

Laporan Pemrograman Web Lanjut

Jobsheet 6 : Ajax Form (AdminLTE) dan Client Validation

Dosen Pengampu : Dimas Wahyu Wibowo, ST.,MT.



Disusun Oleh:

Queenadhynar Azarine Dwipa A.

2341760109

SIB 2B

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2023/2024

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.

```
<!-- SweetAlert2 -->
<link rel="stylesheet" href="{{ asset('adminlte/plugins/sweetalert2-theme-bootstrap-4/bootstrap4.min.css') }}">
```

```
<!-- jquery-validation -->
<script src="{{ asset('adminlte/plugins/jquery-validation/jquery.validate.min.js') }}"></script>
<script src="{{ asset('adminlte/plugins/jquery-validation/additional-methods.min.js') }}"></script>

<!-- SweetAlert2 -->
<script src="{{ asset('adminlte/plugins/sweetalert2/sweetalert2.min.js') }}"></script>
```

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

```
Minggu 6 > PWL_POS > resources > views > user > index.blade.php > script > <function>
1  @extends('layouts.template')
2
3  @section('content')
4  <div class="card card-outline card-primary">
5      <div class="card-header">
6          <h3 class="card-title">{{ $page->title }}</h3>
7          <div class="card-tools">
8              <a class="btn btn-sm btn-primary mt-1" href="{{ url('user/create') }}">Tambah</a>
9              <button onclick="modalAction('{{ url('user/create_ajax') }}')" class="btn btn-sm btn-succes mt-1"> Tambah Ajax</button>
10             </div>
11         </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-succes 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

```

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

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

```

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

```

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

```

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

```
Minggu 6 > PWL_POS > resources > views > user > create_ajax.blade.php > script
1  <form action="{{ url('/user/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 User</h5>
7                  <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
8                      aria-hidden="true">&times;</span></button>
9              </div>
10             <div class="modal-body">
11                 <div class="form-group">
12                     <label>Level Pengguna</label>
13                     <select name="level_id" id="level_id" class="form-control" required>
14                         <option value="">- Pilih Level -</option>
15                         @foreach ($level as $l)
16                             <option value="{{ $l->level_id }}">{{ $l->level_nama }}</option>
17                         @endforeach
18                     </select>
19                     <small id="error-level_id" class="error-text form-text text-danger"></small>
20                 </div>
21                 <div class="form-group">
22                     <label>Username</label>
23                     <input value="" type="text" name="username" id="username" class="form-control"
24                         required>
25                     <small id="error-username" class="error-text form-text text-danger"></small>
26                 </div>
27                 <div class="form-group">
28                     <label>Nama</label>
29                     <input value="" type="text" name="nama" id="nama" class="form-control"
30                         required>
31                     <small id="error-nama" class="error-text form-text text-danger"></small>
32                 </div>
33                 <div class="form-group">
34                     <label>Password</label>
35                     <input value="" type="password" name="password" id="password" class="form-control"
36                         required>
37                     <small id="error-password" class="error-text form-text text-danger"></small>
38                 </div>
39             </div>
40             <div class="modal-footer">
41                 <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
42                 <button type="submit" class="btn btn-primary">Simpan</button>
43             </div>
44         </div>
45     </div>
```

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

```
public function store_ajax(Request $request){
    //cek req ajax
    if ($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',
        ];

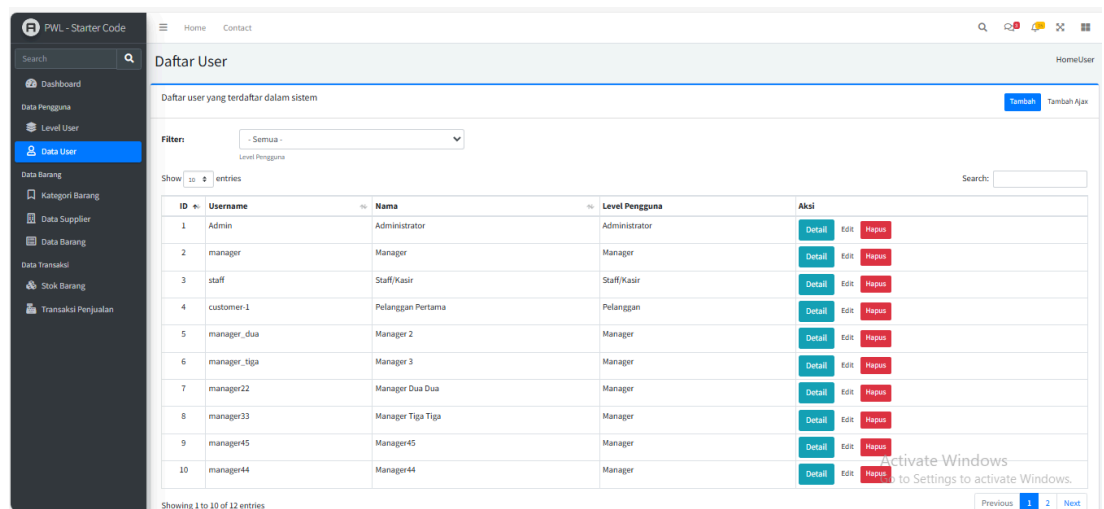
        $validator = Validator::make($request->all(), $rules);

