

A. Identitas Job Sheet

IDENTITAS JOB SHEET

| | | | |
|-----------------------|-------------------------------|---------------|------|
| Perguruan Tinggi | : Politeknik Negeri Bengkalis | Pertemuan Ke | : |
| Jurusan/Program Studi | : Teknik Informatika | Job Ke | : 4 |
| Kode Mata Kuliah | : | Halaman | : 38 |
| Nama Mata Kuliah | : MPPL | Mulai Berlaku | : |

B. Komponen Job Sheet

4.1. Judul Job

JOB III






























Implementasi CRUD (Create, Read, Update, & Delete)

4.2. Tujuan

1. Mahasiswa mampu mengimplementasikan fungsi import dan export di dalam Laravel menggunakan library pihak ketiga.

4.3. Implementasi

Pada bagian sebelumnya, Anda sudah berhasil mengimplementasikan fungsi CRUD (Create, Read, Update, & Delete) dalam framework laravel. Pada bagian ini, Anda akan mempelajari bagaimana caranya untuk melakukan join tabel antara tabel distributor dengan tabel produk dengan menggunakan query builder yang di miliki oleh laravel. Untuk itu, simak langkah-langkahnya dengan seksama agar tidak terjadi kesalahan. Pertama-tama Anda akan menambahkan 1 kolom baru dalam tabel products, untuk melakukan hal tersebut Anda terlebih dahulu perlu menghapus tabel **products** dan data riwayat migrations dari tabel **products** yang tersimpan dalam tabel migrations yang dapat dilihat pada gambar berikut ini (sesuaikan dengan database yang Anda miliki):

|   | | |  | id | migration | batch |
|---|--|--|--|----|---|-------|
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 1 | 0001_01_01_000000_create_users_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 2 | 0001_01_01_000001_create_cache_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 3 | 0001_01_01_000002_create_jobs_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 4 | 2024_09_10_050343_create_admins_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 5 | 2024_09_18_123105_create_products_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 6 | 2024_10_11_072546_create_distributors_table | 1 |
| <input type="checkbox"/> |  Edit |  Copy |  Delete | 7 | 2024_10_16_013252_create_histories_table | 2 |
| <div><div></div><div><input type="checkbox"/> Check all</div><div>With selected:</div><div> Edit</div><div> Copy</div><div> Delete</div><div> Export</div></div> | | | | | | |

Setelah berhasil menghapus tabel **products** dan data migrations dari tabel tersebut, kemudian ubah kode migrations dari tabel **products** dengan nama **2024_09_18_123105_create_products_table.php** (sesuaikan nama class migrations di project anda) seperti pada gambar dibawah ini:

```

1  use Illuminate\Database\Migrations\Migration;
2  use Illuminate\Database\Schema\Blueprint;
3  use Illuminate\Support\Facades\Schema;
4
5  return new class extends Migration
6  {
7      /**
8       * Run the migrations.
9       */
10     public function up(): void
11     {
12         Schema::create('products', function (Blueprint $table) {
13             $table->id();
14             $table->bigInteger('id_distributor');
15             $table->string('name');
16             $table->bigInteger('price');
17             $table->string('category');
18             $table->text('description');
19             $table->string('image');
20             $table->timestamps();
21         });
22     }
23
24     /**
25      * Reverse the migrations.
26      */
27     public function down(): void
28     {
29         Schema::dropIfExists('products');
30     }
31 };

```

Setelah itu, jangan lupa untuk menambahkan `id_distributor` pada models **Product.php** dengan kode seperti dibawah ini:

```
1 class Product extends Model
2 {
3     use HasFactory;
4
5     protected $fillable = [
6         'id_distributor', 'name', 'price', 'category', 'description', 'image'
7     ];
8 }
```

Setelah itu, Anda perlu melakukan migrasi database dengan mengetikkan perintah **"php artisan migrate"** dalam terminal Anda. Jika sudah berhasil melakukan migrate database, kemudian Anda perlu memperbarui beberapa tampilan seperti **indeks.blade.php**, **detail.blade.php**, **create.blade.php**, & **edit.blade.php** yang berada di dalam folder **views/pages/admin/product** seperti kode yang diberi tanda warna kuning berikut ini:

index.blade.php

```
@extends('layouts.admin.main')
@section('title', 'Admin Product')
@section('content')
<div class="main-content">
    <section class="section">
        <div class="section-header">
            <h1>Produk</h1>
            <div class="section-header-breadcrumb">
                <div class="breadcrumb-item active"><a href="{{ route('admin.dashboard') }}">Dashboard</a></div>
                <div class="breadcrumb-item">Produk</div>
            </div>
        </div>
    </div>
</div>
```

```
<a href="{{ route('product.create') }}" class="btn btn-icon icon-left btn-primary"><i class="fas fa-plus"></i> Produk</a>
```

```
<div class="card-body">
```

```
<div class="table-responsive">
```

```
<table class="table table-bordered table-md">
```

```
<tr>
```

```
<th>#</th>
```

```
<th>Nama Produk</th>
```

```
<th>Harga Produk</th>
```

```
<th>Nama Distributor</th>
```

```
<th>Action</th>
```

```
</tr>
```

```
@php
```

```
$no = 0
```

```
@endphp
```

```
@forelse ($data as $item)
```

```
<tr>
```

```
<td>{{ $no += 1 }}</td>
```

```
<td>{{ $item->name }}</td>
```

```
<td>{{ $item->price }} Points</td>
```

```
<td>{{ $item->nama_distributor }}</td>
```

```
<td>
```

```
<a href="{{ route('product.detail', $item->id) }}" class="badge badge-info">Detail</a>
```

```
<a href="{{ route('product.edit', $item->id) }}" class="badge badge-warning">Edit</a>
```

```
<a href="{{ route('product.delete', $item->id) }}" class="badge badge-danger" data-confirm-delete="true">Hapus</a>
```

```
</td>
```

```
</tr>
```

```
@empty
```

```
<td colspan="5" class="text-center">Data Produk Kosong</td>
```



```

        <div class="article-image" data-background="{{ asset('images/' . $data-
>image) }}"></div>
    </div>
    <div class="article-details">
        <div class="article-category"><a href="#">{{ $data->name }}</a> <div
class="bullet"></div> <a href="#">{{ $data->category }}</a></div>
        <div class="article-category"><a href="#">{{ $data->nama_distributor
}}</a></div>
        <div class="article-title">
            <h2><a href="#">Harga: {{ $data->price }} Points</a></h2>
        </div>
        <hr>
        <p>
            {{ $data->description }}
        </p>
    </div>
</article>
</div>
</div>
</section>
</div>
@endsection

