

LAPORAN PRAKTIKUM
JOBSHEET 08
FILE IMPORT DAN EXPORT

Disusun untuk memenuhi nilai tugas
Mata Kuliah : Pemrograman Web Lanjut



Oleh :
Aqueena Regita Hapsari
2341760096
SIB 2B
03

PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS
POLITEKNIK NEGERI MALANG
TAHUN AJARAN 2024/2025

Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 4 (empat) / 5 (lima)
Pertemuan ke- : 8 (delapan)

JOBSHEET 08

File Import dan Export ke PDF dan Excel pada Laravel

Sesuai dengan **studi Kasus PWL.pdf**.

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

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

A. File Upload (import data) pada Laravel

Mengunggah file merupakan salah satu fitur esensial yang sering kali dijumpai di berbagai proyek website. Fitur ini memberikan kemampuan bagi pengguna untuk memindahkan file dari perangkat komputer lokal mereka ke dalam lingkungan server aplikasi web. Proses ini melibatkan pengguna dalam memilih file dari sistem lokal mereka, seperti gambar, dokumen, atau media lainnya, yang kemudian diunggah melalui formulir yang disediakan di website. Setelah file tersebut berhasil diunggah, server akan menyimpan file tersebut di direktori tertentu yang telah ditentukan sebelumnya oleh pengembang. Penyimpanan ini dapat berada di server lokal, penyimpanan awan (*cloud storage*), atau bahkan dalam jaringan distribusi konten (CDN) untuk memastikan kinerja dan ketersediaan yang optimal. Dengan fitur unggah file ini, pengguna dapat memperkaya konten website secara dinamis, sementara pengembang dapat memastikan bahwa file yang diunggah dikelola dengan baik, aman, dan sesuai dengan kebijakan aplikasi.

Mengunggah file bukan hanya digunakan untuk menambahkan media atau dokumen ke sebuah website, tetapi juga memiliki peran penting dalam proses pengelolaan data, khususnya untuk *import* data ke database dalam aplikasi berbasis Laravel. Fitur ini memungkinkan

pengguna untuk mengunggah file berisi data dalam format seperti CSV atau Excel, yang kemudian diproses oleh aplikasi untuk dimasukkan ke dalam database.

Proses ini biasanya dimulai dengan pengguna memilih file yang berisi data dari komputer lokal mereka dan mengunggahnya ke server melalui formulir yang disediakan di aplikasi web. Setelah file berhasil diunggah, Laravel akan memproses file tersebut menggunakan paket seperti Maatwebsite Excel, yang dirancang khusus untuk menangani impor data dari file Excel atau CSV. Dalam jobsheet ini, kita akan mempelajari bagaimana cara mengunggah file yang digunakan untuk proses input data ke dalam database (import) menggunakan Laravel.

Praktikum 1 – Implementasi Upload File untuk import data:

1. Kita buka database yang sudah kita buat sebelumnya, dan kita cek tabel `m_kategori` dan `m_barang`

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	SBK	Sembako	2024-08-16 02:57:53	NULL
2	SNK	Makanan Ringan	2024-08-16 02:57:53	NULL
3	MND	Peralatan Mandi	2024-08-16 02:57:53	NULL
4	BAY	Keperluan Bayi	2024-08-16 02:57:53	NULL
5	MNM	Minuman	2024-08-16 02:57:53	NULL

barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
1	1	SBK-001	Beras Cap Jago (5kg)	65000	68000	2024-08-19 20:07:06	NULL
2	1	SBK-002	Beras Bramo Cap Lele	80000	83000	2024-08-19 20:07:06	NULL
3	2	SNK-001	Happy Toss	10500	11000	2024-08-19 20:07:06	NULL
4	2	SNK-002	Oreo	7200	7800	2024-08-19 20:07:06	NULL
5	3	MND-001	Sabun Lifebouy	4250	5000	2024-08-19 20:07:06	NULL
6	3	MND-002	Pasta Gigi Pepsoden	6750	7500	2024-08-19 20:07:06	NULL
7	4	BAY-001	Susu SGM Coklat 900gr	92500	95000	2024-08-19 20:07:06	NULL
8	4	BAY-002	Popok Mamy Poko	56000	58000	2024-08-19 20:07:06	NULL
9	5	MNM-001	Aqua 600ml	3700	4500	2024-08-19 20:07:06	NULL
10	5	MNM-002	Le Mineral	3500	4000	2024-08-19 20:07:06	NULL

2. Disini kita akan 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
1	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	
2		1 SBK-003	Telur Omega (10 butir)	22000	25000	
3		2 SNK-003	Sari Roti	11500	12500	
4		3 MND-003	Shampo Pantene	17500	18500	
5		4 BAY-003	Baju Bayi 2th	89000	92500	
6		5 MNM-003	Cleo 600ml	3750	4300	
7						

Contoh kita ingin mengimport 5 data barang sekaligus. Kita simpan ke file excel dengan nama `template_barang.xlsx`, dan kita simpan pada folder `public` di project web kita.

- Selanjutnya kita 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>
            @if(session('success'))
                <div class="alert alert-success">{{ session('success') }}</div>
            @endif
            @if(session('error'))
                <div class="alert alert-danger">{{ session('error') }}</div>
            @endif
        </div>
    </div>
</section>
```

```

        <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
        },{
            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);
        }
    },{
        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

```

4. Selanjutnya kita buat view untuk form upload/import file excel dan download file `template_barang.xlsx`. Kita beri nama file dengan nama `barang/import.blade.php`

```

<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"><span aria-hidden="true">&times;</span></button>
            </div>
            <div class="modal-body">
                <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>
</div>
</form>
<script>
    $(document).ready(function() {
        $("#form-import").validate({
            rules: {
                file_barang: {required: true, extension: "xlsx"},
            },
            submitHandler: function(form) {
                var formData = new FormData(form); // Jadikan form ke FormData untuk
menghandle file