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

        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 laporkan pada laporan jobsheet dan commit di github kalian!!!

- Tampilan Ajax



- Menambahkan user dengan form ajax

Tambah Data User

Level Pengguna
Staff/Kasir

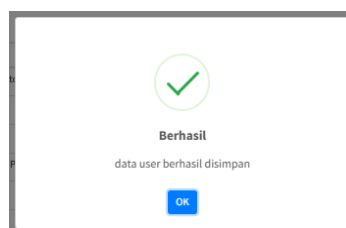
Username
Kasir

Nama
DhynarAzarine

Password
123456dh

Batal Simpan

- Data user yang berhasil ditambahkan



13	Kasir	DhynarAzarine	Staff/Kasir	Detail	Edit	Hapus	Activ Go to :
----	-------	---------------	-------------	--------	------	-------	------------------

Praktikum 2 - Modal Ajax Edit Data (Data User) :

1. Pada Praktikum 2 ini, kita akan melakukan coding 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($users)
        // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
        ->addIndexColumn()
        ->addColumn('aksi', function ($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::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
```

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

```
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

```

Minggu 6 > PWL_POS > resources > views > user > edit_ajax.blade.php > ...
1 @empty($user)
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('/user') }}" class="btn btn-warning">Kembali</a>
16 </div>
17 </div>
18 </div>
19 @else
20 <form action="{{ url('/user/' . $user->user_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 User</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>Level Pengguna</label>
34 <select name="level_id" id="level_id" class="form-control" required>
35 <option value="">- Pilih level --</option>

```

- 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',
            '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 !
 - tampilan edit ajax

- setelah edit user ajax

name	Nama	Level Pengguna	Aksi
Administrator	Administrator	Administrator	Detail Edit Hapus
Manager	Manager	Manager	Detail Edit Hapus
Staff/Kasir	Staff/Kasir	Staff/Kasir	Detail Edit Hapus
Pelanggan P	Pelanggan P	Pelanggan P	Detail Edit Hapus
Manager 2	Manager 2	Manager 2	Detail Edit Hapus
Manager 3	Manager 3	Manager 3	Detail Edit Hapus
Manager Du	Manager Du	Manager Du	Detail Edit Hapus
Manager Tig	Manager Tig	Manager Tig	Detail Edit Hapus
Manager45	Manager45	Manager45	Detail Edit Hapus
Manager44	Manager44	Manager44	Detail Edit Hapus
Manager55	Manager55	Manager55	Detail Edit Hapus
Manager11	Manager11	Manager11	Detail Edit Hapus
dhynar	dhynar	dhynar	Detail Edit Hapus

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::get('/{id}/delete_ajax',[UserController::class,'confirm_ajax']); // untuk tampilan form confirm delete yser Ajax
Route::delete('/{id}/delete_ajax',[UserController::class,'delete_ajax']); // untuk hapus data user Ajax
```

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

```
Minggu 6 > PWL_POS > resources > views > user > confirm_ajax.blade.php > form#form-delete > div#modal-master.modal-dialog.modal-lg > div.modal-content
1 @empty($user)
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('/user') }}" class="btn btn-warning">Kembali</a>
16            </div>
17        </div>
18    </div>
19 @else
20     <form action="{{ url('/user/' . $user->user_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 User</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-danger">
33                         <h5><i class="icon fas fa-ban"></i> Kesalahan!!!</h5>
34                         Data yang anda cari tidak ditemukan
35                     </div>
36                     <a href="{{ url('/user') }}" class="btn btn-warning">Kembali</a>
37                 </div>
38             </div>
39        </div>
40    </form>
41 @endempty
```

5. Kemudian kita buat fungsi delete_ajax() pada UserController.php untuk mengkomodir 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('/');
}
```

6. Setelah semua selesai, mari kita coba untuk melakukan percobaan dari koding yang telah kita lakukan.
- setelah klik button hapus

Hapus Data User

Konfirmasi !!!

