

BAB IV ANALISA DAN PEMBAHASAN

A. Pekerjaan/kegiatan

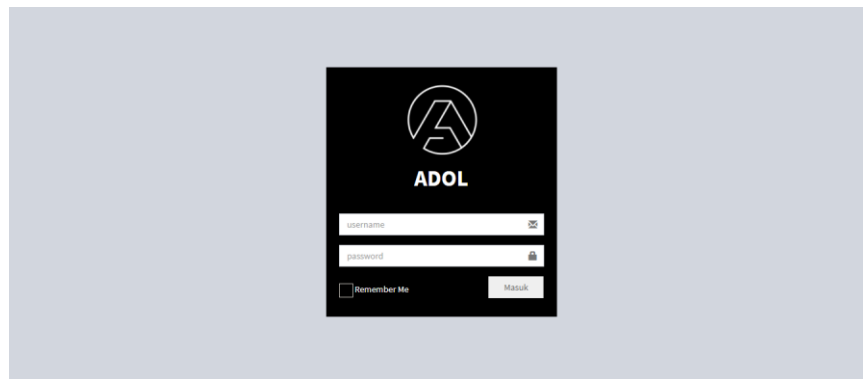
1. Pekerjaan Secara Umum

Pelaksanaan praktik Kerja Lapangan di PT. CAZH Teknologi Inovasi berlokasi di Jl. Waru IV Perumahan Graha Tanjung Elok No.25, Bojong, Tanjung, Kec. Purwokerto Sel., Kabupaten Banyumas, Jawa Tengah 53144.

Penulis melaksanakan Praktik Kerja Lapangan selama 3 bulan lamanya mulai dari tanggal 1 Juli 2020 sampai dengan 1 Oktober 2020. Selama Praktik Kerja Lapangan berlangsung, penulis diberikan tugas oleh pembimbing lapangan untuk membuat sebuah website admin yang bernama ADOL (Ayo Dagang Online). Penulis bertanggung jawab dengan tugas yang diberikan kepada pembimbing lapangan dan wajib melaporkan hasil perkembangan pekerjaan yang dilakukan setiap 1 minggu sekali.

2. Pekerjaan Secara Spesifik

a. Pembuatan fungsi halaman *Log in*



Gambar 4. 1 Halaman Login

Gambar 4.1 Proses pembuatan fungsi *Log in Page* berdasarkan hasil dari pekerjaan *front-end developer* website admin yang membuat halaman login dimana berisi username dan password untuk admin *log in* ke dalam website.

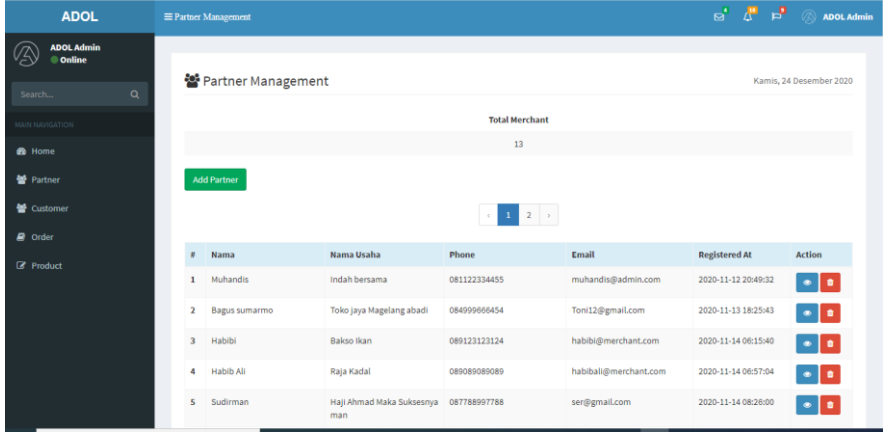
```

public function process(Request $request){
    $validateData = $request->validate([
        'username' => 'required',
        'password' => 'required',
    ]);
    $result = Admin::where('username', '=',
    $validateData['username']->first();
    if($result){
        if (($request->password == $result->password))
        {
            session(['username' => $request->username]);
            return redirect('/index');
        }
        else {
            return back()->withInput()->with('pesan',"Login
Gagal");
        }
    }
    else{
        return back()->withInput()->with('pesan',"Login
Gagal");
    }
}
}

