



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

Kelas : SIB
NIM : 2241760035
Nama : Fifi Novitasari
Jobsheet Ke- : 8

Laporan Jobsheet

Praktikum Ke-1 – Implementasi Upload File Untuk Import Data

Langkah	Jawaban/Deskripsi																																																																																																																																											
1	<p>Membuka database yang sudah kita buat sebelumnya, dan kita cek tabel <code>m_kategori</code> dan <code>m_barang</code></p> <p>- <code>m_kategori</code></p> <table><tr><th><code>kategori_id</code></th><th><code>kategori_kode</code></th><th><code>kategori_nama</code></th><th><code>created_at</code></th><th><code>updated_at</code></th></tr><tr><td>1</td><td>KD001</td><td>Minuman</td><td><i>NULL</i></td><td>2024-09-30 10:11:19</td></tr><tr><td>2</td><td>KD002</td><td>Makanan</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>3</td><td>KD003</td><td>Pakaian</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>4</td><td>KD004</td><td>Kendaraan</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>5</td><td>KD005</td><td>Peralatan</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>8</td><td>KD009</td><td>Elektronik</td><td>2024-10-04 03:52:19</td><td>2024-10-04 03:52:19</td></tr></table> <p>- <code>m_barang</code></p> <table><tr><th><code>barang_id</code></th><th><code>kategori_id</code></th><th><code>barang_kode</code></th><th><code>barang_nama</code></th><th><code>harga_beli</code></th><th><code>harga_jual</code></th><th><code>created_at</code></th><th><code>updated_at</code></th></tr><tr><td>1</td><td>1</td><td>BRK001</td><td>Indomie Goreng</td><td>2200</td><td>2500</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>2</td><td>2</td><td>BRK002</td><td>Roti Bakar</td><td>2000</td><td>3000</td><td><i>NULL</i></td><td>2024-09-30 10:38:46</td></tr><tr><td>3</td><td>1</td><td>BRK003</td><td>Mie Instan</td><td>2500</td><td>3000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>4</td><td>1</td><td>BRK004</td><td>Snack Keripik</td><td>3000</td><td>3500</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>6</td><td>2</td><td>BRK006</td><td>Teh Botol</td><td>2500</td><td>3000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>7</td><td>2</td><td>BRK007</td><td>Air Mineral</td><td>1000</td><td>1500</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>8</td><td>2</td><td>BRK008</td><td>Jus Jeruk</td><td>5000</td><td>6000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>9</td><td>2</td><td>BRK009</td><td>Kopi Instan</td><td>3000</td><td>3500</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>10</td><td>2</td><td>BRK010</td><td>Soda</td><td>4000</td><td>4500</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>11</td><td>3</td><td>BRK011</td><td>Kaos Polos</td><td>30000</td><td>35000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>12</td><td>3</td><td>BRK012</td><td>Jaket</td><td>70000</td><td>80000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr><tr><td>14</td><td>3</td><td>BRK014</td><td>Kemeja</td><td>45000</td><td>50000</td><td><i>NULL</i></td><td><i>NULL</i></td></tr></table>	<code>kategori_id</code>	<code>kategori_kode</code>	<code>kategori_nama</code>	<code>created_at</code>	<code>updated_at</code>	1	KD001	Minuman	<i>NULL</i>	2024-09-30 10:11:19	2	KD002	Makanan	<i>NULL</i>	<i>NULL</i>	3	KD003	Pakaian	<i>NULL</i>	<i>NULL</i>	4	KD004	Kendaraan	<i>NULL</i>	<i>NULL</i>	5	KD005	Peralatan	<i>NULL</i>	<i>NULL</i>	8	KD009	Elektronik	2024-10-04 03:52:19	2024-10-04 03:52:19	<code>barang_id</code>	<code>kategori_id</code>	<code>barang_kode</code>	<code>barang_nama</code>	<code>harga_beli</code>	<code>harga_jual</code>	<code>created_at</code>	<code>updated_at</code>	1	1	BRK001	Indomie Goreng	2200	2500	<i>NULL</i>	<i>NULL</i>	2	2	BRK002	Roti Bakar	2000	3000	<i>NULL</i>	2024-09-30 10:38:46	3	1	BRK003	Mie Instan	2500	3000	<i>NULL</i>	<i>NULL</i>	4	1	BRK004	Snack Keripik	3000	3500	<i>NULL</i>	<i>NULL</i>	6	2	BRK006	Teh Botol	2500	3000	<i>NULL</i>	<i>NULL</i>	7	2	BRK007	Air Mineral	1000	1500	<i>NULL</i>	<i>NULL</i>	8	2	BRK008	Jus Jeruk	5000	6000	<i>NULL</i>	<i>NULL</i>	9	2	BRK009	Kopi Instan	3000	3500	<i>NULL</i>	<i>NULL</i>	10	2	BRK010	Soda	4000	4500	<i>NULL</i>	<i>NULL</i>	11	3	BRK011	Kaos Polos	30000	35000	<i>NULL</i>	<i>NULL</i>	12	3	BRK012	Jaket	70000	80000	<i>NULL</i>	<i>NULL</i>	14	3	BRK014	Kemeja	45000	50000	<i>NULL</i>	<i>NULL</i>
<code>kategori_id</code>	<code>kategori_kode</code>	<code>kategori_nama</code>	<code>created_at</code>	<code>updated_at</code>																																																																																																																																								
1	KD001	Minuman	<i>NULL</i>	2024-09-30 10:11:19																																																																																																																																								
2	KD002	Makanan	<i>NULL</i>	<i>NULL</i>																																																																																																																																								
3	KD003	Pakaian	<i>NULL</i>	<i>NULL</i>																																																																																																																																								
4	KD004	Kendaraan	<i>NULL</i>	<i>NULL</i>																																																																																																																																								
5	KD005	Peralatan	<i>NULL</i>	<i>NULL</i>																																																																																																																																								
8	KD009	Elektronik	2024-10-04 03:52:19	2024-10-04 03:52:19																																																																																																																																								
<code>barang_id</code>	<code>kategori_id</code>	<code>barang_kode</code>	<code>barang_nama</code>	<code>harga_beli</code>	<code>harga_jual</code>	<code>created_at</code>	<code>updated_at</code>																																																																																																																																					
1	1	BRK001	Indomie Goreng	2200	2500	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
2	2	BRK002	Roti Bakar	2000	3000	<i>NULL</i>	2024-09-30 10:38:46																																																																																																																																					
3	1	BRK003	Mie Instan	2500	3000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
4	1	BRK004	Snack Keripik	3000	3500	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
6	2	BRK006	Teh Botol	2500	3000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
7	2	BRK007	Air Mineral	1000	1500	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
8	2	BRK008	Jus Jeruk	5000	6000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
9	2	BRK009	Kopi Instan	3000	3500	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
10	2	BRK010	Soda	4000	4500	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
11	3	BRK011	Kaos Polos	30000	35000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
12	3	BRK012	Jaket	70000	80000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					
14	3	BRK014	Kemeja	45000	50000	<i>NULL</i>	<i>NULL</i>																																																																																																																																					