                $.ajax({
                    url: form.action,
                    type: form.method,
                    data: formData, // Data yang dikirim berupa FormData
                    processData: false, // setting processData dan contentType ke false,
untuk menghandle file
                    contentType: false,
                    success: function(response) {
                        if(response.status){ // jika sukses
                            $('#myModal').modal('hide');
                            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>

```

5. Kemudian kita modifikasi `route/web.php` untuk mengakomodir proses upload file pada menu barang

```

Route::get('/barang',[BarangController::class,'index']);
Route::post('/barang/list',[BarangController::class,'list']);
Route::get('/barang/create_ajax',[BarangController::class,'create_ajax']); // ajax form create
Route::post('/barang_ajax',[BarangController::class,'store_ajax']); // ajax store
Route::get('/barang/{id}/edit_ajax',[BarangController::class,'edit_ajax']); // ajax form edit
Route::put('/barang/{id}/update_ajax',[BarangController::class,'update_ajax']); // ajax update
Route::get('/barang/{id}/delete_ajax',[BarangController::class,'confirm_ajax']); // ajax form confirm
Route::delete('/barang/{id}/delete_ajax',[BarangController::class,'delete_ajax']); // ajax delete
Route::get('/barang/import',[BarangController::class,'import']); // ajax form upload excel
Route::post('/barang/import_ajax',[BarangController::class,'import_ajax']); // ajax import excel

```

6. Untuk bisa membaca/menulis file excel, maka kita butuh library untuk membaca/menulis file excel. Jadi kita bisa memakai library [phpoffice/phpspreadsheet](#). Kita ketikkan perintah di terminal/CMD

```
composer require phpoffice/phpspreadsheet
```

7. Selanjutnya kita modifikasi file [BarangController.php](#) untuk memproses data

```

<?php
namespace App\Http\Controllers;

use App\Models\LevelModel;
use App\Models\BarangModel;
use App\Models\KategoriModel;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Validator;
use PhpOffice\PhpSpreadsheet\IOFactory;
use Yajra\DataTables\Facades\DataTables;

class BarangController extends Controller
{
    public function index (){
        $activeMenu = 'barang';
        $breadcrumb = (object) [
            'title' => 'Data Barang',
            'list' => ['Home', 'Barang']
        ];

        $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();
        return view('barang.index', [
            'activeMenu' => $activeMenu,
            'breadcrumb' => $breadcrumb,
            'kategori' => $kategori
        ]);
    }

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

        $kategori_id = $request->input('filter_kategori');
        if(!empty($kategori_id)){
            $barang->where('kategori_id', $kategori_id);
        }

        return DataTables::of($barang)
            ->addIndexColumn()
            ->addColumn('aksi', function ($barang) { // menambahkan kolom aksi

```



```

        /*$btn = '<a href="'.url('/barang/' . $barang->barang_id).'"' class="btn btn-
info btn-sm">Detail</a> ' ;
        $btn .= '<a href="'.url('/barang/' . $barang->barang_id .
'/edit').'" class="btn btn-warning btn-sm">Edit</a> ' ;
        $btn .= '<form class="d-inline-block" method="POST" action="'.
url('/barang/' . $barang->barang_id).'"'
        . csrf_field() . method_field('DELETE') .
        '<button type="submit" class="btn btn-danger btn-sm" onclick="return
confirm(\'Apakah Kita yakin menghapus data ini?\')">Hapus</button></form>';*/
        $btn = '<button onclick="modalAction(\'.url('/barang/' . $barang->barang_id
. '/show_ajax').'\')"' class="btn btn-info btn-sm">Detail</button> ' ;
        $btn .= '<button onclick="modalAction(\'.url('/barang/' . $barang-
>barang_id . '/edit_ajax').'\')"' class="btn btn-warning btn-sm">Edit</button> ' ;
        $btn .= '<button onclick="modalAction(\'.url('/barang/' . $barang-
>barang_id . '/delete_ajax').'\')"' class="btn btn-danger btn-sm">Hapus</button> ' ;
        return $btn;
    })
    ->rawColumns(['aksi']) // ada teks html
    ->make(true);
}

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

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

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

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

public function edit_ajax($id)
{
    $barang = BarangModel::find($id);
    $level = LevelModel::select('level_id', 'level_nama')->get();
    return view('barang.edit_ajax', ['barang' => $barang, 'level' => $level]);
}

public function update_ajax(Request $request, $id)
{
    // cek apakah request dari ajax

```

```

        if ($request->ajax() || $request->wantsJson()) {
            $rules = [
                'kategori_id' => ['required', 'integer', 'exists:m_kategori,kategori_id'],
                'barang_kode' => ['required', 'min:3', 'max:20',
                'unique:m_barang,barang_kode, '. $id .',barang_id'],
                'barang_nama' => ['required', 'string', 'max:100'],
                'harga_beli' => ['required', 'numeric'],
                'harga_jual' => ['required', 'numeric'],
            ];

            // 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
                ]);
            }

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

    public function confirm_ajax($id)
    {
        $barang = BarangModel::find($id);
        return view('barang.confirm_ajax', ['barang' => $barang]);
    }

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

    public function import()
    {
        return view('barang.import');
    }
}

```

```

public function import_ajax(Request $request)
{
    if($request->ajax() || $request->wantsJson()){
        $rules = [
            // validasi file harus xls atauxlsx, max 1MB
            'file_barang' => ['required', 'mimes:xlsx', 'max:1024']
        ];

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

        $file = $request->file('file_barang'); // ambil file dari request

        $reader = IOFactory::createReader('Xlsx'); // load reader file excel
        $reader->setReadDataOnly(true); // hanya membaca data
        $spreadsheet = $reader->load($file->getRealPath()); // load file excel
        $sheet = $spreadsheet->getActiveSheet(); // ambil sheet yang aktif