```

b. Pembuatan fungsi halaman *Partner*

- Index Partner



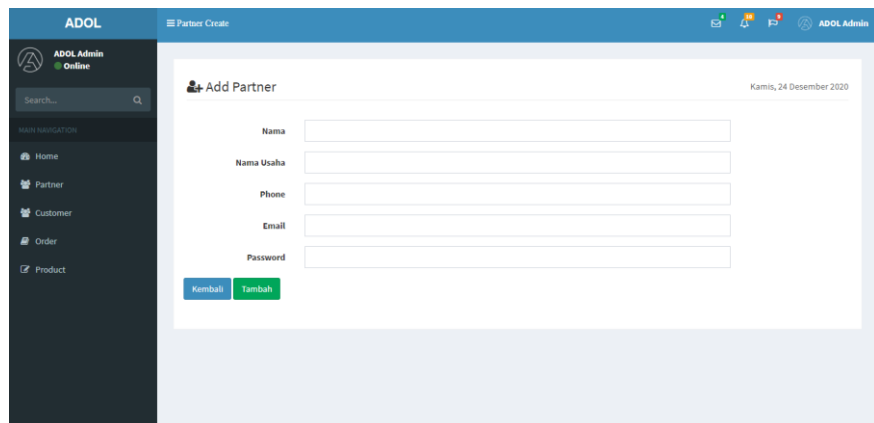
Gambar 4. 2 Halaman Partner

Gambar 4.2 Pada halaman *Partner* membuat fungsi untuk mengambil data pedagang yang sudah menjadi mitra ADOL maupun data dari pedagang yang sudah terdaftar yang terdapat pada database dan kemudian menampilkan data pedagang ke halaman *Partner*. Di

halaman ini admin bisa menambah data partner, mengedit data dan menghapus data.

```
public function index(Request $request)
{
    $data['title'] = "Partner Management";
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    $data['total_merchants'] = User::where('role', '=',
'merchant')->count();
    $data['partners'] = DB::table('users as u')
        ->select('u.id','u.name', 'u.nama_toko',
'u.phone', 'u.email', 'u.created_at', DB::raw('count(t.id) as
trO'),DB::raw('count(m.id) as trM'))
        ->leftJoin('transaksis as t', 'u.id', '=',
't.merchant_id')
        ->leftJoin('manual_transaksis as m', 'u.id', '=',
'm.user_id')
        ->where('u.role', 'merchant')
        ->groupBy('u.id','u.name','u.nama_toko',
'u.phone', 'u.email', 'u.created_at')
        ->paginate(10);
    $data['trO'] = DB::table('users as u')
        ->select('u.id','u.name', 'u.nama_toko', 'u.phone',
'u.email', 'u.created_at', DB::raw('count(t.id) as trO'))
        ->leftJoin('transaksis as t', 'u.id', '=', 't.merchant_id')
        ->where('u.role', 'merchant')
        ->groupBy('u.id','u.name','u.nama_toko', 'u.phone',
'u.email', 'u.created_at')
        ->paginate(10);
}
```

- Add Partner

The screenshot shows a web application interface for ADOL Admin. On the left is a dark sidebar with navigation links: Home, Partner, Customer, Order, and Product. The main content area is titled 'Add Partner' and contains a form with five input fields: 'Nama', 'Nama Usaha', 'Phone', 'Email', and 'Password'. Below the form are two buttons: 'Kembali' (blue) and 'Tambah' (green). The top right of the page shows the date 'Kamis, 24 Desember 2020'.

