

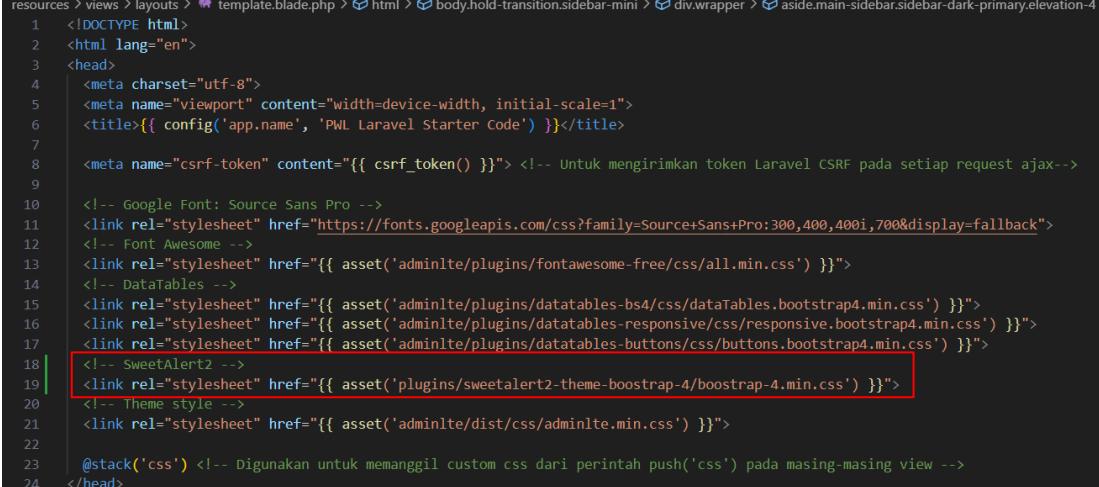


Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Sistem Informasi Bisnis
Semester : 5

Kelas : SIB-3D
NIM : 2241760007
Nama : Anaradi Octa Lavechia
Jobsheet Ke- : 6 (enam)

Laporan Jobsheet 06 - Ajax Form (AdminLTE) dan Client Validation

Praktikum Ke-1. Modal Ajax Tambah Data (Data User)

Langkah	Jawaban/Deskripsi
1	Kita buat form tambah data baru dengan menerapkan modal dan proses ajax.
2	Pertama yang kita siapkan adalah menambahkan library jQuery Validation dan Sweetalert ke aplikasi web kita. Caranya kita tambahkan link kedua library tersebut ke template.blade.php , library sudah disediakan oleh adminLTE.  



3	<p>Selanjutnya Kita modifikasi view <code>user/index.blade.php</code>, kita tambahkan tombol untuk membuat form popup ajax</p> <pre>resources > views > user > 🗂 index.blade.php > ... 1 @extends('layouts.template') 2 @section('content') 3 <div class="card card-outline card-primary"> 4 <div class="card-header"> 5 <h3 class="card-title">{{ \$page->title }}</h3> 6 <div class="card-tools"> 7 Tambah 8 <button onclick="modalAction('{{ url('user/create_ajax') }}')" class="btn btn-sm btn-success mt-1">Tambah Ajax</button> 9 </div> 10 <div class="card-body"></pre> <p>Kita tambahkan kode berikut, untuk membuat form modal tambah data user dengan ajax</p>
4	<p>Selanjutnya kita tambahkan kode berikut pada akhir <code>@section('content')</code> pada view <code>user/index.blade.php</code></p> <pre>resources > views > user > 🗂 index.blade.php > 🗂 div.card.card-outline.card-primary 3 <div class="card card-outline card-primary"> 4 <div class="card-header"> 46 </div> 47 <div id="myModal" class="modal fade animate shake" tabindex="-1" role="dialog" data- 48 backdrop="static" data-keyboard="false" data-width="75%" aria-hidden="true"></div> 49 @endsection</pre>
5	<p>Kemudian kita tambahkan kode berikut pada awal <code>@push('js')</code> pada view <code>user/index.blade.php</code>. Sehingga tampilan kode program akan seperti berikut</p> <pre>resources > views > user > 🗂 index.blade.php > 🗂 div.card.card-outline.card-primary > 🗂 script > 🗂 <function> 3 <div class="card card-outline card-primary"> 54 @push('js') 55 <script> 56 function modalAction(url = '') { 57 \$('#myModal').load(url, function() { 58 \$('#myModal').modal('show'); 59 }); 60 } 61 62 var dataUser; 63 \$(document).ready(function() { 64 dataUser = \$('#table_user').DataTable({ 65 // serverSide: true, jika ingin menggunakan server side processing 66 serverSide: true, 67 ajax: {</pre>
6	<p>Selanjutkan Kita modifikasi <code>route/web.php</code> untuk mengakomodir operasi ajax</p> <pre>routes > 🗂 web.php > 🗂 Closure 14 //TABEL USER 15 Route::group(attributes: ['prefix' => 'user'], routes: function () { 16 Route::get(uri: '/', action: [UserController::class, 'index']); // menampilkan halaman awal user 17 Route::post(uri: '/list', action: [UserController::class, 'list']); // menampilkan data user dalam bentuk json untuk datatables 18 Route::get(uri: '/create', action: [UserController::class, 'create']); // menampilkan halaman form tambah user 19 Route::post(uri: '/', action: [UserController::class, 'store']); // menyimpan data user baru 20 Route::get(uri: '/create_ajax', action: [UserController::class, 'create_ajax']); // menampilkan halaman form tambah user Ajax 21 Route::post(uri: '/ajax', action: [UserController::class, 'store_ajax']); // menyimpan data user baru Ajax 22 Route::get(uri: '{id}', action: [UserController::class, 'show']); // menampilkan detail user 23 Route::get(uri: '{id}/edit', action: [UserController::class, 'edit']); // menampilkan halaman form edit user 24 Route::put(uri: '{id}', action: [UserController::class, 'update']); // menyimpan perubahan data user 25 Route::delete(uri: '{id}', action: [UserController::class, 'destroy']); // menghapus data user 26 });</pre>



7

Kemudian Kita tambahkan fungsi `create_ajax()` pada file `UserController.php`

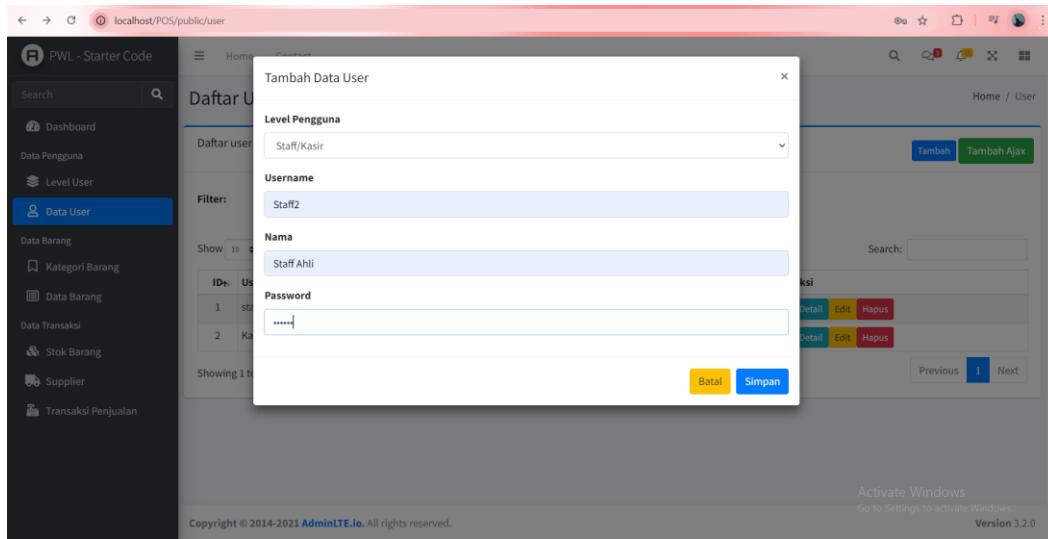
```
app > Http > Controllers > UserController.php > PHP > UserController
12 class UserController extends Controller
169 // Menampilkan halaman form tambah user Ajax|
170 | 1 reference | 0 overrides
171 | public function create_ajax(): mixed|View{
172 |     $level = levelmodel::select(columns: 'level_id','level_nama')->get();
173 |
174 |     return view(view: 'user.create_ajax')->with(key: 'level',value: $level);
175 |
176 ?>
```

8

Setelah itu, kita buat **view baru** `user/create_ajax.blade.php` untuk menampilkan form dengan ajax

```
resources > views > user > create_ajax.blade.php > script > <function> > submitHandler > success
1 <form action="{{ url('user/ajax') }}" method="POST" id="Form-Tambah">
2     @csrf
3     <div id="modal-tambah" class="modal-dialog modal-lg" role="document">
4         <div class="modal-header">
5             <h6 class="modal-title" id="exampleModalLabel">Tambah Data User</h6>
6             <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&amptimes</span></button>
7         </div>
8         <div class="modal-body">
9             <label>Level Pengguna</label>
10            <select name="level_id" id="level_id" class="form-control" required>
11                <option value=""> Pilih Level </option>
12                @foreach($level as $l)
13                    <option value="{{ $l->level_id }}>{{ $l->level_nama }}</option>
14                @endforeach
15            </select>
16            <small id="error-level_id" class="error-text form-text text-danger"></small>
17        </div>
18        <div class="form-group">
19            <label>Username</label>
20            <input value="" type="text" name="username" id="username" class="form-control" required>
21            <small id="error-username" class="error-text form-text text-danger"></small>
22        </div>
23        <div class="form-group">
24            <label>Nama</label>
25            <input value="" type="text" name="name" id="name" class="form-control" required>
26            <small id="error-name" class="error-text form-text text-danger"></small>
27        </div>
28        <div class="form-group">
29            <label>Password</label>
30            <input value="" type="password" name="password" id="password" class="form-control" required>
31            <small id="error-password" class="error-text form-text text-danger"></small>
32        </div>
33    </div>
34    <div class="modal-footer">
35        <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
36        <button type="submit" class="btn btn-primary">Simpan</button>
37    </div>
38 </div>
39 </form>
40 <script>
41 $(document).ready(function() {
42     $("#Form-Tambah").validate({
43         rules: {
44             level_id: {required: true, number: true},
45             username: {required: true, minlength: 3, maxlength: 20},
46             name: {required: true, minlength: 3, maxlength: 10},
47             password: {required: true, minlength: 6, maxlength: 20}
48         },
49         submitHandler: function(form) {
50             $.ajax({
51                 url: form.action,
52                 type: form.method,
53                 data: $(form).serialize(),
54                 success: function(response) {
55                     if(response.status){
56                         $('#exampleModal').modal('hide');
57                         Swal.fire({
58                             icon: 'success',
59                             title: 'Berhasil',
60                             text: response.message
61                         });
62                         dataUser.ajax.reload();
63                     }else{
64                         $('#error-text').text('');
65                         $.each(response.mgField, function(prefix, val) {
66                             $('#'+prefix+'-prefix').text(val[0]);
67                         });
68                         Swal.fire({
69                             icon: 'error',
70                             title: 'Terjadi Kecelahan',
71                             text: response.message
72                         });
73                     }
74                 }
75             });
76         },
77         errorElement: 'span',
78         errorPlacement: function(error, element) {
79             error.addClass('invalid-feedback');
80             element.closest('.form-group').append(error);
81         },
82         highlight: function(element, errorClass, validClass) {
83             $(element).addClass('is-invalid');
84         },
85         unhighlight: function(element, errorClass, validClass) {
86             $(element).removeClass('is-invalid');
87         }
88     });
89 }, 1000);
90 </script>
```



9	<p>Kemudian untuk mengakomodir proses simpan data melalui ajax, kita buat fungsi <code>store_ajax()</code> pada UserController.php</p> <pre>app > Http > Controllers > UserController.php > ... 13 class UserController extends Controller 177 // Ajax menyimpan data user baru 178 public function store_ajax(Request \$request): JsonResponse mixed{ 179 if (\$request->ajax() \$request->wantsJson()){ 180 \$rules = [181 'level_id'=>'required integer', 182 'username'=>'required string min:3 unique:m_user,username', 183 'nama' =>'required string max:100', 184 'password'=>'required min:6' 185]; 186 \$validator = Validator::make(data: \$request->all(), rules: \$rules); 187 if (\$validator->fails()){ 188 return response()->json(data: [189 'status'=>false, 190 'message'=>'Validasi Gagal', 191 'msgField' => \$validator->errors(), 192]); 193 } 194 userModel::create(attributes: \$request->all()); 195 return response()->json(data: [196 'status'=>true, 197 'message'=> 'Data user berhasil disimpan' 198]); 199 } 200 } 201 }</pre>
10	<p>OK, sekarang kita coba melakukan proses tambah data user menggunakan form ajax. Amati apa yang terjadi dan laporankan pada laporan <i>jobsheet</i> dan <i>commit</i> di github kalian!!!</p> 



The screenshot shows a web application interface titled "PWL - Starter Code". On the left is a sidebar with navigation links: Dashboard, Data Pengguna, Level User, Data User (which is selected and highlighted in blue), Data Barang, Kategori Barang, Data Barang, Data Transaksi, Stok Barang, Supplier, and Transaksi Penjualan. The main content area is titled "Daftar User" and displays a table of users with columns: ID, Username, and Aksi (Detail, Edit, Hapus). A modal window is open in the center, showing a green checkmark icon and the text "Berhasil" followed by "Data user berhasil disimpan". At the bottom right of the modal is a blue "OK" button. The footer of the page includes copyright information: "Copyright © 2014-2021 AdminLTE.io. All rights reserved.", a link to "Activate Windows", and the text "Version 3.2.0".

Praktikum Ke-2. Modal Ajax Edit Data (Data User)

Langkah	Jawaban/Deskripsi
1	Pada Praktikum 2 ini, kita akan melakukan coding untuk proses edit menggunakan ajax.
2	Pertama-tama, kita ubah dulu fungsi <code>list()</code> pada <code>UserController.php</code> untuk mengganti tombol edit untuk bisa menggunakan modal
	<pre>app > Http > Controllers > UserController.php > PHP > UserController > list() 13 class UserController extends Controller 14 { 15 public function list(Request \$request): JsonResponse 16 { 17 \$users = UserModel::select('user_id', 'username', 'nama', 'level_id') 18 ->with('relations:level'); 19 20 // filter data user berdasarkan level_id 21 if (\$request->level_id) { 22 \$users->where(column: 'level_id', operator: \$request->level_id); 23 } 24 25 return Datatables::of(\$users) 26 // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex) 27 ->addIndexColumn() 28 ->addColumn('aksi', function (\$user): string { 29 // menambahkan kolom aksi 30 \$btn = 'user_id) . '" class="btn btn-info btn-sm">Detail '; 31 \$btn .= 'user_id) . '/edit" . " class="btn btn-warning btn-sm">Edit '; 32 \$btn .= '<form class="d-inline-block" method="POST" action="'.url('/user/' . \$user->user_id) . '>'; 33 \$btn .= csrf_field() . method_field('DELETE') . '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\\'Apakah Anda yakin menghapus data ini?\')";>Hapus</button></form>'; 34 \$btn = '<button onclick="modalAction(\''.url(path: "/user/" . \$user->user_id) . '/show_ajax').'\')" class="btn btn-info btn-sm">Detail</button> '; 35 \$btn .= '<button onclick="modalAction(\''.url(path: "/user/" . \$user->user_id) . '/edit_ajax').'\')" class="btn btn-warning btn-sm">Edit</button> '; 36 \$btn .= '<button onclick="modalAction(\''.url(path: "/user/" . \$user->user_id) . '/delete_ajax').'\')" class="btn btn-danger btn-sm">Hapus</button> '; 37 }); 38 39 return \$btn; 40 } 41 42 return Datatables::of(\$users) 43 ->drawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html 44 ->make(true); 45 } 46 }</pre>

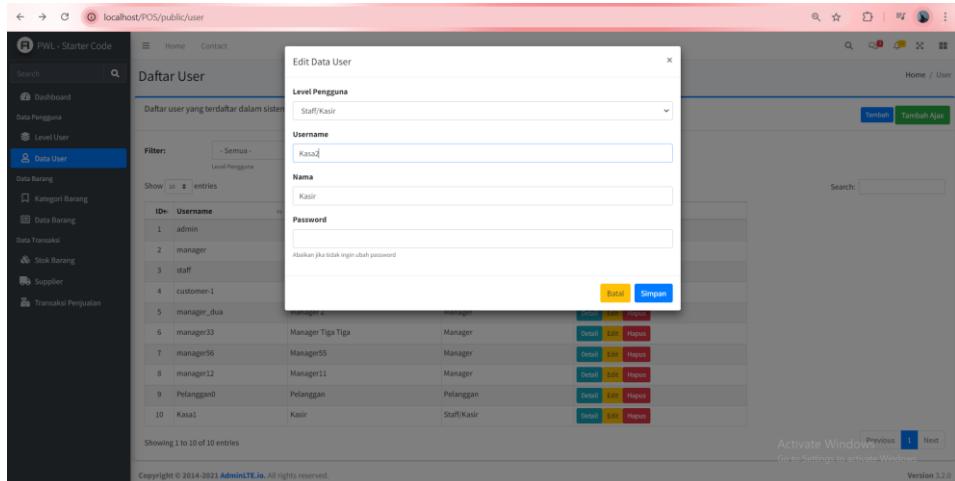




- 6 Selanjutnya, kita buat fungsi `update_ajax()` pada `UserController.php` untuk mengakomodir request update data user melalui ajax

```
app > Http > Controllers > UserController.php > PHP > UserController
13 class UserController extends Controller
216 // Ajax menyimpan update
217 1 reference | 0 overrides
218 public function update_ajax(Request $request, $id): JsonResponse|mixed|Redirector|RedirectRes...
219     // cek apakah request dari ajax
220     if ($request->ajax() || $request->wantsJson()) {
221         $rules = [
222             'level_id' => 'required|integer',
223             'username' => 'required|max:20|unique:m_user,username,' . $id . ',user_id',
224             'nama' => 'required|max:100',
225             'password' => 'nullable|min:6|max:20'
226         ];
227
228         // use Illuminate\Support\Facades\Validator;
229         $validator = Validator::make(data: $request->all(), rules: $rules);
230
231         if ($validator->fails()) {
232             return response()->json(data: [
233                 'status' => false, // respon json, true: berhasil, false: gagal
234                 'message' => 'Validasi gagal.',
235                 'msgField' => $validator->errors() // menunjukkan field mana yang error
236             ]);
237
238         $check = UserModel::find(id: $id);
239         if ($check) {
240             if (!$request->filled(key: 'password')) { // jika password tidak diisi, maka hapus dari request
241                 $request->request->remove(key: 'password');
242             }
243             $check->update(attributes: $request->all());
244
245             return response()->json(data: [
246                 'status' => true,
247                 'message' => 'Data berhasil diupdate'
248             ]);
249         } else {
250             return response()->json(data: [
251                 'status' => false,
252                 'message' => 'Data tidak ditemukan'
253             ]);
254         }
255
256     }
257
258     return redirect(to: '/');
259 }
```

- 7 Sekarang kita coba bagian edit user, amati proses nya. Jangan lupa laporkan dan *commit* ke repository git kalian !





The screenshot shows a web application interface for managing users. On the left is a sidebar with links like Dashboard, Data Pengguna, Level User, Data User (which is selected), Data Barang, Kategori Barang, Data Barang, Data Transaksi, Stok Barang, Supplier, and Transaksi Penjualan. The main content area has a header "Daftar User" and a table showing two users: staff (Nama: Staff/Kasir) and Kasa2 (Nama: Kasir). A modal dialog is centered over the table, displaying a green checkmark icon, the text "Berhasil", and the message "Data berhasil dihapus". At the bottom of the modal is a blue "OK" button. The background of the main page is dimmed.