```

create.bladed.php

```

@extends('layouts.admin.main')
@section('title', 'Admin Tambah Product')
@section('content')
    <div class="main-content">
        <section class="section">
            <div class="section-header">
                <h1>Tambah Produk</h1>
                <div class="section-header-breadcrumb">

```

```

        <div class="breadcrumb-item active"><a href="{{ route('admin.dashboard')
    }}">Dashboard</a></div>

        <div class="breadcrumb-item active"><a href="{{ route('admin.product')
    }}">Produk</a></div>

        <div class="breadcrumb-item">Tambah Produk</div>

    </div>

</div>

<a href="{{ route('admin.product') }}" class="btn btn-icon icon-left btn-warning">
Kembali</a>

<div class="card mt-4">

    <form action="{{ route('product.store') }}" class="needs-validation"
novalidate=""

    enctype="multipart/form-data" method="POST">

        @csrf

        <div class="card-body">

            <div class="row">

                <div class="col-6">

                    <div class="form-group">

                        <div class="form-group">

                            <label for="id_distributor">Nama Distributor</label>

                            <select name="id_distributor" class="form-control">

                                @foreach ($distributor as $item)

                                    <option value="{{ $item->id }}">{{ $item->nama_distributor
                                }}</option>

                                @endforeach

                            </select>

                        </div>

                    </div>

                </div>

                <div class="col-6">

                    <div class="form-group">

```

```

        <label for="name">Nama Produk</label>
        <input id="name" type="text" class="form-control" name="name"
required="">

        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-6">
    <div class="form-group">
        <label for="price">Harga Produk (Point)</label>
        <input id="price" type="number" class="form-control" name="price"
required="">

        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-6">
    <div class="form-group">
        <label for="category">Kategori Produk</label>
        <input id="category" type="text" class="form-control"
name="category"
        required="">

        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-12">
    <div class="form-group">
        <label for="description">Deskripsi Produk</label>

```



```

        <textarea class="form-control" name="description" id="description"
cols="30" rows="40" required=""></textarea>
        <div class="invalid-feedback">
            Isi berita harus di isi!
        </div>
    </div>
</div>
<div class="col-12">
    <div class="form-group">
        <div class="custom-file">
            <input class="custom-file-input" name="image" id="customFile"
type="file"
                required="">
            <label class="custom-file-label" for="customFile">Pilih
Gambar</label>
        </div>
        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
</div>
<button type="submit" class="btn btn-icon icon-left btn-primary">
    <i class="fas fa-plus"></i> Tambah
</button>
</div>
</form>
</div>
</section>
</div>
@endsection

```

Pada kode diatas, anda menambahkan sebuah fungsi looping atau perulangan foreach untuk menampilkan semua data distributor dari database, dimana dalam tag option

memiliki atribut value yang memiliki nilai id dari distributor dan di isi dengan nama distributor itu sendiri.

edit.blade.php

```
@extends('layouts.admin.main')
@section('title', 'Admin Edit Product')
@section('content')
<div class="main-content">
    <section class="section">
        <div class="section-header">
            <h1>Edit Produk</h1>
            <div class="section-header-breadcrumb">
                <div class="breadcrumb-item active"><a href="{{ route('admin.dashboard') }}">Dashboard</a></div>
                <div class="breadcrumb-item active"><a href="{{ route('admin.product') }}">Produk</a></div>
                <div class="breadcrumb-item">Edit Produk</div>
            </div>
        </div>

        <a href="{{ route('admin.product') }}" class="btn btn-icon icon-left btn-warning">
Kembali</a>

        <div class="card mt-4">
            <form action="{{ route('product.update', $product->id) }}" class="needs-validation"
novalidate="" enctype="multipart/form-data" method="POST">
                @csrf
                <div class="card-body">
                    <div class="row">
                        <div class="col-6">
                            <div class="form-group">
                                <div class="form-group">
                                    <label for="id_distributor">Nama Distributor</label>
```

```

        <select name="id_distributor" class="form-control">
            @foreach ($distributor as $item)
                <option value="{{ $item->id }}" {{ $product->id_distributor ==
$item->id ? 'selected' : '' }}>{{ $item->nama_distributor }}</option>
            @endforeach
        </select>
    </div>
</div>
</div>
</div>
<div class="col-6">
    <div class="form-group">
        <label for="name">Nama Produk</label>
        <input id="name" type="text" class="form-control" name="name"
required="" value="{{ $product->name }}">
        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-6">
    <div class="form-group">
        <label for="price">Harga Produk (Point)</label>
        <input id="price" type="number" class="form-control" name="price"
required="" value="{{ $product->price }}">
        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-6">
    <div class="form-group">
        <label for="category">Kategori Produk</label>

```

```

        <input id="category" type="text" class="form-control" name="category"
required="" value="{{ $product->category }}">
        <div class="invalid-feedback">
            Kolom ini harus di isi!
        </div>
    </div>
</div>
<div class="col-12">
    <div class="form-group">
        <label for="description">Deskripsi Produk</label>
        <textarea class="form-control" name="description" id="description"
cols="30" rows="40" required="">{{ $product->description }}</textarea>
        <div class="invalid-feedback">
            Isi berita harus di isi!
        </div>
    </div>
</div>
<div class="col-12">
    <div class="form-group">
        <div class="custom-file">
            <input class="custom-file-input" name="image" id="customFile"
type="file">
                <label class="custom-file-label" for="customFile">Pilih
Gambar</label>
        </div>
    </div>
</div>
<div>
    <button type="submit" class="btn btn-icon icon-left btn-primary">
        <i class="fas fa-save"></i> Simpan
    </button>
</div>
</form>

```

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

Pada kode diatas, anda melakukan looping atau perulangan seperti pada kode **create.blade.php**, namun yang berbeda adalah kode diatas terdapat if statement dimana fungsi if statement tersebut adalah menampilkan nilai yang sesuai dengan data yang terdapat di dalam database pada input select.