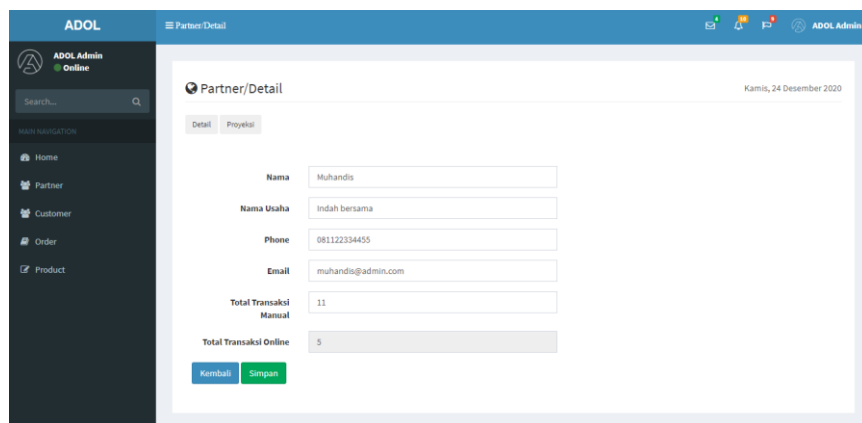
Gambar 4. 3 Halaman Add Partner

Gambar 4.3 Pada halaman Add Partner berfungsi untuk

menambahkan data partner(merchant) dengan mengisi field yang tersedia, data akan divalidasi terlebih dahulu sebelum disimpan ke database.

```
public function store(Request $request)
{
    $validateData = $request->validate([
        'partner_name'      => 'required',
        'company'           => 'required',
        'phone'              => 'required | unique:users',
        'email'              => 'required | email | unique:users',
        'password'           => 'required',
    ]);
    $partner = new User();
    $partner->name = $validateData['partner_name'];
    $partner->nama_toko = $validateData['company'];
    $partner->phone = $validateData['phone'];
    $partner->email = $validateData['email'];
    $partner->role = 'merchant';
    $partner->password = bcrypt($validateData['password']);
    $partner->api_token = Str::random(50);
    $partner->save();
    toastr()->success('Penambahan Data Berhasil');
    return redirect()->route('partner.index');
}
```

- Detail Partner



Gambar 4. 4 Halaman Detail Partner

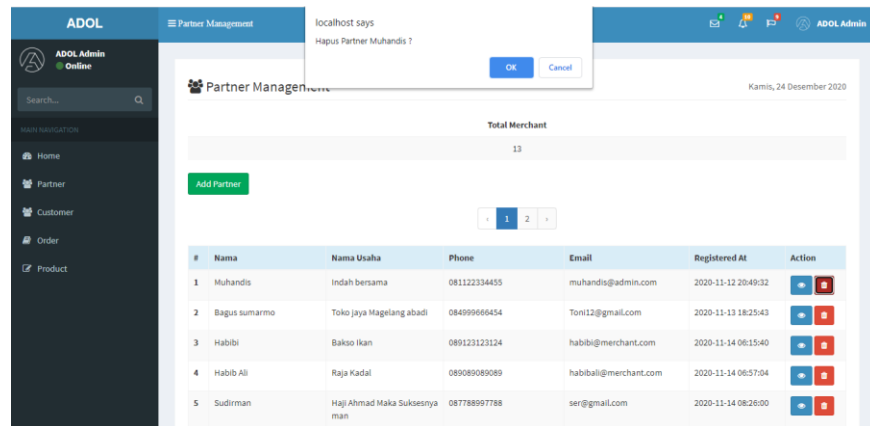
Gambar 4.4 Pada halaman detail partner berfungsi untuk melihat data partner(merchant) dengan lebih detail, selain itu juga dapat berfungsi untuk mengubah data partner(merchant) apabila admin akan mengedit atau mengupdate data partner(merchant).

```

public function detail($id)
{
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    $partner = Partner::findOrFail($id);
    $data['title'] = "Partner/Detail";
    $totalTrxOnline = DB::table('transaksis')-
>where('merchant_id', $id)->count();
    $totalTrxManual = DB::table('manual_transaksis')-
>where('user_id', $id)->count();
    return view('partner.detail', $data, compact('partner',
'totalTrxManual', 'totalTrxOnline'));
}

```

- Hapus Partner



Gambar 4. 5 Konfirmasi Hapus Partner

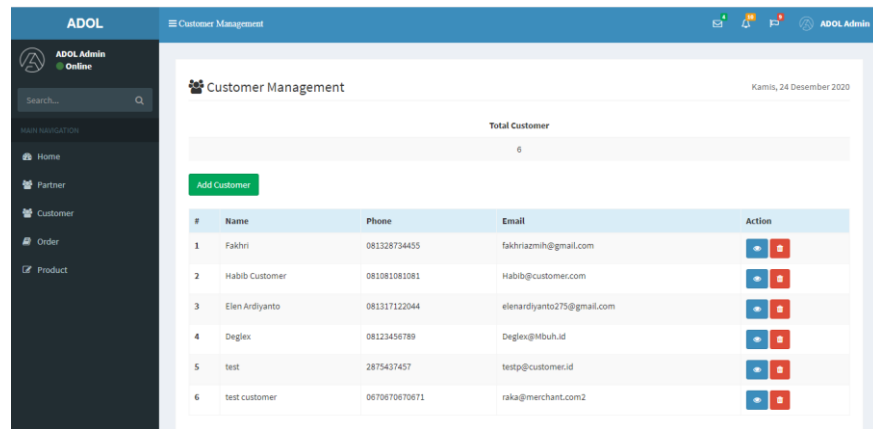
Gambar 4.5 Jika menekan tombol hapus (gambar tempat sampah berwarna merah) pada halaman Partner maka akan menghapus data partner(merchant) pada database, maka dari itu sebelum menghapus akan muncul pop up konfirmasi penghapusan data.

```

public function destroy(Request $request, Partner $partner)
{
    $partner->delete();
    toastr()->error('Hapus Data Berhasil');
    return redirect()->route('partner.index');
}