Praktikum Ke-3. Modal Ajax Hapus Data (Data User)

Langkah	Jawaban/Deskripsi
1	Pada Praktikum 3 ini, kita akan melakukan koding untuk proses hapus menggunakan ajax.
2	Pertama-tama, kita ubah dulu <code>routes/web.php</code> untuk mengakomodir request halamankonfirmasi untuk menghapus data
3	Kemudian kita buat fungsi <code>confirm_ajax()</code> pada <code>UserController.php</code>

```
routes > web.php > Closure
14 //TABEL USER
15 Route::group(attributes: ['prefix' => 'user'], routes: function () {
16     Route::get(uri: '/', action: [UserController::class, 'index']); // menampilkan halaman awal user
17     Route::post(uri: '/list', action: [UserController::class, 'list']); // menampilkan data user dalam bentuk json untuk datatables
18     Route::get(uri: '/create', action: [UserController::class, 'create']); // menampilkan halaman form tambah user
19     Route::post(uri: '/', action: [UserController::class, 'store']); // menyimpan data user baru
20     Route::get(uri: '/create_ajax', action: [UserController::class, 'create_ajax']); // menampilkan halaman form tambah user Ajax
21     Route::post(uri: '/ajax', action: [UserController::class, 'store_ajax']); // menyimpan data user baru Ajax
22     Route::get(uri: '/{id}', action: [UserController::class, 'show']); // menampilkan detail user
23     Route::get(uri: '/{id}/edit', action: [UserController::class, 'edit']); // menampilkan halaman form edit user
24     Route::put(uri: '/{id}', action: [UserController::class, 'update']); // menyimpan perubahan data user
25     Route::get(uri: '/{id}/edit_ajax', action: [UserController::class, 'edit_ajax']); // menampilkan halaman form edit user Ajax
26     Route::put(uri: '/{id}/update_ajax', action: [UserController::class, 'update_ajax']); // menyimpan perubahan data user
27     Route::get(uri: '/{id}/delete_ajax', action: [UserController::class, 'confirm_ajax']); // untuk tampilan form confirm delete user ajax
28     Route::delete(uri: '/{id}/delete_ajax', action: [UserController::class, 'delete_ajax']); // untuk hapus data user ajax
29     Route::delete(uri: '/{id}', action: [UserController::class, 'destroy']); // menghapus data user
30 });

```



	<pre>app > Http > Controllers > UserController.php > PHP > UserController 13 class UserController extends Controller 260 // Ajax hapus data 261 1 reference 0 overrides 262 public function confirm_ajax(String \$id): Factory View{ 263 \$user = UserModel::find(id: \$id); 264 }</pre>
4	Selanjutnya kita view untuk konfirmasi hapus data dengan nama user/confirm_ajax.blade.php

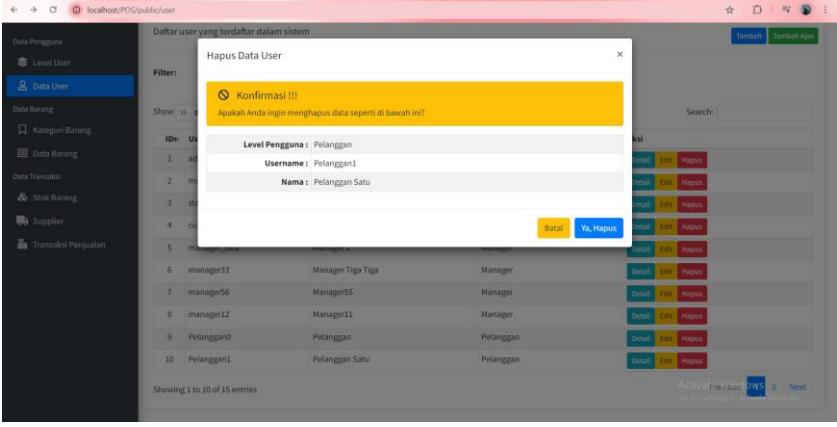
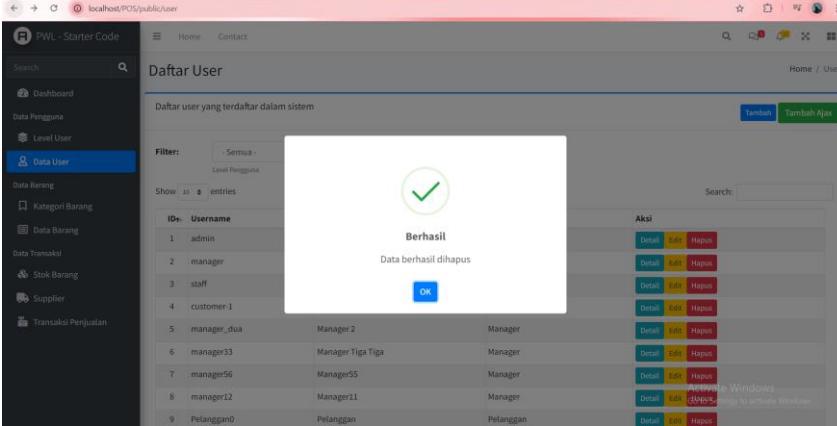


```
messages > user > user.php // controller_user.blade.php
1 <empty>
2 <div id="modal-user" class="modal-dialing modal-lg" role="document">
3   <div class="modal-content">
4     <div class="modal-header">
5       <h5 class="modal-title" id="exampleModalLabel">Kesalahan!&amptimes/span>
8       </button>
9     </div>
10    <div class="modal-body">
11      <div class="alert alert-danger">
12        <div><img alt="Icon fas fa-ban"/> Kesalahan!!</div>
13        Data yang anda cari tidak ditemukan
14      </div>
15      <a href="{{ url('/user') }}" class="btn btn-warning">Kembali</a>
16    </div>
17  </div>
18 </div>
19 <form action="{{ url('/user') . '&user=user_id' . '/delete_ajax' }}" method="POST" id="form-delete">
20   <input type="hidden" name="user_id" value="user_id" />
21   <method>DELETE</method>
22   <div id="modal-user" class="modal-dialing modal-lg" role="document">
23     <div class="modal-content">
24       <div class="modal-header">
25         <h5 class="modal-title" id="exampleModalLabel">Hapus Data User!&amptimes/span>
28         </button>
29       </div>
30       <div class="modal-body">
31         <div class="alert alert-warning">
32           <div><img alt="Icon fas fa-ban"/> Konfirmasi!!!</div>
33           Apakah Anda ingin menghapus data seperti di bawah ini?
34         </div>
35         <table class="table table-bordered table-striped">
36           <thead>
37             <tr>
38               <th class="text-right col-2">Level Pengguna :</th>
39               <td class="col-9">{{ $user->level->level_name }}</td>
40             </tr>
41           <tr>
42             <th class="text-right col-2">Username :</th>
43             <td class="col-9">{{ $user->username }}</td>
44           </tr>
45           <tr>
46             <th class="text-right col-2">Name :</th>
47             <td class="col-9">{{ $user->name }}</td>
48           </tr>
49         </thead>
50         <tbody>
51           <tr>
52             <td class="modal-footer">
53               <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
54               <button type="submit" class="btn btn-primary">Ya, Hapus</button>
55             </td>
56           </tr>
57         </tbody>
58       </table>
59     </div>
60   </div>
61 </form>
62 <script>
63 $(document).ready(function() {
64   $('#form-delete').validate({
65     rules: {},
66     submitHandler: function(form) {
67       $.ajax({
68         url: form.action,
69         type: form.method,
70         data: $(form).serialize(),
71         success: function(response) {
72           if (response.status) {
73             $('#exampleModal').modal('hide');
74             Swal.fire({
75               icon: 'success',
76               title: 'Berhasil',
77               text: response.message
78             });
79             database.ajax.reload();
80           } else {
81             $('#error-text').text('');
82             $.each(response.msgField, function(prefix, val) {
83               $('#error-' + prefix).text(val[0]);
84             });
85             Swal.fire({
86               icon: 'error',
87               title: 'Terjadi Kesalahan',
88               text: response.message
89             });
90           }
91         }
92       );
93       return false;
94     },
95     errorElement: 'span',
96     errorPlacement: function(error, element) {
97       error.addClass('invalid-feedback');
98       element.closest('.form-group').append(error);
99     },
100     highlight: function(element, errorClass, validClass) {
101       $(element).addClass('is-invalid');
102     },
103     unhighlight: function(element, errorClass, validClass) {
104       $(element).removeClass('is-invalid');
105     }
106   });
107 });
108 </script>
109 </empty>
```

5

Kemudian kita buat fungsi `delete_ajax()` pada `UserController.php` untuk mengakomodir `request` hapus data user



	<pre>266 267 268 269 270 public function delete_ajax(Request \$request, \$id): JsonResponse mixed Redirector RedirectRes... 271 { 272 // Cek apakah request dari ajax 273 if (\$request->ajax() \$request->wantsJson()) { 274 \$user = UserModel::find(\$id); 275 276 if (\$user) { 277 \$user->delete(); 278 return response()->json([279 'status' => true, 280 'message' => 'Data berhasil dihapus' 281]); 282 } else { 283 return response()->json([284 'status' => false, 285 'message' => 'Data tidak ditemukan' 286]); 287 } 288 289 } 290 } 291 }</pre>
6	<p>Setelah semua selesai, mari kita coba untuk melakukan percobaan dari koding yang telah kita lakukan.</p>  
7	<p>Jangan lupa laporkan ke laporan jobsheet dan lakukan <i>commit</i> pada <i>repository git</i> kalian!!!</p>

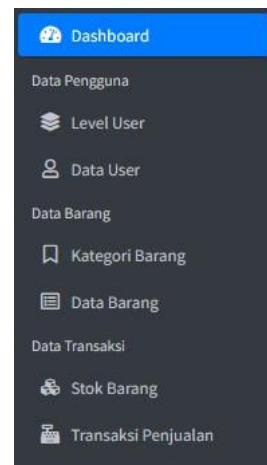


Soal Praktikum

Implementasikan koding untuk Ajax Form dan Client Validation dengan jQuery Validation pada menu berikut ini

- ✓ Tabel m_level
- ✓ Tabel m_kategori
- ✓ Tabel m_supplier
- ✓ Tabel m_barang

Laporkan pada laporan jobsheet dan Jangan lupa di commit dan push pada repository git kalian.



A. Tabel m_level

Langkah	Jawaban/Deskripsi
1	Pertama saya melakukan penambahan pada route untuk menentukan tampilan menu apa saja yang akan ada terkait tabel m_level, seperti gambar di bawah ini yang saya samakan dengan tabel m_user dengan memberi tanda nama ajax dari praktikum di atas yaitu menambah data, edit dan hapus data. <pre>routes > web.php > Closure 31 32 // TABEL LEVEL 33 Route::group(attributes: ['prefix' => 'level'], routes: function () { 34 Route::get(uri: '/', action: [LevelController::class, 'index']); // menampilkan halaman awal level 35 Route::post(uri: '/list', action: [LevelController::class, 'list']); // menampilkan data level dalam bentuk json untuk datatables 36 Route::get(uri: '/create', action: [LevelController::class, 'create']); // menampilkan halaman form tambah level 37 Route::post(uri: '/', action: [LevelController::class, 'store']); // menyimpan data level baru 38 Route::get(uri: '/create_ajax', action: [LevelController::class, 'create_ajax']); // menampilkan halaman form tambah user 39 Route::post(uri: '/ajax', action: [LevelController::class, 'store_ajax']); // menyimpan data user baru 40 Route::get(uri: '/{id}', action: [LevelController::class, 'show']); // menampilkan detail level 41 Route::get(uri: '/{id}/edit', action: [LevelController::class, 'edit']); // menampilkan halaman form edit level 42 Route::put(uri: '/{id}', action: [LevelController::class, 'update']); // menyimpan perubahan data level 43 Route::get(uri: '/{id}/edit_ajax', action: [LevelController::class, 'edit_ajax']); // menampilkan halaman form edit level ajax 44 Route::put(uri: '/{id}/update_ajax', action: [LevelController::class, 'update_ajax']); // menyimpan perubahan data level ajax 45 Route::get(uri: '/{id}/delete_ajax', action: [LevelController::class, 'confirm_ajax']); // untuk tampilkan form confirm delete level ajax 46 Route::delete(uri: '/{id}/delete_ajax', action: [LevelController::class, 'delete_ajax']); // untuk hapus data level ajax 47 48 }); </pre>
2	Kemudian memodifikasi file View tentang m_level pada file level/index.blade.php



	<pre>\$(document).ready(function() { \$('#level').on('show.bs.modal', function(e) { var modal = \$(this); var title = modal.find('.modal-title'); var body = modal.find('.modal-body'); var button = modal.find('.modal-footer'); var form = modal.find('form'); var fileInput = modal.find('#file'); var statusSelect = modal.find('#status'); if (title.length > 0) { title.html('Tambah Level'); } if (body.length > 0) { body.html(` <div class="form-group"> <label>Level Code</label> <input class="form-control" required="" type="text" value="\${level.level_code}"/> </div> <div class="form-group"> <label>Name Level</label> <input class="form-control" required="" type="text" value="\${level.level_name}"/> </div> <div class="form-group"> <label>Description</label> <input class="form-control" required="" type="text" value="\${level.level_desc}"/> </div> <div class="form-group"> <label>Status</label> <input checked="" class="form-control" type="checkbox" value="1"/> </div> <div class="form-group"> <label>File</label> <input class="form-control" type="file" value="Upload"/> </div> `); } if (button.length > 0) { button.html(` <button class="btn btn-primary" data-dismiss="modal" type="button">Batal</button> <button class="btn btn-success" data-dismiss="modal" type="button">Simpan</button> `); } if (form.length > 0) { form.on('submit', function(e) { e.preventDefault(); var formData = new FormData(this); var file = fileInput[0].files[0]; if (file) { formData.append('file', file); } \$.ajax({ url: '/level/create', type: 'POST', data: formData, success: function(response) { if (response.status === 'success') { alert('Level berhasil ditambahkan'); window.location.href = '/level'; } else { alert('Terjadi kesalahan'); } }, error: function(error) { console.error(error); } }); }); } }); });</pre>
3	Kemudian membuat file <code>LevelController.php</code> . Dan juga membuat fungsi <code>create_ajax()</code> dan <code>store_ajax()</code> untuk menambah data level melalui ajax.

```
app > Http > Controllers > LevelController.php > PHP > LevelController > store_ajax()
11  class LevelController extends Controller
12      /**
13      * @return \Illuminate\Http\JsonResponse
14      */
15      public function store_ajax(Request $request)
16      {
17          // cek apakah request berupa ajax
18          if ($request->ajax() || $request->wantsJson()) {
19              $rules = [
20                  'level_code' => 'required|string|min:3|unique:m_level,level_code',
21                  'level_nama' => 'required|string|max:100',
22              ];
23              // use Illuminate\Support\Facades\Validator;
24              $validator = Validator::make($request->all(), $rules);
25
26              if ($validator->fails()) {
27                  return response()->json(['status' => false, 'message' => 'Validasi Gagal', 'msgField' => $validator->errors()]);
28              }
29              $model = LevelModel::create(['level_code' => $request->level_code, 'level_nama' => $request->level_nama]);
30              return response()->json(['status' => true, 'message' => 'Data level berhasil disimpan']);
31          }
32          redirect('/');
33      }
34  }
```



4	Lalu membuat view pada PWL/resources/views/level/create_ajax.blade.php seperti berikut
5	Berikut adalah tampilan tambah data ajax level ketika di akses.

```
resources > views > level > create_ajax.blade.php > form&form-tambah > div>modal modal modal-dialog modal-lg > div>modal-content > div>modal-body
1  <div action="{{ url('/level/ajax') }}" method="POST" id="Form-tambah">
2    @csrf
3    <div id="modal-master" class="modal-dialog modal-lg" role="document">
4      <div class="modal-content">
5        <div class="modal-header">
6          <h5 class="modal-title" id="exampleModallabel">Tambah Data Level</h5>
7          <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8            | <span aria-hidden="true">&times;
```



The screenshots show the process of adding a new user level. In the first two images, a modal dialog titled "Tambah Data Level" is open, prompting for "Kode Level" (PIC) and "Nama Level" (Person in Charge). The third screenshot shows the successful addition of the data, indicated by a green checkmark icon and the message "Berhasil" (Successful), followed by "Data level berhasil disimpan" (User level data has been successfully saved) and an "OK" button.

Screenshot 1: A modal dialog titled "Tambah Data Level" is displayed. It contains fields for "Kode Level" (ADM) and "Nama Level" (Manager). Below the dialog is a table showing existing user levels: ADM, MNG, STF, CUS, and DRK.

ID	Kode Level	Nama Level	Action
1	ADM	Manager	[Detail] [Edit] [Hapus]
2	MNG	Staff/Kasir	[Detail] [Edit] [Hapus]
3	STF	Pelanggan	[Detail] [Edit] [Hapus]
4	CUS	Direktur	[Detail] [Edit] [Hapus]

Screenshot 2: Similar to Screenshot 1, but the "Nama Level" field now contains "Person in Charge".

ID	Kode Level	Nama Level	Action
1	ADM	Manager	[Detail] [Edit] [Hapus]
2	MNG	Staff/Kasir	[Detail] [Edit] [Hapus]
3	STF	Pelanggan	[Detail] [Edit] [Hapus]
4	CUS	Direktur	[Detail] [Edit] [Hapus]
5	DRK	Person in Charge	[Detail] [Edit] [Hapus]

Screenshot 3: The main page after saving. A success message "Berhasil" and "Data level berhasil disimpan" is shown, along with an "OK" button.

Daftar Level User

Daftar level user yang terdaftar dalam sistem

Tambah Tambah Ajax

Show 10

ID	Kode Level	Nama Level	Action
1	ADM	Manager	[Detail] [Edit] [Hapus]
2	MNG	Staff/Kasir	[Detail] [Edit] [Hapus]
3	STF	Pelanggan	[Detail] [Edit] [Hapus]
4	CUS	Direktur	[Detail] [Edit] [Hapus]
5	DRK	Person in Charge	[Detail] [Edit] [Hapus]

Berhasil

Data level berhasil disimpan

OK

Showing 1 to 5 of 5 entries

Previous 1 Next

Gambar di atas adalah tampilan halaman penambahan level user yang mana hanya perlu memasukan kode level dan nama level, kemudian klik simpan untuk menyimpan penambahan data maka akan kembali ke halaman list daftar



	level user yang akan menampilkan data terupdate dengan centang hijau yang menunjukan data berhasil diinputkan
6	Sekarang membuat form untuk mengedit data, maka diperlukan modifikasi pada file LevelController.php dengan isi fungsi list() seperti berikut
	<pre>app > Http > Controllers > LevelController.php > PHP Intelephense > LevelController > list 11 class LevelController extends Controller 30 // Ambil data level dalam bentuk JSON untuk datatables 31 reference overrides 32 public function list(Request \$request): JsonResponse 33 { 34 \$levels = LevelModel::select(columns: 'level_id', 'level_kode', 'level_nama'); 35 36 // Filter data level berdasarkan level_id 37 // if (\$request->level_id) { 38 // \$levels->where('level_id', \$request->level_id); 39 // } 40 41 // return DataTables::of(\$levels) 42 // ->addIndexColumn() 43 // ->addColumn('aksi', function (\$level) { // menambahkan kolom aksi 44 // \$btn = 'level_id) . "' class="btn btn-info btn-sm">Detail '; 45 // \$btn .= 'level_id . '/edit') . "' class="btn btn-warning btn-sm">Edit '; 46 // \$btn .= '<form class="d-inline-block" method="POST" action=' . 47 // url('/level/' . \$level->level_id) . '>' . 48 // csrf_field() . method_field('DELETE') . 49 // '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\\'Apakah Anda yakin menghapus data ini?\')";>Hapus</button></form>'; 50 // } 51 // ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah HTML 52 // ->make(true); 53 } 54 55 return DataTables::of(\$levels) 56 ->addIndexColumn() // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex) 57 ->addColumn('aksi', function (\$level): string { // menambahkan kolom aksi 58 \$btn = 'level_id) . "' class="btn btn-info btn-sm">Detail '; 59 \$btn .= '<button onclick="modalAction(' . url(path: "/level/" . \$level->level_id . '/edit_ajax') . '\')" class="btn btn-warning btn-sm">Edit</button> '; 60 \$btn .= '<button onclick="modalAction(' . url(path: "/level/" . \$level->level_id . '/delete_ajax') . '\')" class="btn btn-danger btn-sm">Hapus</button> '; 61 return \$btn; 62 }) 63 ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html 64 ->make(true); } </pre>



8

Selanjutnya membuat form untuk mengedit data, maka diperlukan file PWL/resources/views/level/edit_ajax.blade.php seperti berikut.

```
resources > views > level > edit_ajax.blade.php > div#modal-owner.modal-dialog.modal-lg > form#form-edit > div#modal-owner.modal-dialog.modal-lg > div#modal-content > div#modal-header > button.close
1  @empty($level)
2    <div id="modal-owner" class="modal-dialog modal-lg" role="document">
3      <div class="modal-content">
4        <div class="modal-header">
5          <h5 class="modal-title" id="exampleModalLabel">Kesalahan!!</h5>
6          <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
7            aria-hidden="true">&amptimes</span></button>
8        </div>
9        <div class="modal-body">
10          <div class="alert alert-danger">
11            <small><span class="icon fas fa-ban"></span> Kesalahan!!</small>
12            Data yang anda cari tidak ditemukan
13          </div>
14          <a href="{{ url('/level') }}" class="btn btn-warning">Kembali</a>
15        </div>
16      </div>
17    </div>
18  @else
19    <form action="{{ url('/level/' . $level->level_id . '/update_ajax') }}" method="POST" id="form-edit">
20      @csrf
21      @method('PUT')
22      <div id="modal-owner" class="modal-dialog modal-lg" role="document">
23        <div class="modal-content">
24          <div class="modal-header">
25            <h5 class="modal-title" id="exampleModalLabel">Edit Data Level</h5>
26            <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
27              aria-hidden="true">&amptimes</span></button>
28          </div>
29          <div class="modal-body">
30            <div class="form-group">
31              <label>Level Kode</label>
32              <input value="{{ $level->level_code }}" type="text" name="level_code" id="level_code"
33                class="form-control" required>
34              <small id="error-level_code" class="error-text form-text text-danger"></small>
35            </div>
36            <div class="form-group">
37              <label>Nama Level</label>
38              <input value="{{ $level->level_name }}" type="text" name="level_name" id="level_name"
39                class="form-control" required>
40              <small id="error-level_name" class="error-text form-text text-danger"></small>
41            </div>
42          </div>
43          <div class="modal-footer">
44            <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
45            <button type="submit" class="btn btn-primary">Simpan</button>
46          </div>
47        </div>
48      </form>
49      <script>
50        $(document).ready(function() {
51          $('#form-edit').validate({
52            rules: {
53              level_code: {
54                required: true,
55                minlength: 3,
56                maxlength: 20
57              },
58              level_name: {
59                required: true,
60                minlength: 3,
61                maxlength: 100
62              }
63            },
64            submitHandler: function(form) {
65              $.ajax({
66                url: form.action,
67                type: form.method,
68                data: $(form).serialize(),
69                success: function(response) {
70                  if (response.status) {
71                    $('#myModal').modal('hide');
72                    Swal.fire({
73                      icon: 'success',
74                      title: 'Berhasil',
75                      text: response.message
76                    });
77                    dataLevel.ajax.reload();
78                  } else {
79                    $('#error-text').text('');
80                    $.each(response.msgField, function(prefix, val) {
81                      $('#error-' + prefix).text(val[0]);
82                    });
83                    Swal.fire({
84                      icon: 'error',
85                      title: 'Terjadi Kesalahan',
86                      text: response.message
87                    });
88                  }
89                });
90                return false;
91            },
92            errorElement: 'span',
93            errorPlacement: function(error, element) {
94              error.addClass('invalid-feedback');
95              element.closest('.form-group').append(error);
96            },
97            highlight: function(element, errorClass, validClass) {
98              $(element).addClass('is-invalid');
99            },
100            unhighlight: function(element, errorClass, validClass) {
101              $(element).removeClass('is-invalid');
102            }
103          });
104        });
105      </script>
106    @endempty
107  
```



Dan berikut adalah tampilan halaman edit level user.

