

Laporan Praktikum Pemrograman Web Lanjut
Jobsheet 6: Ajax Form (AdminLTE) dan Client Validation .



Oleh :

Danica Nasywa Putriniair (2341760122)

Kelas SIB 2B / 05

PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS JURUSAN
TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

Jl. Soekarno Hatta No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang, Jawa Timur

65141

A. 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.

```
17 <link rel="stylesheet" href="{{ asset('adminlte/plugins/datatables-buttons/css/buttons.bootstrap4.min.css') }}">
18 <!-- SweetAlert2 -->
19 <link rel="stylesheet" href="{{ asset('adminlte/plugins/sweetalert2-theme-bootstrap-4/bootstrap4.min.css') }}">
20 <!-- Theme style -->
21 <link rel="stylesheet" href="{{ asset('adminlte/dist/css/adminlte.min.css') }}">

<script src="{{ asset('adminlte/plugins/datatables-buttons/js/buttons.colvis.min.js') }}"></script>
<!-- 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>
<!-- AdminLTE App -->
```

3. Selanjutnya kita modifikasi view user/index.blade.php

```
<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>
```

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

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

var dataUser;
$(document).ready(function () {
    dataUser = $('#table_user').DataTable({
        // serverSide: true, jika ingi menggunakan server
        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 oprasi ajax

```
Route::get('/create_ajax', [UserController::class, 'create_ajax']); // m
Route::post('/ajax', [UserController::class, 'store_ajax']); // m
```

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

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

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', // pesan error validasi
                'msgField' => $validator->errors(), // error message validation
            ]);
        }

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

    redirect('/');
}
```

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

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

```
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($users)
        ->addIndexColumn()
        ->addColumn('aksi', function ($user) {
            $btn = '<button onclick="modalAction(\''.url('/user/' . $user->user_id . '/s
            $btn .= '<button onclick="modalAction(\''.url('/user/' . $user->user_id . '/
            $btn .= '<button onclick="modalAction(\''.url('/user/' . $user->user_id . '/
            // $btn = '<a href="'.url('/user/' . $user->user_id).'" class="btn btn-info
            // $btn .= '<a href="'.url('/user/' . $user->user_id . '/edit').'" class="bt
            // $btn .= '<form class="d-inline-block" method="POST" action="'.url('/user/
            //     csrf_field() . method_field('DELETE') .
            //     '<button type="submit" class="btn btn-danger btn-sm" onclick="return
            return $btn;
        })
        ->rawColumns(['aksi'])
        ->make(true);
}
```

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

```
Route::put('/{id}', [UserController::class, 'update']); // menyimpan per
Route::get('/{id}/edit_ajax', [UserController::class, 'edit_ajax']); // mena
Route::put('/{id}update_ajax', [UserController::class, 'update_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

```

@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>
</div>
@else
<form action="{{ url('/user/' . $user->user_id . '/update_ajax') }}" method="POST" id="form-e
@csrf
@method('PUT')
<div id="modal-master" class="modal-dialog modal-lg" role="document">
  <div class="modal-content">
    <div class="modal-header">
      <h5 class="modal-title" id="exampleModallabel">Edit Data User</h5>
      <button type="button" class="close" data-dismiss="modal" aria-label="Close">
        <span aria-hidden="true">&times;</span></button>
    </div>
    <div class="modal-body">
      <div class="form-group">
        <label>Level Pengguna</label>
        <select name="level_id" id="level_id" class="form-control" required>
          <option value="">- Pilih Level -</option>
          @foreach($level as $l)
            <option {{ ($l->level_id == $user->level_id) ? 'selected' : '' }}
              value="{{ $l->level_id }}">{{ $l->level_nama }}
            </option>
          @endforeach
        </select>
        <small id="error-level_id" class="error-text form-text textdanger"></small
      </div>
      <div class="form-group">

```

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
                $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 prosesnya. Jangan lupa laporkan dan *commit* ke *repository git* kalian.!

The screenshot shows a web application interface for user management. The main content area is titled "Daftar User" and displays a table of users. The table has columns for ID, Username, Nama, Level Pengguna, and Aksi. There are three users listed: admin, manager, and staff. Each user has buttons for "Detail", "Edit", and "Hapus" (Delete). The interface also includes a sidebar with navigation links, a search bar, and a footer with copyright information.