```

- c. Pembuatan fungsi halaman *Customer*
 - Index Customer



Gambar 4. 6 Halaman Customer

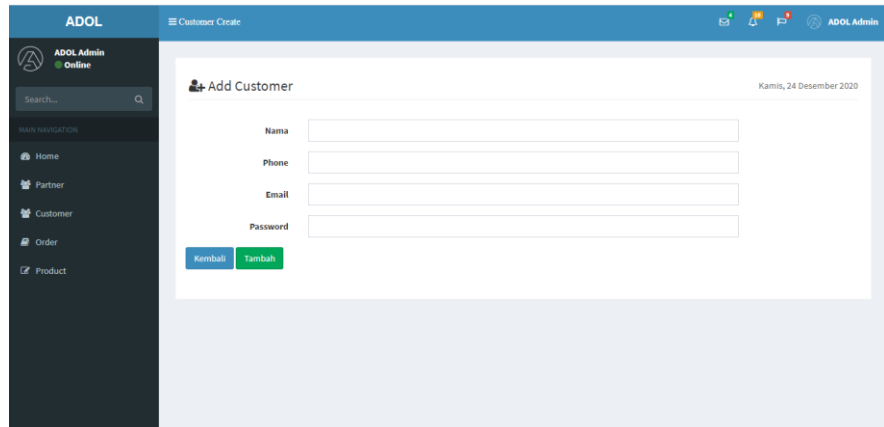
Gambar 4.6 Pada halaman *Customer* tidak jauh beda dengan halaman *Partner*, pada halaman ini membuat fungsi untuk mengambil data dari database dan kemudian menampilkan data pelanggan atau *customer* yang sudah mendaftar dan menggunakan aplikasi ADOL ke halaman *customer*. Di halaman ini admin bisa menambah data customer, mengedit data dan menghapus data.

```

public function index()
{
    $data['title'] = "Customer Management";
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    $data['customers'] = User::where('role', '!=', 'customer')-
>paginate(10);
    $data['total_customers'] = User::where('role', '!=',
'customer')->count();
    return view('customer.index',$data);
}

```

- Add Customer

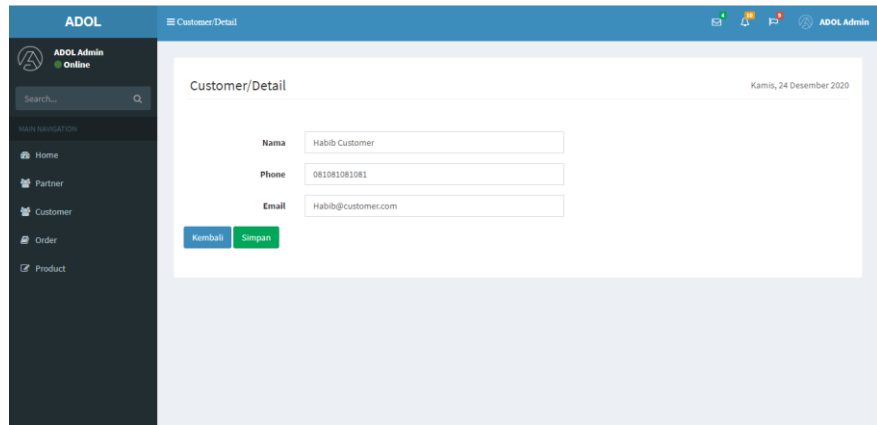


Gambar 4. 7 Halaman Add Customer

Gambar 4.7 Pada halaman Add Customer berfungsi untuk menambahkan data customer(pelanggan) dengan mengisi field yang tersedia, data akan divalidasi terlebih dahulu sebelum disimpan ke database.

```
public function store(Request $request)
{
    $validateData = $request->validate([
        'customer_name' => 'required',
        'phone'          => 'required | unique:users',
        'email'          => 'required | email | unique:users',
        'password'       => 'required',
    ]);
    $customer = new User();
    $customer->name = $validateData['customer_name'];
    $customer->phone = $validateData['phone'];
    $customer->email = $validateData['email'];
    $customer->role = 'customer';
    $customer->password =
    bcrypt($validateData['password']);
    $customer->api_token = Str::random(50);
    $customer->save();
    toastr()->success('Penambahan Data Berhasil');
    return redirect()->route('customer.index');
}
```