The screenshot shows two browser windows. The left window displays a modal dialog titled 'Edit Data level' with fields for 'Level Kode' (set to 'PIC') and 'Nama Level' (set to 'Person In Case'). Below the dialog is a table of level data. The right window shows the 'Daftar Level User' page with a success message: 'Berhasil Data berhasil diupdate' (Success Data successfully updated) and a green checkmark icon.

- 9 Untuk menampilkan detail data level user maka diperlukan memodifikasi file `LevelController.php` seperti berikut. Dengan menambahkan fungsi `confirm_ajax()` dan `delete_ajax()`.

```
app > Http > Controllers >  LevelController.php > PHP >  LevelController >  delete_ajax()
11 class LevelController extends Controller
  1 reference | 0 overrides
251     public function confirm_ajax(string $id): Factory|View
252     {
253         $level = LevelModel::find(id: $id);
254         return view(view: 'level.confirm_ajax', data: ['level' => $level]);
255     }
256
257     1 reference | 0 overrides
258     public function delete_ajax(Request $request, $id): JsonResponse|mixed\Redirector\RedirectRes...
259     [
260         // cek apakah request dari ajax
261         if ($request->ajax() || $request->wantsJson()) {
262             $level = LevelModel::find(id: $id);
263             if ($level) {
264                 $level->delete();
265                 return response()->json(data: [
266                     'status' => true,
267                     'message' => 'Data berhasil dihapus'
268                 ]);
269             } else {
270                 return response()->json(data: [
271                     'status' => false,
272                     'message' => 'Data tidak ditemukan'
273                 ]);
274             }
275         }
276     ]
277 }
```

Dan dilanjutkan dengan membuat file view `confirm_ajax.blade.php`.



```
resources > view > level > m_confirm_ajaxDelete.php > script > <function>
1  @empty($level)
2  	

3    	

4      	

5         

###### Kesalahan!!


6         <span>aria-hidden="true"&lt;/span>
7


8      	

9         <div class="alert alert-danger">
10           <h5><i class="icon fas fa-ban"></i> Kesalahan!!</h5>
11           Data yang anda cari tidak ditemukan
12         </div>
13         <a href="{{ url('/level') }}" class="btn btn-warning">Kembali</a>
14


15


16   </div>
17 </div>
18 <else>
19   <form action="{{ url('/level/' . $level->level_id . '/delete_ajax') }}" method="POST" id="Form-Delete">
20     <input type="hidden" name="_method" value="DELETE" />
21     <div id="modal-master" class="modal-dialog modal-lg" role="document">
22      	

23         <div class="modal-header">
24           <h6 class="modal-title" id="exampleModalLabel">Hapus Data Level</h6>
25           <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>aria-hidden="true"&lt;/span>
26         </div>
27         <div class="modal-body">
28           <div class="alert alert-warning">
29             <h5><i class="icon fas fa-ban"></i> Konfirmasi!!!</h5>
30             Apakah Anda ingin menghapus data seperti di bawah ini?
31           </div>
32           <table class="table table-sm table-bordered table-striped">
33             <tr>
34               <th class="text-right col-3">Level Kode :</th>
35               <td class="col-9">{{ $level->level_kode }}</td>
36             </tr>
37             <tr>
38               <th class="text-right col-3">Nama Level :</th>
39               <td class="col-9">{{ $level->level_nama }}</td>
40             </tr>
41           </table>
42         </div>
43         <div class="modal-footer">
44           <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
45           <button type="submit" class="btn btn-primary">Ya, Hapus</button>
46         </div>
47       </div>
48     </form>
49   </div>
50 </script>
51 <script>
52   $(document).ready(function() {
53     $('#Form-delete').validate({
54       rules: {},
55       submitHandler: function(form) {
56         $.ajax({
57           url: form.action,
58           type: form.method,
59           data: $(form).serialize(),
60           success: function(response) {
61             if (response.status) {
62               $('#myModal').modal('hide');
63               Swal.fire({
64                 icon: 'success',
65                 title: 'Berhasil',
66                 text: response.message
67               });
68               datalevel.ajax.reload();
69             } else {
70               $('.error-text').text('');
71               $.each(response.msgField, function(prefix, val) {
72                 $('.error-' + prefix).text(val[0]);
73               });
74               Swal.fire({
75                 icon: 'error',
76                 title: 'Terjadi Kesalahan',
77                 text: response.message
78               });
79             }
80           }
81         );
82         return false;
83       },
84       errorElement: 'span',
85       errorPlacement: function(error, element) {
86         error.addClass('invalid-feedback');
87         element.closest('.form-group').append(error);
88       },
89       highlight: function(element, errorClass, validClass) {
90         $(element).addClass('is-invalid');
91       },
92       unhighlight: function(element, errorClass, validClass) {
93         $(element).removeClass('is-invalid');
94       }
95     });
96   });
97 </script>
98 </empty>


```

10	Berikut adalah tampilan halaman detail level user yang mengambil data dari database m_level.
----	--



The left screenshot shows a modal dialog titled "Hapus Data level" with a yellow background. It contains a question "Konfirmasi !!!" and "Apakah Anda ingin menghapus data seperti di bawah ini?". Below the question, there is a table with one row: "Level Kode : PIC" and "Nama level : Person In Crime". At the bottom of the modal are two buttons: "Batal" (Cancel) and "Ya, Hapus" (Yes, Delete). The main page behind the modal is a table titled "Daftar Level User" showing six rows of data. The right screenshot shows the same table after the deletion. A message box in the center says "Berhasil" (Successful) and "Data berhasil dihapus" (Data deleted successfully). There is also a green checkmark icon.

Yang mana tampilan halaman nya seperti gambar diatas, yaitu ketika salah satu data dihapus maka akan terdapat pop up konfirmasi penghapusan dan keterangan yang menunjukan berhasil tidaknya data tersebut dihapus.

B. Tabel m_kategori

Langkah	Jawaban/Deskripsi
1	Pertama saya melakukan penambahan pada route untuk menentukan tampilan menu apa saja yang akan ada terkait tabel m_kategori, seperti gambar di bawah ini yang saya samakan dengan tabel m_user dengan memberi tanda nama ajax dari praktikum di atas yaitu menambah data, edit dan hapus data. <pre>routes > web.php > Closure 49 50 // TABEL KATEGORI 51 Route::group(attributes: ['prefix' => 'kategori'], routes: function () { 52 Route::get(uri: '/', action: [KategoriController::class, 'index']); // menampilkan halaman awal kategori 53 Route::post(uri: '/list', action: [KategoriController::class, 'list']); // menampilkan data level dalam bentuk json untuk datatables 54 Route::get(uri: '/create', action: [KategoriController::class, 'create']); // menampilkan halaman form tambah kategori 55 Route::post(uri: '/', action: [KategoriController::class, 'store']); // menyimpan data kategori baru 56 Route::get(uri: '/create_ajax', action: [KategoriController::class, 'create_ajax']); // menampilkan halaman form tambah user 57 Route::post(uri: '/ajax', action: [KategoriController::class, 'store_ajax']); // menyimpan data user baru 58 Route::get(uri: '/{id}', action: [KategoriController::class, 'show']); // menampilkan detail kategori 59 Route::get(uri: '/{id}/edit', action: [KategoriController::class, 'edit']); // menampilkan halaman form edit kategori 60 Route::put(uri: '/{id}', action: [KategoriController::class, 'update']); // menyimpan perubahan data kategori 61 Route::get(uri: '/{id}/edit_ajax', action: [KategoriController::class, 'edit_ajax']); // menampilkan halaman form edit kategori ajax 62 Route::put(uri: '/{id}/update_ajax', action: [KategoriController::class, 'update_ajax']); // menyimpan perubahan data kategori ajax 63 Route::get(uri: '/{id}/delete_ajax', action: [KategoriController::class, 'confirm_ajax']); // untuk tampilan form confirm delete kategori ajax 64 Route::delete(uri: '/{id}/delete_ajax', action: [KategoriController::class, 'delete_ajax']); // untuk hapus data kategori ajax 65 Route::delete(uri: '/{id}', action: [KategoriController::class, 'destroy']); // menghapus data kategori 66 }); });</pre>
2	Kemudian memodifikasi file View tentang m_kategori pada file kategori/index.blade.php



```
resources > views > kategori > index.blade.php > script > <function>
1 @extends('layouts.template')
2  @section('content')
3    <div id="myModal" class="modal fade animate shake" tabindex="-1" role="dialog" data-backdrop="static"
4      data-keyboard="false" data-width="75%" aria-hidden="true"></div>
5    <div class="card card-primary">
6      <div class="card-header">
7        <h3>{{ $page->title }}</h3>
8      </div>
9      <div class="card-body">
10        @if (session('success'))
11          <div class="alert alert-success">{{ session('success') }}</div>
12        @endif
13        @if (session('error'))
14          <div class="alert alert-danger">{{ session('error') }}</div>
15        @endif
16      </div>
17      <div class="row">
18        <div class="col-md-12">
19          <div class="form-group row">
20            <label class="col-1 control-label col-form-label" for="kategori_id">Filter:</label>
21            <div class="col-3">
22              <select class="form-control" id="kategori_id" name="kategori_id" required>
23                <option value=""> Semua </option>
24                @foreach($kategori as $item)
25                  <option value="{{ $item->kategori_id }}>{{ $item->kategori_nama }}</option>
26                @endforeach
27              </select>
28              <small class="form-text text-muted">Kategori Produk</small>
29            </div>
30          </div>
31        </div>
32      </div>
33    </div>
34    <table class="table table-bordered table-striped table-hover table-sm" id="table_kategori">
35      <thead>
36        <tr>
37          <th>ID</th>
38          <th>Kode Kategori</th>
39          <th>Nama Kategori</th>
40          <th>Aksi</th>
41        </tr>
42      </thead>
43      <tbody>
44        <tr>
45        </tr>
46      </tbody>
47    </table>
48  </div>
49  @endsession
50  @push('css')
51  @push('js')
52    <script>
53      function modalAction(url = '') {
54        $('#myModal').load(url, function() {
55          $('#myModal').modal('show');
56        });
57      }
58      var datakategori;
59      $(document).ready(function() {
60        datakategori = $('#table_kategori').DataTable({
61          serverSide: true,
62          ajax: {
63            "url": "{{ url('kategori/list') }}",
64            "datatype": "json",
65            "type": "POST",
66            "data": function(d) {
67              d.kategori_id = $('#kategori_id').val();
68            }
69          },
70          columns: [
71            {
72              data: "DT_RowIndex",
73              className: "text-center",
74              orderable: false,
75              searchable: false
76            },
77            {
78              data: "kategori_kode",
79              className: "",
80              orderable: true,
81              searchable: true
82            },
83            {
84              data: "kategori_nama",
85              className: "",
86              orderable: true,
87              searchable: true
88            },
89            {
90              data: "aksi",
91              className: "",
92              orderable: false,
93              searchable: false
94            }
95          ]
96        });
97        // $('#kategori_id').on('change', function() {
98        //   datakategori.ajax.reload();
99        // });
100      });
101    </script>
102  @endpush
103  @endpush
104  @endpush
105  @endpush
106  @endpush
107  @endpush
108  @endpush
109  @endpush
110  @endpush
111  @endpush
112  @endpush
113  @endpush
114  @endpush
115  @endpush
116  @endpush
117  @endpush
118  @endpush
119  @endpush
120  @endpush
121  @endpush
122  @endpush
123  @endpush
124  @endpush
125  @endpush
126  @endpush
127  @endpush
128  @endpush
129  @endpush
130  @endpush
131  @endpush
132  @endpush
133  @endpush
134  @endpush
135  @endpush
136  @endpush
137  @endpush
138  @endpush
139  @endpush
140  @endpush
141  @endpush
142  @endpush
143  @endpush
144  @endpush
145  @endpush
146  @endpush
147  @endpush
148  @endpush
149  @endpush
150  @endpush
151  @endpush
152  @endpush
153  @endpush
154  @endpush
155  @endpush
156  @endpush
157  @endpush
158  @endpush
159  @endpush
160  @endpush
161  @endpush
162  @endpush
163  @endpush
164  @endpush
165  @endpush
166  @endpush
167  @endpush
168  @endpush
169  @endpush
170  @endpush
171  @endpush
172  @endpush
173  @endpush
174  @endpush
175  @endpush
176  @endpush
177  @endpush
178  @endpush
179  @endpush
180  @endpush
181  @endpush
182  @endpush
183  @endpush
184  @endpush
185  @endpush
186  @endpush
187  @endpush
188  @endpush
189  @endpush
190  @endpush
191  @endpush
192  @endpush
193  @endpush
194  @endpush
195  @endpush
196  @endpush
197  @endpush
198  @endpush
199  @endpush
200  @endpush
201  @endpush
202  @endpush
203  @endpush
204  @endpush
205  @endpush
206  @endpush
207  @endpush
208  @endpush
209  @endpush
210  @endpush
211  @endpush
212  @endpush
213  @endpush
214  @endpush
215  @endpush
216  @endpush
217  @endpush
218  @endpush
219  @endpush
220  @endpush
221  @endpush
222  @endpush
223  @endpush
224  @endpush
225  @endpush
226  @endpush
227  @endpush
228  @endpush
229  @endpush
230  @endpush
231  @endpush
232  @endpush
233  @endpush
234  @endpush
235  @endpush
236  @endpush
237  @endpush
238  @endpush
239  @endpush
240  @endpush
241  @endpush
242  @endpush
243  @endpush
244  @endpush
245  @endpush
246  @endpush
247  @endpush
248  @endpush
249  @endpush
250  @endpush
251  @endpush
252  @endpush
253  @endpush
254  @endpush
255  @endpush
256  @endpush
257  @endpush
258  @endpush
259  @endpush
260  @endpush
261  @endpush
262  @endpush
263  @endpush
264  @endpush
265  @endpush
266  @endpush
267  @endpush
268  @endpush
269  @endpush
270  @endpush
271  @endpush
272  @endpush
273  @endpush
274  @endpush
275  @endpush
276  @endpush
277  @endpush
278  @endpush
279  @endpush
280  @endpush
281  @endpush
282  @endpush
283  @endpush
284  @endpush
285  @endpush
286  @endpush
287  @endpush
288  @endpush
289  @endpush
290  @endpush
291  @endpush
292  @endpush
293  @endpush
294  @endpush
295  @endpush
296  @endpush
297  @endpush
298  @endpush
299  @endpush
300  @endpush
301  @endpush
302  @endpush
303  @endpush
304  @endpush
305  @endpush
306  @endpush
307  @endpush
308  @endpush
309  @endpush
310  @endpush
311  @endpush
312  @endpush
313  @endpush
314  @endpush
315  @endpush
316  @endpush
317  @endpush
318  @endpush
319  @endpush
320  @endpush
321  @endpush
322  @endpush
323  @endpush
324  @endpush
325  @endpush
326  @endpush
327  @endpush
328  @endpush
329  @endpush
330  @endpush
331  @endpush
332  @endpush
333  @endpush
334  @endpush
335  @endpush
336  @endpush
337  @endpush
338  @endpush
339  @endpush
340  @endpush
341  @endpush
342  @endpush
343  @endpush
344  @endpush
345  @endpush
346  @endpush
347  @endpush
348  @endpush
349  @endpush
350  @endpush
351  @endpush
352  @endpush
353  @endpush
354  @endpush
355  @endpush
356  @endpush
357  @endpush
358  @endpush
359  @endpush
360  @endpush
361  @endpush
362  @endpush
363  @endpush
364  @endpush
365  @endpush
366  @endpush
367  @endpush
368  @endpush
369  @endpush
370  @endpush
371  @endpush
372  @endpush
373  @endpush
374  @endpush
375  @endpush
376  @endpush
377  @endpush
378  @endpush
379  @endpush
380  @endpush
381  @endpush
382  @endpush
383  @endpush
384  @endpush
385  @endpush
386  @endpush
387  @endpush
388  @endpush
389  @endpush
390  @endpush
391  @endpush
392  @endpush
393  @endpush
394  @endpush
395  @endpush
396  @endpush
397  @endpush
398  @endpush
399  @endpush
400  @endpush
401  @endpush
402  @endpush
403  @endpush
404  @endpush
405  @endpush
406  @endpush
407  @endpush
408  @endpush
409  @endpush
410  @endpush
411  @endpush
412  @endpush
413  @endpush
414  @endpush
415  @endpush
416  @endpush
417  @endpush
418  @endpush
419  @endpush
420  @endpush
421  @endpush
422  @endpush
423  @endpush
424  @endpush
425  @endpush
426  @endpush
427  @endpush
428  @endpush
429  @endpush
430  @endpush
431  @endpush
432  @endpush
433  @endpush
434  @endpush
435  @endpush
436  @endpush
437  @endpush
438  @endpush
439  @endpush
440  @endpush
441  @endpush
442  @endpush
443  @endpush
444  @endpush
445  @endpush
446  @endpush
447  @endpush
448  @endpush
449  @endpush
450  @endpush
451  @endpush
452  @endpush
453  @endpush
454  @endpush
455  @endpush
456  @endpush
457  @endpush
458  @endpush
459  @endpush
460  @endpush
461  @endpush
462  @endpush
463  @endpush
464  @endpush
465  @endpush
466  @endpush
467  @endpush
468  @endpush
469  @endpush
470  @endpush
471  @endpush
472  @endpush
473  @endpush
474  @endpush
475  @endpush
476  @endpush
477  @endpush
478  @endpush
479  @endpush
480  @endpush
481  @endpush
482  @endpush
483  @endpush
484  @endpush
485  @endpush
486  @endpush
487  @endpush
488  @endpush
489  @endpush
490  @endpush
491  @endpush
492  @endpush
493  @endpush
494  @endpush
495  @endpush
496  @endpush
497  @endpush
498  @endpush
499  @endpush
500  @endpush
501  @endpush
502  @endpush
503  @endpush
504  @endpush
505  @endpush
506  @endpush
507  @endpush
508  @endpush
509  @endpush
510  @endpush
511  @endpush
512  @endpush
513  @endpush
514  @endpush
515  @endpush
516  @endpush
517  @endpush
518  @endpush
519  @endpush
520  @endpush
521  @endpush
522  @endpush
523  @endpush
524  @endpush
525  @endpush
526  @endpush
527  @endpush
528  @endpush
529  @endpush
530  @endpush
531  @endpush
532  @endpush
533  @endpush
534  @endpush
535  @endpush
536  @endpush
537  @endpush
538  @endpush
539  @endpush
540  @endpush
541  @endpush
542  @endpush
543  @endpush
544  @endpush
545  @endpush
546  @endpush
547  @endpush
548  @endpush
549  @endpush
550  @endpush
551  @endpush
552  @endpush
553  @endpush
554  @endpush
555  @endpush
556  @endpush
557  @endpush
558  @endpush
559  @endpush
560  @endpush
561  @endpush
562  @endpush
563  @endpush
564  @endpush
565  @endpush
566  @endpush
567  @endpush
568  @endpush
569  @endpush
570  @endpush
571  @endpush
572  @endpush
573  @endpush
574  @endpush
575  @endpush
576  @endpush
577  @endpush
578  @endpush
579  @endpush
580  @endpush
581  @endpush
582  @endpush
583  @endpush
584  @endpush
585  @endpush
586  @endpush
587  @endpush
588  @endpush
589  @endpush
590  @endpush
591  @endpush
592  @endpush
593  @endpush
594  @endpush
595  @endpush
596  @endpush
597  @endpush
598  @endpush
599  @endpush
600  @endpush
601  @endpush
602  @endpush
603  @endpush
604  @endpush
605  @endpush
606  @endpush
607  @endpush
608  @endpush
609  @endpush
610  @endpush
611  @endpush
612  @endpush
613  @endpush
614  @endpush
615  @endpush
616  @endpush
617  @endpush
618  @endpush
619  @endpush
620  @endpush
621  @endpush
622  @endpush
623  @endpush
624  @endpush
625  @endpush
626  @endpush
627  @endpush
628  @endpush
629  @endpush
630  @endpush
631  @endpush
632  @endpush
633  @endpush
634  @endpush
635  @endpush
636  @endpush
637  @endpush
638  @endpush
639  @endpush
640  @endpush
641  @endpush
642  @endpush
643  @endpush
644  @endpush
645  @endpush
646  @endpush
647  @endpush
648  @endpush
649  @endpush
650  @endpush
651  @endpush
652  @endpush
653  @endpush
654  @endpush
655  @endpush
656  @endpush
657  @endpush
658  @endpush
659  @endpush
660  @endpush
661  @endpush
662  @endpush
663  @endpush
664  @endpush
665  @endpush
666  @endpush
667  @endpush
668  @endpush
669  @endpush
670  @endpush
671  @endpush
672  @endpush
673  @endpush
674  @endpush
675  @endpush
676  @endpush
677  @endpush
678  @endpush
679  @endpush
680  @endpush
681  @endpush
682  @endpush
683  @endpush
684  @endpush
685  @endpush
686  @endpush
687  @endpush
688  @endpush
689  @endpush
690  @endpush
691  @endpush
692  @endpush
693  @endpush
694  @endpush
695  @endpush
696  @endpush
697  @endpush
698  @endpush
699  @endpush
700  @endpush
701  @endpush
702  @endpush
703  @endpush
704  @endpush
705  @endpush
706  @endpush
707  @endpush
708  @endpush
709  @endpush
710  @endpush
711  @endpush
712  @endpush
713  @endpush
714  @endpush
715  @endpush
716  @endpush
717  @endpush
718  @endpush
719  @endpush
720  @endpush
721  @endpush
722  @endpush
723  @endpush
724  @endpush
725  @endpush
726  @endpush
727  @endpush
728  @endpush
729  @endpush
730  @endpush
731  @endpush
732  @endpush
733  @endpush
734  @endpush
735  @endpush
736  @endpush
737  @endpush
738  @endpush
739  @endpush
740  @endpush
741  @endpush
742  @endpush
743  @endpush
744  @endpush
745  @endpush
746  @endpush
747  @endpush
748  @endpush
749  @endpush
750  @endpush
751  @endpush
752  @endpush
753  @endpush
754  @endpush
755  @endpush
756  @endpush
757  @endpush
758  @endpush
759  @endpush
760  @endpush
761  @endpush
762  @endpush
763  @endpush
764  @endpush
765  @endpush
766  @endpush
767  @endpush
768  @endpush
769  @endpush
770  @endpush
771  @endpush
772  @endpush
773  @endpush
774  @endpush
775  @endpush
776  @endpush
777  @endpush
778  @endpush
779  @endpush
780  @endpush
781  @endpush
782  @endpush
783  @endpush
784  @endpush
785  @endpush
786  @endpush
787  @endpush
788  @endpush
789  @endpush
790  @endpush
791  @endpush
792  @endpush
793  @endpush
794  @endpush
795  @endpush
796  @endpush
797  @endpush
798  @endpush
799  @endpush
800  @endpush
801  @endpush
802  @endpush
803  @endpush
804  @endpush
805  @endpush
806  @endpush
807  @endpush
808  @endpush
809  @endpush
810  @endpush
811  @endpush
812  @endpush
813  @endpush
814  @endpush
815  @endpush
816  @endpush
817  @endpush
818  @endpush
819  @endpush
820  @endpush
821  @endpush
822  @endpush
823  @endpush
824  @endpush
825  @endpush
826  @endpush
827  @endpush
828  @endpush
829  @endpush
830  @endpush
831  @endpush
832  @endpush
833  @endpush
834  @endpush
835  @endpush
836  @endpush
837  @endpush
838  @endpush
839  @endpush
840  @endpush
841  @endpush
842  @endpush
843  @endpush
844  @endpush
845  @endpush
846  @endpush
847  @endpush
848  @endpush
849  @endpush
850  @endpush
851  @endpush
852  @endpush
853  @endpush
854  @endpush
855  @endpush
856  @endpush
857  @endpush
858  @endpush
859  @endpush
860  @endpush
861  @endpush
862  @endpush
863  @endpush
864  @endpush
865  @endpush
866  @endpush
867  @endpush
868  @endpush
869  @endpush
870  @endpush
871  @endpush
872  @endpush
873  @endpush
874  @endpush
875  @endpush
876  @endpush
877  @endpush
878  @endpush
879  @endpush
880  @endpush
881  @endpush
882  @endpush
883  @endpush
884  @endpush
885  @endpush
886  @endpush
887  @endpush
888  @endpush
889  @endpush
890  @endpush
891  @endpush
892  @endpush
893  @endpush
894  @endpush
895  @endpush
896  @endpush
897  @endpush
898  @endpush
899  @endpush
900  @endpush
901  @endpush
902  @endpush
903  @endpush
904  @endpush
905  @endpush
906  @endpush
907  @endpush
908  @endpush
909  @endpush
910  @endpush
911  @endpush
912  @endpush
913  @endpush
914  @endpush
915  @endpush
916  @endpush
917  @endpush
918  @endpush
919  @endpush
920  @endpush
921  @endpush
922  @endpush
923  @endpush
924  @endpush
925  @endpush
926  @endpush
927  @endpush
928  @endpush
929  @endpush
930  @endpush
931  @endpush
932  @endpush
933  @endpush
934  @endpush
935  @endpush
936  @endpush
937  @endpush
938  @endpush
939  @endpush
940  @endpush
941  @endpush
942  @endpush
943  @endpush
944  @endpush
945  @endpush
946  @endpush
947  @endpush
948  @endpush
949  @endpush
950  @endpush
951  @endpush
952  @endpush
953  @endpush
954  @endpush
955  @endpush
956  @endpush
957  @endpush
958  @endpush
959  @endpush
960  @endpush
961  @endpush
962  @endpush
963  @endpush
964  @endpush
965  @endpush
966  @endpush
967  @endpush
968  @endpush
969  @endpush
970  @endpush
971  @endpush
972  @endpush
973  @endpush
974  @endpush
975  @endpush
976  @endpush
977  @endpush
978  @endpush
979  @endpush
980  @endpush
981  @endpush
982  @endpush
983  @endpush
984  @endpush
985  @endpush
986  @endpush
987  @endpush
988  @endpush
989  @endpush
990  @endpush
991  @endpush
992  @endpush
993  @endpush
994  @endpush
995  @endpush
996  @endpush
997  @endpush
998  @endpush
999  @endpush
1000  @endpush
1001  @endpush
1002  @endpush
1003  @endpush
1004  @endpush
1005  @endpush
1006  @endpush
1007  @endpush
1008  @endpush
1009  @endpush
1010  @endpush
1011  @endpush
1012  @endpush
1013  @endpush
1014  @endpush
1015  @endpush
1016  @endpush
1017  @endpush
1018  @endpush
1019  @endpush
1020  @endpush
1021  @endpush
1022  @endpush
1023  @endpush
1024  @endpush
1025  @endpush
1026  @endpush
1027  @endpush
1028  @endpush
1029  @endpush
1030  @endpush
1031  @endpush
1032  @endpush
1033  @endpush
1034  @endpush
1035  @endpush
1036  @endpush
1037  @endpush
1038  @endpush
1039  @endpush
1040  @endpush
1041  @endpush
1042  @endpush
1043  @endpush
1044  @endpush
1045  @endpush
1046  @endpush
1047  @endpush
1048  @endpush
1049  @endpush
1050  @endpush
1051  @endpush
1052  @endpush
1053  @endpush
1054  @endpush
1055  @endpush
1056  @endpush
1057  @endpush
1058  @endpush
1059  @endpush
1060  @endpush
1061  @endpush
1062  @endpush
1063  @endpush
1064  @endpush
1065  @endpush
1066  @endpush
1067  @endpush
1068  @endpush
1069  @endpush
1070  @endpush
1071  @endpush
1072  @endpush
1073  @endpush
1074  @endpush
1075  @endpush
1076  @endpush
1077  @endpush
1078  @endpush
1079  @endpush
1080  @endpush
1081  @endpush
1082  @endpush
1083  @endpush
1084  @endpush
1085  @endpush
1086  @endpush
1087  @endpush
1088  @endpush
1089  @endpush
1090  @endpush
1091  @endpush
1092  @endpush
1093  @endpush
1094  @endpush
1095  @endpush
1096  @endpush
1097  @endpush
1098  @endpush
1099  @endpush
1100  @endpush
1101  @endpush
1102  @endpush
1103  @endpush
1104  @endpush
1105  @endpush
1106  @endpush
1107  @endpush
1108  @endpush
1109  @endpush
1110  @endpush
1111  @endpush
1112  @endpush
1113  @endpush
1114  @endpush
1115  @endpush
1116  @endpush
1117  @endpush
1118  @endpush
1119  @endpush
1120  @endpush
1121  @endpush
1122  @endpush
1123  @endpush
1124  @endpush
1125  @endpush
1126  @endpush
1127  @endpush
1128  @endpush
1129  @endpush
1130  @endpush
1131  @endpush
1132  @endpush
1133  @endpush
1134  @endpush
1135  @endpush
1136  @endpush
1137  @endpush
1138  @endpush
1139  @endpush
1140  @endpush
1141  @endpush
1142  @endpush
1143  @endpush
1144  @endpush
1145  @endpush
1146  @endpush
1147  @endpush
1148  @endpush
1149  @endpush
1150  @endpush
1151  @endpush
1152  @endpush
1153  @endpush
1154  @endpush
1155  @endpush
1156  @endpush
1157  @endpush
1158  @endpush
1159  @endpush
1160  @endpush
1161  @endpush
1162  @endpush
1163  @endpush
1164  @endpush
1165  @endpush
1166  @endpush
1167  @endpush
1168  @endpush
1169  @endpush
1170  @endpush
1171  @endpush
1172  @endpush
1173  @endpush
1174  @endpush
1175  @endpush
1176  @endpush
1177  @endpush
1178  @endpush
1179  @endpush
1180  @endpush
1181  @endpush
1182  @endpush
1183  @endpush
1184  @endpush
1185  @endpush
1186  @endpush
1187  @endpush
1188  @endpush
1189  @endpush
1190  @endpush
1191  @endpush
1192  @endpush
1193  @endpush
1194  @endpush
1195  @endpush
1196  @endpush
1197  @endpush
1198  @endpush
1199  @endpush
1200  @endpush
1201  @endpush
1202  @endpush
1203  @endpush
1204  @endpush
1205  @endpush
1206  @endpush
1207  @endpush
1208  @endpush
1209  @endpush
1210  @endpush
1211  @endpush
1212  @endpush
1213  @endpush
1214  @endpush
1215  @endpush
1216  @endpush
1217  @endpush
1218  @endpush
1219  @endpush
1220  @endpush
1221  @endpush
1222  @endpush
1223  @endpush
1224  @endpush
1225  @endpush
1226  @endpush
1227  @endpush
1228  @endpush
1229  @endpush
1230  @endpush
1231  @endpush
1232  @endpush
1233  @endpush
1234  @endpush
1235  @endpush
1236  @endpush
1237  @endpush
1238  @endpush
1239  @endpush
1240  @endpush
1241  @endpush
1242  @endpush
1243  @endpush
1244  @endpush
1245  @endpush
1246  @endpush
1247  @endpush
1248  @endpush
1249  @endpush
1250  @endpush
1251  @endpush
1252  @endpush
1253  @endpush
1254  @endpush
1255  @endpush
1256  @endpush
1257  @endpush
1258  @endpush
1259  @endpush
1260  @endpush
1261  @endpush
1262  @endpush
1263  @endpush
1264  @endpush
1265  @endpush
1266  @endpush
1267  @endpush
1268  @endpush
1269  @endpush
1270  @endpush
1271  @endpush
1272  @endpush
1273  @endpush
1274  @endpush
1275  @endpush
1276  @endpush
1277  @endpush
1278  @endpush
1279  @endpush
1280  @endpush
1281  @endpush
1282  @endpush
1283  @endpush
1284  @endpush
1285  @endpush
1286  @endpush
1287  @endpush
1288  @endpush
1289  @endpush
1290  @endpush
1291  @endpush
```