Setelah Anda memperbarui tampilan blade pada bagian product, selanjutnya Anda perlu mengubah beberapa function yang terdapat dalam **ProductController.php** dengan kode seperti berikut ini:

Function index

```
1 public function index()
2 {
3     $data = DB::table('distributors')
4         ->join('products', 'distributors.id', '=', 'products.id_distributor')
5         ->select('distributors.*', 'products.*')
6         ->get();
7
8     confirmDelete('Hapus Data!', 'Apakah anda yakin ingin menghapus data ini?');
9
10    return view('pages.admin.product.index', compact('data'));
11 }
```

Pada kode diatas, bagian inner join tabel distributors dan products dihubungkan menggunakan metode join pada kolom id dari tabel distributors dan kolom id_distributor dari tabel products. Sintaks **->join('products', 'distributors.id', '=', 'products.id_distributor')** memastikan bahwa hanya data yang memiliki kecocokan antara kedua tabel yang akan diambil, sehingga membentuk relasi inner join. Hasil join ini kemudian diambil menggunakan select untuk memilih semua kolom dari kedua tabel dengan format **select('distributors.*', 'products.*')**

Function detail

```
1 public function detail($id)
2 {
3     $data = DB::table('distributors')
4         →join('products', 'distributors.id', '=', 'products.id_distributor')
5         →select('products.*', 'distributors.*')
6         →where('products.id', '=', $id)
7         →first();
8
9     return view('pages.admin.product.detail', compact('data'));
10 }
```

Pada kode diatas, inner join dilakukan antara tabel distributors dan products dengan menggunakan `->join('products', 'distributors.id', '=', 'products.id_distributor')`, yang menghubungkan **distributors.id** dengan **products.id_distributor**. Join ini memastikan hanya data yang cocok antara kedua tabel yang diambil. Setelah itu, **where('products.id', '=', \$id)** digunakan untuk memfilter produk tertentu berdasarkan id yang diterima sebagai parameter. Kemudian, **select('products.*', 'distributors.*')** memilih semua kolom dari kedua tabel, dan **first()** digunakan untuk mengambil hasil pertama dari query ini. Hasilnya adalah detail produk beserta data distributor yang terkait dengan produk tersebut.

Function create

```
1 public function create()
2 {
3     $distributor = Distributor::all();
4
5     return view('pages.admin.product.create', compact('distributor'));
6 }
```

Function store

```
1 public function store(Request $request)
2 {
3     $validator = Validator::make($request->all(), [
4         'id_distributor' => 'required|numeric',
5         'name' => 'required',
6         'price' => 'numeric',
7         'category' => 'required',
8         'description' => 'required',
9         'image' => 'required|mimes:png,jpeg,jpg',
10    ]);
11
12    if ($validator->fails()) {
13        Alert::error('Gagal!', 'Pastikan semua terisi dengan benar!');
14        return redirect()->back();
15    }
16
17    if ($request->hasFile('image')) {
18        $image = $request->file('image');
19        $imageName = time() . '.' . $image->getClientOriginalExtension();
20        $image->move('images/', $imageName);
21    }
22
23    $product = Product::create([
24        'id_distributor' => $request->id_distributor,
25        'name' => $request->name,
26        'price' => $request->price,
27        'category' => $request->category,
28        'description' => $request->description,
29        'image' => $imageName,
30    ]);
31
32    if ($product) {
33        Alert::success('Berhasil!', 'Produk berhasil ditambahkan!');
34        return redirect()->route('admin.product');
35    } else {
36        Alert::error('Gagal!', 'Produk gagal ditambahkan!');
37        return redirect()->back();
38    }
39 }
```

Function edit

```
1 public function edit($id)
2 {
3     $product = Product::findOrFail($id);
4     $distributor = Distributor::all();
5
6     return view('pages.admin.product.edit', compact('product', 'distributor'));
7 }
```

Function update

```
1 public function update(Request $request, $id)
2 {
3     $validator = Validator::make($request->all(), [
4         'id_distributor' => 'required|numeric',
5         'name' => 'required',
6         'price' => 'numeric',
7         'category' => 'required',
8         'description' => 'required',
9         'image' => 'nullable|mimes:png,jpeg,jpg',
10    ]);
11
12    if ($validator->fails()) {
13        Alert::error('Gagal!', 'Pastikan semua terisi dengan benar!');
14        return redirect()->back();
15    }
16
17    $product = Product::findOrFail($id);
18
19    if ($request->hasFile('image')) {
20        $oldPath = public_path('images/' . $product->image);
21        if (File::exists($oldPath)) {
22            File::delete($oldPath);
23        }
24
25        $image = $request->file('image');
26        $imageName = time() . '.' . $image->getClientOriginalExtension();
27        $image->move('images/', $imageName);
28    } else {
29        $imageName = $product->image;
30    }
31
32    $product->update([
33        'id_distributor' => $request->id_distributor,
34        'name' => $request->name,
35        'price' => $request->price,
36        'category' => $request->category,
37        'description' => $request->description,
38        'image' => $imageName,
39    ]);
40
41    if ($product) {
42        Alert::success('Berhasil!', 'Produk berhasil diperbarui!');
43        return redirect()->route('admin.product');
44    } else {
45        Alert::error('Gagal!', 'Produk gagal diperbarui!');
46        return redirect()->back();
47    }
48 }
```

Jangan lupa untuk menambahkan models **Distributor** pada bagian paling atas sebelum class **ProductController**. Setelah memperbarui bagian product, Anda akan mengimplementasikan fungsi import dan export dalam project Anda. Sebelum itu, Anda perlu melakukan proses instalasi terhadap 2 library yakni **maatwebsite (excel)** dan **barryvdh (PDF)**. Untuk melakukan proses instalasi, Anda perlu menjalankan perintah “**composer require maatwebsite/excel**” kemudian “**composer require barryvdh/laravel-dompdf**”, serta jangan lupa untuk mempublish kedua library tersebut

dengan perintah “**php artisan vendor:publish --all**” yang Anda jalankan dalam terminal Anda. Setelah berhasil melakukan instalasi terhadap 2 library tersebut, jalankan perintah “**php artisan make:import DistributorImport --model=Distributor**” untuk menambahkan class yang akan menangani fungsi import data pada tabel **distributors**. Ketika berhasil menjalankan perintah tersebut, maka laravel akan membuat sebuah folder baru dengan nama **import** yang berada dalam folder **app**. Selanjutnya, tulislah kode dibawah ini pada class **DistributorImport.php** yang berada dalam folder **app/import**:

```
1 namespace App\Imports;
2
3 use App\Models\Distributor;
4 use Maatwebsite\Excel\Concerns\ToModel;
5 use Maatwebsite\Excel\Concerns\WithHeadingRow;
6
7 class DistributorImport implements ToModel, WithHeadingRow
8 {
9     /**
10      * @param array $row
11      *
12      * @return \Illuminate\Database\Eloquent\Model|null
13      */
14     public function model(array $row)
15     {
16         return new Distributor([
17             'nama_distributor' => $row['nama_distributor'],
18             'kota' => $row['kota'],
19             'provinsi' => $row['provinsi'],
20             'kontak' => $row['kontak'],
21             'email' => $row['email'],
22         ]);
23     }
24 }
```

Kode di atas adalah implementasi dari package **maatwebsite/excel** yang digunakan untuk mengimpor data distributor dari file Excel ke dalam database Laravel. Pada kode ini, kelas **DistributorImport** mengimplementasikan dua interface: **ToModel** dan **WithHeadingRow**. Interface **ToModel** memungkinkan setiap baris dalam file Excel untuk dikonversi menjadi model **Distributor**. Metode **model()** menerima data setiap baris sebagai array **\$row**, kemudian mengembalikan objek **Distributor** baru yang berisi data

dari kolom nama_**distributor**, **kota**, **provinsi**, **kontak**, dan **email** sesuai dengan key di array **\$row**. Interface **WithHeadingRow** memberi tahu Laravel bahwa baris pertama dari file Excel berisi nama kolom, sehingga **maatwebsite/excel** akan otomatis menyesuaikan data berdasarkan heading tersebut, membuat kode lebih mudah dibaca dan menghindari kesalahan urutan kolom.

Kemudian, Anda perlu menambahkan 1 function baru ke dalam **DistributorController.php** untuk menangani proses import data dengan kode dibawah ini:

```

use App\Imports\DistributorImport;

use Maatwebsite\Excel\Facades\Excel;

public function import(Request $request)
{
    try {
        $file = $request->file('file');
        Excel::import(new DistributorImport, $file);

        Alert::success('Berhasil!', 'Data berhasil di import!');
    } catch (\Maatwebsite\Excel\Validators\ValidationException $e) {
        $failures = $e->failures();
        $messages = "";
        foreach ($failures as $failure) {
            $messages .= 'Kesalahan pada baris ' . $failure->row() . ': ' . implode(', ', $failure-
>errors()) . ' . ';
        }

        Alert::error('Gagal!', 'Validasi Gagal: ' . $messages);
    } catch (\Exception $e) {
        Alert::error('Gagal!', 'Pastikan format dan isi sudah benar! Error: ' . $e-
>getMessage());
    } finally {
        return redirect()->back();
    }
}

```

```
}
```

Pada function import diatas, memiliki fungsi untuk mengimpor data dari file Excel yang diunggah oleh pengguna. Proses dimulai dengan menerima file dari request menggunakan `$request->file('file')`. File yang diterima kemudian diproses menggunakan `'Excel::import()'` yang mengarah pada kelas `'DistributorImport'`. Kelas ini bertugas untuk memetakan data dari file Excel ke dalam struktur yang sesuai dengan database, serta melakukan validasi data. Jika terdapat kesalahan selama proses validasi, seperti data yang tidak sesuai format atau aturan yang telah ditetapkan, maka exception `'ValidationException'` akan ditangani dalam blok `'catch'`. Setiap kesalahan yang terjadi akan dikumpulkan dan ditampilkan dalam pesan error, yang menunjukkan baris mana dalam file yang mengandung kesalahan dan jenis kesalahannya. Jika tidak ada kesalahan dalam validasi, maka proses import dianggap berhasil dan pengguna akan diberi notifikasi sukses menggunakan `Alert`. Jika terjadi kesalahan lain (seperti error sistem atau masalah dengan file), maka exception umum akan ditangani dan pesan error yang lebih umum akan ditampilkan. Setelah semua proses selesai, halaman akan diarahkan kembali ke halaman sebelumnya.

Setelah menambahkan function untuk melakukan proses import data, tambahkan juga function untuk menangani proses import data ke dalam file PDF seperti berikut ini:

```
use Barryvdh\DomPDF\Facade\Pdf;

public function export()
{
    $distributors = Distributor::all();

    $pdf = Pdf::loadView('pages.admin.distributor.export', compact('distributors'))
        ->setPaper('a4', 'landscape');

    return $pdf->download('distributor.pdf');
}
```

Pada kode diatas, function export berfungsi untuk mengeksport data distributor dari database ke dalam format PDF. Fungsi ini dimulai dengan mengambil semua data distributor menggunakan `'Distributor::all()'`. Data yang diambil kemudian diteruskan ke tampilan `'pages.admin.distributor.export'` dengan cara mengirimkan data tersebut

melalui `compact('distributors')`. Tampilan ini akan di-render untuk mempersiapkan data yang akan dimasukkan dalam file PDF. Setelah itu, PDF dibuat menggunakan Laravel DomPDF dengan `Pdf::loadView()`, yang memuat tampilan yang telah dipersiapkan. Format PDF diatur dengan ukuran kertas A4 dan orientasi landscape. Setelah PDF selesai dibuat, file tersebut langsung didownload oleh pengguna dengan nama `distributor.pdf`. Fungsi ini memungkinkan pengguna untuk mendapatkan laporan data distributor dalam bentuk yang mudah dibaca dan dicetak.

