Technical Documentation

MITRAX

September, 2018

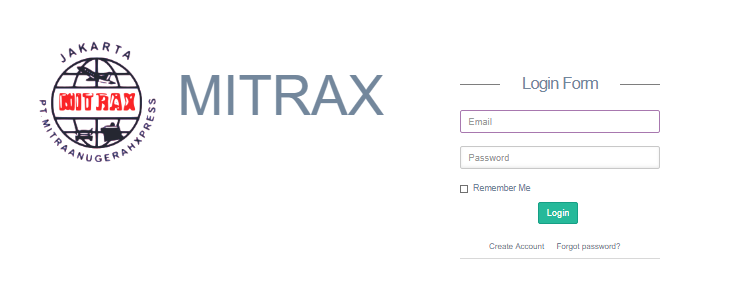
|  |  |  |
| --- | --- | --- |
| Prepared by | Signature | Date |
| Meilan Anastasia Maharani  [Technical Writer] |  |  |

|  |  |  |
| --- | --- | --- |
| Approved by | Signature | Date |
| [ARRY NOVIANTO]  [Director] |  |  |

# FRONT END SPECIFICATION

## Login

* + - Layout

****

* + - Penjelasan Proses

1. Url : /login
2. Halaman php yang diload : “login.blade.php” berada pada folder : “resources/views/auth”
3. User mengisikan kolom-kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Email | *Varchar* |  | 191 | Email |
| 2 | password | *Varchar* |  | 191 | Password untuk login (tampildalam asterisk \*) |
| 3 | CaptchaCode |  |  |  | Kode Captcha |

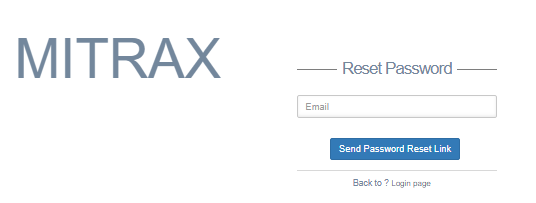
1. Tombol LOGIN diklik, maka akan memanggil fungsi “validateLogin” pada file “LoginController.php” di folder “app/Http/Controllers/Auth” dan menghasilkan result :

* Jika login sukses lanjut kehalaman “/home” atau dashboard.
* Jika login gagal akan menampilkan message “Login/Password Salah. Silahkan ulangi lagi”. Form login tetap tampil.

1. Link Forgot Password jika diklik akan memanggil form Reset Password
2. Link Create New Account jika diklik akan memanggil form Create Account

## Forgot Password

* + - Layout

****

* + - Penjelasan Proses

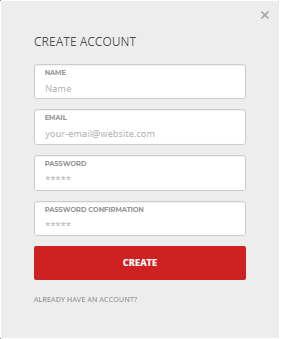
1. Link forgot password terdapat pada file “login.blade.php” dengan memanggil fungsi “#forgot\_panel”, fungsi tersebut meload form dengan “id= forgot\_panel” pada file “layout\_without\_menu.blade.php” di file “resources/views/front/”
2. Fungsi “forgot\_panel” menampilkan form popup forgot passwordPada form tersebut User mengisikan kolom berikut:

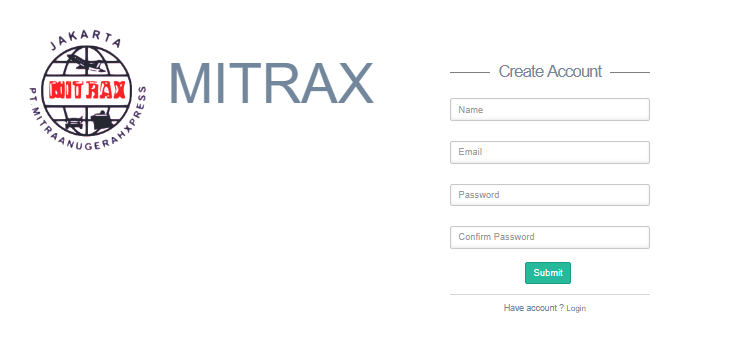
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Email | *Varchar* |  | 191 | Email |

1. Saat tombol “SEND PASSWORD RESET” diklik, action diarahkan ke halaman email.blade.php pada folder “resources/views/password/”
2. Tombol “Send Password Reset Link” diklik maka akan menggunakan fungsi “SendsPasswordResetEmails” yang berada pada folder “vendor/laravel/framework/src/Illuminate/Foundation/Auth/” dan mengirimkan password baru ke email

## Create New Account

* + - Layout





* + - Penjelasan Proses

1. Pada file login.blade.php, link Create New Account diarahkan ke “#register\_panel”, fungsi “#register\_panel" bisa dilihat di file “layout\_without\_menu.blade.php” di file “resources/views/front/” dan cari form dengan id =”register\_panel”.
2. Pada form yang diload tersebut, User mengisikan kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Name | *Varchar* |  | 191 | Nama user |
| 2 | email | *Varchar* |  | 191 | Email User |
| 3 | password | *Varchar* |  | 191 | Password untuk login |
| 4 | confirm\_password |  |  |  | Konfirmasi password , isi harus sama dengan password |

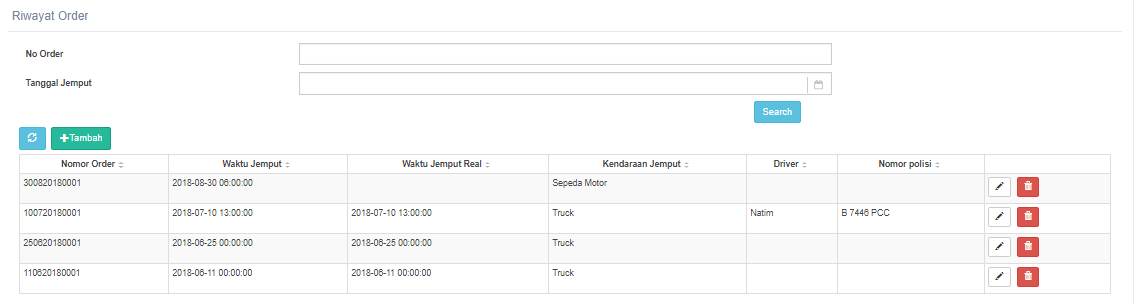
1. Tombol Submit diklik maka akan memanggil url “register” dan menampilkan result

* Jika konfirmasi password tidak cocok, maka akan menampilkan pesan : **Konfirmasi password tidak cocok.**
* **Apabila email sudah pernah regsitrasi maka akan menampilkan pesan : Isian email sudah digunakan.**

**Jika berhasil registrasi maka otomatis akan masuk ke halaman /home atau dashboard**

## Order Jemput

* + - Layout



* + - Penjelasan Proses

url : /penjemputan

untuk mengetahui file yang akan diload Kita bisa cek di folder routes/web.php, Kita bisa mencari untuk url penjemputan diarahkan kemana. Pada file ini bisa Kita lihat:

*Route::get('/penjemputan','Penjemputan\PenjemputanController@page');* artinya saat membuka url : /penjemputan, sistem akan meload file app/Http/Controllers/Penjemputan/PenjemputanController.php dengan fungsi “page”

fungsi “page”, memanggil fungsi “JenisBarang” dan “JenisKendaraan” untuk mengambil data jenis barang dan jenis kendaraan, kemudian data tersebut ditampilkan pada halaman views, yaitu di file “resouces/views/pages/penjemputan.blade.php”

Setelah halaman penjemputan tampil, User dapat memilih tab antara lain Order Jemput, Jadwal Jemput dan Daftar Driver & Kendaraan.

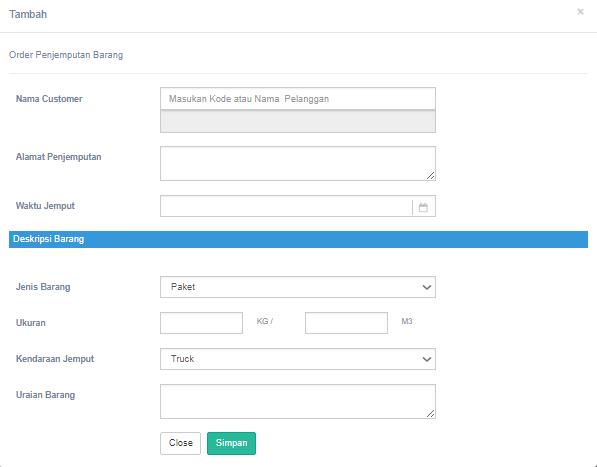
Untuk tab Order Jemput atau default, meload halaman “penjemputanOrderJemput.blade.php” yang berada di folder “resouces/views/panels/ penjemputanOrderJemput.blade.php”

Pada halaman tab Order Jemput, User dapat melakukan pencarian list riwayat order dengan mengisikan kolom No Order dan Tanggal Jemput

Klik tombol “Tambah” maka akan menampilkan form Tambah order penjemputan

## Form Order Penjemputan

* + - Layout

****

* + - Penjelasan Proses

File yang diload : “resouces/views/panels/ penjemputanOrderJemput.blade.php” dengan memanggil API “penjemputan/order”

Setelah halaman form penjemputan tampil, User dapat melakukan pengisian form :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| nama\_customer | *Varchar* |  | 191 | Nama customer / agent (autocomplete)  Memanggil API “customersearch” |
| alamat\_penjemputan | *text* |  |  | Alamat customer / agent |
| waktu\_jemput\_request | *timestamp* |  |  | Permintaan waktu penjemputan  (tanggal dan jam) |
| jenis\_barang\_id | *int* |  | 11 | Jenis barang, combo box dengan pilihan:  Paket  Dokumen |
| ukuran\_kg | *int* |  | 12 | Ukuran atau berat barang(Berupa estimasi jika customer belum ) |
| ukuran\_m3 | *int* |  | 8 | Ukuran atau volume barang(Berupa estimasi jika customer belum mengukur barang) |
| jenis\_kendaraan\_id |  |  |  | Jenis kendaraan yang digunakan untuk menjemput , dengan pilihan berupa combo box:   * Truck * Pickup * Sepeda Motor |
| uraian\_barang | *text* |  |  | Uraian barang |

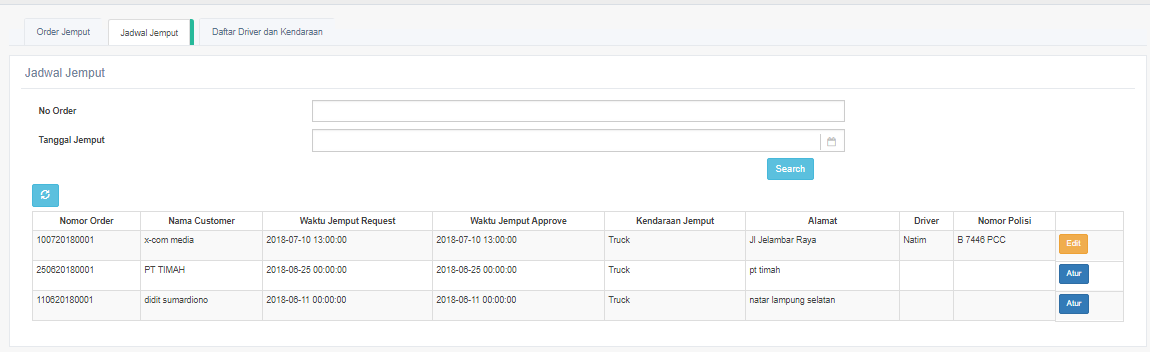
Tombol simpan diklik maka akan memanggil “vm.submitModal()” dan akan menampilan result :

Jika berhasil : akan menampilkan pop up success dan menampilkan output berupa riwayat order dengan data antara lain:

* Nomor order, dengan format : DDMMYYXXXX
* DD=Tanggal order
* MM=Bulan order
* YY=Tahun order
* XXXX = Nomor urut order . (Nomor diul ang ketika bergant i tahun)
* Nama customer / agent
* Alamat customer / agent
* Waktu penjemputan request (tanggal dan jam)
* Waktu penjemputan real
* Kendaraan yang digunakan
* Nomor polisi kendaraan
* Nama driver
* Nomor telpon driver

## Penjadwalan Penjemputan

* + - Layout



* + - Penjelasan Proses

File yang diload : “resouces/views/panels/ penjemputanJadwalJemput.blade.php”

File “penjemputanJadwalJemput.blade.php” menginclude url :

* api/penjemputan/order
* api/penjemputan/driver
* api/penjemputan/order\_arrange

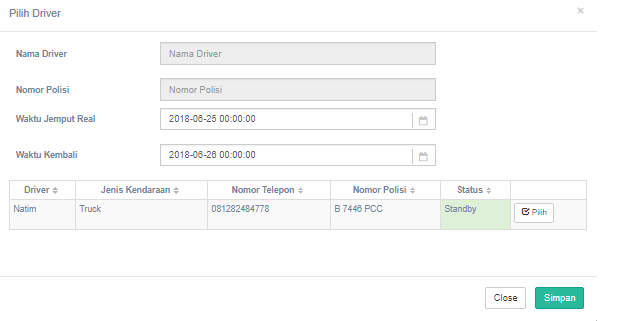
Pada halaman Jadwal Jemput, User melakukan pencarian data dengan mengisi kolom berikut ini

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nomor\_order | *char* |  | 12 | Nomor order |
| 2 | waktu\_jemput\_real | *timestamp* |  |  | Tanggal jemput |

* Ketika tombol search diklik maka akan tampil list jadwal jemput dibawah dengan elemen data antara lain :
* Nomor order
* Nama customer
* Permintaan Waktu jemput(tanggal dan jam)
* Waktu jemput yang diapprove
* Alamat penjemputan
* Jenis kendaraan
* Nomor Polisi kendaraan
* Nama dan Nomor HP driver
* Tombol Edit atau atur diklik maka akan memanggil form pilih driver

## Form Pilih Driver

* Layout



* + - Penjelasan Proses

File yang diload : “resouces/views/panels/ penjemputanJadwalJemput.blade.php” dengan memanggil fungsi “vm.show(row)”

User klik tombol pilih pada daftar driver yang standby

Ketika tombol pilih diklik, maka akan memanggil fungsi “vm.pilihDriver(row)” dan akan menampilkan nama driver (penanggung\_jawab) dan nomor polisi pada form pilih driver.

Setelah tampil, User mengisikan kolom-kolom :

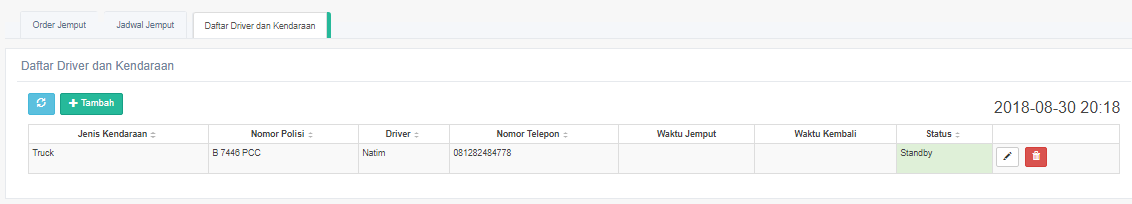
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | penanggung\_jawab | *varchar* |  | 191 | Nama driver (otomatis tampil setelah pilih driver) |
| 2 | nomor\_polisi | *varchar* |  | 191 | Nomor polisi  (otomatis tampil setelah pilih driver) |
| 3 | waktu\_jemput\_real | *timestamp* |  |  | Waktu Jemput |
| 4 | waktu\_kembali\_real | *timestamp* |  |  | Waktu kembali |

Tombol simpan diklik, maka akan memanggil function “vm.submit()”

Tombol close diklik, maka akan menutup form

## Daftar Driver dan Kendaraan

* + - Layout



* + - Penjelasan Proses

File yang diload : “resouces/views/panels/ penjemputanDaftarDriver.blade.php”

User klik tombol “Tambah”, maka akan memanggil fungsi “vm.add()” (backend) lalu membuka form driver kendaraan

User klik tombol “Refresh”, maka akan memanggil fungsi “vm.refresh()”

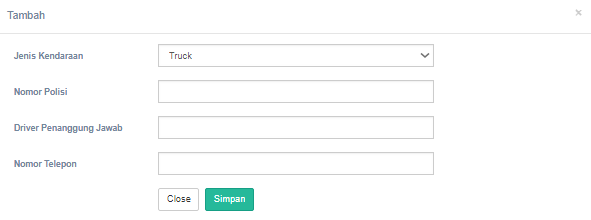
User klik tombol icon pencil pada salah satu daftar driver, maka akan memanggil fungsi “vm.edit(row)” dan menampilkan popup form driver dan kendaraan

User klik tombol icon hapus pada salah satu daftar driver, maka akan memanggil fungsi “vm.del (row)” dan akan muncul pop up dengan pesan “are you sure?”

* Jika klik tombol “Yes” maka akan memanggil fungsi hapus
* Jika klik tombol “No” maka akan kembali menampilkan daftar driver dan kendaraan.

## Form Driver Kendaraan

* + - Layout



* + - Penjelasan Proses

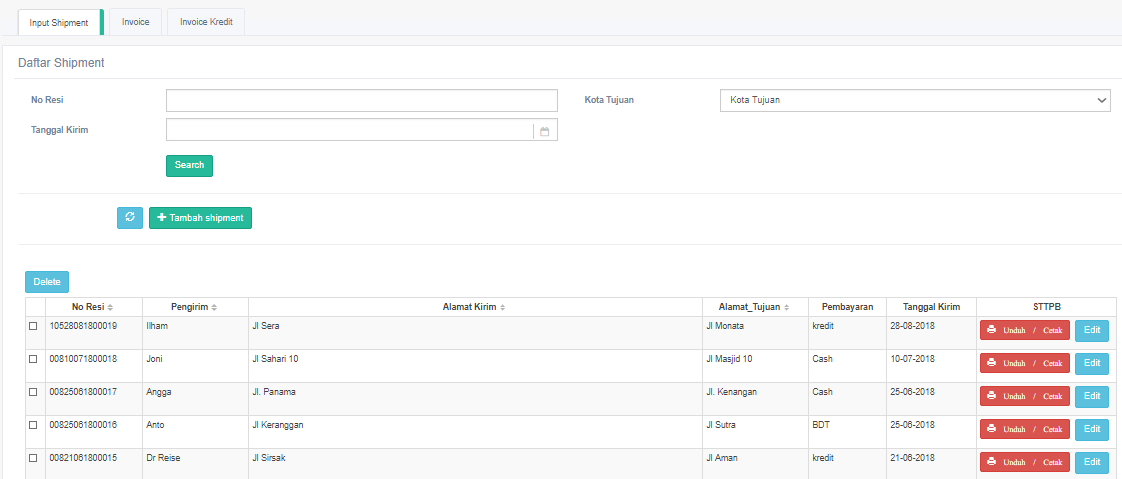
1. File yang diload : “resouces/views/panels/ penjemputanDaftarDriver.blade.php” dengan memanggil fungsi “vm.add()” atau vm.edit(row)”
2. Isi dari Fungsi “vm.add()” dan “vm.edit(row)” dapat dilihat di folder : public/js/angular\_js/controllers dengan nama file PenjemputanControlllers.js
3. User mengisikan kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | jenis\_kendaraan\_id | *int* |  | 11 | Jenis Kendar aan berupa combo box dengan pilihan:   * Truk * Pickup * Sepeda Mot or |
| 2 | nomor\_polisi | *Varchar* |  | 191 | Nomor Polisi Kendaraan |
| 3 | penanggung\_jawab | *Varchar* |  | 191 | Driver penanggung jawab |
| 4 | nomor\_telepon | *Varchar* |  | 191 | Nomor Telepon |

1. Tombol Simpan diklik, maka akan memanggil fungsi “vm.submitModal “
2. Tombol Close diklik, maka akan menutup popup

## Input Shipment

* + - Layout



* + - Penjelasan Proses

1. Url yang dipanggil : [/page/shipment](http://192.168.5.233/home#/page/shipment)
2. untuk menentukan file php yang dipanggil, Kita bisa melihat dari routes.js yang berada di folder public/js/angular\_js/routes.js
3. Setelah dicek, untuk url page/shipment Kita bisa cek kembali di folder routes/web.php, Kita bisa mencari untuk shipment diarahkan kemana. Pada file ini bisa Kita lihat:

Route::get('/shipment' , 'Shipment\ShipmentController@index');

Maka dari itu halaman shipment meload file app/Http/Controllers/Shipment/ShipmentController.php dengan fungsi index

1. Fungsi tersebut meload halaman view pages/shipment.blade.php
2. Setelah halaman shipment tampil, User dapat melakukan pencarian list dengan mengisikan kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 5 | No Resi |
| 2 | Tanggal | *Varchar* |  | 191 | Tanggal Kirim |
| 3 | Tujuan\_Kabupaten | *Int* |  | 11 | Kota Tujuan berupa combo box |

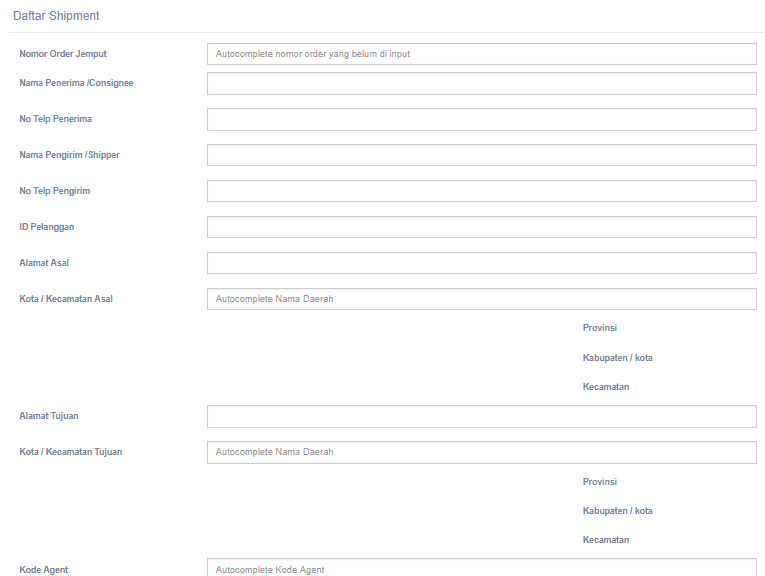
1. Kota Tujuan memanggil API “tujuanvillages” (backend) dan menampilkan list berupa kota tujuan
2. Tombol “Search” diklik, maka akan memanggil fungsi Cari dan menampilkan daftar Shipment
3. Tombol “Tambah Shipment“ diklik maka akan memanggil fungsi “inputshipment” dan menampilkan form daftar shipment
4. Tombol “Hapus” diklik maka akan memanggil API “deteteshipment” dan menampilkan popup “Are You Sure?”