        $data = $sheet->toArray(null, false, true, true); // ambil data excel

        $insert = [];
        if(count($data) > 1){ // jika data lebih dari 1 baris
            foreach ($data as $baris => $value) {
                if($baris > 1){ // baris ke 1 adalah header, maka lewati
                    $insert[] = [
                        'kategori_id' => $value['A'],
                        'barang_kode' => $value['B'],
                        'barang_nama' => $value['C'],
                        'harga_beli' => $value['D'],
                        'harga_jual' => $value['E'],
                        'created_at' => now(),
                    ];
                }
            }
        }

        if(count($insert) > 0){
            // insert data ke database, jika data sudah ada, maka diabaikan
            BarangModel::insertOrIgnore($insert);
        }

        return response()->json([
            'status' => true,
            'message' => 'Data berhasil diimport'
        ]);
    }else{
        return response()->json([
            'status' => false,
            'message' => 'Tidak ada data yang diimport'
        ]);
    }
}
return redirect('/');
}
}

```

8. Sekarang kita coba jalankan browser dan klik tombol import pada menu Barang

Import Data Barang

Download Template

Download

Pilih File

Browse...

No file selected.

Batal

Upload

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

Hasil :

19	SBK-003	Telur Omega (10 Butir)	22,000	25,000	Elektronik	Detail	Edit	Hapus
20	SNK-003	Sari Roti	11,500	12,500	Cemilan	Detail	Edit	Hapus
21	MND-003	Shampoo Pantene	17,500	18,500	Minuman	Detail	Edit	Hapus
22	BAY-003	Baju Bayi 2th	89,000	92,500	Fashion	Detail	Edit	Hapus
23	MNM-003	Cleo 600ml	3,750	4,300	Otomotif	Detail	Edit	Hapus

Tugas 1 – Implementasi File Upload untuk Import Data:

1. Silahkan implementasikan praktikum 1 pada project kalian masing-masing untuk semua menu

Jawaban : Baik (Kode program bisa lihat di link github

https://github.com/aqwenaaa/PWL_2025/

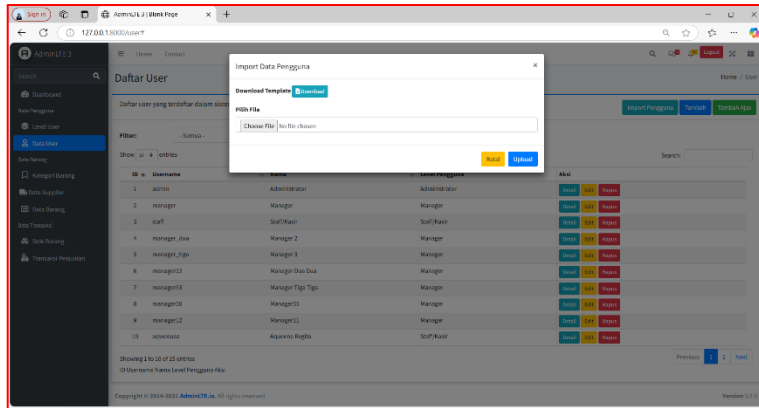
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

Jawaban : Untuk tiap tahapnya adaah memodifikasi dan menambahkan file,

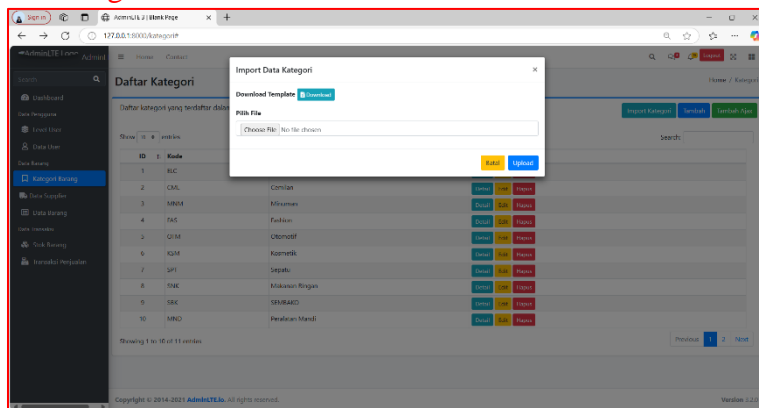
- Membuat template untuk tiap menu
- Modifikasi index untuk bagian button import saja
- Membuat file import baru
- Memodifikasi routing
- Memodifikasi controller dengan menambahkan function import dan import_excel

Hasil :

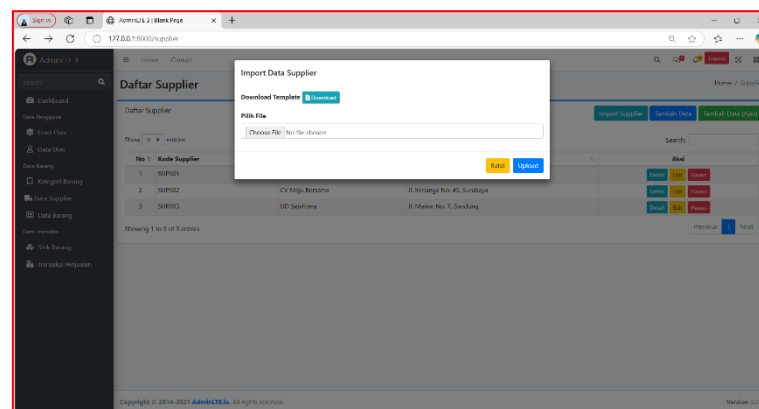
- 1) m user



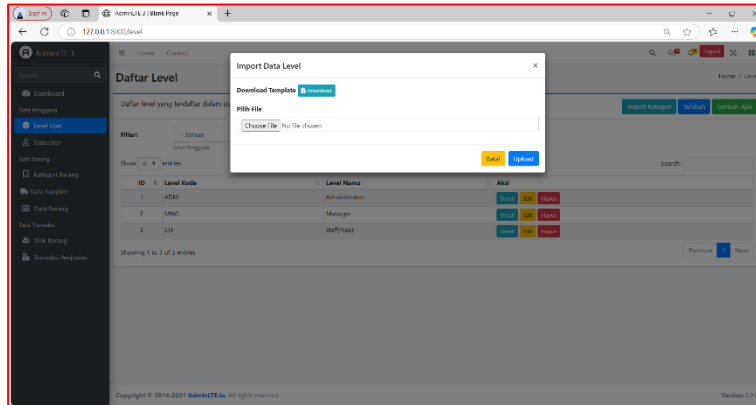
2) m_kategori



3) m_supplier



4) m_level



3. Submit kode untuk impementasi prakktikum 1 pada repository github kalian.

Jawaban : Sudah

B. Export Excel di Laravel

Fitur export ke Excel pada Laravel merupakan alat yang sangat berguna dalam aplikasi web yang membutuhkan pengolahan dan penyajian data secara profesional. Fitur ini memungkinkan pengguna untuk mengekspor data dari database atau hasil pemrosesan lainnya ke dalam format file Excel (.xlsx atau .xls) yang bisa diunduh dan digunakan untuk berbagai tujuan, seperti pelaporan, analisis data, atau penyimpanan arsip.

Fitur export ke Excel di Laravel memungkinkan Kita untuk menyediakan data dalam format yang lebih fleksibel dan mudah digunakan oleh pengguna akhir.

Praktikum 2 – Export Data ke Excel

Kita akan menerapkan export data dari database ke file excel pada project Laravel dengan menggunakan library yang sama dengan praktikum 1. Langkah-langkah yang kita kerjakan sebagai berikut:

1. Kita modifikasi `barang/index.blade.php` dengan mengganti kode berikut