2

Mencoba untuk memasukkan data ke dalam sistem kita secara banyak. Kita bisa menggunakan file excel untuk mencoba memasukkan data barang ke dalam sistem kita. Kita buat template file excel yang akan kita gunakan untuk import data barang.

	A	B	C	D	E	F	G	H
1	barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
2	1	1	BRK001	Indomie Goreng	2200	2500	NULL	NULL
3	2	2	BRK002	Roti Bakar	2000	3000	NULL	30/09/2024 10:38
4	3	1	BRK003	Mie Instan	2500	3000	NULL	NULL
5	4	1	BRK004	Snack Kerip	3000	3500	NULL	NULL
6	6	2	BRK006	Teh Botol	2500	3000	NULL	NULL
7	7	2	BRK007	Air Mineral	1000	1500	NULL	NULL
8	8	2	BRK008	Jus Jeruk	5000	6000	NULL	NULL
9	9	2	BRK009	Kopi Instan	3000	3500	NULL	NULL
10	10	2	BRK010	Soda	4000	4500	NULL	NULL
11	11	3	BRK011	Kaos Polos	30000	35000	NULL	NULL
12	12	3	BRK012	Jaket	70000	80000	NULL	NULL
13	14	3	BRK014	Kemeja	45000	50000	NULL	NULL
14	15	3	BRK015	Dress	60000	70000	NULL	NULL
15	18	2	BRK068	Milo Cubes	9500	10000	04/10/2024 03:47	04/10/2024 03:47
16	19	5	BRK20	Cat Akrilik	20000	50000	04/10/2024 03:58	04/10/2024 04:01

3

Modifikasi view pada [barang/index.blade.php](#) untuk bisa menambahkan tombol menambah form untuk upload untuk import data barang

```
@extends('layouts.template')

@section('content')
    <div class="card">
        <div class="card-header">
            <h3 class="card-title">Daftar barang</h3>
            <div class="card-tools">
                <button onclick="modalAction('{{ url('/barang/import') }}')" class="btn btn-info">Import Barang</button>
                <a href="{{ url('/barang/create') }}" class="btn btn-primary">Tambah Data</a>
                <button onclick="modalAction('{{ url('/barang/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
            </div>
        </div>
        <div class="card-body">
            <!-- untuk Filter data -->
            <div id="filter" class="form-horizontal filter-date p-2 border-bottom mb-2">
                <div class="row">
                    <div class="col-md-12">
                        <div class="form-group form-group-sm row text-sm mb-0">
                            <label for="filter_date" class="col-md-1 col-form-label">Filter</label>
                            <div class="col-md-3">
```



```
<select name="filter_kategori" class="form-control form-
control-sm filter_kategori">

    <option value="">- Semua -</option>
    @foreach($kategori as $1)
        <option value="{{ $1->kategori_id }}">{{ $1-
>kategori_nama }}</option>

    @endforeach
</select>

<small class="form-text text-muted">Kategori
Barang</small>

</div>
</div>
</div>
</div>
</div>
</div>
@if(session('success'))
    <div class="alert alert-success">{{ session('success') }}</div>
@endif
@if(session('error'))
    <div class="alert alert-danger">{{ session('error') }}</div>
@endif
<table class="table table-bordered table-sm table-striped table-
hover" id="table-barang">
    <thead>
        <tr><th>No</th><th>Kode Barang</th><th>Kode Barang</th><th>Harga
Beli</th><th>Harga Jual</th><th>Kategori</th><th>Aksi</th></tr>
    </thead>
    <tbody></tbody>
</table>
</div>
</div>
<div id="myModal" class="modal fade animate shake" tabindex="-1" data-
backdrop="static" data-keyboard="false" data-width="75%"></div>

@endsection

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



```
tableBarang = $('#table-barang').DataTable({
    processing: true,
    serverSide: true,
    ajax: {
        "url": "{{ url('barang/list') }}",
        "dataType": "json",
        "type": "POST",
        "data": function (d) {
            d.filter_kategori = $('#filter_kategori').val();
        }
    },
    columns: [{
        data: "No_Urut",
        className: "text-center",
        width: "5%",
        orderable: false,
        searchable: false
    }, {
        data: "barang_kode",
        className: "",
        width: "10%",
        orderable: true,
        searchable: true
    }, {
        data: "barang_nama",
        className: "",
        width: "37%",
        orderable: true,
        searchable: true,
        render: function(data, type, row){
            return new Intl.NumberFormat('id-ID').format(data);
        }
    }, {
        data: "harga_beli",
        className: "",
        width: "10%",
        orderable: true,
        searchable: false,
        render: function(data, type, row){
            return new Intl.NumberFormat('id-ID').format(data);
        }
    }, {
        data: "harga_jual",
        className: "",
        width: "10%",
        orderable: true,
        searchable: false,
        render: function(data, type, row){
            return new Intl.NumberFormat('id-ID').format(data);
        }
    }
    ]
});
```



	<pre> return new Intl.NumberFormat('id-ID').format(data); } }, { data: "kategori.kategori_nama", className: "", width: "14%", orderable: true, searchable: false }, { data: "aksi", className: "text-center", width: "14%", orderable: false, searchable: false }] }); \$('#table-barang_filter input').unbind().bind().on('keyup', function(e){ if(e.keyCode == 13){ // enter key tableBarang.search(this.value).draw(); } }); \$('.filter_kategori').change(function(){ tableBarang.draw(); }); }); </script> @endpush</pre>
4	<p>Selanjutnya buat view untuk form upload/import file excel dan download file template_barang.xlsx. Kita beri nama file dengan nama barang/import.blade.php</p> <pre><form action="{ url('/barang/import_ajax') }" method="POST" id="form-import" enctype="multipart/form-data"> @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">Import Data Barang</h5> <button type="button" class="close" data-dismiss="modal" aria- label="Close"> &times; </button> </div> <div class="modal-body"></pre>