* Jika klik tombol “Yes” maka akan memanggil fungsi hapus
* Jika klik tombol “No” maka akan kembali menampilkan daftar shipment

1. Tombol Unduh / Cetak diklik maka akan memanggil cetakSTTPB dan menampilkan file STTPB untuk diprint
2. Tombol “Edit” diklik maka akan memanggil fungsi “editshipemt” dan menampilkan form Edit shipment

## Form Daftar / Edit Shipment

* Layout



* + - Penjelasan Proses

1. Untuk input shipment, url yang dibuka adalah : /inputshipment yaitu meload file app/Http/Controllers/Shipment/ShipmentController.php dengan fungsi “inputshipment”, pada fungsi tersebut memanggil views inputShipmentform.blade.php di folder resources/views/Shipment/ inputShipmentform.blade.php
2. Untuk edit shipment, url yang dibuka adalah : / editshipemt yaitu meload file app/Http/Controllers/Shipment/ShipmentController.php dengan fungsi “editshipemt”, pada fungsi tersebut memanggil views editshipment.blade.php di folder resources/views/Shipment/ editshipment.blade.php
3. Setelah halaman form shipment tampil, User mengisikan kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | id | *int* |  | 10 | Kode |
| 2 | no\_resi | *char* |  | 15 | No Resi pengiriman |
| 3 | Nomor\_Order\_Jemput | *char* |  | 12 | No order penjemputan |
| 4 | Nama\_Penerima | *varchar* |  | 191 | Nama penerima |
| 5 | No\_Telp\_Penerima | *varchar* |  | 191 | No Telpon Penerima |
| 6 | Nama\_Pengirim | *varchar* |  | 191 | Nama Pengirim |
| 7 | No\_Telp\_Pengirim | *varchar* |  | 191 | No Telpon pengirim |
| 8 | ID\_Pelanggan | *int* |  | 11 | Kode pelanggan |
| 9 | Alamat\_Asal | *varchar* |  | 191 | Alamat asal pengiriman |
| 10 | Asal\_Provinsi | *int* |  | 11 | Kode provinsi asal pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 11 | Asal\_Kabupaten | *int* |  | 11 | Kode kabupaten asal pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 12 | Asal\_kecamatan | *int* |  | 11 | Kode kecamatan asal pengiriman. Berupa autocomplete. |
| 13 | Asal\_Desa | *int* |  | 11 | Kode desa asal pengiriman |
| 14 | Asal\_Kode\_Pos | *int* |  | 11 | Kode pos asal pengiriman |
| 15 | Alamat\_Tujuan | *varchar* |  | 191 | Alamat tujuan pengiriman |
| 16 | Tujuan\_Provinsi | *int* |  | 11 | Kode provinsi tujuan pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 17 | Tujuan\_Kabupaten | *int* |  | 11 | Kode kabupaten tujuan pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 18 | Tujuan\_kecamatan | *int* |  | 11 | Kode kecamatan tujuan pengiriman. Berupa autocomplete |
| 19 | Tujuan\_Desa | *int* |  | 11 | Kode desa tujuan pengiriman |
| 20 | Tujuan\_Kode\_Pos | *int* |  | 11 | Kode pos tujuan pengiriman |
| 21 | Kode\_Agent | *int* |  | 11 | Kode Agen pengirim. Berupa autocomplete agen |
| 22 | Kode\_Transit | *int* |  | 11 | Kode transit |
| 23 | Metode\_Pembayaran | *varchar* |  | 191 | Metode pembayaran. Berupa combo box dengan pilihan :Cash, Kredit, bayar di tujuan |
| 24 | Jalur\_Angkut | *varchar* |  | 191 | Jalur angkut pengiriman. Berupa combo box dengan pilihan : Darat, Laut, Udara |
| 25 | Jenis\_Barang | *varchar* |  | 191 | Jenis barang yang dikirim. Berupa combo box dengan pilihan : Paket dan Dokumen |
| 26 | Jumlah\_Barang | *varchar* |  | 191 | Jumlah barang yang dikirim berupa combo box dengan pilihan jumlah |
| 27 | Jenis\_kemasan | *varchar* |  | 191 | Jenis kemasan paket yang dikirim |
| 28 | Nama\_Barang | *varchar* |  | 191 | Nama barang yang dikirim |
| 29 | Berat\_Barang | *varchar* |  | 191 | Berat barang yang dikirim |
| 30 | Harga\_Barang | *int* |  | 11 | Harga barang yang dikirim |
| 31 | Volume | *int* |  | 11 | Volume barang yang dikirim |
| 32 | Dimensi\_Barang\_P | *int* |  | 11 | Dimensi barang (panjang) |
| 33 | Dimensi\_Barang\_L | *int* |  | 11 | Dimensi barang (lebar) |
| 34 | Dimensi\_Barang\_T | *int* |  | 11 | Dimensi barang (tinggi) |
| 35 | Berat\_Dibayar | *int* |  | 11 | Berat yang harus dibayar |
| 36 | Layanan\_Tambahan | *varchar* |  | 191 | Layanan tambahan yang dipilih. Berupa check box dengan pilihan :   * Packing Kayu / peti * Paking Khusus * Asuransi * Jemput |
| 37 | Instruksi\_Khusus | *varchar* |  | 191 | Instruksi khusu |
| 38 | Biaya\_Pengiriman | *int* |  | 11 | Biaya pengiriman barang |
| 39 | Keterangan\_Barang | *text* |  |  | Keterangan barang |
| 40 | Diskon | *int* |  |  | Diskon yang diberikan. Berupa combo box dengan beberapa pilihan diskon dalam persentase. |
| 41 | Biaya\_Asuransi | *int* |  |  | Biaya asuransi yang dikenakan |
| 42 | Biaya\_Packing | *int* |  |  | Biaya packing |
| 43 | biaya\_surchange | *int* |  |  | Biaya surchange |
| 44 | biaya\_Penjemputan | *int* |  |  | Biaya penjemputan |
| 45 | Sub\_Total | *int* |  |  | total |
| 46 | PPN | *int* |  |  | PPN |
| 47 | Sub\_Biaya | *int* |  |  | Sub biaya |

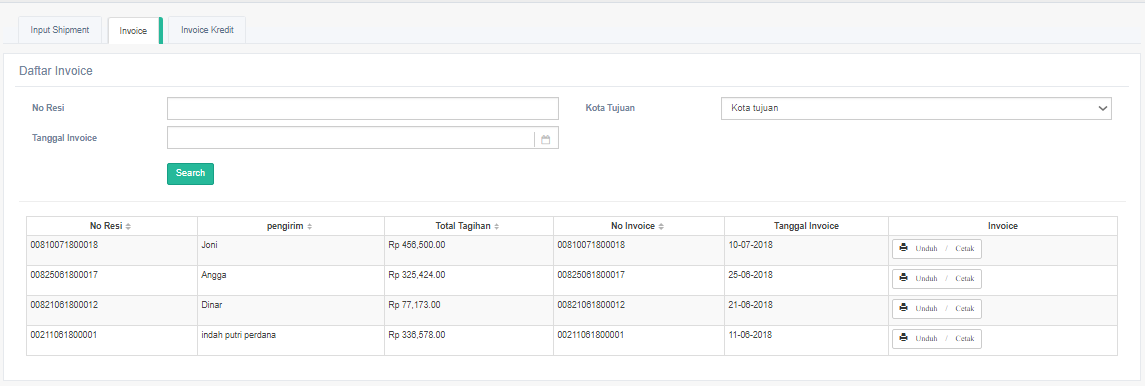
1. No order penjemputan memanggil API “**noorder**” dan menampilkan nomor order yang belum diinput
2. Kota / Kecamatan Asal memanggil API “wilayah” dan Kota / Kecamatan Tujuan memanggil API “tujuan”
3. Biaya Sub Total memanggil API “hitungsubtotal” dan otomatis menampilkan biaya Sub Total
4. Tombol “Simpan” diklik, maka akan memanggil fungsi “vm.submitModal()” dan menampilkan output berupa Tabel daftar shipment yang telah dientry, dengan elemen data:

* Nomor resi
* Nama pengirim
* Kota pengirim
* Kota tujuan
* Metode pembayaran
* Tanggal kirim (tanggal system ketika proses simpan entry data)
* Dokumen Surat Tanda Terima Pengiriman Barang(STTPB) dengan rincian elemen data sebagai berikut :
* STTPB Lembar pertama, dengan elemen data:
* Nomor resi
* Format nomor resi adalah sebagai berikut :
* AAADDMMYYXXXXX
* AAA: 3 digit angka kode kota tujuan, sesuai dengan master area( modul FR. 1. 24)
* DD : 2 digit tanggal terbit STTPB
* MM: 2digit bulan terbit STTPB
* YY : 2digit terakhir tahun terbit STTPB
* XXXXX: 5digit nomor urut resi(dimulai dari 00001). Nomor tersebut di mulai dari 00001 setiap berganti tahun
* Tanggal cetak STTPB
* Nama penerima/ consignee
* Nomor telpon pengirim
* Alamat pengirim
* Nama pengirim/ shipper
* Nomor Telpon shipper
* Alamat pengirim
* Call sign kota asal(tergenerate otomatis berdasarkan kota asal)
* Call sign kota tujuan (tergenerate otomatis berdasarkan kota tujuan)
* Kode agent
* Kode transit
* Cara pembayaran:
* Cash
* Credit
* Jenis pengangkutan(darat / udara)
* Jumlah barang
* Harga barang
* Jenis barang:
* Paket
* Dokumen
* Layanan tambahan, dengan pilihan:
* Packing kayu/ peti
* Packing khusus
* Asuransi
* Pilihan barang dijemput atau diantar
* Jenis kemasan
* Berat barang(dalam satuan kg)
* Volume barang
* Ukuran barang(panjang, lebar , tinggi)
* Berat dibayar
* Instruksi khusus
* Rincian biaya, dengan elemen data:
* Biaya pengiriman
* Biaya packing
* Biaya asuransi
* Diskon
* Subtotal
* PPN
* Diskon
* Total biaya
* Keterangan barang
* STTPB lembar kedua, dengan elemen data sama dengan lembar pertama, namun tanpa rincian biaya.
* STTPB lembar ketiga, dengan elemen data sama dengan lembar pertama, namun tanpa rincian biaya.
* STTPB lembar keempat, dengan elemen data sama dengan lembar pertama, namun tanpa rincian biaya.
* STTPB lembar kelima, dengan elemen data sama dengan lembar pertama, namun tanpa rincian biaya.

1. Sedangkan edit, maka akan memanggil API “editshit” (backend)

## Invoice

* Layout



* Penjelasan proses

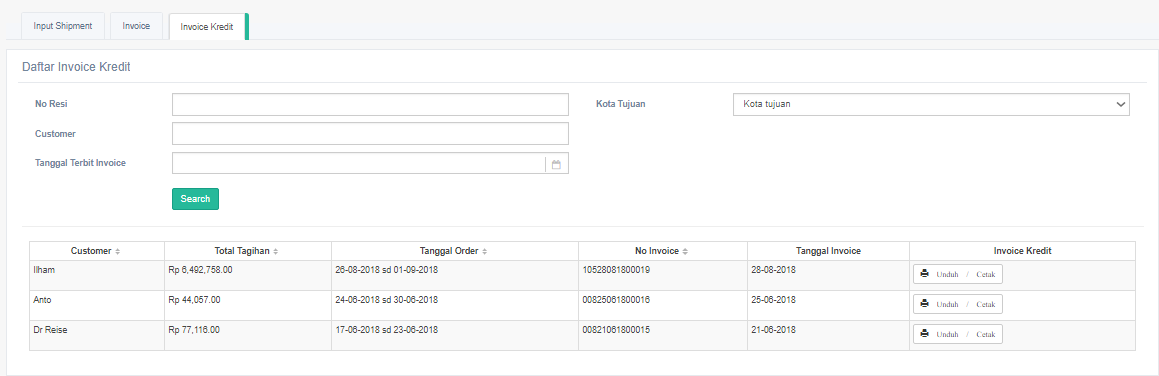
1. Untuk melakukan pencarian, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 5 | No Resi |
| 2 | Tanggal | *Date* |  |  | Tanggal Invoice |
| 3 | Tujuan\_Kabupaten | *Int* |  | 11 | Kota Tujuan berupa combo box |

1. Tombol “Search” diklik, maka akan memanggil fungsi “Cari” dan menampilkan daftar Invoice
2. Tombol “ Unduh / Cetak” pada daftar invoice diklik, maka akan memanggil fungsi Cetak dan menampilkan lembar invoice pada halaman baru

## Invoice Kredit

* Layout



* Penjelasan proses

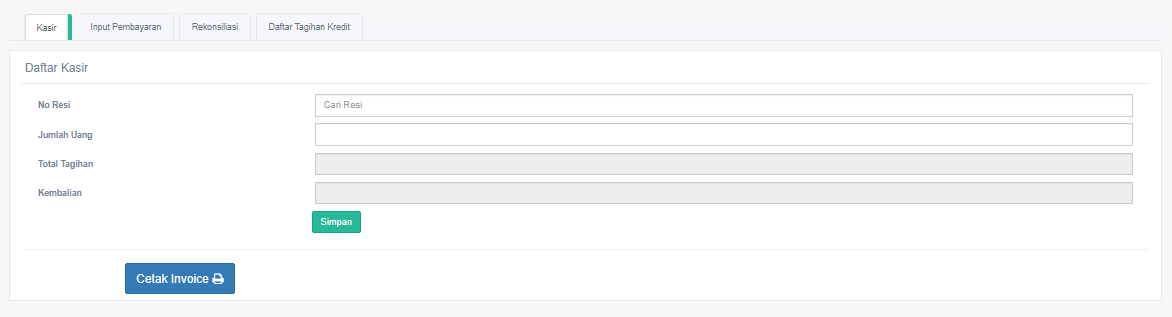
1. Untuk melakukan pencarian, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 5 | No Resi |
| 2 | Customer | *string* |  |  | Nama customer |
| 3 | Tanggal | *Date* |  |  | Tanggal invoice |
| 3 | Tujuan\_Kabupaten | *Int* |  | 11 | Kota Tujuan berupa combo box |

1. Tombol “Search” diklik, maka akan memanggil fungsi Cari dan menampilkan daftar Invoice kredit
2. Tombol “ Unduh / Cetak” pada daftar invoice diklik, maka akan memanggil fungsi Cetak dan menampilkan lembar invoice kredit pada halaman baru

## Form Kasir

* Layout



* Penjelasan Proses

Url yang diload : /pembayaran

Berasarkan routing di routes/web.php, yaitu *Route::get('/pembayaran', 'MasterData\PembayaranController@page')*;

File php yang dibuka : PembayaranController.php dan memanggil fungsi page, yang berada di folder app/Http/Controllers/MasterData/ PembayaranController.php

Fungsi page memanggil file view Pembayaran.blade.php yang berada di folder resources/views/pages/Pembayaran.blade.php

Pada halaman tersebut terdapat pilihan tab antara lain :

* Kasir

include file kasir.blade.php yang berada di folder resources/views/pembayaran/kasir.blade.php

* Input Pembayaran

include file pembayaran.blade.php yang berada di folder resources/views/pembayaran/ pembayaran.blade.php

* Rekonsiliasi

include file Rekonsiliasi.blade.php yang berada di folder resources/views/pembayaran/ Rekonsiliasi.blade.php

* Daftar Tagihan Kredit

include file daftarTagihan.blade.php yang berada di folder resources/views/pembayaran/ daftarTagihan.blade.php

Pada tab halaman Form Kasir, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 15 | No Resi  menggunakan autosuggest daftar nomor resi |
| 2 | jumlah\_dibayar | *int* |  | 20 | Jumlah uang yang dibayarkan |

No resi berupa auto suggest yang memanggil API “noresipembayaran”

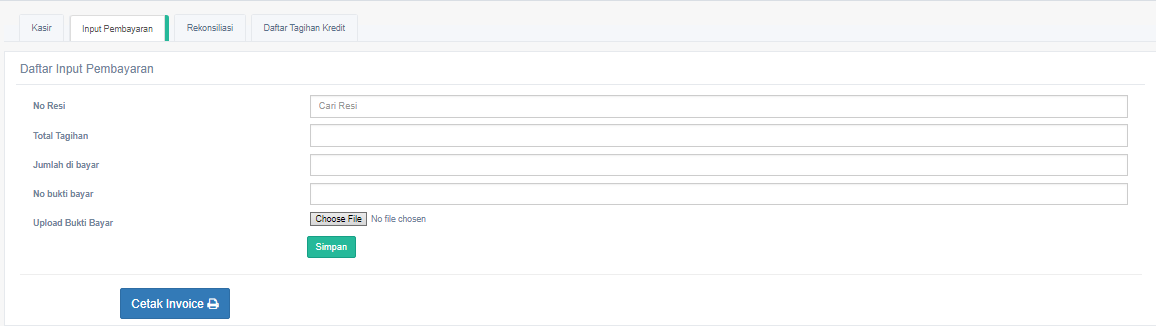
Pada saat input jumlah uang pembayaran, sistem memanggil fungsi “vm.hitungkembalian()”

Tombol simpan diklik maka akan memanggil fungsi “vm.add()”

Tombol Cetak Invoice diklik maka akan memanggil fungsi “vm.cetakkasir('{{url('cetakInvoice')}}” dan menampilkan invoice pada halaman baru

## Input Pembayaran

* Layout



* Penjelasan Proses

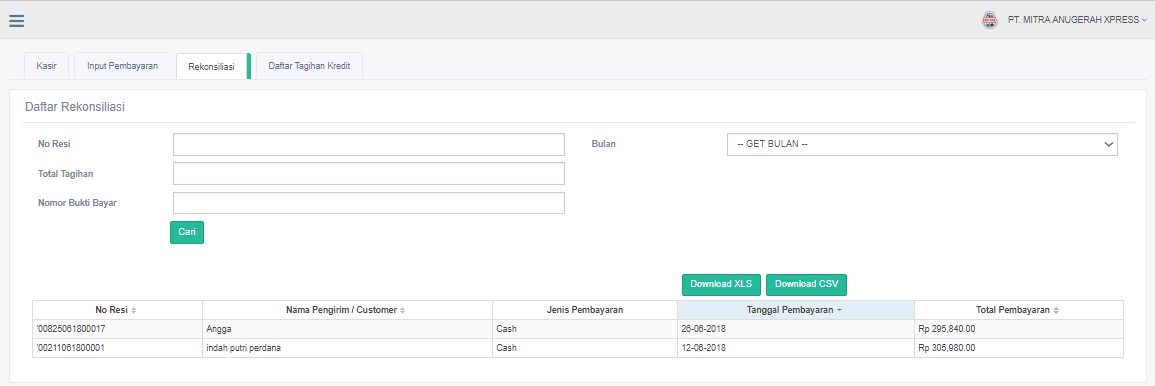
1. File yang diload : resources/views/pembayaran/pembayaran.blade.php
2. user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 15 | No Resi  menggunakan autosuggest daftar nomor resi |
| 2 | jumlah\_total\_tagihan | *int* |  | 11 | Total tagihan |
| 3 | jumlah\_uang\_pembayaran | *int* |  | 11 | Jumlah dibayar |
| 4 | upload | *varchar* |  | 191 | Upload bukti bayar(jika ada)  Biasanya berupa gambar |

1. saat input no resi memanggil APIi “noresiinputpembayarankredit”
2. user klik tombol simpan , maka akan memanggil fungsi “vm.add()”
3. user klik tombol “Cetak Invoice” maka akan memanggil fungsi “vm.cetakkasir('{{url('cetakInvoice')}}'” dan menampilkan invoice pada halaman baru, url 'cetakInvoice' memanggil file invoiceController.php yang berada di app/Http/Controllers/ dengan fungsi “cetak”

## Rekonsiliasi Pembayaran

* Layout



* Penjelasan Proses

1. Form Rekonsiliasi meload file php : “resources/views/pembayaran/Rekonsiliasi.blade.php”
2. Untuk melakukan pencarian, user mengisi kolom antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 15 | No Resi |
| 2 | jumlah\_total\_tagihan | *int* |  | 11 | Total tagihan |
| 3 | No bukti bayar | *string* |  |  | No bukti pembayaran |
| 4 | Bulan |  |  |  | Bulan yang dipilih |

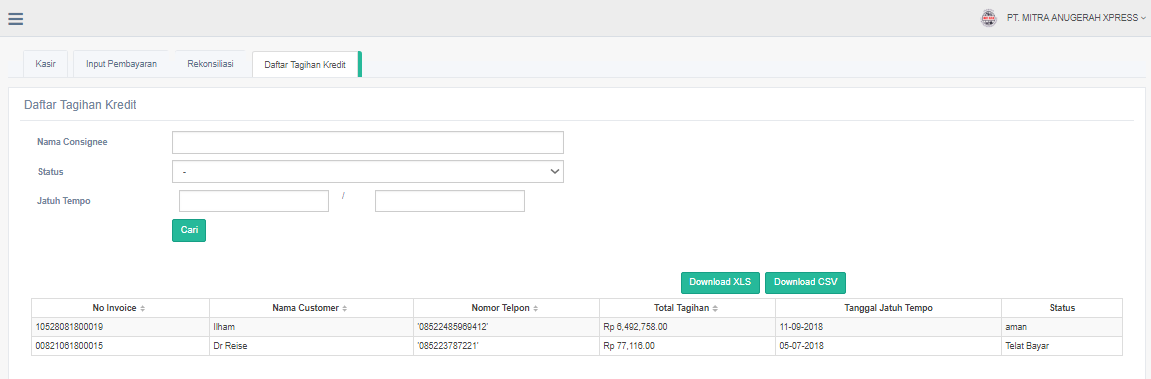
1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch” dan menampilkan daftar rekonsiliasi berupa :

* Nomor invoice
* Nama customer / pengirim
* Jenis pembayaran
* Tanggal pembayaran
* Nilai yang dibayarkan
* Total nilai pembayaran dari seluruh invoice