```
<a href="{{ url('/barang/create') }}" class="btn btn-primary">Tambah Data</a>
```

Kita ubah dengan

```
<a href="{{ url('/barang/export_excel') }}" class="btn btn-primary"><i class="fa fa-file-excel"></i> Export Barang</a>
```

Hal ini kita lakukan karena tombol **Tambah Data** sudah tidak kita gunakan karena kita sudah menggunakan tombol **Tambah Data (Ajax)**.

2. Kemudian kita tambahkan route pada route/web.php untuk bisa memproses export excel

```
Route::get('/barang/import',[BarangController::class,'import']); // ajax form upload excel
Route::post('/barang/import_ajax',[BarangController::class,'import_ajax']); // ajax import excel
Route::get('/barang/export_excel',[BarangController::class,'export_excel']); // export excel
```

3. Selanjutnya kita tambahkan fungsi `export_excel()` pada file `BarangController.php`
4. Kita ambil data barang yang akan kita export ke excel (tentu dengan menyertakan relasi kategori barang)

```
232 public function export_excel()
233 {
234     // ambil data barang yang akan di export
235     $barang = BarangModel::select('kategori_id','barang_kode','barang_nama','harga_beli','harga_jual')
236         ->orderBy('kategori_id')
237         ->with('kategori')
238         ->get();
```

5. Kemudian kita load library Spreadsheet dan kita tentukan header data pada baris pertama di excel

```
240 // load library excel
241 $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
242 $sheet = $spreadsheet->getActiveSheet(); // ambil sheet yang aktif
243
244 $sheet->setCellValue('A1', 'No');
245 $sheet->setCellValue('B1', 'Kode Barang');
246 $sheet->setCellValue('C1', 'Nama Barang');
247 $sheet->setCellValue('D1', 'Harga Beli');
248 $sheet->setCellValue('E1', 'Harga Jual');
249 $sheet->setCellValue('F1', 'Kategori');
250
251 $sheet->getStyle('A1:F1')->getFont()->setBold(true); // bold header
```

6. Selanjutnya, kita looping data yang telah kita dapatkan dari database, kemudian kita masukkan ke dalam cell excel

```
253 $no = 1; // nomor data dimulai dari 1
254 $baris = 2; // baris data dimulai dari baris ke 2
255 foreach ($barang as $key => $value) {
256     $sheet->setCellValue('A'.$baris, $no);
257     $sheet->setCellValue('B'.$baris, $value->barang_kode);
258     $sheet->setCellValue('C'.$baris, $value->barang_nama);
259     $sheet->setCellValue('D'.$baris, $value->harga_beli);
260     $sheet->setCellValue('E'.$baris, $value->harga_jual);
261     $sheet->setCellValue('F'.$baris, $value->kategori->kategori_nama); // ambil nama kategori
262     $baris++;
263     $no++;
264 }
```

7. Kita set lebar tiap kolom di excel untuk menyesuaikan dengan panjang karakter pada masing-masing kolom

```
266 foreach(range('A','F') as $columnID) {
267     $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size untuk kolom
268 }
```

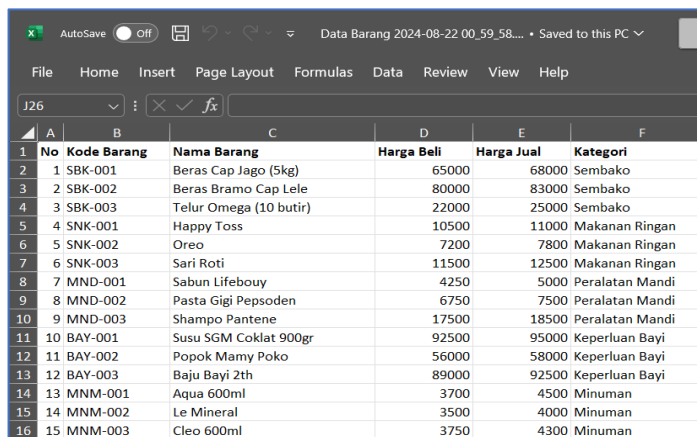
8. Bagian akhir proses export excel adalah kita set nama sheet, dan proses untuk dapat di download oleh pengguna