```
<div class="form-group">
<label>Download Template</label>
<a href="{{ asset('template_barang.xlsx') }}" class="btn btn-info btn-sm"
download>
    <i class="fa fa-file-excel"></i> Download
</a>
<small id="error-kategori_id" class="error-text form-text text-
danger"></small>
</div>
<div class="form-group">
<label>Pilih File</label>
<input type="file" name="file_barang" id="file_barang" class="form-
control" required>
<small id="error-file_barang" 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">Upload</button>
</div>
</div>
</form>

<script>
$(document).ready(function() {
    $("#form-import").validate({
        rules: {
            file_barang: { required: true, extension: ".xlsx" },
        },
        submitHandler: function(form) {
            var formData = new FormData(form); // Mengonversi form ke FormData untuk
menangani file
            $.ajax({
                url: form.action,
                type: form.method,
                data: formData, // Data yang dikirim berupa FormData
                processData: false, // Mengatur processData dan contentType ke false
untuk menangani file
                contentType: false,
                success: function(response) {
                    if (response.status) { // Jika sukses
                        $('#myModal').modal('hide');
```



	<pre>Swal.fire({ icon: 'success', title: 'Berhasil', text: response.message }); tableBarang.ajax.reload(); // Reload datatable } else { // Jika error \$('.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></pre>
5	<p>Kemudian modifikasi <code>route/web.php</code> untuk mengakomodir proses upload file pada menu barang</p> <pre>//Semua route di grup ini harus punya role ADM (Administrator) dan MNG (Manager) Route::group(['prefix' => 'barang', 'middleware'=> 'authorize:ADM,MNG'], function() { Route::get('/', [BarangController::class, 'index']); Route::post('/list', [BarangController::class, 'list']); Route::get('/create', [BarangController::class, 'create']); Route::post('/', [BarangController::class, 'store']); });</pre>



	<pre>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::get('/import', [BarangController::class, 'import']); //menampilkan form impor data barang Route::post('/import_ajax', [BarangController::class, 'import_ajax']); //ajax import excel });</pre>
6	<p>Membaca/menulis file excel, maka kita butuhkan library untuk membaca/menulis file excel. Jadi kita bisa memakai library phpoffice/phpspreadsheet. Kita ketikkan perintah di terminal/CMD</p> 
7	Selanjutnya modifikasi file BarangController.php untuk memproses data



```
web.php M index.blade.php M import.blade.php U BarangController.php M X
app > Http > Controllers > BarangController.php > BarangController > index
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\LevelModel;
6 use App\Models\BarangModel;
7 use App\Models\KategoriModel;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Hash;
10 use Illuminate\Support\Facades\Validator;
11 use PhpOffice\PhpSpreadsheet\IOFactory;
12 use Yajra\DataTables\Facades\DataTables;
13
14 class BarangController extends Controller
15 {
16     public function index()
17     {
18         $breadcrumb = (object) [
19             'title' => 'Daftar Barang',
20             'list' => ['Home', 'Barang']
21         ];
22
23         $page = (object) [
24             'title' => 'Daftar barang yang terdaftar dalam sistem'
25         ];
26
27         $activeMenu = 'barang';
28
29         $kategori = KategoriModel::all();
30
31         return view('barang.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
32     }
33
34     public function list(Request $request)
35     {
36         $barang = BarangModel::select('barang_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual', 'kategori_id')->with('kategori');
37
38         if ($request->kategori_id) {
39             $barang->where('kategori_id', $request->kategori_id);
40         }
41
42         return DataTables::of($barang)
43             // menambahkan kolom index / no urut (default nama kolom: DT_RowIndex)
44             ->addIndexColumn()
45             ->addColumn('aksi', function ($barang) { // menambahkan kolom aksi
46                 // $btn = '<a href="' . url('/barang/' . $barang->barang_id) . '" class="btn btn-info btn-sm">Detail</a>';
47                 // $btn .= '<a href="' . url('/barang/' . $barang->barang_id) . '/edit'" . '" class="btn btn-warning btn-sm">Edit</a>';
48                 // $btn .= '<form class="d-inline-block" method="POST" action="' .
49                 //     url('/barang/' . $barang->barang_id) . '">
50                 //     <div><input type="text" value="' . $barang->barang_id . '"></div>
51                 //     <input type="button" value="Hapus" class="btn btn-danger btn-sm" onclick="return confirm(\'' . $barang->barang_id . '\')></div></form>';
52                 $btn = '<button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id) .
53                     '/show_ajax" . '\')"' class="btn btn-info btn-sm">Detail</button>';
54                 $btn .= '<button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id) .
55                     '/edit_ajax" . '\')"' class="btn btn-warning btn-sm">Edit</button>';
56                 $btn .= '<button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id) .
57                     '/delete_ajax" . '\')"' class="btn btn-danger btn-sm">Hapus</button>';
58
59                 return $btn;
60             })
61             ->rawColumns(['aksi']) // memberitahu bahwa kolom aksi adalah html
62             ->make(true);
63     }
64
65     public function create()
66     {
67         $breadcrumb = (object) [
68             'title' => 'Tambah Barang',
69             'list' => ['Home', 'Barang', 'Tambah']
70         ];
71
72         $page = (object) [
73             'title' => 'Tambah barang baru'
74         ];
75
76         $kategori = KategoriModel::all();
77         $activeMenu = 'barang';
78
79         return view('barang.create', ['breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu, 'kategori' => $kategori]);
80     }
81
82     public function store(Request $request)
83     {
84         $request->validate([
85             'barang_kode' => 'required|string|min:3|unique:m_barang,barang_kode',
86             'barang_nama' => 'required|string|max:100',
87             'harga_beli' => 'required|integer',
88             'harga_jual' => 'required|integer',
89             'kategori_id' => 'required|integer'
90         ]);
91     }
92 }
```