1. Tombol “Download XLS” diklik maka akan memanggil fungsi “ngTableExcelExport.excel” dan mendowload file rekonsiliasi.xls
2. Tombol “Download CSV” diklik maka akan memanggil fungsi “ngTableExcelExport.csv” dan mendowload file rekonsiliasi.csv

## Daftar Tagihan Kredit

* Layout



* Penjelasan Proses

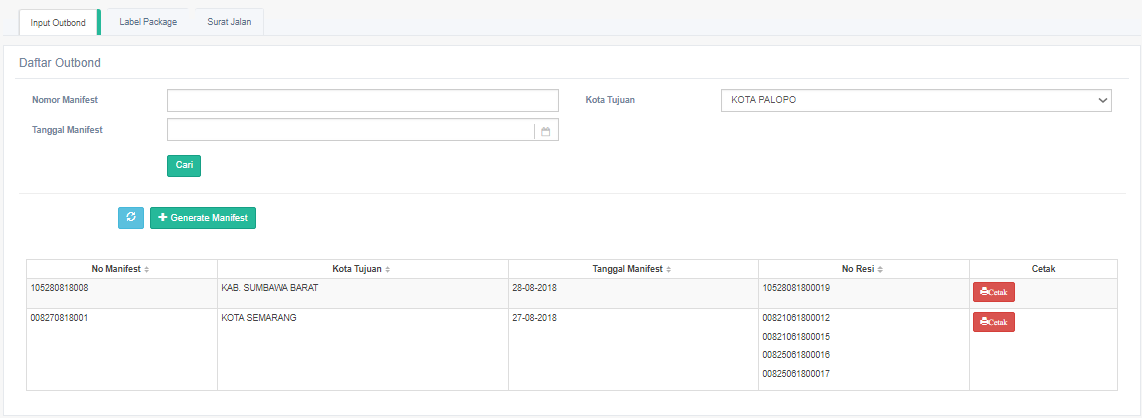
1. Untuk form daftar tagihan kredit meload file php : ”resources/views/pembayaran/daftarTagihan.blade.php”
2. Untuk melakukan pencarian, user mengisi kolom antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nama\_Pengirim | *varchar* |  | 191 | Nama customer / pengirim |
| 2 | Status |  |  |  | Status tanggal jatuh tempo.  Berupa combo box dengan pilihan : aman, segera bayar, Jatuh Tempo, Telat Bayar |
| 3 | Jatuh Tempo |  |  |  | Rentang tanggal jatuh tempo |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch” dan menampilkan daftar tagihan kredit
2. Tombol “Download XLS” diklik maka akan memanggil fungsi “ngTableExcelExport.excel” dan mendowload file rekonsiliasi.xls
3. Tombol “Download CSV” diklik maka akan memanggil fungsi “ngTableExcelExport.csv” dan mendowload file rekonsiliasi.csv

## Form Input Outbond

* Layout



* Penjelasan proses

Url yang diload : / outbond

Berasarkan routing di routes/web.php, yaitu *Route::* Route::get('/outbond', 'MasterData\OutbondController@page');

1. File php yang dibuka : OutbondController.php dan memanggil fungsi page, yang berada di folder app/Http/Controllers/MasterData/ OutbondController.php
2. Fungsi page memanggil file view Outbond.blade.php yang berada di folder resources/views/pages/ Outbond.blade.php
3. Pada halaman tersebut terdapat pilihan tab antara lain :

* Input Outbond

include file inputOutbond.blade.php yang berada di folder resources/views/outbond/ inputOutbond.blade.php

* Label Package

include file Package.blade.php yang berada di folder resources/views/ outbond / Package.blade.php

* Surat Jalan

include file Suratjalan.blade.php yang berada di folder resources/views/ outbond / Suratjalan.blade.php

1. Pada tab halaman Input Outbond user dapat melakukan pencarian daftar outbpnd :
2. Untuk melakukan pencarian, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest | varchar |  | 191 | No Manifest |
| 2 | Tanggal | varchar |  | 191 | Tanggal Manifest |
| 3 | Kota | varchar |  | 191 | Kota tujuan |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch” dan menampilkan daftar Outbond dengan elemen:

* Nomor manifest , dengan format sebagai berikut
* Xxxddmmyynnn
* xxx : kode kota tujuan(sesuai master area)
* dd : dua digit tanggal pembuatan manifest
* mm : dua digit bulan pembuatan manifest
* y y : dua digit tahun pembuatan manifest
* nnn :tiga digit nomor urut manifest .

Nomor urut setiap kota berbeda dengan kota lain.

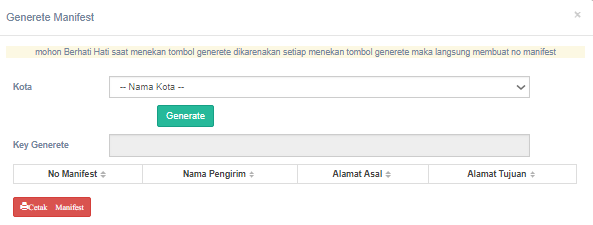
Sehingga nomor urut kota tidak saling tercampur dengan kota lain.

* Nama kota tujuan
* Tanggal pembuatan manifest
* No Resi

1. Tombol “Generate Manifest” diklik, maka akan menampilkan form Generate Manifest
2. Tombol Cetak diklik, maka akan memanggil fungsi “vm.gg()” dan menampilkan halaman cetak manifest

## Form Generate Manifest

* Layout



* Penjelasan proses

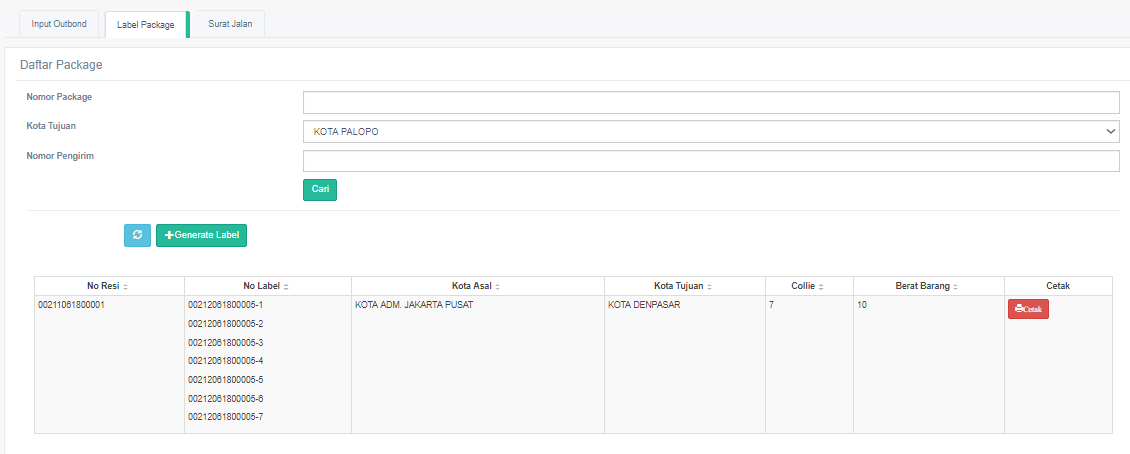
1. Form generate manifest ditampilkan saat user klik tombol “Generate Manifest” pada input Outbond, yaitu memanggil fungsi vm.tambah()
2. Untuk generate manifest, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Kota | varchar |  | 191 | Kota tujuan  (berupa combo box) |

1. Tombol “Generate” diklik, maka akan memanggil fungsi “vm.GenereteManifest()” dan membuat manifest
2. Modul mengambil data-data nomor resi dengan kota tujuan yang sama.
3. Tombol “Cetak Manifest” diklik maka akan memanggil fungsi “vm.gg()” dan menampilkan Manifest

## Label Package

* Layout



* Penjelasan proses

1. Label package menampilkan views Package.blade.php yang berada di folder resources/views/ outbond / Package.blade.php
2. Untuk melakukan pencarian, user harus mengisi kolom berikut :

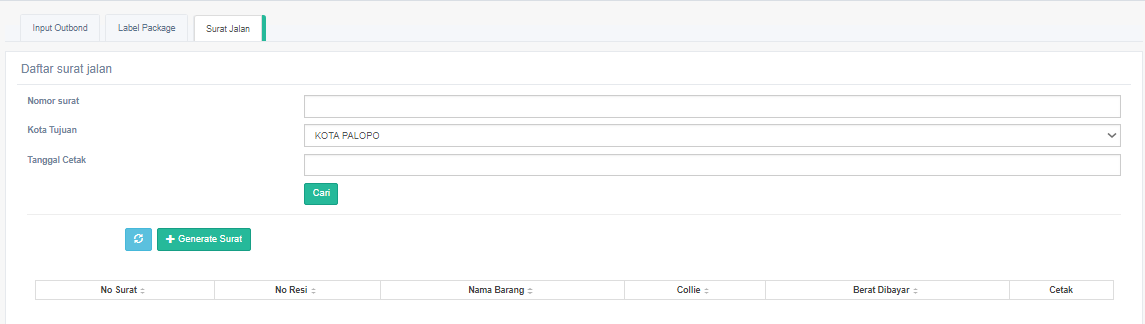
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nopackage | varchar |  | 191 | Nomor Package |
| 2 | Tujuan\_Kabupaten | int |  | 11 | Kota Tujuan |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan Daftar Package
2. Tombol dengan icon “refresh” maka akan memanggil fungsi ” vm.refresh()” dan memperbarui daftar package
3. Tombol “Generate Label” diklik, maka akan memanggil fungsi “vm.tambah()”

* Generate label paket. Nomor label paket di-*generate* dengan format : xxxddmmyynnnnnz
* xxx : kode kota tujuan( berdasarkan master area)
* dd : t anggal generate label
* mm : bulan generate label
* yy : t ahun generate label
* nnnnn : nomor urut label. Nomor urut setiap kota berbeda dengan kota lain. Sehingga nomor urut kota tidak saling tercampur dengan kota lain.
* Z : nomor seri label .

## Surat Jalan

* Layout



* Penjelasan Proses

1. Saat tab Surat Jalan dipilih, file php yang diload adalah Suratjalan.blade.php yang berada di folder resources/views/ outbond / Suratjalan.blade.php
2. Untuk melakukan pencarian, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_suratjalan | char |  | 15 | Nomor surat |
| 2 | Tujuan\_Kabupaten | int |  | 11 | Kota Tujuan |
| 3 | tanggal | Date |  |  | Tanggal cetak |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan Daftar Surat jalan
2. Tombol dengan icon “refresh” maka akan memanggil fungsi ” vm.refresh()” dan memperbarui daftar surat jalan
3. Tombol “Generate Surat” diklik, maka akan memanggil fungsi “vm.tambah()”
4. Cetak surat jalan memanggil fungsi “vm.cetaksurat('{{url('suratjalan')}}”

Untuk melihat detailnya, bisa Kita telusuri dengan membuka file routes/web.php, perhatikan :

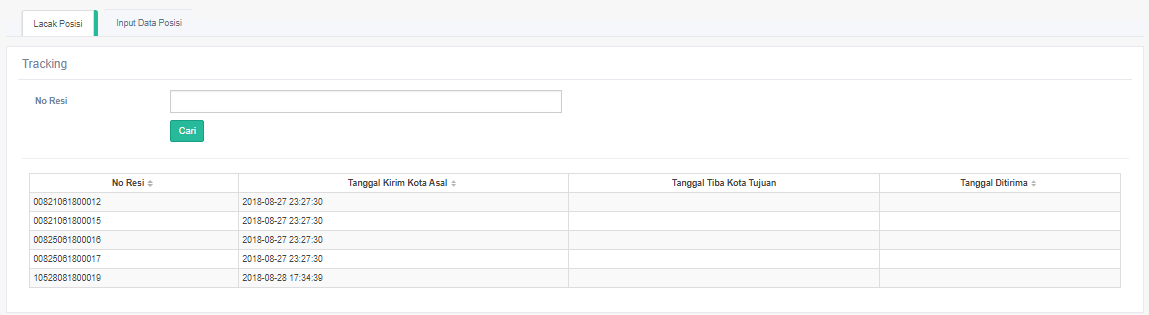
Route::get('/suratjalan/{id\_surat}' , 'cetak\cetakController@suratjalan');

Artinya fungsi tersebut memanggil fungsi “suratjalan” yang berada di file cetakController.php di folder app/Http/cetak/

1. Pada fungsi “suratjalan” tersebut, memerintahkan untuk meload views “suratjalan.blade.php” yang berada di folder resources/views/cetak/

## Lacak Posisi

* Layout



* Penjelasan Proses

1. Alamat url : /tracking
2. Sesuai dengan routing berikut :

Route::get('/tracking', 'MasterData\trackingController@page');

Halaman tersebut meload fungsi “page” di file “trackingController .php” yang berada di folder “app/Http/Controllers/MasterData/”

1. Pada fungsi “page” tersebut, memerintahkan untuk load file “tracking.blade.php” yang berada di “resources/views/pages/”
2. Pada halaman tersebut terdapat pilihan tab antara lain :

* Lacak Posisi

include file “tracking.blade.php” yang berada di folder resources/views/tracking/”

* Input Data Posisi

include file “inputtracking.blade.php” yang berada di folder resources/views/tracking/”

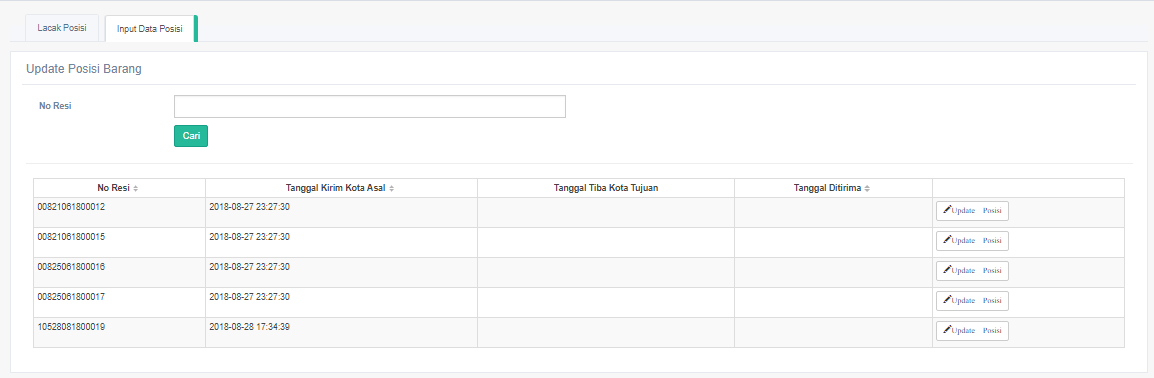
1. Ketika tab lacak posisi di klik, maka akan meload halaman “tracking.blade.php” yang menampilkan daftar posisi
2. Untuk melacak posisi, user harus mengisi kolom No resi :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan informasi paket yang dikirim

## Input Data Posisi

* Layout



* Penjelasan Proses

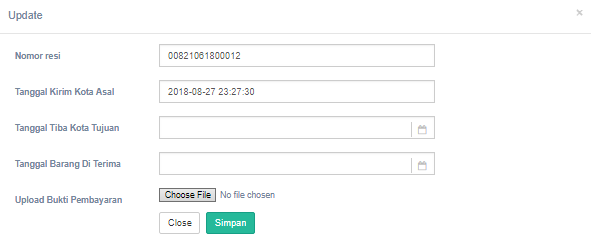
1. Sesuai penjelasan proses dari halaman “lacak posisi”, tab “input data posisi” include file “inputtracking.blade.php” yang berada di folder resources/views/tracking/”
2. Untuk memudahkan pencarian posisi barang yang hendak diupdate, user dapat mengisi kolom No resi :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan informasi paket yang dikirim
2. Klik tombol “Update Posisi” pada data yang hendak diupdate, maka akan memanggil fungsi “vm.edit(row)” dan menampilkan form update posisi barang

## Form Update Posisi Barang

* Layout



* Penjelasan Proses

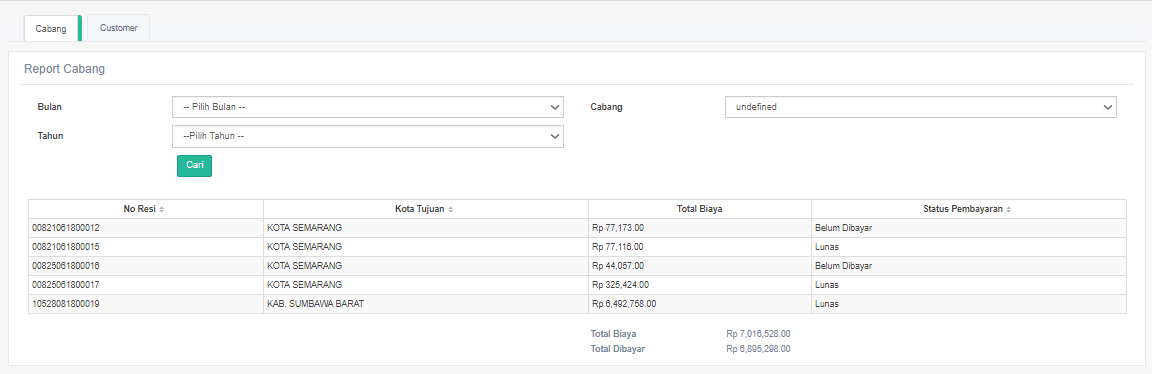
1. Untuk update posisi barang, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | Nomor Resi |
| 2 | Tujuan\_Kabupaten | int |  | 11 | Kota Tujuan |
| 3 | tanggal | Date |  |  | Tanggal cetak |
| 4 | tanggaltibakotatujuan | Datetime |  |  | Tanggal tiba di kota tujuan |
| 5 | tanggalBarangDiTerima | varchar |  | 120 | Tanggal barang diterima |
| 6 | upload | varchar |  | 191 | Upload bukti pembayaran |

1. Tombol “Simpan” diklik, maka akan memanggil fungsi “vm.submitModalStatus()” dan memeperbarui data tracking
2. Tombol close diklik, maka akan menutup form update posisi barang

## Report Cabang

* Layout



* Penjelasan progress
  1. Alamat url : /Report
  2. Sesuai dengan routing berikut :

Route::get('/Report', 'MasterData\ReportController@page');

Halaman tersebut meload fungsi “page” di file “ReportController.php” yang berada di folder “app/Http/Controllers/MasterData/”

* 1. Pada fungsi “page” tersebut, memerintahkan untuk load file “reporting.blade.php” yang berada di “resources/views/pages/”
  2. Pada halaman tersebut terdapat pilihan tab antara lain :
* Cabang

include file “cabang.blade.php” yang berada di folder resources/views/ reporting/”

* Customer

include file “custemer.blade.php” yang berada di folder resources/views/reporting/”

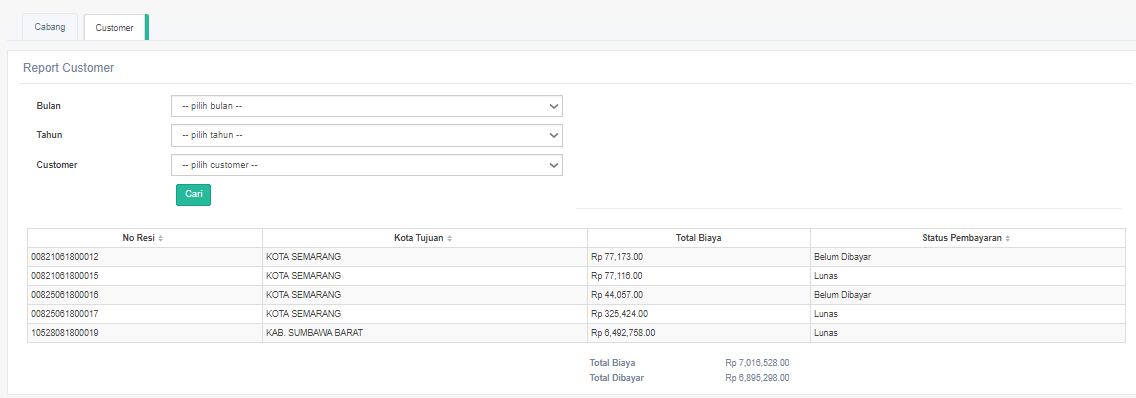
* 1. Ketika tab Cabang di klik, maka akan meload halaman “cabang.blade.php” yang menampilkan halaman report cabang
  2. Untuk menampilkan report cabang sesuai dengan kebutuhan, user dapat mengisi kolom dibawah sebagai filter :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Bulan transaksi |  |  |  | Berupa combo box nama bulan |
| 2 | Tahun transaksi |  |  |  | Berupa combo box tahun |
| 3 | Kode cabang |  |  |  | Berupa combo box cabang |

* 1. Klik tombol “Cari” maka akan memangil fungsi “vm.onSearch()” dan menampilkan daftar transaksi bulanan dengan elemen data:
* Nomor resi
* Kota tujuan
* Total biaya setiap nomor resi
* Status pembayaran (Lunas/Belum Dibayar)
* Total biaya seluruh resi
* Total nilai tagihan yang telah lunas

## Report Customer

* Layout



* Penjelasan proses

1. Sesuai penjelasan proses dari halaman “report cabang”, tab “report customer” include file “custemer.blade.php” yang berada di folder resources/views/reporting/”
2. Untuk menampilkan report customer sesuai dengan kebutuhan, user dapat mengisi kolom dibawah sebagai filter :

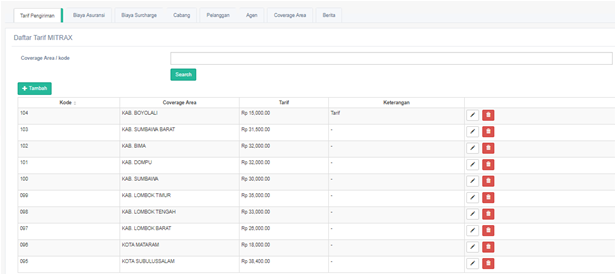
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Bulan transaksi |  |  |  | Berupa combo box nama bulan |
| 2 | Tahun transaksi |  |  |  | Berupa combo box tahun |
| 3 | Kode |  |  |  | Berupa combo Nama customer |

1. Klik tombol “Cari” maka akan memangil fungsi “vm.onSearch()” dan menampilkan daftar transaksi customer dengan elemen data:

* Nomor resi
* Kota tujuan
* Total biaya setiap nomor resi
* Status pembayaran( Lunas/Belum Dibayar )
* Total biaya seluruh resi
* Total nilai tagihan yang telah lunas