- Detail Customer

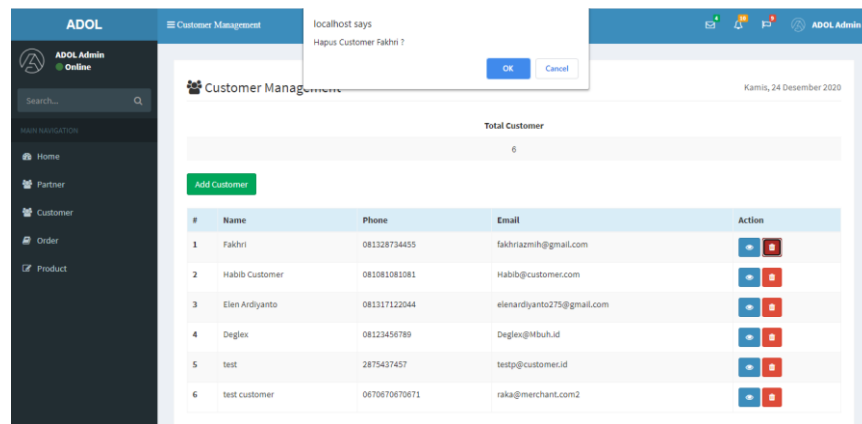


Gambar 4. 8 Halaman Detail Customer

Gambar 4.8 Pada halaman Detail Customer berfungsi untuk melihat data customer(pelanggan) dengan lebih detail, selain itu juga dapat berfungsi untuk mengubah data customer(pelanggan) apabila admin akan mengedit atau mengupdate data customer(pelanggan).

```
public function update(Request $request, Customer
$customer)
{
    $validateData = $request->validate([
        'name' => 'required',
        'phone' => 'required',
        'email' => 'required',
    ]);
    $customer->name = $validateData['name'];
    $customer->phone = $validateData['phone'];
    $customer->email = $validateData['email'];
    $customer->save();
    toastr()->success('Perubahan data berhasil');
    return redirect()->route('customer.index');
}
```


- Hapus Customer



Gambar 4. 9 Konfirmasi Hapus Customer

Gambar 4.9 Jika menekan tombol hapus (gambar tempat sampah berwarna merah) pada halaman Customer maka akan menghapus data Customer pada database, maka dari itu sebelum menghapus akan muncul pop up konfirmasi penghapusan data.

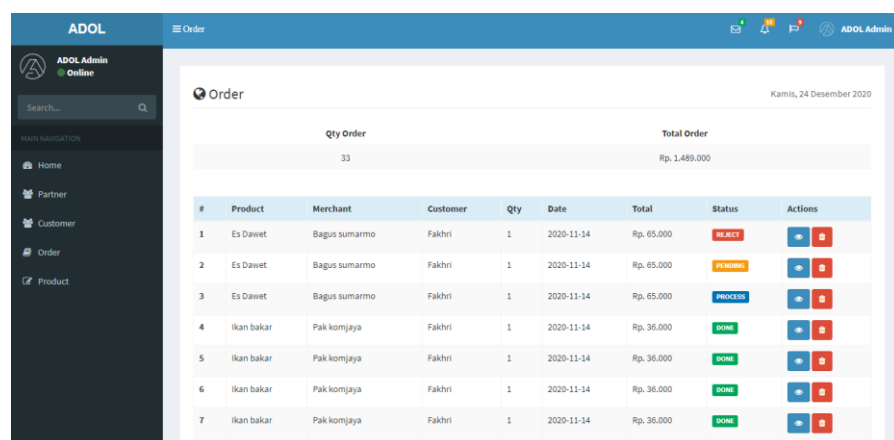
```

public function destroy(Request $request, Customer $customer)
{
    $customer->delete();
    toastr->error('Hapus Data Berhasil');
    return redirect()->route('customer.index');
}