```

270 $sheet->setTitle('Data Barang'); // set title sheet
271
272 $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
273 $filename = 'Data Barang '.date('Y-m-d H:i:s').'.xlsx';
274
275 header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
276 header('Content-Disposition: attachment;filename="'.$filename.'");
277 header('Cache-Control: max-age=0');
278 header('Cache-Control: max-age=1');
279 header('Expires: Mon, 26 Jul 1997 05:00:00 GMT');
280 header('Last-Modified: ' . gmdate('D, d M Y H:i:s') . ' GMT');
281 header('Cache-Control: cache, must-revalidate');
282 header('Pragma: public');
283
284 $writer->save('php://output');
285 exit;
286 } // end function export_excel

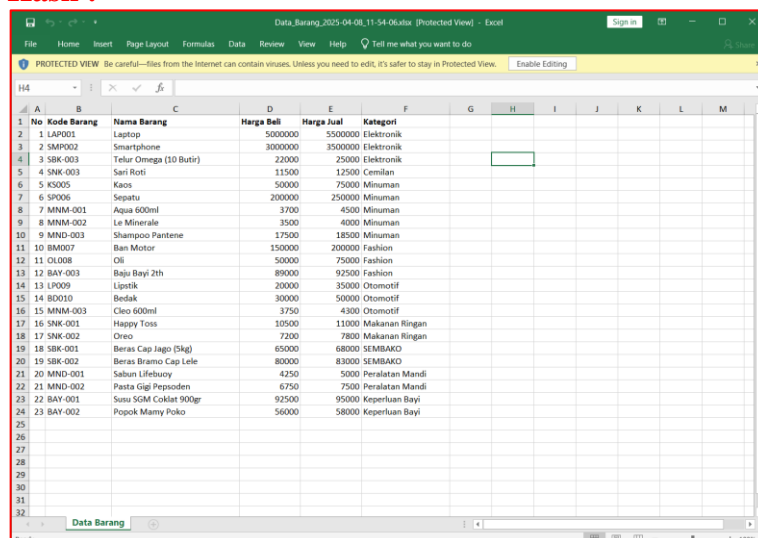
```

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



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	SBK-001	Beras Cap Jago (5kg)	65000	68000	Sembako
2	SBK-002	Beras Bramo Cap Lele	80000	83000	Sembako
3	SBK-003	Telur Omega (10 butir)	22000	25000	Sembako
4	SNK-001	Happy Toss	10500	11000	Makanan Ringan
5	SNK-002	Oreo	7200	7800	Makanan Ringan
6	SNK-003	Sari Roti	11500	12500	Makanan Ringan
7	MND-001	Sabun Lifebouy	4250	5000	Peralatan Mandi
8	MND-002	Pasta Gigi Pepsoden	6750	7500	Peralatan Mandi
9	MND-003	Shampo Pantene	17500	18500	Peralatan Mandi
10	BAY-001	Susu SGM Coklat 900gr	92500	95000	Keperluan Bayi
11	BAY-002	Popok Mamy Poko	56000	58000	Keperluan Bayi
12	BAY-003	Baju Bayi 2th	89000	92500	Keperluan Bayi
13	MNM-001	Aqua 600ml	3700	4500	Minuman
14	MNM-002	Le Mineral	3500	4000	Minuman
15	MNM-003	Cleo 600ml	3750	4300	Minuman

Hasil :



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	G	H	I	J	K	L	M
1	LAP001	Laptop	5000000	5500000	Elektronik							
2	SMPO02	Smartphone	3000000	3500000	Elektronik							
3	SBK-003	Telur Omega (10 Butir)	22000	25000	Elektronik							
4	SNK-003	Sari Roti	11500	12500	Camilan							
5	KSO05	Kaos	50000	75000	Minuman							
6	SPO06	Sepatu	200000	250000	Minuman							
7	MNM-001	Aqua 600ml	3700	4500	Minuman							
8	MNM-002	Le Minerale	3500	4000	Minuman							
9	MND-003	Shampoo Pantene	17500	18500	Minuman							
10	BM007	Ban Motor	150000	200000	Fashion							
11	OL008	Oli	50000	75000	Fashion							
12	BAY-003	Baju Bayi 2th	89000	92500	Fashion							
13	LP009	Lipstik	20000	35000	Otomotif							
14	BD010	Bendak	30000	50000	Otomotif							
15	MNM-003	Cleo 600ml	3750	4300	Otomotif							
16	SNK-001	Happy Toss	10500	11000	Makanan Ringan							
17	SNK-002	Oreo	7200	7800	Makanan Ringan							
18	SBK-001	Beras Cap Jago (5kg)	65000	68000	SEBBAKO							
19	SBK-002	Beras Bramo Cap Lele	80000	83000	SEBBAKO							
20	MND-001	Sabun Lifebouy	4250	5000	Peralatan Mandi							
21	MND-002	Pasta Gigi Pepsoden	6750	7500	Peralatan Mandi							
22	BAY-001	Susu SGM Coklat 900gr	92500	95000	Keperluan Bayi							
23	BAY-002	Popok Mamy Poko	56000	58000	Keperluan Bayi							

Tugas 2 – Implementasi File Export Excel:

1. Silahkan implementasikan praktikum 2 pada project kalian masing-masing untuk semua menu

Jawaban : Baik (Kode program bisa lihat di link github

https://github.com/aqwenaaa/PWL_2025/)

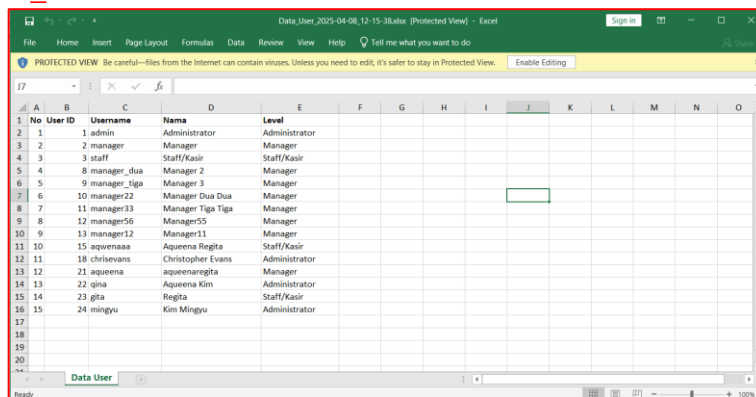
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

Jawaban :

- a. Memodifikasi button pada index.blade.php untuk mengganti dengan export excel
- b. Menambahkan routing untuk button setelah di klik akan diteruskan ke proses controller export