```
92
93 // dd($request->kategori_id);
94
95 BarangModel::create([
96     'kategori_id' => $request->kategori_id,
97     'barang_kode' => $request->barang_kode,
98     'barang_nama' => $request->barang_nama,
99     'harga_beli' => $request->harga_beli,
100     'harga_jual' => $request->harga_jual,
101 ]);
102
103 return redirect('/barang')->with('success', 'Data barang berhasil disimpan');
104 }
105
106 public function show(string $id)
107 {
108     $barang = BarangModel::with('kategori')->find($id);
109
110     $breadcrumb = (object) [
111         'title' => 'Detail Barang',
112         'list' => ['Home', 'Barang', 'Detail']
113     ];
114
115     $page = (object) [
116         'title' => 'Detail barang'
117     ];
118
119     $activeMenu = 'barang';
120
121     return view('barang.show', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'activeMenu' => $activeMenu]);
122 }
123
124 public function edit(string $id)
125 {
126     $barang = BarangModel::find($id);
127     $kategori = KategoriModel::all();
128
129     $breadcrumb = (object) [
130         'title' => 'Edit Barang',
131         'list' => ['Home', 'Barang', 'Edit']
132     ];
133
134     $page = (object) [
135         'title' => 'Edit barang'
136     ];
137
138     $activeMenu = 'barang';
139
140     return view('barang.edit', ['breadcrumb' => $breadcrumb, 'page' => $page, 'barang' => $barang, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
141 }
142
143 public function update(Request $request, string $id)
144 {
145     $request->validate([
146         'barang_kode' => 'required|string|min:3|max:10|unique:m_barang,barang_kode,' . $id . ',barang_id',
147         'barang_nama' => 'required|string|max:100',
148         'harga_beli' => 'required|integer',
149         'harga_jual' => 'required|integer',
150         'kategori_id' => 'required|integer'
151     ]);
152
153     // dd($request);
154
155     BarangModel::find($id)->update([
156         'barang_kode' => $request->barang_kode,
157         'barang_nama' => $request->barang_nama,
158         'harga_beli' => $request->harga_beli,
159         'harga_jual' => $request->harga_jual,
160         'kategori_id' => $request->kategori_id
161     ]);
162
163     return redirect('/barang')->with('success', 'Data barang berhasil diubah');
164 }
165
166 public function destroy(string $id)
167 {
168     $check = BarangModel::find($id);
169     if (!$check) {
170         return redirect('/barang')->with('error', 'Data barang tidak ditemukan');
171     }
172
173     try {
174         BarangModel::destroy($id);
175
176         return redirect('/barang')->with('success', 'Data barang berhasil dihapus');
177     } catch (\Illuminate\Database\QueryException $e) {
178         return redirect('/barang')->with('error', 'Data barang gagal dihapus karena masih terdapat tabel lain yang terkait dengan data ini');
179     }
180 }
181
182 public function create_ajax()
183 {
184     $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();
185
186     return view('barang.create_ajax')->with('kategori', $kategori);
187 }
188 }
```



```
189 public function store_ajax(Request $request)
190 {
191     if ($request->ajax() || $request->wantsJson()) {
192         $rules = [
193             'barang_kode' => 'required|string|min:3|unique:m_barang,barang_kode',
194             'barang_nama' => 'required|string|max:100',
195             'harga_beli' => 'required|integer',
196             'harga_jual' => 'required|integer',
197             'kategori_id' => 'required|integer'
198         ];
199         $validator = Validator::make($request->all(), $rules);
200
201         if ($validator->fails()) {
202             return response()->json([
203                 'status' => false,
204                 'message' => 'Validasi gagal',
205                 'msgField' => $validator->errors(),
206             ]);
207         }
208
209         BarangModel::create($request->all());
210         return response()->json([
211             'status' => true,
212             'message' => 'Data barang berhasil disimpan'
213         ]);
214     }
215     redirect('/');
216 }
217
218 public function edit_ajax(string $id)
219 {
220     $barang = BarangModel::find($id);
221     $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();
222     return view('barang.edit_ajax', ['barang' => $barang, 'kategori' => $kategori]);
223 }
224
225 public function update_ajax(Request $request, $id)
226 {
227     // cek apakah request dari ajax
228     if ($request->ajax() || $request->wantsJson()) {
229         $rules = [
230             'barang_kode' => 'required|string|min:3|unique:m_barang,barang_kode,' . $id . ',barang_id',
231             'barang_nama' => 'required|string|max:100',
232             'harga_beli' => 'required|integer',
233             'harga_jual' => 'required|integer',
234             'kategori_id' => 'required|integer'
235         ];
236         // use Illuminate\Support\Facades\Validator;
237         $validator = Validator::make($request->all(), $rules);
238
239         if ($validator->fails()) {
240             return response()->json([
241                 'status' => false, // respon json, true: berhasil, false: gagal
242                 'message' => 'Validasi gagal',
243                 'msgField' => $validator->errors() // menunjukkan field mana yang error
244             ]);
245         }
246
247         $check = BarangModel::find($id);
248         if ($check) {
249             if (!$request->filled('password')) { // jika password tidak diisi, maka hapus dari request
250                 $request->request->remove('password');
251             }
252             $check->update($request->all());
253             return response()->json([
254                 'status' => true,
255                 'message' => 'Data berhasil diupdate'
256             ]);
257         } else {
258             return response()->json([
259                 'status' => false,
260                 'message' => 'Data tidak ditemukan'
261             ]);
262         }
263     }
264     return redirect('/');
265 }
266
267 public function confirm_ajax(string $id)
268 {
269     $barang = BarangModel::find($id);
270     return view('barang.confirm_ajax', ['barang' => $barang]);
271 }
272 }
```