## Daftar Tarif Pengiriman

* Layout



* Penjelasan Proses

1. url : /masterData
2. Sesuai dengan routing berikut :

Route::get('/masterData','MasterData\MasterDataController@page');

Halaman tersebut meload fungsi “page” di file “MasterDataController.php” yang berada di folder “app/Http/Controllers/MasterData/”

1. Pada fungsi “page” tersebut, memerintahkan untuk load file “masterData.blade.php” yang berada di “resources/views/pages/”
2. Pada halaman tersebut terdapat pilihan tab antara lain :

* Tarif Pengiriman

include file “masterDataTarif.blade.php” yang berada di folder resources/views/ panels/”

* Biaya Asuransi

include file “masterBiayaAsuransi.blade.php” yang berada di folder resources/views/ panels/”

* Cabang

include file “masterDataCabang.blade.php” yang berada di folder resources/views/ panels/”

* Pelanggan

include file “masterDataCustomer.blade.php” yang berada di folder resources/views/ panels/”

* Agen

include file “masterDataAgen.blade.php” yang berada di folder resources/views/ panels/”

* Coverage Area

include file “masterDataCoverageArea.blade.php” yang berada di folder resources/views/ panels/”

* Berita

include file “masterDataBerita.blade.php” yang berada di folder resources/views/ panels/”

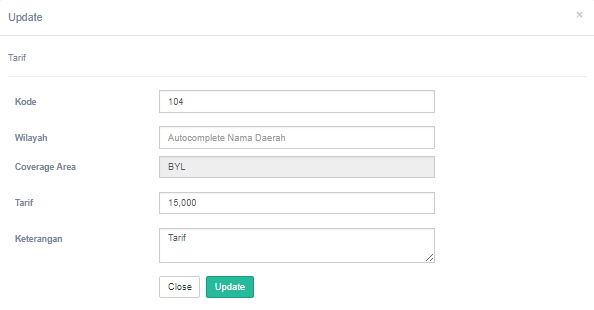
1. Ketika tab “Tarif Pengiriman” di klik, maka akan meload halaman “masterDataTarif.blade.php” yang menampilkan halaman daftar tarif MitraX
2. Untuk menampilkan daftar tarif yang dicari, user dapat mengisi kolom dibawah sebagai filter :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | int |  | 10 | Kode Tarif |
| 2 | name (tbl rgn\_city)  name(tbl rgn\_district) | varchar |  | 255 | Coverage Area |

1. Tombol “Search” diklik maka akan memanggil fungsi “vm.onSearch()” dan menampilkan daftar tarif yang dicari sesuai kata kunci coverage\_area\_id yang diinput

## Form Tarif Pengiriman

* Layout



* Penjelasan proses

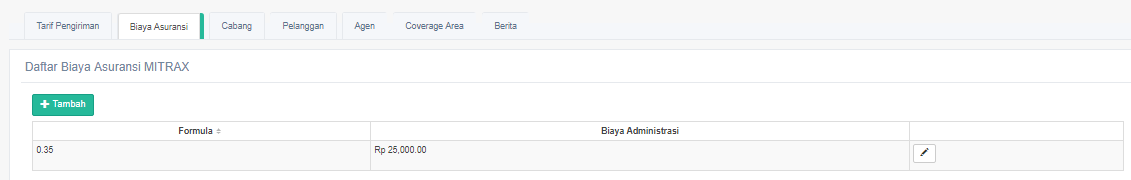
1. Saat user klik tombol “Tambah” pada “Daftar Tarif Pengiriman”, maka akan memanggil fungsi “vm.add()” dan menampilkan form tarif pengiriman
2. Untuk menambah tarif pengiriman baru atau melakukan update, user harus mengisikan kolom sebagai berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode tarif, otomatis digenerate sistem |
| 2 | coverage\_area\_id | int |  | 11 | Berupa autocomplete nama wilayah |
| 3 | Tarif | double |  | 11,2 | Tarif pengiriman |
| 4 | keterangan | text |  |  | Keterangan |

1. Klik tombol “Simpan” maka akan memanggil fungsi “vm.submitModal()”
2. Klik tombol “Update” maka akan memanggil fungsi “vm.submitModal()””
3. Klik tombol “Closed” maka akan menutup form tarif pengiriman

## Biaya Asuransi

* Layout



* Penjelasan Proses

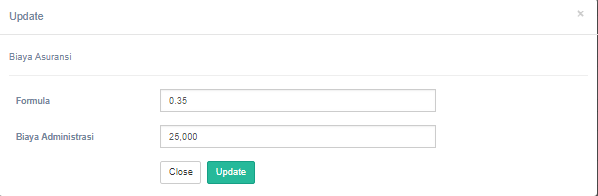
1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Biaya Asuransi” include file “masterBiayaAsuransi.blade.php” yang berada di folder “resources/views/ panels/”
2. Pada Menu Biaya Asuransi, akan ditampilkan daftar biaya asuransi dengan elemen antara lain :

* Formula
* Biaya Administrasi

1. Klik tombol “Tambah” maka akan memanggil fungsi “vm.add()” dan menampilkan “form biaya asuransi”
2. Klik tombol dengan icon “pencil” maka akan memanggil fungsi “vm.edit(row)” dan membuka “form biaya asuransi”

## Form Biaya Asuransi

* Layout



* Penjelasan Proses

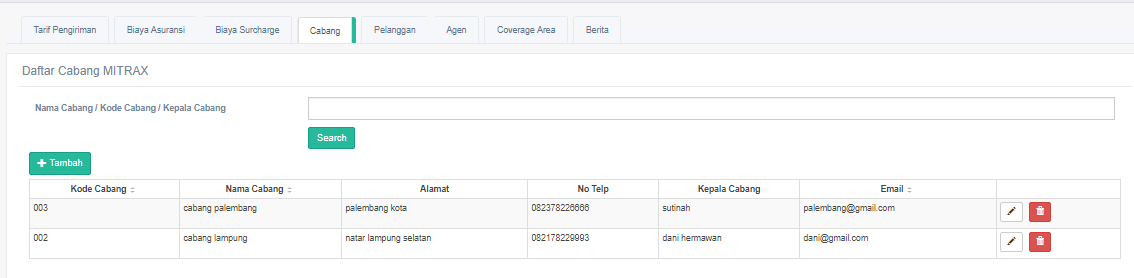
1. Untuk menambah atau memperbarui biaya asuransi, user harus mengisikan kolom sebagai berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | formula | varchar |  | 191 | Persentase untuk perhitungan biaya asuransi |
| 2 | biaya\_administrasi | int |  | 11 | Biaya Administrasi |

1. Klik tombol “Simpan” maka akan memanggil fungsi “vm.submitModal()”
2. Klik tombol “Update” maka akan memanggil fungsi “vm.submitModal()”
3. Klik tombol “Closed” maka akan menutup form

## Daftar Cabang

* Layout



* Penjelasan Proses

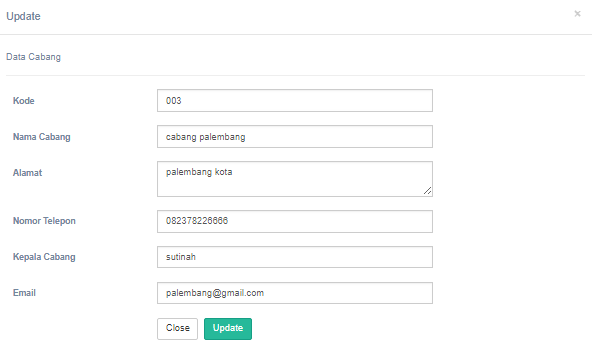
1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Daftar Cabang” include file “masterDataCabang.blade.php” yang berada di folder “resources/views/panels/”
2. Untuk menampilkan daftar Cabang yang dicari, user dapat mengisi kolom kode cabang atau nama cabang atau kepala cabang sebagai filter :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Cabang |
| 2 | nama | varchar |  | 191 | Nama Cabang |
| 3 | kepala\_cabang | varchar |  | 191 | Kepala Cabang |

1. Tombol “Search” diklik maka akan memanggil fungsi “vm.onSearch()” dan menampilkan daftar cabang yang dicari sesuai kata kunci yang diinput
2. Tombol “Tambah” diklik, maka akan memanggil fungsi “vm.add()” dan menampilkan form cabang
3. Pada daftar cabang, apabila tombol dengan icon “pencil” diklik, maka akan memanggil fungsi “vm.edit(row)” dan menampilkan form cabang
4. Pada daftar cabang, apabila tombol dengan icon “trash” diklik, maka akan memanggil fungsi “vm.del(row)” menghapus data cabang yang dipilih

## Form Cabang

* Layout



* Penjelasan Proses

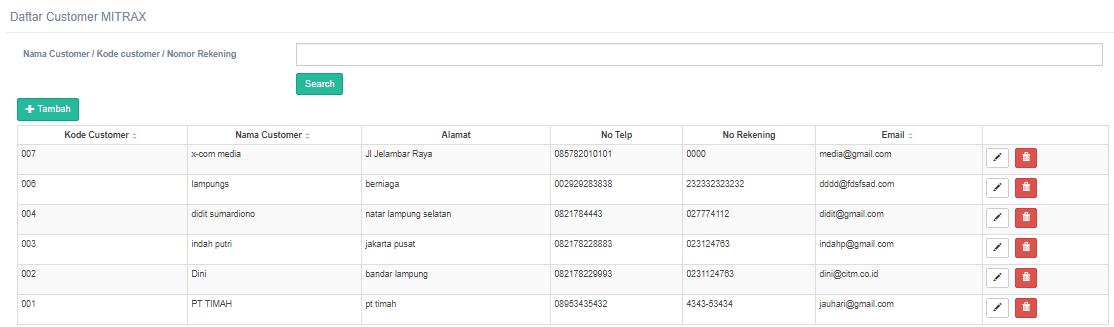
1. Untuk menambah atau memperbarui cabang, user harus mengisikan kolom sebagai berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Cabang. Tergenerate otomatis oleh sistem. |
| 2 | nama | varchar |  | 191 | Nama Cabang |
| 3 | alamat | text |  |  | Alamat cabang |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor Telpon Cabang |
| 5 | kepala\_cabang | varchar |  | 191 | Kepala Cabang |
| 6 | email | varchar |  | 191 | Alamat Email |

1. Klik tombol “Simpan” maka akan memanggil fungsi “vm.submitModal()”
2. Klik tombol “Update” maka akan memanggil fungsi “vm.submitModal()”
3. Klik tombol “Closed” maka akan menutup form cabang

## Daftar Pelanggan

* Layout



* Penjelasan Proses

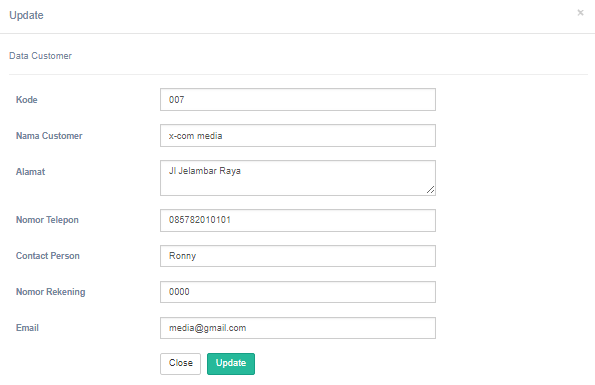
1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Daftar Pelanggan” include file “masterDataCustomer.blade.php” yang berada di folder “resources/views/panels/”
2. Untuk menampilkan daftar Pelanggan yang dicari, user dapat mengisi kolom Nama Customer / Kode customer / Nomor Rekeningsebagai filter :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Pelanggan |
| 2 | nama | varchar |  | 191 | Nama Pelanggan |
| 3 | nomor\_rekening | varchar |  | 191 | Nomor Rekening |

1. Tombol “Search” diklik maka akan memanggil fungsi “vm.onSearch()” dan menampilkan daftar pelanggan yang dicari sesuai kata kunci yang diinput
2. Apabila tombol dengan icon “pencil” diklik, maka akan memanggil fungsi “vm.edit(row)” dan menampilkan form cabang
3. Pada daftar cabang, apabila tombol dengan icon “trash” diklik, maka akan memanggil fungsi “vm.del(row)” menghapus data cabang yang dipilih

## Form Pelanggan

* Layout



* Penjelasan proses

1. Untuk melakukan update atau tambah pelanggan, user harus mengisi kolom:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | *char* |  | 3 | Kode Pelanggan (otomatis tergenerate) |
| 2 | nama | *varchar* |  | 191 | Nama pelanggan |
| 3 | alamat | *text* |  |  | Alamat pelanggan |
| 4 | nomor\_telepon | *varchar* |  | 191 | Nomor telpon pelanggan |
| 5 | contact\_person | *varchar* |  | 191 | Nama kontak person pelanggan |
| 6 | nomor\_rekening | *varchar* |  | 191 | Nomor rekening pelanggan |
| 7 | email | *varchar* |  | 191 | Email pelanggan |

1. Untuk form update pelanggan:

Tombol update diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data pelanggan

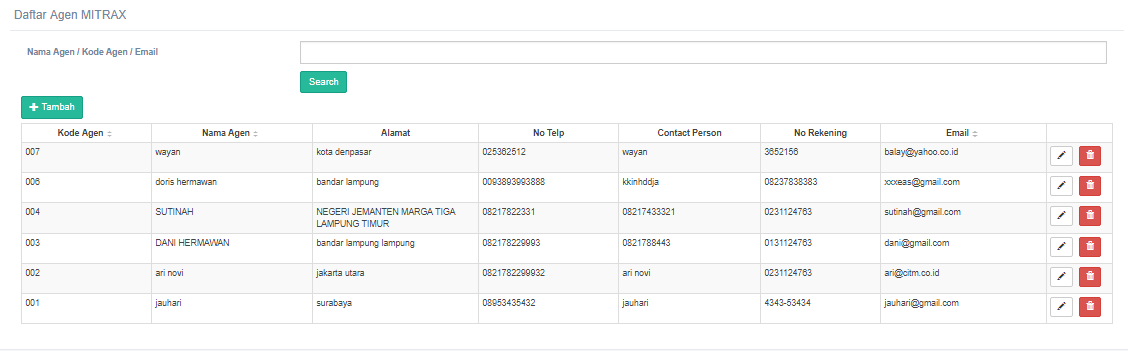
1. Untuk form tambah pelanggan:

Tombol simpan diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data pelanggan

1. Tombol close diklik, maka akan menutup form pelanggan

## Daftar Agen

* Layout



* Penjelasan Proses

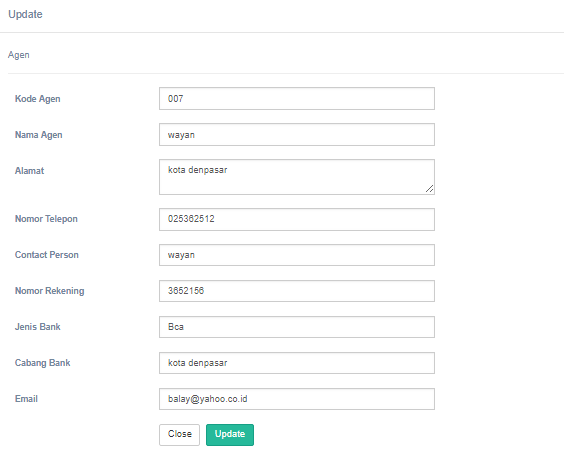
1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Daftar Agen” include file “masterDataAgen.blade.php” yang berada di folder “resources/views/panels/”
2. Untuk melakukan pencarian agen, user harus mengisi kolom nama agen atau kode agen atau email:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_agen | char |  | 3 | Kode Agen |
| 2 | nama\_agen | varchar |  | 191 | Nama Agen |
| 3 | email | *varchar* |  | 191 | Email agen |

1. Tombol Search diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan daftar agen
2. Tombol “Tambah” diklik, maka akan memanggil fungsi “vm.add()” dan menampilkan form tambah agen
3. Tombol edit dengan icon “pencil” diklik, maka akan memanggil “vm.edit(row)” fungsi menampilkan form update agen
4. Tombol hapus dengan icon “trash” diklik, maka akan memanggil fungsi “vm.del(row)” dan menghapus daftar agen yang dipilih

## Form Agen

* Layout



* Penjelasan Proses

1. Untuk melakukan update atau tambah agen, user harus mengisi kolom:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_agen | char |  | 3 | Kode Agen |
| 2 | nama\_agen | varchar |  | 191 | Nama Agen |
| 3 | alamat | *text* |  |  | Alamat agen |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor telpon agen |
| 5 | contact\_person | *varchar* |  | 191 | Nama kontak person |
| 6 | nomor\_rekening | *varchar* |  | 191 | Nomor rekening agen |
| 7 | Jenis\_Bank | *varchar* |  | 255 | Jenis bank |
| 8 | Cabang\_Bank | *varchar* |  | 255 | Cabang bank |
| 9 | email | *varchar* |  | 191 | Email agen |

1. Untuk form update agen :

Tombol update diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data agen

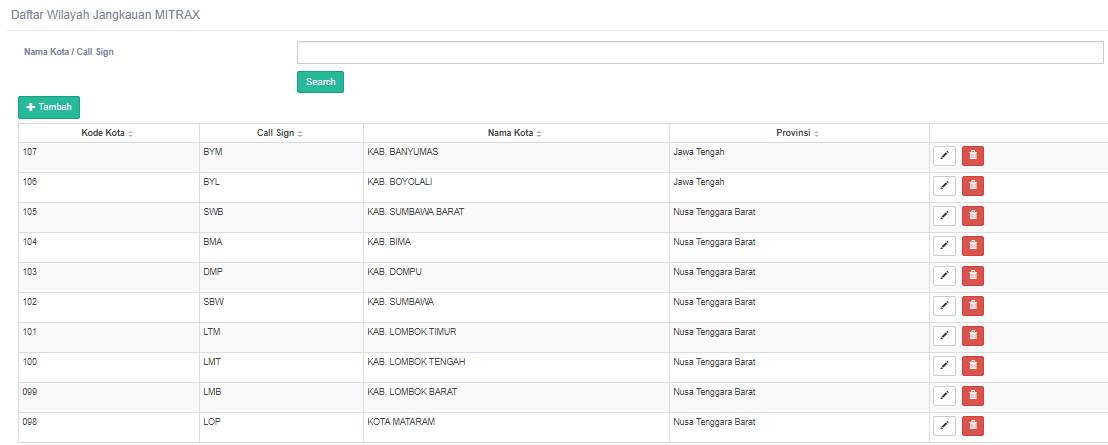
1. Untuk form tambah agen:

Tombol simpan diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data agen

1. Tombol close diklik, maka akan menutup form agen

## Daftar Wilayah Jangkauan

* Layout



* Penjelasan proses

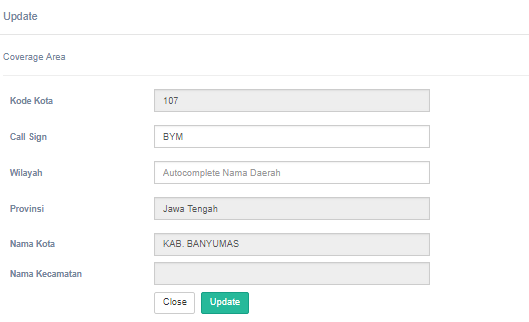
1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Daftar Wilayah Jangkauan” include file “masterDataCoverageArea.blade.php” yang berada di folder “resources/views/panels/”
2. Untuk melakukan pencarian wilayah jangkauan, user harus mengisikan kolom

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_kota | char |  | 3 | Kode Kota |

1. Tombol Search diklik, maka akan memanggil fungsi “vm.onSearch()” dan menampilkan daftar agen
2. Tombol “Tambah” diklik, maka akan memanggil fungsi “vm.add()” dan menampilkan form agen tambah agen
3. Tombol edit dengan icon “pencil” diklik, maka akan memanggil fungsi “vm. editca(row)” dan menampilkan form edit coverage area
4. Tombol hapus dengan icon “trash” diklik, maka akan memanggil fungsi “vm.del(row)” dan menghapus daftar agen yang dipilih

## Form Wilayah Jangkauan

* Layout



* Penjelasan proses

1. Untuk melakukan update atau tambah coverage area, user harus mengisi kolom:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_kota | *char* |  | 3 | Kode Kota |
| 2 | call\_sign | *char* |  | 3 | Call sign kota |
| 3 | nama\_kota | *varchar* |  | 191 | Wilayah berupa autocomplete nama daerah |

1. Untuk form update coverage area :

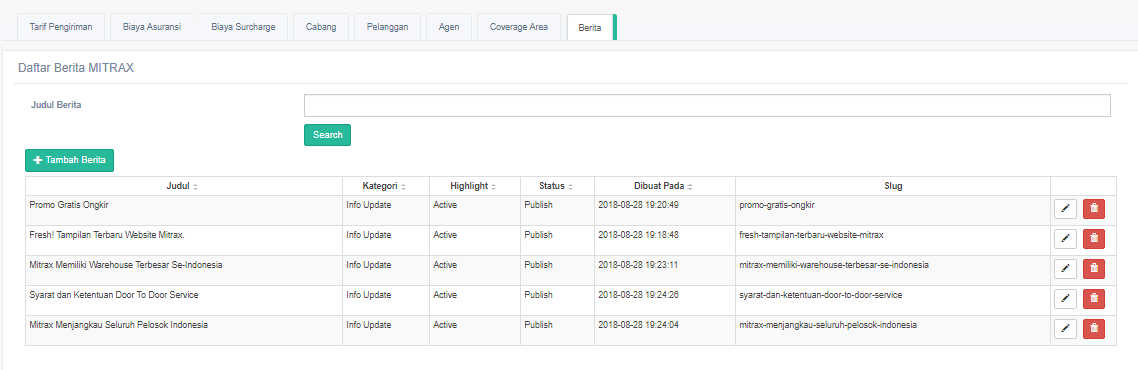
Tombol update diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data