- c. Controller akan memproses export dengan ambil data dari datatables kemudian di load di library excel kemudian looping data tersebut dimasukkan kedalam excel dan set lebar kolom masing-masing menyesuaikan isi kolom.

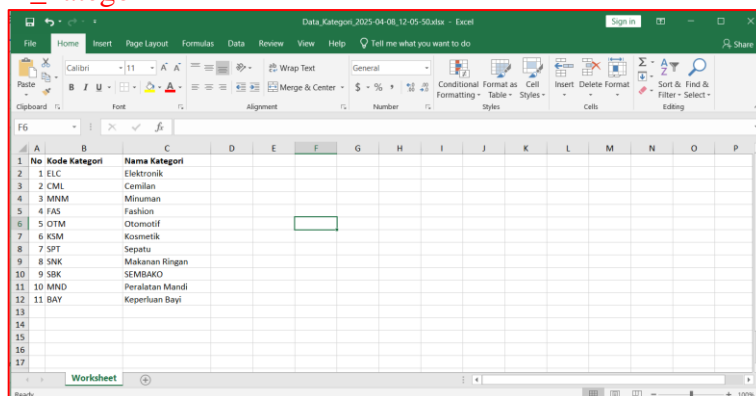
Hasil :

a. m_user



No	User ID	Username	Nama	Level
1	1	admin	Administrator	Administrator
2	2	manager	Manager	Manager
3	3	staff	Staff/Kasir	Staff/Kasir
4	8	manager_dua	Manager 2	Manager
5	9	manager_tiga	Manager 3	Manager
6	10	manager22	Manager Dua Dua	Manager
7	11	manager33	Manager Tiga Tiga	Manager
8	12	manager56	Manager55	Manager
9	13	manager12	Manager11	Manager
10	15	aqwenaaa	Aqueena Riegita	Staff/Kasir
11	18	chrisevans	Christopher Evans	Administrator
12	21	aqueena	aqueenaregita	Manager
13	22	gina	Aqueena Kim	Administrator
14	23	gita	Riegita	Staff/Kasir
15	24	mingyu	Kim Mingyu	Administrator

b. m_kategori



No	Kode Kategori	Nama Kategori
1	ELC	Elektronik
2	CML	Cemilan
3	MNM	Minuman
4	FAS	Fashion
5	OTM	Otomotif
6	KSM	Kosmetik
7	SPT	Sepatu
8	SNK	Makanan Ringan
9	SBK	SEMBAKO
10	MND	Peralatan Mandi
11	BAY	Keperluan Bayi

c. m_level

No	ID Level	Kode Level	Nama Level
1	1	1 ADM	Administrator
2	2	2 MNG	Manager
3	3	3 STF	Staff/Kasir

d. m_supplier

No	Supplier ID	Kode Supplier	Nama Supplier	Alamat Supplier
1	1	1 SUP001	PT Sumber Jaya	Jl. Merdeka No. 123, Jakarta
2	2	2 SUP002	CV Maju Bersama	Jl. Kenanga No. 45, Surabaya
3	3	3 SUP003	UD Sejahtera	Jl. Mawar No. 7, Bandung

3. Submit kode untuk impementasi prakktikum 2 pada repository github kalian.

Jawaban : Sudah

C. Export PDF di Laravel

Fitur **export ke PDF** pada Laravel sangat berguna dalam situasi di mana Kita perlu menghasilkan dokumen yang terstruktur, terformat dengan baik, dan mudah dibagikan. Dokumen PDF (Portable Document Format) adalah salah satu format yang paling sering digunakan untuk laporan, faktur, tiket, dan berbagai dokumen resmi lainnya karena stabilitasnya dan kemampuan untuk mempertahankan tata letak yang konsisten di berbagai perangkat.

Untuk mengimplementasikan export ke PDF di Laravel, salah satu paket yang paling populer adalah **dompdf**. Dengan menggunakan paket **dompdf**, Kita dapat dengan mudah mengubah tampilan HTML menjadi PDF yang dapat disesuaikan sesuai kebutuhan. Fitur ini sangat bermanfaat dalam berbagai konteks, seperti pembuatan faktur, laporan, sertifikat, dan dokumen-dokumen penting lainnya.

Praktikum 3 – Implementasi Export PDF di Laravel dengan **dompdf**

Kita akan menerapkan export data ke file pdf pada project Laravel dengan menggunakan library **dompdf**. Langkah-langkah yang kita kerjakan sebagai berikut:

1. Kita lakukan proses instalasi library **dompdf** terlebih dahulu dengan mengetikkan perintah pada terminal/CMD

```
composer require barryvdh/laravel-dompdf
```

2. Setelah proses instalasi **dompdf** berhasil, selanjutnya Kita tambahkan kode berikut untuk menambahkan tombol export ke pdf pada **barang/index.blade.php**

```
<a href="{{ url('/barang/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Barang</a>
```

3. Setelah itu tinggal kita perbaiki route/web.php untuk proses export pdf

```
Route::get('/barang/import',[BarangController::class,'import']); // ajax form upload excel
Route::post('/barang/import_ajax',[BarangController::class,'import_ajax']); // ajax import excel
Route::get('/barang/export_excel',[BarangController::class,'export_excel']); // export excel
Route::get('/barang/export_pdf',[BarangController::class,'export_pdf']); // export pdf
```

4. Selanjutnya, kita buat fungsi **export_pdf()** pada **BarangController.php**