Apakah Anda ingin menghapus data seperti di bawah ini?

Level Pengguna :	
Username :	staff
Nama :	Staff/Kasir

Batal

Ya, Hapus

- Data berhasil dihapus

ID	Username	Nama	Level Pengguna	Aksi
1	Admin	Administrator	Administrator	Detail Edit Hapus
2	manager	Manager		Detail Edit Hapus
3	staff	Staff/Kasir		Detail Edit Hapus
4	customer-1	Pelanggan P		Detail Edit Hapus
5	manager_dua	Manager 2		Detail Edit Hapus
6	manager_tiga	Manager 3		Detail Edit Hapus
7	manager22	Manager Du		Detail Edit Hapus
8	manager33	Manager Tig		Detail Edit Hapus
9	manager45	Manager45		Detail Edit Hapus
10	manager44	Manager44	Manager	Detail Edit Hapus
11	manager56	Manager55	Manager	Detail Edit Hapus
12	manager12	Manager11	Manager	Detail Edit Hapus

Berhasil

Data berhasil dihapus

OK

Showing 1 to 12 of 12 entries

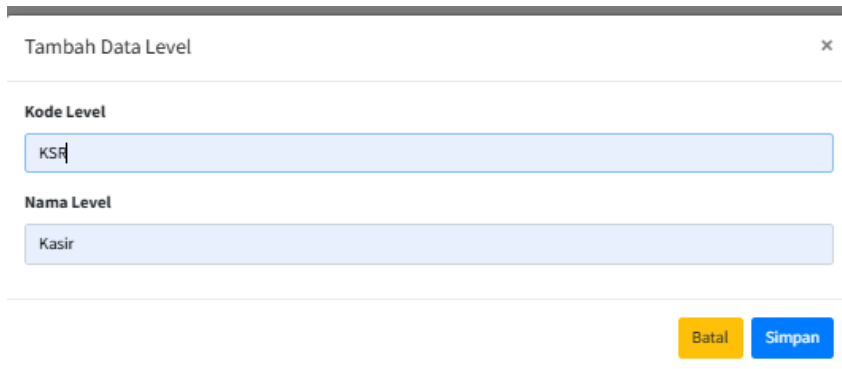
- Jangan lupa laporkan ke laporan jobsheet dan lakukan commit pada repository git kalian.!!!

TUGAS :

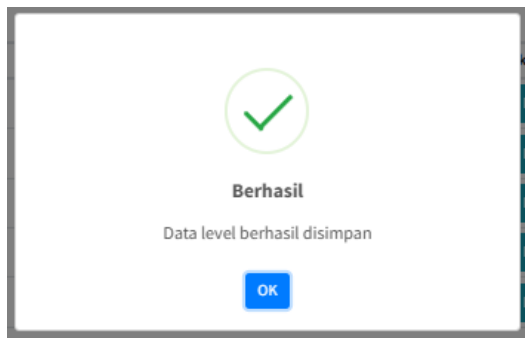
Implementasikan koding untuk Ajax Form dan Client Validation dengan jQuery Validation pada menu berikut ini