```

d. Pembuatan fungsi halaman *Order*

- Index Order

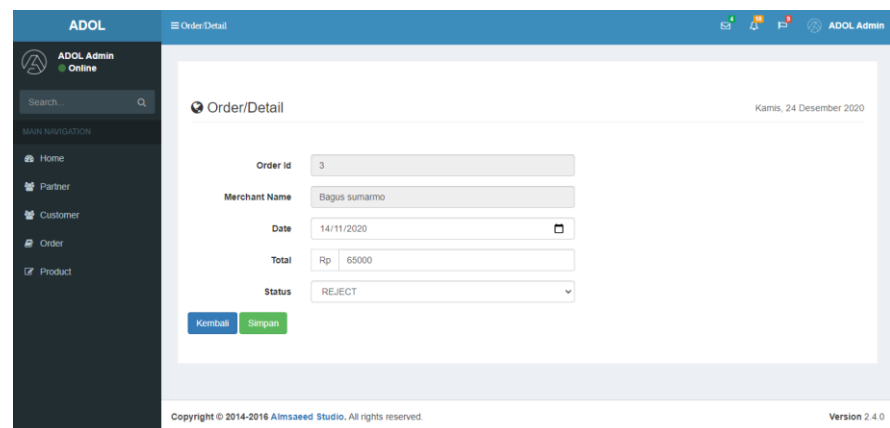


Gambar 4. 10 Halaman Order

Gambar 4. 10 Pada halaman *Order* membuat fungsi untuk mengambil data transaksi dari database dan kemudian menampilkan data transaksi antara pedagang dan pelanggan yang dilakukan pada aplikasi ADOL ke halaman *Order*. Di halaman ini admin bisa mengedit dan menghapus data Order.

```
public function index()
{
    $orders = Order::paginate(10);
    $data['title'] = "Order";
    $data['qty_order'] = Order::count('kuantiti');
    $data['total_order'] = Order::sum('harga_total');
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    return view('order.index',$data, compact('orders'));
}
```

- Detail Order

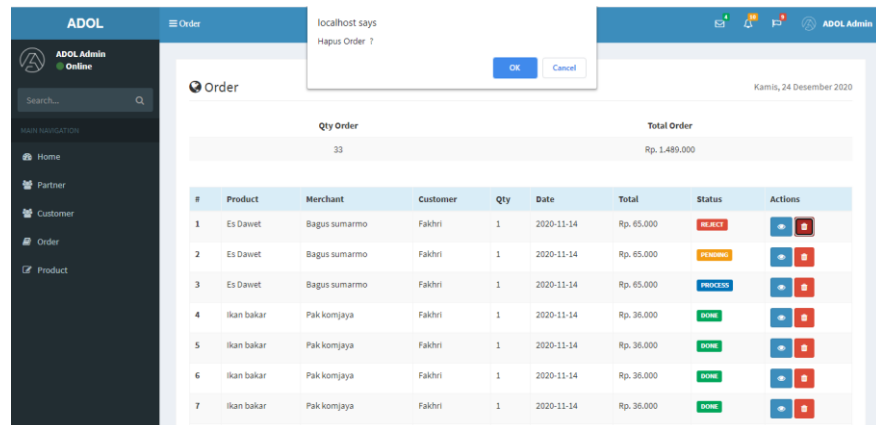


Gambar 4. 11 Halaman Detail Order

Gambar 4.11 Pada halaman Detail Order berfungsi untuk melihat data order(transaksi) dengan lebih detail, selain itu juga dapat berfungsi untuk mengubah data order(transaksi) apabila admin akan mengedit atau mengupdate data order(transaksi).

```
public function detail($id)
{
    $order = Order::findOrFail($id);
    $data['title'] = "Order/Detail";
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    return view('order.detail',$data, compact('order'));
}
```

- Hapus Order



Gambar 4. 12 Konfirmasi Hapus Order

Gambar 4.12 Jika menekan tombol hapus (gambar tempat sampah berwarna merah) pada halaman Order maka akan menghapus data Order pada database, maka dari itu sebelum menghapus akan muncul pop up konfirmasi penghapusan data.

