	28	29	30
3	4	5	6	7	8	9	1	2	3	4	5	6	7

08/01/2023 - 09/02/2023

### Tampilan Laporan

**Screenshot 2: Cetak Laporan Transaksi**

Nama Umkm : Sanjai Oviga  
Transaksi pada : 08/01/2023 - 09/02/2023

No	Tanggal pembelian	Pembeli	Produk	Kemasan	Jumlah	Harga	Subtotal	
1	24/08/2023	customer cust	Jagung	sedang	2	Rp 12.000	Rp 24.000	
						Biaya Pengiriman:	Rp 33.000	
						Total Harga	Rp 57.000	
2	25/08/2023	Dhea Irma Putri	Stik Ubi Jalar	250 gr	1	Rp 12.000	Rp 12.000	
						Biaya Pengiriman:	Rp 42.000	
						Total Harga	Rp 54.000	
3	01/09/2023	Fadhillah Aini	Kripik Lidi Balado	kecil	200 gr	2	Rp 8.000	Rp 16.000
						Rp 10.000	Rp 10.000	
						Biaya Pengiriman:	Rp 33.000	
						Total Harga	Rp 59.000	
4	02/09/2023	Fadhillah Aini	Ganepo	sedang	1	Rp 10.000	Rp 10.000	
						Biaya Pengiriman:	Rp 8.000	
						Total Harga	Rp 18.000	
						Total Pemasukan	Rp. 188.000,00	

**Screenshot 3: Laporan Transaksi Penjualan Produk**

Laporan Transaksi Penjualan Produk

Nama Umkm : Sanjai Oviga  
Transaksi pada : 08/01/2023 - 09/02/2023

No	Tanggal pembelian	Pembeli	Produk	Kemasan	Jumlah	Harga	Subtotal	
1	24/08/2023	customer cust	Jagung	sedang	2	Rp 12.000	Rp 24.000	
						Biaya Pengiriman:	Rp 33.000	
						Total Harga	Rp 57.000	
2	25/08/2023	Dhea Irma Putri	Stik Ubi Jalar	250 gr	1	Rp 12.000	Rp 12.000	
						Biaya Pengiriman:	Rp 42.000	
						Total Harga	Rp 54.000	
3	01/09/2023	Fadhillah Aini	Jagung Kripik Lidi Balado	kecil	200 gr	2	Rp 8.000	Rp 16.000
						Rp 10.000	Rp 10.000	
						Biaya Pengiriman:	Rp 33.000	
						Total Harga	Rp 59.000	
4	02/09/2023	Fadhillah Aini	Ganepo	sedang	1	Rp 10.000	Rp 10.000	
						Biaya Pengiriman:	Rp 8.000	

Tampilan Cetak Laporan

Berdasarkan gambar diatas dapat dilihat bahwa mencetak laporan berhasil dilakukan dan sistem menampilkan halaman laporan.

## 26. Melihat Riwayat Transaksi

Pengujian melihat riwayat transaksi ini merupakan pengujian yang dilakukan oleh *user* untuk melihat riwayat transaksi. Hasil pengujian melihat riwayat transaksi dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk melihat riwayat transaksi
Yang diharapkan	Sistem menampilkan halaman riwayat transaksi
Pengamatan	Dapat melihat riwayat transaksi
Hasil	Sesuai

Pengujian dilakukan ketika *user* mengakses halaman “History” lalu sistem menampilkan daftar transaksi yang pernah dilakukan oleh *user*. Tampilan hasil pengujian melihat riwayat transaksi dapat dilihat pada Gambar di bawah ini.

The screenshot shows the UMKM Sanjai application interface. On the left is a sidebar with navigation links: Dashboard, UMKM, Transaksi, History (which is highlighted in blue), Laporan, and Chat. The main content area has a title "History Transaksi Pembelian". Below it is a table with the following data:

Tanggal	Umkm	Nama Pembeli	Total Pembelian	Action
01-09-2023	Sanjai Oviga	Fadhillah Aini	59000	<i>i</i> Detail
02-09-2023	Sanjai Oviga	Fadhillah Aini	18000	<i>i</i> Detail
24-08-2023	Sanjai Oviga	customer cust	57000	<i>i</i> Detail
25-08-2023	Sanjai Oviga	Dhea Irma Putri	54000	<i>i</i> Detail