4	Lalu membuat view pada PWL/resources/views/kategori/create_ajax.blade.php seperti berikut
5	Berikut adalah tampilan tambah data ajax kategori ketika di akses.

```
resources > views > kategori > create_ajax.blade.php < form>form-tambah < div>modal modal modal-dialog modal-lg < div>modal-content < div>modal-body < div>form-group < label>kategori_kode< input>kategori_kode < small>error-kategori_kode< label>kategori_nama< input>kategori_nama < small>error-kategori_nama< button>Simpan< button>Batal
```



The screenshot shows a web-based point-of-sale (POS) system interface. On the left, there's a sidebar with navigation links: Dashboard, Data Pengguna, Level User, Data User, Data Barang, Kategori Barang (which is highlighted in blue), Data Barangan, Data Transaksi, Stok Barang, Supplier, and Transaksi Penjualan. The main content area has a header 'Daftar Kategori' and a sub-header 'Daftar kategori yang terdaftar dalam sistem'. A modal window titled 'Tambah Data kategori' is open, prompting for 'kategori Kode' (input: spt) and 'Nama kategori' (input: sport). Below the modal, a table lists categories with columns 'ID', 'Kode Kategori', and 'Nama Kategori'. The table shows entries 1 through 5, with the last entry being 'elk' under 'Elektronik'. At the bottom of the page, a success message 'Data kategori berhasil disimpan' (Category data saved successfully) is displayed within a modal, along with a green checkmark icon.

Gambar di atas adalah tampilan halaman penambahan kategori yang mana hanya perlu memasukan kode kategori dan nama kategori, kemudian klik simpan untuk menyimpan penambahan data maka akan kembali ke halaman list kategori yang akan menampilkan data terupdate dengan centang hijau yang menunjukan data berhasil diinputkan

- | | |
|---|---|
| 6 | Sekarang membuat form untuk mengedit data, maka diperlukan modifikasi pada file <code>KategoriController.php</code> dengan isi fungsi <code>list()</code> seperti berikut |
|---|---|



	<pre>app > Http > Controllers > KategoriController.php > PHP > KategoriController > update_ajax() 10 class KategoriController extends Controller 210 public function edit_ajax(string \$id): Factory View 211 { 212 \$kategori = KategoriModel::find(id: \$id); 213 return view(view: 'kategori.edit_ajax', data: ['kategori' => \$kategori]); 214 } 215 216 // reference 0 overrides 217 public function update_ajax(Request \$request, \$id): JsonResponse mixed\Redirector\RedirectResponse 218 { 219 // cek apakah request dari ajax 220 if (\$request->ajax() \$request->wantsJson()) { 221 \$rules = [222 'kategori.kode' => 'required max:20 unique:m_kategori,kategori.kode,' . \$id . ',kategori_id', 223 'kategori.nama' => 'required max:100' 224]; 225 // use Illuminate\Support\Facades\Validator; 226 \$validator = Validator::make(data: \$request->all(), rules: \$rules); 227 if (\$validator->fails()) { 228 return response()->json(data: [229 'status' => false, // respon json, true: berhasil, false: gagal 230 'message' => 'Validasi gagal.', 231 'msgField' => \$validator->errors() // menunjukkan field mana yang error 232]); 233 } 234 \$check = KategoriModel::find(id: \$id); 235 if (\$check) { 236 \$check->update(attributes: \$request->all()); 237 return response()->json(data: [238 'status' => true, 239 'message' => 'Data berhasil diupdate' 240]); 241 } else { 242 return response()->json(data: [243 'status' => false, 244 'message' => 'Data tidak ditemukan' 245]); 246 } 247 } 248 } 249 }</pre>
7	Kemudian membuat file KategoriController.php. Dan juga membuat fungsi <code>edit_ajax()</code> dan <code>update_ajax()</code> untuk mengedit data kategori melalui ajax.



8

Selanjutnya membuat form untuk mengedit data, maka diperlukan file PWL/resources/views/kategori/edit_ajax.blade.php seperti berikut.

```
resources > views > kategori > edit_ajax.blade.php > div#modal-master.modal-dialog.modal-lg > script > <function> > submitHandler > success > <function>
1 @empty($kategori)
2     <div id="modal-master" class="modal-dialog modal-lg" role="document">
3         <div class="modal-content">
4             <div class="modal-header">
5                 <h5 class="modal-title" id="exampleModalLabel">Kesalahan</h5>
6                 <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>×</span></button>
7             <div class="modal-body">
8                 <div class="alert alert-danger">
9                     <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
10                    Data yang anda cari tidak ditemukan
11                </div>
12                <a href="{{ url('/kategori') }}" class="btn btn-warning">Kembali</a>
13            </div>
14        </div>
15    </div>
16 @else
17     <form action="{{ url('/kategori/' . $kategori->kategori_id . '/update_ajax') }}" method="POST" id="form-edit">
18         @csrf
19         @method('PUT')
20         <div id="modal-master" class="modal-dialog modal-lg" role="document">
21             <div class="modal-content">
22                 <div class="modal-header">
23                     <h5 class="modal-title" id="exampleModalLabel">Edit Data kategori</h5>
24                     <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>×</span></button>
25                 </div>
26                 <div class="modal-body">
27                     <div class="form-group">
28                         <label>Kode</label>
29                         <input value="{{ $kategori->kategori_kode }}" type="text" name="kategori.kode" id="kategori.kode" class="form-control required">
30                         <small id="error-kategori.kode" class="error-text form-text text-danger"></small>
31                     </div>
32                     <div class="form-group">
33                         <label>Nama kategori</label>
34                         <input value="{{ $kategori->kategori_nama }}" type="text" name="kategori.nama" id="kategori.nama" class="form-control required">
35                         <small id="error-kategori.nama" class="error-text form-text text-danger"></small>
36                     </div>
37                 </div>
38                 <div class="modal-footer">
39                     <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
40                     <button type="submit" class="btn btn-primary">Simpan</button>
41                 </div>
42             </div>
43         </div>
44     </form>
45 
```

```
resources > views > kategori > edit_ajax.blade.php > div#modal-master.modal-dialog.modal-lg > form#form-edit > div#modal-master.modal-dialog.modal-lg > div.modal-content > div.modal-body > div.form-group > input#kategori.kode.form-control
1 <div id="modal-master" class="modal-dialog modal-lg" role="document">
2     <script>
3         $(document).ready(function() {
4             $('#form-edit').validate({
5                 rules: {
6                     kategori.kode: {
7                         required: true,
8                         minlength: 1,
9                         maxlength: 20
10                    },
11                     kategori.nama: {
12                         required: true,
13                         minlength: 3,
14                         maxlength: 100
15                    }
16                 },
17                 submitHandler: function(form) {
18                     $.ajax({
19                         url: form.action,
20                         type: form.method,
21                         data: $(form).serialize(),
22                         success: function(response) {
23                             if (response.status) {
24                                 $('#myModal').modal('hide');
25                                 Swal.fire({
26                                     icon: 'success',
27                                     title: 'Berhasil',
28                                     text: response.message
29                                 });
30                                 dataKategori.ajax.reload();
31                             } else {
32                                 $('.error-text').text('');
33                                 $.each(response.msgError, function(prefix, val) {
34                                     ('.error-' + prefix).text(val[0]);
35                                 });
36                                 Swal.fire({
37                                     icon: 'error',
38                                     title: 'Terjadi Kesalahan',
39                                     text: response.message
40                                 });
41                             }
42                         });
43                     return false;
44                 },
45                 errorElement: 'span',
46                 errorPlacement: function(error, element) {
47                     error.addClass('invalid-feedback');
48                     element.closest('.form-group').append(error);
49                 },
50                 highlight: function(element, errorClass, validClass) {
51                     $(element).addClass('is-invalid');
52                 },
53                 unhighlight: function(element, errorClass, validClass) {
54                     $(element).removeClass('is-invalid');
55                 }
56             });
57         });
58     </script>
59 
```



Dan berikut adalah tampilan halaman edit level user.

The screenshot shows a modal window titled "Edit Data kategori". Inside the modal, there is a form with fields for "kategori Kode" (containing "spt") and "Nama kategori" (containing "sportif"). Below the modal is a table titled "Daftar kategori" showing six entries:

ID	Kode Kategori	Nama Kategori	Aksi
1	fnb	Food and Beverage	[Detail] [Edit] [Hapus]
2	bnh	Beauty and Health	[Detail] [Edit] [Hapus]
3	bnk	Baby and Kid	[Detail] [Edit] [Hapus]
4	hnc	Home and Care	[Detail] [Edit] [Hapus]
5	elk	Elektronik	[Detail] [Edit] [Hapus]
6	spt	sport	[Detail] [Edit] [Hapus]

The screenshot shows the "Daftar Kategori" page. A modal window in the center displays a green checkmark icon and the text "Berhasil" followed by "Data berhasil diupdate". The background table shows the same category data as the previous screenshot.

- | | |
|---|--|
| 9 | Untuk menampilkan detail data level user maka diperlukan memodifikasi file KategoriController.php seperti berikut. Dengan menambahkan fungsi confirm_ajax() dan delete_ajax(). |
|---|--|



```
app > Http > Controllers > 🗂️ KategoriController.php > PHP Intelephense > 🗂️ KategoriController > ✎ delete_ajax
10 class KategoriController extends Controller
11
12     1 reference | 0 overrides
13     public function confirm_ajax(string $id): Factory|View
14     {
15         $kategori = KategoriModel::find(id: $id);
16         return view(view: 'kategori.confirm_ajax', data: ['kategori' => $kategori]);
17     }
18
19     1 reference | 0 overrides
20     public function delete_ajax(Request $request, $id): JsonResponse|mixed\Redirector\RedirectResponse
21     {
22         // cek apakah request dari ajax
23         if ($request->ajax() || $request->wantsJson()) {
24             $kategori = KategoriModel::find(id: $id);
25             if ($kategori) {
26                 $kategori->delete();
27                 return response()->json(data: [
28                     'status' => true,
29                     'message' => 'Data berhasil dihapus'
30                 ]);
31             } else {
32                 return response()->json(data: [
33                     'status' => false,
34                     'message' => 'Data tidak ditemukan'
35                 ]);
36             }
37         }
38         return redirect(to: '/');
39     }
40 }
41
42 }
```

Dan dilanjutkan dengan membuat file view `confirm_ajax.blade.php`.

```
resources > views > kategori > 🗂️ confirm_ajax.blade.php > ...
1  @empty($kategori)
2    <div id="modal-master" class="modal-dialog modal-lg" role="document">
3      <div class="modal-content">
4        <div class="modal-header">
5          <h5 class="modal-title" id="exampleModallabel">Kesalahan</h5>
6          <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
7              | aria-hidden="true">&times;</span></button>
8        </div>
9        <div class="modal-body">
10           <div class="alert alert-danger">
11             <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
12             Data yang anda cari tidak ditemukan
13           </div>
14           <a href="{{ url('/kategori') }}" class="btn btn-warning">Kembali</a>
15         </div>
16       </div>
17     </div>
18   @else
19     <form action="{{ url('/kategori/' . $kategori->kategori_id . '/delete_ajax') }}" method="POST" id="form-delete">
20       @csrf
21       @method('DELETE')
22       <div id="modal-master" class="modal-dialog modal-lg" role="document">
23         <div class="modal-content">
24           <div class="modal-header">
25             <h5 class="modal-title" id="exampleModallabel">Hapus Data kategori</h5>
26             <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
27                 | aria-hidden="true">&times;</span></button>
28           </div>
29           <div class="modal-body">
30             <div class="alert alert-warning">
31               <h5><i class="icon fas fa-ban"></i> Konfirmasi !!!</h5>
32               Apakah Anda ingin menghapus data seperti di bawah ini?
33             </div>
34             <table class="table table-sm table-bordered table-striped">
35               <thead>
36                 <tr>
37                   <th class="text-right col-3">kategori Kode :</th>
38                   <td class="col-9">{{ $kategori->kategori_kode }}</td>
39               </tr>
40               <tr>
41                   <th class="text-right col-3">>Nama kategori :</th>
42                   <td class="col-9">{{ $kategori->kategori_nama }}</td>
43               </tr>
44             </thead>
45             <tbody>
46               <tr>
47                 <td class="text-right col-3"><button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
48                 <td class="col-9"><button type="submit" class="btn btn-primary">Ya, Hapus</button>
49               </tr>
50             </tbody>
51           </table>
52         </div>
53       </div>
54     </form>
55   </div>
56 
```



	<pre>resources > views > kategori > 🗑 confirm_ajax.blade.php > ... 52 <script> 53 \$(document).ready(function() { 54 \$("#form-delete").validate({ 55 rules: {}, 56 submitHandler: function(form) { 57 \$.ajax({ 58 url: form.action, 59 type: form.method, 60 data: \$(form).serialize(), 61 success: function(response) { 62 if (response.status) { 63 \$('#myModal').modal('hide'); 64 Swal.fire({ 65 icon: 'success', 66 title: 'Berhasil', 67 text: response.message 68 }); 69 dataKategori.ajax.reload(); 70 } else { 71 \$('.error-text').text(''); 72 \$.each(response.msgField, function(prefix, val) { 73 \$('#error-' + prefix).text(val[0]); 74 }); 75 Swal.fire({ 76 icon: 'error', 77 title: 'Terjadi Kesalahan', 78 text: response.message 79 }); 80 } 81 } 82); 83 return false; 84 }, 85 errorElement: 'span', 86 errorPlacement: function(error, element) { 87 error.addClass('invalid-feedback'); 88 element.closest('.form-group').append(error); 89 }, 90 highlight: function(element, errorClass, validClass) { 91 \$(element).addClass('is-invalid'); 92 }, 93 unhighlight: function(element, errorClass, validClass) { 94 \$(element).removeClass('is-invalid'); 95 } 96 }); 97 }); 98 </script> 99 @endempty</pre>
10	Berikut adalah tampilan halaman detail kategori barang yang mengambil data dari database m_kategori.



The screenshot shows a modal dialog titled "Hapus Data kategori" (Delete Category Data) with a yellow background. It contains a question "Konfirmasi !!!" (Confirmation!!!) and "Apakah Anda ingin menghapus data seperti di bawah ini?" (Do you want to delete data like the one below?). Below the dialog, the main table has a row for "spt" with the value "sportif". At the bottom right of the dialog are "Batal" (Cancel) and "Ya, Hapus" (Yes, Delete) buttons.

Daftar Kategori

ID	Kode Kategori	Nama kategori	Aksi
1	fnb	Food and Beverage	[Detail] [Edit] [Hapus]
2	bnh	Beauty and Health	[Detail] [Edit] [Hapus]
3	bnk	Baby and Kid	[Detail] [Edit] [Hapus]
4	hnc	Home and Care	[Detail] [Edit] [Hapus]
5	elk	Elektronik	[Detail] [Edit] [Hapus]
6	spt	sportif	[Detail] [Edit] [Hapus]

Showing 1 to 6 of 6 entries

The screenshot shows a modal dialog with a green checkmark icon and the text "Berhasil" (Success) and "Data berhasil dihapus" (Data deleted successfully). At the bottom right is an "OK" button. The background table now only shows 5 entries.

Daftar Kategori

ID	Kode Kategori	Nama kategori	Aksi
1	fnb		[Detail] [Edit] [Hapus]
2	bnh		[Detail] [Edit] [Hapus]
3	bnk		[Detail] [Edit] [Hapus]
4	hnc		[Detail] [Edit] [Hapus]
5	elk	Elektronik	[Detail] [Edit] [Hapus]

Showing 1 to 5 of 5 entries

Yang mana tampilan halaman nya seperti gambar diatas, yaitu ketika salah satu data dihapus maka akan terdapat pop up konfirmasi penghapusan dan keterangan yang menunjukan berhasil tidaknya data tersebut dihapus.

C. Tabel m_supplier

Langkah	Jawaban/Deskripsi
1	Pertama saya melakukan penambahan pada route untuk menentukan tampilan menu apa saja yang akan ada terkait tabel m_supplier, seperti gambar di bawah ini



	<p>yang saya samakan dengan tabel m_user dengan memberi tanda nama ajax dari praktikum di atas yaitu menambah data, edit dan hapus data.</p> <pre>routes > web.php > Closure 68 // TABEL SUPPLIER 69 Route::group(attributes: ['prefix' => 'supplier'], routes: function () { 70 Route::get(uri: '/', action: [SupplierController::class, 'index']); // menampilkan halaman awal supplier 71 Route::post(uri: '/list', action: [SupplierController::class, 'list']); // menampilkan data supplier dalam bentuk json untuk datatables 72 Route::post(uri: '/create', action: [SupplierController::class, 'create']); // menampilkan halaman form tambah supplier 73 Route::post(uri: '/', action: [SupplierController::class, 'store']); // menyimpan data supplier baru 74 Route::get(uri: '/create_ajax', action: [SupplierController::class, 'create_ajax']); // menampilkan halaman form tambah supplier ajax 75 Route::post(uri: '/ajax', action: [SupplierController::class, 'store_ajax']); // menyimpan data supplier baru 76 Route::get(uri: '/{id}', action: [SupplierController::class, 'show']); // menampilkan detail supplier 77 Route::get(uri: '/{id}/edit', action: [SupplierController::class, 'edit']); // menampilkan halaman form edit supplier 78 Route::put(uri: '/{id}', action: [SupplierController::class, 'update']); // menyimpan perubahan data supplier 79 Route::get(uri: '/{id}/edit_ajax', action: [SupplierController::class, 'edit_ajax']); // menampilkan halaman form edit supplier ajax 80 Route::put(uri: '/{id}/update_ajax', action: [SupplierController::class, 'update_ajax']); // menyimpan perubahan data supplier ajax 81 Route::get(uri: '/{id}/delete_ajax', action: [SupplierController::class, 'confirm_ajax']); // untuk tampilkan form confirm delete supplier ajax 82 Route::delete(uri: '/{id}/delete_ajax', action: [SupplierController::class, 'delete_ajax']); // untuk hapus data supplier ajax 83 Route::delete(uri: '/{id}', action: [SupplierController::class, 'destroy']); // menghapus data supplier 84 }); 85 </pre>
2	<p>Kemudian memodifikasi file View tentang m_supplier pada file supplier/index.blade.php</p> <pre>resources > views > supplier > index.blade.php > script > <function> 1 @extends('layouts.template') 2 @section('content') 3 <div id="myModal" class="modal fade animate shake" tabindex="-1" role="dialog" data-backdrop="static" 4 data-keyboard="false" data-width="75%" aria-hidden="true"></div> 5 <div class="card card-outline card-primary"> 6 <div class="card-header"> 7 <h3 class="card-title">{{ \$page->title }}</h3> 8 <div class="card-tools"> 9 Tambah 10 <button onclick="modalAction('{{ url('stok/create_ajax') }}'" class="btn btn-success mt-1">Tambah Ajax</button> 11 </div> 12 </div> 13 <div class="card-body"> 14 @if (session('success')) 15 <div class="alert alert-success">{{ session('success') }}</div> 16 @endif 17 @if (session('error')) 18 <div class="alert alert-danger">{{ session('error') }}</div> 19 @endif 20 <div class="row"> 21 <div class="col-md-12"> 22 <div class="form-group row"> 23 <label class="col-1 control-label col-form-label" for="supplier_id">Filter:</label> 24 <div class="col-3"> 25 <select class="form-control" id="supplier_id" name="supplier_id" required> 26 <option value=""> Semua </option> 27 @foreach(\$supplier as \$supplier) 28 <option value="{{ \$supplier->supplier_id }}>{{ \$supplier->supplier_nama }}</option> 29 @endforeach 30 </select> 31 <small class="form-text text-muted">Nama Supplier</small> 32 </div> 33 </div> 34 </div> 35 </div> 36 <table class="table table-bordered table-striped table-hover table-sm" id="table_supplier"> 37 <thead> 38 <tr> 39 <th>ID</th> 40 <th>Kode Supplier</th> 41 <th>Nama Supplier</th> 42 <th>Alamat Supplier</th> 43 <th>Aksi</th> 44 </tr> 45 </thead> 46 <tbody> 47 </tbody> 48 </table> 49 </div> 50 @endsection 51 @push('css') 52 </pre>



```
resources > views > supplier > index.blade.php > div.card.card-outline.card-primary > div.card-body > div.row > div.col-md
51 @endpush
52 @push('js')
53 <script>
54     function modalAction(url = '') {
55         $('#myModal').load(url, function() {
56             $('#myModal').modal('show');
57         });
58     }
59     var dataSupplier;
60
61     $(document).ready(function() {
62         dataSupplier = $('#table_supplier').DataTable({
63             serverSide: true,
64             ajax: {
65                 "url": "{{ url('supplier/list') }}",
66                 "dataType": "json",
67                 "type": "POST",
68                 "data": function(d){
69                     d.supplier_id = $('#supplier_id').val();
70                 }
71             },
72             columns: [
73                 {
74                     data: "DT_RowIndex",
75                     className: "text-center",
76                     orderable: false,
77                     searchable: false
78                 },
79                 {
80                     data: "supplier_kode",
81                     className: "",
82                     orderable: true,
83                     searchable: true
84                 },
85                 {
86                     data: "supplier_nama",
87                     className: "",
88                     orderable: true,
89                     searchable: true
90                 },
91                 {
92                     data: "supplier_alamat",
93                     className: "",
94                     orderable: true,
95                     searchable: true
96                 },
97                 {
98                     data: "aksi",
99                     className: "",
100                    orderable: false,
101                    searchable: false
102                 }
103             ]
104         });
105         // $('#supplier_id').on('change', function(){
106         //     dataSupplier.ajax.reload();
107         // });
108     });
109     </script>
110 @endpush
```


3	Kemudian membuat file SupplierController.php. Dan juga membuat fungsi `create_ajax()` dan `store_ajax()` untuk menambah data supplier melalui ajax. ``` app > Http > Controllers > 🏷 SupplierController.php > ... 11 class SupplierController extends Controller 12 13 1 reference	0 overrides 14 public function create_ajax(): Factory	View 15 { 16 return view(view: 'stok.create_ajax'); 17 } 18 19 1 reference	0 overrides 20 public function store_ajax(Request $request): JsonResponse	mixed 21 { 22 // cek apakah request berupa ajax 23 if ($request->ajax()		$request->wantsJson()) { 24 $rules = [25 'supplier_kode' => 'required	string	unique:m_supplier,supplier_kode', // supplier_kode harus unik 26 'supplier_nama' => 'required	string	max:100', // supplier_nama harus diisi dan maksimal 100 karakter 27 'supplier_alamat' => 'nullable	string	max:255' // supplier_alamat opsional 28]; 29 // use Illuminate\Support\Facades\Validator; 30 $validator = Validator::make(data: $request->all(), rules: $rules); 31 32 if ($validator->fails()) { 33 return response()->json(data: [34 'status' => false, // response status, false: error/gagal, true: berhasil 35 'message' => 'Validasi Gagal', 36 'msgField' => $validator->errors(), // pesan error validasi 37]); 38 } 39 SupplierModel::create(attributes: $request->all()); 40 return response()->json(data: [41 'status' => true, 42 'message' => 'Data level berhasil disimpan' 43]); 44 } 45 redirect(to: '/'); 46 } 47 48 2 reference	0 overrides 49 public function index(): Factory	View 50 { 51 return view(view: 'stok.index'); 52 } 53 54 1 reference	0 overrides 55 public function show(int $id): Factory	View 56 { 57 $supplier = SupplierModel::find($id); 58 if ($supplier) { 59 return view(view: 'stok.show', data: [60 'supplier' => $supplier 61]); 62 } 63 return redirect(to: '/'); 64 } 65 66 1 reference	0 overrides 67 public function edit(int $id): Factory	View 68 { 69 $supplier = SupplierModel::find($id); 70 if ($supplier) { 71 return view(view: 'stok.edit', data: [72 'supplier' => $supplier 73]); 74 } 75 return redirect(to: '/'); 76 } 77 78 1 reference	0 overrides 79 public function update(Request $request, int $id): JsonResponse	mixed 80 { 81 $supplier = SupplierModel::find($id); 82 if ($supplier) { 83 $supplier->fill($request->all()); 84 $supplier->save(); 85 return response()->json(data: [86 'status' => true, 87 'message' => 'Data level berhasil diupdate' 88]); 89 } 90 return response()->json(data: [91 'status' => false, 92 'message' => 'Data level gagal diupdate' 93]); 94 } 95 96 1 reference	0 overrides 97 public function destroy(int $id): JsonResponse	mixed 98 { 99 $supplier = SupplierModel::find($id); 100 if ($supplier) { 101 $supplier->delete(); 102 return response()->json(data: [103 'status' => true, 104 'message' => 'Data level berhasil dihapus' 105]); 106 } 107 return response()->json(data: [108 'status' => false, 109 'message' => 'Data level gagal dihapus' 110]); 111 } 112 113 1 reference	0 overrides 114 public function delete_all(Request $request): JsonResponse	mixed 115 { 116 $suppliers = SupplierModel::all(); 117 foreach ($suppliers as $supplier) { 118 $supplier->delete(); 119 } 120 return response()->json(data: [121 'status' => true, 122 'message' => 'Semua data level berhasil dihapus' 123]); 124 } 125 126 1 reference	0 overrides 127 public function search(Request $request): JsonResponse	mixed 128 { 129 $query = $request->get('q'); 130 $suppliers = SupplierModel::where('supplier_nama', 'like', "%{$query}%") 131 ->get(); 132 return response()->json(data: [133 'suppliers' => $suppliers 134]); 135 } 136 137 1 reference	0 overrides 138 public function get_supplier(Request $request): JsonResponse	mixed 139 { 140 $id = $request->get('id'); 141 $supplier = SupplierModel::find($id); 142 if ($supplier) { 143 return response()->json(data: [144 'supplier' => $supplier 145]); 146 } 147 return response()->json(data: [148 'status' => false, 149 'message' => 'Supplier tidak ditemukan' 150]); 151 } 152 153 1 reference	0 overrides 154 public function get_stok(Request $request): JsonResponse	mixed 155 { 156 $id = $request->get('id'); 157 $stok = StokModel::find($id); 158 if ($stok) { 159 return response()->json(data: [160 'stok' => $stok 161]); 162 } 163 return response()->json(data: [164 'status' => false, 165 'message' => 'Stok tidak ditemukan' 166]); 167 } 168 169 1 reference	0 overrides 170 public function get_stok_by_supplier(Request $request): JsonResponse	mixed 171 { 172 $supplier_id = $request->get('supplier_id'); 173 $stoks = StokModel::where('supplier_id', $supplier_id) 174 ->get(); 175 return response()->json(data: [176 'stoks' => $stoks 177]); 178 } 179 180 1 reference	0 overrides 181 public function get_stok_by_barang(Request $request): JsonResponse	mixed 182 { 183 $barang_id = $request->get('barang_id'); 184 $stoks = StokModel::where('barang_id', $barang_id) 185 ->get(); 186 return response()->json(data: [187 'stoks' => $stoks 188]); 189 } 190 191 1 reference	0 overrides 192 public function get_stok_by_tanggal(Request $request): JsonResponse	mixed 193 { 194 $tanggal = $request->get('tanggal'); 195 $stoks = StokModel::where('tanggal', $tanggal) 196 ->get(); 197 return response()->json(data: [198 'stoks' => $stoks 199]); 200 } 201 202 1 reference	0 overrides 203 public function get_stok_by_kategori(Request $request): JsonResponse	mixed 204 { 205 $kategori_id = $request->get('kategori_id'); 206 $stoks = StokModel::where('kategori_id', $kategori_id) 207 ->get(); 208 return response()->json(data: [209 'stoks' => $stoks 210]); 211 } 212 213 1 reference	0 overrides 214 public function get_stok_by_lokasi(Request $request): JsonResponse	mixed 215 { 216 $lokasi_id = $request->get('lokasi_id'); 217 $stoks = StokModel::where('lokasi_id', $lokasi_id) 218 ->get(); 219 return response()->json(data: [220 'stoks' => $stoks 221]); 222 } 223 224 1 reference	0 overrides 225 public function get_stok_by_jenis(Request $request): JsonResponse	mixed 226 { 227 $jenis_id = $request->get('jenis_id'); 228 $stoks = StokModel::where('jenis_id', $jenis_id) 229 ->get(); 230 return response()->json(data: [231 'stoks' => $stoks 232]); 233 } 234 235 1 reference	0 overrides 236 public function get_stok_by_kondisi(Request $request): JsonResponse	mixed 237 { 238 $kondisi_id = $request->get('kondisi_id'); 239 $stoks = StokModel::where('kondisi_id', $kondisi_id) 240 ->get(); 241 return response()->json(data: [242 'stoks' => $stoks 243]); 244 } 245 246 1 reference	0 overrides 247 public function get_stok_by_kemasan(Request $request): JsonResponse	mixed 248 { 249 $kemasan_id = $request->get('kemasan_id'); 250 $stoks = StokModel::where('kemasan_id', $kemasan_id) 251 ->get(); 252 return response()->json(data: [253 'stoks' => $stoks 254]); 255 } 256 257 1 reference	0 overrides 258 public function get_stok_by_satuan(Request $request): JsonResponse	mixed 259 { 260 $satuan_id = $request->get('satuan_id'); 261 $stoks = StokModel::where('satuan_id', $satuan_id) 262 ->get(); 263 return response()->json(data: [264 'stoks' => $stoks 265]); 266 } 267 268 1 reference	0 overrides 269 public function get_stok_by_kemasan_satuan(Request $request): JsonResponse	mixed 270 { 271 $kemasan_id = $request->get('kemasan_id'); 272 $satuan_id = $request->get('satuan_id'); 273 $stoks = StokModel::where('kemasan_id', $kemasan_id) 274 ->where('satuan_id', $satuan_id) 275 ->get(); 276 return response()->json(data: [277 'stoks' => $stoks 278]); 279 } 280 281 1 reference	0 overrides 282 public function get_stok_by_kondisi_kemasan(Request $request): JsonResponse	mixed 283 { 284 $kondisi_id = $request->get('kondisi_id'); 285 $kemasan_id = $request->get('kemasan_id'); 286 $stoks = StokModel::where('kondisi_id', $kondisi_id) 287 ->where('kemasan_id', $kemasan_id) 288 ->get(); 289 return response()->json(data: [290 'stoks' => $stoks 291]); 292 } 293 294 1 reference	0 overrides 295 public function get_stok_by_kondisi_kemasan_satuan(Request $request): JsonResponse	mixed 296 { 297 $kondisi_id = $request->get('kondisi_id'); 298 $kemasan_id = $request->get('kemasan_id'); 299 $satuan_id = $request->get('satuan_id'); 300 $stoks = StokModel::where('kondisi_id', $kondisi_id) 301 ->where('kemasan_id', $kemasan_id) 302 ->where('satuan_id', $satuan_id) 303 ->get(); 304 return response()->json(data: [305 'stoks' => $stoks 306]); 307 } 308 309 1 reference	0 overrides 310 public function get_stok_by_kondisi_kemasan_satuan_jenis(Request $request): JsonResponse	mixed 311 { 312 $kondisi_id = $request->get('kondisi_id'); 313 $kemasan_id = $request->get('kemasan_id'); 314 $satuan_id = $request->get('satuan_id'); 315 $jenis_id = $request->get('jenis_id'); 316 $stoks = StokModel::where('kondisi_id', $kondisi_id) 317 ->where('kemasan_id', $kemasan_id) 318 ->where('satuan_id', $satuan_id) 319 ->where('jenis_id', $jenis_id) 320 ->get(); 321 return response()->json(data: [322 'stoks' => $stoks 323]); 324 } 325 326 1 reference	0 overrides 327 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi(Request $request): JsonResponse	mixed 328 { 329 $kondisi_id = $request->get('kondisi_id'); 330 $kemasan_id = $request->get('kemasan_id'); 331 $satuan_id = $request->get('satuan_id'); 332 $jenis_id = $request->get('jenis_id'); 333 $lokasi_id = $request->get('lokasi_id'); 334 $stoks = StokModel::where('kondisi_id', $kondisi_id) 335 ->where('kemasan_id', $kemasan_id) 336 ->where('satuan_id', $satuan_id) 337 ->where('jenis_id', $jenis_id) 338 ->where('lokasi_id', $lokasi_id) 339 ->get(); 340 return response()->json(data: [341 'stoks' => $stoks 342]); 343 } 344 345 1 reference	0 overrides 346 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis(Request $request): JsonResponse	mixed 347 { 348 $kondisi_id = $request->get('kondisi_id'); 349 $kemasan_id = $request->get('kemasan_id'); 350 $satuan_id = $request->get('satuan_id'); 351 $jenis_id = $request->get('jenis_id'); 352 $lokasi_id = $request->get('lokasi_id'); 353 $stoks = StokModel::where('kondisi_id', $kondisi_id) 354 ->where('kemasan_id', $kemasan_id) 355 ->where('satuan_id', $satuan_id) 356 ->where('jenis_id', $jenis_id) 357 ->where('lokasi_id', $lokasi_id) 358 ->get(); 359 return response()->json(data: [360 'stoks' => $stoks 361]); 362 } 363 364 1 reference	0 overrides 365 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi(Request $request): JsonResponse	mixed 366 { 367 $kondisi_id = $request->get('kondisi_id'); 368 $kemasan_id = $request->get('kemasan_id'); 369 $satuan_id = $request->get('satuan_id'); 370 $jenis_id = $request->get('jenis_id'); 371 $lokasi_id = $request->get('lokasi_id'); 372 $kondisi_id2 = $request->get('kondisi_id2'); 373 $stoks = StokModel::where('kondisi_id', $kondisi_id) 374 ->where('kemasan_id', $kemasan_id) 375 ->where('satuan_id', $satuan_id) 376 ->where('jenis_id', $jenis_id) 377 ->where('lokasi_id', $lokasi_id) 378 ->where('kondisi_id', $kondisi_id2) 379 ->get(); 380 return response()->json(data: [381 'stoks' => $stoks 382]); 383 } 384 385 1 reference	0 overrides 386 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi(Request $request): JsonResponse	mixed 387 { 388 $kondisi_id = $request->get('kondisi_id'); 389 $kemasan_id = $request->get('kemasan_id'); 390 $satuan_id = $request->get('satuan_id'); 391 $jenis_id = $request->get('jenis_id'); 392 $lokasi_id = $request->get('lokasi_id'); 393 $kondisi_id2 = $request->get('kondisi_id2'); 394 $kondisi_id3 = $request->get('kondisi_id3'); 395 $stoks = StokModel::where('kondisi_id', $kondisi_id) 396 ->where('kemasan_id', $kemasan_id) 397 ->where('satuan_id', $satuan_id) 398 ->where('jenis_id', $jenis_id) 399 ->where('lokasi_id', $lokasi_id) 400 ->where('kondisi_id', $kondisi_id2) 401 ->where('kondisi_id', $kondisi_id3) 402 ->get(); 403 return response()->json(data: [404 'stoks' => $stoks 405]); 406 } 407 408 1 reference	0 overrides 409 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 410 { 411 $kondisi_id = $request->get('kondisi_id'); 412 $kemasan_id = $request->get('kemasan_id'); 413 $satuan_id = $request->get('satuan_id'); 414 $jenis_id = $request->get('jenis_id'); 415 $lokasi_id = $request->get('lokasi_id'); 416 $kondisi_id2 = $request->get('kondisi_id2'); 417 $kondisi_id3 = $request->get('kondisi_id3'); 418 $kondisi_id4 = $request->get('kondisi_id4'); 419 $stoks = StokModel::where('kondisi_id', $kondisi_id) 420 ->where('kemasan_id', $kemasan_id) 421 ->where('satuan_id', $satuan_id) 422 ->where('jenis_id', $jenis_id) 423 ->where('lokasi_id', $lokasi_id) 424 ->where('kondisi_id', $kondisi_id2) 425 ->where('kondisi_id', $kondisi_id3) 426 ->where('kondisi_id', $kondisi_id4) 427 ->get(); 428 return response()->json(data: [429 'stoks' => $stoks 430]); 431 } 432 433 1 reference	0 overrides 434 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 435 { 436 $kondisi_id = $request->get('kondisi_id'); 437 $kemasan_id = $request->get('kemasan_id'); 438 $satuan_id = $request->get('satuan_id'); 439 $jenis_id = $request->get('jenis_id'); 440 $lokasi_id = $request->get('lokasi_id'); 441 $kondisi_id2 = $request->get('kondisi_id2'); 442 $kondisi_id3 = $request->get('kondisi_id3'); 443 $kondisi_id4 = $request->get('kondisi_id4'); 444 $kondisi_id5 = $request->get('kondisi_id5'); 445 $stoks = StokModel::where('kondisi_id', $kondisi_id) 446 ->where('kemasan_id', $kemasan_id) 447 ->where('satuan_id', $satuan_id) 448 ->where('jenis_id', $jenis_id) 449 ->where('lokasi_id', $lokasi_id) 450 ->where('kondisi_id', $kondisi_id2) 451 ->where('kondisi_id', $kondisi_id3) 452 ->where('kondisi_id', $kondisi_id4) 453 ->where('kondisi_id', $kondisi_id5) 454 ->get(); 455 return response()->json(data: [456 'stoks' => $stoks 457]); 458 } 459 460 1 reference	0 overrides 461 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 462 { 463 $kondisi_id = $request->get('kondisi_id'); 464 $kemasan_id = $request->get('kemasan_id'); 465 $satuan_id = $request->get('satuan_id'); 466 $jenis_id = $request->get('jenis_id'); 467 $lokasi_id = $request->get('lokasi_id'); 468 $kondisi_id2 = $request->get('kondisi_id2'); 469 $kondisi_id3 = $request->get('kondisi_id3'); 470 $kondisi_id4 = $request->get('kondisi_id4'); 471 $kondisi_id5 = $request->get('kondisi_id5'); 472 $kondisi_id6 = $request->get('kondisi_id6'); 473 $stoks = StokModel::where('kondisi_id', $kondisi_id) 474 ->where('kemasan_id', $kemasan_id) 475 ->where('satuan_id', $satuan_id) 476 ->where('jenis_id', $jenis_id) 477 ->where('lokasi_id', $lokasi_id) 478 ->where('kondisi_id', $kondisi_id2) 479 ->where('kondisi_id', $kondisi_id3) 480 ->where('kondisi_id', $kondisi_id4) 481 ->where('kondisi_id', $kondisi_id5) 482 ->where('kondisi_id', $kondisi_id6) 483 ->get(); 484 return response()->json(data: [485 'stoks' => $stoks 486]); 487 } 488 489 1 reference	0 overrides 490 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 491 { 492 $kondisi_id = $request->get('kondisi_id'); 493 $kemasan_id = $request->get('kemasan_id'); 494 $satuan_id = $request->get('satuan_id'); 495 $jenis_id = $request->get('jenis_id'); 496 $lokasi_id = $request->get('lokasi_id'); 497 $kondisi_id2 = $request->get('kondisi_id2'); 498 $kondisi_id3 = $request->get('kondisi_id3'); 499 $kondisi_id4 = $request->get('kondisi_id4'); 500 $kondisi_id5 = $request->get('kondisi_id5'); 501 $kondisi_id6 = $request->get('kondisi_id6'); 502 $kondisi_id7 = $request->get('kondisi_id7'); 503 $stoks = StokModel::where('kondisi_id', $kondisi_id) 504 ->where('kemasan_id', $kemasan_id) 505 ->where('satuan_id', $satuan_id) 506 ->where('jenis_id', $jenis_id) 507 ->where('lokasi_id', $lokasi_id) 508 ->where('kondisi_id', $kondisi_id2) 509 ->where('kondisi_id', $kondisi_id3) 510 ->where('kondisi_id', $kondisi_id4) 511 ->where('kondisi_id', $kondisi_id5) 512 ->where('kondisi_id', $kondisi_id6) 513 ->where('kondisi_id', $kondisi_id7) 514 ->get(); 515 return response()->json(data: [516 'stoks' => $stoks 517]); 518 } 519 520 1 reference	0 overrides 521 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 522 { 523 $kondisi_id = $request->get('kondisi_id'); 524 $kemasan_id = $request->get('kemasan_id'); 525 $satuan_id = $request->get('satuan_id'); 526 $jenis_id = $request->get('jenis_id'); 527 $lokasi_id = $request->get('lokasi_id'); 528 $kondisi_id2 = $request->get('kondisi_id2'); 529 $kondisi_id3 = $request->get('kondisi_id3'); 530 $kondisi_id4 = $request->get('kondisi_id4'); 531 $kondisi_id5 = $request->get('kondisi_id5'); 532 $kondisi_id6 = $request->get('kondisi_id6'); 533 $kondisi_id7 = $request->get('kondisi_id7'); 534 $kondisi_id8 = $request->get('kondisi_id8'); 535 $stoks = StokModel::where('kondisi_id', $kondisi_id) 536 ->where('kemasan_id', $kemasan_id) 537 ->where('satuan_id', $satuan_id) 538 ->where('jenis_id', $jenis_id) 539 ->where('lokasi_id', $lokasi_id) 540 ->where('kondisi_id', $kondisi_id2) 541 ->where('kondisi_id', $kondisi_id3) 542 ->where('kondisi_id', $kondisi_id4) 543 ->where('kondisi_id', $kondisi_id5) 544 ->where('kondisi_id', $kondisi_id6) 545 ->where('kondisi_id', $kondisi_id7) 546 ->where('kondisi_id', $kondisi_id8) 547 ->get(); 548 return response()->json(data: [549 'stoks' => $stoks 550]); 551 } 552 553 1 reference	0 overrides 554 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 555 { 556 $kondisi_id = $request->get('kondisi_id'); 557 $kemasan_id = $request->get('kemasan_id'); 558 $satuan_id = $request->get('satuan_id'); 559 $jenis_id = $request->get('jenis_id'); 560 $lokasi_id = $request->get('lokasi_id'); 561 $kondisi_id2 = $request->get('kondisi_id2'); 562 $kondisi_id3 = $request->get('kondisi_id3'); 563 $kondisi_id4 = $request->get('kondisi_id4'); 564 $kondisi_id5 = $request->get('kondisi_id5'); 565 $kondisi_id6 = $request->get('kondisi_id6'); 566 $kondisi_id7 = $request->get('kondisi_id7'); 567 $kondisi_id8 = $request->get('kondisi_id8'); 568 $kondisi_id9 = $request->get('kondisi_id9'); 569 $stoks = StokModel::where('kondisi_id', $kondisi_id) 570 ->where('kemasan_id', $kemasan_id) 571 ->where('satuan_id', $satuan_id) 572 ->where('jenis_id', $jenis_id) 573 ->where('lokasi_id', $lokasi_id) 574 ->where('kondisi_id', $kondisi_id2) 575 ->where('kondisi_id', $kondisi_id3) 576 ->where('kondisi_id', $kondisi_id4) 577 ->where('kondisi_id', $kondisi_id5) 578 ->where('kondisi_id', $kondisi_id6) 579 ->where('kondisi_id', $kondisi_id7) 580 ->where('kondisi_id', $kondisi_id8) 581 ->where('kondisi_id', $kondisi_id9) 582 ->get(); 583 return response()->json(data: [584 'stoks' => $stoks 585]); 586 } 587 588 1 reference	0 overrides 589 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 590 { 591 $kondisi_id = $request->get('kondisi_id'); 592 $kemasan_id = $request->get('kemasan_id'); 593 $satuan_id = $request->get('satuan_id'); 594 $jenis_id = $request->get('jenis_id'); 595 $lokasi_id = $request->get('lokasi_id'); 596 $kondisi_id2 = $request->get('kondisi_id2'); 597 $kondisi_id3 = $request->get('kondisi_id3'); 598 $kondisi_id4 = $request->get('kondisi_id4'); 599 $kondisi_id5 = $request->get('kondisi_id5'); 600 $kondisi_id6 = $request->get('kondisi_id6'); 601 $kondisi_id7 = $request->get('kondisi_id7'); 602 $kondisi_id8 = $request->get('kondisi_id8'); 603 $kondisi_id9 = $request->get('kondisi_id9'); 604 $kondisi_id10 = $request->get('kondisi_id10'); 605 $stoks = StokModel::where('kondisi_id', $kondisi_id) 606 ->where('kemasan_id', $kemasan_id) 607 ->where('satuan_id', $satuan_id) 608 ->where('jenis_id', $jenis_id) 609 ->where('lokasi_id', $lokasi_id) 610 ->where('kondisi_id', $kondisi_id2) 611 ->where('kondisi_id', $kondisi_id3) 612 ->where('kondisi_id', $kondisi_id4) 613 ->where('kondisi_id', $kondisi_id5) 614 ->where('kondisi_id', $kondisi_id6) 615 ->where('kondisi_id', $kondisi_id7) 616 ->where('kondisi_id', $kondisi_id8) 617 ->where('kondisi_id', $kondisi_id9) 618 ->where('kondisi_id', $kondisi_id10) 619 ->get(); 620 return response()->json(data: [621 'stoks' => $stoks 622]); 623 } 624 625 1 reference	0 overrides 626 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 627 { 628 $kondisi_id = $request->get('kondisi_id'); 629 $kemasan_id = $request->get('kemasan_id'); 630 $satuan_id = $request->get('satuan_id'); 631 $jenis_id = $request->get('jenis_id'); 632 $lokasi_id = $request->get('lokasi_id'); 633 $kondisi_id2 = $request->get('kondisi_id2'); 634 $kondisi_id3 = $request->get('kondisi_id3'); 635 $kondisi_id4 = $request->get('kondisi_id4'); 636 $kondisi_id5 = $request->get('kondisi_id5'); 637 $kondisi_id6 = $request->get('kondisi_id6'); 638 $kondisi_id7 = $request->get('kondisi_id7'); 639 $kondisi_id8 = $request->get('kondisi_id8'); 640 $kondisi_id9 = $request->get('kondisi_id9'); 641 $kondisi_id10 = $request->get('kondisi_id10'); 642 $kondisi_id11 = $request->get('kondisi_id11'); 643 $stoks = StokModel::where('kondisi_id', $kondisi_id) 644 ->where('kemasan_id', $kemasan_id) 645 ->where('satuan_id', $satuan_id) 646 ->where('jenis_id', $jenis_id) 647 ->where('lokasi_id', $lokasi_id) 648 ->where('kondisi_id', $kondisi_id2) 649 ->where('kondisi_id', $kondisi_id3) 650 ->where('kondisi_id', $kondisi_id4) 651 ->where('kondisi_id', $kondisi_id5) 652 ->where('kondisi_id', $kondisi_id6) 653 ->where('kondisi_id', $kondisi_id7) 654 ->where('kondisi_id', $kondisi_id8) 655 ->where('kondisi_id', $kondisi_id9) 656 ->where('kondisi_id', $kondisi_id10) 657 ->where('kondisi_id', $kondisi_id11) 658 ->get(); 659 return response()->json(data: [660 'stoks' => $stoks 661]); 662 } 663 664 1 reference	0 overrides 665 public function get_stok_by_kondisi_kemasan_satuan_jenis_lokasi_jenis_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi_kondisi(Request $request): JsonResponse	mixed 666 { 667 $kondisi_id = $request->get('kondisi_id'); 668 $kemasan_id = $request->get('kemasan_id'); 669 $satuan_id = $request->get('satuan_id'); 670 $jenis_id = $request->get('jenis_id'); 671 $lokasi_id = $request->get('lokasi_id'); 672 $kondisi_id2 = $request->get('kondisi_id2'); 673 $kondisi_id3 = $request->get('kondisi_id3'); 674 $kondisi_id4 = $request->get('kondisi_id4'); 675 $kondisi_id5 = $request->get('kondisi_id5'); 676 $kondisi_id6 = $request->get('kondisi_id6'); 677 $kondisi_id7 = $request->get('kondisi_id7'); 678 $kondisi_id8 = $request->get('kondisi_id8'); 679 $kondisi_id9 = $request->get('kondisi_id9'); 680 $kondisi_id10 = $request->get('kondisi_id10'); 681 $kondisi_id11 = $request->get('kondisi_id11'); 682 $kondisi_id12 = $request->get('kondisi_id12'); 683 $stoks = StokModel::where('kondisi_id', $kondisi_id) 684 ->where('kemasan_id', $kemasan_id) 685 ->where('satuan_id', $satuan_id) 686 ->where('jenis_id', $jenis_id) 687 ->where('lokasi_id', $lokasi_id) 688 ->where('kondisi_id', $kondisi_id2) 689 ->where('kondisi_id', $kondisi_id3) 690 ->where('kondisi_id', $kondisi_id4) 691 ->where('kondisi_id', $kondisi_id5) 692 ->where('kondisi_id', $kondisi_id6) 693 ->where('kondisi_id', $kondisi_id7) 694 ->where('kondisi_id', $kondisi_id8) 695 ->where('kondisi_id', $kondisi_id9) 696 ->where('kondisi_id', ```



```
resources > views > supplier > create_javascript.php < form-form-tambah > div modal modal-dialog modal-lg > div modal-content > div modal-body > div form-group
1  <form action="{{ url('/stok/ajax') }}" method="POST" id="form-tambah">
2    @csrf
3    <div id="modal-master" class="modal-dialog modal-lg" role="document">
4      <div class="modal-content">
5        <div class="modal-header">
6          <h5 class="modal-title" id="exampleModalLabel">Tambah Data supplier</h5>
7          <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>
8        </div>
9        <div class="modal-body">
10          <div class="form-group">
11            <label>Supplier Kode</label>
12            <input value="" type="text" name="supplier_kode" id="supplier_kode" class="form-control" required>
13            <small id="error-supplier_kode" class="error-text form-text text-danger"></small>
14          </div>
15          <div class="form-group">
16            <label>Nama supplier</label>
17            <input value="" type="text" name="supplier_nama" id="supplier_nama" class="form-control" required>
18            <small id="error-supplier_nama" class="error-text form-text text-danger"></small>
19          </div>
20          <div class="form-group">
21            <label>Alamat supplier</label>
22            <input value="" type="text" name="supplier_alamat" id="supplier_alamat" class="form-control" required>
23            <small id="error-supplier_alamat" class="error-text form-text text-danger"></small>
24          </div>
25        </div>
26        <div class="modal-footer">
27          <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
28          <button type="submit" class="btn btn-primary">Simpan</button>
29        </div>
30      </div>
31    </div>
32  </form>
33  <script>
34    $(document).ready(function() {
35      $("#form-tambah").validate({
36        rules: {
37          supplier_kode: {
38            required: true,
39            minlength: 1,
40            maxlength: 20
41          },
42          supplier_nama: {
43            required: true,
44            minlength: 3,
45            maxlength: 100
46          },
47          supplier_alamat: {
48            required: true,
49            minlength: 3,
50            maxlength: 100
51          }
52        },
53        submitHandler: function(form) {
54          $.ajax({
55            url: form.action,
56            type: form.method,
57            data: $(form).serialize(),
58            success: function(response) {
59              if (response.status) {
60                $('#myModal').modal('hide');
61                Swal.fire({
62                  icon: 'success',
63                  title: 'Berhasil',
64                  text: response.message
65                });
66                dataStok.ajax.reload();
67              } else {
68                $('.error-text').text('');
69                $.each(response.msgField, function(prefix, val) {
70                  $('#' + prefix + val[0]).text(val[1]);
71                });
72                Swal.fire({
73                  icon: 'error',
74                  title: 'Terjadi Kesalahan',
75                  text: response.message
76                });
77              }
78            }
79          });
80          return false;
81        },
82        errorElement: 'span',
83        errorPlacement: function(error, element) {
84          error.addClass('invalid-feedback');
85          element.closest('.form-group').append(error);
86        },
87        highlight: function(element, errorClass, validClass) {
88          $(element).addClass('is-invalid');
89        },
90        unhighlight: function(element, errorClass, validClass) {
91          $(element).removeClass('is-invalid');
92        }
93      });
94    });
95  </script>
```



- 5 Berikut adalah tampilan tambah data ajax supplier ketika di akses.

The screenshot shows a modal dialog titled 'Tambah Data supplier'. Inside the dialog, there are two input fields: 'Supplier Kode' containing 'neya' and 'Nama supplier' containing 'Neya Supplier'. Below the dialog is a table with four entries:

ID	Kode Supplier	Nama Supplier	Alamat supplier
1	yol	Liya Supplier	Jln Tuban no 65
2	snd	Nela Supplier	Jln yerusalem No.4
3	ily		
4	nels		

At the bottom right of the dialog are 'Batal' and 'Simpan' buttons. The main page has a 'Tambah' and 'Tambah Ajax' button at the top right.

The screenshot shows a modal dialog with a green checkmark icon and the text 'Berhasil' (Successful). Below it, it says 'Data level berhasil disimpan' (Data level successfully saved) and has an 'OK' button. This dialog is overlaid on a table of suppliers. The table has five entries:

ID	Kode Supplier	Nama Supplier	Alamat supplier
1	yol	Liya Supplier	Jln Tuban no 65
2	snd	Nela Supplier	Jln yerusalem No.4
3	ily		
4	nels		
5	neya	Neya Supplier	Jalan Bandung No.56

At the bottom right of the dialog are 'Detail', 'Edit', and 'Hapus' buttons. The main page has a 'Tambah' and 'Tambah Ajax' button at the top right.

Gambar di atas adalah tampilan halaman penambahan supplier yang mana hanya perlu memasukan kode supplier dan nama supplier, kemudian klik simpan untuk menyimpan penambahan data maka akan kembali ke halaman list supplier yang akan menampilkan data terupdate dengan centang hijau yang menunjukkan data berhasil diinputkan

- 6 Sekarang membuat form untuk mengedit data, maka diperlukan modifikasi pada file `SupplierController.php` dengan isi fungsi `list()` seperti berikut



```
app > Http > Controllers > 🏠 SupplierController.php > PHP > 🛡 SupplierController > ⚙ list()
```

```
11 class SupplierController extends Controller
30     // Ambil data supplier dalam bentuk JSON untuk datatables
31     1 reference | 0 overrides
32     public function list(Request $request): JsonResponse
33     {
34         // $suppliers = SupplierModel::select('supplier_id', 'supplier_kode', 'supplier_nama', 'supplier_alamat');
35
36         // Filter data supplier berdasarkan supplier_id
37         // if ($request->supplier_id) {
38         //     $suppliers->where('supplier_id', $request->supplier_id);
39         // }
40
41         // return DataTables::of($suppliers)
42         //     ->addIndexColumn()
43         //     ->addColumn('aksi', function ($supplier) { // menambahkan kolom aksi
44             // $btn = '<a href="'. url('/supplier/' . $supplier->supplier_id) . '" class="btn btn-info btn-sm">Detail</a> ';
45             // $btn .= '<a href="'. url('/supplier/' . $supplier->supplier_id) . '/edit' . '" class="btn btn-warning btn-sm">Edit</a> ';
46             // $btn .= '<form class="d-inline-block" method="POST" action="'. url('/supplier/' . $supplier->supplier_id) . '">';
47             //     . csrf_field() . method_field('DELETE') .
48             //     '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\')">' . '<button>Hapus</button>' . '</form>';
49             //     return $btn;
50         // })
51         // ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah HTML
52         // ->make(true);
53
54         $supplier = SupplierModel::select(columns: 'supplier_id', 'supplier_kode', 'supplier_nama', 'supplier_alamat');
55         // Filter data supplier berdasarkan supplier_id
56         // if ($request->supplier_id) {
57         //     $supplier->where('supplier_id', $request->supplier_id);
58         // }
59         return DataTables::of($supplier)
60             ->addIndexColumn() // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
61             ->addColumn('aksi', function ($supplier) { string { // menambahkan kolom aksi
62                 $btn = '<a href="'. url(path: '/stok/' . $supplier->supplier_id) . '" class="btn btn-info btn-sm">Detail</a> ';
63                 $btn .= '<button onclick="modalAction(\'' . url(path: '/stok/' . $supplier->supplier_id . '/edit_ajax') . '\')"' . ' class="btn btn-warning btn-sm">Edit</button> ';
64                 $btn .= '<button onclick="modalAction(\'' . url(path: '/stok/' . $supplier->supplier_id . '/delete_ajax') . '\')"' . ' class="btn btn-danger btn-sm">Hapus</button> ';
65                 return $btn;
66             })
67             ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html
68             ->make(true);
69     }
70 }
```

You have Windows Subsystem for Linux (WSL) installed on your system. Do you want to install the recommended extension from Microsoft for it? ...

- 7 Kemudian membuat file `SupplierController.php`. Dan juga membuat fungsi `edit_ajax()` dan `update_ajax()` untuk mengedit data kategori melalui ajax.

```
app > Http > Controllers > 🏠 SupplierController.php > PHP > 🛡 SupplierController > ⚙ update_ajax()
```

```
11 class SupplierController extends Controller
20     1 reference | 0 overrides
21     public function edit_ajax(string $id): Factory|View
22     {
23         $supplier = SupplierModel::find(id: $id);
24         return view(view: 'stok.edit_ajax', data: ['supplier' => $supplier]);
25     }
26
27     1 reference | 0 overrides
28     public function update_ajax(Request $request, $id): JsonResponse|mixed\Redirector\RedirectResponse
29     {
30         // cek apakah request dari ajax
31         if ($request->ajax() || $request->wantsJson()) {
32             $rules = [
33                 'supplier_kode' => 'required|string|unique:m_supplier,supplier_kode,' . $id . ',supplier_id',
34                 'supplier_nama' => 'required|string|max:100',
35                 'supplier_alamat' => 'nullable|string|max:255'
36             ];
37             // use Illuminate\Support\Facades\Validator;
38             $validator = Validator::make(data: $request->all(), rules: $rules);
39             if ($validator->fails()) {
40                 return response()->json(data: [
41                     'status' => false, // respon json, true: berhasil, false: gagal
42                     'message' => 'Validasi gagal.',
43                     'msgField' => $validator->errors() // menunjukkan field mana yang error
44                 ]);
45             }
46             $check = SupplierModel::find(id: $id);
47             if ($check) {
48                 $check->update(attributes: $request->all());
49                 return response()->json(data: [
50                     'status' => true,
51                     'message' => 'Data berhasil diupdate'
52                 ]);
53             } else {
54                 return response()->json(data: [
55                     'status' => false,
56                     'message' => 'Data tidak ditemukan'
57                 ]);
58             }
59         }
60     }
61 }
```



8

Selanjutnya membuat form untuk mengedit data, maka diperlukan file PWL/resources/views/supplier/edit_ajax.blade.php seperti berikut.

```
resources > views > supplier > edit_ajax.blade.php @<div><modal-extrinsic><div>
1 <empty>{$Supplier}
2   <div id="modal-edit" class="modal-dialog modal-lg" role="document">
3     <div class="modal-content">
4       <div class="modal-header">
5         <h4 class="modal-title" id="exampleModallabel">Kesalahan!!</h4>
6         <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>
7           &aacute;ria-hidden="true"&lt;/span></button>
8       <div class="modal-body">
9         <div class="alert alert-danger">
10           <div class="list-group-item has-fa-beer"></div> Kesalahan!!</div>
11           Data yang anda cari tidak ditemukan
12         </div>
13         <a href="{{ url('/home') }}" class="btn btn-warning">Kembali</a>
14       </div>
15     </div>
16   </div>
17   <form action="{{ url('/shuk/' . $supplier->supplier_id . '/updateAjax') }}" method="POST" id="form-edit">
18     <input type="hidden" name="supplier_id" value="{{$Supplier->supplier_id}}"/>
19     <input type="hidden" name="method" value="PUT"/>
20     <div id="modal-edit" class="modal-dialog modal-lg" role="document">
21       <div class="modal-content">
22         <div class="modal-header">
23           <h4 class="modal-title" id="exampleModallabel">Edit Data Supplier</h4>
24           <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>
25             &aacute;ria-hidden="true"&lt;/span></button>
26         </div>
27         <div class="modal-body">
28           <div class="form-group">
29             <label>Supplier Kode</label>
30             <input value="{{ $Supplier->supplier_code }}" type="text" name="supplier_code" id="supplier_code"
31               class="form-control" required>
32             <small id="error-supplier_code" class="error-text form-text text-danger"></small>
33           </div>
34           <div class="form-group">
35             <label>Nama supplier</label>
36             <input value="{{ $Supplier->supplier_name }}" type="text" name="supplier_name" id="supplier_name"
37               class="form-control" required>
38             <small id="error-supplier_name" class="error-text form-text text-danger"></small>
39           </div>
40           <div class="form-group">
41             <label>Alamat supplier</label>
42             <input value="{{ $Supplier->supplier_alamat }}" type="text" name="supplier_alamat" id="supplier_alamat"
43               class="form-control" required>
44             <small id="error-supplier_alamat" class="error-text form-text text-danger"></small>
45           </div>
46           <div class="modal-footer">
47             <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
48             <button type="submit" class="btn btn-primary">Simpan</button>
49           </div>
50         </div>
51       </div>
52     </div>
53   </form>
54   <script>
55     $(document).ready(function() {
56       $('#form-edit').validate({
57         rules: {
58           supplier_code: {
59             required: true,
60             minlength: 1,
61             maxlength: 20
62           },
63           supplier_name: {
64             required: true,
65             minlength: 2,
66             maxlength: 100
67           },
68           supplier_alamat: {
69             required: true,
70             minlength: 2,
71             maxlength: 100
72           }
73         },
74         submitHandler: function(form) {
75           $.ajax({
76             url: form.action,
77             type: form.method,
78             data: $form.serialize(),
79             success: function(response) {
80               if (response.status) {
81                 $('#myModal').modal('hide');
82                 Swal.fire(
83                   'Berhasil',
84                   'Terima Kesanlahan',
85                   'success'
86                 );
87                 dataTables.ajax.reload();
88               } else {
89                 $('#error-text').text('');
90                 $.each(response.msgField, function(prefix, val) {
91                   $('#error-' + prefix).text(val[0]);
92                 });
93                 Swal.fire(
94                   'Error',
95                   'Terjadi Kesalahan',
96                   'error'
97                 );
98               }
99             }
100           });
101           return false;
102         },
103         errorElement: 'span',
104         errorPlacement: function(error, element) {
105           error.addClass('invalid-feedback');
106           element.closest('.form-group').append(error);
107         },
108         highlight: function(element, errorClass, validClass) {
109           element.addClass('is-invalid');
110         },
111         unhighlight: function(element, errorClass, validClass) {
112           element.removeClass('is-invalid');
113         }
114       });
115     });
116   </script>
117 </body>
```



Dan berikut adalah tampilan halaman edit supplier.

ID	Kode Supplier	Nama Supplier	Alamat supplier
1	yol	Liya Supplier	Jln Tuban no 65
2	snd	Nela Supplier	Jln yerusalem No.4
3	liy	Neya Supplier	Jalan Bandung No.56
4	nels		
5	neya		

ID	Kode Supplier	Nama Supplier	Alamat supplier
1	yol	Liya Supplier	Jln Tuban no 65
2	snd	Nela Supplier	Jln yerusalem No.4
3	liy	Neya Supplier	Jalan Bandung No.56
4	nels		
5	neya		

- 9 Untuk menampilkan detail data level user maka diperlukan memodifikasi file `SupplierController.php` seperti berikut. Dengan menambahkan fungsi `confirm_ajax()` dan `delete_ajax()`.



```
app > Http > Controllers > 🌐 SupplierController.php > ...
11  class SupplierController extends Controller
12      |
13      |     1 reference | 0 overrides
14  261  public function confirm_ajax(string $id): Factory|View
15  262  {
16  263      $supplier = SupplierModel::find(id: $id);
17  264      return view(view: 'supplier.confirm_ajax', data: ['supplier' => $supplier]);
18  265  }
19  266
20  267      |
21  268      |     1 reference | 0 overrides
22  269  268  public function delete_ajax(Request $request, $id): JsonResponse|mixed\RedirectResponse
23  269  {
24  270      // cek apakah request dari ajax
25  271      if ($request->ajax() || $request->wantsJson()) {
26  272          $supplier = SupplierModel::find(id: $id);
27  273          if ($supplier) {
28  274              $supplier->delete();
29  275              return response()->json(data: [
30  276                  'status' => true,
31  277                  'message' => 'Data berhasil dihapus'
32  278              ]);
33  279          } else {
34  280              return response()->json(data: [
35  281                  'status' => false,
36  282                  'message' => 'Data tidak ditemukan'
37  283              ]);
38  284          }
39  285      }
40  286      return redirect(to: '/');
41  287  }
42  288  }
43  289  
```

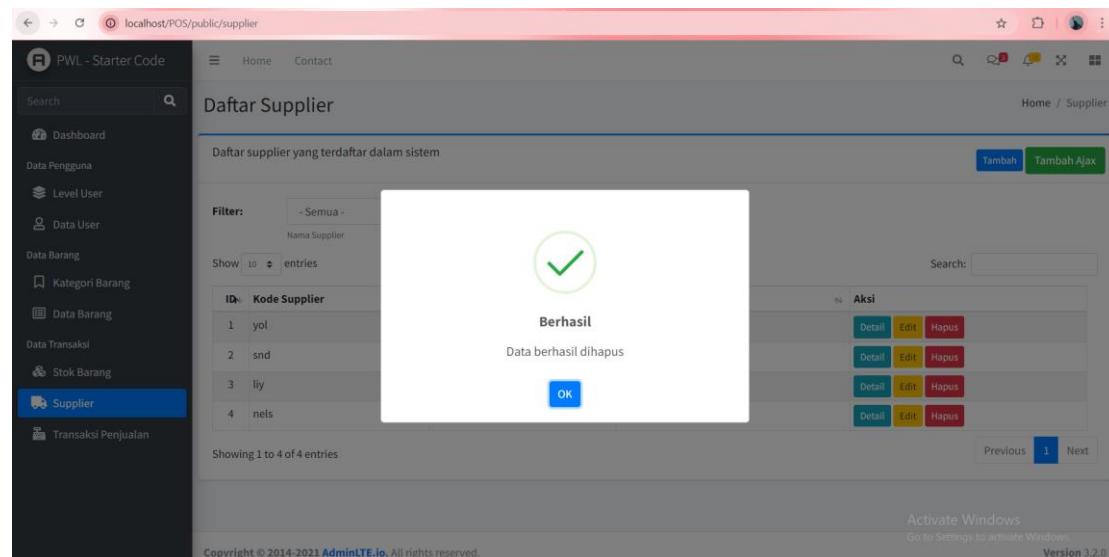
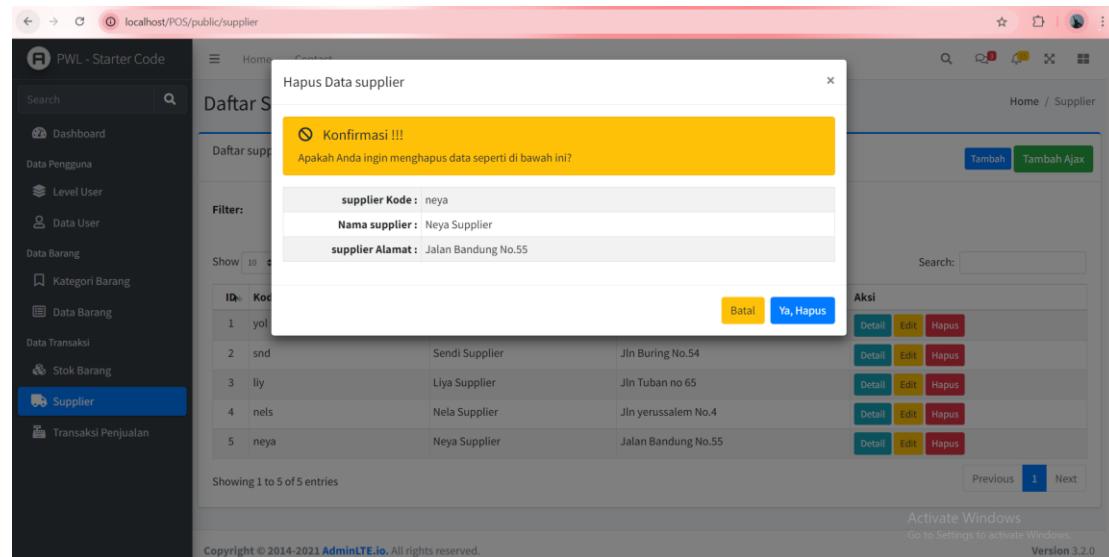
Dan dilanjutkan dengan membuat file view `confirm_ajax.blade.php`.

```
resources > views > supplier > 🌐 confirm_ajax.blade.php > click modal-examples-modal-clicking-existing > 🌐 the modal-content > 🌐 the modal-body
1  <div id="modal-existing" class="modal-dialing modal-lg" role="document">
2      <div class="modal-content">
3          <div class="modal-header">
4              <h4 class="modal-title" id="exampleModalLabel">Kesalahan×
18                 <input type="button" class="modal-dialing modal-lg" role="button" value="Hapus Data supplier" />
19             </form>
20             <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>aria-hidden="true" style="font-size: 1em; font-weight: bold;">×
```



10

Berikut adalah tampilan halaman detail supplier yang mengambil data dari database m_supplier.



Yang mana tampilan halaman nya seperti gambar diatas, yaitu ketika salah satu data dihapus maka akan terdapat pop up konfirmasi penghapusan dan keterangan yang menunjukan berhasil tidaknya data tersebut dihapus.

D. Tabel m_barang

Langkah	Jawaban/Deskripsi
---------	-------------------



1	<p>Pertama saya melakukan penambahan pada route untuk menentukan tampilan menu apa saja yang akan ada terkait tabel <code>m_barang</code>, seperti gambar di bawah ini yang saya samakan dengan tabel <code>m_user</code> dengan memberi tanda nama ajax dari praktikum di atas yaitu menambah data, edit dan hapus data.</p> 
2	Kemudian memodifikasi file View tentang <code>m_barang</code> pada file <code>supplier/index.blade.php</code>





3	<p>Kemudian membuat file BarangController.php. Dan juga membuat fungsi <code>create_ajax()</code> dan <code>store_ajax()</code> untuk menambah data supplier melalui ajax.</p> <pre>app > Http > Controllers > BarangController.php > PHP Intelephense > BarangController > store_ajax 11 class BarangController extends Controller 12 0 references 0 overrides 13 public function create_ajax(): mixed View 14 { 15 \$kategori = KategoriModel::select(columns: 'kategori_id', 'kategori_nama')->get(); 16 return view(view: 'barang.create_ajax') 17 ->with(key: 'kategori', value: \$kategori); 18 } 19 20 0 references 0 overrides 21 public function store_ajax(Request \$request): JsonResponse mixed 22 { 23 // cek apakah request berupa ajax 24 if (\$request->ajax() \$request->wantsJson()) [25 \$rules = [26 'barang_kode' => 'required numeric unique:m_barang,barang_kode', // barang_kode harus unik dan numeric 27 'kategori_id' => 'required numeric max:20', // kategori_id harus diisi, numeric, dan maksimal 20 karakter 28 'barang_nama' => 'required string max:100', // barang_nama harus diisi dan maksimal 100 karakter 29 'harga_beli' => 'required numeric', // harga_beli harus diisi dan berupa angka 30 'harga_jual' => 'required numeric', // harga_jual harus diisi dan berupa angka 31]; 32 // use Illuminate\Support\Facades\Validator; 33 \$validator = Validator::make(data: \$request->all(), rules: \$rules); 34 35 if (\$validator->fails()) 36 return response()->json(data: [37 'status' => false, // response status, false: error/gagal, true: berhasil 38 'message' => 'Validasi Gagal', 39 'msgField' => \$validator->errors(), // pesan error validasi 40]); 41 BarangModel::create(attributes: \$request->all()); 42 return response()->json(data: [43 'status' => true, 44 'message' => 'Data Kategori berhasil disimpan' 45]); 46] 47 redirect(to: '/'); 48 } 49 </pre>
4	<p>Lalu membuat view pada PWL/resources/views/barang/<code>create_ajax.blade.php</code> seperti berikut</p>



```
resources > view > barang > create_javascript.js > ...
```

```
1  form action="{{ url('/barang/ajax') }}" method="POST" id="form-tambah">
2  @csrf
3  <div id="modal-master" class="modal-dialog modal-lg" role="document">
4    <div class="modal-content">
5      <div class="modal-header">
6        <h5>Tambah Data Barang</h5>
7        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8          <span aria-hidden="true">&amptimes</span>
9        </button>
10      </div>
11      <div class="modal-body">
12        <div class="form-group">
13          <label>Kode Barang</label>
14          <input value="" type="text" name="barang_kode" id="barang_kode" class="form-control" required>
15          <small id="error-barang_kode" class="error-text form-text text-danger"></small>
16        </div>
17        <div class="form-group">
18          <label>Nama Barang</label>
19          <input value="" type="text" name="barang_nama" id="barang_nama" class="form-control" required>
20          <small id="error-barang_nama" class="error-text form-text text-danger"></small>
21        </div>
22        <div class="form-group">
23          <label>Harga Beli</label>
24          <input value="" type="number" name="harga_beli" id="harga_beli" class="form-control" required>
25          <small id="error-harga_beli" class="error-text form-text text-danger"></small>
26        </div>
27        <div class="form-group">
28          <label>Harga Jual</label>
29          <input value="" type="number" name="harga_jual" id="harga_jual" class="form-control" required>
30          <small id="error-harga_jual" class="error-text form-text text-danger"></small>
31        </div>
32        <div class="form-group">
33          <label>Kategori Barang</label>
34          <select name="kategori_id" id="kategori_id" class="form-control" required>
35            <option value=""><Pilih Kategori</option>
36            @foreach($kategori as $item)
37              <option value="{{ $item->kategori_id }}>{{ $item->kategori_nama }}</option>
38            @endforeach
39          </select>
40          <small id="error-kategori_id" class="error-text form-text text-danger"></small>
41        </div>
42        <div class="modal-footer">
43          <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
44          <button type="submit" class="btn btn-primary">Simpan</button>
45        </div>
46      </div>
47    </div>
48  </form>
49  <script>
50    $(document).ready(function() {
51      $("#form-tambah").validate({
52        rules: {
53          barang_kode: {
54            required: true,
55            minlength: 1,
56            maxlength: 20
57          },
58          barang_nama: {
59            required: true,
60            minlength: 3,
61            maxlength: 100
62          },
63          harga_beli: {
64            required: true,
65            number: true
66          },
67          harga_jual: {
68            required: true,
69            number: true
70          },
71          kategori_id: {
72            required: true
73          }
74        },
75        submitHandler: function(form) {
76          $.ajax({
77            url: form.action,
78            type: form.method,
79            data: $(form).serialize(),
80            success: function(response) {
81              if (response.status) {
82                $('#myModal').modal('hide');
83                Swal.fire({
84                  icon: 'success',
85                  title: 'Berhasil',
86                  text: response.message
87                });
88                databarang.ajax.reload(); // Ganti 'databarang' dengan 'datalbarang'
89              } else {
90                $('#error-text').text('');
91                $.each(response.msgField, function(prefix, val) {
92                  $('#error-' + prefix).text(val[0]);
93                });
94                Swal.fire({
95                  icon: 'error',
96                  title: 'Terjadi Kesalahan',
97                  text: response.message
98                });
99              }
100            }
101          });
102          return false;
103        },
104        errorElement: 'span',
105        errorPlacement: function(error, element) {
106          error.addClass('invalid-feedback');
107          element.closest('.form-group').append(error);
108        },
109        highlight: function(element, errorClass, validClass) {
110          $(element).addClass('is-invalid');
111        },
112        unhighlight: function(element, errorClass, validClass) {
113          $(element).removeClass('is-invalid');
114        }
115      });
116    });
117  </script>
```



5

Berikut adalah tampilan tambah data ajax barang ketika di akses.

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	nas	Nasi	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
2	aym	Ayam	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
3	saw	Sawi	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
4	pmg	Promag	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
5	scr	Sumscreen	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
6	dtx	Dulcolax	Rp 8.000	Rp 10.000	Beauty and Health	[Detail] [Edit] [Hapus]
7	myt	Minyak Telon	Rp 15.000	Rp 19.000	Baby and Kid	[Detail] [Edit] [Hapus]
8	pop	Popok Bayi	Rp 8.000	Rp 20.000	Baby and Kid	[Detail] [Edit] [Hapus]
9	bdk	Bedak Bayi	Rp 8.000	Rp 10.000	Baby and Kid	[Detail] [Edit] [Hapus]
10	krs	Kursi Plastik	Rp 20.000	Rp 22.000	Home and Care	[Detail] [Edit] [Hapus]
11	pel	Pel	Rp 8.000	Rp 10.000	Home and Care	[Detail] [Edit] [Hapus]
12	sap	Sapu	Rp 8.000	Rp 10.000	Home and Care	[Detail] [Edit] [Hapus]

Berhasil

Data Kategori berhasil disimpan

OK

Gambar di atas adalah tampilan halaman penambahan barang yang mana hanya perlu memasukan kode barang dan nama barang, kemudian klik simpan untuk menyimpan penambahan data maka akan kembali ke halaman list barang yang akan menampilkan data terupdate dengan centang hijau yang menunjukan data berhasil diinputkan

6

Sekarang membuat form untuk mengedit data, maka diperlukan modifikasi pada file `BarangController.php` dengan isi fungsi `list()` seperti berikut



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<https://jti.polinema.ac.id>

```
app > Http > Controllers > BarangController.php > PHP > BarangController > update_ajax()
11 class BarangController extends Controller
12 {
13     // reference | 0 overrides
14
15     public function list(Request $request): JsonResponse
16     {
17         $barang = barangModel::select('barang_id', 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
18             ->with(['relations' : 'kategori']);
19
20         // if($request->kategori_id){
21         //     $barang->where('kategori_id', $request->kategori_id);
22         // }
23
24         // return DataTables::of($barang)
25         //     // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
26         //     ->addIndexColumn()
27         //     ->addColumn('aksi', function ($barang) { // menambahkan kolom aksi
28             $btn = '<a href="'. url('/barang/' . $barang->barang_id) . '" class="btn btn-info btn-sm">Detail</a> ';
29             $btn .= '<a href="'. url('/barang/' . $barang->barang_id . '/edit') . '" class="btn btn-warning btn-sm">Edit</a> ';
30             $btn .= '<form class="d-inline-block" method="POST" action="'. url('/barang/' . $barang->barang_id) . '>' .
31                 csrf_field() . method_field('DELETE') .
32                 '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'' . Apakah Anda yakin menghapus data ini?\')";>Hapus</button></form>';
33             return $btn;
34         })
35         // ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah HTML
36         // ->make(true);
37
38         return DataTables::of($barang)
39             ->addIndexColumn() // Menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
40             ->addColumn('aksi', function ($barang) { string // Menambahkan kolom aksi
41                 $btn = '<a href="'. url(path: '/barang/' . $barang->barang_id) . '" class="btn btn-info btn-sm">Detail</a> ';
42                 $btn .= '<button onclick="modalAction(' . url(path: '/barang/' . $barang->barang_id . '/edit_ajax') . ')" class="btn btn-warning btn-sm">Edit</button> ';
43                 $btn .= '<button onclick="modalAction(' . url(path: '/barang/' . $barang->barang_id . '/delete_ajax') . ')" class="btn btn-danger btn-sm">Hapus</button> ';
44             return $btn;
45         })
46         // ->rawColumns(['aksi']) // Memberitahu bahwa kolom aksi adalah HTML
47         // ->make(true);
48     }
49 }
```

- 7 Kemudian membuat file BarangController.php. Dan juga membuat fungsi edit_ajax() dan update_ajax() untuk mengedit data kategori melalui ajax.

```
app > Http > Controllers > BarangController.php > PHP > BarangController > update_ajax()
11 class BarangController extends Controller
12 {
13     // reference | 0 overrides
14
15     public function edit_ajax(string $id): Factory|View
16     {
17         $barang = BarangModel::find(id: $id);
18         $kategori = KategoriModel::all(); // Untuk dropdown kategori
19         return view(view: 'barang.edit_ajax', data: ['barang' => $barang, 'kategori' => $kategori]);
20     }
21
22     // reference | 0 overrides
23     public function update_ajax(Request $request, string $id): JsonResponse|mixed\Redirector\RedirectResponse
24     {
25         // cek apakah request dari ajax
26         if ($request->ajax() || $request->wantsJson()) {
27             $rules = [
28                 'kategori_id' => 'required|integer',
29                 'barang_kode' => '_required|max:20|unique:m_barang,barang_kode,' . $id . ',barang_id',
30                 'barang_nama' => 'required|max:100',
31                 'harga_beli' => 'required|integer',
32                 'harga_jual' => 'required|integer'
33             ];
34
35             // use Illuminate\Support\Facades\Validator;
36             $validator = Validator::make(data: $request->all(), rules: $rules);
37
38             if ($validator->fails()) {
39                 return response()->json(data: [
40                     'status' => false, // respon json, true: berhasil, false: gagal
41                     'message' => 'Validasi gagal.',
42                     'msgField' => $validator->errors() // menunjukkan field mana yang error
43                 ]);
44             }
45
46             $barang = BarangModel::find(id: $id);
47             if ($barang) {
48                 $barang->update(attributes: $request->all());
49                 return response()->json(data: [
50                     'status' => true,
51                     'message' => 'Data barang berhasil diupdate'
52                 ]);
53             } else {
54                 return response()->json(data: [
55                     'status' => false,
56                     'message' => 'Data barang tidak ditemukan'
57                 ]);
58             }
59
60             return redirect(to: '/'); // Jika bukan Ajax, redirect ke halaman utama
61         }
62     }
63 }
```



8

Selanjutnya membuat form untuk mengedit data, maka diperlukan file PWL/resources/views/barang/edit ajax.blade.php seperti berikut.

```
resources > views > barang > edit.js?body=php > div>modal-master.modal-dialog.modal-lg > script
2   <div id="modal-master" class="modal-dialog modal-lg" role="document">
3     <script>
4       $(document).ready(function() {
5         $("#form-edit").validate({
6           rules: {
7             kategori_id: {
8               required: true,
9               number: true
10            },
11            barang_kode: {
12              required: true,
13              minlength: 1,
14              maxlength: 20
15            },
16            barang_nama: {
17              required: true,
18              minlength: 3,
19              maxlength: 100
20            },
21            harga_beli: {
22              required: true,
23              minlength: 3
24            },
25            harga_jual: {
26              required: true,
27              minlength: 3,
28            }
29          },
30          submitHandler: function(form) {
31            $.ajax({
32              url: form.action,
33              type: form.method,
34              data: $(form).serialize(),
35              success: function(response) {
36                if (response.status) {
37                  $('#myModal').modal('hide');
38                  Swal.fire({
39                    icon: 'success',
40                    title: 'Berhasil',
41                    text: response.message
42                  });
43                  dataBarang.ajax.reload();
44                } else {
45                  $('.error-text').text('');
46                  $.each(response.msgfield, function(prefix, val) {
47                    $('.error-' + prefix).text(val[0]);
48                  });
49                  Swal.fire({
50                    icon: 'error',
51                    title: 'Terjadi Kesalahan',
52                    text: response.message
53                  });
54                }
55              });
56            return false;
57          },
58          errorElement: 'span',
59          errorPlacement: function(error, element) {
60            error.addClass('invalid-feedback');
61            element.closest('.form-group').append(error);
62          },
63          highlight: function(element, errorClass, validClass) {
64            $(element).addClass('is-invalid');
65          },
66          unhighlight: function(element, errorClass, validClass) {
67            $(element).removeClass('is-invalid');
68          }
69        });
70      });
71    </script>
72  </div>
```

Dan berikut adalah tampilan halaman edit supplier.

The screenshot shows a web-based application interface. On the left, there's a sidebar with various menu items: Dashboard, Data Pengguna, Level User, Data User, Data Barang, Kategori Barang, Data Barang (which is currently selected and highlighted in blue), Data Transaksi, Stok Barang, Supplier, and Transaksi Penjualan. The main content area has a header "Daftar Barang" and a sub-header "Edit Data barang". Below this, there are two sections: "kategori Pengguna" with a dropdown menu showing "Food and Beverage" and "Barang Kode" with a dropdown menu showing "22". A "Filter:" section is also present. The main form contains fields for "Nama Barang" (WithValue: "Sawi"), "Harga Beli" (WithValue: "5000"), and "Harga Jual" (WithValue: "5500"). To the right of the form is a table titled "Aksi" with several rows, each containing "Detail", "Edit", and "Hapus" buttons. At the bottom of the form are "Batal" and "Simpan" buttons. The footer of the page includes copyright information ("Copyright © 2014-2021 AdminLTE.io. All rights reserved."), a Windows activation notice ("Activate Windows Go to Settings to activate Windows"), and a footer link ("Version 3.2.0").



The screenshot shows a web application interface titled 'Daftar Barang'. On the left, there's a sidebar with navigation links like 'Dashboard', 'Data Pengguna', 'Level User', 'Data User', 'Data Barang', 'Kategori Barang', and 'Data Barang' (which is highlighted). The main content area shows a table of products with columns for ID, Kode Barang, Nama, Harga Beli, Harga Jual, and Aksi. A modal window in the center displays a green checkmark icon and the text 'Berhasil' followed by 'Data barang berhasil diupdate'. At the bottom of the page, there's a footer with copyright information and a link to AdminLTE.io.

- 9 Untuk menampilkan detail data barang maka diperlukan memodifikasi file `BarangController.php` seperti berikut. Dengan menambahkan fungsi `confirm_ajax()` dan `delete_ajax()`.

```
app > Http > Controllers > BarangController.php > PHP > BarangController > delete_ajax()
11 class BarangController extends Controller
237 // Fungsi untuk mengonfirmasi penghapusan barang melalui Ajax
238 1 reference| 0 overrides
239 public function confirm_ajax(string $id): Factory|View
240 {
241     $barang = BarangModel::find(id: $id);
242     return view(view: 'barang.confirm_ajax', data: ['barang' => $barang]);
243 }
244
// Fungsi untuk menghapus barang melalui Ajax
245 1 reference| 0 overrides
246 public function delete_ajax(Request $request, $id): JsonResponse|mixed\Redirector\RedirectRes.
247 [
248     // Cek apakah request berasal dari Ajax atau request JSON
249     if ($request->ajax() || $request->wantsJson()) {
250         $barang = BarangModel::find(id: $id);
251         if ($barang) {
252             $barang->delete();
253             return response()->json(data: [
254                 'status' => true,
255                 'message' => 'Data berhasil dihapus'
256             ]);
257         } else {
258             return response()->json(data: [
259                 'status' => false,
260                 'message' => 'Data tidak ditemukan'
261             ]);
262         }
263     }
264 ]
265 }
```

Dan dilanjutkan dengan membuat file view `confirm_ajax.blade.php`.



```
resource > area > barang > ■■■ config_ajaxDelete.php > @<script> > @<function>
1  @empty($barang)
2      <div id="modal-master" class="modal-dialog modal-lg" role="document">
3          <div class="modal-content">
4              <div class="modal-header">
5                  <h5 class="modal-title" id="exampleModalLabel">Kesalahan!
6                  <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>
7                      aria-hidden="true">&times;</span></button>
8              </div>
9              <div class="modal-body">
10                  <div class="alert alert-danger">
11                      <i class="icon fas fa-ban"></i> Kesalahan!!!<h5>
12                      Data yang anda cari tidak ditemukan
13                  </div>
14                  <a href="{{ url('/barang') }}" class="btn btn-warning">Kembali</a>
15              </div>
16          </div>
17      </div>
18  @else
19      form action="{{ url('/barang/' . $barang->barang_id . '/delete_ajax') }}" method="POST" id="form-delete"
20      @cart
21      @method('DELETE')
22      <div id="modal-master" class="modal-dialog modal-lg" role="document">
23          <div class="modal-content">
24              <div class="modal-header">
25                  <h5 class="modal-title" id="exampleModalLabel">Hapus Data barang
26                  <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span>
27                      aria-hidden="true">&times;</span></button>
28              </div>
29              <div class="modal-body">
30                  <div class="alert alert-warning">
31                      <i class="icon fas fa-ban"></i> Konfirmasi !!!<h5>
32                      Apakah Anda ingin menghapus data seperti di bawah ini?
33                  </div>
34                  <table class="table table-sm table-bordered table-striped">
35                      <tr>
36                          <th class="text-right col-3">Kategori Barang:</th>
37                          <td class="col-9">{{ $barang->kategori->kategori_name }}</td>
38                      </tr>
39                      <tr>
40                          <th class="text-right col-3">Kode Barang :</th>
41                          <td class="col-9">{{ $barang->barang_kode }}</td>
42                      </tr>
43                      <tr>
44                          <th class="text-right col-3">Nama Barang :</th>
45                          <td class="col-9">{{ $barang->barang_nama }}</td>
46                      </tr>
47                      <tr>
48                          <th class="text-right col-3">Harga Beli :</th>
49                          <td class="col-9">{{ $barang->harga_beli }}</td>
50                      </tr>
51                      <tr>
52                          <th class="text-right col-3">Harga Jual :</th>
53                          <td class="col-9">{{ $barang->barang_harga }}</td>
54                      </tr>
55                  </table>
56              </div>
57              <div class="modal-footer">
58                  <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
59                  <button type="submit" class="btn btn-primary">Ya, Hapus</button>
60              </div>
61          </div>
62      </div>
63  </form>
64  <script>
65      $(document).ready(function() {
66          $('#form-delete').validate([
67              rules: {},
68              submitHandler: function(form) {
69                  $.ajax({
70                      url: form.action,
71                      type: form.method,
72                      data: $(form).serialize(),
73                      success: function(response) {
74                          if (response.status) {
75                              $('#myModal').modal('hide');
76                              Swal.fire({
77                                  icon: 'success',
78                                  title: 'Berhasil',
79                                  text: response.message
80                              });
81                              dataBarangAjax.reload();
82                          } else {
83                              $('.error-text').text('');
84                              $.each(response.msgField, function(prefix, val) {
85                                  $('.error-' + prefix).text(val[0]);
86                              });
87                              Swal.fire({
88                                  icon: 'error',
89                                  title: 'Terjadi kesalahan',
90                                  text: response.message
91                              });
92                          }
93                      });
94                  return false;
95              },
96              errorElement: 'span',
97              errorPlacement: function(error, element) {
98                  error.addClass('invalid-feedback');
99                  element.closest('.form-group').append(error);
100             },
101             highlight: function(element, errorClass, validClass) {
102                 $(element).addClass('is-invalid');
103             },
104             unhighlight: function(element, errorClass, validClass) {
105                 $(element).removeClass('is-invalid');
106             }
107         });
108     });
109     @empty
110 
```



10

Berikut adalah tampilan halaman detail supplier yang mengambil data dari database m_barang.

The screenshot shows a modal dialog titled "Hapus Data barang" (Delete Data Item). It contains a confirmation message: "Konfirmasi !!! Apakah Anda ingin menghapus data seperti di bawah ini?". Below this, it lists product details: "Kategori Barang: Food and Beverage", "Kode Barang : 22", "Nama Barang : Sawi Daging", "Harga Beli : 5000", and "Harga Jual : Sawi Daging". At the bottom of the dialog are two buttons: "Batal" (Cancel) and "Ya, Hapus" (Yes, Delete).

The screenshot shows a modal dialog titled "Berhasil" (Successful) with the message "Data berhasil dihapus" (Data has been deleted successfully). Below this is an "OK" button. The background shows a table of products with columns: ID, Kode Barang, Name, Price Buy, Price Sell, and Category. The table includes rows for Kabel, Jagung, and Sawi.

Yang mana tampilan halaman nya seperti gambar diatas, yaitu ketika salah satu data dihapus maka akan terdapat pop up konfirmasi penghapusan dan keterangan yang menunjukan berhasil tidaknya data tersebut dihapus.