```

public function export_pdf()
{
    $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');
}

```

5. Selanjutnya, kita buat view untuk dijadikan pdf dari layout html. File bisa kita buat dengan nama `barang/export_pdf.blade.php`

INFO

Perlu diingat, pada `export_pdf`, kita gunakan style CSS secara manual, dan tidak menggunakan framework CSS seperti bootstrap. Karena library `dmpdf` tidak support semua fitur styling yang ada di framework CSS.

6. Selanjutnya kita buat view untuk me-generate html untuk tampilan pdf yang akan kita sajikan. View berada di `barang/export_pdf.blade.php`

```

<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;
            max-width: 150px;
            max-height: 150px;
        }
        .text-right {
            text-align: right;
        }
    </style>

```

```

        .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>
                <span class="text-center d-block font-10">Laman: www.polinema.ac.id</span>
            </td>
        </tr>
    </table>

    <h3 class="text-center">LAPORAN DATA BARANG</h4>
    <table class="border-all">
        <thead>
            <tr>
                <th class="text-center">No</th>
                <th>Kode Barang</th>
                <th>Nama Barang</th>
                <th class="text-right">Harga Beli</th>
                <th class="text-right">Harga Jual</th>
                <th>Kategori</th>
            </tr>
        </thead>
        <tbody>
            @foreach($barang as $b)
            <tr>
                <td class="text-center">{{ $loop->iteration }}</td>
                <td>{{ $b->barang_kode }}</td>
                <td>{{ $b->barang_nama }}</td>
                <td class="text-right">{{ number_format($b->harga_beli, 0, ',', '.') }}</td>
                <td class="text-right">{{ number_format($b->harga_jual, 0, ',', '.') }}</td>
                <td>{{ $b->kategori->kategori_nama }}</td>
            </tr>
        </tbody>
    </table>

```