```
279 | public function delete_ajax(Request $request, string $id)
280 | {
281 |     if ($request->ajax() || $request->wantsJson()) {
282 |         $barang = BarangModel::find($id);
283 |
284 |         if ($barang) {
285 |             $barang->delete();
286 |             return response()->json([
287 |                 'status' => true,
288 |                 'message' => 'Data berhasil dihapus'
289 |             ]);
290 |         } else {
291 |             return response()->json([
292 |                 'status' => false,
293 |                 'message' => 'Data tidak ditemukan'
294 |             ]);
295 |         }
296 |         return redirect('/');
297 |     }
298 | }
299 |
300 | public function import()
301 | {
302 |     return view('barang.import');
303 | }
304 |
305 | public function import_ajax(Request $request)
306 | {
307 |     if ($request->ajax() || $request->wantsJson()) {
308 |         $rules = [
309 |             // validasi file harus xls atauxlsx, max 1MB
310 |             'file_barang' => ['required', 'mimes:xlsx', 'max:1024']
311 |         ];
312 |
313 |         $validator = Validator::make($request->all(), $rules);
314 |         if ($validator->fails()) {
315 |             return response()->json([
316 |                 'status' => false,
317 |                 'message' => 'Validasi Gagal',
318 |                 'msgField' => $validator->errors()
319 |             ]);
320 |         }
321 |
322 |         $file = $request->file('file_barang'); // ambil file dari request
323 |
324 |         $reader = IDFactory::createReader('Xlsx'); // load reader file excel
325 |         $reader->setReadDataOnly(true); // hanya membaca data
326 |         $spreadsheet = $reader->load($file->getRealPath()); // load file excel
327 |         $sheet = $spreadsheet->getActiveSheet(); // ambil sheet yang aktif
328 |
329 |         $data = $sheet->toArray(null, false, true, true); // ambil data excel
330 |
331 |         $insert = [];
332 |         if (count($data) > 1) { // jika data lebih dari 1 baris
333 |             foreach ($data as $baris => $value) {
334 |                 if ($baris > 1) { // baris ke 1 adalah header, maka lewati
335 |                     $insert[] = [
336 |                         'kategori_id' => $value['A'],
337 |                         'barang_kode' => $value['B'],
338 |                         'barang_nama' => $value['C'],
339 |                         'harga_beli' => $value['D'],
340 |                         'harga_jual' => $value['E'],
341 |                         'created_at' => now(),
342 |                     ];
343 |                 }
344 |             }
345 |
346 |             if (count($insert) > 0) {
347 |                 // insert data ke database, jika data sudah ada, maka diabaikan
348 |                 BarangModel::insertOrIgnore($insert);
349 |             }
350 |
351 |             return response()->json([
352 |                 'status' => true,
353 |                 'message' => 'Data berhasil diimport'
354 |             ]);
355 |         } else {
356 |             return response()->json([
357 |                 'status' => false,
358 |                 'message' => 'Tidak ada data yang diimport'
359 |             ]);
360 |         }
361 |     }
362 |     return redirect('/');
363 | }
364 | }
```

8

Sekarang coba jalankan browser dan klik tombol import pada menu Barang



localhost/PWL_POS/public/barang

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Supplier

Supplier

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Barang

Filter: Semua

Kategori Barang

Show 10 entries

Search:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000	3.500	Minuman	Detail Edit Hapus

Import Barang

Tambah Data (Ajax)

localhost/PWL_POS/public/barang

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Supplier

Supplier

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Barang

Filter: Semua

Kategori Barang

Show 10 entries

Search:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000	3.500	Minuman	Detail Edit Hapus

Import Barang

Tambah Data (Ajax)

Import Data Barang

Download Template

Pilih File

Choose File

No file chosen

Batal

Upload

localhost/PWL_POS/public/barang

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Supplier

Supplier

Data Transaksi

Stok Barang

Transaksi Penjualan

Home Contact

Daftar Barang

Filter: Semua

Kategori Barang

Show 10 entries

Search:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000	3.500	Minuman	Detail Edit Hapus

Import Barang

Tambah Data (Ajax)

Import Data Barang

Download Template

Pilih File

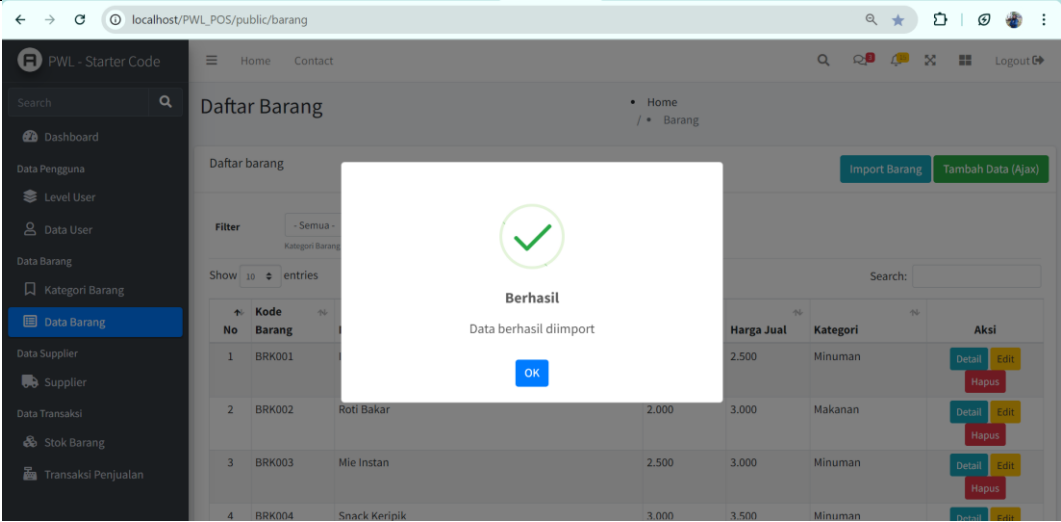
Choose File

template_barang.xlsx

Batal

Upload



	
9	Kita upload template data yang sudah kita siapkan, dan amati apa yang terjadi.

Tugas 1 – Implementasi File Upload untuk Import Data:

1. Silahkan implementasikan praktikum 1 pada project kalian masing-masing
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
3. Submit kode untuk implementasi praktikum 1 pada repository github kalian.

Praktikum Ke-2 – Export Data ke Excel

Langkah	Jawaban/Deskripsi
1	<p>Memodifikasi <code>barang/index.blade.php</code> dengan mengganti kode berikut</p> <pre>{{-- Praktikum 2 JS08 --}} <i class="fa fa-file-excel"></i> Export Barang</pre> <p>Hal ini kita lakukan karena tombol Tambah Data sudah tidak kita gunakan karena kita sudah menggunakan tombol Tambah Data (Ajax).</p>
2	<p>Kemudian kita tambahkan route pada <code>route/web.php</code> untuk bisa memproses export excel</p> <pre>Route::get('/export_excel', [BarangController::class, 'export_excel']); // export excel Route::delete('/{id}', [BarangController::class, 'destroy']);</pre>
3	Selanjutnya kita tambahkan fungsi <code>export_excel()</code> pada file <code>BarangController.php</code>



4	<p>Kita ambil data barang yang akan kita export ke excel (tentu dengan menyertakan relasi kategori barang)</p> <pre>public function export_excel() { // ambil data barang yang akan di export \$barang = BarangModel::select('kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual') ->orderBy('kategori_id') ->with('kategori') ->get(); }</pre>
5	<p>Kemudian kita load library Spreadsheet dan kita tentukan header data pada baris pertama di excel</p> <pre>// load library excel \$spreadsheet = new Spreadsheet(); \$sheet = \$spreadsheet->getActiveSheet(); \$sheet->setCellValue('A1', 'No'); \$sheet->setCellValue('B1', 'Kode Barang'); \$sheet->setCellValue('C1', 'Nama Barang'); \$sheet->setCellValue('D1', 'Harga Beli'); \$sheet->setCellValue('E1', 'Harga Jual'); \$sheet->setCellValue('F1', 'Kategori'); \$sheet->getStyle('A1:F1')->getFont()->setBold(true);</pre>
6	<p>Selanjutnya, kita looping data yang telah kita dapatkan dari database, kemudian kita masukkan ke dalam cell excel</p> <pre>\$no = 1; // nomor data dimulai dari 1 \$baris = 2; // baris data dimulai dari baris ke 2 foreach (\$barang as \$key => \$value) { \$sheet->setCellValue('A' . \$baris, \$no); \$sheet->setCellValue('B' . \$baris, \$value->barang_kode); \$sheet->setCellValue('C' . \$baris, \$value->barang_nama); \$sheet->setCellValue('D' . \$baris, \$value->harga_beli); \$sheet->setCellValue('E' . \$baris, \$value->harga_jual); \$sheet->setCellValue('F' . \$baris, \$value->kategori->kategori_nama); \$baris++; \$no++; }</pre>
7	<p>Kita set lebar tiap kolom di excel untuk menyesuaikan dengan panjang karakter pada masing-masing kolom</p>



