



Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 4 (empat) / 5 (lima)
Pertemuan ke- : 6 (enam)

JOBSHEET 06

Ajax Form (AdminLTE) dan Client Validation

Proses pembuatan form CRUD (Create, Read, Update, Delete) dengan validasi di Laravel 10 menggunakan jQuery Validation melibatkan beberapa langkah penting yang mencakup pengaturan database, pembuatan model dan migrasi, pengembangan controller, penulisan view, dan penambahan validasi form di sisi klien. *Client side form validation* lebih dilakukan disisi browser dan bukan untuk tujuan keamanan, tetapi lebih ke kenyamanan pengguna. Sedangkan *server side validation* dilakukan di sisi server dengan tujuan keamanan dengan *filter* semua *request* yang masuk sebelum akhirnya diproses lanjutan.

Salah satu cara yang populer untuk melakukan validasi di sisi klien adalah dengan menggunakan plugin jQuery Validation. Plugin **jQuery Validation** digunakan untuk menambahkan validasi sisi klien pada form. Misalnya, Kita bisa mengatur agar suatu input wajib diisi dan tidak boleh lebih dari 255 karakter. Validasi ini membantu dalam memberikan umpan balik langsung kepada pengguna tentang kesalahan input tanpa perlu memuat ulang halaman ataupun mengirim *request* ke server.

Sesuai dengan **Studi Kasus PWL.pdf**.

Jadi project Laravel 10 kita masih sama dengan menggunakan repositori **PWL_POS**.

Project PWL_POS akan kita gunakan sampai pertemuan 12 nanti, sebagai project yang akan kita pelajari

A. **AJAX form**

AJAX (Asynchronous JavaScript and XML) adalah sebuah teknik atau metode dalam pengembangan web yang memungkinkan aplikasi web untuk mengirim dan menerima data dari server secara asinkron (tanpa memuat ulang seluruh halaman). Dengan AJAX, interaksi antara



klien dan server menjadi lebih dinamis dan responsif, karena pengguna dapat berinteraksi dengan halaman web dan melihat perubahan langsung tanpa harus melakukan refresh halaman.

Ajax form adalah teknik di mana sebuah form HTML dikirim ke server secara asinkron menggunakan AJAX, tanpa memuat ulang seluruh halaman web. Dengan AJAX form, Kita bisa mengirim data ke server dan menampilkan respons secara dinamis di halaman, sehingga meningkatkan pengalaman pengguna dengan membuat interaksi lebih cepat dan lebih responsif.

Mengapa Menggunakan AJAX Form?

1. **Response Instan:** AJAX memungkinkan Kita untuk mengirim data dan menerima respons dari server tanpa perlu memuat ulang halaman.
2. **Pengalaman Pengguna yang Lebih Baik:** Karena tidak ada pemuat ulang halaman, aplikasi terasa lebih cepat dan lebih interaktif, mirip dengan aplikasi desktop.

Pengurangan Beban Server: Dengan mengirim hanya data yang diperlukan, AJAX dapat mengurangi penggunaan bandwidth dan beban di server.

B. Validasi Sisi Client

Validasi di sisi klien adalah proses pemeriksaan data yang dimasukkan oleh pengguna pada form web sebelum data tersebut dikirim ke server. Validasi ini dilakukan menggunakan kode yang berjalan di browser pengguna, seperti JavaScript, dan bertujuan untuk memastikan bahwa data yang dimasukkan sesuai dengan aturan tertentu, seperti format email yang benar, panjang karakter yang sesuai, atau tidak adanya kolom kosong yang wajib diisi.

Tujuan dan Manfaat Validasi di Sisi Klien

1. Umpulan Balik Instan

Pengguna mendapatkan umpan balik segera setelah mereka memasukkan data yang tidak valid, seperti kesalahan format email atau kolom yang tidak diisi. Ini meningkatkan pengalaman pengguna (*user experience*) karena mereka tidak perlu menunggu respon dari server untuk mengetahui apakah input mereka benar atau salah.

2. Mengurangi Beban Server

Dengan melakukan validasi di sisi klien, kesalahan dapat diidentifikasi dan diperbaiki sebelum data dikirim ke server, sehingga server tidak perlu memproses permintaan yang tidak valid. Ini dapat mengurangi beban kerja server dan meningkatkan kinerja aplikasi.

3. Meningkatkan Efisiensi



Validasi di sisi klien membantu mencegah pengiriman data yang tidak valid, sehingga mengurangi jumlah permintaan HTTP yang perlu diproses oleh server. Hal ini menghemat bandwidth dan waktu pemrosesan, membuat aplikasi lebih efisien.

4. Memastikan Integritas Data

Dengan validasi di sisi klien, banyak kesalahan input yang dapat dicegah sebelum data mencapai server. Misalnya, memastikan bahwa nomor telepon hanya berisi angka atau alamat email mengikuti format yang benar.

5. Menyederhanakan Proses Pengembangan

Dengan validasi di sisi klien, pengembang dapat menangani banyak potensi kesalahan input di awal, yang menyederhanakan logika pemrosesan di sisi server. Ini memungkinkan pengembang untuk fokus pada validasi yang lebih kompleks atau logika bisnis lainnya di server.

6. Meningkatkan Keamanan

Meskipun validasi sisi klien tidak bisa menggantikan validasi di sisi server (karena dapat dengan mudah diabaikan atau dimanipulasi oleh pengguna yang berpengalaman), validasi ini tetap dapat membantu dalam mengurangi jumlah data yang tidak valid yang mencapai server. Ini berfungsi sebagai lapisan pertama pertahanan, mencegah beberapa jenis input yang tidak diinginkan.

7. Memberikan Panduan Pengguna

Validasi sisi klien memungkinkan pengembang untuk memberikan panduan dan instruksi yang lebih baik kepada pengguna tentang cara memasukkan data dengan benar. Misalnya, pesan kesalahan bisa ditampilkan di bawah kolom yang salah, memberikan petunjuk spesifik kepada pengguna

Bagaimana Validasi di Sisi Klien Bekerja?

Validasi di sisi klien biasanya dilakukan menggunakan JavaScript atau framework JavaScript seperti jQuery. Berikut adalah contoh sederhana validasi form di sisi klien:

```
<!DOCTYPE html>
<html>
<head>
    <title>Client-Side Validation Example</title>
</head>
<body>
    <form name="myForm" onsubmit="return validateForm()" method="post">
        Name: <input type="text" name="name"><br><br>
        Email: <input type="text" name="email"><br><br>
        <input type="submit" value="Submit">
    </form>
    <script>
        function validateForm() {
            var email = document.forms["myForm"]["email"].value;
            var name = document.forms["myForm"]["name"].value;
        }
    </script>
</body>
</html>
```



```
if (name == "") {  
    alert("Name must be filled out");  
    return false;  
}  
  
var emailPattern = /^[a-zA-Z0-9._-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,6}$/;  
if (!emailPattern.test(email)) {  
    alert("Please enter a valid email address");  
    return false;  
}  
return true;  
}  
</script>  
</body>  
</html>
```

Keuntungan Validasi di Sisi Klien

1. Responsif

Pengguna mendapatkan respons cepat terhadap input mereka tanpa perlu menunggu interaksi dengan server.

2. Interaktif

Dapat memberikan instruksi tambahan dan lebih kontekstual kepada pengguna untuk memperbaiki kesalahan.

3. Penghematan Sumber Daya

Mengurangi jumlah permintaan ke server yang tidak perlu, sehingga menghemat bandwidth dan sumber daya server.

Keterbatasan Validasi di Sisi Klien

1. Tidak Mengganti Validasi di Sisi Server

Validasi di sisi klien dapat dilewati oleh pengguna berpengalaman atau perangkat otomatis. Oleh karena itu, validasi di sisi klien harus selalu dilengkapi dengan validasi di sisi server untuk memastikan keamanan dan integritas data.

2. Ketergantungan pada JavaScript

Jika pengguna menonaktifkan JavaScript di browser mereka, validasi di sisi klien tidak akan berfungsi.

Validasi di sisi klien merupakan komponen penting dalam pengembangan aplikasi web modern, karena meningkatkan pengalaman pengguna dan efisiensi aplikasi. Namun, ini harus selalu digunakan bersama dengan validasi di sisi server untuk menjaga keamanan dan memastikan data yang diterima oleh aplikasi adalah valid dan sesuai dengan aturan bisnis.



C. jQuery Validation

Salah satu cara yang populer untuk melakukan validasi di sisi klien adalah dengan menggunakan plugin jQuery Validation. **jQuery Validation** adalah plugin jQuery yang digunakan untuk memvalidasi form HTML di sisi klien secara efisien dan interaktif. Plugin ini memudahkan pengembang untuk menambahkan logika validasi pada form dengan cara yang mudah dan dapat disesuaikan, memberikan umpan balik langsung kepada pengguna mengenai kesalahan input mereka sebelum data dikirim ke server.

Fitur Utama jQuery Validation

1. Kemudahan Penggunaan

jQuery Validation dirancang untuk memudahkan integrasi dan penggunaan. Dengan beberapa baris kode, Kita dapat menambahkan validasi ke form HTML tanpa perlu menulis logika validasi dari awal.

2. Validasi Real-Time

Plugin ini memvalidasi input form secara real-time saat pengguna mengetik atau setelah mereka pindah dari satu field ke field lainnya. Ini memberikan umpan balik langsung kepada pengguna mengenai kesalahan input mereka.

3. Aturan Validasi yang Siap Pakai:

jQuery Validation menyediakan berbagai aturan validasi yang siap digunakan, seperti:

- *required*: Memastikan bahwa field tidak kosong.
- *email*: Memastikan bahwa input berformat alamat email yang valid.
- *url*: Memastikan bahwa input berformat URL yang valid.
- *minlength* dan *maxlength*: Membatasi jumlah karakter minimum atau maksimum dalam input.
- *number*: Memastikan bahwa input hanya berisi angka.

4. Pesan Kesalahan Kustom

Kita dapat menyesuaikan pesan kesalahan yang ditampilkan kepada pengguna. Misalnya, Kita dapat mengubah pesan default seperti "This field is required" menjadi sesuatu yang lebih spesifik atau sesuai dengan konteks aplikasi Kita.

5. Integrasi dengan jQuery UI

jQuery Validation dapat dengan mudah diintegrasikan dengan jQuery UI untuk menampilkan pesan kesalahan dalam format yang lebih menarik, seperti menggunakan tooltip atau dialog box.

6. Validasi Multi-Field



Plugin ini mendukung validasi yang melibatkan lebih dari satu field. Misalnya, Kita bisa memastikan bahwa dua field password dan konfirmasi password memiliki nilai yang sama.

7. Plugin dan Ekstensi

jQuery Validation memiliki ekosistem plugin dan ekstensi yang memungkinkan Kita menambahkan aturan validasi kustom atau mengubah perilaku default.

Cara Menggunakan jQuery Validation

Berikut adalah contoh sederhana bagaimana jQuery Validation digunakan untuk memvalidasi form:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>jQuery Validation Example</title>
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/jqueryui/1.12.1/jquery-ui.css">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-validate/1.19.3/jquery.validate.min.js"></script>
</head>
<body>
    <form id="myForm">
        <label for="name">Name:</label><input type="text" name="name" id="name"><br>
        <label for="email">Email:</label><input type="text" name="email" id="email"><br>
        <input type="submit" value="Submit">
    </form>

    <script>
        $(document).ready(function() {
            $("#myForm").validate({
                rules: {
                    name: "required",
                    email: {
                        required: true,
                        email: true
                    }
                },
                messages: {
                    name: "Please enter your name",
                    email: "Please enter a valid email address"
                }
            });
        });
    </script>
</body>
</html>
```

Penjelasan:

- **rules**: mendefinisikan aturan validasi untuk setiap field. Dalam contoh di atas:
 - *Field name* harus diisi (required).
 - *Field email* harus diisi dan harus berformat email yang valid (email).



- **messages:** mendefinisikan pesan kesalahan yang akan ditampilkan jika aturan validasi tidak terpenuhi.

Keuntungan Menggunakan jQuery Validation

1. Pengalaman Pengguna yang Lebih Baik

Pengguna mendapatkan umpan balik langsung, yang membantu mereka memperbaiki kesalahan input dengan cepat.

2. Pengurangan Beban Server

Validasi di sisi klien mengurangi jumlah permintaan yang tidak valid yang dikirim ke server, menghemat sumber daya server.

3. Fleksibilitas dan Kustomisasi

Plugin ini sangat fleksibel dan dapat dikustomisasi sesuai kebutuhan aplikasi, dari aturan validasi hingga pesan kesalahan yang ditampilkan.

4. Kompatibilitas dengan Semua Browser Modern

jQuery Validation kompatibel dengan hampir semua browser modern, sehingga dapat digunakan di berbagai lingkungan pengguna.

jQuery Validation memiliki berbagai metode bawaan yang sangat berguna untuk memvalidasi form di sisi klien. Selain metode standar seperti required, email, dan number, Kita juga dapat menambahkan metode validasi kustom menggunakan addMethod. Ini memungkinkan Kita untuk membuat aturan validasi yang lebih spesifik sesuai kebutuhan aplikasi Kita.

D. Method jQuery Validation

jQuery Validation menyediakan beberapa metode bawaan (built-in methods) yang dapat digunakan untuk memvalidasi form dengan berbagai jenis aturan. Selain itu, jQuery Validation juga memungkinkan pengembang untuk menambahkan metode kustom dengan addMethod, seperti yang telah dijelaskan sebelumnya. Berikut adalah beberapa metode tambahan yang tersedia dalam jQuery Validation

No	Method	Deskripsi
1	<i>required</i>	<ul style="list-style-type: none">• Memastikan bahwa field tidak kosong.• Contoh: <code>required: true</code>
2	<i>email</i>	<ul style="list-style-type: none">• Memastikan bahwa input berformat alamat email yang valid.• Contoh: <code>email: true</code>
3	<i>url</i>	<ul style="list-style-type: none">• Memastikan bahwa input berformat URL yang valid.• Contoh: <code>url: true</code>



4	<i>date</i>	<ul style="list-style-type: none">Memastikan bahwa input berformat tanggal yang valid (berdasarkan pengaturan regional)Contoh: <code>date: true</code>
5	<i>dateISO</i>	<ul style="list-style-type: none">Memastikan bahwa input berformat tanggal yang valid dalam format ISO (YYYY-MM-DD)Contoh: <code>dateISO: true</code>
6	<i>number</i>	<ul style="list-style-type: none">Memastikan bahwa input hanya berisi angka (integer atau desimal).Contoh: <code>number: true</code>
7	<i>digits</i>	<ul style="list-style-type: none">Memastikan bahwa input hanya berisi angka (tanpa desimal).Contoh: <code>digits: true</code>
8	<i>creditcard</i>	<ul style="list-style-type: none">Memastikan bahwa input berformat nomor kartu kredit yang valid.Contoh: <code>creditcard: true</code>
9	<i>equalTo</i>	<ul style="list-style-type: none">Memastikan bahwa nilai elemen form sama dengan elemen lain (misalnya, untuk konfirmasi password).Contoh: <code>equalTo: "#password"</code>
10	<i>maxLength</i>	<ul style="list-style-type: none">Memastikan bahwa input tidak melebihi jumlah karakter tertentu.Contoh: <code>maxlength: 10</code>
11	<i>minLength</i>	<ul style="list-style-type: none">Memastikan bahwa input memiliki minimal jumlah karakter tertentu.Contoh: <code> minlength: 5</code>
12	<i>rangeLength</i>	<ul style="list-style-type: none">Memastikan bahwa panjang input berada dalam rentang karakter tertentu.Contoh: <code>rangelength: [5, 10]</code>
13	<i>range</i>	<ul style="list-style-type: none">Memastikan bahwa nilai input berada dalam rentang tertentu (misalnya, angka 1 sampai 100)Contoh: <code>range: [1, 100]</code>
14	<i>max</i>	<ul style="list-style-type: none">Memastikan bahwa nilai input tidak melebihi angka maksimum tertentu.Contoh: <code>max: 100</code>
15	<i>min</i>	<ul style="list-style-type: none">Memastikan bahwa nilai input tidak kurang dari angka minimum tertentu.Contoh: <code>min: 1</code>
16	<i>remote</i>	<ul style="list-style-type: none">Memvalidasi nilai dengan mengirimkan permintaan ke server untuk memeriksa apakah nilai tersebut valid atau tersedia (misalnya, memeriksa ketersediaan username)Contoh<pre>remote: { url: "/check-username", type: "post" }</pre>
17	<i>step</i>	<ul style="list-style-type: none">Memastikan bahwa nilai input adalah kelipatan dari angka tertentu (berguna untuk validasi angka desimal).Contoh: <code>step: 10</code>
18	<i>phoneUS</i>	<ul style="list-style-type: none">Memastikan bahwa input berformat nomor telepon yang valid di AS.Contoh: <code>phoneUS: true</code>



19	<i>extension</i>	<ul style="list-style-type: none">Memastikan bahwa file yang diupload memiliki ekstensi tertentu.Contoh: extension: "jpg png gif"
20	<i>accept</i>	<ul style="list-style-type: none">Memastikan bahwa file yang diupload memiliki jenis MIME tertentu.Contoh: accept: "image/*"
21	<i>exactlength</i>	<ul style="list-style-type: none">Memastikan bahwa input hanya berisi karakter yang panjangnya sama persis dengan ketentuan.Contoh: exactlength: 10

jQuery Validation adalah alat yang sangat berguna untuk memastikan data yang dimasukkan ke dalam form web valid dan sesuai dengan aturan yang ditetapkan sebelum data tersebut dikirim ke server. Ini meningkatkan pengalaman pengguna, mengurangi kesalahan, dan mempermudah pengelolaan validasi form di sisi klien dalam pengembangan aplikasi web.

E. Praktikum Jobsheet

Langsung saja kita praktikkan untuk menggunakan Ajax form dan validasi disisi client.

Praktikum 1. Modal Ajax Tambah Data (Data User)

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.

```
template.blade.php X
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="utf-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1">
6      <title>AdminLTE 3 | Blank Page</title>
7
8      <meta name="csrf-token" content="{{ csrf_token() }}>
9
10     <!-- Font Awesome -->
11     <link rel="stylesheet" href="{{ asset('plugins/fontawesome-free/css/all.min.css') }}>
12     <!-- Datatables -->
13     <link rel="stylesheet" href="{{ asset('plugins/datatables-bs4/css/dataTables.bootstrap4.min.css') }}>
14     <link rel="stylesheet" href="{{ asset('plugins/datatables-responsive/css/responsive.bootstrap4.min.css') }}>
15     <link rel="stylesheet" href="{{ asset('plugins/datatables-buttons/css/buttons.bootstrap4.min.css') }}>
16     <!-- SweetAlert2 -->
17     <link rel="stylesheet" href="{{ asset('plugins/sweetalert2-theme-bootstrap-4/bootstrap-4.min.css') }}>
18     <!-- Theme style -->
19     <link rel="stylesheet" href="{{ asset('dist/css/adminlte.min.css') }}>
20
21     @stack('css')
22 </head>
```



```
template.blade.php ×  
65 <script src="{{ asset('plugins/pdfmake/pdfmake.min.js') }}"></script>  
66 <script src="{{ asset('plugins/pdfmake/vfs_fonts.js') }}"></script>  
67 <script src="{{ asset('plugins/datatables-buttons/js/buttons.html5.min.js') }}"></script>  
68 <script src="{{ asset('plugins/datatables-buttons/js/buttons.print.min.js') }}"></script>  
69 <script src="{{ asset('plugins/datatables-buttons/js/buttons.colVis.min.js') }}"></script>  
70  
71 <!-- jquery-validation -->  
72 <script src="{{ asset('plugins/jquery-validation/jquery.validate.min.js') }}"></script>  
73 <script src="{{ asset('plugins/jquery-validation/additional-methods.min.js') }}"></script>  
74  
75 <!-- SweetAlert2 -->  
76 <script src="{{ asset('plugins/sweetalert2/sweetalert2.min.js') }}"></script>  
77  
78 <!-- AdminLTE App -->  
79 <script src="{{ asset('dist/js/adminlte.min.js') }}"></script>  
80 <script>  
81   $.ajaxSetup({  
82     headers: {  
83       'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content')  
84     }  
85   });  
86 </script>  
87  
88 @stack('js')  
89 </body>  
90 </html>
```

3. Selanjutnya Kita modifikasi view `user/index.blade.php`, kita tambahkan tombol untuk membuat form popup ajax

```
@extends('layouts.template')  
  
@section('content')  
  <div class="card card-outline card-primary">  
    <div class="card-header">  
      <h3 class="card-title">{{ $page->title }}</h3>  
      <div class="card-tools">  
        <a class="btn btn-sm btn-primary mt-1" href="{{ url('user/create') }}">Tambah</a>  
        <button onclick="modalAction('{{ url('user/create_ajax') }}'" class="btn btn-sm btn-success mt-1">Tambah Ajax</button>  
      </div>  
    </div>  
    <div class="card-body">  
      @if (@session('success'))  
        <div class="alert alert-success">{{ session('success') }}</div>  
      @endif  
      @if (session('error'))  
        <div class="alert alert-danger">{{ session('error') }}</div>  
      @endif  
      <div class="row">  
        <div class="col-md-12">  
          <div class="form-group row">  
            <label class="col-1 control-label col-form-label">Filter:</label>  
            <div class="col-3">  
              <select class="form-control" id="level_id" name="level_id" required>  
                <option value="">- Semua -</option>  
                @foreach ($level as $item)  
                  <option value="{{ $item->level_id }}>{{ $item->level_nama }}</option>  
                @endforeach  
              </select>  
            </div>  
          </div>  
        </div>  
      </div>  
    </div>  
  </div>
```

Kita tambahkan kode berikut, untuk membuat form modal tambah data user dengan ajax



```
<button onclick="modalAction('{{ url('/user/create_ajax') }}')" class="btn btn-sm btn-success mt-1">Tambah Ajax</button>
```

4. Selanjutnya kita tambahkan kode berikut pada **akhir @section('content')** pada view `user/index.blade.php`

```
<div id="myModal" class="modal fade animate shake" tabindex="-1" role="dialog" data-backdrop="static" data-keyboard="false" data-width="75%" aria-hidden="true"></div>
```

5. Kemudian kita tambahkan kode berikut pada **awal @push('js')** pada view `user/index.blade.php`

```
function modalAction(url = ''){
    $('#myModal').load(url, function(){
        $('#myModal').modal('show');
    });
}
```

Sehingga tampilan kode program akan seperti berikut

```
<div id="myModal" class="modal fade animate shake" tabindex="-1" role="dialog" data-backdrop="static"
    data-keyboard="false" data-width="75%" aria-hidden="true"></div>
@endsection

@push('css')
@endpush

@push('js')
<script>
    function modalAction(url = ''){
        $('#myModal').load(url, function(){
            $('#myModal').modal('show');
        });
    }

    var dataUser;
    $(document).ready(function() {
        dataUser = $('#table_user').DataTable({
            // serverSide: true, jika ingin menggunakan server side processing
            serverSide: true,
            ajax: {
                "url": "{{ url('user/list') }}",
                "dataType": "json",
                "type": "POST",
                "data": function (d){
                    d.level_id = $('#level_id').val();
                }
            },
            columns: [

```

Ubah seperti ini

6. Selanjutkan Kita modifikasi `route/web.php` untuk mengakomodir operasi ajax



```
Route::group(['prefix' => 'user'], function () {
    Route::get('/', [UserController::class, 'index']); // Menampilkan halaman awal user
    Route::post('/list', [UserController::class, 'list']); // Menampilkan data user dalam bentuk json untuk datatables
    Route::get('/create', [UserController::class, 'create']); // Menampilkan halaman form tambah user
    Route::post('/', [UserController::class, 'store']); // Menyimpan data user baru
    Route::get('/create_ajax', [UserController::class, 'create_ajax']); // Menampilkan halaman form tambah user Ajax
    Route::post('/ajax', [UserController::class, 'store_ajax']); // Menyimpan data user baru Ajax
    Route::get('/{id}', [UserController::class, 'show']); // Menampilkan detail user
    Route::get('/{id}/edit', [UserController::class, 'edit']); // Menampilkan halaman form edit user
    Route::put('/{id}', [UserController::class, 'update']); // Menyimpan perubahan data user
    Route::delete('/{id}', [UserController::class, 'destroy']); // Menghapus data user
});
```

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

```
public function create_ajax()
{
    $level = LevelModel::select('level_id', 'level_nama')->get();

    return view('user.create_ajax')
        ->with('level', $level);
}
```

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

```
<form action="{{ url('/user/ajax') }}" method="POST" id="form-tambah">
@csrf
<div id="modal-master" class="modal-dialog modal-lg" role="document">
    <div class="modal-content">
        <div class="modal-header">
            <h5 class="modal-title" id="exampleModalLabel">Tambah Data User</h5>
            <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span aria-hidden="true">&times;</span></button>
        </div>
        <div class="modal-body">
            <div class="form-group">
                <label>Level Pengguna</label>
                <select name="level_id" id="level_id" class="form-control" required>
                    <option value="">- Pilih Level -</option>
                    @foreach($level as $l)
                        <option value="{{ $l->level_id }}>{{ $l->level_nama }}</option>
                    @endforeach
                </select>
                <small id="error-level_id" class="error-text form-text text-danger"></small>
            </div>
            <div class="form-group">
                <label>Username</label>
                <input value="" type="text" name="username" id="username" class="form-control" required>
                <small id="error-username" class="error-text form-text text-danger"></small>
            </div>
            <div class="form-group">
                <label>>Nama</label>
                <input value="" type="text" name="nama" id="nama" class="form-control" required>
                <small id="error-nama" class="error-text form-text text-danger"></small>
            </div>
            <div class="form-group">
                <label>Password</label>
                <input value="" type="password" name="password" id="password" class="form-control" required>
                <small id="error-password" class="error-text form-text text-danger"></small>
            </div>
        </div>
    </div>
</div>
```



```
<div class="modal-footer">
    <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
    <button type="submit" class="btn btn-primary">Simpan</button>
</div>
</div>
</form>
<script>
$(document).ready(function() {
    $("#form-tambah").validate({
        rules: {
            level_id: {required: true, number: true},
            username: {required: true, minlength: 3, maxlength: 20},
            nama: {required: true, minlength: 3, maxlength: 100},
            password: {required: true, minlength: 6, maxlength: 20}
        },
        submitHandler: function(form) {
            $.ajax({
                url: form.action,
                type: form.method,
                data: $(form).serialize(),
                success: function(response) {
                    if(response.status){
                        $('#myModal').modal('hide');
                        Swal.fire({
                            icon: 'success',
                            title: 'Berhasil',
                            text: response.message
                        });
                        dataUser.ajax.reload();
                    }else{
                        $('.error-text').text('');
                        $.each(response.msgField, function(prefix, val) {
                            $('#error-' +prefix).text(val[0]);
                        });
                        Swal.fire({
                            icon: 'error',
                            title: 'Terjadi Kesalahan',
                            text: response.message
                        });
                    }
                }
            });
            return false;
        },
        errorElement: 'span',
        errorPlacement: function (error, element) {
            error.addClass('invalid-feedback');
            element.closest('.form-group').append(error);
        },
        highlight: function (element, errorClass, validClass) {
            $(element).addClass('is-invalid');
        },
        unhighlight: function (element, errorClass, validClass) {
            $(element).removeClass('is-invalid');
        }
    });
}</script>
```

9. Kemudian untuk mengakomodir proses simpan data melalui ajax, kita buat fungsi `store_ajax()` pada `UserController.php`



```
public function store_ajax(Request $request) {
    // cek apakah request berupa ajax
    if($request->ajax() || $request->wantsJson()){
        $rules = [
            'level_id' => 'required|integer',
            'username' => 'required|string|min:3|unique:m_user,username',
            'nama'      => 'required|string|max:100',
            'password'  => 'required|min:6'
        ];

        // use Illuminate\Support\Facades\Validator;
        $validator = Validator::make($request->all(), $rules);

        if($validator->fails()){
            return response()->json([
                'status' => false, // response status, false: error/gagal, true: berhasil
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors(), // pesan error validasi
            ]);
        }

        UserModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data user berhasil disimpan'
        ]);
    }
    redirect('/');
}
```

10. OK, sekarang kita coba melakukan proses tambah data user menggunakan form ajax. Amati apa yang terjadi dan laporankan pada laporan *jobsheet* dan *commit* di github kalian!!!

Tambah Data User ×

Level Pengguna
Administrator

Username
Xonexorcist

Nama
Audric Andhika

Password

Batal Simpan



Praktikum 2. Modal Ajax Edit Data (Data User)

1. Pada Praktikum 2 ini, kita akan melakukan koding untuk proses edit menggunakan ajax.
2. Pertama-tama, kita **ubah** dulu fungsi `list()` pada `UserController.php` untuk mengganti **tombol edit** untuk bisa menggunakan modal

```
// Ambil data user dalam bentuk json untuk datatables
public function list(Request $request)
{
    $users = UserModel::select('user_id', 'username', 'nama', 'level_id')
        ->with('level');

    // Filter data user berdasarkan level_id
    if ($request->level_id){
        $users->where('level_id',$request->level_id);
    }

    return DataTables::of($users)
        ->addIndexColumn() // menambahkan kolom index / no urut (default nama kolom:
DT_RowIndex)
        ->addColumn('aksi', function ($user) { // menambahkan kolom aksi
```



```
        /* $btn = '<a href="'.url('/user/' . $user->user_id).'" class="btn btn-info btn-sm">Detail</a> ';
           $btn .= '<a href="'.url('/user/' . $user->user_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
           $btn .= '<form class="d-inline-block" method="POST" action="'.url('/user/' . $user->user_id).'">
                    . 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>';
           $btn = '<button onclick="modalAction('.url('/user/' . $user->user_id .
'/show_ajax').')"' class="btn btn-info btn-sm">Detail</button> ';
           $btn .= '<button onclick="modalAction('.url('/user/' . $user->user_id .
'/edit_ajax').')"' class="btn btn-warning btn-sm">Edit</button> ';
           $btn .= '<button onclick="modalAction('.url('/user/' . $user->user_id .
'/delete_ajax').')"' class="btn btn-danger btn-sm">Hapus</button> ';

           return $btn;
    })
    ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html
    ->make(true);
}
```

3. Selanjutnya kita modifikasi `routes/web.php` untuk mengakomodir request edit menggunakan ajax

```
Route::group(['prefix' => 'user'], function () {
    Route::get('/', [UserController::class, 'index']); // Menampilkan halaman awal user
    Route::post('/list', [UserController::class, 'list']); // Menampilkan data user dalam bentuk json untuk datatables
    Route::get('/create', [UserController::class, 'create']); // Menampilkan halaman form tambah user
    Route::post('/', [UserController::class, 'store']); // Menyimpan data user baru
    Route::get('/create_ajax', [UserController::class, 'create_ajax']); // Menampilkan halaman form tambah user Ajax
    Route::post('/ajax', [UserController::class, 'store_ajax']); // Menyimpan data user baru Ajax
    Route::get('/{id}', [UserController::class, 'show']); // Menampilkan detail user
    Route::get('/{id}/edit', [UserController::class, 'edit']); // Menampilkan halaman form edit user
    Route::put('/{id}', [UserController::class, 'update']); // Menyimpan perubahan data user
    Route::get('/{id}/edit_ajax', [UserController::class, 'edit_ajax']); // Menampilkan halaman form edit user Ajax
    Route::put('/{id}/update_ajax', [UserController::class, 'update_ajax']); // Menyimpan perubahan data user Ajax
    Route::delete('/{id}', [UserController::class, 'destroy']); // Menghapus data user
});
```

4. Kemudian, kita buat fungsi `edit_ajax()` pada `UserController.php`

```
// Menampilkan halaman form edit user ajax
public function edit_ajax(string $id)
{
    $user = UserModel::find($id);
    $level = LevelModel::select('level_id', 'level_nama')->get();

    return view('user.edit_ajax', ['user' => $user, 'level' => $level]);
}
```

5. Kita buat **view baru** pada `user/edit_ajax.blade.php` untuk menampilkan form view ajax

```
@empty($user)
    <div id="modal-master" class="modal-dialog modal-lg" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Kesalahan</h5>
                <button type="button" class="close" data-dismiss="modal" aria-
label="Close"><span aria-hidden="true">&times;</span></button>
```



```
</div>
<div class="modal-body">
    <div class="alert alert-danger">
        <i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
        Data yang anda cari tidak ditemukan</div>
    <a href="{{ url('/user') }}" class="btn btn-warning">Kembali</a>
</div>
</div>
</div>
@else
    <form action="{{ url('/user/' . $user->user_id . '/update_ajax') }}" method="POST" id="form-edit">
        @csrf
        @method('PUT')
        <div id="modal-master" class="modal-dialog modal-lg" role="document">
            <div class="modal-content">
                <div class="modal-header">
                    <h5 class="modal-title" id="exampleModalLabel">Edit Data User</h5>
                    <button type="button" class="close" data-dismiss="modal" aria-
label="Close"><span aria-hidden="true">&times;</span></button>
                </div>
                <div class="modal-body">
                    <div class="form-group">
                        <label>Level Pengguna</label>
                        <select name="level_id" id="level_id" class="form-control" required>
                            <option value="">- Pilih Level -</option>
                            @foreach($level as $l)
                                <option {{ ($l->level_id == $user->level_id)? 'selected' : '' }}>{{ $l->level_nama }}</option>
                            @endforeach
                        </select>
                        <small id="error-level_id" class="error-text form-text text-
danger"></small>
                    </div>
                    <div class="form-group">
                        <label>Username</label>
                        <input value="{{ $user->username }}" type="text" name="username" id="username" class="form-control" required>
                        <small id="error-username" class="error-text form-text text-
danger"></small>
                    </div>
                    <div class="form-group">
                        <label>Nama</label>
                        <input value="{{ $user->nama }}" type="text" name="nama" id="nama" class="form-control" required>
                        <small id="error-nama" class="error-text form-text text-danger"></small>
                    </div>
                    <div class="form-group">
                        <label>Password</label>
                        <input value="" type="password" name="password" id="password" class="form-
control">
                        <small class="form-text text-muted">Abaikan jika tidak ingin ubah
password</small>
                        <small id="error-password" class="error-text form-text text-
danger"></small>
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" data-dismiss="modal" class="btn btn-
warning">Batal</button>
                    <button type="submit" class="btn btn-primary">Simpan</button>
                </div>
            </div>
        </form>
        <script>
```



```
$(document).ready(function() {
    $("#form-edit").validate({
        rules: {
            level_id: {required: true, number: true},
            username: {required: true, minlength: 3, maxlength: 20},
            nama: {required: true, minlength: 3, maxlength: 100},
            password: {minlength: 6, maxlength: 20}
        },
        submitHandler: function(form) {
            $.ajax({
                url: form.action,
                type: form.method,
                data: $(form).serialize(),
                success: function(response) {
                    if(response.status){
                        $('#myModal').modal('hide');
                        Swal.fire({
                            icon: 'success',
                            title: 'Berhasil',
                            text: response.message
                        });
                        dataUser.ajax.reload();
                    }else{
                        $('.error-text').text('');
                        $.each(response.msgField, function(prefix, val) {
                            $('#error-'+prefix).text(val[0]);
                        });
                        Swal.fire({
                            icon: 'error',
                            title: 'Terjadi Kesalahan',
                            text: response.message
                        });
                    }
                }
            });
            return false;
        },
        errorElement: 'span',
        errorPlacement: function (error, element) {
            error.addClass('invalid-feedback');
            element.closest('.form-group').append(error);
        },
        highlight: function (element, errorClass, validClass) {
            $(element).addClass('is-invalid');
        },
        unhighlight: function (element, errorClass, validClass) {
            $(element).removeClass('is-invalid');
        }
    });
});
</script>
@endempty
```

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

```
public function update_ajax(Request $request, $id){
    // cek apakah request dari ajax
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'level_id' => 'required|integer',
            'username' => 'required|max:20|unique:m_user,username,'.$id.',user_id',
            'nama'      => 'required|max:100',
            'password'  => 'nullable|min:6|max:20'
        ];
    }
}
```



```
// use Illuminate\Support\Facades\Validator;
$validator = Validator::make($request->all(), $rules);

if ($validator->fails()) {
    return response()->json([
        'status' => false, // respon json, true: berhasil, false: gagal
        'message' => 'Validasi gagal.',
        'msgField' => $validator->errors() // menunjukkan field mana yang error
    ]);
}

$check = UserModel::find($id);
if ($check) {
    if (!$request->filled('password')) { // jika password tidak diisi, maka hapus dari
request
        $request->request->remove('password');
    }

    $check->update($request->all());
    return response()->json([
        'status' => true,
        'message' => 'Data berhasil diupdate'
    ]);
} else{
    return response()->json([
        'status' => false,
        'message' => 'Data tidak ditemukan'
    ]);
}
}
return redirect('/');
}
```

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 user data. On the left, there's a sidebar with various menu items like Dashboard, Data Pengguna, Level User, Data User (which is currently selected and highlighted in blue), Data Barang, Kategori Barang, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main content area has a title 'Daftar User' and a sub-section 'Daftar user yang terdaftar'. A modal window titled 'Edit Data User' is open, showing fields for 'Username' (ADM), 'Nama' (Audric Andhika), and 'Password'. Below the modal, a table lists user data with columns for ID, Username, Name, and Level Pengguna. The table shows 9 entries. At the bottom right of the modal, there are 'Batal' and 'Simpan' buttons.



Praktikum 3. Modal Ajax Hapus Data (Data User)

1. Pada Praktikum 3 ini, kita akan melakukan koding untuk proses hapus menggunakan ajax.
2. Pertama-tama, kita ubah dulu `routes/web.php` untuk mengakomodir request halaman konfirmasi untuk menghapus data

```
Route::group(['prefix' => 'user'], function () {  
    Route::get('/', [UserController::class, 'index']); // Menampilkan halaman awal user  
    Route::post('/list', [UserController::class, 'list']); // Menampilkan data user dalam bentuk json untuk datatables  
    Route::get('/create', [UserController::class, 'create']); // Menampilkan halaman form tambah user  
    Route::post('/', [UserController::class, 'store']); // Menyimpan data user baru  
    Route::get('/create_ajax', [UserController::class, 'create_ajax']); // Menampilkan halaman form tambah user Ajax  
    Route::post('/ajax', [UserController::class, 'store_ajax']); // Menyimpan data user baru Ajax  
    Route::get('{id}', [UserController::class, 'show']); // Menampilkan detail user  
    Route::get('{id}/edit', [UserController::class, 'edit']); // Menampilkan halaman form edit user  
    Route::put('{id}', [UserController::class, 'update']); // Menyimpan perubahan data user  
    Route::get('{id}/edit_ajax', [UserController::class, 'edit_ajax']); // Menampilkan halaman form edit user Ajax  
    Route::put('{id}/update_ajax', [UserController::class, 'update_ajax']); // Menyimpan perubahan data user Ajax  
    Route::get('{id}/delete_ajax', [UserController::class, 'confirm_ajax']); // Untuk tampilan form confirm delete user Ajax  
    Route::delete('{id}/delete_ajax', [UserController::class, 'delete_ajax']); // Untuk hapus data user Ajax  
    Route::delete('{id}', [UserController::class, 'destroy']); // Menghapus data user  
});
```

3. Kemudian kita buat fungsi `confirm_ajax()` pada `UserController.php`



```
public function confirm_ajax(string $id){  
    $user = UserModel::find($id);  
  
    return view('user.confirm_ajax', ['user' => $user]);  
}
```

4. Selanjutnya kita view untuk konfirmasi hapus data dengan nama `user/confirm_ajax.blade.php`

```
@empty($user)  
    <div id="modal-master" class="modal-dialog modal-lg" role="document">  
        <div class="modal-content">  
            <div class="modal-header">  
                <h5 class="modal-title" id="exampleModalLabel">Kesalahan</h5>  
                <button type="button" class="close" data-dismiss="modal" aria-  
label="Close"><span aria-hidden="true">&times;</span></button>  
            </div>  
            <div class="modal-body">  
                <div class="alert alert-danger">  
                    <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>  
                    Data yang anda cari tidak ditemukan</div>  
                    <a href="{{ url('/user') }}" class="btn btn-warning">Kembali</a>  
                </div>  
            </div>  
        </div>  
    @else  
        <form action="{{ url('/user/' . $user->user_id . '/delete_ajax') }}" method="POST" id="form-  
delete">  
            @csrf  
            @method('DELETE')  
            <div id="modal-master" class="modal-dialog modal-lg" role="document">  
                <div class="modal-content">  
                    <div class="modal-header">  
                        <h5 class="modal-title" id="exampleModalLabel">Hapus Data User</h5>  
                        <button type="button" class="close" data-dismiss="modal" aria-  
label="Close"><span aria-hidden="true">&times;</span></button>  
                    </div>  
                    <div class="modal-body">  
                        <div class="alert alert-warning">  
                            <h5><i class="icon fas fa-ban"></i> Konfirmasi !!!</h5>  
                            Apakah Anda ingin menghapus data seperti di bawah ini?  
                        </div>  
                        <table class="table table-sm table-bordered table-striped">  
                            <tr><th class="text-right col-3">Level Pengguna :</th><td class="col-9">{{  
$user->level->level_nama }}</td></tr>  
                            <tr><th class="text-right col-3">Username :</th><td class="col-9">{{  
$user->username }}</td></tr>  
                            <tr><th class="text-right col-3">>Nama :</th><td class="col-9">{{ $user-  
>nama }}</td></tr>  
                        </table>  
                    </div>  
                    <div class="modal-footer">  
                        <button type="button" data-dismiss="modal" class="btn btn-  
warning">Batal</button>  
                        <button type="submit" class="btn btn-primary">Ya, Hapus</button>  
                    </div>  
                </div>  
            </form>  
            <script>  
                $(document).ready(function() {  
                    $("#form-delete").validate({  
                        rules: {},  
                        submitHandler: function(form) {  
                            $.ajax({  
                                type: "POST",  
                                url: form.action,  
                                data: $(form).serialize(),  
                                success: function(response) {  
                                    if (response === "success") {  
                                        $(form).closest(".modal").modal("hide");  
                                        // Handle success response  
                                    }  
                                },  
                                error: function(response) {  
                                    console.log(response);  
                                }  
                            });  
                        }  
                    });  
                })  
            </script>  
        
```



```
        url: form.action,
        type: form.method,
        data: $(form).serialize(),
        success: function(response) {
            if(response.status){
                $('#myModal').modal('hide');
                Swal.fire({
                    icon: 'success',
                    title: 'Berhasil',
                    text: response.message
                });
                dataUser.ajax.reload();
            }else{
                $('.error-text').text('');
                $.each(response.msgField, function(prefix, val) {
                    $('#error-' + prefix).text(val[0]);
                });
                Swal.fire({
                    icon: 'error',
                    title: 'Terjadi Kesalahan',
                    text: response.message
                });
            }
        });
        return false;
    },
    errorElement: 'span',
    errorPlacement: function (error, element) {
        error.addClass('invalid-feedback');
        element.closest('.form-group').append(error);
    },
    highlight: function (element, errorClass, validClass) {
        $(element).addClass('is-invalid');
    },
    unhighlight: function (element, errorClass, validClass) {
        $(element).removeClass('is-invalid');
    }
});
});  

</script>  

@endempty
```

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



```
public function delete_ajax(Request $request, $id)
{
    // cek apakah request dari ajax
    if ($request->ajax() || $request->wantsJson()) {
        $user = UserModel::find($id);
        if ($user) {
            $user->delete();
            return response()->json([
                'status' => true,
                'message' => 'Data berhasil dihapus'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}
```

- Setelah semua selesai, mari kita coba untuk melakukan percobaan dari koding yang telah kita lakukan.

The screenshot shows a user management interface. On the left, there's a sidebar with links like Dashboard, Data Pengguna, Level User, and Data User. The main area shows a table of users with columns for ID, Username, Name, and Level Pengguna. A modal window is open over the table, prompting the user to confirm the deletion of a specific user. The modal has buttons for 'Batal' (Cancel) and 'Ya, Hapus' (Yes, Delete).

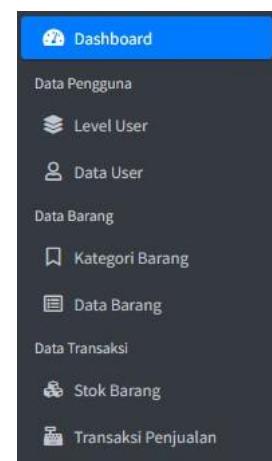
- Jangan lupa laporkan ke laporan jobsheet dan lakukan *commit* pada *repository git* kalian!!!

F. Tugas Jobsheet

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.





1. Tabel m_level

- Views

Create_ajax.blade.php

```
1 <form 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="exampleModallLabel">Tambah Data Level</h5>
7   <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8     | <span aria-hidden="true">&times;
```



Create_ajax.blade.php

```
1 <form 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;
```



Edit_ajax.blade.php

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



Confirm_ajax.blade.php

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



LevelController

```
public function list()
{
    $levels = LevelModel::select('level_id', 'level_kode', 'level_nama');

    // return DataTables::of($levels)
    // ->addIndexColumn()
    // ->addColumn('aksi', function ($level) {
    //     $btn = '<a href="'.url('/level/' . $level->level_id).'" class="btn btn-info btn-sm">Detail</a> ';
    //     $btn .= '<a href="'.url('/level/' . $level->level_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
    //     $btn .= '<form class="d-inline-block" method="POST" action="'.url('/level/' . $level->level_id).'">';
    //         <button type="submit" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin menghapus data ini?');">Hapus</button>
    //     </form> ';
    //     return $btn;
    // });

    return DataTables::of($levels)
        ->addIndexColumn()
        ->addColumn('aksi', function ($level) {
            $btn = '<button onclick="modalAction(\'' . url('/level/' . $level->level_id . '/show_ajax').'\')"' class="btn btn-info btn-sm">Detail</button> ';
            $btn .= '<button onclick="modalAction(\'' . url('/level/' . $level->level_id . '/edit_ajax').'\')"' class="btn btn-warning btn-sm">Edit</button> ';
            $btn .= '<button onclick="modalAction(\'' . url('/level/' . $level->level_id . '/delete_ajax').'\')"' class="btn btn-danger btn-sm">Hapus</button> ';
            return $btn;
        })
        ->rawColumns(['aksi'])
        ->make(true);
}

public function create_ajax()
{
    return view('level.create_ajax');
}

public function store_ajax(Request $request)
{
    // Cek apakah request berupa AJAX
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'level_kode' => 'required|string|min:3|max:20|unique:m_level,level_kode',
            'level_nama' => 'required|string|max:100',
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors(),
            ]);
        }

        LevelModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data level berhasil disimpan'
        ]);
    }

    return redirect('/');
}

public function edit_ajax(string $id)
{
    $level = LevelModel::find($id);

    return view('level.edit_ajax', ['level' => $level]);
}
```



```
public function update_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'level_kode' => 'required|string|min:3|max:20|unique:m_level,level_kode,'.$id.',level_id',
            'level_nama' => 'required|string|max:100',
        ];
        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi gagal.',
                'msgField' => $validator->errors()
            ]);
        }

        $check = LevelModel::find($id);
        if ($check) {
            $check->update($request->all());
            return response()->json([
                'status' => true,
                'message' => 'Data level berhasil diupdate'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}

public function confirm_ajax(string $id)
{
    $level = LevelModel::find($id);

    return view('level.confirm_ajax', ['level' => $level]);
}

public function delete_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $level = LevelModel::find($id);
        if ($level) {
            $level->delete();
            return response()->json([
                'status' => true,
                'message' => 'Data level berhasil dihapus'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}
```

- Route

```
Route::group(['prefix' => 'level'], function () {
    Route::get('/', [LevelController::class, 'index']);
    Route::post('/list', [LevelController::class, 'list']);
    Route::get('/create', [LevelController::class, 'create']);
    Route::post('/', [LevelController::class, 'store']);
    Route::get('/create_ajax', [LevelController::class, 'create_ajax']);
    Route::post('/ajax', [LevelController::class, 'store_ajax']);
    Route::get('{id}', [LevelController::class, 'show']);
    Route::get('{id}/edit', [LevelController::class, 'edit']);
    Route::put('{id}', [LevelController::class, 'update']);
    Route::get('{id}/edit_ajax', [LevelController::class, 'edit_ajax']);
    Route::put('{id}/update_ajax', [LevelController::class, 'update_ajax']);
    Route::get('{id}/delete_ajax', [LevelController::class, 'confirm_ajax']);
    Route::delete('{id}/delete_ajax', [LevelController::class, 'delete_ajax']);
    Route::delete('{id}', [LevelController::class, 'destroy']);
});
```



Daftar Level

ID	Kode Level	Nama Level	Aksi
1	ADM	Administrator	[Detail] [Edit] [Hapus]
2	MNG	Manager	[Detail] [Edit] [Hapus]
3	STF	Staff/Kasir	[Detail] [Edit] [Hapus]

Tambah Data

Tambah Data Level

ID	Kode Level	Nama Level
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir
4	CEO	Chief Executive Officer

Edit Data

Edit Data Level

ID	Kode Level	Nama Level
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir
4	CEO	Chief Executive Officer

Delete Data

Hapus Data Level

ID	Kode Level	Nama Level
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir
4	CEO	Chief Executive Officer



2. Tabel m_kategori

- Views

Create_ajax.blade.php

```
1 <form action="{{ url('/kategori/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 Kategori</h5>
7                 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8                     <span aria-hidden="true">&times;
```



Edit_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">
7                      <span aria-hidden="true">&times;&times;
```



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">
7                     <span aria-hidden="true">&times;;</span>
8                 </button>
9             </div>
10            <div class="modal-body">
11                <div class="alert alert-danger">
12                    <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
13                    Data yang anda cari tidak ditemukan
14                </div>
15                <a href="{{ url('/kategori') }}" class="btn btn-warning">Kembali</a>
16            </div>
17        </div>
18    </div>
19 @else
20     <form action="{{ url('/kategori/' . $kategori->kategori_id . '/delete_ajax') }}" method="POST" id="form-delete">
21         @csrf
22         @method("DELETE")
23         <div id="modal-master" class="modal-dialog modal-lg" role="document">
24             <div class="modal-content">
25                 <div class="modal-header">
26                     <h5 class="modal-title" id="exampleModallabel">Hapus Data Kategori</h5>
27                     <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28                         <span aria-hidden="true">&times;;</span>
29                     </button>
30                 </div>
31                 <div class="modal-body">
32                     <div class="alert alert-warning">
33                         <h5><i class="icon fas fa-ham"></i> Konfirmasi !!!</h5>
34                         Apakah Anda ingin menghapus data seperti di bawah ini?
35                     </div>
36                     <table class="table table-sm table-bordered table-striped">
37                         <tr>
38                             <th class="text-right col-3">Kode Kategori :</th>
39                             <td class="col-9" style="text-align: center;">{{ $kategori->kategori_kode }}{{ $kategori->kategori_nama }}
```



- KategoriController

```
public function list()
{
    $kategori = KategoriModel::select('kategori_id', 'kategori_kode', 'kategori_nama');

    // return DataTables::of($kategori)
    // ->addIndexColumn()
    // ->addColumn('aksi', function ($kategori) {
    //     $btn = '<a href="'.url('/kategori/').$kategori->kategori_id.'>Detail</a> ';
    //     $btn .= '<a href="'.url('/kategori/').$kategori->kategori_id.'>Edit</a> ';
    //     $btn .= '<form class="d-inline-block" method="POST" action="'.url('/kategori/').$kategori->kategori_id.'>';
    //         <button type="submit" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin menghapus data ini?');">Hapus</button>
    //     </form> ';
    //     return $btn;
    // });

    return DataTables::of($kategori)
        ->addIndexColumn()
        ->addColumn('aksi', function ($kategori) {
            $btn = '<button onclick="modalAction('.url('/kategori/').$kategori->kategori_id.'>show_ajax').>' class="btn btn-info btn-sm">Detail</button>';
            $btn .= '<button onclick="modalAction('.url('/kategori/').$kategori->kategori_id.'>edit_ajax').>' class="btn btn-warning btn-sm">Edit</button>';
            $btn .= '<button onclick="modalAction('.url('/kategori/').$kategori->kategori_id.'>delete_ajax').>' class="btn btn-danger btn-sm">Hapus</button>';
            return $btn;
        });

    ->rawColumns(['aksi']);
    ->make(true);
}

public function create_ajax()
{
    return view('kategori.create_ajax');
}

public function store_ajax(Request $request)
{
    // Cek apakah request berupa AJAX
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'kategori_kode' => 'required|string|min:3|max:20|unique:m_kategori,kategori_kode',
            'kategori_nama' => 'required|string|max:100',
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors(),
            ]);
        }

        KategoriModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data kategori berhasil disimpan'
        ]);
    }

    return redirect('/');
}

public function edit_ajax(string $id)
{
    $kategori = KategoriModel::find($id);

    return view('kategori.edit_ajax', ['kategori' => $kategori]);
}
```



```
public function update_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'kategori_kode' => 'required|string|min:3|max:20|unique:m_kategori,kategori.kode,'.$id.',kategori_id',
            'kategori_nama' => 'required|string|max:100',
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi gagal.',
                'msgField' => $validator->errors()
            ]);
        }

        $check = KategoriModel::find($id);
        if ($check) {
            $check->update($request->all());
            return response()->json([
                'status' => true,
                'message' => 'Data kategori berhasil diupdate'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}

public function confirm_ajax(string $id)
{
    $kategori = KategoriModel::find($id);

    return view('kategori.confirm_ajax', ['kategori' => $kategori]);
}

public function delete_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $kategori = KategoriModel::find($id);
        if ($kategori) {
            if ($kategori->barang->exists()) {
                return response()->json([
                    'status' => false,
                    'message' => 'Data kategori tidak dapat dihapus karena masih digunakan di data barang.'
                ]);
            }

            $kategori->delete();
            return response()->json([
                'status' => true,
                'message' => 'Data kategori berhasil dihapus'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}
```

- Route

```
Route::group(['prefix' => 'kategori'], function () {
    Route::get('/', [KategoriController::class, 'index']);
    Route::post('/list', [KategoriController::class, 'list']);
    Route::get('/create', [KategoriController::class, 'create']);
    Route::post('/', [KategoriController::class, 'store']);
    Route::get('/create_ajax', [KategoriController::class, 'create_ajax']);
    Route::post('/ajax', [KategoriController::class, 'store_ajax']);
    Route::get('/{id}', [KategoriController::class, 'show']);
    Route::get('/{id}/edit', [KategoriController::class, 'edit']);
    Route::put('/{id}', [KategoriController::class, 'update']);
    Route::get('/{id}/edit_ajax', [KategoriController::class, 'edit_ajax']);
    Route::put('/{id}/update_ajax', [KategoriController::class, 'update_ajax']);
    Route::get('/{id}/delete_ajax', [KategoriController::class, 'confirm_ajax']);
    Route::delete('/{id}/delete_ajax', [KategoriController::class, 'delete_ajax']);
    Route::delete('/{id}', [KategoriController::class, 'destroy']);
});
```



Daftar Kategori

ID	Kode Kategori	Nama Kategori	Aksi
1	MKN	Makanan	[Detail] [Edit] [Hapus]
2	PKN	Pakaian	[Detail] [Edit] [Hapus]
3	SNCK	Snack	[Detail] [Edit] [Hapus]
4	BHBB	Buah-bahan	[Detail] [Edit] [Hapus]
5	SYR	Sayuran	[Detail] [Edit] [Hapus]
6	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]

Tambah Data

Tambah Data Kategori

ID	Kode Kategori	Nama Kategori	Aksi
1	MKN		[Detail] [Edit] [Hapus]
2	PKN		[Detail] [Edit] [Hapus]
3	SNCK	Snack	[Detail] [Edit] [Hapus]
4	BHBB	Buah-bahan	[Detail] [Edit] [Hapus]
5	SYR	Sayuran	[Detail] [Edit] [Hapus]
6	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]

Edit Data

Edit Data Kategori

ID	Kode Kategori	Nama Kategori	Aksi
1	MKN		[Detail] [Edit] [Hapus]
2	PKN		[Detail] [Edit] [Hapus]
3	SNCK	Snack	[Detail] [Edit] [Hapus]
4	BHBB	Buah-bahan	[Detail] [Edit] [Hapus]
5	SYR	Sayuran	[Detail] [Edit] [Hapus]
6	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]
7	SPT	Sepatu	[Detail] [Edit] [Hapus]

Delete Data

Hapus Data Kategori

Konfirmasi !!!
Apakah Anda ingin menghapus data seperti di bawah ini?

Kode Kategori: SPT
Nama Kategori: Sepatu

[Batal] [Ya, Hapus]

ID	Kode Kategori	Nama Kategori	Aksi
1	MKN		[Detail] [Edit] [Hapus]
2	PKN		[Detail] [Edit] [Hapus]
3	SNCK	Snack	[Detail] [Edit] [Hapus]
4	BHBB	Buah-bahan	[Detail] [Edit] [Hapus]
5	SYR	Sayuran	[Detail] [Edit] [Hapus]
6	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]
7	SPT	Sepatu	[Detail] [Edit] [Hapus]



3. Tabel t_stok

- Views

Create_ajax.blade.php

```
<form action="{{ url('/stok/ajax') }}" method="POST" id="form-tambah">
    @csrf
    <div id="modal-master" class="modal-dialog modal-lg" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Tambah Data Stok</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&amptimes</span>
                </button>
            </div>
            <div class="modal-body">
                <div class="form-group">
                    <label>Nama Barang</label>
                    <select name="barang_id" id="barang_id" class="form-control" required>
                        <option value="">-- Pilih Barang --</option>
                        @foreach($barang as $b)
                            <option value="{{ $b->barang_id }}>{{ $b->barang_nama }}</option>
                        @endforeach
                    </select>
                    <small id="error-barang_id" class="error-text form-text text-danger"></small>
                </div>
                <div class="form-group">
                    <label>User</label>
                    <select name="user_id" id="user_id" class="form-control" required>
                        <option value="">-- Pilih User --</option>
                        @foreach($user as $u)
                            <option value="{{ $u->user_id }}>{{ $u->nama }}</option>
                        @endforeach
                    </select>
                    <small id="error-user_id" class="error-text form-text text-danger"></small>
                </div>
                <div class="form-group">
                    <label>Tanggal Stok</label>
                    <input value="" type="date" name="stok_tanggal" id="stok_tanggal" class="form-control" required>
                    <small id="error-stok_tanggal" class="error-text form-text text-danger"></small>
                </div>
                <div class="form-group">
                    <label>Jumlah Stok</label>
                    <input value="" type="number" name="stok_jumlah" id="stok_jumlah" class="form-control" required>
                    <small id="error-stok_jumlah" class="error-text form-text text-danger"></small>
                </div>
            </div>
            <div class="modal-footer">
                <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
                <button type="submit" class="btn btn-primary">Simpan</button>
            </div>
        </div>
    </div>
</form>
<script>
    $(document).ready(function() {
        $("#form-tambah").validate({
            rules: {
                barang_id: { required: true, number: true },
                user_id: { required: true, number: true },
                stok_tanggal: { required: true, date: true },
                stok_jumlah: { required: true, number: true, min: 1 }
            },
            submitHandler: function(form) {
                $.ajax({
                    url: form.action,
                    type: form.method,
                    data: $(form).serialize(),
                    success: function(response) {
                        if(response.status){
                            $('#myModal').modal('hide');
                            Swal.fire([
                                icon: 'success',
                                title: 'Berhasil',
                                text: response.message
                            ]);
                            dataStok.ajax.reload();
                        } else {
                            $('.error-text').text('');
                            $.each(response.msgField, function(prefix, val) {
                                $('#error-' + prefix).text(val[0]);
                            });
                            Swal.fire({
                                icon: 'error',
                                title: 'Terjadi Kesalahan',
                                text: response.message
                            });
                        }
                    },
                    error: function(xhr, status, error) {
                        Swal.fire({
                            icon: 'error',
                            title: 'Error AJAX',
                            text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
                        });
                        console.log('Error:', xhr.responseText);
                    }
                });
                return false;
            },
            errorElement: 'span',
            errorPlacement: function (error, element) {
                error.addClass('invalid-feedback');
                element.closest('.form-group').append(error);
            },
            highlight: function (element, errorClass, validClass) {
                $(element).addClass('is-invalid');
            },
            unhighlight: function (element, errorClass, validClass) {
                $(element).removeClass('is-invalid');
            }
        });
    });
</script>
```



Edit_ajax.blade.php

```
1  @extends('stok')
2  @section('content')
3      <div id="modal-master" class="modal-dialog modal-lg" role="document">
4          <div class="modal-content">
5              <div class="modal-header">
6                  <h5>{{ exampleModalLabel }}</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="alert alert-danger">
13                     <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
14                     Data yang anda cari tidak ditemukan
15                 </div>
16                 <a href="{{ url('/stok') }}" class="btn btn-warning">Kembali</a>
17             </div>
18         </div>
19     @else
20         <form action="{{ url('/stok/' . $stok->stok_id . '/update_ajax') }}" method="POST" id="form-edit">
21             @csrf
22             @method('PUT')
23             <div id="modal-master" class="modal-dialog modal-lg" role="document">
24                 <div class="modal-content">
25                     <div class="modal-header">
26                         <h5>{{ exampleModalLabel }}</h5>
27                         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28                             <span aria-hidden="true">&amptimes</span>
29                         </button>
30                     </div>
31                     <div class="modal-body">
32                         <div class="form-group">
33                             <label>Nama Barang:</label>
34                             <select name="barang_id" id="barang_id" class="form-control" required>
35                                 <option value="">> Pilih Barang </option>
36                                 @foreach($barang as $b)
37                                     <option value="{{ $b->barang_id }}> {{ $b->barang_nama }} </option>
38                                 @endforeach
39                             </select>
40                             <small id="error-barang_id" class="error-text form-text text-danger"></small>
41                         </div>
42                         <div class="form-group">
43                             <label>User:</label>
44                             <select name="user_id" id="user_id" class="form-control" required>
45                                 <option value="">> Pilih User </option>
46                                 @foreach($user as $u)
47                                     <option value="{{ $u->user_id }}> {{ $u->name }} </option>
48                                 @endforeach
49                             </select>
50                             <small id="error-user_id" class="error-text form-text text-danger"></small>
51                         </div>
52                         <div class="form-group">
53                             <label>Jumlah Stok:</label>
54                             <input type="number" name="stok_tanggal" id="stok_tanggal" class="form-control" value="{{ $stok->stok_tanggal }}" required>
55                             <small id="error-stok_tanggal" class="error-text form-text text-danger"></small>
56                         </div>
57                         <div class="form-group">
58                             <label>Jumlah:</label>
59                             <input type="number" name="stok_jumlah" id="stok_jumlah" class="form-control" value="{{ $stok->stok_jumlah }}" required>
60                             <small id="error-stok_jumlah" class="error-text form-text text-danger"></small>
61                         </div>
62                     </div>
63                     <div class="modal-footer">
64                         <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
65                         <button type="submit" class="btn btn-primary">Simpan</button>
66                     </div>
67                 </div>
68             </div>
69         </form>
70     </div>
71     </div>
72     </div>
73     </div>
74     </div>
75     <script>
76         $(document).ready(function() {
77             $('#form-edit').validate({
78                 rules: {
79                     barang_id: { required: true, number: true },
80                     user_id: { required: true, number: true },
81                     stok_tanggal: { required: true, date: true },
82                     stok_jumlah: { required: true, number: true, min: 1 }
83                 },
84                 submitHandler: function(form) {
85                     $.ajax({
86                         url: form.action,
87                         type: form.method,
88                         data: $(form).serialize(),
89                         success: function(response) {
90                             if (response.status) {
91                                 $('#myModal').modal('hide');
92                                 Swal.fire({
93                                     icon: 'success',
94                                     title: 'Berhasil',
95                                     text: response.message
96                                 });
97                                 dataStok.ajax.reload();
98                             } else {
99                                 $('.error-text').text('');
100                                 $.each(response.msgField, function(prefix, val) {
101                                     $('#error-' + prefix).text(val[0]);
102                                 });
103                                 Swal.fire({
104                                     icon: 'error',
105                                     title: 'Terjadi Kesalahan',
106                                     text: response.message
107                                 });
108                             }
109                         },
110                         error: function(xhr, status, error) {
111                             Swal.fire({
112                                 icon: 'error',
113                                 title: 'Error AJAX',
114                                 text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
115                             });
116                             console.log('Error:', xhr.responseText);
117                         }
118                     });
119                     return false;
120                 },
121                 errorElement: 'span',
122                 errorPlacement: function(error, element) {
123                     error.addClass('invalid-feedback');
124                     element.closest('.form-group').append(error);
125                 }
126             });
127         });
128     </script>
129 
```



Confirm_ajax.blade.php

```
1 @empty($stok)
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">
7                     <span aria-hidden="true">&amptimes;</span>
8                 </button>
9             </div>
10            <div class="modal-body">
11                <div class="alert alert-danger">
12                    <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
13                    Data yang anda cari tidak ditemukan
14                </div>
15                <a href="{{ url('/stok') }}" class="btn btn-warning">Kembali</a>
16            </div>
17        </div>
18    </div>
19 @else
20     <form action="{{ url('/stok/' . $stok->stok_id . '/delete_ajax') }}" method="POST" id="form-delete">
21         @csrf
22         @method('DELETE')
23         <div id="modal-master" class="modal-dialog modal-lg" role="document">
24             <div class="modal-content">
25                 <div class="modal-header">
26                     <h5 class="modal-title" id="exampleModallabel">Hapus Data Stok</h5>
27                     <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28                         <span aria-hidden="true">&amptimes;</span>
29                     </button>
30                 </div>
31                 <div class="modal-body">
32                     <div class="alert alert-warning">
33                         <h5><i class="icon fas fa-ban"></i> Konfirmasi !!!</h5>
34                         Apakah Anda ingin menghapus data seperti di bawah ini?
35                     </div>
36                     <table class="table table-sm table-bordered table-striped">
37                         <tr>
38                             <th class="text-right col-3">Nama Barang :</th>
39                             <td class="col-9">{{ $stok->barang->barang_nama }}{{ $stok->user->nama }}{{ $stok->stok_tanggal }}{{ $stok->stok_jumlah }}
```



- StokController

```
public function list(Request $request)
{
    $stok = StokModel::select('stok_id', 'barang_id', 'user_id', 'stok_tanggal', 'stok_jumlah')
        ->with(['barang', 'user']);

    if ($request->user_id) {
        $stok->where('user_id', $request->user_id);
    }

    // return DataTables::of($stok)
    //     ->addIndexColumn()
    //     ->addColumn('aksi', function ($stok) {
    //         $btn = '<a href="'.url('/stok/' . $stok->stok_id).'" class="btn btn-info btn-sm">Detail</a> ';
    //         $btn .= '<a href="'.url('/stok/' . $stok->stok_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
    //         $btn .= '<form class="d-inline-block" method="POST" action="'.url('/stok/' . $stok->stok_id).'">';
    //         $btn .= '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\')>Hapus</button></form>';
    //         return $btn;
    //     })
    // );
}

return DataTables::of($stok)
    ->addIndexColumn()
    ->addColumn('aksi', function ($stok) {
        $btn = '<button onclick="modalAction(\'' . url('/stok/' . $stok->stok_id . '/show_ajax').'\')"' . class="btn btn-info btn-sm">Detail</button> ';
        $btn .= '<button onclick="modalAction(\'' . url('/stok/' . $stok->stok_id . '/edit_ajax').'\')"' . class="btn btn-warning btn-sm">Edit</button> ';
        $btn .= '<button onclick="modalAction(\'' . url('/stok/' . $stok->stok_id . '/delete_ajax').'\')"' . class="btn btn-danger btn-sm">Hapus</button> ';
        return $btn;
    ))
);

->rawColumns(['aksi']);
->make(true);
}

public function create_ajax()
{
    $barang = BarangModel::select('barang_id', 'barang_nama')->get();
    $user = UserModel::select('user_id', 'nama')->get();
    return view('stok.create_ajax')
        ->with('barang', $barang)
        ->with('user', $user);
}

public function store_ajax(Request $request)
{
    // Cek apakah request berupa ajax
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'barang_id' => 'required|integer|exists:m_barang,barang_id',
            'user_id' => 'required|integer|exists:m_user,user_id',
            'stok_tanggal' => 'required|date',
            'stok_jumlah' => 'required|integer|min:1'
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors(),
            ]);
        }

        StokModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data stok berhasil disimpan'
        ]);
    }

    return redirect('/');
}

public function edit_ajax(string $id)
{
    $stok = StokModel::find($id);
    $barang = BarangModel::select('barang_id', 'barang_nama')->get();
    $user = UserModel::select('user_id', 'nama')->get();

    return view('stok.edit_ajax', ['stok' => $stok, 'barang' => $barang, 'user' => $user]);
}
```



```
public function update_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'barang_id'      => 'required|integer|exists:m_barang,barang_id',
            'user_id'        => 'required|integer|exists:m_user,user_id',
            'stok_tanggal'   => 'required|date',
            'stok_jumlah'    => 'required|integer|min:1'
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status'  => false, // respon json, true: berhasil, false: gagal
                'message' => 'Validasi gagal.',
                'msgField' => $validator->errors() // menunjukkan field mana yang error
            ]);
        }

        $check = StokModel::find($id);
        if ($check) {
            $check->update($request->all());
            return response()->json([
                'status'  => true,
                'message' => 'Data stok berhasil diupdate'
            ]);
        } else {
            return response()->json([
                'status'  => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}

public function confirm_ajax(string $id)
{
    $stok = StokModel::with(['barang', 'user'])->find($id);

    return view('stok.confirm_ajax', ['stok' => $stok]);
}

public function delete_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $stok = StokModel::find($id);
        if ($stok) {
            $stok->delete();
            return response()->json([
                'status'  => true,
                'message' => 'Data stok berhasil dihapus'
            ]);
        } else {
            return response()->json([
                'status'  => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}
```

- Route

```
Route::group(['prefix' => 'stok'], function () {
    Route::get('/', [StokController::class, 'index']);
    Route::post('/list', [StokController::class, 'list']);
    Route::get('/create', [StokController::class, 'create']);
    Route::post('/', [StokController::class, 'store']);
    Route::get('/create_ajax', [StokController::class, 'create_ajax']);
    Route::post('/ajax', [StokController::class, 'store_ajax']);
    Route::get('/{id}', [StokController::class, 'show']);
    Route::get('/{id}/edit', [StokController::class, 'edit']);
    Route::put('/{id}', [StokController::class, 'update']);
    Route::get('/{id}/edit_ajax', [StokController::class, 'edit_ajax']);
    Route::put('/{id}/update_ajax', [StokController::class, 'update_ajax']);
    Route::get('/{id}/delete_ajax', [StokController::class, 'confirm_ajax']);
    Route::delete('/{id}/delete_ajax', [StokController::class, 'delete_ajax']);
    Route::delete('/{id}', [StokController::class, 'destroy']);
});
```



ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
1	Mie Goreng	Administrator	2025-03-02 17:48:40	30	[Detail] [Edit] [Hapus]
2	Ayam Bakar	Administrator	2025-03-02 17:48:40	25	[Detail] [Edit] [Hapus]
3	Kaos Polos	Manager	2025-03-02 17:48:40	20	[Detail] [Edit] [Hapus]
4	Celana Jeans	Manager	2025-03-02 17:48:40	15	[Detail] [Edit] [Hapus]
5	Keripik Kentang	Manager	2025-03-02 17:48:40	10	[Detail] [Edit] [Hapus]
6	Biskuit Coklat	Staff/Kasir	2025-03-02 17:48:40	50	[Detail] [Edit] [Hapus]
7	Apel Fuji	Staff/Kasir	2025-03-02 17:48:40	40	[Detail] [Edit] [Hapus]
8	Jeruk Manis	Staff/Kasir	2025-03-02 17:48:40	35	[Detail] [Edit] [Hapus]
9	Wortel Segar	Administrator	2025-03-02 17:48:40	45	[Detail] [Edit] [Hapus]
10	Bayam Hijau	Staff/Kasir	2005-04-09 00:00:00	20	[Detail] [Edit] [Hapus]

Tambah Data

The screenshot shows a modal dialog titled 'Tambah Data Stok'. It contains fields for 'Nama Barang' (selected value: Bayam Hijau), 'User' (selected value: Staff/Kasir), and 'Tanggal Stok' (dd/mm/yyyy format). Below the modal is a table titled 'Daftar Stok' showing item details.

ID	Nama Barang	User	Tanggal Stok	Jumlah Stok
1	Mie Goreng	Administrator	2025-03-02 17:48:40	30
2	Ayam Bakar	Administrator	2025-03-02 17:48:40	25
3	Kaos Polos	Manager	2025-03-02 17:48:40	20
4	Celana Jeans	Manager	2025-03-02 17:48:40	15
5	Keripik Kentang	Manager	2025-03-02 17:48:40	10
6	Biskuit Coklat	Staff/Kasir	2025-03-02 17:48:40	50
7	Apel Fuji	Staff/Kasir	2025-03-02 17:48:40	40
8	Jeruk Manis	Staff/Kasir	2025-03-02 17:48:40	35
9	Wortel Segar	Administrator	2025-03-02 17:48:40	45
10	Bayam Hijau	Staff/Kasir	2005-04-09 00:00:00	20

Edit Data

The screenshot shows a modal dialog titled 'Edit Data Stok'. It contains fields for 'Nama Barang' (selected value: Bayam Hijau), 'User' (selected value: Staff/Kasir), and 'Tanggal Stok' (dd/mm/yyyy format). Below the modal is a table titled 'Daftar stok yang terdapat' showing item details.

ID	Nama Barang	User	Tanggal Stok	Jumlah Stok
1	Mie Goreng	Administrator	2025-03-02 17:48:40	30
2	Ayam Bakar	Administrator	2025-03-02 17:48:40	25
3	Kaos Polos	Manager	2025-03-02 17:48:40	20
4	Celana Jeans	Manager	2025-03-02 17:48:40	15
5	Keripik Kentang	Manager	2025-03-02 17:48:40	10
6	Biskuit Coklat	Staff/Kasir	2025-03-02 17:48:40	50
7	Apel Fuji	Staff/Kasir	2025-03-02 17:48:40	40
8	Jeruk Manis	Staff/Kasir	2025-03-02 17:48:40	35
9	Wortel Segar	Administrator	2025-03-02 17:48:40	45
10	Bayam Hijau	Staff/Kasir	2005-04-09 00:00:00	20

Delete Data

The screenshot shows a confirmation dialog box asking 'Konfirmasi!!! Apakah Anda ingin menghapus data seperti di bawah ini?'. It lists the item details: Nama Barang: Bayam Hijau, User: Staff/Kasir, Tanggal Stok: 2005-04-09 00:00:00, and Jumlah Stok: 20. Below the dialog is a table titled 'Daftar stok yang terdapat' showing item details.

ID	Nama Barang	User	Tanggal Stok	Jumlah Stok
1	Mie Goreng	Administrator	2025-03-02 17:48:40	30
2	Ayam Bakar	Administrator	2025-03-02 17:48:40	25
3	Kaos Polos	Manager	2025-03-02 17:48:40	20
4	Celana Jeans	Manager	2025-03-02 17:48:40	15
5	Keripik Kentang	Manager	2025-03-02 17:48:40	10
6	Biskuit Coklat	Staff/Kasir	2025-03-02 17:48:40	50
7	Apel Fuji	Staff/Kasir	2025-03-02 17:48:40	40
8	Jeruk Manis	Staff/Kasir	2025-03-02 17:48:40	35
9	Wortel Segar	Administrator	2025-03-02 17:48:40	45
10	Bayam Hijau	Staff/Kasir	2005-04-09 00:00:00	20



4. Tabel m_barang

- Views

Create_ajax.blade.php

```
1 <form action="{{ URL::route('ajax') }}" method="Post" id="Form-tambah">
2     @csrf
3     <div id="myModal" class="modal-dialog modal-lg" role="document">
4         <div class="modal-content">
5             <div class="modal-header">
6                 <h5 class="modal-title">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>Kategori</label>
14                    <select name="kategori_id" id="kategori_id" class="form-control" required>
15                        <option value="">Pilih Kategori</option>
16                        @foreach($category as $k)
17                            <option value="{{ $k->kategori_id }}">{ { $k->kategori_nama }}</option>
18                        @endforeach
19                    </select>
20                    <small id="error-kategori_id" class="error-text form-text text-danger"></small>
21                </div>
22                <div class="form-group">
23                    <label>Kode Barang</label>
24                    <input value="" type="text" name="barang_kode" id="barang_kode" class="form-control" required>
25                    <small id="error-barang_kode" class="error-text form-text text-danger"></small>
26                </div>
27                <div class="form-group">
28                    <label>Nama Barang</label>
29                    <input value="" type="text" name="barang_nama" id="barang_nama" class="form-control" required>
30                    <small id="error-barang_nama" class="error-text form-text text-danger"></small>
31                </div>
32                <div class="form-group">
33                    <label>Harga Beli</label>
34                    <input value="" type="number" name="harga_beli" id="harga_beli" class="form-control" step="1" required>
35                    <small id="error-harga_beli" class="error-text form-text text-danger"></small>
36                </div>
37                <div class="form-group">
38                    <label>Harga Jual</label>
39                    <input value="" type="number" name="harga_jual" id="harga_jual" class="form-control" step="1" required>
40                    <small id="error-harga_jual" 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-tambah").validate({
52                rules: {
53                    kategori_id: { required: true, number: true },
54                    barang_kode: { required: true, minlength: 3, maxlength: 20 },
55                    barang_nama: { required: true, minlength: 3, maxlength: 100 },
56                    harga_beli: { required: true, digits: true, min: 1 },
57                    harga_jual: { required: true, digits: true, min: 1 }
58                },
59                submitHandler: function(form) {
60                    $.ajax({
61                        url: form.action,
62                        type: form.method,
63                        data: $(form).serialize(),
64                        success: function(response) {
65                            if(response.status){
66                                $('#myModal').modal('hide');
67                                Swal.fire({
68                                    icon: 'success',
69                                    title: 'Berhasil',
70                                    text: response.message
71                                });
72                                dataBarang.ajax.reload();
73                            } else {
74                                $('._error-text').text('');
75                                $.each(response.msgField, function(prefix, val) {
76                                    $('##error-' + prefix).text(val[0]);
77                                });
78                                Swal.fire({
79                                    icon: 'error',
80                                    title: 'Terjadi Kesalahan',
81                                    text: response.message
82                                });
83                            }
84                        },
85                        error: function(xhr, status, error) {
86                            Swal.fire({
87                                icon: 'error',
88                                title: 'Error AJAX',
89                                text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
90                            });
91                            console.log('Error:', xhr.responseText);
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    </script>
```



Edit_ajax.blade.php

```
1 <?php $barang>
2 <?php if($barang){ ?>
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">Kesalahan</h5>
7 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8 | <span aria-hidden="true">&times;</span>
9 </button>
10 </div>
11 <div class="modal-body">
12 <div class="alert alert-danger">
13 | <h5><i class="fa fa-ban"></i> Kesalahan!!!</h5>
14 | Data yang anda cari tidak ditemukan
15 | <a href="{{ url('/barang') }}" class="btn btn-warning">Kembali</a>
16 </div>
17 </div>
18 </div>
19 <?php else{ ?>
20 <form action="{{ url('/barang/' . $barang->barang_id . '/update_ajax') }}" method="POST" id="form-edit">
21 <script>
22 <method("PUT")>
23 <div id="modal-master" class="modal-dialog modal-lg" role="document">
24 <div class="modal-content">
25 <div class="modal-header">
26 | <h5 class="modal-title" id="exampleModalLabel">Edit Data Barang</h5>
27 | <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28 | | <span aria-hidden="true">&times;</span>
29 </button>
30 </div>
31 <div class="modal-body">
32 <div class="form-group">
33 | <label>Kategori</label>
34 | <select name="kategori_id" id="kategori_id" class="form-control" required>
35 | | <option value=""> Pilih Kategori </option>
36 | <foreach $kategori as $k >
37 | | <option value="{{ $k->kategori_id }}> {{ $k->kategori_nama }}</option>
38 | </foreach>
39 </select>
40 </div>
41 <small id="error-kategori_id" class="error-text form-text text-danger"></small>
42 </div>
43 <div class="form-group">
44 | <label>Kode Barang</label>
45 <input type="text" name="barang_kode" id="barang_kode" class="form-control" value="{{ $barang->barang_kode }}" required>
46 <small id="error-barang_kode" class="error-text form-text text-danger"></small>
47 </div>
48 <div class="form-group">
49 | <label>Nama Barang</label>
50 | <input type="text" name="barang_nama" id="barang_nama" class="form-control" value="{{ $barang->barang_nama }}" required>
51 <small id="error-barang_nama" class="error-text form-text text-danger"></small>
52 </div>
53 <div class="form-group">
54 | <label>Harga Beli</label>
55 | <input type="number" name="harga_beli" id="harga_beli" class="form-control" value="{{ $barang->harga_beli }}" step="1" required>
56 <small id="error-harga_beli" class="error-text form-text text-danger"></small>
57 </div>
58 <div class="form-group">
59 | <label>Harga Jual</label>
60 | <input type="number" name="harga_jual" id="harga_jual" class="form-control" value="{{ $barang->harga_jual }}" step="1" required>
61 <small id="error-harga_jual" class="error-text form-text text-danger"></small>
62 </div>
63 </div>
64 <div class="modal-footer">
65 | <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
66 | <button type="submit" class="btn btn-primary">Simpan</button>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 <script>
75 $(document).ready(function() {
76   $("#form-edit").validate({
77     rules: {
78       kategori_id: { required: true, number: true },
79       barang_kode: { required: true, minlength: 3, maxlength: 20 },
80       barang_nama: { required: true, minlength: 3, maxlength: 100 },
81       harga_beli: { required: true, digits: true, min: 1 },
82       harga_jual: { required: true, digits: true, min: 1 }
83     },
84     submitHandler: function(form) {
85       $.ajax({
86         url: form.action,
87         type: form.method,
88         data: $(form).serialize(),
89         success: function(response) {
90           if (response.status) {
91             $('#myModal').modal('hide');
92             Swal.fire({
93               icon: 'success',
94               title: 'Berhasil',
95               text: response.message
96             });
97             dataBarang.ajax.reload();
98           } else {
99             $('.error-text').text('');
100             $.each(response.msgField, function(prefix, val) {
101               $('#error-' + prefix).text(val[0]);
102             });
103             Swal.fire({
104               icon: 'error',
105               title: 'Terjadi Kesalahan',
106               text: response.message
107             });
108           },
109           error: function(xhr, status, error) {
110             Swal.fire({
111               icon: 'error',
112               title: 'Error AJAX',
113               text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
114             });
115             console.log('Error:', xhr.responseText);
116           }
117         });
118         return false;
119       },
120       errorElement: 'span',
121       errorPlacement: function(error, element) {
122         error.addClass('invalid-feedback');
123         element.closest('.form-group').append(error);
124       },
125     }
126   })
127 })
```



Confirm_ajax.blade.php

```
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!!!</h5>
6         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
7           <span aria-hidden="true">&times;Kembali</a>
16      </div>
17    </div>
18  </div>
19 @else
20   <form action="{{ url('/barang/' . $barang->barang_id . '/delete_ajax') }}" method="POST" id="form-delete">
21     @csrf
22     @method('DELETE')
23   <div id="modal-master" class="modal-dialog modal-lg" role="document">
24     <div class="modal-content">
25       <div class="modal-header">
26         <h5 class="modal-title" id="exampleModallabel">Hapus Data Barang</h5>
27         <button type="button" class="close" data-dismiss="modal" aria-label="Close">
28           <span aria-hidden="true">&times;{{ $barang->kategori->kategori_nama }}</td>
40           </tr>
41           <tr>
42             <th class="text-right col-3">Kode Barang :</th>
43             <td class="col-9">{{ $barang->barang_kode }}</td>
44           </tr>
45           <tr>
46             <th class="text-right col-3">Nama Barang :</th>
47             <td class="col-9">{{ $barang->barang_nama }}</td>
48           </tr>
49           <tr>
50             <th class="text-right col-3">Harga Beli :</th>
51             <td class="col-9">{{ $barang->harga_beli }}</td>
52           </tr>
53           <tr>
54             <th class="text-right col-3">Harga Jual :</th>
55             <td class="col-9">{{ $barang->harga_jual }}</td>
56           </tr>
57         </table>
58       <div class="modal-footer">
59         <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
60         <button type="submit" class="btn btn-primary">Ya, Hapus</button>
61       </div>
62     </div>
63   </div>
64 </form>
65 </script>
66 $(document).ready(function() {
67   $('#form-delete').validate({
68     rules: {},
69     submitHandler: function(form) {
70       $.ajax({
71         url: form.action,
72         type: form.method,
73         data: $(form).serialize(),
74         success: function(response) {
75           if (response.status) {
76             $('#myModal').modal('hide');
77             Swal.fire({
78               icon: 'success',
79               title: 'Berhasil',
80               text: response.message
81             });
82             dataBarang.ajax.reload();
83           } else {
84             $('.error-text').text('');
85             $.each(response.msgField, function(prefix, val) {
86               $('#error-' + prefix).text(val[0]);
87             });
88             Swal.fire({
89               icon: 'error',
90               title: 'Terjadi Kesalahan',
91               text: response.message
92             });
93           }
94         },
95         error: function(xhr, status, error) {
96           Swal.fire({
97             icon: 'error',
98             title: 'Error AJAX',
99             text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
100           });
101           console.log('Error:', xhr.responseText);
102         }
103       });
104       return false;
105     },
106     errorElement: 'span',
107     errorPlacement: function(error, element) {
108       error.addClass('invalid-feedback');
109       element.closest('.form-group').append(error);
110     },
111     highlight: function(element, errorClass, validClass) {
112       $(element).addClass('is-invalid');
113     },
114     unhighlight: function(element, errorClass, validClass) {
115       $(element).removeClass('is-invalid');
116     }
117   });
118 });
119 </script>
120 @endif
```



- BarangController

```
public function list(Request $request)
{
    $barang = BarangModel::select('barang_id', 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
        ->with('kategori');

    if ($request->kategori_id) {
        $barang->where('kategori_id', $request->kategori_id);
    }

    // return DataTables::of($barang)
    //     ->addIndexColumn()
    //     ->addColumn('aksi', function ($barang) {
    //         $btn = '<a href="'.url('/barang/' . $barang->barang_id).'" class="btn btn-info btn-sm">Detail</a> ';
    //         $btn .= '<a href="'.url('/barang/' . $barang->barang_id . '/edit').'" class="btn btn-warning btn-sm">Edit</a> ';
    //         $btn .= '<form class="d-inline-block" method="POST" action="'.url('/barang/' . $barang->barang_id).'">';
    //         $btn .= '<button type="submit" class="btn btn-danger btn-sm" onclick="return confirm(\'Apakah Anda yakin menghapus data ini?\')">>Hapus</button></form>';
    //     });
    //     return $btn;
    // });

    return DataTables::of($barang)
        ->addIndexColumn()
        ->addColumn('aksi', function ($barang) {
            $btn = '<button onclick="modalAction(\"'.url('/barang/' . $barang->barang_id . '/show_ajax').'\")" class="btn btn-info btn-sm">Detail</button> ';
            $btn .= '<button onclick="modalAction(\"'.url('/barang/' . $barang->barang_id . '/edit_ajax').'\")" class="btn btn-warning btn-sm">Edit</button> ';
            $btn .= '<button onclick="modalAction(\"'.url('/barang/' . $barang->barang_id . '/delete_ajax').'\")" class="btn btn-danger btn-sm">Hapus</button> ';
            return $btn;
        });
    });

    ->rawColumns(['aksi']);
    ->make(true);
}

public function create_ajax()
{
    $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();
    return view('barang.create_ajax')
        ->with('kategori', $kategori);
}

public function store_ajax(Request $request)
{
    // Cek apakah request berupa ajax
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'kategori_id' => 'required|integer|exists:m_kategori,kategori_id',
            'barang_kode' => 'required|string|min:3|max:20|unique:m_barang,barang_kode',
            'barang_nama' => 'required|string|max:100',
            'harga_beli' => 'required|integer|min:1',
            'harga_jual' => 'required|integer|min:1'
        ];

        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors(),
            ]);
        }

        BarangModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data barang berhasil disimpan'
        ]);
    }

    return redirect('/');
}

public function edit_ajax(string $id)
{
    $barang = BarangModel::find($id);
    $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();

    return view('barang.edit_ajax', ['barang' => $barang, 'kategori' => $kategori]);
}
```



```
public function update_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'kategori_id' => 'required|integer|exists:m_kategori,kategori_id',
            'barang_kode' => 'required|string|min:3|max:20|unique:m_barang,barang_kode,' . $id . ',barang_id',
            'barang_nama' => 'required|string|max:100',
            'harga_beli' => 'required|integer|min:1',
            'harga_jual' => 'required|integer|min:1'
        ];
        $validator = Validator::make($request->all(), $rules);

        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi gagal.',
                'msgField' => $validator->errors()
            ]);
        }

        $check = BarangModel::find($id);
        if ($check) {
            $check->update($request->all());
            return response()->json([
                'status' => true,
                'message' => 'Data barang berhasil diupdate'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}

public function confirm_ajax(string $id)
{
    $barang = BarangModel::with('kategori')->find($id);

    return view('barang.confirm_ajax', ['barang' => $barang]);
}

public function delete_ajax(Request $request, $id)
{
    if ($request->ajax() || $request->wantsJson()) {
        $barang = BarangModel::find($id);
        if ($barang) {
            $barang->delete();
            return response()->json([
                'status' => true,
                'message' => 'Data barang berhasil dihapus'
            ]);
        } else {
            return response()->json([
                'status' => false,
                'message' => 'Data tidak ditemukan'
            ]);
        }
    }
    return redirect('/');
}
```

- Route

```
Route::group(['prefix' => 'barang'], function () {
    Route::get('/', [BarangController::class, 'index']);
    Route::post('/list', [BarangController::class, 'list']);
    Route::get('/create', [BarangController::class, 'create']);
    Route::post('/', [BarangController::class, 'store']);
    Route::get('/create_ajax', [BarangController::class, 'create_ajax']);
    Route::post('/ajax', [BarangController::class, 'store_ajax']);
    Route::get('/{id}', [BarangController::class, 'show']);
    Route::get('/{id}/edit', [BarangController::class, 'edit']);
    Route::put('/{id}', [BarangController::class, 'update']);
    Route::get('/{id}/edit_ajax', [BarangController::class, 'edit_ajax']);
    Route::put('/{id}/update_ajax', [BarangController::class, 'update_ajax']);
    Route::get('/{id}/delete_ajax', [BarangController::class, 'confirm_ajax']);
    Route::delete('/{id}/delete_ajax', [BarangController::class, 'delete_ajax']);
    Route::delete('/{id}', [BarangController::class, 'destroy']);
});
```



ID	Kode Barang	Nama Barang	Kategori	Harga Beli	Harga Jual	Aksi
1	BRG002	Mie Goreng	Makanan	9000	14000	[Detail] [Edit] [Hapus]
2	BRG003	Apel Fuji	Buah-buahan	12000	18000	[Detail] [Edit] [Hapus]
3	BRG004	Kaos Polos	Pakaian	35000	45000	[Detail] [Edit] [Hapus]
4	BRG005	Celana Jeans	Pakaian	20000	30000	[Detail] [Edit] [Hapus]
5	BRG006	Keripik Kentang	Snack	8000	12000	[Detail] [Edit] [Hapus]
6	BRG007	Riskuit Coklat	Snack	5000	8000	[Detail] [Edit] [Hapus]
7	BRG008	Apel Fuji	Buah-buahan	15000	20000	[Detail] [Edit] [Hapus]
8	BRG009	Jeruk Manis	Buah-buahan	10000	15000	[Detail] [Edit] [Hapus]
9	BRG010	Wortel Segar	Sayuran	7000	10000	[Detail] [Edit] [Hapus]
10	BRG011	Bayam Hijau	Sayuran	5000	8000	[Detail] [Edit] [Hapus]

Tambah Data

Tambah Data Barang

Kategori: Makanan

Kode Barang: BRG002

Nama Barang: Mie Goreng

Harga Beli: 9000

Harga Jual: 14000

Aksi: [Detail] [Edit] [Hapus]

Edit Data

Edit Data Barang

Kategori: Makanan

Kode Barang: BRG002

Nama Barang: Mie Goreng

Harga Beli: 9000

Harga Jual: 14000

Aksi: [Detail] [Edit] [Hapus]

Delete Data

Hapus Data Barang

Konfirmasi!!!
Apakah Anda ingin menghapus data seperti di bawah ini?

Kategori: Sayuran

Kode Barang: BRG011

Nama Barang: Bayam Hijau

Harga Beli: 5000

Harga Jual: 8000

Aksi: [Detail] [Edit] [Hapus]