Tampilan Riwayat Transaksi

UMKM Sanjai

Detail History Transaksi Pemesanan

Dhea Irma Putri  
25-08-2023

Kode Pemesanan : 20230825150440

No	Nama Produk	Kemasan	Jumlah	Harga Produk	Subtotal
1	Stik Ubi Jalar	250 gr	1	Rp 12.000	Rp 12.000
				<b>Subtotal Produk:</b>	<b>Rp 12.000</b>
				<b>Biaya Pengiriman:</b>	<b>Rp 42.000</b>
				<b>Total Harga:</b>	<b>Rp 54.000</b>

### Tampilan Detail Riwayat Transaksi

Berdasarkan gambar diatas dapat dilihat bahwa melihat riwayat berhasil dilakukan dan sistem menampilkan halaman *history*.

#### 27. Menerima notifikasi UMKM Butuh Konfirmasi

Pengujian menerima notifikasi UMKM butuh konfirmasi ini merupakan pengujian yang dilakukan oleh admin ketika menerima notifikasi adanya UMKM yang melakukan permohonan perizinan. Hasil pengujian notifikasi dapat dilihat pada tabel di bawah ini.

<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk menerima notifikasi UMKM butuh konfirmasi
Yang diharapkan	Sistem menampilkan notifikasi jumlah UMKM butuh konfirmasi
Pengamatan	Dapat menerima notifikasi jumlah UMKM butuh konfirmasi
Hasil	Sesuai

Pengujian dilakukan ketika admin mengklik ikon notifikasi pada headbar sebelah kanan lalu sistem menampilkan jumlah UMKM yang membutuhkan

konfirmasi perizinan. Tampilan hasil pengujian notifikasi perizinan UMKM dapat dilihat pada Gambar di bawah ini.



### Tampilan Notifikasi Konfirmasi UMKM

Berdasarkan gambar diatas dapat dilihat bahwa menerima notifikasi berhasil dilakukan dan sistem menampilkan halaman UMKM butuh konfirmasi.

#### 28. Menerima Notifikasi Pesanan Baru

Pengujian menerima notifikasi pesanan baru ini merupakan pengujian yang dilakukan oleh penjual ketika menerima notifikasi adanya pesanan. Hasil pengujian notifikasi dapat dilihat pada tabel di bawah ini.

Kasus dan Hasil Uji (Benar)	
Data masukan	Seluruh data yang dibutuhkan untuk menerima notifikasi pesanan baru
Yang diharapkan	Sistem menampilkan notifikasi jumlah pesanan butuh konfirmasi
Pengamatan	Dapat menerima notifikasi jumlah pesanan butuh konfirmasi
Hasil	Sesuai

Pengujian dilakukan ketika penjual mengklik ikon notifikasi pada headbar sebelah kanan lalu sistem menampilkan jumlah pesanan yang membutuhkan konfirmasi. Tampilan hasil pengujian notifikasi pesanan baru dapat dilihat pada Gambar di bawah ini.



### Tampilan Notifikasi Pesanan Baru

Berdasarkan gambar diatas dapat dilihat bahwa menerima notifikasi pesanan berhasil dilakukan dan sistem menampilkan halaman transaksi.

## 29. Logout

Pengujian *logout* ini merupakan pengujian yang dilakukan oleh *user* ketika ingin keluar dari aplikasi. Hasil pengujian *logout* dapat dilihat pada tabel di bawah ini.

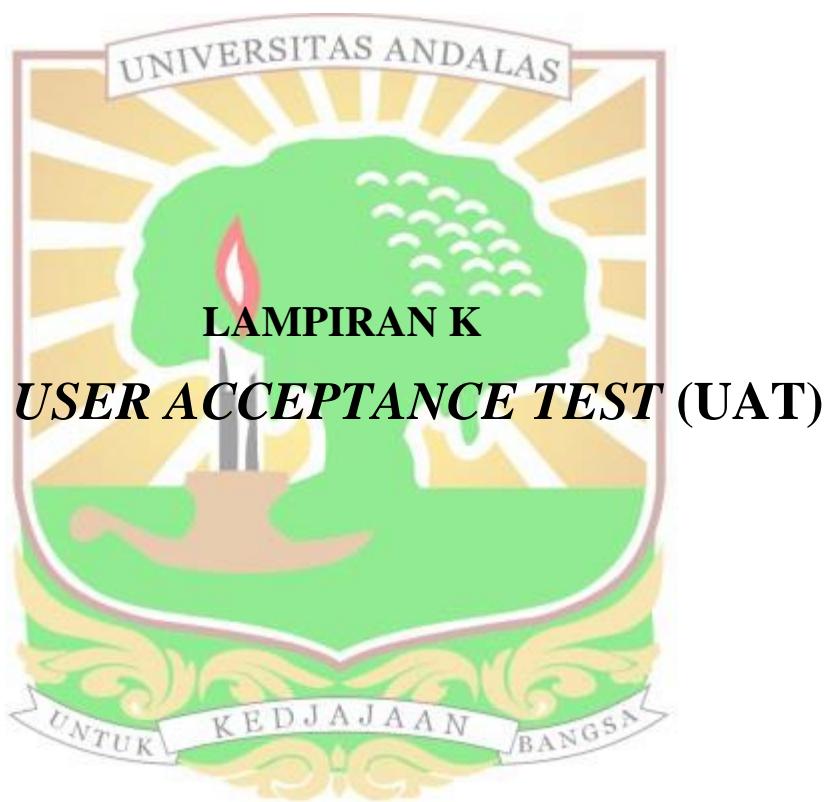
<b>Kasus dan Hasil Uji (Benar)</b>	
Data masukan	Seluruh data yang dibutuhkan untuk <i>logout</i>
Yang diharapkan	Sistem menampilkan halaman <i>landing page</i>
Pengamatan	Dapat <i>logout</i>
Hasil	Sesuai

Pengujian dilakukan ketika *user* telah *login* kedalam aplikasi. Kemudian *user* mengklik tombol “Logout” dan sistem akan memprosesnya sehingga *user* keluar dari aplikasi dan kembali ke halaman *landing page*. Tampilan hasil pengujian *logout* dapat dilihat pada Gambar di bawah ini.



Tampilan Setelah Logout

Berdasarkan gambar diatas dapat dilihat bahwa *logout* berhasil dilakukan dan sistem menampilkan halaman *landing page*.



Dilakukan penyebaran *form user acceptance test* untuk melihat kesimpulan dari penggunaan aplikasi yang telah dibangun. UAT dilakukan kepada pelaku usaha sebagai pemilik UMKM dan kepada pembeli sebagai pihak yang akan membeli produk pada aplikasi. Dengan skala 1-5, semakin tinggi skala semakin menunjukkan bahwa pihak yang mengisi *form* sangat sesuai dengan pernyataan yang diajukan dan semakin rendah skala menunjukkan bahwa pengisi *form* tidak sesuai dengan pernyataan yang diberikan.

### 1. Hasil UAT pihak penjual

Nama Penjual  
5 jawaban

Vina kurnia  
Sanah (putri chaniago)  
Gadis (3 saudara)  
Nirina (sanjai nirina)  
Indah (sanjai zaifa)

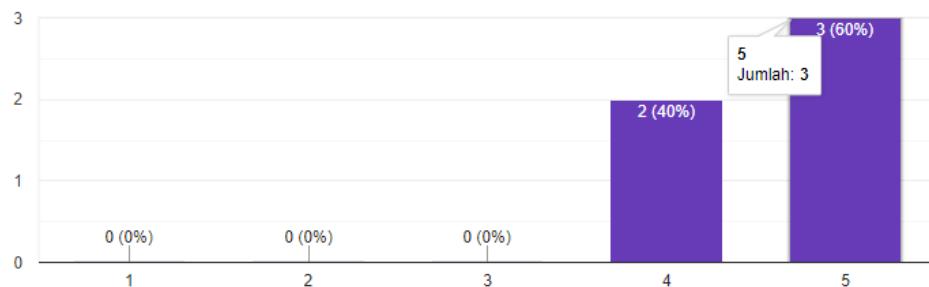
Menurut anda, apakah tampilan aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini sudah jelas dan tidak membingungkan? Salin  
5 jawaban

Rating	Persentase
1	0%
2	0%
3	20%
4	40%
5	40%

Menurut anda, apakah menu-menu dari aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini mudah dipahami?