Karena function export memerlukan sebuah halaman blade, Anda perlu menambahkan file blade baru dengan nama **export.blade.php** di dalam folder **views/pages/admin/distributor**, kemudian tulislah kode dibawah ini pada halaman blade tersebut:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Export Data Distributor</title>

  <style>
    table {
      width: 100%;
      border-collapse: collapse;
      font-size: 14px;
    }
    th, td {
      border: 1px solid black;
      text-align: center;
      padding: 4px;
    }
    th {
      background-color: #f2f2f2;
```

```

    }
    .vertical-text {
        writing-mode: vertical-lr;
        transform: rotate(180deg);
    }
    .no-border {
        border: none;
    }
    .horizontal-span {
        text-align: center;
    }
    body {
        font-family: Arial, sans-serif;
    }
</style>
</head>
<body>
    <h4 style="text-align: center;">DATA DISTRIBUTOR</h4>

    <table>
        <thead>
            <tr>
                <th>No</th>
                <th>Nama Distributor</th>
                <th>Kota</th>
                <th>Provinsi</th>
                <th>Kontak</th>
                <th>Email</th>
            </tr>
        </thead>
    @php
        $no = 0;
    @endphp

```

```

<tbody>
    @foreach ($distributors as $item)
        <tr>
            <td>{{ $no += 1 }}</td>
            <td>{{ $item->nama_distributor }}</td>
            <td>{{ $item->kota }}</td>
            <td>{{ $item->provinsi }}</td>
            <td>{{ $item->kontak }}</td>
            <td>{{ $item->email }}</td>
        </tr>
    @endforeach
</tbody>
</table>
</body>
</html>

```

Setelah itu, Anda harus memperbarui route yang akan mengarahkan ke function yang baru saja Anda buat dengan kode seperti dibawah ini **(Ingat! Tambahkan kode dibawah ini pada route group admin!)**

```

1 Route::post('/distributor/import', [DistributorController::class, 'import'])->name('distributor.import');
2 Route::get('/distributor/export', [DistributorController::class, 'export'])->name('distributor.export');

```

Terkahir, Anda perlu memperbarui tampilan **index.blade.php** yang berada di dalam folder **views/pages/admin/distributor** kemudian tulislah kode dibawah ini:

```

@extends('layouts.admin.main')
@section('title', 'Admin Distributor')
@section('content')
    <div class="main-content">
        <section class="section">
            <div class="section-header">
                <h1>Distributor</h1>
            </div>
        </section>
    </div>

```

```
<div class="section-header-breadcrumb">
  <div class="breadcrumb-item active"><a href="{{ route('admin.dashboard')
}}">Dashboard</a></div>
  <div class="breadcrumb-item">Distributor</div>
</div>

<div class="row">
  <div class="col-md-4 col-sm-3">
    <a href="{{ route('distributor.create') }}" class="btn btn-icon icon-left btn-
primary">
      <i class="fas fa-plus"></i> Distributor
    </a>
    <a href="{{ route('distributor.export') }}" class="btn btn-icon icon-left btn-
info">
      <i class="fas fa-print"></i> Export
    </a>
  </div>
  <div class="col-md-8 col-sm-9">
    <form action="{{ route('distributor.import') }}" method="POST"
enctype="multipart/form-data">
      @csrf
      <div class="d-flex align-items-center">
        <div class="form-group mb-0 mr-2">
          <div class="custom-file">
            <input class="custom-file-input" name="file" id="customFile"
type="file" required="">
            <label class="custom-file-label" for="customFile">Pilih File
Excel</label>
          </div>
        </div>
        <button type="submit" class="btn btn-icon icon-left btn-primary">
          <i class="fas fa-plus"></i> Import
        </button>
      </div>
    </form>
  </div>
</div>
```

```

        </button>

    </div>

</form>

</div>

</div>

<div class="card-body">
    <div class="table-responsive">
        <table class="table table-bordered table-md">
            <tr>
                <th>#</th>
                <th>Nama Distributor</th>
                <th>Kota</th>
                <th>Provinsi</th>
                <th>Kontak</th>
                <th>Email</th>
                <th>Action</th>
            </tr>
            @php
                $no = 0;
            @endphp
            @forelse ($distributor as $item)
                <tr>
                    <td>{{ $no += 1 }}</td>
                    <td>{{ $item->nama_distributor }}</td>
                    <td>{{ $item->kota }}</td>
                    <td>{{ $item->provinsi }}</td>
                    <td>{{ $item->kontak }}</td>
                    <td>{{ $item->email }}</td>
                    <td>
                        <a href="{{ route('distributor.edit', $item->id) }}"
                            class="badge badge-warning">Edit</a>
                    </td>
                </tr>
            @endforelse
        </table>
    </div>
</div>

```



```

<a href="{{ route('distributor.delete', $item->id) }}" class="badge
badge-danger"

data-confirm-delete="true">Hapus</a>

</td>

</tr>

@empty

<td colspan="5" class="text-center">Data Distributor Kosong</td>

@endforelse

</table>

</div>

</div>

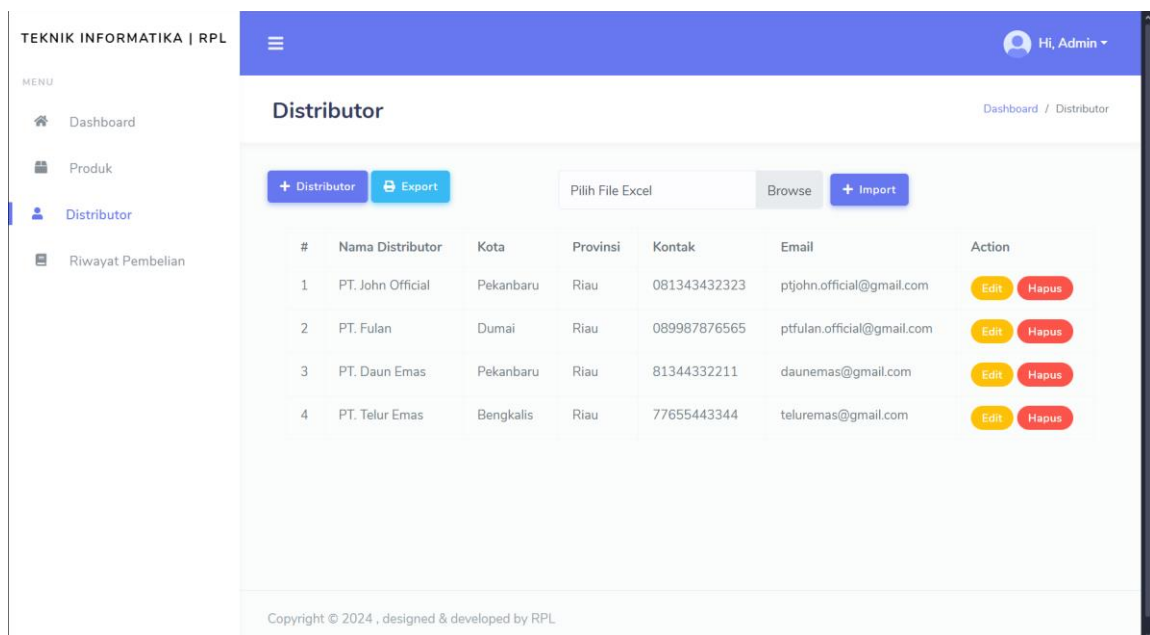
</section>

</div>

@endsection

```

Jika sudah berhasil maka, tampilan untuk halaman distributor akan terlihat seperti pada gambar dibawah ini:



Perlu diketahui juga bahwa format daripada file excel yang akan diimport harus sama dan sesuai dengan yang sudah Anda buat pada bagian **DistributorImport.php**, karena jika tidak sama maka akan terjadi kesalahan.

Selanjutnya Anda akan menambahkan 1 fitur lagi yang berada di bagian user, yakni history atau riwayat dari pembelian yang dilakukan oleh user. Untuk menambahkan fitur tersebut, Anda perlu menambahkan model baru dengan nama **History** dengan mengetikkan perintah berikut “**php artisan make:model History -m**”, lalu tambahkan kode dibawah ini pada class **History.php**:

```
1 namespace App\Models;
2
3 use Illuminate\Database\Eloquent\Factories\HasFactory;
4 use Illuminate\Database\Eloquent\Model;
5
6 class History extends Model
7 {
8     use HasFactory;
9
10     protected $fillable = ['id_user', 'id_product', 'total_harga'];
11 }
```

Kemudian di dalam file **2024_10_16_013252_create_histories_table.php** (sesuaikan dengan nama file dalam project anda), tambahkan kode dibawah ini:

```
1 use Illuminate\Database\Migrations\Migration;
2 use Illuminate\Database\Schema\Blueprint;
3 use Illuminate\Support\Facades\Schema;
4
5 return new class extends Migration
6 {
7     /**
8      * Run the migrations.
9      */
10    public function up(): void
11    {
12        Schema::create('histories', function (Blueprint $table) {
13            $table->id();
14            $table->bigInteger('id_user');
15            $table->bigInteger('id_product');
16            $table->bigInteger('total_harga');
17            $table->timestamps();
18        });
19    }
20
21    /**
22     * Reverse the migrations.
23     */
24    public function down(): void
25    {
26        Schema::dropIfExists('histories');
27    }
28 };
```

Setelah itu jalankan perintah “**php artisan migrate**”, lalu ubahlah function **purchase** yang terdapat pada **UserController.php** seperti pada kode dibawah ini:

```
1 public function purchase($productId, $userId)
2 {
3     $product = Product::findOrFail($productId);
4     $user = User::findOrFail($userId);
5
6     $currentTime = Carbon::now()->setTimezone('Asia/Jakarta');
7     $flashSaleStart = Carbon::createFromTimeString('12:00:00', 'Asia/Jakarta');
8     $flashSaleEnd = Carbon::createFromTimeString('23:00:00', 'Asia/Jakarta');
9
10    if ($currentTime->between($flashSaleStart, $flashSaleEnd)) {
11        $discountedPrice = $product->price * 0.5;
12    } else {
13        $discountedPrice = $product->price;
14    }
15
16    if ($user->point >= $discountedPrice) {
17        $totalPoints = $user->point - $discountedPrice;
18
19        $user->update([
20            'point' => $totalPoints,
21        ]);
22
23        History::create([
24            'id_user' => $userId,
25            'id_product' => $productId,
26            'total_harga' => $discountedPrice,
27        ]);
28
29        Alert::success('Berhasil!', 'Produk berhasil dibeli dengan harga ' . $discountedPrice);
30        return redirect()->back();
31    } else {
32        Alert::error('Gagal!', 'Point anda tidak cukup!');
33        return redirect()->back();
34    }
35 }
```

Pada kode diatas, Anda melakukan 2 proses yakni update dan store. Proses update dilakukan pada tabel **users** untuk memperbarui poin setelah user melakukan pembelian, sedangkan proses store dilakukan untuk menambahkan data pembelian baru ke dalam tabel **histories** yang akan digunakan untuk menampilkan riwayat pembelian user berdasarkan **id_user** dan **id_product**. Setelah itu, Anda tambahkan juga 2 function yang digunakan untuk menampilkan semua data riwayat pembelian user dan detail dari riwayat pembelian tersebut seperti berikut ini:

Function history

```
1 public function history($id)
2 {
3     $data = DB::table('histories')
4         →join('products', 'products.id', '=', 'histories.id_product')
5         →where('histories.id_user', '=', $id)
6         →get();
7
8     return view('pages.user.history', compact('data'));
9 }
```

Pada kode diatas, Anda melakukan inner join terhadap tabel **histories** dan **products** dimana anda menambahkan kondisi **where()** untuk hanya menampilkan atau mengambil data dari tabel **histories** yang **id_user** nya sesuai dengan **id** dari user yang sedang login saat itu. Setelah Anda menambahkan function untuk menampilkan semua data history, selanjutnya tulislah kode dibawah ini untuk function **detail_history**:

Function detail_history

```
1 public function detail_history($id)
2 {
3     $data = DB::table('histories')
4         →join('products', 'products.id', '=', 'histories.id_product')
5         →where('histories.id', '=', $id)
6         →first();
7
8     return view('pages.user.detail-history', compact('data'));
9 }
```

Pada kode diatas, Anda melakukan inner join terhadap tabel **histories** dan **products** dimana anda menambahkan kondisi **where()** untuk hanya menampilkan atau mengambil data tertentu dari tabel **histories** yang **id** nya sesuai dengan **id** yang dipilih oleh user. Setelah itu, Anda perlu menambahkan route baru untuk kedua function yang baru saja Anda tambahkan, dan ingat untuk menambahkannya pada route group **web**.

```
1 Route::get('/user/history/{id}', [UserController::class, 'history'])→name('user.history');
2 Route::get('/user/history/detail/{id}', [UserController::class, 'detail_history'])→name('user.detail.history');
```

Setelah menambahkan route baru tersebut, Anda perlu menambahkan 2 file blade baru di dalam folder **resources/views/pages/user** dengan nama file nya adalah **history.blade.php** dan **detail-history.blade.php** lalu tambahkan kode dibawah ini untuk masing-masing file:

history.blade.php

```
@extends('layouts.user.main')
@section('content')
    <!-- Start Banner Area -->
    <section class="banner-area organic-breadcrumb">
        <div class="container">
            <div class="breadcrumb-banner d-flex flex-wrap align-items-center justify-content-end">
                <div class="col-first">
                    <h1>Halaman History Pembelian</h1>
                    <nav class="d-flex align-items-center">
                        <a href="{{ route('user.dashboard') }}">Home<span class="lnr lnr-arrow-right"></span></a>
                        <a href="#">History Pembelian</a>
                    </nav>
                </div>
            </div>
        </div>
    </section>
    <!-- End Banner Area -->

    <section class="section_gap">
        <!-- single product slide -->
        <div class="container">
            <div class="row">
                <!-- single product -->
                @forelse ($data as $item)
                    <div class="col-lg-3 col-md-6">
                        <div class="single-product">
                            
                            <div class="product-details">
                                <h6>{{ $item->name }}</h6>
                                <div class="prd-bottom">
                                    <a href="{{ route('user.detail.history', $item->id) }}" class="social-info">
                                        <span class="lnr lnr-move"></span>
                                        <p class="hover-text">Detail</p>
                                    </a>
                                </div>
                            </div>
                        </div>
                    </div>
                @empty
            </div>
        </div>
    </section>
endsection
```

```

        <div class="col-lg-12 col-md-12">
            <div class="single-product">
                <h3 class="text-center">Tidak ada produk</h3>
            </div>
        </div>
    @endforelse
</div>
</div>
</section>
<!-- end product Area -->
@endsection

```

detail-history.blade.php

```

@extends('layouts.user.main')
@section('content')
    <!-- Start Banner Area -->
    <section class="banner-area organic-breadcrumb">
        <div class="container">
            <div class="breadcrumb-banner d-flex flex-wrap align-items-center justify-content-end">
                <div class="col-first">
                    <h1>Halaman Detail Pembelian</h1>
                    <nav class="d-flex align-items-center">
                        <a href="{{ route('user.dashboard') }}">Home<span class="lnr lnr-arrow-right"></span></a>
                        <a href="#">Detail Pembelian</a>
                    </nav>
                </div>
            </div>
        </div>
    </section>
    <!-- End Banner Area -->

    <section class="section_gap">
        <!--=====Single Product Area =====-->
        <div class="product_image_area">
            <div class="container">
                <div class="row s_product_inner">
                    <div class="col-lg-6">
                        <div class="single-prd-item">
                            
                        </div>
                    </div>
                    <div class="col-lg-5 offset-lg-1">
                        <div class="s_product_text">
                            <h3>{{ $data->name }}</h3>
                            <h2>{{ $data->price }} Points</h2>
                            <ul class="list">

```

```

<li><a class="active" href="#"><span>Kategori</span> : {{ $data-
>category }}</a>
    </li>
</ul>
<p>{{ $data->description }}</p>
</div>
</div>
</div>
</div>
</div>
<!--=====End Single Product Area =====-->
</section>
@endsection

```

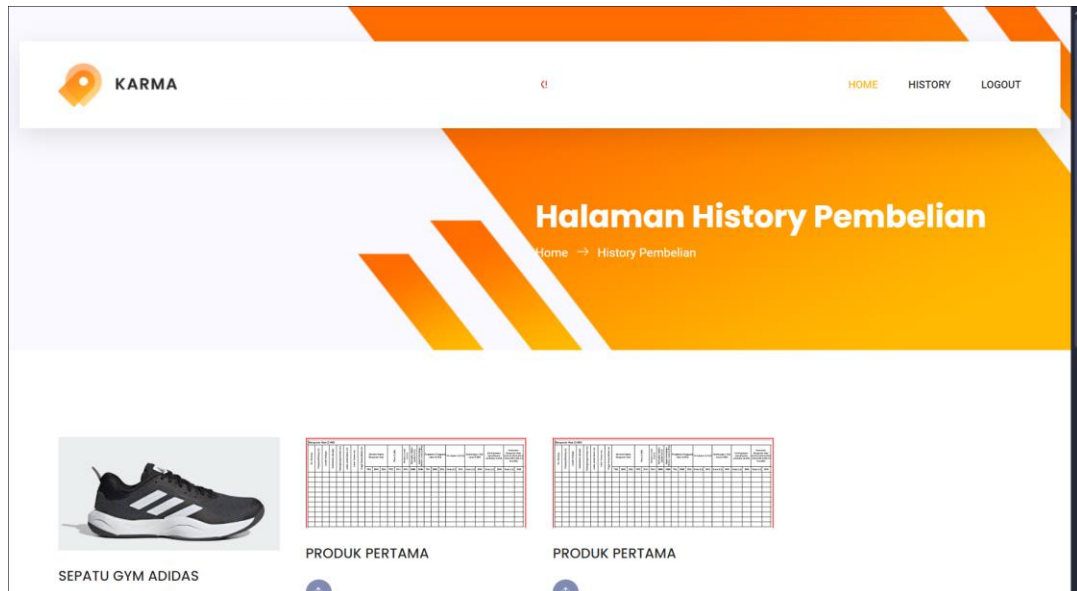
Kemudian, jangan lupa untuk memperbarui navbar pada bagian user yang berada di dalam folder **resources/views/pages/layouts/user/navbar.php** dengan kode dibawah ini:

```

<header class="header_area sticky-header">
<div class="main_menu">
<nav class="navbar navbar-expand-lg navbar-light main_box">
<div class="container">
<a class="navbar-brand logo_h" href="{{ route('user.dashboard') }}"></a>
<button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
<div class="collapse navbar-collapse offset" id="navbarSupportedContent">
<ul class="nav navbar-nav menu_nav ml-auto">
<li class="nav-item active"><a class="nav-link" href="{{
route('user.dashboard') }}">Home</a></li>
</ul>
<ul class="nav navbar-nav navbar-right">
<li class="nav-item"><a class="nav-link"
href="{{ route('user.history', Auth::user()->id) }}">History</a></li>
</ul>
<ul class="nav navbar-nav navbar-right">
<li class="nav-item"><a class="nav-link" href="{{ route('user.logout')
}}">Logout</a></li>
</ul>
</div>
</div>
</nav>
</div>
</header>

```

Jika berhasil, maka berikut ini adalah tampilan dari hasil kode yang sudah Anda tulis diatas:



Selanjutnya, anda akan menambahkan Riwayat atau history pembelian barang di dalam bagian admin. Pertama-tama, buatlah folder baru dengan nama **history** yang berada di dalam **resources/views/pages/admin/** , kemudian tambahkan 2 file blade baru dengan nama **index.blade.php** dan **detail.blade.php** di dalam folder tersebut lalu tulislah kode dibawah ini dalam kedua file tersebut:

index.blade.php:

```
@extends('layouts.admin.main')
@section('title', 'Admin History Pembelian')
@section('content')
    <div class="main-content">
        <section class="section">
            <div class="section-header">
                <h1>History Pembelian</h1>
                <div class="section-header-breadcrumb">
                    <div class="breadcrumb-item active"><a href="{{ route('admin.dashboard') }}">Dashboard</a></div>
                </div>
            </div>
        </section>
    </div>
```



```

        <div class="breadcrumb-item">History Pembelian</div>
    </div>
</div>

<div class="card-body">
    <div class="table-responsive">
        <table class="table table-bordered table-md">
            <tr>
                <th>#</th>
                <th>Nama Pengguna</th>
                <th>Nama Produk</th>
                <th>Total Harga</th>
                <th>Action</th>
            </tr>
            @php
                $no = 0;
            @endphp
            @forelse ($data as $item)
                <tr>
                    <td>{{ $no += 1 }}</td>
                    <td>{{ $item->name }}</td>
                    <td>{{ $item->nama_produk }}</td>
                    <td>{{ $item->total_harga }}</td>
                    <td>
                        <a href="{{ route('history.detail', $item->id) }}"
                            class="badge badge-info">Detail</a>
                    </td>
                </tr>
            @empty
                <td colspan="5" class="text-center">Data Pembelian Kosong</td>
            @endforelse
        </table>
    </div>
</div>

```



```
<label for="user_name">Nama Pengguna</label>
<input id="user_name" type="text" class="form-control"
  value="{{ $data->name }}" readonly>
</div>
</div>
<div class="col-6">
  <div class="form-group">
    <label for="nama_produk">Nama Produk</label>
    <input id="nama_produk" type="text" class="form-control"
      value="{{ $data->nama_produk }}" readonly>
  </div>
</div>
<div class="col-6">
  <div class="form-group">
    <label for="category">Kategori Produk</label>
    <input id="category" type="text" class="form-control"
      value="{{ $data->category }}" readonly>
  </div>
</div>
<div class="col-6">
  <div class="form-group">
    <label for="total_harga">Total Harga Pembelian</label>
    <input id="total_harga" type="text" class="form-control"
      value="{{ $data->total_harga }}" readonly>
  </div>
</div>
</div>
</div>
</section>
</div>
```

@endsection

Setelah itu, tambahkan controller baru dengan nama **HistoryController.php** dengan mengetikkan perintah “**php artisan make:controller Admin/HistoryController**”, kemudian tulislah 2 function dengan nama **index** dan **detail** seperti kode dibawah ini di dalam controller tersebut:

function index:

```
1 public function index()
2 {
3     $data = DB::table('histories')
4         →join('products', 'products.id', '=', 'histories.id_product')
5         →join('users', 'users.id', '=', 'histories.id_user')
6         →select('histories.*','products.*', 'products.name as nama_produk', 'users.*')
7         →get();
8
9     return view('pages.admin.history.index', compact('data'));
10 }
```

function detail:

```
1 public function detail($id)
2 {
3     $data = DB::table('histories')
4         →join('products', 'products.id', '=', 'histories.id_product')
5         →join('users', 'users.id', '=', 'histories.id_user')
6         →select('histories.*','products.*', 'products.name as nama_produk', 'users.*')
7         →where('histories.id', '=', $id)
8         →first();
9
10    return view('pages.admin.history.detail', compact('data'));
11 }
```

Dari kedua function tersebut memiliki fungsi yang sama-sama melakukan inner join, yang berbeda adalah pada function **detail** anda menambahkan kondisi **where**, dimana hanya data dengan id dalam table **histories** tertentu saja yang akan diambil. Selanjutnya, tambahkan kedua route tersebut di dalam route bagian **admin** dalam file **web.php**:

```

1 Route::get('/history', [HistoryController::class, 'index'])->name('admin.history');
2 Route::get('/history/detail/{id}', [HistoryController::class, 'detail'])->name('history.detail');

```

Terakhir, perbarui kode **sidebar.blade.php** yang berada di dalam folder **resources/views/layouts/admin** dengan kode dibawah ini yang diberikan tanda warna kuning:

```

<div class="main-sidebar sidebar-style-2">
  <aside id="sidebar-wrapper">
    <div class="sidebar-brand">
      <a href="#">Teknik Informatika | RPL</a>
    </div>
    <div class="sidebar-brand sidebar-brand-sm">
      <a href="#">RPL</a>
    </div>
    <ul class="sidebar-menu">
      <li class="menu-header">Menu</li>
      <li class="{{ Route::is('admin.dashboard') ? 'active' : '' }}"><a class="nav-link"
        href="{{ route('admin.dashboard') }}"><i class="fas fa-home"></i>
        <span>Dashboard</span></a>
      </li>
      <li class="{{ Request::is('product*') ? 'active' : '' }}"><a class="nav-link"
        href="{{ route('admin.product') }}"><i class="fas fa-box"></i>
        <span>Produk</span></a></li>
      <li
        class="{{ Request::is('distributor*') ? 'active' : '' }}"><a class="nav-link"
        href="{{ route('admin.distributor') }}"><i class="fas fa-user"></i>
        <span>Distributor</span></a></li>
      <li class="{{ Request::is('history*') ? 'active' : '' }}"><a class="nav-link"
        href="{{ route('admin.history') }}"><i class="fas fa-book"></i> <span>Riwayat
        Pembelian</span></a></li>
    </ul>
  </aside>

```

</div>

Jika sudah, maka tampilan halaman history pembelian di bagian admin akan terlihat seperti pada gambar dibawah ini:

