

## MODUL MINGGU 9 – IMPLEMENTASI AJAX DATATABLES

### 1.1 Tujuan Instruksi Khusus

Setelah mengikuti modul ini, mahasiswa diharapkan mampu:

- a. Mampu menggunakan library Datatables dengan metode AJAX
- b. Mampu membuat fitur filter data menggunakan AJAX Datatables

### 1.2 Instruksi Modul

#### 1.2.1 Setup project

- 1) Siapkan kembali project **cart\_online** yang sudah dibuat sebelumnya menggunakan framework Laravel
- 2) Siapkan file **.env** beserta konfigurasi databasenya (jika belum)
- 3) Jangan lupa menyalakan Apache dan Mysql pada XAMPP
- 4) Buka file manager dan masuk ke folder project kalian, klik kanan dan pilih **open git bash here** (pada windows 11 klik **show more option** terlebih dahulu)
- 5) Update project dengan kode baru dari repo dengan perintah pada terminal/git bash :

```
git pull old-origin main
```

- 6) Proses download dan menyatukan kode akan terjadi
- 7) Lakukan git push untuk mengupdate repo kalian masing-masing

#### 1.2.2 Implementasi Datatables

- 1) Ekstrak file **Datatables.zip** (gunakan extract to folder, bukan extract here)
- 2) Copy folder Datatables ke project **cart\_online** kalian tepatnya di dalam folder **public**
- 3) Buka file **Controllers Produk.php**, lalu buat fungsi API Datatables sebagai berikut :

```
public function dataDatables(Request $request)
{
    $search = ???;
    $order = ???;

    switch (???) {
        case '0':
            $orderby = '???';
            break;
```

```
        case '1':
            $orderby = '???';
            break;

        case '2':
            $orderby = '???';
            break;

        case '3':
            $orderby = '???';
            break;

        case '4':
            $orderby = '???';
            break;

        default:
            $orderby = 'id_produk';
            break;
    }

    $data_db_total = ???
    $data_db_filtered = ???

    $data_db = ???

    $data_formatted = [];

    foreach ($data_db as $key => $value) {
        $url_edit = url("produk/form"). '/' . $value->id_produk;;
        $seventhapus = 'onclick="hapusData('.$value->id_produk.')"';
        $seventaddtocart = 'onclick="add_to_cart('.$value->id_produk.')"';
        $action = '<a href="'.$url_edit.'" class="btn btn-info">Ubah</a> <a href="#"
        '.$seventhapus.'" class="btn btn-danger">Hapus</a> <hr/> <a href="#" '.$seventaddtocart.'"
        class="btn btn-warning">Tambahkan Ke Keranjang</a>';

        $harga = 'Rp '.number_format($value->harga);
        $image = '';

        $row_data = [];
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;
        $row_data[] = ???;

        $data_formatted[] = $row_data;
    }

    $data_json = [
        'draw' => ???,
        'recordsTotal' => ???,
```

```
'recordsFiltered' => ???,
'data' => ???
];

return json_encode($data_json);
}
```

Ubah Kode “???” sehingga kodenya menjadi sebagai berikut :

```
202 public function dataDatables(Request $request)
203 {
204     $search = $request->query('search');
205     $order = $request->query('order');
206
207     switch ($order[0]['column']) {
208         case '0':
209             $orderby = 'id_produk';
210             break;
211
212         case '1':
213             $orderby = 'kode_produk';
214             break;
215
216         case '2':
217             $orderby = 'nama_produk';
218             break;
219
220         case '3':
221             $orderby = 'stok';
222             break;
223
224         case '4':
225             $orderby = 'harga';
226             break;
227
228         default:
229             $orderby = 'id_produk';
230             break;
231     }
232
233     $data_db_total = Produkmodel::all();
234     $data_db_filtered = Produkmodel::where('nama_produk', 'like', '%'.$search['value'].'%')->
        get();
235
236     $data_db = Produkmodel::where('nama_produk', 'like', '%'.$search['value'].'%')
237     ->offset($request->query('start'))
238     ->limit($request->query('length'))
239     ->orderByRaw($orderby.' '.$order[0]['dir'])
240     ->get(['produk.*']);
241
242
243     $data_formatted = [];
244
245     foreach ($data_db as $key => $value) {
246         $url_edit = url("produk/form"). '/' . $value->id_produk;;
247         $eventHapus = 'onclick="hapusData('.$value->id_produk.')"';
248         $event_add_to_cart = 'onclick="add_to_cart('.$value->id_produk.')"';
249         $action = '<a href="'.$url_edit.'" class="btn btn-info">Ubah</a> <a href="#" '.$
            eventHapus.' class="btn btn-danger">Hapus</a> <hr/> <a href="#" '.$
            event_add_to_cart.' class="btn btn-warning">Tambahkan Ke Keranjang</a>';
250

```

```

251         $harga = 'Rp '.number_format($value->harga);
252         $image = '';
253
254         $row_data = [];
255         $row_data[] = $key+1;
256         $row_data[] = $value->kode_produk;
257         $row_data[] = $value->nama_produk;
258         $row_data[] = $value->stok;
259         $row_data[] = $harga;
260         $row_data[] = $value->deskripsi;
261         $row_data[] = $image;
262         $row_data[] = $action;
263
264         $data_formatted[] = $row_data;
265     }
266
267     $data_json = [
268         'draw' => $request->query('draw'),
269         'recordsTotal' => count($data_db_total),
270         'recordsFiltered' => count($data_db_filtered),
271         'data' => $data_formatted
272     ];
273
274     return json_encode($data_json);
275 }

```

- 4) Buka file **Views produk/list.blade.php**, dan ubah kodenya menjadi berikut :

```

@extends('template/layout')

@section('judul')
    List Produk
@endsection

@section('konten')
<link rel="stylesheet" type="text/css" href="{{ url('DataTables/DataTables-1.10.25/css/dataTables.bootstrap4.min.css') }}">

<form>
    <div class="row">
        <div class="col">
            <label>Masukkan Kode</label>
            <input class="form-control" type="text" name="cari" id="cari">
        </div>
        <div class="col">
            <input type="submit" value="Cari ID" class="btn btn-primary" style="margin-top:33px">
        </div>
    </div>
</form>

<table border="1" id="data-list" class="table">
    <thead>
        <tr>
            <th>No.</th>
            <th>Kode</th>
            <th>Nama</th>
            <th>Stok</th>
            <th>Harga</th>

```

```
<th>Deskripsi</th>
<th>Gambar</th>
<th>OPSI</th>
</tr>
</thead>
</table>
@endsection

@section('script_custom')
<script>
    function hapusData(idProduk) {
        var url_post = '{{ url("api/produk/hapus") }}';

        $.ajax(url_post, {
            type: 'DELETE',
            data: {'id' : idProduk},
            success: function (data, status, xhr) {
                var data_str = JSON.parse(data);

                alert(data_str['pesan']);
                ambil_data();
            },
            error: function (jqXHR, textStatus, errorMessage) {
                alert('Error : ' + jqXHR.responseText.message);
            }
        })
    }

    function add_to_cart(idProduk) {
        var url_post = '{{ url("api/keranjang") }}';

        $.ajax(url_post, {
            type: 'POST',
            data: {'id_produk' : idProduk},
            success: function (data, status, xhr) {
                var data_str = JSON.parse(data);

                alert(data_str['pesan']);
            },
            error: function (jqXHR, textStatus, errorMessage) {
                alert('Error : ' + jqXHR.responseText.message);
            }
        })
    }

    function ambil_data(kode = "") {
        if (kode == "") {
            var link = '{{ url("api/produk/all") }}';
        } else {
            var link = '{{ url("api/produk/kode/") }}' + '/' + kode;
        }

        $.ajax(link, {
            type: 'GET',
            success : function (data, status, xhr) {
```

```
resetTable();
var objData = JSON.parse(data);
var dataTable = "";
$.each( objData, function( key, value ) {
    var url_edit = '{{ url("produk/form") }}' + '/' + value.id_produk;
    var eventHapus = 'onclick="hapusData('+value.id_produk+')"';
    var event_add_to_cart = 'onclick="add_to_cart('+value.id_produk+')"';

    dataTable += '<tr>';
    dataTable += '<td>'+(key+1)+'</td>';
    dataTable += '<td>'+value.kode_produk+'</td>';
    dataTable += '<td>'+value.nama_produk+'</td>';
    dataTable += '<td>'+value.stok+'</td>';
    dataTable += '<td>Rp '+value.harga+'</td>';
    dataTable += '<td>'+value.deskripsi+'</td>';
    dataTable += '<td></td>';
    dataTable += '<td><a href="'+url_edit+'" class="btn btn-info">Ubah</a> <a href="#" '+eventHapus+' class="btn btn-danger">Hapus</a> <hr/> <a href="#" '+event_add_to_cart+' class="btn btn-warning">Tambahkan Ke Keranjang</a></td>';
    dataTable += '</tr>';
});

$( "#data-list" ).append( dataTable );
},
error : function (jqXHR, textStatus, errorMsg) {
    alert('Error Pengambilan Data : ' + errorMsg);
}
})
}

// ambil_data();

$("form").on('submit', function(e){
    e.preventDefault();
    var kode = $("input[name=cari]").val();
    // ambil_data(kode);
})

function resetTable() {
    $( "#data-list" ).html( "<thead> <tr> <th>No.</th> <th>Kode</th> <th>Nama</th> <th>Stok</th> <th>Harga</th> <th>Deskripsi</th> <th>Gambar</th> <th>OPSI</th> </tr> </thead>" );
}
</script>
@endsection
```

5) Masih di file yang sama, pada Section **script\_custom** tambahkan kode berikut :

```
<script type="text/javascript" src="{{ url('DataTables/datatables.min.js') }}"></script>

<script type="text/javascript">
    var url = '{{ url("api/produk/dataTable") }}';

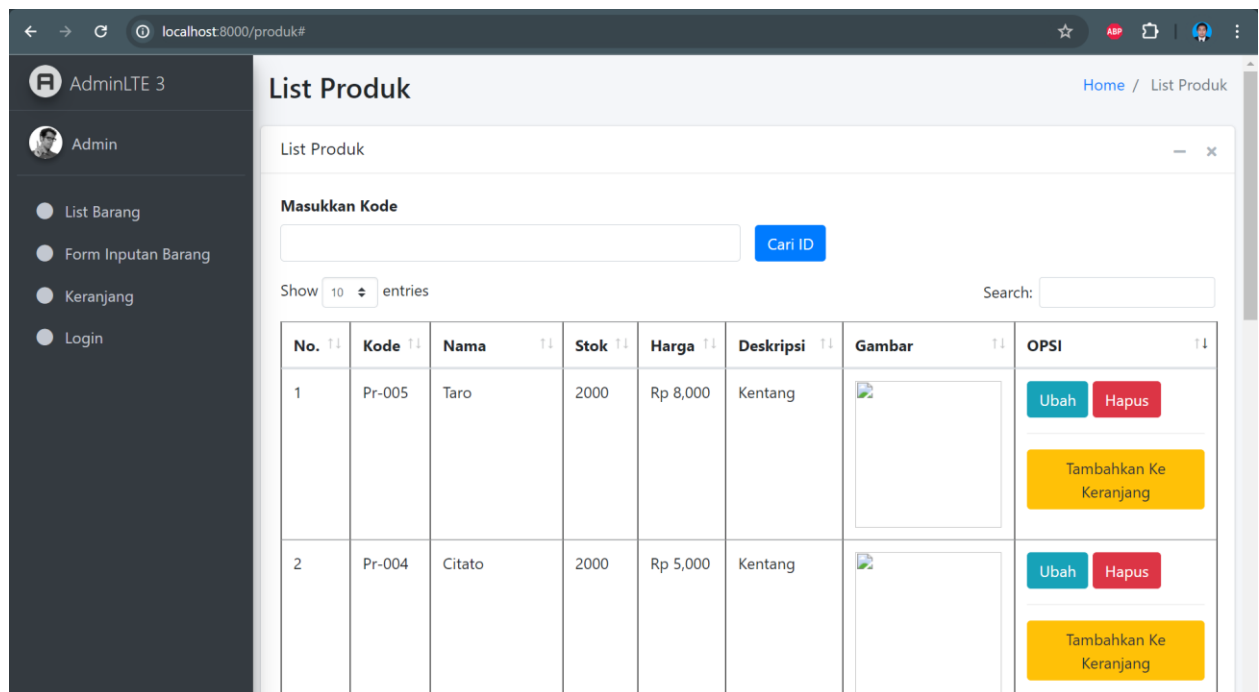
    var tabel = $("#data-list").DataTable({
        "processing": true,
```

```
"serverSide": true,  
"ajax": {  
    url: url,  
    data: function (d) {  
  
    }  
},  
});  
</script>
```

- 6) Buka file **routes/api.php** dan tambahkan kode route baru di akhir file :

```
Route::get('/produk/dataTable', [Produk::class, 'dataDatables']);
```

- 7) Lakukan Git add, Git Commit, dan Git Push  
8) Jalankan php artisan serve melalui command prompt dan ujicoba halaman produk



### 1.2.3 Custom Pencarian

- 1) Buka file Views produk/list.blade.php, ubah kode formnya menjadi seperti berikut :

```
<form>  
    <div class="row">  
        <div class="col">  
            <label>Masukkan Kode</label>  
            <label>Masukkan Batas Maksimal Stok</label>  
            <input class="form-control" type="text" name="cari" id="cari">  
        </div>  
        <div class="col">
```

```
<input type="submit" value="Cari ID" class="btn btn-primary" style="margin-top:33px">
<input type="submit" value="Cari Produk" class="btn btn-primary" style="margin-top:33px"
id="">
</div>
</div>
</form>
```

- 2) Masih di file yang sama, ubah fungsi pemanggilan ajaxnya menjadi seperti berikut :

```
<script type="text/javascript">
    var url = '{{ url("api/produk/dataTable") }}';

    var tabel = $("#data-list").DataTable({
        "processing": true,
        "serverSide": true,
        "ajax": {
            url: url,
            data: function (d) {
                d.stok = $("#cari").val();
            }
        },

    });

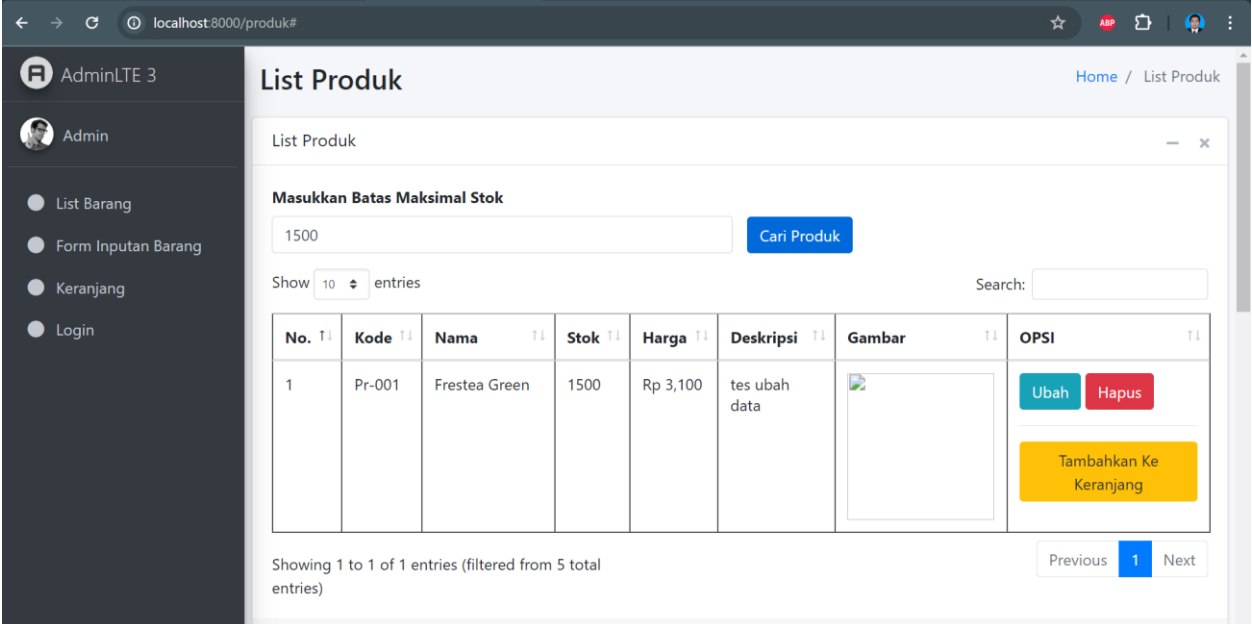
    $("form").on('submit', function(e){
        e.preventDefault();
        tabel.ajax.reload();
    })
</script>
```

- 3) Buka file **Controllers Produk.php** dan ubah kodenya menjadi seperti berikut :

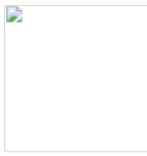
```
205         $order = $request->query('order');
206         $search_stok = $request->query('stok');
207
208         switch ($order[0]['column']) {
209             case '0':
210
211
212
213
214             $data_db_total = Produkmodel::all();
215             $data_db_filtered = Produkmodel::where('nama_produk', 'like', '%'.$search['value'].'%');
216
217             if ($search_stok != '' && $search_stok != null) {
218                 $data_db_filtered = $data_db_filtered->where('stok', '<=', $search_stok);
219             }
220
221             $data_db_filtered = $data_db_filtered->get();
222
223             $data_db = Produkmodel::where('nama_produk', 'like', '%'.$search['value'].'%');
224
225             if ($search_stok != '' && $search_stok != null) {
226                 $data_db = $data_db->where('stok', '<=', $search_stok);
227             }
228
229             $data_db = $data_db->offset($request->query('start'))
230             ->limit($request->query('length'))
231             ->orderByRaw($orderby.' '.$order[0]['dir'])
232             ->get(['produk.*']);
```



- 4) Lakukan Git add, Git Commit, dan Git Push
- 5) Jalankan php artisan serve melalui command prompt dan ujicoba halaman produk



The screenshot displays the 'List Produk' page in AdminLTE 3. The page features a sidebar with navigation links: 'List Barang', 'Form Inputan Barang', 'Keranjang', and 'Login'. The main content area includes a search bar with the text 'Masukkan Batas Maksimal Stok' and a 'Cari Produk' button. Below the search bar, there is a table with the following columns: No., Kode, Nama, Stok, Harga, Deskripsi, Gambar, and OPSI. The table contains one entry with the following data:

No.	Kode	Nama	Stok	Harga	Deskripsi	Gambar	OPSI
1	Pr-001	Frestea Green	1500	Rp 3,100	tes ubah data		<a href="#">Ubah</a> <a href="#">Hapus</a> <a href="#">Tambahkan Ke Keranjang</a>

Below the table, it states 'Showing 1 to 1 of 1 entries (filtered from 5 total entries)'. The page also includes a pagination control with 'Previous', '1', and 'Next' buttons.