```
foreach (range('A', 'F') as $columnID) {  
    $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size untuk kolom  
}
```

8

Bagian akhir proses export excel adalah kita set nama sheet, dan proses untuk dapat di download oleh pengguna

```
$sheet->setTitle('Data Barang'); // set title sheet  
  
$writer = IOFactory::createWriter($spreadsheet, 'Xlsx');  
$filename = 'Data Barang ' . date('Y-m-d H:i:s') . '.xlsx';  
  
header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');  
header('Content-Disposition: attachment;filename="' . $filename . '"');  
header('Cache-Control: max-age=0');  
header('Cache-Control: max-age=1');  
header('Expires: Mon, 26 Jul 1997 05:00:00 GMT');  
header('Last-Modified: ' . gmdate('D, dMY H:i:s') . ' GMT');  
header('Cache-Control: cache, must-revalidate');  
header('Pragma: public');  
  
$writer->save('php://output');  
exit;  
} // end function export_excel
```

9

Jika sudah selesai diimplementasikan, kita coba untuk melakukan Download file Export tersebut.

The screenshot shows a web application interface for managing goods. The browser address bar shows 'localhost/PWL_POS/public/barang'. The application has a sidebar with a search bar and a list of menu items: Dashboard, Data Pengguna, Level User, Data User, Data Barang, Kategori Barang, Data Supplier, Supplier, Data Transaksi, Stok Barang, and Transaksi Penjualan. The 'Data Barang' menu is highlighted. The main content area is titled 'Daftar Barang' and includes buttons for 'Impor Barang', 'Export Barang', and 'Tambah Data (Ajax)'. Below these buttons is a filter dropdown set to 'Semua' and a search bar. A table displays a list of goods with the following columns: ID, Kode Barang, Nama Barang, Harga Beli, Harga Jual, Kategori Barang, and Aksi. The table contains 7 rows of data, each with a 'Detail', 'Edit', and 'Hapus' button in the 'Aksi' column.

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000	3.500	Minuman	Detail Edit Hapus
5	BRK006	Teh Botol	2.500	3.000	Makanan	Detail Edit Hapus
6	BRK007	Air Mineral	1.000	1.500	Makanan	Detail Edit Hapus
7	BRK008	Kopi Bubuk	2.000	2.500	Makanan	Detail Edit Hapus



localhost/PWL_POS/public/barang

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Supplier

Supplier

Data Transaksi

Stok Barang

Transaksi Penjualan

Daftar Barang

Filter: Semua

Kategori Barang

Show 10 entries

Search:

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000			
5	BRK006	Teh Botol	2.500			
6	BRK007	Air Mineral	1.000			
7	BRK008	Jus Jeruk	5.000	6.000	Makanan	Detail Edit Hapus

Received Files

Data Barang 2024-10-17 16_01_35.xlsx

C:\Users\LENOVO\Downloads

Open

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRK001	Indomie Goreng	2200	2500	Minuman
2	BRK003	Mie Instan	2500	3000	Minuman
3	BRK004	Snack Keripik	3000	3500	Minuman
4	SBK-003	Telur Omega (10 butir)	22000	25000	Minuman
5	BRK002	Roti Bakar	2000	3000	Makanan
6	BRK006	Teh Botol	2500	3000	Makanan
7	BRK007	Air Mineral	1000	1500	Makanan
8	BRK008	Jus Jeruk	5000	6000	Makanan
9	BRK009	Kopi Instan	3000	3500	Makanan
10	BRK010	Soda	4000	4500	Makanan
11	BRK068	Milo Cubes	9500	10000	Makanan
12	SNK-003	Sari Roti	11500	12500	Makanan
13	BRK011	Kaos Polos	30000	35000	Pakaian
14	BRK012	Jaket	70000	80000	Pakaian
15	BRK014	Kemeja	45000	50000	Pakaian
16	BRK015	Dress	60000	70000	Pakaian
17	MND-003	Shampoo Pantene	17500	18500	Pakaian
18	BAY-003	Baju Bayi 2th	89000	92500	Kendaraan
19	BRK20	Cat Akrilik	20000	50000	Peralatan
20	MNM-003	Cle 600mL	3750	4300	Peralatan

Tugas 2 – Implementasi File Export Excel:

1. Silahkan implementasikan praktikum 2 pada project kalian masing-masing
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
3. Submit kode untuk impementasi prakktikum 2 pada repository github kalian.

Praktikum Ke-3 – Implementasi Export PDF di Laravel dengan dompdf

Langkah	Jawaban/Deskripsi
1	Kita lakukan proses instalasi library dompdf terlebih dahulu dengan mengetikkan perintah pada terminal/CMD