✓ Tabel m_level

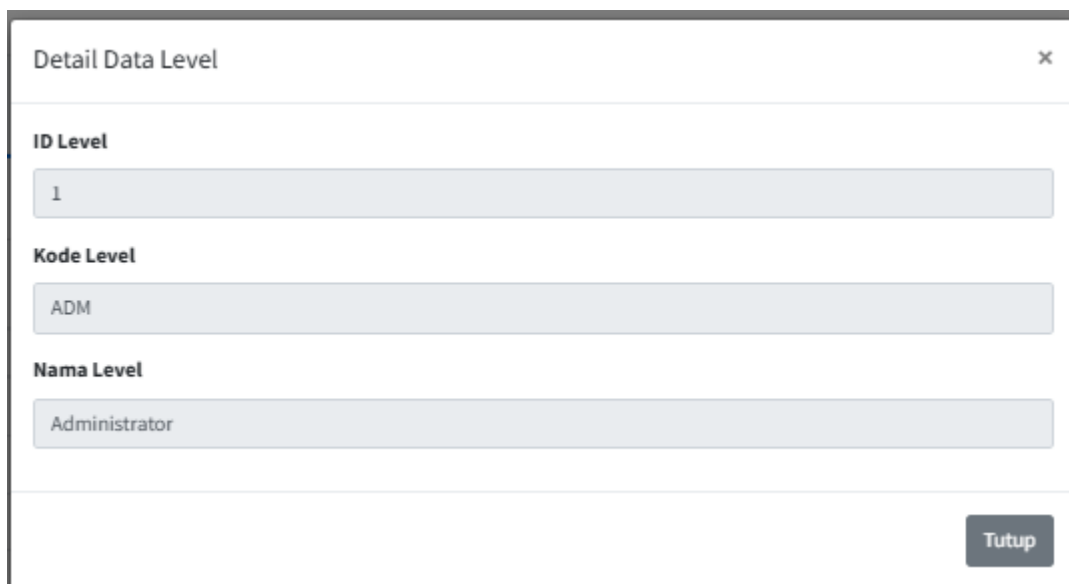
- Menambahkan data level



The screenshot shows a modal window titled "Tambah Data Level" with a close button (X) in the top right corner. It contains two text input fields: "Kode Level" with the value "KSR" and "Nama Level" with the value "Kasir". At the bottom right, there are two buttons: "Batal" (yellow) and "Simpan" (blue).



- Detail data level



The screenshot shows a modal window titled "Detail Data Level" with a close button (X) in the top right corner. It contains three text input fields: "ID Level" with the value "1", "Kode Level" with the value "ADM", and "Nama Level" with the value "Administrator". At the bottom right, there is a grey "Tutup" (Close) button.

- Edit Data level

Edit Data Level

Level Kode

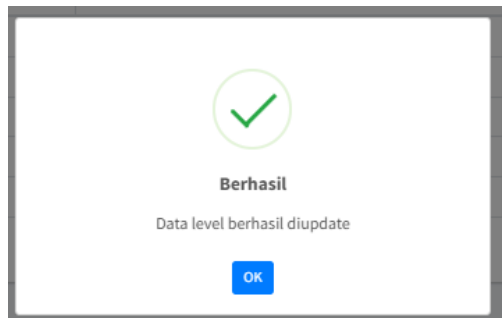
KSR

Level Nama

Kasir 1

Batal

Simpan



5	KSR	Kasir 1
---	-----	---------

- Menghapus data level

Hapus Data level

ⓘ

Konfirmasi !!!

Apakah Anda ingin menghapus data seperti di bawah ini?