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator		Detail Edit Hapus
2	manager	Manager		Detail Edit Hapus
3	staff	Staff/Kasir		Detail Edit Hapus

Hasilnya error

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 untuk menghapus data

```
Route::put('/{id}/update_ajax', [UserController::class, 'update_ajax']); // menyimpan
Route::get('/{id}/delete_ajax', [UserController::class, 'confirm_ajax']); // menampilkan
Route::put('/{id}/delete_ajax', [UserController::class, 'delete_ajax']); // menyimpan
```

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

```
public function confirm_ajax(String $id){
    $user = UserModel::find($id);

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

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

```
resources > views > user > confirm_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. <a href="{{ url('/user') }}" class="
14                 </div>
15             </div>
16         </div>
17     </div>
18 @else
19     <form action="{{ url('/user/' . $user->user_id . '/delete_ajax') }}" method="POST" id="fo
20         @csrf
21         @method('DELETE')
22         <div id="modal-master" class="modal-dialog modal-lg" role="document">
23             <div class="modal-content">
24                 <div class="modal-header">
25                     <h5 class="modal-title" id="exampleModalLabel">Hapus Data User</h5>
26                     <button type="button" class="close" data-dismiss="modal" aria-label="Clos
27                         <span aria-hidden="true">&times;</span>
28                     </button>
29                 </div>
30                 <div class="modal-body">
31                     <div class="alert alert-warning">
32                         <h5><i class="icon fas fa-exclamation-triangle"></i> Konfirmasi !!!</
33                         Apakah Anda ingin menghapus data seperti di bawah ini?
34                     </div>
35                     <table class="table table-sm table-bordered table-striped">
```

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('/');
}

```

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

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

The screenshot displays a web application interface for user management. The sidebar on the left contains various navigation options, with 'Data User' currently selected. The main content area, titled 'Daftar User', shows a list of users registered in the system. The table has columns for ID, Username, Nama, Level Pengguna, and Aksi. There are four entries in the table, each with 'Detail', 'Edit', and 'Hapus' (Delete) buttons. The interface also includes a search bar, a filter dropdown, and pagination controls at the bottom of the table. The footer indicates the application is AdminLTE.io, version 3.2.0, with a copyright notice from 2014 to 2021.

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	Detail Edit Hapus
2	manager	Manager	Manager	Detail Edit Hapus
3	staff	Staff/Kasir	Staff/Kasir	Detail Edit Hapus
4	hai_kamu	da	Staff/Kasir	Detail Edit Hapus

Tugas jobsheet

Implementasikan coding untuk ajax form dan client validation dengan jQuery Validation pada menu berikut ini

● Tabel m_level

Daftar Level yang terdaftar dalam sistem

Show 10 entries Search:

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

Showing 1 to 3 of 3 entries Previous 1 Next

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

Tambah Level

Kode Level: USR

Nama Level: User

Tutup Simpan

127.0.0.1:8000 says
Data level berhasil disimpan!

OK

Kode Level: USR

Nama Level: User

Tutup Simpan

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

HomeContact

Daftar Level

HomeLevel

Daftar Level yang terdaftar dalam sistem

+Tambah

Show10entriesSearch:

ID	Kode Level	Nama Level	Aksi
1	ADM	Administrator	EditHapus
2	MNG	Manager	EditHapus
3	STF	Staff/Kasir	EditHapus
4	USR	User	EditHapus

Showing 1 to 4 of 4 entries

Previous1Next

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

● Tabel m_kategori

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

HomeContact

Daftar Kategori

HomeKategori

Daftar Kategori yang terdaftar dalam sistem

Tambah

Show10entriesSearch:

ID	Kode Kategori	Nama Kategori	Aksi
1	BABY	BabyKid	DetailEditHapus
2	FASH	Fashion	DetailEditHapus
3	FOOD	Makanan	DetailEditHapus
4	BEAU	Kecantikan	DetailEditHapus
5	HOME	HomeCare	DetailEditHapus

Showing 1 to 5 of 5 entries

Previous1Next

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

HomeContact

Tambah Kategori

Home / KategoriTambah

Tambah Kategori baru

Kode Kategori

BOOK

Nama Kategori

BukuBacadanKamus

SimpanKembali

Copyright © 2014-2021 AdminLTE.io. All rights reserved. Version 3.2.0

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Kategori

Daftar Kategori yang terdaftar dalam sistem

Data kategori berhasil disimpan

Show 10 entries

Search:

ID	Kode Kategori	Nama Kategori	Aksi
1	BABY	BabyKid	Detail Edit Hapus
2	FASH	Fashion	Detail Edit Hapus
3	FOOD	Makanan	Detail Edit Hapus
4	BEAU	Kecantikan	Detail Edit Hapus
5	HOME	HomeCare	Detail Edit Hapus
6	BOOK	BukuBacadaKamus	Detail Edit Hapus

● Tabel m_supplier

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Supplier

Daftar Supplier yang terdaftar dalam sistem

+Tambah

Show 10 entries

Search:

ID	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	CV Sumber Rejeki	Jl. Merdeka No.123, Surabaya	Edit Hapus
2	SUP002	PT Maju Jaya	Jl. Diponegoro No.45, Malang	Edit Hapus
3	SUP003	Toko Alat Tulis Kita	Jl. Sudirman No.67, Jakarta	Edit Hapus
4	SUP004	UD Sinar Terang	Jl. Imam Bonjol No.10, Bandung	Edit Hapus
5	SUP005	CV Berkah Abadi	Jl. Gajah Mada No.88, Yogyakarta	Edit Hapus

Showing 1 to 5 of 5 entries

Previous 1 Next

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Supplier

Daftar Supplier yang terdaftar dalam sistem

+Tambah

Show 10 entries

Search:

ID	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	CV Sumber Rejeki	Jl. Merdeka No.123, Surabaya	Edit Hapus
2	SUP002	PT Maju Jaya	Jl. Diponegoro No.45, Malang	Edit Hapus
3	SUP003	Toko Alat Tulis Kita	Jl. Sudirman No.67, Jakarta	Edit Hapus
4	SUP004	UD Sinar Terang	Jl. Imam Bonjol No.10, Bandung	Edit Hapus
5	SUP005	CV Berkah Abadi	Jl. Gajah Mada No.88, Yogyakarta	Edit Hapus

Showing 1 to 5 of 5 entries

Previous 1 Next

Tambah Supplier

Kode Supplier:

SUP006

Nama Supplier:

PT Susah Letih

Alamat Supplier:

Jl. Semua Harus Semangat No.1, Ubud

Tutup Simpan

PWL - Starter Code

Search

Q

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home

Contact

127.0.0.1:8000 says

Data supplier berhasil disimpan!

OK

Daftar Supplier

Daftar Supplier yang terdaftar dalam sistem

+ Tambah

Show 10 entries

Search:

ID	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	PT Susah Letih	Jl. Merdeka No.123, Surabaya	Edit Hapus
2	SUP002	PT Maju Jaya	Jl. Diponegoro No.45, Malang	Edit Hapus
3	SUP003	Toko Alat Tulis Kita	Jl. Sudirman No.67, Jakarta	Edit Hapus
4	SUP004	UD Sinar Terang	Jl. Imam Bonjol No.10, Bandung	Edit Hapus
5	SUP005	CV Berkah Abadi	Jl. Gajah Mada No.88, Yogyakarta	Edit Hapus

Showing 1 to 5 of 5 entries

Previous 1 Next

PWL - Starter Code

Search

Q

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home

Contact

Daftar Supplier

Daftar Supplier yang terdaftar dalam sistem

+ Tambah

Show 10 entries

Search:

ID	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	CV Sumber Rejeki	Jl. Merdeka No.123, Surabaya	Edit Hapus
2	SUP002	PT Maju Jaya	Jl. Diponegoro No.45, Malang	Edit Hapus
3	SUP003	Toko Alat Tulis Kita	Jl. Sudirman No.67, Jakarta	Edit Hapus
4	SUP004	UD Sinar Terang	Jl. Imam Bonjol No.10, Bandung	Edit Hapus
5	SUP005	CV Berkah Abadi	Jl. Gajah Mada No.88, Yogyakarta	Edit Hapus
7	SUP006	PT Susah Letih	Jl. Semua Harus Semangat No.1, Ubud	Edit Hapus

● Tabel m_barang

PWL - Starter Code

Search

Q

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Home

Contact

Daftar Barang

Daftar barang yang terdaftar dalam sistem

Tambah

Filter: - Semua -

Kategori Barang

Show 10 entries

Search:

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BNG001	Toner	150000	185000	BabyKid	Detail Edit Hapus
2	BNG002	Serum	250000	315000	BabyKid	Detail Edit Hapus
3	BNG003	Rok Plisket	50000	65000	Fashion	Detail Edit Hapus
4	BNG004	Kemeja Bunga	65000	75000	Fashion	Detail Edit Hapus
5	BNG005	Penggaris	2500	4000	Makanan	Detail Edit Hapus
6	BNG006	Kalkulator	100000	115000	Makanan	Detail Edit Hapus
7	BNG007	Matras Yoga	80000	120000	Kecantikan	Detail Edit Hapus

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Supplier

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

HomeContact

Daftar Barang

HomeBarang

Daftar barang yang terdaftar dalam sistem

Tambah

Data barang berhasil disimpan

Filter: - Semua -

Kategori Barang

Show 10 entries

Search:

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BNG001	Toner	150000	185000	BabyKid	Detail Edit Hapus
2	BNG002	Serum	250000	315000	BabyKid	Detail Edit Hapus
3	BNG003	Rok Plisket	50000	65000	Fashion	Detail Edit Hapus
4	BNG004	Kemeja Bunga	65000	75000	Fashion	Detail Edit Hapus
5	BNG005	Penggaris	2500	4000	Makanan	Detail Edit Hapus

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