	<p>composer require barryvdh/laravel-dompdf</p> <pre>C:\laragon\www\PMIL_POS>composer require barryvdh/laravel-dompdf ./composer.json has been updated Running composer update barryvdh/laravel-dompdf Loading composer repositories with package information Updating dependencies Lock file operations: 6 installs, 0 updates, 0 removals - Locking barryvdh/laravel-dompdf (v3.0.0) - Locking dompdf/dompdf (v3.0.0) - Locking dompdf/php-font-lib (1.0.0) - Locking dompdf/php-svg-lib (1.0.0) - Locking masterminds/html5 (2.9.0) - Locking sabberworm/php-css-parser (v8.6.0) Writing lock file Installing dependencies from lock file (including require-dev) Package operations: 6 installs, 0 updates, 0 removals - Downloading masterminds/html5 (2.9.0) - Downloading sabberworm/php-css-parser (v8.6.0) - Downloading dompdf/php-svg-lib (1.0.0) - Downloading dompdf/php-font-lib (1.0.0) - Downloading dompdf/dompdf (v3.0.0) - Downloading barryvdh/laravel-dompdf (v3.0.0) - Installing masterminds/html5 (2.9.0): Extracting archive - Installing sabberworm/php-css-parser (v8.6.0): Extracting archive - Installing dompdf/php-svg-lib (1.0.0): Extracting archive - Installing dompdf/php-font-lib (1.0.0): Extracting archive - Installing dompdf/dompdf (v3.0.0): Extracting archive - Installing barryvdh/laravel-dompdf (v3.0.0): Extracting archive 2 package suggestions were added by new dependencies, use 'composer suggest' to see details. Generating optimized autoload files Class Database\Seeders\PenjualanDetailSeeder located in ./database/seeders/PenjualanDetailSeeder.php does not comply with psr-4 autoloading standard (rule: Database\Seeders => ./database/seeders). Skipping. Class App\Models\kategoriModel located in ./app/Models/kategoriModel.php does not comply with psr-4 autoloading standard (rule: App\ => ./app). Skipping. Class App\Models\level located in ./app/Models/level.php does not comply with psr-4 autoloading standard (rule: App\ => ./app). Skipping. > illuminate\Foundation\ComposerScripts::postAutoloadDump > @php artisan package:discover --ansi INFO Discovering packages. barryvdh/laravel-dompdf DONE laravel/sail DONE laravel/sanctum DONE laravel/tinker DONE nesbot/carbon DONE nunomaduro/collision DONE nunomaduro/termwind DONE spatie/laravel-ignition DONE yajra/laravel-datatables-oracle DONE 85 packages you are using are looking for funding. Use the 'composer fund' command to find out more! > @php artisan vendor:publish --tags=laravel-assets --ansi --force INFO No publishable resources for tag [laravel-assets]. No security vulnerability advisories found. Using version ^3.0 for barryvdh/laravel-dompdf C:\laragon\www\PMIL_POS></pre>
2	<p>Setelah proses instalasi dompdf berhasil, selanjutnya Kita tambahkan kode berikut untuk menambahkan tombol export ke pdf pada barang/index.blade.php</p> <pre>{{-- JS08 Praktikum 3 --}} <i class="fa fa-file-pdf"></i> Export Barang</pre>
3	<p>Setelah itu tinggal kita perbaiki route/web.php untuk proses export pdf</p> <pre>// JS08 Praktikum 3 Route::get('/export_pdf', [BarangController::class, 'export_pdf']); // export pdf</pre>
4	<p>Selanjutnya, kita buat fungsi export_pdf() pada BarangController.php</p> <pre>use Barryvdh\DomPDF\Facade\Pdf; public function export_pdf()</pre>



	<pre>{ \$barang = BarangModel::select('kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual') ->orderBy('kategori_id') ->orderBy('barang_kode') ->with('kategori') ->get(); // use Barryvdh\DomPDF\Facade\Pdf; \$pdf = Pdf::loadView('barang.export_pdf', ['barang' => \$barang]); \$pdf->setPaper('a4', 'portrait'); // set ukuran kertas dan orientasi \$pdf->setOption("isRemoteEnabled", true); // set true jika ada gambar dari url \$pdf->render(); return \$pdf->stream('Data Barang'.date('Y-m-d H:i:s').'.pdf'); }</pre>
5	<p>Selanjutnya, kita buat view untuk dijadikan pdf dari layout html. File bisa kita buat dengan nama <code>barang/export_pdf.blade.php</code></p> <pre><html> <head> <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/> <style> body { font-family: "Times New Roman", Times, serif; margin: 6px 20px 5px 20px; line-height: 15px; } table { width: 100%; border-collapse: collapse; } td, th { padding: 4px 3px; } th { text-align: left; } .d-block { display: block; } img.image { width: auto; height: 80px;</pre>