level Kode :	ADM
level Nama :	Administrator

Batal

Ya, Hapus

✓ Tabel m_kategori

- Menambahkan data kategori

Tambah Data Kategori

Kode Kategori

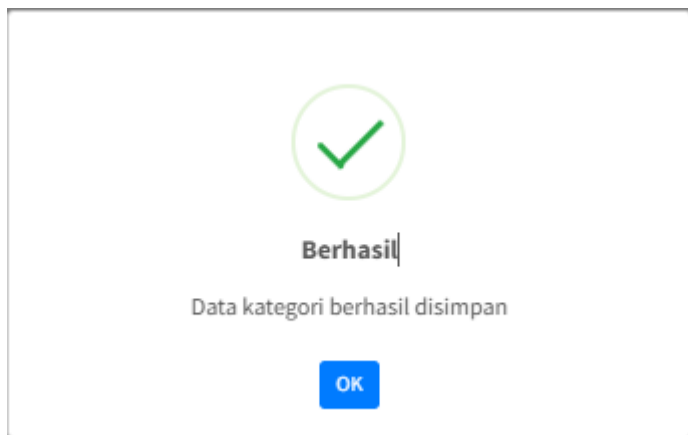
FTR

Nama Kategori

Furniture

Batal

Simpan



- Detail data kategori

Detail Data Kategori

ID

26

Kode Kategori

FTR

Nama Kategori

Furniture

Tutup

- Edit Data kategori

Edit Data Kategori

Kode Kategori

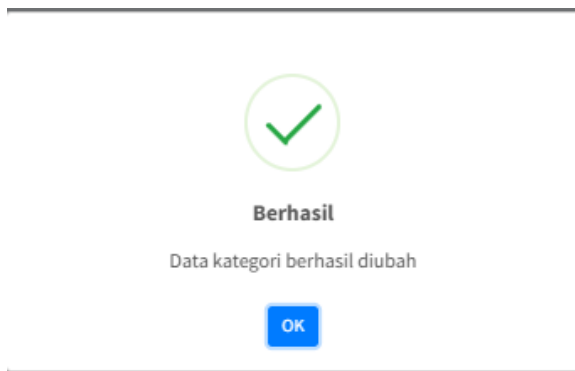
FTR

Nama Kategori

Furnitu

Batal

Simpan



- Menghapus data kategori

Hapus Data Kategori

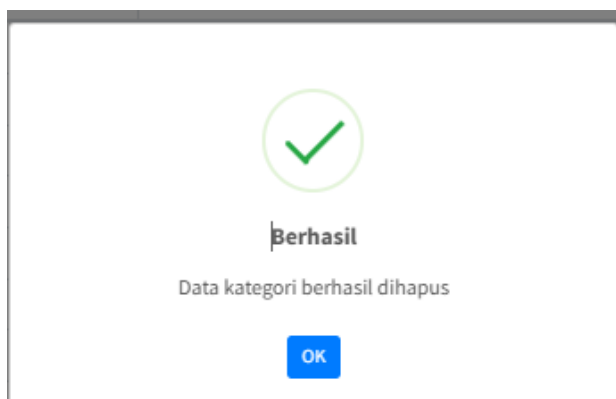
Konfirmasi !!!

Apakah Anda ingin menghapus data seperti di bawah ini?

Kategori Kode :	FTR
Kategori Nama :	Furniture

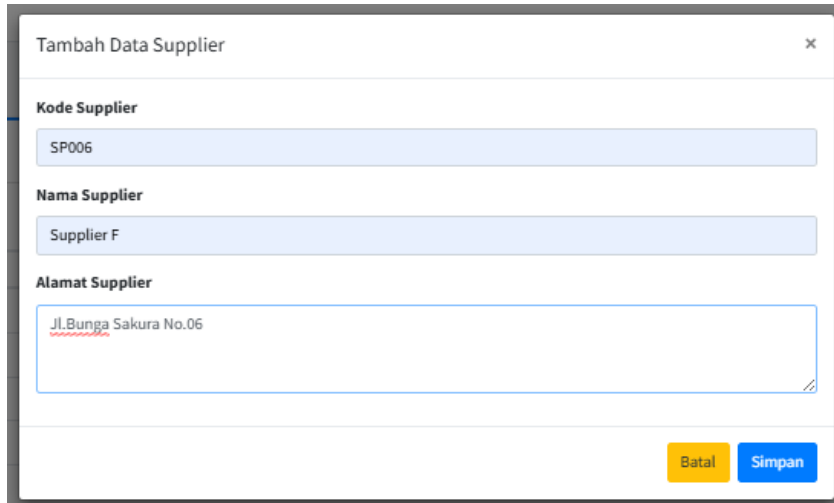
Batal

Ya, Hapus

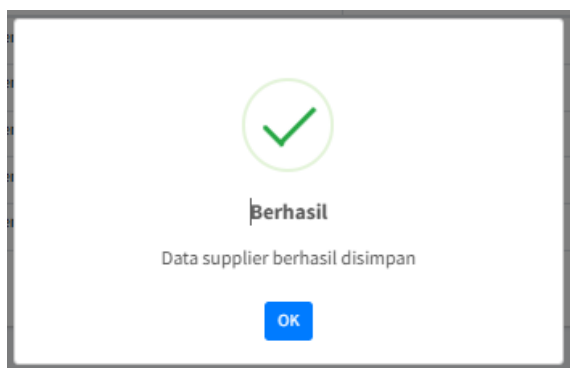


✓ Tabel m_supplier

- Menambahkan data supplier



The screenshot shows a web form titled "Tambah Data Supplier" with a close button (X) in the top right corner. The form contains three input fields: "Kode Supplier" with the value "SP006", "Nama Supplier" with the value "Supplier F", and "Alamat Supplier" with the value "Jl.Bunga Sakura No.06". At the bottom right, there are two buttons: "Batal" (yellow) and "Simpan" (blue).



- Detail data supplier



The screenshot shows a web form titled "Detail Data Supplier" with a close button (X) in the top right corner. The form displays four fields with their respective values: "ID" (8), "Kode" (SP006), "Nama" (Supplier F), and "Alamat" (Jl.Bunga Sakura No.06). At the bottom right, there is a grey "Tutup" button.