1. Untuk form tambah coverage area :

Tombol simpan diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data

## Daftar Berita

* Layout

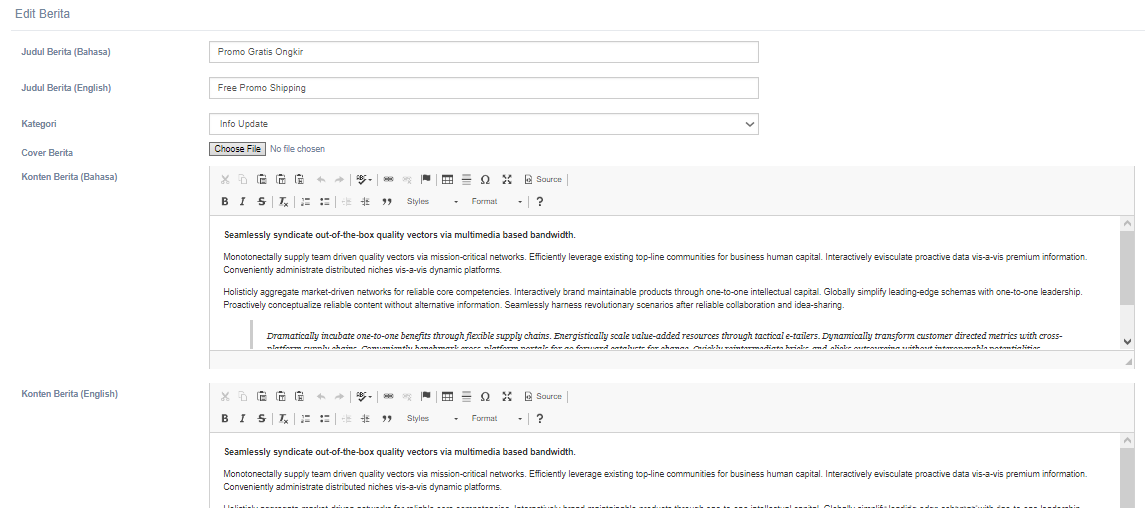


* Penjelasan proses

1. Sesuai penjelasan proses dari halaman “Daftar Tarif Pengiriman”, tab “Daftar Berita” include file “masterDataBerita.blade.php” yang berada di folder “resources/views/panels/”
2. Untuk melakukan pencarian, user harus mengisi kolom Judul berita
3. Tombol “Tambah Berita” diklik, maka akan menampilkan form “Tambah berita”
4. Tombol dengan icon “pencil” maka akan memanggil fungsi ” editberita({id\_news: row.id\_news})” dan menampilkan form update berita
5. Tombol dengan icon “trash” diklik, maka akan memanggil fungsi “vm.delete(row)” dan menghapus data berita yang dipilih

## Form Berita

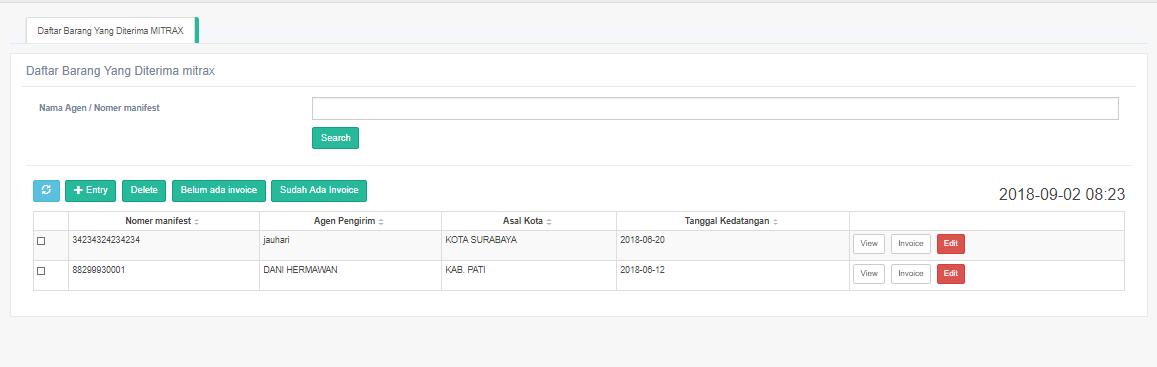
* Layout



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | title\_ind |  |  |  | Judul berita berbahasa Indonesia |
| 2 | title\_eng |  |  |  | Judul berita berbahasa Inggris |
| 3 | kat\_berita |  |  |  | Keterangan berita |
| 4 | content\_ind |  |  |  | Isi berita berbahasa Indonesia |
| 5 | content\_eng |  |  |  | Isi berita berbahasa inggris |
| 6 | highlight |  |  |  | Draft / publish |
| 7 | status |  |  |  | Active/inactive |

## Daftar Barang yang Diterima MITRAX

* Layout



* Penjelasan proses
  1. url : /BarangKiriman
  2. Sesuai dengan routing berikut :

Route::get('/BarangKiriman','MasterData\BarangKirimanController@page');

Halaman tersebut meload fungsi “page” di file “BarangKirimanController.php” yang berada di folder “app/Http/Controllers/MasterData/”

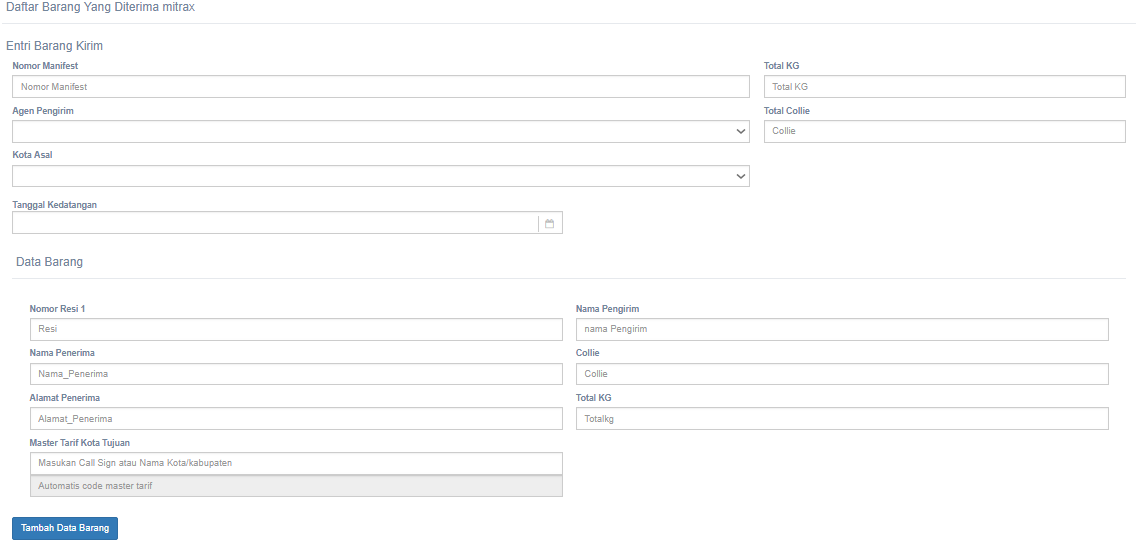
* 1. Pada fungsi “page” tersebut, memerintahkan untuk load file “barangkirim.blade.php” yang berada di “resources/views/pages/”, file tersebut meload tab “Daftar Barang Yang Diterima MITRAX”
  2. Sedangkan form daftar barang berasal dari file php : “resources/views/barangkirim/daftarBarang.blade.php”
  3. Untuk menampilkan daftar barang yang diterima Mitrax, user dapat melakukan filter dengan mengisi kolom berupa kata kunci

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nama\_agen | varchar |  | 191 | Nama agen |
| 2 | Nomanifest\_barangkirim | varchar |  | 191 | Nomor manifest |

* 1. Klik Tombol “Search” untuk menampilkan daftar barang yang diterima MitraX
  2. Klik tombol dengan icon “refresh” maka akan memperbarui daftar barang yang diterima MitraX
  3. Centang daftar barang yang hendak dihapus lalu klik tombol “delete”, maka daftar barang yang dicentang akan dihapus
  4. Klik tombol “Entry” , maka akan menampilkan form daftar barang yang diterima MitraX
  5. Klik tombol “Belum ada Invoice” maka akan menampilkan form daftar barang yang belum ada invoice
  6. Klik tombol “Sudah ada Invoice” maka akan menampilkan form daftar barang yang sudah ada invoice

## Form Entry

* Layout



* Penjelasan Proses

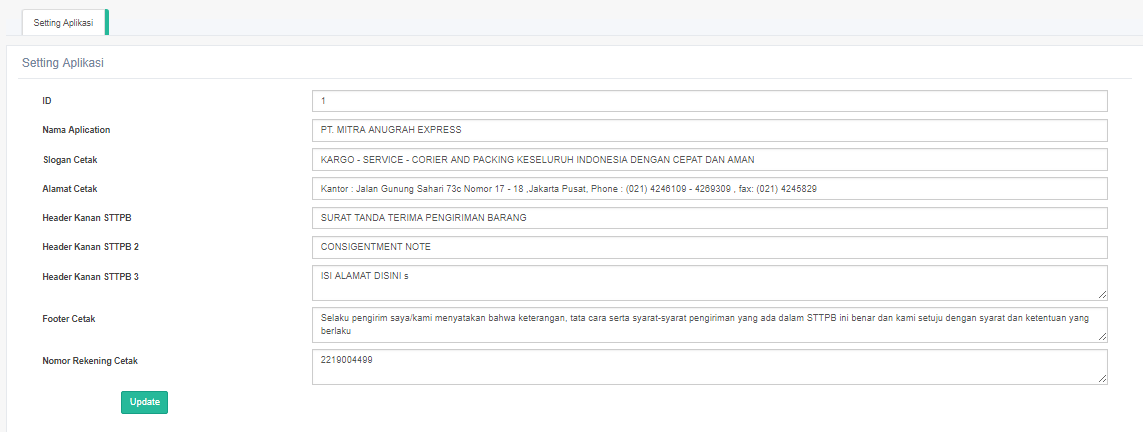
1. File php yang diload : resources/views/barangkirim/daftarBarang.blade.php
2. Untuk mengisi form entry barang yang diterima, kolom yang diisi antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest | *varchar* |  | 191 | No Manifest |
| 2 | Kode\_Agent | *int* |  | 11 | Agen pengirim |
| 3 | Asal\_Kabupaten | *int* |  | 11 | Kota Asal |
| 4 | Tanggal | *varchar* |  | 191 | Tanggal Kedatangan |
| 5 | Berat\_Barang | *varchar* |  | 191 | Total Kg |
| 6 | no\_resi | *char* |  | 15 | No Resi |
| 7 | Nama\_Penerima | *varchar* |  | 191 | Nama Penerima |
| 8 | Alamat\_Tujuan | *varchar* |  | 191 | Alamat penerima |
| 9 | Nama\_Pengirim | *varchar* |  | 191 | Nama pengirim |

1. Tombol “Tambah Data barang” diklik, maka akan menambah data barang

## Pengaturan Aplikasi

* Layout



* Penjelasan Proses

1. url : /Pengaturan
2. Sesuai dengan routing berikut :

Route::get('/Pengaturan','MasterData\PengaturanController@page');

Halaman tersebut meload fungsi “page” di file “PengaturanController.php” yang berada di folder “app/Http/Controllers/MasterData/”

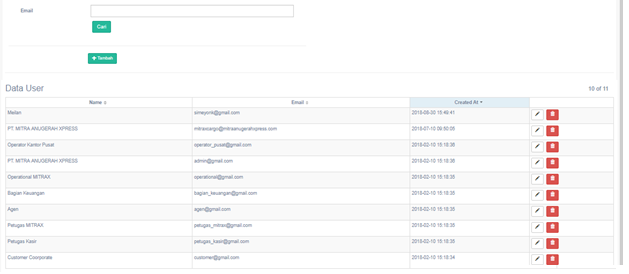
1. Pada fungsi “page” tersebut, memerintahkan untuk load file “pengaturan.blade.php” yang berada di “resources/views/pages/”
2. Pada file pengaturan.blade.php menginclude file app.blade.php di folder “resources/views/setting”
3. Untuk memperbarui setting aplikasi, kolom yang diisi antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nameApp | *varchar* |  | 191 | Nama aplikasi |
| 2 | sloganApp | *varchar* |  | 191 | Slogan yang tampil pada aplikasi |
| 3 | alamatApp | *varchar* |  | 191 | Alamat yang tampil pada aplikasi |
| 4 | headSTTPB | *varchar* |  | 191 | Informasi header Kanan STTPB |
| 5 | headSTTPB2 | *varchar* |  | 191 | Informasi header Kanan STTPB 2 |
| 6 | headSTTPB3 | *varchar* |  | 191 | Informasi header Kanan STTPB 3 |
| 7 | logo | *varchar* |  | 191 | Logo aplikasi |
| 8 | headerpengirim | *text* |  |  | Footer cetak (informasi yang tercetak pada footer) |
| 9 | nomor\_rekening | *char* |  | 60 | Nomor rekening |

1. Tombol “Update” diklik, maka akan memanggil fungsi “vm.formDatasearch()” dan memperbarui setting aplikasi

## Manage User

* Layout



* Penjelasan Proses
  1. url : /manageUsers
  2. Sesuai dengan routing berikut :

Route::get('/manageUsers','UserController@manageUsers')>name('manageUsers');

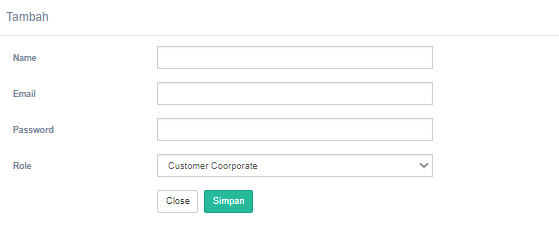
* 1. Halaman tersebut meload fungsi “manageUsers'” di file “UserController.php” yang berada di folder “app/Http/Controllers /”
  2. Pada fungsi “manageUsers'” tersebut, memerintahkan untuk load file “manageUsers.blade.php” yang berada di “resources/views/pages/”
  3. Untuk melakukan pencarian data user, user harus mengisi kolom berikut:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | email | *varchar* |  | 191 |  |

1. Tombol “Cari” diklik, maka akan memanggil fungsi “vm.Cari()” dan menampilkan Data user yang dicari
2. Tombol “Tambah” diklik, maka akan memanggil fungsi “vm.add()” dan menampilkan form tambah user
3. Tombol edit dengan icon pencil diklik, maka akan memanggil fungsi “vm.edit(row)” dan menampilkan form edit user
4. Tombol hapus dengan icon trash diklik, maka akan memanggil fungsi “vm.del(row)” dan menghapus data yang dipilih

## Tambah User

* Layout



* Penjelasan Proses

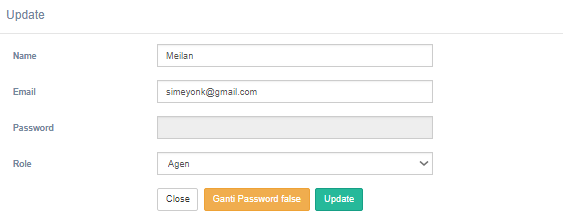
1. Untuk menambah user baru, user harus mengisi kolom berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | name | *varchar* |  | 191 | Nama user |
| 2 | email | *varchar* |  | 191 | Email user |
| 3 | Password | *varchar* |  | 191 | Password untuk user login |
| 4 | role\_id | *int* |  | 10 | Hak akses user terhadap aplikasi |

1. Tombol “Simpan” diklik, maka akan memanggil fungsi “vm.submitModal()” dan menyimpan data user ke database
2. Tombol “Close” diklik, maka akan menutup popup form Tambah user

## Update User

* Update user



* Penjelasan Proses

1. Untuk mengubah data user, kolom yang bisa diubah antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | name | name |  | 191 | Nama user |
| 2 | email | *varchar* |  | 191 | Email user |
| 3 | Password | *varchar* |  | 191 | Password untuk user login |
| 4 | role\_id | *int* |  | 10 | Hak akses user terhadap aplikasi |

1. Tombol “Ganti Password” diklik, maka akan meng-enable (membuka) inputan password
2. Tombol “Update” diklik, maka akan memanggil fungsi “vm.submitModal()” dan memperbarui data user
3. Tombol “Close” diklik, maka akan menutup popup form Update user

# BACK END SPESIFICATION

## Login

* 1. Function Name : validateLogin
  2. Alur Proses :

1. Function menerima request login dari form login (front end)
2. Function melakukan validasi dengan cara cek user dan password ke table users
3. Function memberikan response message sesuai hasil pengecekan apakah Sukses atau gagal.
   1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Email | *varchar* |  | 191 | Email user |
| 2 | Password | *varchar* |  | 191 | Password untuk user login |
| 3 | CaptchaCode |  |  |  | Kode Captcha |

* 1. **Forgot Password**
     1. Service Name : SendsPasswordResetEmails
     2. Alur Proses :

Function menerima request “Send Password Reset Link” dari form “forgot password” (front end)

Function memvalidasi email, apabila email tersebut milik salah satu user, sistem mengirim password baru ke email yang diinput

Apabila email tidak ditemukan, maka akan mengirimkan notifikasi error

Function memberikan response message sesuai hasil pengiriman password apakah Sukses atau gagal.

* + 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Email | Varchar |  | 191 | Email |

* + 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Status |  |  |  | 1= success  0=failed dengan informasi error |

* 1. **Create New Account**

1. Service Name : Register
2. Alur Proses :

fungsi menerima request register dari form “Create New Account”

fungsi melakukan validasi dengan cara cek nama, email dan password yang diisi apakah sudah sesuai apa belum. Kesesuaian dinilai dari : nama harus diisi, email haru diisi dan email belum pernah terdaftar sebelumnya, password harus diisi dengan minimal 6 karakter dan konfirmasi password harus sama dengan password

Jika sudah sesuai, maka akan disimpan dalam table “user”.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Name | Varchar |  | 191 | Nama user |
| 2 | email | Varchar |  | 191 | Email User |
| 3 | password | Varchar |  | 191 | Password untuk login |

## Order Jemput

* + 1. Service Name :penjemputan/order
    2. Alur Proses :

API menerima request pencarian daftar order jemput pada table “orders”

API memberikan response message berupa daftar order jemput sesuai request

* + 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nomor\_order | char |  | 12 | Nomor Order |
| 2 | waktu\_jemput\_request | timestamp |  |  | Tanggal Jemput |

* + 1. Response Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| nomor\_order | *char* |  | 12 | Nomor Order |
| nama\_customer | *Varchar* |  | 191 | Nama customer / agent |
| waktu\_jemput\_request | *timestamp* |  |  | Waktu jemput request |
| waktu\_jemput\_real | *timestamp* |  |  | Waktu jemput approve |
| jenis\_kendaraan | *Varchar* |  | 191 | Kendaraan jemput |
| penanggung\_jawab | *Varchar* |  | 191 | driver |
| nomor\_polisi | *varchar* |  | 191 | Nomor Polisi |

* 1. **Form Order Penjemputan**
* Service Name : customersearch

1. Alur Proses :
2. menerima request buka order penjemputan
3. Pada nama\_customer memanggil API “customersearch”
4. API tersebut diarahkan ke :

Route::get('customersearch/{noresi}','api\orderController@customersearch');

Dimana API tersebut memanggil fungsi “customersearch” di file “orderController.php” pada folder app/Http/Controllers/api/orderController.php

1. Fungsi tersebut melakukan pencarian customer berdasarkan kode atau nama pelanggan yang diinput
2. Request Message :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| kode | char |  | 3 | Kode customer |
| nama\_customer | Varchar |  | 191 | Nama customer / agent (autocomplete) |

1. Response Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| kode | char |  | 3 | Kode customer |
| nama\_customer | Varchar |  | 191 | Nama customer / agent |

* Function Name : JenisBarang
  1. Alur Proses :

Fungsi JenisBarang meload file JenisBarang.php yang berada di folder “app/JenisBarang.php”

Sumber Data dari fungsi JenisBarang adalah table “jenis\_barangs”

* Function Name : JenisKendaraan
  + 1. Alur Proses :

Fungsi JenisKendaraan meload file JenisKendaraan.php yang berada di folder “app/ JenisKendaraan.php”

Sumber Data dari fungsi JenisKendaraan adalah table “jenis\_kendaraans”

* 1. **Penjadwalan Penjemputan**

1. Service Name : api/penjemputan/order
2. Alur Proses :

## Api diatas bisa ditrack dengan melihat file routes/api.php

Route::resource('penjemputan/order', 'Penjemputan\OrderController');

Artinya diarahkan ke “app/Http/Controllers/Penjemputan/OrderController”

## API menerima request pencarian jadwal dengan kata kunci nomor\_order atau tanggal\_jemput

## API melakukan pencarian jadwal ke table “Order” yang berelasi dengan table “driver” , “jenis\_kendaraans”

1. Request Message :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| nomor\_order | *char* |  | 12 | Nomor Order |
| waktu\_jemput\_request | *timestamp* |  |  | Tanggal Jemput |

1. Response Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| nomor\_order | *char* |  | 12 | Nomor Order |
| nama\_customer | *Varchar* |  | 191 | Nama customer / agent |
| waktu\_jemput\_request | *timestamp* |  |  | Waktu jemput request |
| waktu\_jemput\_real | *timestamp* |  |  | Waktu jemput approve |
| jenis\_kendaraan | *Varchar* |  | 191 | Kendaraan jemput |
| alamat\_penjemputan | *text* |  |  | alamat |
| penanggung\_jawab | *Varchar* |  | 191 | driver |
| nomor\_polisi | *varchar* |  | 191 | Nomor Polisi |

* 1. **Form Pilih Driver**

1. Service Name :
2. Alur Proses :
3. Pada daftar jadwal jemput, terdapat tombol untuk atur atau edit, tombol tesebut memanggil fungsi “vm.show(row)” saat diklik
4. fungsi “vm.show(row)” terdapat di file penjemputanController.js pada folder “public/js/angular\_js/controllers/penjemputanControllers.js”
5. “row” pada fungsi tersebut adalah “order\_id”
6. Fungsi tersebut menampilkan form pilih driver, dan list driver yang bisa dipilih.
7. Form pilih driver terdiri dari :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| penanggung\_jawab | *varchar* |  | 191 | Driver |
| nomor\_polisi | *varchar* |  | 191 | Nomor Polisi |
| waktu\_jemput\_real | *timestamp* |  |  | Waktu jemput real |
| waktu\_kembali\_real | *timestamp* |  |  | Waktu kembali real |

1. Sedangkan Data yang ditampilkan di list antara lain :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| penanggung\_jawab | *varchar* |  | 191 | Driver |
| jenis\_kendaraan | *varchar* |  | 191 | Jenis Kendaraan |
| nomor\_telepon | *varchar* |  | 191 | Nomor Telepon |
| nomor\_polisi | *varchar* |  | 191 | Nomor Polisi |
| status | *Varchar* |  | 191 | Jika 1 = success  0=null |