Salin

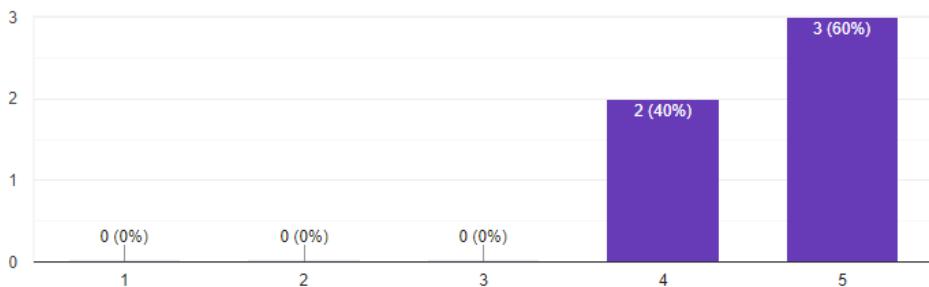
5 jawaban



Menurut anda, apakah dengan adanya aplikasi ini dapat membantu anda ketika ingin menjual produk pada UMKM anda?

Salin

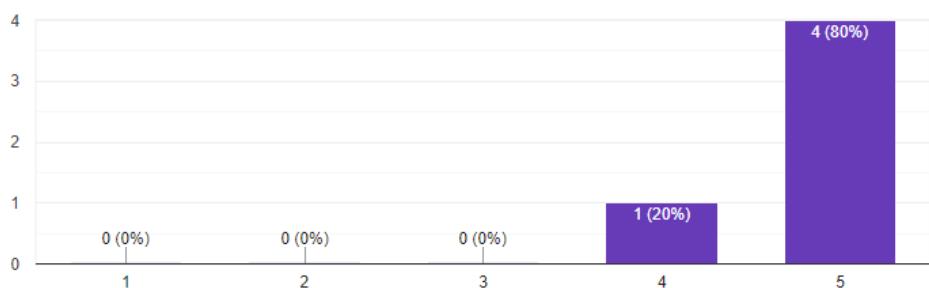
5 jawaban



Menurut anda, apakah dengan adanya aplikasi ini lebih membantu dalam mempromosikan produk usaha yang anda miliki?

Salin

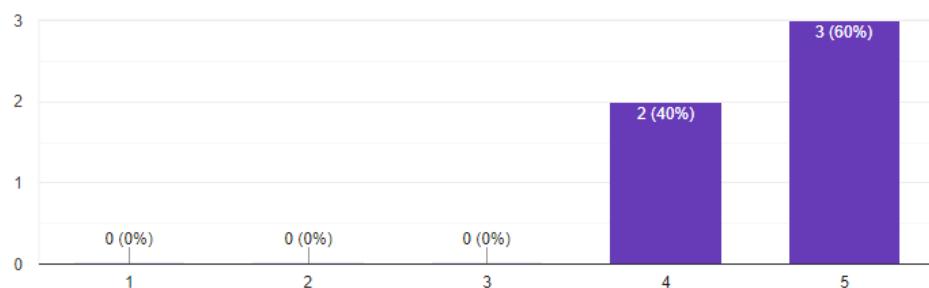
5 jawaban



Menurut anda, apakah aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan memberikan kemudahan dalam pengelolaan transaksi secara online?

Salin

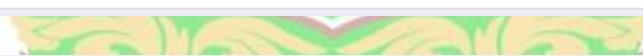
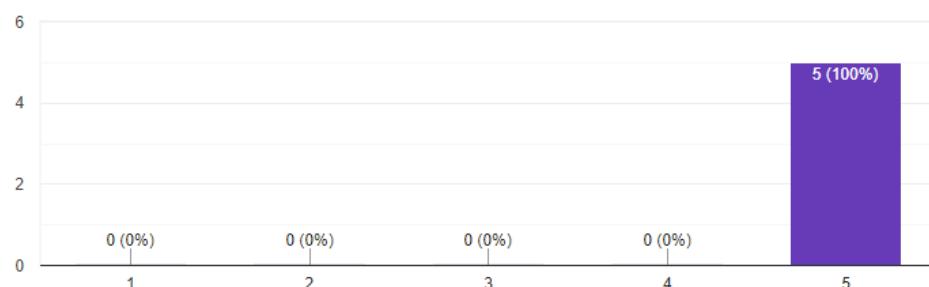
5 jawaban



Menurut anda, apakah aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini dapat membantu menghemat waktu dan biaya dalam proses transaksi?

Salin

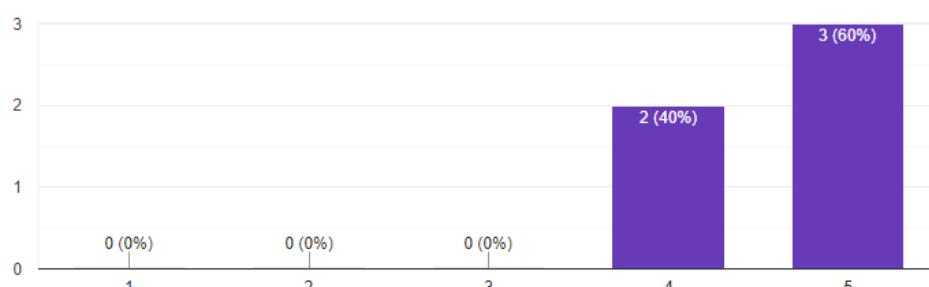
5 jawaban



Menurut anda, apakah perlu adanya aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini bagi para pelaku usaha di Kota Payakumbuh?

Salin

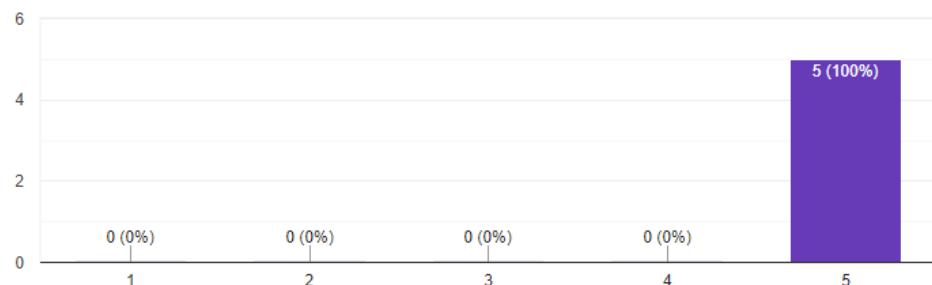
5 jawaban



Menurut anda, apakah dengan adanya aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan pada Kota Payakumbuh ini dapat membantu anda dalam menjangkau konsumen seluas-luasnya?

 Salin

5 jawaban



Dari hasil diatas, dapat disimpulkan bahwa pelaku usaha setuju bahwa dengan adanya aplikasi ini dapat membantu promosi produk yang mereka jual dan juga dapat membantu mereka dalam memasarkan produk seluas-luasnya.

## 2. Hasil UAT pihak pembeli

Nama Pembeli

4 jawaban

Annisha Aprillya

Dhea

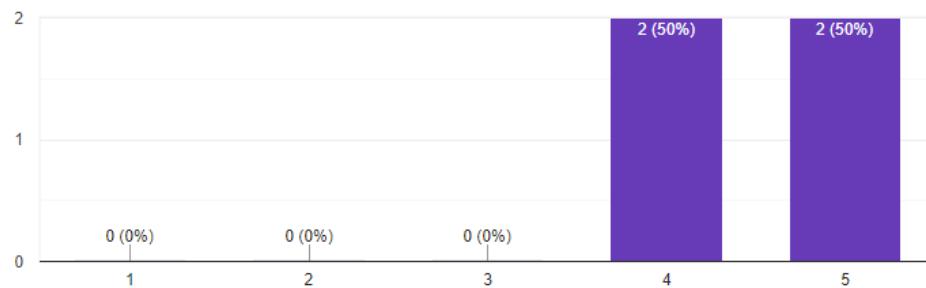
Lathif Nur Irsyad

Aini

Menurut anda, apakah tampilan aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan di Payakumbuh ini sudah jelas dan tidak membingungkan?

Salin

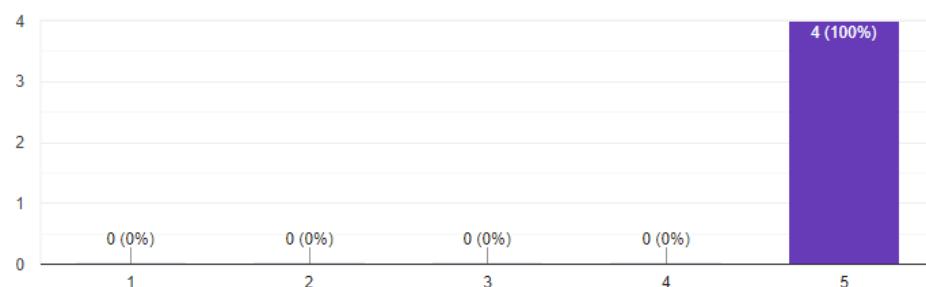
4 jawaban



Menurut anda, apakah menu-menu dari aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini mudah dipahami?

Salin

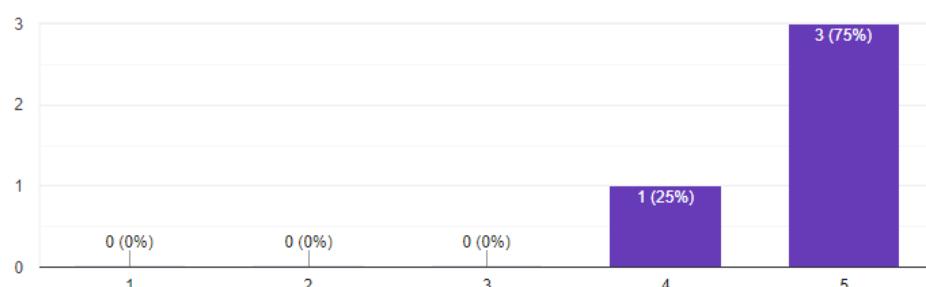
4 jawaban



Menurut anda, apakah dengan adanya aplikasi ini dapat membantu anda ketika ingin membeli produk sanjai?

Salin

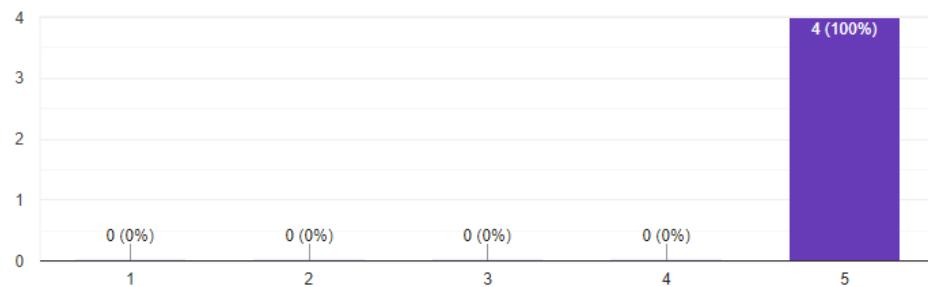
4 jawaban



Menurut anda, apakah dengan adanya aplikasi ini dapat menghemat biaya dan waktu dalam proses transaksi?

Salin

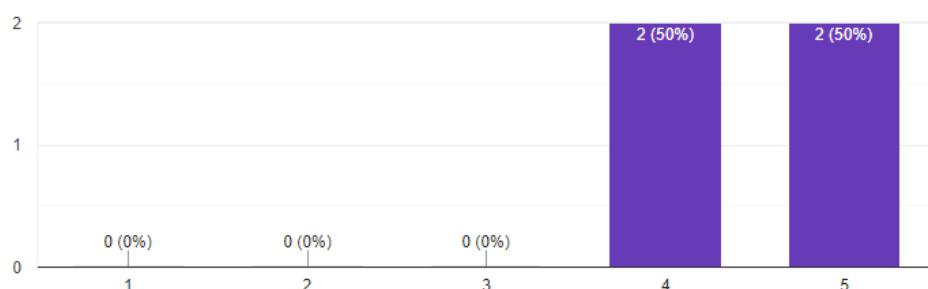
4 jawaban



Menurut anda, apakah aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini memberikan kemudahan dalam proses pembelian produk?

Salin

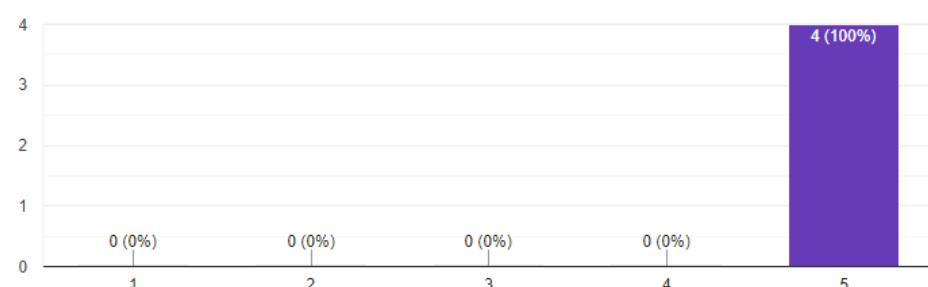
4 jawaban



Menurut anda, apakah aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini telah memberikan informasi yang cukup mengenai produk-produk pada toko tersebut?

Salin

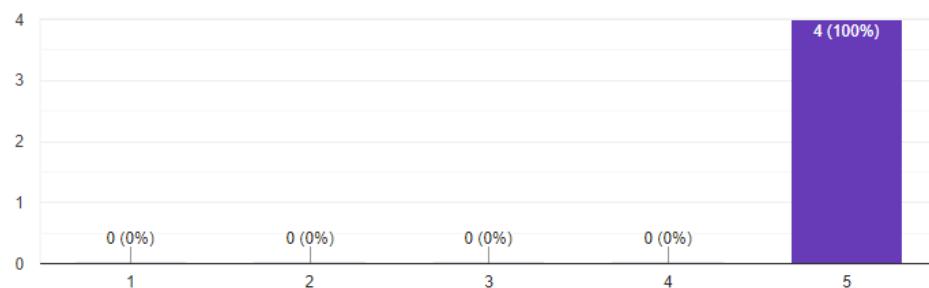
4 jawaban



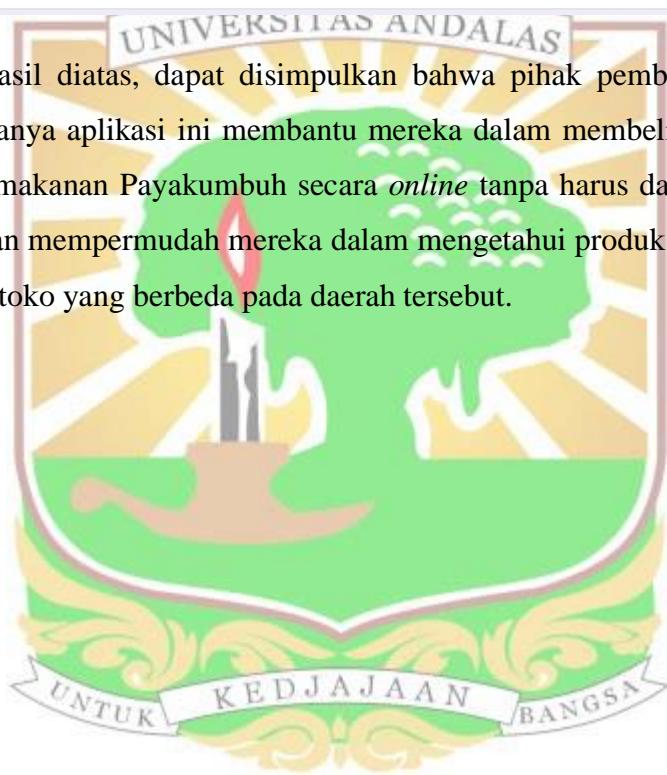
Menurut anda, apakah perlu adanya aplikasi dengan model bisnis marketplace pada UMKM oleh-oleh makanan ini bagi para pembeli produk makanan khas minang?

Salin

4 jawaban



Dari hasil diatas, dapat disimpulkan bahwa pihak pembeli setuju bahwa dengan adanya aplikasi ini membantu mereka dalam membeli produk UMKM oleh-oleh makanan Payakumbuh secara *online* tanpa harus datang langsung ke tokonya dan mempermudah mereka dalam mengetahui produk-produk yang ada pada toko-toko yang berbeda pada daerah tersebut.



PEMBANGUNAN APLIKASI DENGAN MODEL BISNIS  
MARKETPLACE UNTUK MEMBANTU PEMASARAN OLEH-OLEH  
MAKANAN PADA UMKM KOTA PAYAKUMBUH

ORIGINALITY REPORT

<b>13%</b>	<b>13%</b>	<b>2%</b>	<b>2%</b>
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	<a href="#">scholar.unand.ac.id</a> Internet Source	7%
2	<a href="#">www.scribd.com</a> Internet Source	1%
3	<a href="#">teknosi.fti.unand.ac.id</a> Internet Source	1%
4	<a href="#">123dok.com</a> Internet Source	1%
5	<a href="#">repo.unand.ac.id</a> Internet Source	1%
6	<a href="#">doku.pub</a> Internet Source	1%
7	<a href="#">dspace.uii.ac.id</a> Internet Source	1%
8	<a href="#">repository.usd.ac.id</a> Internet Source	1%