- Edit Data supplier

Edit Data Supplier

Kode Supplier

SP005

Nama Supplier

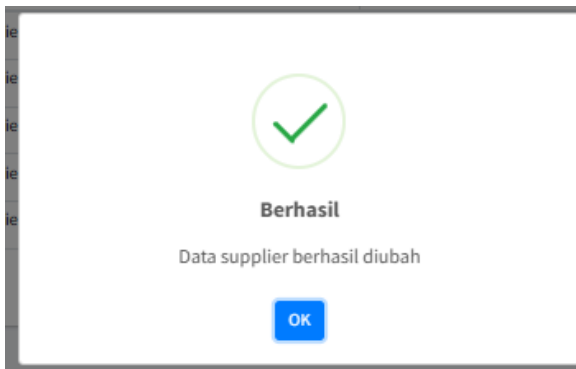
Supplier E

Alamat Supplier

Jl.Bunga Sakura No.06

Batal

Simpan



- Menghapus data supplier

Hapus Data Supplier

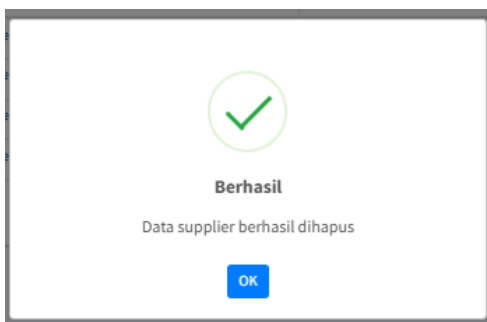
Konfirmasi !!!

Apakah Anda ingin menghapus data seperti di bawah ini?

Kode Supplier :	SP005
Nama Supplier :	Supplier E
Alamat Supplier :	Jl.Bunga Sakura No.06

Batal

Ya, Hapus



✓ Tabel m_barang

- Menambahkan data barang

Tambah Data Barang

Kategori

Skincare

Kode Barang

BNG011

Nama Barang

Pelembab

Harga Beli

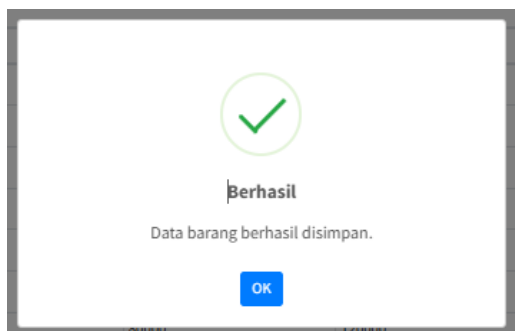
90000

Harga Jual

115000

Batal

Simpan



- Detail data barang

Detail Data Barang

ID

13

Kategori

Skincare

Kode Barang

BNG011

Nama Barang

Pelembab

Harga Beli

90.000

Harga Jual

115.000

Tutup

- Edit Data barang

Edit Data Barang

Kategori

Skincare

Kode Barang

BNG011

Nama Barang

Pelembab

Harga Beli

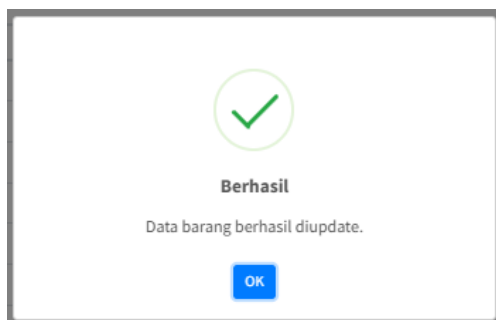
90000

Harga Jual

110000

Batal

Simpan



- Menghapus data barang

Hapus Data Barang

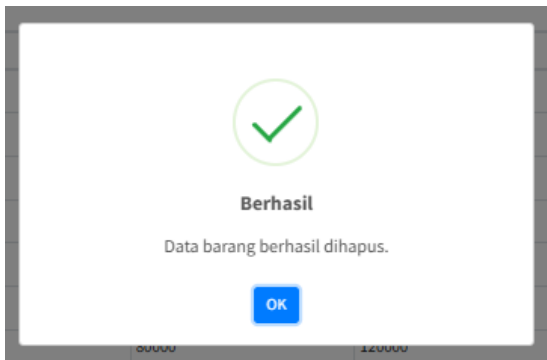
Konfirmasi !!!

Apakah Anda yakin ingin menghapus data barang berikut?

Kategori :	Skincare
Kode Barang :	BNG011
Nama Barang :	Pelembab
Harga Beli :	Rp 90.000
Harga Jual :	Rp 110.000

Batal

Ya, Hapus



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