1. Saat tombol Pilih diklik, maka akan memanggil fungsi “vm.pilihDriver(row)” yang berada di file penjemputanController.js pada folder “public/js/angular\_js/controllers/penjemputanControllers.js”
2. Fungsi tersebut akan memanggil driver\_id, nama\_driver, dan nomor\_polisi lalu ditampilkan dalam “form pilih driver”
3. User melengkapi waktu\_jemput dan waktu\_kembali
   1. **Daftar Driver dan Kendaraan**
      * 1. File php yang diload : penjemputanDaftarDriver.blade.php yang berada di folder : resources/views/panels/
        2. Alur Proses :

Table yang berkaitan : drivers

Halaman tersebut menampilkan daftar driver dan kendaraan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Status | Length | Description |
| jenis\_kendaraan\_id | *int* |  | 11 | Jenis Kendaraan |
| nomor\_polisi | *varchar* |  | 191 | Nomor Polisi |
| penanggung\_jawab | *varchar* |  | 191 | Nama Driver |
| nomor\_telepon | *varchar* |  | 191 | Nomor Telepon |
| waktu\_jemput\_real | *timestamp* |  |  | Waktu jemput real |
| waktu\_kembali\_real | *timestamp* |  |  | Waktu Kembali |
| status |  |  |  | status |

Fungsi “vm.add()”, vm disini adalah driverController yang terletak di public/js/angular\_js/controllers/penjemputanController.js dan bagian “myApp.controller("driverController", controller)” fungsi vm.add menampilkan modal/popup tambah kendaraan

Untuk update menggunakan primary key id, data yang bisa diupdate

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | jenis\_kendaraan\_id | int |  | 11 | Jenis Kendaraan |
| 2 | nomor\_polisi | Varchar |  | 191 | Nomor Polisi |
| 3 | penanggung\_jawab | *varchar* |  | 191 | Nama Driver |
| 4 | nomor\_telepon | *varchar* |  | 191 | Nomor Telepon |

* 1. **Input Shipment**
     1. Url yang dipanggil : [/page/shipment](http://192.168.5.233/home#/page/shipment)
     2. Untuk menentukan file php yang dipanggil, Kita bisa melihat dari routes.js yang berada di folder public/js/angular\_js/routes.js
     3. Url page/shipment meload file app/Http/Controllers/Shipment/ShipmentController.php dengan fungsi index
     4. Fungsi tersebut meload halaman “pages/shipment.blade.php”
     5. Pada form input shipment terdapat combo box kota tujuan yang memanggil api tujuanvillages
     6. Alur Proses :

1. API “tujuanvillages” terdapat pada “public/js/angular\_js/controllers/shipmentController.js” API tersebut meload url “api/villages/” yang bisa Kita lihat dari file “routes/api.php”, pada file tersebut diiformasikan untuk api villages meload fungsi villages pada file “api/placeController.php”
2. API “tujuanvillages” memberikan response list Kota
3. Saat tombol “Search” diklik, maka akan memanggi fungsi “ onSearchinputshipment”
4. Fungsi “inputshipment” memanggil halaman “Input\_shipment.blade.php” yang berada di folder “resources/views/Shipment/”
5. Pada halaman ini meload API “fungsishipment” , API tesebut diarahkan ke “app/Http/Controllers/api/OrderContoller.php”
6. API tersebut menerima request pencarian berdasarkan kata kunci No Resi, Tanggal Kirim dan Kota Tujuan ke table “shipment”
7. API memberikan response message sesuai pencarian data “shipment”
8. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 5 | No Resi |
| 2 | Tanggal | Varchar |  | 191 | Tanggal Kirim |
| 3 | Tujuan\_Kabupaten | Int |  | 11 | Kota Tujuan berupa combo box |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | id | *int* |  | 10 | Kode |
| 2 | no\_resi | *char* |  | 15 | No Resi pengiriman |
| 6 | Nama\_Pengirim | *varchar* |  | 191 | Nama Pengirim |
| 9 | Alamat\_Asal | *varchar* |  | 191 | Alamat asal pengiriman |
| 13 | Asal\_Desa | *int* |  | 11 | Kode desa asal pengiriman |
| 15 | Alamat\_Tujuan | *varchar* |  | 191 | Alamat tujuan pengiriman |
| 23 | Metode\_Pembayaran | *varchar* |  | 191 | Metode pembayaran. Berupa combo box dengan pilihan :Cash, Kredit, bayar di tujuan |
| 39 | created\_at | *timestamp* |  |  | Tanggal Kirim |
| 40 | STTPB |  |  |  | . |

* 1. **Form Daftar / Edit Shipment**
* Service Name : vm.submitModal()

1. Alur Proses :

Untuk input shipment, fungsi yang di load adalah “inputShpment” yang terletak di “app/Http/Controllers/api/orderController.php”

1. Fungsi menerima request input
2. Fungsi melakukan validasi pengiriman data ke table “shipment”
3. Pertama fungsi tersebut melakukan validasi terhadap CoverageArea, apabila CoverageArea tidak ditemukan, maka akan memberikan respon status “data kota tujuan tidak terdaftar di master tarif”
4. Pembuatan no resi otomatis yaitu gabungan dari kode kota + tanggal (format dmy) + no urut
5. Request Message Format

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Id | *Int* |  | 10 | Kode |
| 2 | no\_resi | *char* |  | 15 | No Resi pengiriman |
| 3 | Nomor\_Order\_Jemput | *char* |  | 12 | No order penjemputan |
| 4 | Nama\_Penerima | *varchar* |  | 191 | Nama penerima |
| 5 | No\_Telp\_Penerima | *varchar* |  | 191 | No Telpon Penerima |
| 6 | Nama\_Pengirim | *varchar* |  | 191 | Nama Pengirim |
| 7 | No\_Telp\_Pengirim | *varchar* |  | 191 | No Telpon pengirim |
| 8 | ID\_Pelanggan | *Int* |  | 11 | Kode pelanggan |
| 9 | Alamat\_Asal | *varchar* |  | 191 | Alamat asal pengiriman |
| 10 | Asal\_Provinsi | *int* |  | 11 | Kode provinsi asal pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 11 | Asal\_Kabupaten | *Int* |  | 11 | Kode kabupaten asal pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 12 | Asal\_kecamatan | *Int* |  | 11 | Kode kecamatan asal pengiriman. Berupa autocomplete. |
| 13 | Asal\_Desa | *Int* |  | 11 | Kode desa asal pengiriman |
| 14 | Asal\_Kode\_Pos | *Int* |  | 11 | Kode pos asal pengiriman |
| 15 | Alamat\_Tujuan | *varchar* |  | 191 | Alamat tujuan pengiriman |
| 16 | Tujuan\_Provinsi | *Int* |  | 11 | Kode provinsi tujuan pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 17 | Tujuan\_Kabupaten | *int* |  | 11 | Kode kabupaten tujuan pengiriman. Otomatis terisi saat user memilih autocomple kecamatan |
| 18 | Tujuan\_kecamatan | *Int* |  | 11 | Kode kecamatan tujuan pengiriman. Berupa autocomplete |
| 19 | Tujuan\_Desa | *Int* |  | 11 | Kode desa tujuan pengiriman |
| 20 | Tujuan\_Kode\_Pos | *Int* |  | 11 | Kode pos tujuan pengiriman |
| 21 | Kode\_Agent | *Int* |  | 11 | Kode Agen pengirim. Berupa autocomplete agen |
| 22 | Kode\_Transit | *Int* |  | 11 | Kode transit |
| 23 | Metode\_Pembayaran | *varchar* |  | 191 | Metode pembayaran. Berupa combo box dengan pilihan :Cash, Kredit, bayar di tujuan |
| 24 | Jalur\_Angkut | *varchar* |  | 191 | Jalur angkut pengiriman. Berupa combo box dengan pilihan : Darat, Laut, Udara |
| 25 | Jenis\_Barang | *varchar* |  | 191 | Jenis barang yang dikirim. Berupa combo box dengan pilihan : Paket dan Dokumen |
| 26 | Jumlah\_Barang | *varchar* |  | 191 | Jumlah barang yang dikirim berupa combo box dengan pilihan jumlah |
| 27 | Jenis\_kemasan | *varchar* |  | 191 | Jenis kemasan paket yang dikirim |
| 28 | Nama\_Barang | *varchar* |  | 191 | Nama barang yang dikirim |
| 29 | Berat\_Barang | *varchar* |  | 191 | Berat barang yang dikirim |
| 30 | Harga\_Barang | *Int* |  | 11 | Harga barang yang dikirim |
| 31 | Volume | *Int* |  | 11 | Volume barang yang dikirim |
| 32 | Dimensi\_Barang\_P | *Int* |  | 11 | Dimensi barang (panjang) |
| 33 | Dimensi\_Barang\_L | *Int* |  | 11 | Dimensi barang (lebar) |
| 34 | Dimensi\_Barang\_T | *Int* |  | 11 | Dimensi barang (tinggi) |
| 35 | Berat\_Dibayar | *Int* |  | 11 | Berat yang harus dibayar |
| 36 | Layanan\_Tambahan | *varchar* |  | 191 | Layanan tambahan yang dipilih. Berupa check box dengan pilihan :   * Packing Kayu / peti * Paking Khusus * Asuransi * Jemput |
| 37 | Instruksi\_Khusus | *varchar* |  | 191 | Instruksi khusu |
| 38 | Biaya\_Pengiriman | *Int* |  | 11 | Biaya pengiriman barang |
| 39 | Keterangan\_Barang | *Text* |  |  | Keterangan barang |
| 40 | Diskon | *Int* |  |  | Diskon yang diberikan. Berupa combo box dengan beberapa pilihan diskon dalam persentase. |
| 41 | Biaya\_Asuransi | *Int* |  |  | Biaya asuransi yang dikenakan |
| 42 | Biaya\_Packing | *Int* |  |  | Biaya packing |
| 43 | biaya\_surchange | *Int* |  |  | Biaya surchange |
| 44 | biaya\_Penjemputan | *Int* |  |  | Biaya penjemputan |
| 45 | Sub\_Total | *Int* |  |  | Total |
| 46 | PPN | *Int* |  |  | PPN |
| 47 | Sub\_Biaya | *Int* |  |  | Sub biaya |
| 48 | Created\_at | *timestamp* |  |  | Waktu pembuatan |
| 49 | wil |  |  |  | wilayah |
| 50 | tuj |  |  |  | tujuan |

1. Response Message Format

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | status |  |  |  | True or false |
| 2 | Message |  |  |  | keterangan |

* 1. **invoice**

1. Service Name : invoice
2. Alur Proses :
3. API menerima request search dari form invoice (frontend)
4. API melakukan pencarian ke table “shipment”
5. API memberikan response message sesuai hasil pencarian
6. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 5 | No Resi |
| 2 | Tanggal | Date |  |  | Tanggal Invoice |
| 3 | Tujuan\_Kabupaten | Int |  | 11 | Kota Tujuan berupa combo box |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | *char* |  | 5 | No Resi |
| 2 | Nama\_Pengirim |  |  |  | Pengirim |
| 3 | Sub\_Biaya | *int* |  | 11 | Total Tagihan |
| 4 | no\_resi | *char* |  | 5 | Nomor Invoice |
| 5 | created\_at | *Date* |  |  | Tanggal Invoice |

1. Cetak / Unduh

Tombol Cetak memanggil fungsi “vm.print” dan meload halaman cetakInvoice, file php yang diload adalah “cetakInvoice.blade.php” yang berada di folder “resources/views/STTPB/”

* 1. **Invoice kredit**
  2. Service Name : invoicekredit

1. API menerima request search dari form invoice kredit (frontend)
2. API melakukan pencarian ke table “shipment”
3. API memberikan response message sesuai hasil pencarian
   1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 5 | No Resi |
| 2 | Customer | string |  |  | Nama customer |
| 3 | Tanggal | Date |  |  | Tanggal invoice |
| 3 | Tujuan\_Kabupaten | Int |  | 11 | Kota Tujuan berupa combo box |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 2 | Nama\_Pengirim | *varchar* |  | 191 | Customer |
| 3 | Sub\_Biaya | *int* |  | 11 | Total Tagihan |
| 4 | Weekstartdate weekenddate | *Date* |  |  | Tanggal Order |
| 5 | no\_resi | *char* |  | 5 | No Invoice |
| 6 | created\_at | *Date* |  |  | Tanggal Invoice |

1. Cetak / Unduh

Tombol Cetak memanggil fungsi “vm.print” dan meload halaman “cetakInvoicekredit”, file php yang diload adalah “cetakInvoicekredit.blade.php” yang berada di folder “resources/views/STTPB/”

* 1. **Form kasir** 
     + 1. Service Name : transaction
       2. Alur Proses :

1. API menerima request call No Resi, No Resi menggunakan autosuggest dengan memanggil API “noresipembayaran”
2. API melakukan pencarian No Resi sesuai dengan yang diinput user pada table “shipment”
3. API memberikan response message berupa Total tagihan
4. Setelah user input jumlah uang yang dibayarkan,maka akan memanggil fungsi “hitungkembalian” , fungsi tersebut bisa dilihat di “public/js/angular\_js/controllers/PembayaranController.js”, fungsi tersebut akan memberikan response uang kembalian.
   * + 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 5 | No Resi  menggunakan autosuggest daftar nomor resi |
| 2 | jumlah\_dibayar | int |  | 20 | Jumlah uang yang dibayarkan |

* + - 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | jumlah\_total\_tagihan | int |  | 11 | Jumlah Tagihan |

Saat tombol “Simpan” diklik maka akan memanggil fungsi “vm.add()”,fungsi tersebut menghitung uang kembalian, dan melakukan validasi data yang diinput user, apabila No\_Resi sudah diisi dan bersifat unik, sub\_total sudah terisi, jumlah\_uang sudah diisi, kembalian sudah diisi dan bersifat numerik serta minimal nilainya adalah 0, maka fungsi tersebut akan menyimpan data tersebut ke table “transactions” dan memberikan response status “success” atau “error”

Tombol Cetak Invoice diklik maka akan memanggil fungsi “vm.cetakkasir('{{url('cetakInvoice')}}” dimana membuka halaman browser baru dan isi dari invoice tersebut bisa dilihat pada url “cetakInvoice”, url tersebut diarahkan menuju file “invoiceController.php” dengan fungsi “cetak”.

Untuk membuat suatu halaman invoice berikut data yang dibutuhkan :

* Logo
* Data perusahaan (diambil dari table setting)
* Data pengiriman (diambil dari table shipment)
  1. **Input pembayaran**

1. Service Name : inputpembayaran
2. Alur Proses :
3. API menerima request call No Resi, No Resi menggunakan autosuggest dengan memanggil API “noresiinputpembayarankredit”
4. API melakukan pencarian No Resi sesuai dengan yang diinput user pada table “shipment”
5. API memberikan response message berupa Total tagihan dan jumlah dibayar
6. user input no bukti bayar dan upload bukti bayar
7. User klik tombol “simpan” maka akan menyimpan data ke table “inputpembayaran”
8. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | No\_Resi | char |  | 15 | No Resi  menggunakan autosuggest daftar nomor resi |
| 2 | id\_transactions | int |  | 11 | No bukti bayar |
| 3 | Upload | *varchar* |  | 191 | Upload bukti bayar(jika ada)  Biasanya berupa gambar |

* 1. **Rekonsiliasi Pembayaran**

1. Service Name : rekonsiliasi
2. Alur Proses :
3. API menerima request pencarian daftar rekonsiliasi (frontend)
4. API melakukan pencarian ke table “transactions” dan “shipment” sesuai dengan filter yang diinput user
5. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No Resi |
| 2 | jumlah\_total\_tagihan | int |  | 11 | Total tagihan |
| 3 | No bukti bayar | string |  |  | No bukti pembayaran |
| 4 | Bulan |  |  |  | Bulan transaksi (diambil dari created\_at) |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No Resi |
| 2 | Nama\_Pengirim | varchar |  | 191 | Nama Pengirim |
| 3 | Metode\_Pembayaran | varchar |  | 191 | Jenis Pembayaran |
| 4 | created\_at | timestamp |  |  | Tanggal Pembayaran |
| 5 | jumlah\_total\_tagihan | int |  | 11 | Total tagihan |

* 1. **Daftar Tagihan Kredit**

1. Service Name : daftartagihan
2. Alur Proses :
3. API menerima request pencarian dari form Daftar Tagihan Kredit (front end)
4. API melakukan pencarian ke table “Shipment” dimana Metode\_Pembayaran berupa kredit dan sesuai kata kunci yang direquest user
5. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nama\_Pengirim | varchar |  | 191 | Nama consignee |
| 2 | Status |  |  |  | Status tanggal jatuh tempo.  Berupa combo box dengan pilihan : aman, segera bayar, Jatuh Tempo, Telat Bayar |
| 3 | Jatuh Tempo |  |  |  | Rentang tanggal jatuh tempo |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No Invoice |
| 2 | Nama\_Pengirim | varchar |  | 191 | Nama Customer |
| 3 | No\_Telp\_Pengirim | varchar |  | 191 | Nomor Telpon |
| 4 | Sub\_Biaya | int |  | 11 | Total Tagihan |
| 5 | jatuhtempo | timestamp |  |  | Tanggal jatuh tempo |
| 6 | jatuhtempostatus |  |  |  | Diambil dari perhitungan pada kolom created\_at |

* 1. **Input outbond**

1. Service Name : inputOutbond
2. Alur Proses :
3. API menerima Request cari dari form input outbond (front end)
4. API melakukan pencarian ke table “manifest” berdasarkan kata kunci yang diinput
5. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest | varchar |  | 191 | No Manifest |
| 2 | Tanggal | varchar |  | 191 | Tanggal Manifest |
| 3 | name | varchar |  | 191 | Kota tujuan |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest | varchar |  | 191 | No Manifest |
| 2 | name | varchar |  | 191 | Kota tujuan |
| 3 | Tanggal | timestamp |  |  | Tanggal Manifest |
| 4 | No\_resi | char |  | 15 | No Resi |

* 1. **Form Generate Manifest**
     1. Service Name : findoutbond
     2. Alur Proses :

1. API menerima request “generate” dari popup / modal Generate Manifest (front end)
2. API findoutbond meload file OutbondController.php dengan memanggil fungsi “find” , routingnya bisa dilihat dari : Route::post('findoutbond', 'api\OutbondController@find');
3. API melakukan pencarian detail manifest ke table shipment dan detailmanifest
   * 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Kota | varchar |  | 191 | Kota tujuan  (berupa combo box) |

* + 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | status |  |  |  | 0=failed  1=success |
| 2 | Message |  |  |  | 0=Data Shipment untuk di generate kosong |
| 3. | Nomanifest | varchar |  | 191 | No Manifest |
| 4 | Nama\_Pengirim | varchar |  | 191 | Nama Pengirim |
| 5 | Alamat\_Asal | varchar |  | 191 | Alamat Asal |
| 6 | Alamat\_Tujuan | varchar |  | 191 | Alamat Tujuan |

* 1. **Label Package**

1. Service Name : PackageOutbond
2. Alur Proses :
3. API menerima request cari dari Daftar Package (front end)
4. API melakukan pencarian daftar package ke table shipment dan package
5. API memberikan response message sesuai hasil pencarian
6. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nopackage | varchar |  | 191 | Nomor Package |
| 2 | Tujuan\_Kabupaten | int |  | 11 | Kota Tujuan |
| 3 | No\_Telp\_Penerima | varchar |  | 191 | Nomor Pengirim |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | No\_resi | char |  | 15 | No Resi |
| 2 | Nopackage | varchar |  | 191 | No Label |
| 3 | kotaasal.name | varchar |  | 191 | Kota Asal |
| 4 | kotatujuan.name | varchar |  | 191 | Kota Tujuan |
| 5 | ship.Jumlah\_Barang | varchar |  | 191 | Collie |
| 6 | ship.Berat\_Barang | varchar |  | 191 | Berat Barang |

* 1. **Surat Jalan**

1. Service Name : suratjalan
2. Alur Proses :
3. API menerima request cari dari form surat jalan(front end)
4. API melakukan pencarian surat jalan ke table “suratjalan”
5. API memberikan response message berupa daftar surat jalan sesuai kata kunci yang direquest
6. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_suratjalan | char |  | 15 | Nomor surat |
| 2 | Tujuan\_Kabupaten | int |  | 11 | Kota Tujuan |
| 3 | Tanggal | Date |  |  | Tanggal cetak |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_suratjalan | char |  | 15 | Nomor surat |
| 2 | detail.no\_resi | char |  | 15 | no\_resi (detail) |
| 3 | d.Nopackage | varchar |  | 191 | D= detail No Package |
| 4 | ship.Nama\_Barang | varchar |  | 191 | Nama Barang |
| 5 | ship.Jumlah\_Barang | varchar |  | 191 | Jumlah Barang |
| 6 | ship.Berat\_Dibayar | Int |  | 11 | Berat Dibayar |

* 1. **Lacak Posisi**
  2. Service Name : Tracking
  3. Alur Proses :

1. API menerima request pencarian dari halaman Tracking (frontend)
2. API melakukan pencarian data ke table “shipment”, “manifest”, “detailmanifest” berdasarkan No Resi yang dikirim
3. API memberikan response message berupa data pengiriman
   1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |

* 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |
| 2 | Tanggal | date |  |  | Tanggal Kirim Kota Asal |
| 3 | tanggaltibakotatujuan | date |  |  | Tanggal Tiba Kota Tujuan |
| 4 | tanggalBarangDiTerima | date |  |  | Tanggal Ditirima |

* 1. **Input data posisi**

1. Service Name : Tracking
2. Alur Proses :

API menerima request pencarian dari halaman Tracking (frontend)

API melakukan pencarian data ke table “shipment”, “manifest”, “detailmanifest” berdasarkan No Resi yang dikiri

API memberikan response message berupa data pengiriman

1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No resi |
| 2 | Tanggal | date |  |  | Tanggal Kirim Kota Asal |
| 3 | tanggaltibakotatujuan | date |  |  | Tanggal Tiba Kota Tujuan |
| 4 | tanggalBarangDiTerima | date |  |  | Tanggal Ditirima |

* 1. **Form Update Posisi Barang**
     1. Service Name :
     2. Alur Proses :