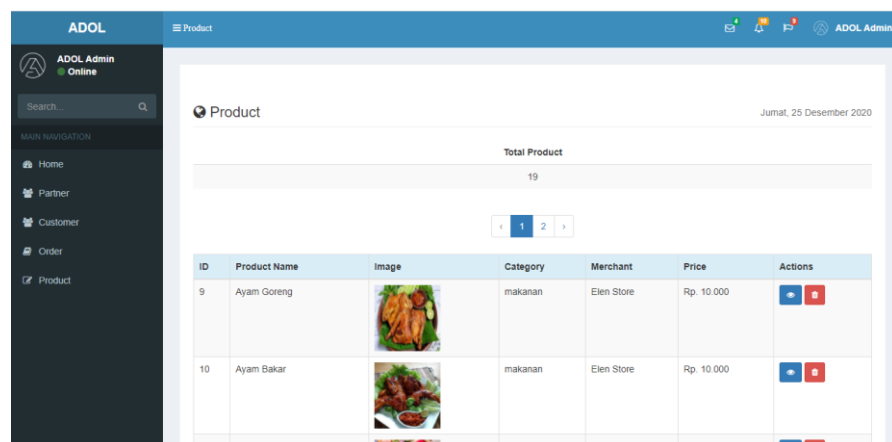
```

public function destroy(Request $request, $id)
{
    // File::delete($brg->gambar);
    Order::find($id)->delete();
    toastr()->error('Hapus Data Berhasil');
    return redirect()->route('order.index');
}

```

e. Pembuatan fungsi halaman *Product*

- Index Product

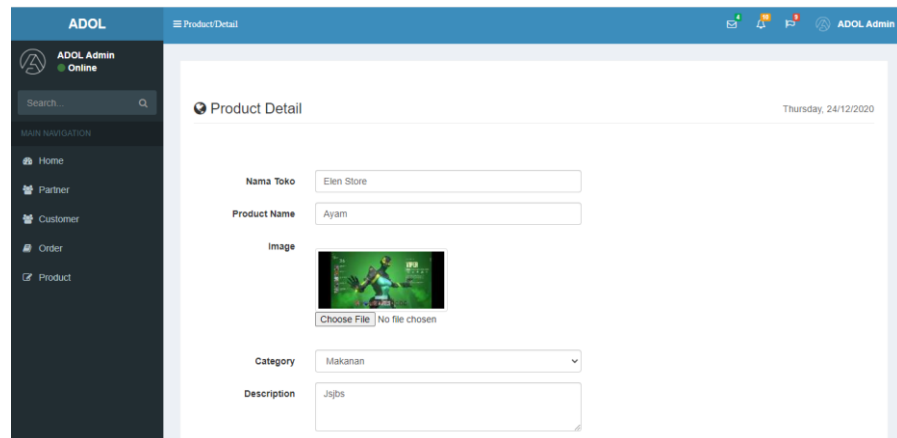


Gambar 4. 13 Halaman Product

Gambar 4.13 Pada halaman *Product* membuat fungsi untuk mengambil data produk dari database dan kemudian menampilkan daftar product dari pedagang yang sudah ditambahkan pada aplikasi ADOL ke halaman *Product*. Di halaman ini admin bisa mengedit dan menghapus data Product.

```
public function index(Request $request)
{
    $data['title'] = "Product";
    $search_name = $request->search_name;
    $search_category = $request->search_category;
    $search_merchant = $request->search_merchant;
    $data['list_data'] = Product::paginate(10);
    $data['jumlah'] = Product::count();
    $data['hari'] = Carbon::now()->isoFormat('dddd, DD
MMMM Y');
    return view('product.index', $data );
}
```

- Detail Product

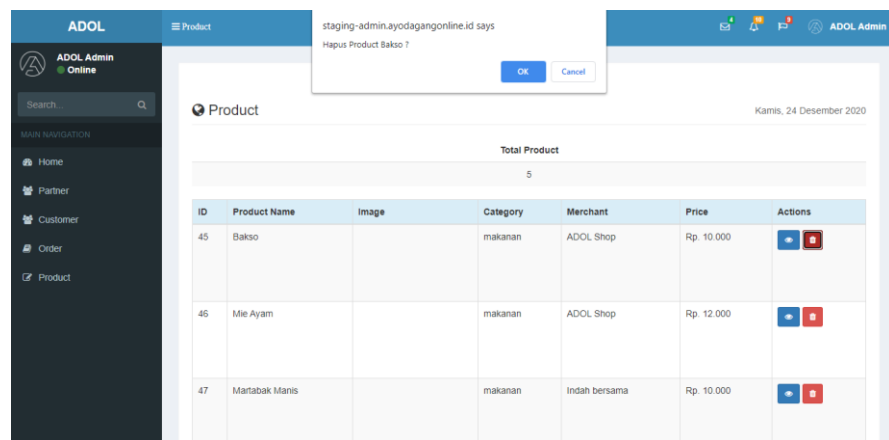


Gambar 4. 14 Halaman Detail Product

Gambar 4.14 Pada halaman Detail Product berfungsi untuk melihat data product dengan lebih detail, selain itu juga dapat berfungsi untuk mengubah data product apabila admin akan mengedit atau mengupdate data product.

```
public function detail($id)
{
    $product = Product::findOrFail($id);
    $data['title']= "Product/Detail";
    return view('product.detail',$data,compact('product'));
}
```