```
        max-width: 150px;
        max-height: 150px;
    }
    .text-right {
        text-align: right;
    }
    .text-center {
        text-align: center;
    }
    .p-1 {
        padding: 5px 1px 5px 1px;
    }
    .font-10 {
        font-size: 10pt;
    }
    .font-11 {
        font-size: 11pt;
    }
    .font-12 {
        font-size: 12pt;
    }
    .font-13 {
        font-size: 13pt;
    }
    .border-bottom-header {
        border-bottom: 1px solid;
    }
    .border-all, .border-all th, .border-all td {
        border: 1px solid;
    }
</style>
</head>
<body>
    <table class="border-bottom-header">
        <tr>
            <td width="15%" class="text-center">
                
            </td>
            <td width="85%">
                <span class="text-center d-block font-11 font-bold mb-1">KEMENTERIAN PENDIDIKAN,
                KEBUDAYAAN, RISET, DAN TEKNOLOGI</span>
                <span class="text-center d-block font-13 font-bold mb-1">POLITEKNIK NEGERI MALANG</span>
                <span class="text-center d-block font-10">Jl. Soekarno-Hatta No. 9 Malang 65141</span>
                <span class="text-center d-block font-10">Telepon (0341) 404424 Pes. 101-105, 0341-404420,
                Fax. (0341) 404420</span>
            </td>
        </tr>
    </table>
</body>
</html>
```



6

Selanjutnya, Kita coba untuk melakukan proses download export pdf.

localhost/PWL_POS/public/barang

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Supplier

Supplier

Data Transaksi

Stok Barang

Transaksi Penjualan

Daftar Barang

Home / Barang

Daftar Barang

Import Barang Export Barang Export Barang Tambah Data (Ajax)

Filter - Semua - Kategori Barang

Show 10 entries Search:

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BRK001	Indomie Goreng	2.200	2.500	Minuman	Detail Edit Hapus
2	BRK002	Roti Bakar	2.000	3.000	Makanan	Detail Edit Hapus
3	BRK003	Mie Instan	2.500	3.000	Minuman	Detail Edit Hapus
4	BRK004	Snack Keripik	3.000	3.500	Minuman	Detail Edit Hapus
5	BRK006	Teh Botol	2.500	3.000	Makanan	Detail Edit Hapus
6	BRK007	Air Mineral	1.000	1.500	Makanan	Detail Edit Hapus

localhost/PWL_POS/public/barang/export_pdf

export_pdf

1 / 1 100% +

Data Barang2024-10-18 06_35_10.pdf 4.9 KB Done

Data Barang 2024-10-18 06_28_25.xlsx 7.1 KB 8 minutes ago

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
Jl. Soekarno-Hatta No. 9 Malang 65141
Telepon (0341) 404424 Pcs. 101- 105, 0341-404420, Fax. (0341) 404420
Laman: www.polinema.ac.id

LAPORAN DATA BARANG

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRK001	Indomie Goreng	2.200	2.500	Minuman
2	BRK003	Mie Instan	2.500	3.000	Minuman
3	BRK004	Snack Keripik	3.000	3.500	Minuman
4	SBK-003	Telur Omega (10 butir)	22.000	25.000	Minuman
5	BRK002	Roti Bakar	2.000	3.000	Minuman
6	BRK006	Teh Botol	2.500	3.000	Makanan
7	BRK007	Air Mineral	1.000	1.500	Makanan
8	BRK008	Jus Jeruk	5.000	6.000	Makanan
9	BRK009	Kopi Instan	3.000	3.500	Makanan
10	BRK010	Soda	4.000	4.500	Makanan
11	BRK068	Milo Cubes	9.500	10.000	Makanan
12	SNK-003	Sari Roti	11.500	12.500	Makanan
13	BRK011	Kaos Polos	30.000	35.000	Pakaian

Tugas 3 – Implementasi Export PDF pada Laravel :

1. Silahkan implementasikan export pdf pada project kalian masing-masing untuk semua menu
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

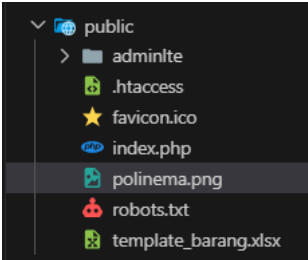


3. Submit kode untuk implementasi export pdf pada repository github kalian

Tugas 4 – Implementasi Upload File Gambar :

1. Silahkan implementasikan fitur **upload file** untuk mengubah **foto profile** di project web kalian
2. Jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
3. Submit kode untuk implementasi export pdf pada repository github kalian.

Praktikum Tugas ke-4

Langkah	Jawaban/Deskripsi
1	<p>Menambahkan gambar logo polinema untuk profile file pdf pada file public.</p> 
2	<p>Selanjutnya, memodifikasi file export_pdf.blade.php pada resources/views/barang</p> ="" >.="" <="" <body>="" <html>="" <img="" <pre>1="" <span="" <table="" <td="" <tr>="" (0341)="" 0341-404420,="" 10="" 101-105,="" 11="" 12="" 13="" 14="" 15="" 16="" 17="" 18="" 19="" 2="" 20="" 21="" 22="" 23="" 24="" 25="" 26="" 27="" 28="" 3="" 4="" 404420="" 404424="" 5="" 6="" 65141="" 7="" 8="" 9="" alt="Screenshot of a code editor showing the 'export_pdf.blade.php' file. The code is in a dark theme. A red box highlights the line: ="" class='\"text-center' code="" d-block="" dan="" fax.="" follows:="" font-10\">="" font-11="" font-13="" font-bold="" html><="" is="" jl.="" kebudayaan,="" kementerian="" laman:="" malang="" mb-1\">="" negeri="" no.="" pendidikan,="" pes.="" politeknik="" pre="" riset,="" soekarno-hatta="" span>="" src='\"{{' table>="" td>="" teknologi="" telepon="" the="" tr>="" width='\"85%\">' www.polinema.ac.id="" {{="" }}\"=""/>



3

Dengan menambahkan gambar profile di text pdf

Hasil implementasi upload file gambar