```

        @endforeach
    </tbody>
</table>
</body>
</html>

```

7. Selanjutnya, Kita coba untuk melakukan proses download export pdf. Amati dan pelajari....!!!

Hasil :

 <p>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</p>					
LAPORAN DATA BARANG					
No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	LAP001	Laptop	5.000.000	5.500.000	Elektronik
2	SBK-003	Telur Omega (10 Butir)	22.000	25.000	Elektronik
3	SMP002	Smartphone	3.000.000	3.500.000	Elektronik
4	SNK-003	Sari Roti	11.500	12.500	Cemilan
5	KS005	Kaos	50.000	75.000	Minuman
6	MND-003	Shampoo Pantene	17.500	18.500	Minuman
7	MNM-001	Aqua 600ml	3.700	4.500	Minuman
8	MNM-002	Le Minerale	3.500	4.000	Minuman
9	SP006	Sepatu	200.000	250.000	Minuman
10	BAY-003	Baju Bayi 2th	89.000	92.500	Fashion
11	BM007	Ban Motor	150.000	200.000	Fashion
12	OL008	Oli	50.000	75.000	Fashion
13	BD010	Bedak	30.000	50.000	Otomotif
14	LP009	Lipstik	20.000	35.000	Otomotif
15	MNM-003	Cleo 600ml	3.750	4.300	Otomotif
16	SNK-001	Happy Toss	10.500	11.000	Makanan Ringan
17	SNK-002	Oreo	7.200	7.800	Makanan Ringan
18	SBK-001	Beras Cap Jago (5kg)	65.000	68.000	SEMPAKO
19	SBK-002	Beras Brama Cap Lele	80.000	83.000	SEMPAKO
20	MND-001	Sabun Lifebuoy	4.250	5.000	Peralatan Mandi
21	MND-002	Pasta Gigi Pepsoden	6.750	7.500	Peralatan Mandi
22	BAY-001	Susu SGM Coklat 900gr	92.500	95.000	Keperluan Bayi
23	BAY-002	Popok Mamy Poko	56.000	58.000	Keperluan Bayi

Tugas 3 – Implementasi Export PDF pada Laravel :

- Silahkan implementasikan export pdf pada project kalian masing-masing untuk semua menu

Jawaban : Baik (Kode program bisa lihat di link github

https://github.com/aqwenaaa/PWL_2025/)

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

Jawaban :

Pada praktikum 3 ini, untuk export_pdf nya melalui pola yang sama, yakni

- Install library dompdf dulu supaya bisa render menjadi pdf
- Setelah itu menambahkan tombol export_pdf yang mengarah pada hasil export pdf
- Setelah itu menambahkan route export_pdf di tiap menu untuk proses export pdf
- Membuat fungsi export pdf pada controller, nah disini kita bisa set ukuran kertas dan orientasi, kemudian set true jika ada gambar dari url
- Buat view export_pdf untuk mengenerate (render) isi dari pdf nya. Pada tahap ini saya menambahkan logo polinema dengan menambahkan asset polinema-bw.png dari public/images/polinema-bw.png

Hasil:

1. m_user

<div><div>KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI POLITEKNIK NEGERI MALANG Jl. Soekarno-Hatta No. 9 Malang 65141 Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420 Laman: www.polinema.ac.id</div></div>				
LAPORAN DATA USER				
No	ID Pengguna	Username	Nama	Level
1	1	admin	Administrator	Administrator
2	2	manager	Manager	Manager
3	3	staff	Staff/Kasir	Staff/Kasir
4	8	manager_dua	Manager 2	Manager
5	9	manager_tiga	Manager 3	Manager
6	10	manager22	Manager Dua Dua	Manager
7	11	manager33	Manager Tiga Tiga	Manager
8	12	manager56	Manager55	Manager
9	13	manager12	Manager11	Manager
10	15	aqwena	Aqueena Regita	Staff/Kasir
11	18	chrisevans	Christopher Evans	Administrator
12	21	aqueena	aqueenaregita	Manager
13	22	qina	Aqueena Kim	Administrator
14	23	gita	Regita	Staff/Kasir
15	24	mingyu	Kim Mingyu	Administrator

2. m_kategori



**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG**

Jl. Soekarno-Hatta No. 9 Malang 65141
Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420
Laman: www.polinema.ac.id

LAPORAN DATA KATEGORI

No	Kode Kategori	Nama Kategori
1	ELC	Elektronik
2	CML	Cemilan
3	MNM	Minuman
4	FAS	Fashion
5	OTM	Otomotif
6	KSM	Kosmetik
7	SPT	Sepatu
8	SNK	Makanan Ringan
9	SBK	SEMBAKO
10	MND	Peralatan Mandi
11	BAY	Keperluan Bayi

3. m_level




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 LEVEL

No	Level Kode	Level Nama
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir

4. m_supplier



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 SUPPLIER

No	Kode Supplier	Nama Supplier	Alamat Supplier
1	SUP001	PT Sumber Jaya	Jl. Merdeka No. 123, Jakarta
2	SUP002	CV Maju Bersama	Jl. Kenanga No. 45, Surabaya
3	SUP003	UD Sejahtera	Jl. Mawar No. 7, Bandung

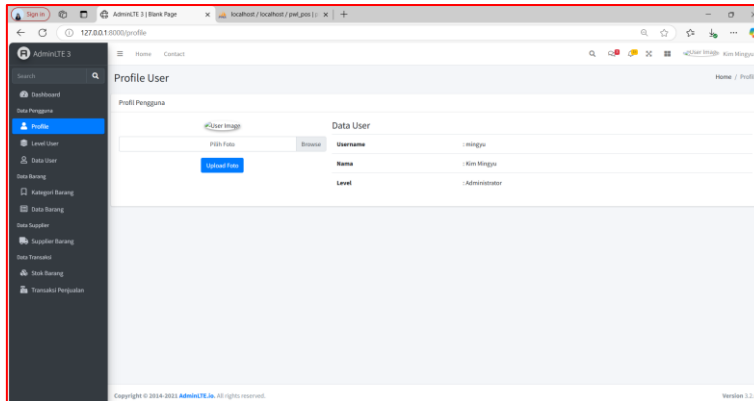
3. Submit kode untuk impementasi export pdf pada repository github kalian.

Jawaban : Sudah

Tugas 4 – Implementasi Upload File Gambar :

1. Silahkan implementasikan fitur **upload file** untuk mengubah **foto profile** di project web kalian

Jawaban : Baik



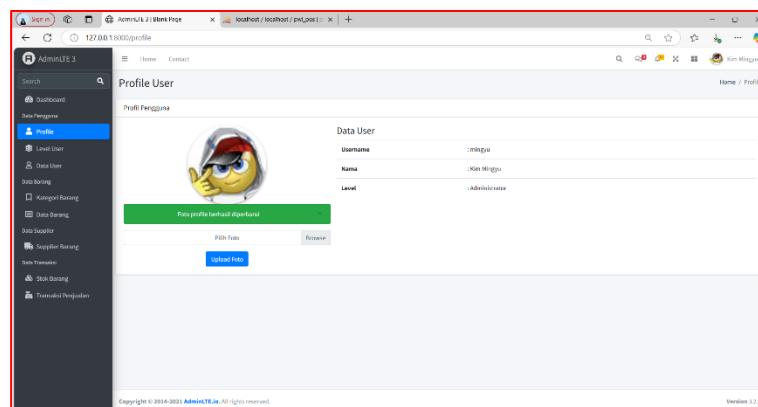
2. Jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

Jawaban : Baik

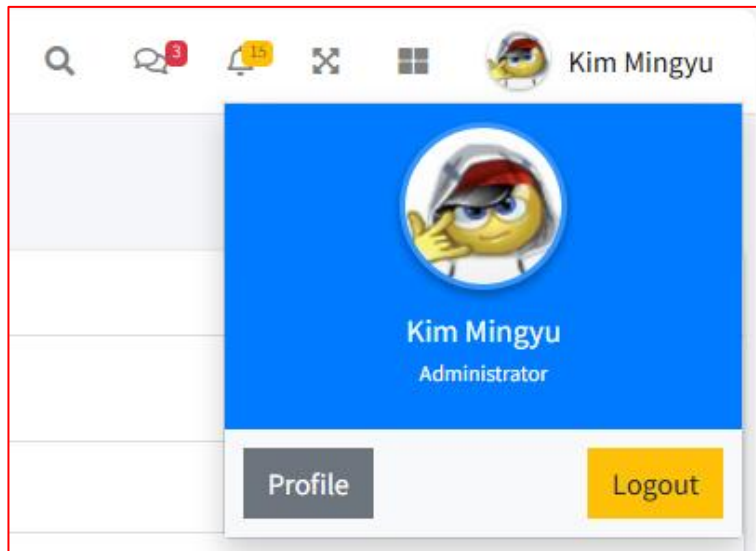
1. Pertama-tama modifikasi UserModel dengan menambahkan variable 'foto_profil' di \$fillable
2. Kemudian membuat migration untuk menambahkan Field foto_profil dengan mengisi function up() down() nya
3. Menambahkan symbolic link
4. Membuat halaman untuk Profil di views/user/profile.blade
5. Membuat method untuk UserController profilePage() dan editPhoto()
6. Selanjutnya tambahkan route pada web.php untuk fungsi foto profil ini
7. Selanjutnya modifikasi tampilan di /layouts/sidebar.blade.php untuk menambahkan element button profile
8. Selanjutnya modifikasi header.blade.php untuk menampilkan foto profil, nama, dan level nya

Hasil :

a. Halaman profile



b. Pop Up Profile



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

Jawaban : Sudah

*** *Sekian, dan selamat belajar* ***