1. API menerima request vm.edit(row) dari form input Data Posisi(front end)
2. API membuka form Update dengan kolom isian :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | Char |  | 15 | Nomor Resi |
| 2 | Tujuan\_Kabupaten | Int |  | 11 | Kota Tujuan |
| 3 | Tanggal | Date |  |  | Tanggal cetak |
| 4 | tanggaltibakotatujuan | Date |  |  | Tanggal tiba di kota tujuan |
| 5 | tanggalBarangDiTerima | Date |  | 120 | Tanggal barang diterima |
| 6 | Upload | varchar |  | 191 | Upload bukti pembayaran |

Fungsi “Simpan” memanggil function “inputTracking” di file “TrackingController.php” (app/Http/Controllers/api/ TrackingController.php)

* 1. **Report Cabang**

Service Name : reportingcabang

Alur Proses :

1. API menerima request pencarian dari form Report Cabang (front end)
2. API melakukan pencarian ke table detailmanifest, shipment, manifest (join)
3. API memberikan response message sesuai request message yang dikirim

Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Bulan transaksi |  |  |  | Berupa combo box nama bulan |
| 2 | Tahun transaksi |  |  |  | Berupa combo box tahun |
| 3 | Kode cabang |  |  |  | Berupa combo box cabang |

Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No Resi |
| 2 | kotatujuan.name | varchar |  | 191 | Kota Tujua |
| 3 | Sub\_Biaya | Int |  | 11 | Sub Biaya |
| 4 | status\_Pembayaran |  |  |  | Status Pembayaran |

* 1. **Report Customer**

1. Service Name : reportingcabang
2. Alur Proses :
3. API menerima request pencarian data dari form Report Customer (front end)
4. API melakukan pencarian ke table detailmanifest, shipment, manifest (join)
5. API memberikan response message sesuai request message yang dikirim
6. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Bulan transaksi |  |  |  | Berupa combo box nama bulan |
| 2 | Tahun transaksi |  |  |  | Berupa combo box tahun |
| 3 | nama\_agen |  |  |  | Customer |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | no\_resi | char |  | 15 | No Resi |
| 2 | kotatujuan.name | varchar |  | 191 | Kota Tujua |
| 3 | Sub\_Biaya | Int |  | 11 | Sub Biaya |
| 4 | status\_Pembayaran |  |  |  | Status Pembayaran |

* 1. **Daftar Tarif Pengiriman**
  2. Service Name : masterData/tarif
  3. Alur Proses :

1. API menerima request onSearch dari form Tarif Pengiriman
2. API melakukan pencarian tarif ke table “tarifs” berdasarkan kata kunci “Coverage Area / kode”
3. API memberikan response message berupa tarif pada masing-masing Coverage Area
   1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | int |  | 10 | Kode Tarif |
| 2 | name (tbl rgn\_city)  name(tbl rgn\_district)  nama\_kota(tbl coverage\_areas) | varchar |  | 255 | Coverage Area |

* 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | int |  | 10 | Kode Tarif |
| 2 | name (tbl rgn\_city)  name(tbl rgn\_district)  nama\_kota(tbl coverage\_areas) | varchar |  | 255 | Coverage Area |
| 3 | tarif | double |  | (11,2) | Tarif |
| 4 | keterangan | text |  |  | Keterangan |

* 1. **Form Tarif Pengiriman**

Service Name : Function vm.add() dan vm.editca(row)

Alur Proses :

1. API menerima request buka form Tarif pengiriman dari Tarif Pengiriman (front end)
2. Untuk edit, API akan memanggil data dari table “tarifs” sesuai dengan id yang dipilih dan ditampilkan pada form
3. Setelah user mengisi / mengubah data pada form dan melakukan request “vm.submitModal()”
4. API memberikan response message berhasil atau gagal sesuai berhasil / tidaknya menyimpan data pada table “tarifs”

Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Tarif |
| 2 | coverage\_area\_id | int |  | 11 | Berupa autocomplete nama wilayah |
| 3 | Tarif | double |  | 11,2 | Tarif pengiriman |
| 4 | Keterangan | text |  |  | Keterangan |

* 1. **Form biaya asuransi**

1. Service Name : masterData/biaya\_asuransi
2. Alur Proses :
3. API menampilkan Daftar Asuransi MITRAX dari table “biaya\_asuransies”
4. Data yang ditampilkan antara lain :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Formula | varchar |  | 191 | Persentase untuk perhitungan biaya asuransi |
| 2 | biaya\_administrasi | int |  | 11 | Biaya Administrasi |

1. “vm.edit(row)” memanggil fungsi update pada file BiayaAsuransiController.php, fungsi tersebut digunakan untuk memperbarui data sesuai “id”
   1. **Daftar Cabang**
   2. Service Name : masterData/cabang
   3. Alur Proses :
2. API menerima request pencarian (vm.onSearch()) daftar cabang dari form Daftar Cabang (front end)
3. API melakukan pencarian ke table “cabangs” sesuai kata kunci nama cabang atau kode cabang atau kepala cabang
4. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Kode | char |  | 3 | Kode Cabang |
| 2 | Nama | varchar |  | 191 | Nama Cabang |
| 3 | kepala\_cabang | varchar |  | 191 | Kepala Cabang |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Cabang |
| 2 | nama | varchar |  | 191 | Nama Cabang |
| 3 | alamat | text |  |  | Alamat cabang |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor Telpon Cabang |
| 5 | kepala\_cabang | varchar |  | 191 | Kepala Cabang |
| 6 | email | varchar |  | 191 | Alamat Email |

1. Fungsi vm.edit(row), vm.add(), vm.del(row) bisa dilihat di file masterDataControllers (folder public/js/angular\_js/controllers) bagian cabangController, dan fungsi tersebut menjalankan fungsi-fungsi di CabangController.php (app/Http/MasterData/CabangController.php)
   1. **Form cabang**
2. Service Name : api/getkode/cabang
3. Alur Proses :
4. API menerima request buka form dari daftar cabang (front end)
5. Pada fungsi edit, API melakukan pencarian data ke table “cabangs” sesuai id yang dipilih lalu menampilkannya pada form cabang
6. Pada fungsi “add” API menampilkan Kode secara otomatis
7. Kolom yang harus diisi pada form cabang :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Kode | char |  | 3 | Kode Cabang. Tergenerate otomatis oleh sistem. |
| 2 | Nama | varchar |  | 191 | Nama Cabang |
| 3 | Alamat | text |  |  | Alamat cabang |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor Telpon Cabang |
| 5 | kepala\_cabang | varchar |  | 191 | Kepala Cabang |
| 6 | Email | varchar |  | 191 | Alamat Email |

1. Saat klik tombol ”Simpan”, maka akan memanggil fungsi “store” pada file “’CabangController.php” (terletak di folder app/Http/Controllers/MasterData/cabangController.php)
   1. **Daftar pelanggan**
2. Service Name : masterData/customer
3. Alur Proses :
4. API menerima request pencarian cutomer dari form Daftar customer (front end)
5. API melakukan pencarian ke table “`customers`”
6. API memberikan response message berupa data customer sesuai request message pencarian

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Kode | char |  | 3 | Kode Pelanggan |
| 2 | Nama | varchar |  | 191 | Nama Pelanggan |
| 3 | nomor\_rekening | varchar |  | 191 | Nomor Rekening |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Pelanggan |
| 2 | nama | varchar |  | 191 | Nama Pelanggan |
| 3 | alamat | Text |  |  | Alamat pelanggan |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor Telepon |
| 5 | nomor\_rekening | varchar |  | 191 | Nomor Rekening |
| 6 | email | varchar |  | 191 | Email |

* 1. **Form Pelanggan**

1. Service Name : masterData/customer
2. Alur Proses :
3. API menerima request buka form dari daftar Pelanggan (front end)
4. API menampilkan kolom isian sebagai berikut

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode | char |  | 3 | Kode Pelanggan (otomatis tergenerate) |
| 2 | nama | varchar |  | 191 | Nama pelanggan |
| 3 | alamat | *text* |  |  | Alamat pelanggan |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor telpon pelanggan |
| 5 | contact\_person | *varchar* |  | 191 | Nama kontak person pelanggan |
| 6 | nomor\_rekening | *varchar* |  | 191 | Nomor rekening pelanggan |
| 7 | email | *varchar* |  | 191 | Email pelanggan |

* 1. **Daftar Agen**

1. Service Name : masterData/agen
2. Alur Proses :
   1. API menerima request pencarian agen dari form Daftar Agen (front end)
   2. API melakukan pencarian ke table “agens”
   3. API memberikan response message berupa data agen sesuai request message pencarian
3. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_agen | char |  | 3 | Kode Agen |
| 2 | nama\_agen | varchar |  | 191 | Nama Agen |
| 3 | email | varchar |  | 191 | Email agen |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_agen | char |  | 3 | Kode Agen |
| 2 | nama\_agen | varchar |  | 191 | Nama Agen |
| 3 | alamat | text |  |  | Alamat |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor telepon |
| 5 | contact\_person | varchar |  | 191 | Contact Person |
| 6 | nomor\_rekening | varchar |  | 191 | Nomor rekening |
| 7 | email | varchar |  | 191 | Email |

* 1. **Form Agen**
     1. Service Name :
     2. Alur Proses :

1. API menerima request buka form agen dari daftar agen (front end)
2. API menampilkan kolom isian sebagai berikut

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_agen | char |  | 3 | Kode Agen |
| 2 | nama\_agen | varchar |  | 191 | Nama Agen |
| 3 | Alamat | *text* |  |  | Alamat agen |
| 4 | nomor\_telepon | varchar |  | 191 | Nomor telpon agen |
| 5 | contact\_person | *varchar* |  | 191 | Nama kontak person |
| 6 | nomor\_rekening | *varchar* |  | 191 | Nomor rekening agen |
| 7 | Jenis\_Bank | *varchar* |  | 255 | Jenis bank |
| 8 | Cabang\_Bank | *varchar* |  | 255 | Cabang bank |
| 9 | Email | *varchar* |  | 191 | Email agen |

* 1. **Daftar Wilayah Jangkauan**
     + 1. Service Name : masterData/coverageArea
       2. Alur Proses :
  2. API menerima request pencarian wilayah jangkauan dari form Daftar Coverage Area (front end)
  3. API melakukan pencarian ke table “coverage\_areas”
  4. API memberikan response message berupa data agen sesuai request message pencarian
     + 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_kota | char |  | 3 | Kode Kota |
| 2 | call\_sign | char |  | 3 | Call Sign |

* + - 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_kota | char |  | 3 | Kode Kota |
| 2 | call\_sign | char |  | 3 | Call Sign |
| 3 | nama\_kota | varchar |  | 191 | Nama Kota |
| 4 | province\_id | int |  | 11 | Provinsi |

* 1. **Form Wilayah Jangkauan**
     + 1. Service Name : masterData/coverageArea
       2. Service Name :
       3. Alur Proses :

API menerima request buka form coverage area dari daftar coverage area (front end)

API menampilkan kolom isian sebagai berikut

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | kode\_kota | char |  | 3 | Kode Kota |
| 2 | call\_sign | char |  | 3 | Call Sign |
| 3 | wilayah | varchar |  | 191 | Wilayah berupa autocomplete nama daerah |
| 4 | province\_id | int |  | 11 | Provinsi |
| 5 | nama\_kota | varchar |  | 191 | Nama kota |
| 6 | id | int |  | 10 | Kode kecamatan |

* 1. **Daftar Berita**

1. Service Name : masterData/berita
2. Alur Proses :

API menerima request pencarian berita dari form Daftar berita (front end)

API memberikan response message berupa data beita sesuai request message pencarian

1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | title\_ind / title\_eng / id\_news |  |  |  | Judul berita / kode berita |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | title\_ind |  |  |  | Judul berita berbahasa Indonesia |
| 2 | title\_eng |  |  |  | Judul berita berbahasa Inggris |
| 3 | kat\_berita |  |  |  | Keterangan berita |
| 4 | content\_ind |  |  |  | Isi berita berbahasa Indonesia |
| 5 | content\_eng |  |  |  | Isi berita berbahasa inggris |
| 6 | highlight |  |  |  | Draft / publish |
| 7 | status |  |  |  | Active/inactive |

* 1. **Daftar Barang Yang DIterima MITRAX**
     + 1. Service Name : barangkirim
       2. Alur Proses :

1. API menerima request pencarian dari form daftar barang yang diterima MitraX(front end)
2. API melakukan pencarian data ke table “manifest\_barangkirim” dan “agens”
   * 1. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nama\_agen | *varchar* |  | 191 | Nama agen |
| 2 | Nomanifest\_barangkirim | *varchar* |  | 191 | Nomor manifest |

* + 1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest\_barangkirim | *varchar* |  | 191 | Nomor manifest |
| 2 | nama\_agen | *varchar* |  | 191 | Agen Pengirim |
| 3 | name | *varchar* |  | 255 | Asal Kota |
| 4 | Tanggal | *timestamp* |  |  | Tanggal Kedatangan |

* 1. **Form Entry**

1. Alur Proses :
2. API menerima request buka form entry dari form daftar barang (front end)
3. API menampilkan kolom yang harus diisi, sebagai berikut :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Nomanifest | *varchar* |  | 191 | No Manifest |
| 2 | Kode\_Agent | *int* |  | 11 | Agen pengirim |
| 3 | Asal\_Kabupaten | *int* |  | 11 | Kota Asal |
| 4 | Tanggal | *varchar* |  | 191 | Tanggal Kedatangan |
| 5 | Berat\_Barang | *varchar* |  | 191 | Total Kg |
| 6 | no\_resi | *char* |  | 15 | No Resi |
| 7 | Nama\_Penerima | *varchar* |  | 191 | Nama Penerima |
| 8 | Alamat\_Tujuan | *varchar* |  | 191 | Alamat penerima |
| 9 | Nama\_Pengirim | *varchar* |  | 191 | Nama pengirim |

* 1. **Pengaturan Aplikasi**

1. Service Name : settingaplikasi
2. Alur Proses :
3. API “settingaplikasi” meload fungsi “setting\_submit” di file “PengaturanController.php” yang terletak di folder ”app/Http/Controllers/api/”
4. API menerima request update dari Pengaturan Aplikasi (Front End )
5. API memberikan response message sesuai dengan hasil update pada table “setting”
6. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | nameApp | *varchar* |  | 191 | Nama aplikasi |
| 2 | sloganApp | *varchar* |  | 191 | Slogan yang tampil pada aplikasi |
| 3 | alamatApp | *varchar* |  | 191 | Alamat yang tampil pada aplikasi |
| 4 | headSTTPB | *varchar* |  | 191 | Informasi header Kanan STTPB |
| 5 | headSTTPB2 | *varchar* |  | 191 | Informasi header Kanan STTPB 2 |
| 6 | headSTTPB3 | *varchar* |  | 191 | Informasi header Kanan STTPB 3 |
| 7 | Logo | *varchar* |  | 191 | Logo aplikasi |
| 8 | Headerpengirim | *text* |  |  | Footer cetak (informasi yang tercetak pada footer) |
| 9 | nomor\_rekening | *char* |  | 60 | Nomor rekening |

* 1. **Manage User**

Fungsi-fungsi terdapat di public/js/angular\_js/controller/manageUsersController.js

* Service Name : vm.Cari()

1. Alur Proses :
2. Sistem menerima request pencarian daftar user dengan kolom “email”
3. Sistem melakukan pencarian dengan cara cek email ke tabel “users”
4. Sistem memberikan response message sesuai hasil pengecekan data user yang mengandung email tersebut
5. Request Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Email | *varchar* |  | 191 | Email user |

1. Response Message

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | Name | *varchar* |  | 191 | Nama user |
| 2 | Email | *varchar* |  | 191 | Email user |
| 3 | created\_at | *timestamp* |  |  | Waktu pembuatan user |

* Service Name : vm.del(row)

1. Alur Proses :
2. Sistem menerima request hapus user
3. Row pada fungsi tersebut adalah primary key users yang hendak dihapus, yaitu “id”
4. Sistem menghapus data sesuai id yang dipilih

**1.44. Tambah User**

Service Name : vm.add()

1. Alur Proses :
2. Sistem menerima request tambah user
3. Sistem membuka popup/modal form tambah user
4. Pada form tambah user, kolom yang harus diisi antara lain:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | name | *varchar* |  | 191 | Nama user |
| 2 | email | *varchar* |  | 191 | Email user |
| 3 | Password | *varchar* |  | 191 | Password untuk user login, minimal 6 karakter |
| 4 | role\_id | *int* |  | 10 | Hak akses user terhadap aplikasi |

1. Sistem memberikan response message sesuai hasil input data

**1.45. Update User**

Service Name : vm.edit(row)

1. Alur Proses :
2. Sistem menerima request edit user
3. Row pada fungsi tersebut adalah primary key users yang hendak diedit, yaitu “id”
4. Sistem membuka popup/modal form edit user sesuai id yang akan diupdate
5. Pada form edit user, kolom yang dapat diedit antara lain:

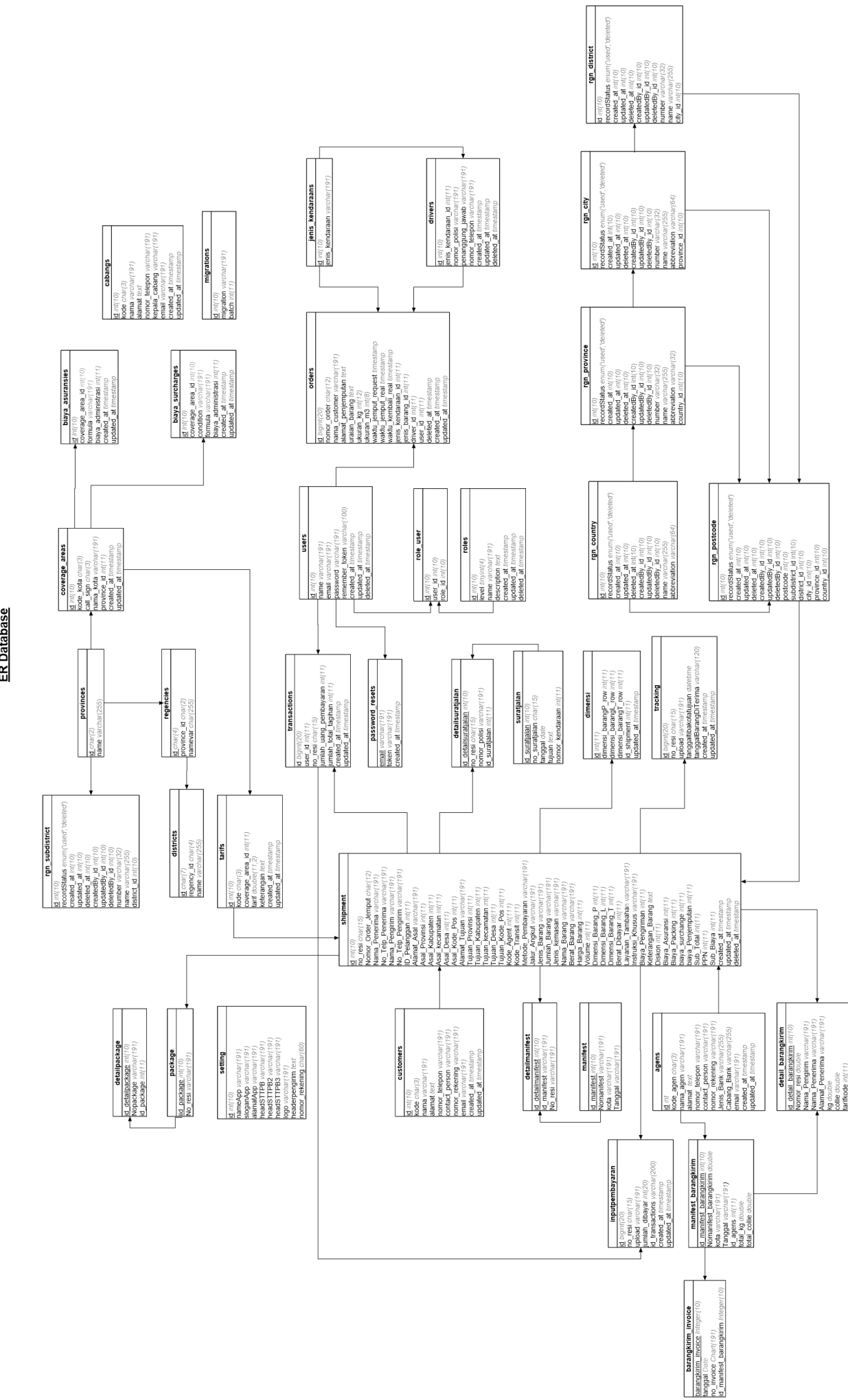
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | name | *varchar* |  | 191 | Nama user |
| 2 | email | *varchar* |  | 191 | Email user |
| 3 | role\_id | *int* |  | 10 | Hak akses user terhadap aplikasi |

1. Sedangkan untuk update password, user harus klik tombol “Ganti password”, maka sistem akan meng-enable kolom password.
2. Pada “Ganti Password”, kolom yang dapat diedit antara lain:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Name | Type | Status | Length | Description |
| 1 | name | *varchar* |  | 191 | Nama user |
| 2 | email | *varchar* |  | 191 | Email user |
|  | password | *varchar* |  | 191 | Password |
| 3 | role\_id | *int* |  | 10 | Hak akses user terhadap aplikasi |

1. Saat user klik tombol Update, sistem akan mengirim request tersebut dan memperbarui table “users”

# DIAGRAM RELASI TABLE



# SPESIFIKASI BASIS DATA

1. Table barangkirim\_invoice

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | Barangkirim\_invoice | Y | *Int* |  | 10 |  | Kode tagihan barangkirim |
| 2 | tanggal |  | *Date* |  |  |  | Tanggal |
| 3 | no\_invoice |  | *Char* |  | 191 |  | Nomor Invoice |
| 4 | id\_manifest\_barangkirim |  | *Int* |  | 10 |  | Id manifest |

1. Table manifest\_barangkirim

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_manifest\_barangkirim | Y | *Int* |  | 10 |  | Id |
| 2 | Nomanifest\_barangkirim |  | *double* |  |  |  | Nomor manifest |
| 3 | kota |  | *varchar* |  | 191 |  | Kota |
| 4 | Tanggal |  | *Date* |  |  |  | Tanggal |
| 5 | id\_agens |  | *int* |  | 11 |  | Kode agen |
| 6 | total\_kg |  | *double* |  |  |  | Total dalam kg |
| 7 | total\_collie |  | *double* |  |  |  | Total collie |