- Hapus Product



Gambar 4. 15 Konfirmasi Hapus Product

Gambar 4.15 Jika menekan tombol hapus (gambar tempat sampah berwarna merah) pada halaman Product maka akan menghapus data product pada database, maka dari itu sebelum menghapus akan muncul pop up konfirmasi penghapusan data.

```
public function destroy(Request $request, Product $product)
{
    File::delete($product->image);
    $product->delete();
    toastr()->error('Hapus Data Berhasil');
    return redirect()->route('product.index');
}
```

B. Analisa Pembahasan Hasil Kerja

Pada pembuatan fungsi untuk website ini dilakukan sesuai dengan hasil yang telah dibuat oleh *front-end developer* website admin, penulis menggunakan framework Laravel untuk membuat fungsi agar website admin dapat berjalan dengan fungsi sebagaimana mestinya.

Pembuatan fungsi halaman *Log in* agar admin dapat masuk kedalam website admin untuk memantau dan mengelola data yang terdapat pada aplikasi ADOL (Ayo Dagang Online).

Pembuatan fungsi halaman *Partner* yang didalamnya berisi data pedagang yang telah terdaftar sebagai mitra ADOL. Pada halaman ini terdapat fitur *detail* dan *add partner*. Fitur *detail* memiliki fungsi untuk melihat rincian data

partner lebih detail, fitur *add partner* masih belum bisa digunakan karena masih ada beberapa perubahan.

Halaman pelanggan atau *customer* memiliki tampilan dan fungsi halaman yang tidak jauh berbeda dengan halaman *partner*. Yang membedakan hanyalah isinya saja yang berisikan data *customer* dari aplikasi ADOL. Fitur pada halaman ini juga sama, terdapat fitur *detail* dan fitur *add Customer* yang masih belum bisa digunakan.

Selanjutnya pada halaman *Order*, berisikan list transaksi antara pelanggan dan pedagang pada aplikasi ADOL. Fitur pada halaman ini ada *detail*, dan *delete*. Semua fitur pada halaman ini sudah dibuat dan disesuaikan dengan fungsi yang dibutuhkan aplikasi ADOL.

Halaman *Product*, berisikan list *product* dari pedagang yang sudah didaftarkan pada aplikasi ADOL. Terdapat beberapa fitur pada halaman ini, yaitu : *detail*, *delete*. Fitur pada halaman ini sudah dibuat semua dan sesuai dengan fungsi yang dibutuhkan aplikasi ADOL.

Dari *planning* awal yang sudah dibuat, masih ada beberapa fitur yang belum sempat terbuat karena terdapat beberapa perubahan *planning* dan waktu yang tidak cukup menyadari kurangnya ilmu dan kemampuan penulis. Tapi penulis sudah berusaha mengerjakannya sesuai dan secepat mungkin.

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan Analisa dan pengamatan yang telah dilakukan selama melaksanakan praktek kerja lapangan PKL di PT. CAZH Teknologi Inovasi, maka dapat ditarik beberapa kesimpulan seperti berikut :

1. Pembuatan *Website Admin* disesuaikan dengan *design* dari team UI/UX yang terdapat pada figma dan fungsinya disesuaikan dengan kebutuhan aplikasi adol.
2. Pembuatan *Website Admin* masih belum maksimal dilakukan, masih banyak halaman dan fitur yang belum dibuat oleh penulis. Penulis sudah membuat beberapa fitur diantaranya membuat fungsi login, fungsi manajemen customer, partner, order, dan product yaitu CRUD (Create, Read, Update, Delete) untuk masing-masing menu.

B. Saran

Berdasarkan hasil dan kesimpulan, maka masih dapat dilakukan pengembangan lebih lanjut antara lain:

1. Lengkapi fitur pada *website* tersebut agar sesuai dengan kebutuhan aplikasi adol.
2. Memperbaiki *website* tersebut, agar lebih responsive dan nyaman digunakan serta menjalankan semua fungsi dengan lancar.
3. Memberbaiki fitur yang belum bisa digunakan sebagaimana mestinya.
4. Terus mengembangkan project starup ADOL (Ayo Dagang Online) dengan menambah fitur lain.