1. Table agens

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | ID |
| 2 | kode\_agen |  | *char* |  | 3 |  | Kode Agen |
| 3 | nama\_agen |  | *varchar* |  | 191 |  | Nama Agen |
| 4 | alamat |  | *text* |  |  |  | Alamat Agen |
| 5 | nomor\_telepon |  | *varchar* |  | 191 |  | No Telpon Agen |
| 6 | contact\_person |  | *varchar* |  | 191 |  |  |
| 7 | nomor\_rekening |  | *varchar* |  | 191 |  | No Rekening Agen |
| 8 | Jenis\_Bank |  | *varchar* |  | 255 |  | Jenis Bank no rekening terdaftar |
| 9 | Cabang\_Bank |  | *varchar* |  | 255 |  | Cabang dari bank no rekening terdaftar |
| 10 | email |  | *varchar* |  | 191 |  | Alamat email |
| 11 | created\_at |  | *timestamp* |  |  |  | Waktu input data agen |
| 12 | updated\_at |  | *timestamp* |  |  |  | Waktu update data agen |

1. Table users

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Kode user / pengguna sistem |
| 2 | name |  | *varchar* |  | 191 |  | Nama user / pengguna sistem |
| 3 | email |  | *varchar* |  | 191 |  | Alamat email user / pengguna sistem |
| 4 | password |  | *varchar* |  | 191 |  | Password masuk sistem |
| 5 | remember\_token |  | *varchar* |  | 100 |  |  |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input user |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update user |
| 8 | deleted\_at |  | *timestamp* |  |  |  | Waktu hapus user |

1. Table role\_user

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Kode role user |
| 2 | user\_id |  | *Int* |  | 10 |  | Kode user |
| 3 | role\_id |  | *int* |  | 10 |  | Kode role |

1. Table roles

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Kode Role user |
| 2 | level |  | *tinyint* |  | 4 |  | Level Role |
| 3 | name |  | *varchar* |  | 191 |  | Nama Role |
| 4 | description |  | *text* |  |  |  | Keterangan |
| 5 | created\_at |  | *timestamp* |  |  |  | Waktu pembuatan |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update |
| 8 | deleted\_at |  | *timestamp* |  |  |  | Waktu hapus |

1. Table shipment

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode |
| 2 | no\_resi |  | *char* |  | 15 |  | No Resi pengiriman |
| 3 | Nomor\_Order\_Jemput |  | *char* |  | 12 |  | No order penjemputan |
| 4 | Nama\_Penerima |  | *varchar* |  | 191 |  | Nama penerima |
| 5 | No\_Telp\_Penerima |  | *varchar* |  | 191 |  | No Telpon Penerima |
| 6 | Nama\_Pengirim |  | *varchar* |  | 191 |  | Nama Pengirim |
| 7 | No\_Telp\_Pengirim |  | *varchar* |  | 191 |  | No Telpon pengirim |
| 8 | ID\_Pelanggan |  | *int* |  | 11 |  | Kode pelanggan |
| 9 | Alamat\_Asal |  | *varchar* |  | 191 |  | Alamat asal pengiriman |
| 10 | Asal\_Provinsi |  | *int* |  | 11 |  | Kode provinsi asal pengiriman |
| 11 | Asal\_Kabupaten |  | *int* |  | 11 |  | Kode kabupaten asal pengiriman |
| 12 | Asal\_kecamatan |  | *int* |  | 11 |  | Kode kecamatan asal pengiriman |
| 13 | Asal\_Desa |  | *int* |  | 11 |  | Kode desa asal pengiriman |
| 14 | Asal\_Kode\_Pos |  | *int* |  | 11 |  | Kode pos asal pengiriman |
| 15 | Alamat\_Tujuan |  | *varchar* |  | 191 |  | Alamat tujuan pengiriman |
| 16 | Tujuan\_Provinsi |  | *int* |  | 11 |  | Kode provinsi tujuan pengiriman |
| 17 | Tujuan\_Kabupaten |  | *int* |  | 11 |  | Kode kabupaten tujuan pengiriman |
| 18 | Tujuan\_kecamatan |  | *int* |  | 11 |  | Kode kecamatan tujuan pengiriman |
| 19 | Tujuan\_Desa |  | *int* |  | 11 |  | Kode desa tujuan pengiriman |
| 20 | Tujuan\_Kode\_Pos |  | *int* |  | 11 |  | Kode pos tujuan pengiriman |
| 21 | Kode\_Agent |  | *int* |  | 11 |  | Kode Agen pengirim |
| 22 | Kode\_Transit |  | *int* |  | 11 |  | Kode transit |
| 23 | Metode\_Pembayaran |  | *varchar* |  | 191 |  | Metode pembayaran |
| 24 | Jalur\_Angkut |  | *varchar* |  | 191 |  | Jalur angkut pengiriman |
| 25 | Jenis\_Barang |  | *varchar* |  | 191 |  | Jenis barang yang dikirim |
| 26 | Jumlah\_Barang |  | *varchar* |  | 191 |  | Jumlah barang yang dikirim |
| 27 | Jenis\_kemasan |  | *varchar* |  | 191 |  | Jenis kemasan paket yang dikirim |
| 28 | Nama\_Barang |  | *varchar* |  | 191 |  | Nama barang yang dikirim |
| 29 | Berat\_Barang |  | *varchar* |  | 191 |  | Berat barang yang dikirim |
| 30 | Harga\_Barang |  | *int* |  | 11 |  | Harga barang yang dikirim |
| 31 | Volume |  | *int* |  | 11 |  | Volume barang yang dikirim |
| 32 | Dimensi\_Barang\_P |  | *int* |  | 11 |  | Dimensi barang (panjang) |
| 33 | Dimensi\_Barang\_L |  | *int* |  | 11 |  | Dimensi barang (lebar) |
| 34 | Dimensi\_Barang\_T |  | *int* |  | 11 |  | Dimensi barang (tinggi) |
| 35 | Berat\_Dibayar |  | *int* |  | 11 |  | Berat yang harus dibayar |
| 36 | Layanan\_Tambahan |  | *varchar* |  | 191 |  | Layanan tambahan yang dipilih |
| 37 | Instruksi\_Khusus |  | *varchar* |  | 191 |  | Instruksi khusu |
| 38 | Biaya\_Pengiriman |  | *int* |  | 11 |  | Biaya pengiriman barang |
| 39 | Keterangan\_Barang |  | *text* |  |  |  | Keterangan barang |
| 40 | Diskon |  | *int* | 11 |  |  | Diskon yang diberikan |
| 41 | Biaya\_Asuransi |  | *int* | 11 |  |  | Biaya asuransi yang dikenakan |
| 42 | Biaya\_Packing |  | *int* | 11 |  |  | Biaya packing |
| 43 | biaya\_surchange |  | *int* | 11 |  |  | Biaya surchange |
| 44 | biaya\_Penjemputan |  | *int* | 11 |  |  | Biaya penjemputan |
| 45 | Sub\_Total |  | *int* | 11 |  |  | total |
| 46 | PPN |  | *int* | 11 |  |  | PPN |
| 47 | Sub\_Biaya |  | *int* | 11 |  |  | Sub biaya |
| 48 | created\_at |  | *timestamp* |  |  |  | Waktu dibuat |
| 49 | updated\_at |  | *timestamp* |  |  |  | Waktu update |
| 50 | deleted\_at |  | *timestamp* |  |  |  | Waktu hapus |

1. Table detail\_barangkirim

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_detail\_barangkirim | Y | *Int* |  | 10 |  | Kode detail barang kirim |
| 2 | Nomor\_resi |  | *double* |  |  |  | Nomor resi pengiriman |
| 3 | Nama\_Pengirim |  | *varchar* |  | 191 |  | Nama pengirim |
| 4 | Nama\_Penerima |  | *varchar* |  | 191 |  | Nama penerima |
| 5 | Alamat\_Penerima |  | *varchar* |  | 191 |  | Alamat penerima |
| 6 | kg |  | *double* |  |  |  |  |
| 7 | collie |  | *double* |  |  |  |  |
| 8 | tarifkode |  | *int* |  | 11 |  | Kode tarif |
| 9 | id\_manifest\_barangkirim |  | *int* |  | 10 |  | Kode manifest barang kirim |

1. Table inputpembayaran

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *bigint* |  | 20 |  | kode |
| 2 | Nomor\_resi |  | *char* |  | 15 |  | Nomor resi pengiriman |
| 3 | upload |  | *varchar* |  | 191 |  |  |
| 4 | jumlah\_dibayar |  | *int* |  | 20 |  | Jumlah yang dibayarkan |
| 5 | id\_transactions |  | *varchar* |  | 200 |  | Kode transaksi |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input pembayaran |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update pembayaran |

1. Table customers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Id pelanggan |
| 2 | kode |  | *char* |  | 3 |  | Kode pelanggan |
| 3 | nama |  | *varchar* |  | 191 |  | Nama pelanggan |
| 4 | alamat |  | *text* |  |  |  | Alamat pelanggan |
| 5 | nomor\_telepon |  | *varchar* |  | 191 |  | Nomor telepon pelanggan |
| 6 | contact\_person |  | *varchar* |  | 191 |  | Kontak pelanggan |
| 7 | nomor\_rekening |  | *varchar* |  | 191 |  | Nomor rekening pelanggan |
| 8 | email |  | *varchar* |  | 191 |  | Alamat email pelanggan |
| 9 | created\_at |  | *timestamp* |  |  |  | Waktu input pelanggan |
| 10 | updated\_at |  | *timestamp* |  |  |  | Waktu update pelanggan |

1. Table manifest

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_manifest | Y | *int* |  | 10 |  | Kode manifest |
| 2 | Nomanifest |  | *varchar* |  | 191 |  | Nomor manifest |
| 3 | kota |  | *varchar* |  | 191 |  | Kota |
| 4 | Tanggal |  | *Date* |  |  |  | Tanggal |

1. Table detailmanifest

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_detailmanifest | Y | *int* |  | 10 |  | Kode detail manifest |
| 2 | id\_manifest |  | *varchar* |  | 191 |  | Kode manifest |
| 3 | No\_resi |  | *varchar* |  | 191 |  | Nomor resi |

1. Table transactions

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *bigint* |  | 20 |  | Kode transaksi |
| 2 | user\_id |  | *int* |  | 11 |  | Kode user |
| 3 | no\_resi |  | *char* |  | 15 |  | Nomor Resi |
| 4 | jumlah\_uang\_pembayaran |  | *int* |  | 11 |  | Jumlah uang yang dibayarkan |
| 5 | jumlah\_total\_tagihan |  | *int* |  |  |  | Jumlah tagihan |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input transaksi |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update transaksi |

1. Table tarifs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Id tarif |
| 2 | kode |  | *char* |  | 3 |  | Kode tarif |
| 3 | coverage\_area\_id |  | *char* |  | 14 |  | Kode area wilayah |
| 4 | tarif |  | *double* |  | 11,2 |  | tarif |
| 5 | keterangan |  | *text* |  |  |  | keterangan |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input tarif |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update tarif |

1. Table districts

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *char* |  | 7 |  | Kode |
| 2 | regency\_id |  | *char* |  | 4 |  | Kode Regency |
| 3 | name |  | *varchar* |  | 255 |  | Nama Distrik |

1. Table rgn\_subdistrict

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status Sub Distrik |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Diinput oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | number |  | *varchar* |  | 32 |  | Nomot sub distrik |
| 10 | name |  | *varchar* |  | 255 |  | Nama sub distrik |
| 11 | district\_id |  | *int* |  | 10 |  | Kode distrik |

1. Table **suratjalan**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_suratjalan | Y | *int* |  | 10 |  | Kode surat jalan |
| 2 | no\_suratjalan |  | *char* |  | 15 |  | No Surat jalan |
| 3 | tanggal |  | *date* |  |  |  | Tanggal surat Jalan |
| 4 | tujuan |  | *text* |  |  |  | Tujuan |
| 5 | nomor\_kendaraan |  | *int* |  | 11 |  | Nomor kendaraan pengantar |

1. Table detailsuratjalan

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_detailsuratjalan | Y | *int* |  | 10 |  | Kode detail surat jalan |
| 2 | no\_resi |  | *char* |  | 15 |  | No resi |
| 3 | nomor\_polisi |  | *Varchar* |  | 191 |  | Nomor polisi kendaraan |
| 4 | id\_suratjalan |  | *int* |  | 11 |  | Kode surat jalan |

1. Table dimensi

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 11 |  |  |
| 2 | dimensi\_barangP\_row |  | *int* |  | 11 |  | Dimensi barang (panjang) |
| 3 | dimensi\_barangL\_row |  | *int* |  | 11 |  | Dimensi barang (lebar) |
| 4 | dimensi\_barangT\_row |  | *int* |  | 11 |  | Dimensi barang (tinggi) |
| 5 | id\_shipment |  | *int* |  | 11 |  | Kode pengiriman |
| 6 | updated\_at |  | *timestamp* |  |  |  | Waktu update |

1. Table tracking

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *bigint* |  | 20 |  | Kode tracking |
| 2 | no\_resi |  | *char* |  | 15 |  | Nomor resi pengiriman |
| 3 | upload |  | *varchar* |  | 191 |  | File upload |
| 4 | tanggaltibakotatujuan |  | *datetime* |  |  |  | Tanggal tiba di kota tujuan |
| 5 | tanggalBarangDiTerima |  | *varchar* |  | 120 |  | Tanggal barang diterima |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input tracking |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update tracking |

1. Table rgn\_country

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | name |  | *varchar* |  | 255 |  | Nama |
| 10 | abbreviation |  | *varchar* |  | 64 |  |  |

1. Table rgn\_province

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | number |  | *varchar* |  | 32 |  |  |
| 10 | name |  | *varchar* |  | 255 |  | nama |
| 11 | abbreviation |  | *varchar* |  | 32 |  |  |
| 12 | country\_id |  | *int* |  | 10 |  | Kode negara |

1. Table rgn\_city

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | number |  | *varchar* |  | 32 |  | nomor |
| 10 | name |  | *varchar* |  | 255 |  | nama |
| 11 | abbreviation |  | *varchar* |  | 64 |  |  |
| 12 | province\_id |  | *int* |  | 10 |  | Kode provinsi |

1. Table rgn\_postcode

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | postcode |  | *Int* |  | 10 |  | Kode pos |
| 10 | subdistrict\_id |  | *Int* |  | 10 |  | Kode sub distrik |
| 11 | district\_id |  | *Int* |  | 10 |  | Kode distrik |
| 12 | city\_id |  | *int* |  | 10 |  | Kode kota |
| 13 | province\_id |  | *int* |  | 10 |  | Kode provinsi |
| 14 | country\_id |  | *int* |  | 10 |  | Kode negara |

1. Table rgn\_district

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu input |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu update |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu hapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | number |  | *varchar* |  | 32 |  | nomor |
| 10 | name |  | *varchar* |  | 255 |  | nama |
| 11 | city\_id |  | *int* |  | 10 |  | Kode kota |

1. Table drivers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | kode |
| 2 | jenis\_kendaraan\_id |  | *int* |  | 11 |  | Kode jenis kendaraan |
| 3 | nomor\_polisi |  | *varchar* |  | 191 |  | Nomor polisi kendaraan |
| 4 | penanggung\_jawab |  | *varchar* |  | 191 |  | Penanggung jawab |
| 5 | nomor\_telepon |  | *varchar* |  | 191 |  | Nomor telepon |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu input |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu update |
| 8 | deleted\_at |  | *timestamp* |  |  |  | Waktu hapus |

1. Table jenis\_kendaraans

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode jenis kendaraan |
| 2 | jenis\_kendaraan |  | *varchar* |  | 191 |  | Nama jenis kendaraan |

1. Table orders

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *bigint* |  | 20 |  | kode |
| 2 | nomor\_order |  | *char* |  | 12 |  | Nomor pemesanan |
| 3 | nama\_customer |  | *varchar* |  | 191 |  | Nama pelanggan |
| 4 | alamat\_penjemputan |  | *text* |  |  |  | Alamat penjemputan |
| 5 | uraian\_barang |  | *text* |  |  |  | Uraian barang yang dikirim |
| 6 | ukuran\_kg |  | *int* |  | 12 |  | Ukuran dalam kg |
| 7 | ukuran\_m3 |  | *int* |  | 8 |  | Ukuran volume |
| 8 | waktu\_jemput\_request |  | *timestamp* |  |  |  | Waktu jemput request pelanggan |
| 9 | waktu\_jemput\_real |  | *timestamp* |  |  |  | Waktu jemput real |
| 10 | waktu\_kembali\_real |  | *timestamp* |  |  |  | Waktu kembali real |
| 11 | jenis\_kendaraan\_id |  | *int* |  | 11 |  | Kode jenis kendaraan |
| 12 | jenis\_barang\_id |  | *int* |  | 11 |  | Kode jenis barang |
| 13 | driver\_id |  | *int* |  | 11 |  | Kode driver |
| 14 | user\_id |  | *int* |  | 11 |  | Kode user |
| 15 | deleted\_at |  | *timestamp* |  |  |  | Waktu hapus |
| 16 | created\_at |  | *timestamp* |  |  |  | Waktu input |
| 17 | updated\_at |  | *timestamp* |  |  |  | Waktu update |

1. Table cabangs

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Id cabang |
| 2 | kode |  | *char* |  | 3 |  | Kode cabang |
| 3 | nama |  | *varchar* |  | 191 |  | Nama cabang |
| 4 | alamat |  | *text* |  |  |  | Alamat |
| 5 | nomor\_telepon |  | *varchar* |  | 191 |  | Nomor telpon cabang |
| 6 | kepala\_cabang |  | *varchar* |  | 191 |  | Nama kepala cabang |
| 7 | email |  | *varchar* |  | 191 |  | Email |
| 8 | created\_at |  | *timestamp* |  |  |  | Waktu dibuat |
| 9 | updated\_at |  | *timestamp* |  |  |  | Waktu diupdate |

1. Table biaya\_asuransies

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Kode biaya asuransi |
| 2 | coverage\_area\_id |  | *Int* |  | 10 |  | Kode area |
| 3 | formula |  | *varchar* |  | 191 |  | Formula / rumus perhitungan |
| 4 | biaya\_administrasi |  | *int* |  | 11 |  | Biaya administrasi |
| 5 | created\_at |  | *timestamp* |  |  |  | Waktu dibuat |
| 6 | updated\_at |  | *timestamp* |  |  |  | Waktu diupdate |

1. Table biaya\_surcharges

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Id |
| 2 | coverage\_area\_id |  | *Int* |  | 10 |  | Kode area |
| 3 | condition |  | *varchar* |  | 191 |  | Kondisi |
| 4 | formula |  | *varchar* |  | 191 |  | Formula / rumus perhitungan biaya |
| 5 | biaya\_administrasi |  | *int* |  | 11 |  | Biaya administrasi |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu dibuat |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu diupdate |

1. Table coverage\_areas

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *Int* |  | 10 |  | Kode |
| 2 | kode\_kota |  | *char* |  | 3 |  | Kode kota |
| 3 | call\_sign |  | *char* |  | 3 |  | Call sign |
| 4 | nama\_kota |  | *varchar* |  | 191 |  | Nama kota |
| 5 | province\_id |  | *int* |  | 11 |  | Kode provinsi |
| 6 | created\_at |  | *timestamp* |  |  |  | Waktu dibuat |
| 7 | updated\_at |  | *timestamp* |  |  |  | Waktu diupdate |

1. Table provinces

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *char* |  | 2 |  | Kode provinsi |
| 2 | name |  | *varchar* |  | 255 |  | Nama provinsi |

1. Table regencies

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *char* |  | 4 |  | Kode |
| 2 | province\_id |  | *char* |  | 2 |  | Kode provinsi |
| 3 | name |  | *varchar* |  | 255 |  | Nama regency |

1. Table districts

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *char* |  | 7 |  | Kode |
| 2 | regency\_id |  | *char* |  | 4 |  | Kode regency |
| 3 | name |  | *varchar* |  | 255 |  | Nama distrik |

1. Table rgn\_subdistrict

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode |
| 2 | recordStatus |  | *enum('used','deleted')* |  |  |  | Status |
| 3 | created\_at |  | *int* |  | 10 |  | Waktu dibuat |
| 4 | updated\_at |  | *int* |  | 10 |  | Waktu diupdate |
| 5 | deleted\_at |  | *int* |  | 10 |  | Waktu dihapus |
| 6 | createdBy\_id |  | *int* |  | 10 |  | Dibuat oleh |
| 7 | updatedBy\_id |  | *int* |  | 10 |  | Diupdate oleh |
| 8 | deletedBy\_id |  | *int* |  | 10 |  | Dihapus oleh |
| 9 | number |  | *varchar* |  | 32 |  | Nomor |
| 10 | name |  | *varchar* |  | 255 |  | Nama sub distrik |
| 11 | district\_id |  | *int* |  | 10 |  | Kode distrik |

1. Table package

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_package | Y | *int* |  | 10 |  | Kode paket |
| 2 | No\_resi |  | *varchar* |  | 191 |  | Nomor resi |

1. Table detailpackage

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id\_detailpackage | Y | *int* |  | 10 |  | Kode detail paket |
| 2 | Nopackage |  | *varchar* |  | 191 |  | Nomor paket |
| 3 | id\_package |  | *int* |  | 11 |  | Kode paket |

1. Table setting

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode |
| 2 | nameApp |  | *char* |  | 2 |  | Nama aplikasi |
| 3 | sloganApp |  | *char* |  | 255 |  | Slogan aplikasi |
|  | alamatApp |  | *varchar* |  | 191 |  | Alamat aplikasi |
|  | headSTTPB |  | *varchar* |  | 191 |  | Informasi header 1 |
|  | headSTTPB2 |  | *varchar* |  | 191 |  | Informasi header 2 |
|  | headSTTPB3 |  | *varchar* |  | 191 |  | Informasi header 3 |
|  | logo |  | *varchar* |  | 191 |  | Logo |
|  | headerpengirim |  | *text* |  |  |  | Pengirim |
|  | nomor\_rekening |  | *char* |  | 60 |  | Nomor rekening |

1. Table migrations

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Column Name | PK | Type | NOT  NULL? | Length | Auto  Increment? | Description |
| 1 | id | Y | *int* |  | 10 |  | Kode migrasi |
| 2 | migration |  | *varchar* |  | 191 |  | Keterangan / nama migrasi |
| 3 | batch |  | *int* |  | 11 |  | batch |

Revision Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Version | Change Reference | Author | Reviewed by |
| [2018-09-04] | 0.1 |  | MEILAN ANASTASIA |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |