



Nama: Yonanda Mayla Rusdiaty
Kelas: SIB 2A/30

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

Jawab:

- a. Tabel `m_kategori`

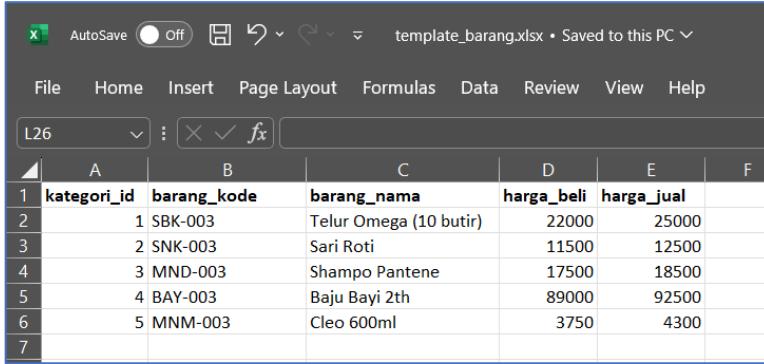
	kategori_id	kategori.kode	kategori.nama	created_at	updated_at
	1	KTG001	Elektronik	NULL	NULL
	2	KTG002	Pakaian	NULL	NULL
	3	KTG003	Makanan	NULL	NULL
	4	KTG004	Minuman	NULL	NULL
	5	KTG005	Obat-obatan	NULL	NULL
	6	KTG006	Cemilan	2025-03-16 22:48:16	NULL
	7	KTG007	Minuman Ringan	2025-03-16 22:48:16	NULL
	10	KTG008	Sepatu	NULL	NULL
	11	KTG009	Sandal	NULL	NULL



b. Tabel m_barang

barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
1	1	BRG001	Laptop ASUS ROG	15000000	17500000	NULL	NULL
2	1	BRG002	MacBook Air M2	18000000	20500000	NULL	NULL
3	1	BRG003	Monitor 27 inch IPS	3500000	4200000	NULL	NULL
4	2	BRG004	Kaos Polos Katun	50000	75000	NULL	NULL
5	2	BRG005	Jaket Denim	200000	250000	NULL	NULL
6	2	BRG006	Sepatu Sneakers	400000	550000	NULL	NULL
7	3	BRG007	Roti Tawar	15000	20000	NULL	NULL
8	3	BRG008	Biskuit Gandum	25000	35000	NULL	NULL
9	3	BRG009	Mie Instan	3500	5000	NULL	NULL
10	4	BRG010	Air Mineral 1L	5000	8000	NULL	NULL
11	4	BRG011	Jus Jeruk Botol	11000	18000	NULL	2025-03-25 11:49:10
12	4	BRG012	Kopi Bubuk 250g	30000	40000	NULL	NULL
13	5	BRG013	Paracetamol 500mg	2500	5000	NULL	NULL
14	5	BRG014	Vitamin C 1000mg	10000	15000	NULL	NULL
15	5	BRG015	Antiseptik Cair 100ml	18000	25000	NULL	NULL
18	3	BRG017	Masako Sapi	1000	2000	2025-03-25 15:03:09	2025-03-25 15:05:52

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.

Jawab:

kategori_id	barang_kode	barang_nama	harga_beli	harga_jual
4	MNR-001	Sprite 500ml	5000	6000
1	CRM-001	Lampu 20watt	45000	55000
1	KBL-001	Kabel HDMI	120000	150000
5	OBT-001	Paracetamol 500mg	5000	7000
3	MKN-002	Mie Instan	2500	3000

3. 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-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 $l)
                                    <option value="{{ $l->kategori_id }}>{{ $l->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>


```



```
        <table class="table table-bordered table-sm table-striped table-hover" id="table-barang">
            <thead>
                <tr>
                    <th>No</th>
                    <th>Kode Barang</th>
                    <th>Nama 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>
@endsession

@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, You, 2 minutes ago * Uncommitted changes
            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
                },
                {

```



```
        searchable: true
    },
    {
        data: "barang_nama",
        className: "",
        width: "37%",
        orderable: true,
        searchable: true
    },
    {
        data: "harga_beli",
        className: "",
        width: "18%",
        orderable: true,
        searchable: false,
        render: function(data, type, row) {
            return new Intl.NumberFormat('id-ID').format(data);
        }
    },
    {
        data: "harga_jual",
        className: "",
        width: "18%",
        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();
});
});

```



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 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"
            },
        ],
    });
}</script>
```



```
<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
            $.ajax({
                url: form.action,
                type: form.method,
                data: formData, // Data yang dikirim berupa FormData
                processData: false, // setting processData dan contentType ke false
                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 untuk Admin dan Manager saja (create, update, delete barang)
Route::middleware(['authorize:ADM,MNG'])->group(function () {
    // Akses tambahan untuk barang (create, update, delete)
    Route::group(['prefix' => 'barang'], function () {
        Route::get('/create', [BarangController::class, 'create']); // menampilkan halaman form tambah barang
        Route::post('/', [BarangController::class, 'store']); // menyimpan data barang baru
        Route::get('/create_ajax', [BarangController::class, 'create_ajax']); // Menampilkan halaman form tam
        Route::post('/ajax', [BarangController::class, 'store_ajax']); // Menyimpan data barang baru Ajax
        Route::get('/{id}/edit_ajax', [BarangController::class, 'edit_ajax']); // Menampilkan halaman form ed
        Route::put('/{id}/update_ajax', [BarangController::class, 'update_ajax']); // Menyimpan perubahan dat
        Route::get('/{id}/edit', [BarangController::class, 'edit']); // menampilkan halaman form edit barang
        Route::put('/{id}', [BarangController::class, 'update']); // menyimpan perubahan data barang
        Route::get('/{id}/delete_ajax', [BarangController::class, 'confirm_ajax']); // Untuk tampilan form c
        Route::delete('/{id}/delete_ajax', [BarangController::class, 'delete_ajax']); // Untuk hapus data bar
        Route::delete('/{id}', [BarangController::class, 'destroy']); // menghapus data barang
        Route::get('/import', [BarangController::class, 'import']); // ajax-form-upload-excel
        Route::post('/import_ajax', [BarangController::class, 'import_ajax']); // AJAX-import-excel
    });
});
```



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

```
composer require phoffice/phpspreadsheet
```

```
PS D:\Apps\laragon\www\PWL2025\week 8\jobsheet> composer require phoffice/phpspreadsheet
./composer.json has been updated
Running composer update phoffice/phpspreadsheet
Loading composer repositories with package information
Updating dependencies
Lock file operations: 5 installs, 0 updates, 0 removals
- Locking composer/pcre (3.3.2)
- Locking maennchen/zipstream-php (3.1.2)
- Locking markbaker/complex (3.0.2)
- Locking markbaker/matrix (3.0.1)
- Locking phoffice/phpspreadsheet (4.1.0)

88 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

[INFO] No publishable resources for tag [laravel-assets].
```

No security vulnerability advisories found.

Using version ^4.1 for phoffice/phpspreadsheet

Notes: Mengaktifkan extention gd dan zip dulu di file php.ini, baru bisa dijalankan

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

```
You, 1 second ago | author (You)
<?php
namespace App\Http\Controllers;

use App\Models\LevelModel;
use App\Models\BarangModel;
use App\Models\KategoriModel;
use Illuminate\Support\Facades\Validator;
use Phoffice\PhpSpreadsheet\IOFactory;
use Yajra\Datatables\Facades\Datatables;

1 reference (0 implementors) | You, 1 second ago | author (You)
class BarangController extends Controller
{
    /**
     * You, 1 hour ago + 10s: initial commit
     */
    // menampilkan halaman utama barang
    public function index(): factory\View
    {
        $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
        ]);
    }

    /**
     * Mengambil data barang untuk Datatables
     */
    // process (0 implementors)
    public function list(Request $request): JsonResponse
    {
        $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($barang)
            ->addColumn('aksi', function ($barang) {
                $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'])
            ->make(true);
    }
}
```



```
/**  
 * Menampilkan form tambah barang via ajax  
 */  
1 reference | 0 overrides  
public function create_ajax(): mixed|View  
{  
    $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();  
    return view('barang.create_ajax')->with('kategori', $kategori);  
}  
  
/**  
 * Menyimpan data barang baru via ajax  
 */  
1 reference | 0 overrides  
public function store_ajax(Request $request): JsonResponse|mixed\Redirector\RedirectRes...  
{  
    if ($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'  
        ]);  
    }  
  
    return redirect('/');  
}  
  
/**  
 * Menampilkan form edit barang via ajax  
 */  
1 reference | 0 overrides  
public function edit_ajax($id): Factory|View  
{  
    $barang = BarangModel::find($id);  
    $level = LevelModel::select('level_id', 'level_nama')->get();  
    return view('barang.edit_ajax', ['barang' -> $barang, 'level' -> $level]);  
}
```



```
/**  
 * Menyimpan perubahan data barang via ajax  
 */  
1 reference | 0 overrides  
public function update_ajax(Request $request, $id): JsonResponse|mixed|Redirector|RedirectRes..  
{  
    if ($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'],  
        ];  
  
        $validator = Validator::make($request->all(), $rules);  
        if ($validator->fails()) {  
            return response()->json([  
                'status' => false,  
                'message' => 'Validasi gagal.',  
                'msgField' => $validator->errors()  
            ]);  
        }  
  
        $check = BarangModel::find($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('/');  
    }  
  
    /**  
     * Menampilkan konfirmasi hapus via ajax  
     */  
1 reference | 0 overrides  
public function confirm_ajax($id): Factory|View  
{  
    $barang = BarangModel::find($id);  
    return view('barang.confirm_ajax', ['barang' => $barang]);  
}
```

```
/**  
 * Menghapus data barang via ajax  
 */  
1 reference | 0 overrides  
public function delete_ajax(Request $request, $id): JsonResponse|mixed|Redirector|RedirectRes..  
{  
    if ($request->ajax() || $request->wantsJson()) {  
        $barang = BarangModel::find($id);  
        if ($barang) {  
            $barang->delete();  
            return response()->json([  
                'status' => true,  
                'message' => 'Data berhasil dihapus'  
            ]);  
        } else {  
            return response()->json([  
                'status' => false,  
                'message' => 'Data tidak ditemukan'  
            ]);  
        }  
  
        return redirect('/');  
    }  
  
    /**  
     * Menampilkan form import barang  
     */  
1 reference | 0 overrides  
public function import(): Factory|View  
{  
    return view('barang.import');  
}
```



```
/**  
 * Mengimport data barang dari file excel  
 */  
1 reference | 0 overrides  
public function import_ajax(Request $request): JsonResponse|mixed|Redirector|RedirectRes...  
{  
    if ($request->ajax() || $request->wantsJson()) {  
        $rules = [  
            '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');  
        $reader = IOFactory::createReader('Xlsx');  
        $reader->setReadDataOnly(true);  
        $spreadsheet = $reader->load($file->getRealPath());  
        $sheet = $spreadsheet->getActiveSheet();  
        $data = $sheet->toArray(null, false, true, true);  
  
        $insert = [];  
        if (count(value: $data) > 1) {  
            foreach ($data as $baris => $value) {  
                if ($baris > 1) { // baris ke 1 adalah header  
                    $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(value: $insert) > 0) {  
                    BarangModel::insertOrIgnore($insert);  
                    return response()->json([  
                        'status' => true,  
                        'message' => 'Data berhasil diimport'  
                    ]);  
                }  
            }  
            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

Jawab:

Import Data Barang ×

Download Template [Download](#)

Pilih File

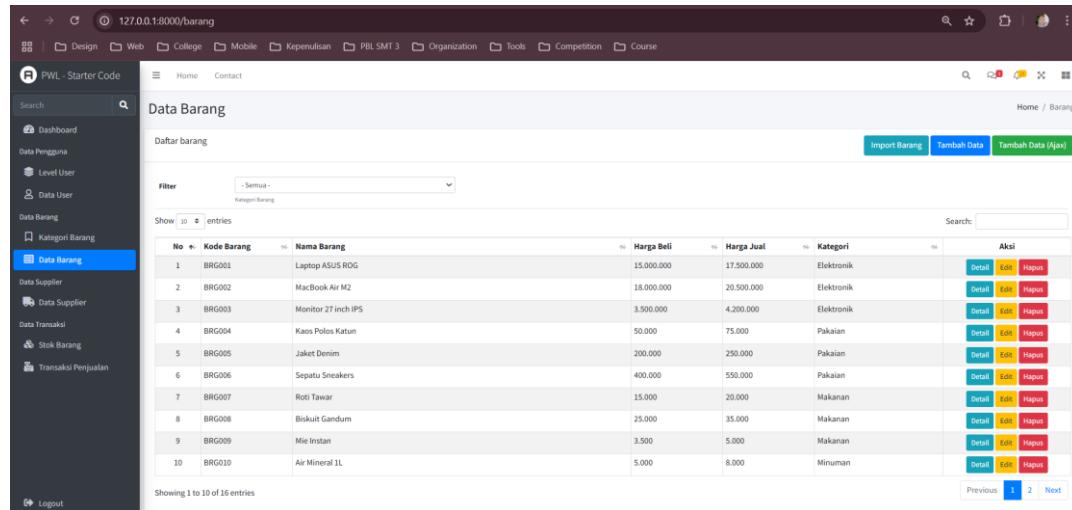
No file selected.

Upload Batal



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

Jawab:



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Laptop ASUS ROG	15.000.000	17.500.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
2	BRG002	MacBook Air M2	18.000.000	20.500.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
3	BRG003	Monitor 27 inch IPS	3.500.000	4.200.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
4	BRG004	Kaos Polos Katun	50.000	75.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
5	BRG005	Jaket Denim	200.000	250.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
6	BRG006	Sepatu Sneakers	400.000	550.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
7	BRG007	Roti Tawar	15.000	20.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
8	BRG008	Biskuit Gandum	25.000	35.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
9	BRG009	Mie Instan	3.500	5.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
10	BRG010	Air Mineral 1L	5.000	8.000	Minuman	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Import Data Barang

Download Template [Download](#)

Pilih File

Choose File No file chosen

Batal Upload

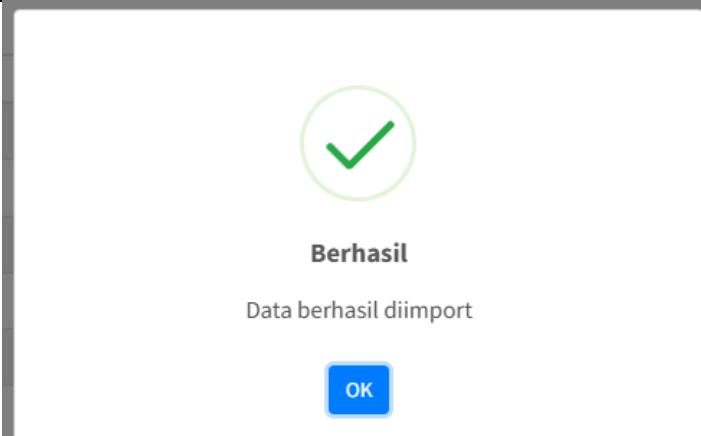
Import Data Barang

Download Template [Download](#)

Pilih File

Choose File template_barang.xlsx

Batal Upload



Secara otomatis di halaman data barang akan menampilkan data yang diimport

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
11	BRG011	Jus Jeruk Botol	12.000	18.000	Minuman	[Detail] [Edit] [Hapus]
12	BRG012	Kopi Bubuk 250g	30.000	40.000	Minuman	[Detail] [Edit] [Hapus]
13	BRG013	Paracetamol 500mg	2.500	5.000	Obat-obatan	[Detail] [Edit] [Hapus]
14	BRG014	Vitamin C 1000mg	10.000	15.000	Obat-obatan	[Detail] [Edit] [Hapus]
15	BRG015	Antiseptik Cair 100ml	18.000	25.000	Obat-obatan	[Detail] [Edit] [Hapus]
16	MNR-001	Sprite 500ml	5.000	6.000	Minuman	[Detail] [Edit] [Hapus]
17	CRM-001	Lampu 20watt	45.000	55.000	Elektronik	[Detail] [Edit] [Hapus]
18	KBL-001	Kabel HDMI	120.000	150.000	Elektronik	[Detail] [Edit] [Hapus]
19	OBT-001	Paracetamol 500mg	5.000	7.000	Obat-obatan	[Detail] [Edit] [Hapus]
20	MKN-002	Mie Instan	2.500	3.000	Makanan	[Detail] [Edit] [Hapus]

Showing 11 to 20 of 20 entries

Previous 1 2 Next

- Proses yg terjadi setelah import berhasil yaitu:
 - Data dari file Excel dikonversi menjadi array PHP. Data tersebut dimasukkan ke database pada tabel m_barang melalui model BarangModel
 - JavaScript di halaman menerima respons sukses dari server. Notifikasi sukses ditampilkan menggunakan SweetAlert dengan pesan "Data berhasil diimport"
 - Jika ada kode barang yang sudah ada (duplikat), data tersebut akan dilewati berkat penggunaan method insertOrIgnore()

Tugas 1 – Implementasi File Upload untuk Import Data:

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

Jawab:

- Data User
 - a. Membuat template_user.xls



user_id	username	nama	level_id
9	manager99	Andi Pratama	2
10	staff123	Budi Santoso	3
11	customer123	Rina Wiava	4

b. Modifikasi user/index.blade

```
10:23 minute ago | author(you)
@extends('layouts.template')

@section('content')


### Daftar Pengguna



<button onclick="modalAction('{{ url('/user/import') }}'" class="btn btn-info">Import Pengguna</button>



<label for="filter_level" class="col-md-1 col-form-label">Filter</label>


<select name="filter_level" class="form-control form-control-sm filter_level">
<option value=""> Semua </option>
<option value="1">Admin</option>
<option value="2">Manager</option> <!-- Added for level_id 2 -->
<option value="3">Staff</option> <!-- Added for level_id 3 -->
<option value="4">Customer</option>
</select>
Level Pengguna


@if(session('success'))
<div class="alert alert-success">{{ session('success') }}</div>
@endif

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



| No | ID Pengguna | Username | Nama | Level | Aksi |
|----|-------------|----------|------|-------|------|
|----|-------------|----------|------|-------|------|


```



```
--- Modal For Import Form ---


</div>
@endsection

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

    var tableUser;
    $(document).ready(function() {
        tableUser = $('#table-user').DataTable({
            processing: true,
            serverSide: true,
            ajax: {
                "url": "{{ url('user/list') }}",
                "dataType": "json",
                "type": "POST",
                "data": function (d) {
                    d.filter_level = $(".filter_level").val();
                }
            },
            columns: [
                {
                    data: "DT_RowIndex",
                    className: "text-center",
                    width: "5%",
                    orderable: false,
                    searchable: false
                },
                {
                    data: "user_id",
                    className: "",
                    width: "15%",
                    orderable: true,
                    searchable: true
                },
                {
                    data: "username",
                    className: "",
                    width: "25%",
                    orderable: true,
                    searchable: true
                },
                {
                    data: "nama",
                    className: "",
                    width: "25%",
                    orderable: true,
                    searchable: true
                },
                {
                    data: "level_id",
                    className: "",
                    width: "15%",
                    orderable: true,
                    searchable: false,
                    render: function(data) {
                        return levels[data] || 'Unknown';
                    }
                }
            ]
        });
    });

    $('#table-user_filter input').unbind().bind().on('keyup', function(e) {
        if (e.keyCode == 13) { // Enter key
            tableUser.search(this.value).draw();
        }
    });

    $('.filter_level').change(function() {
        tableUser.draw();
    });
});
</script>
@endpush


```

```
    const levels = {
        1: 'Admin',
        2: 'Manager',
        3: 'Staff',
        4: 'Customer'
    };
    return levels[data] || 'Unknown';
}

{
    data: "aksi",
    className: "text-center",
    width: "15%",
    orderable: false,
    searchable: false
}
]
});

$('#table-user_filter input').unbind().bind().on('keyup', function(e) {
    if (e.keyCode == 13) { // Enter key
        tableUser.search(this.value).draw();
    }
});

$('.filter_level').change(function() {
    tableUser.draw();
});
});
```



c. Membuat user/import.blade

```
<Form action="{{ url('/user/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 Pengguna</h5>
      <button type="button" class="close" data-dismiss="modal" aria-label="Close">
        <span aria-hidden="true">&amptimes</span>
      </button>
    </div>
    <div class="modal-body">
      <div class="form-group">
        <label>Download Template</label>
        <a href="{{ asset('template_user.xlsx') }}" class="btn btn-info btn-sm" download>
          <i class="fa fa-file-excel"></i> Download
        </a>
        <small id="error-template" class="error-text form-text text-danger"></small>
      </div>
      <div class="form-group">
        <label>Pilih File</label>
        <input type="file" name="file_user" id="file_user" class="form-control" required>
        <small id="error-file_user" 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_user: {
        required: true,
        extension: "xlsx"
      },
      submitHandler: function(form) {
        var formData = new FormData(form); // Convert form to FormData to handle file upload

        $.ajax({
          url: form.action,
          type: form.method,
          data: formData,
          processData: false, // Prevent jquery from processing the data (needed for file upload)
          contentType: false, // Prevent jquery from setting content type (needed for file upload)
          success: function(response) {
            if (response.status) { // If successful
              $('#myModal').modal('hide');
              Swal.fire({
                icon: 'success',
                title: 'Berhasil',
                text: response.message
              });
              tableUser.ajax.reload(); // Reload the datatable (assumes tableUser is from user/index.blade.php)
            } else { // If there's an 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
              });
            }
          },
          error: function(xhr, status, error) {
            Swal.fire({
              icon: 'error',
              title: 'Kesalahan Server',
              text: 'Terjadi kesalahan saat mengunggah file. Silakan coba lagi.'
            })
          }
        });
        return false; // Prevent default form submission
      },
      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>
```



- d. Modifikasi route/web.php untuk mengakomodir proses upload file pada menu user

```
> Route::middleware(['auth:ADM'])->group(function () { void {
    // Tambahkan route grup untuk user yg memerlukan autentifikasi disini
    Route::group(['prefix' => 'user'], function () { void {
        Route::get('/', [UserController::class, 'index']); // menampilkan halaman awal user
        Route::post('/list', [UserController::class, 'list']); // menampilkan data user dalam bentuk json untuk datatable
        Route::get('/create', [UserController::class, 'create']); // menampilkan halaman form tambah user
        Route::get('/create_ajax', [UserController::class, 'create_ajax']); // menampilkan halaman form tambah user dengan ajax
        Route::post('/ajax', [UserController::class, 'store_ajax']); // menyimpan data user baru dengan ajax
        Route::post('/', [UserController::class, 'store']); // menyimpan data user baru
        Route::get('/{id}', [UserController::class, 'show']); // menampilkan detail user
        Route::get('/{id}/edit', [UserController::class, 'edit']); // menampilkan halaman form edit user
        Route::get('/{id}/edit_ajax', [UserController::class, 'edit_ajax']); // menampilkan halaman form edit user dengan ajax
        Route::put('/{id}/update_ajax', [UserController::class, 'update_ajax']); // menyimpan perubahan data user
        Route::get('/{id}/delete_ajax', [UserController::class, 'confirm_ajax']); // menampilkan konfirmasi hapus user dengan ajax
        Route::delete('/{id}', [UserController::class, 'delete_ajax']); // menghapus data user AJAX
        Route::put('/{id}', [UserController::class, 'update']); // menyimpan perubahan data user
        Route::delete('/{id}', [UserController::class, 'destroy']); // menghapus data user
    }}); void {
        ...Route::get('/import', [UserController::class, 'import']); // .ajax-form-upload-excel
        ...Route::post('/import_ajax', [UserController::class, 'import_ajax']); // .AJAX-import-excel
    });
}); void {
```

- e. Tambahkan function import dan import_ajax pada UserController

```
1 reference | 0 overrides
public function import(): Factory|View
{
    return view('user.import');
}

1 reference | 0 overrides
public function import_ajax(Request $request): JsonResponse|mixed\Redirector\RedirectResponse
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'file_user' => ['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_user');
        $reader = IOFactory::createReader('Xlsx');
        $reader->setReadDataOnly(true);
        $spreadsheet = $reader->load($file->getRealPath());
        $sheet = $spreadsheet->getActiveSheet();
        $data = $sheet->toArray(null, false, true, true);
        You, 14 minutes ago * Uncommitted changes
        $insert = [];
        if (count($data) > 1) {
            foreach ($data as $row => $value) {
                if ($row > 1) { // Skip header row
                    $insert[] = [
                        'user_id' => $value['A'],
                        'username' => $value['B'],
                        'nama' => $value['C'],
                        'level_id' => $value['D'],
                        'password' => bcrypt('password'), // Default password, adjust as needed
                        'created_at' => now(),
                    ];
                }
            }
            if (count($value: $insert) > 0) {
                UserModel::insertOrIgnore($insert);
                return response()->json([
                    'status' => true,
                    'message' => 'Data berhasil diimport'
                ]);
            }
        }
        return response()->json([
            'status' => false,
            'message' => 'Tidak ada data yang diimport'
        ]);
    }
    return redirect('/');
}
```



f. Tampilannya

The screenshot shows the 'Daftar Pengguna' (User List) page. On the left, there's a sidebar with navigation links: Dashboard, Data Pengguna, Level User, and Data User (which is currently selected). The main area displays a table with columns: No, ID Pengguna, Username, Name, Level, and Aksi (Actions). The table contains 6 entries. At the top right of the table are three buttons: Import Pengguna, Tambah Pengguna, and Tambah Pengguna [Ajax]. Below the table, there's a search bar and a message indicating 1 to 6 of 6 entries.

Import Data Pengguna

Download Template [Download](#)

Pilih File

Choose File template_user.xlsx

Batal Upload

Berhasil

Data berhasil diimport

OK

The 'Berhasil' (Successful) message indicates that data has been imported successfully. The 'OK' button allows the user to close the modal.

Daftar Pengguna

Showing 1 to 9 of 9 entries

No	ID Pengguna	Username	Nama	Level	Aksi
1	1	admin	Administrator	Admin	[Detail, Edit, Hapus]
2	2	manager	Manager	Manager	[Detail, Edit, Hapus]
3	3	staff	Staff/Kasir	Staff	[Detail, Edit, Hapus]
4	4	customer-1	Customer 1	Customer	[Detail, Edit, Hapus]
5	7	customer-2	Vonanda Mayla Rusdity	Customer	[Detail, Edit, Hapus]
6	8	customer-3	Adibah Hansa	Customer	[Detail, Edit, Hapus]
7	9	manager99	Andi Pratama	Manager	[Detail, Edit, Hapus]
8	10	staff123	Budi Santoso	Staff	[Detail, Edit, Hapus]
9	11	customer123	Rina Wijaya	Customer	[Detail, Edit, Hapus]



➤ Data Supplier

a. Membuat template_supplier.xls

supplier_kode	supplier_nama	supplier_alamat	supplier_telp	supplier_email	supplier_kontak
SUP006	PT. Teknologi Maju	Jl. Inovasi No. 10, Surabaya	081-23456789	info@teknologimaju.co.id	Andi Pratama
SUP007	CV. Sumber Makmur	Jl. Raya No. 88, Yogyakarta	082-34567890	sales@sumbermakmur.co.id	Rina Susanti
SUP008	UD. Jaya Abadi	Jl. Merdeka No. 15, Medan	085-45678901	order@jadiyabadi.com	Budi Hartono
SUP009	PT. Cahaya Sentosa	Jl. Industri Baru No. 22, Bekasi	081-56789012	cs@cahayasentosa.co.id	Siti Aminah
SUP010	CV. Prima	Jl. Pahlawan No. 33, Semarang	087-67890123	info@primasejahtera.co.id	Dedi Kurniawan

b. Modifikasi supplier/index.blade

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

@section('content')


### Daftar Pengguna



<-->
Import Supplier
Existing Buttons
Tambah Data
Tambah Pengguna \(Ajax\)



Filter



Level Pengguna



You: 35 minutes ago • Uncommitted changes


@if(session('success'))
  <div class="alert alert-success">{{ session('success') }}</div>
@endif

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



| No | ID Pengguna | Username | Nama | Level | Aksi |
|----|-------------|----------|------|-------|------|
|----|-------------|----------|------|-------|------|


```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
... Modal for Import Form ...


</div>
@endsection

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

var tableUser;
$(document).ready(function() {
    tableUser = $('#table-user').DataTable({
        processing: true,
        serverSide: true,
        ajax: {
            url: "{{ url('user/list') }}",
            "datatype": "json",
            "type": "POST",
            "data": function (d) {
                d._token = "{{ csrf_token() }}"; // Tambahkan CSRF token
                d.filter_level = $('#filter_level').val();
            }
        },
        columns: [
            {
                data: 'DT_RowIndex',
                className: "text-center",
                width: "5%",
                orderable: false,
                searchable: false
            },
            {
                data: "user_id",
                className: "",
                width: "15%",
                orderable: true,
                searchable: true
            },
            {
                data: "username",
                className: "",
                width: "25%",
                orderable: true,
                searchable: true
            },
            {
                data: "name",
                className: "",
                width: "25%",
                orderable: true,
                searchable: true
            },
            {
                data: "level_id",
                className: "",
                width: "15%",
                orderable: true,
                searchable: false,
                render: function(data) {
                    const levels = {
                        1: 'Admin',
                        2: 'Manager',
                        3: 'Staff',
                        4: 'Customer'
                    };
                    return levels[data] || 'Unknown';
                }
            },
            {
                data: "aksi",
                className: "text-center",
                width: "15%",
                orderable: false,
                searchable: false
            }
        ]
    });

    $('#table-user_filter input').unbind().bind().on('keyup', function(e) {
        if (e.keyCode == 13) { // Enter key
            tableUser.search(this.value).draw();
        }
    });

    $('#filter_level').change(function() {
        tableUser.draw();
    });
});
</script>
@endpush


```

```
$(document).ready(function() {
    searchable: true
},
{
    data: "name",
    className: "",
    width: "25%",
    orderable: true,
    searchable: true
},
{
    data: "level_id",
    className: "",
    width: "15%",
    orderable: true,
    searchable: false,
    render: function(data) {
        const levels = {
            1: 'Admin',
            2: 'Manager',
            3: 'Staff',
            4: 'Customer'
        };
        return levels[data] || 'Unknown';
    }
},
{
    data: "aksi",
    className: "text-center",
    width: "15%",
    orderable: false,
    searchable: false
}
);

$('#table-user_filter input').unbind().bind().on('keyup', function(e) {
    if (e.keyCode == 13) { // Enter key
        tableUser.search(this.value).draw();
    }
});

$('#filter_level').change(function() {
    tableUser.draw();
});
});
</script>
@endpush
```



c. Membuat supplier/import.blade

```
<form action="{{ url('/supplier/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 Supplier</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close"></span>
            </button>
        </div>
        <div class="modal-body">
            <div class="form-group">
                <label>Download Template</label>
                <a href="{{ asset('template_supplier.xlsx') }}" class="btn btn-info btn-sm" download>
                    <i class="fa fa-file-excel"></i> Download
                </a>
                <small id="error-template" class="error-text form-text text-danger"></small>
            </div>
            <div class="form-group">
                <label>Pilih file:</label>
                <input type="file" name="file_supplier" id="file_supplier" class="form-control" required>
                <small id="error_file_supplier" 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_supplier: {
                required: true,
                extension: "xlsx|xls"
            },
        },
        submitHandler: function(form) {
            var formData = new FormData(form);

            $.ajax({
                url: form.action,
                type: form.method,
                data: formData,
                processData: false,
                contentType: false,
                success: function(response) {
                    if (response.status) {
                        $('#myModal').modal('hide');
                        Swal.fire({
                            icon: 'success',
                            title: 'Berhasil',
                            text: response.message
                        });
                        window.dataSupplier.ajax.reload(); // Reload the DataTable
                    } else {
                        $('.error-text').text('');
                        $.each(response.msgField, function(prefix, val) {
                            $('#error-' + prefix).text(val[0]);
                        });
                        Swal.fire({
                            icon: 'error',
                            title: 'Terjadi Kesalahan',
                            text: response.message
                        });
                    }
                }
            });
        }
    });
});
</script>
$(document).ready(function() {
    submitHandler: function(form) {
        error: function(xhr, status, error) {
            console.log('AJAX Error:', xhr.responseText);
            Swal.fire({
                icon: 'error',
                title: 'Kesalahan Server',
                text: 'Terjadi kesalahan saat mengunggah file: ' + (xhr.responseJSON?.message || 'Silakan coba lagi.')
            });
        }
    };
    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>
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI MALANG

JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

- d. Modifikasi route/web.php untuk mengakomodir proses upload file pada menu supplier

```
Route::middleware(['authorize:ADM,MNG'])->group(function (): void {
    // Tambahkan route grup untuk supplier
    Route::group(['prefix' => 'supplier'], function (): void {
        Route::get('/', [SupplierController::class, 'index']); // Menampilkan halaman tabel supplier
        Route::post('/list', [SupplierController::class, 'list']); // Mengambil data supplier untuk DataTables
        Route::get('/create', [SupplierController::class, 'create']); // Menampilkan form tambah supplier
        Route::get('/create_ajax', [SupplierController::class, 'create_ajax']); // Menampilkan form tambah supplier ajax
        Route::post('/store_ajax', [SupplierController::class, 'store_ajax']); // Menyimpan data supplier baru ajax
        Route::post('/', [SupplierController::class, 'store']); // Menyimpan data supplier baru
        Route::get('/{id}', [SupplierController::class, 'show']); // Menampilkan detail supplier
        Route::get('/{id}/show_ajax', [SupplierController::class, 'show_ajax']); // Menampilkan detail supplier ajax
        Route::get('/{id}/edit', [SupplierController::class, 'edit']); // Menampilkan form edit supplier
        Route::get('/{id}/edit_ajax', [SupplierController::class, 'edit_ajax']); // Menampilkan form edit supplier ajax
        Route::put('/{id}/update_ajax', [SupplierController::class, 'update_ajax']); // Menyimpan perubahan supplier ajax
        Route::get('/{id}/delete_ajax', [SupplierController::class, 'confirm_ajax']); // Menampilkan konfirmasi hapus supplier ajax
        Route::delete('/{id}', [SupplierController::class, 'delete_ajax']); // Menghapus supplier ajax
        Route::put('/{id}', [SupplierController::class, 'update']); // Menyimpan perubahan supplier
        Route::get('/{id}/show_ajax', [SupplierController::class, 'show_ajax']); // Menampilkan detail supplier ajax
        Route::delete('/{id}', [SupplierController::class, 'destroy']); // Menghapus supplier
        Route::get('/import', [SupplierController::class, 'import']); // Loads the import form
        Route::post('/import_ajax', [SupplierController::class, 'import_ajax']); // Handles the file upload
    });
});
```

- e. Tambahkan function import dan import_ajax pada SupplierController

```
1 reference | 0 overrides
public function import(): Factory|View
{
    return view('supplier.import');
}

1 reference | 0 overrides
public function import_ajax(Request $request): JsonResponse|mixed|Redirector|RedirectResponse
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'file_supplier' => ['required', 'mimes:xlsx,xls', 'max:1024']
        ];
        $validator = Validator::make($request->all(), $rules);
        if ($validator->fails()) {
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors()
            ]);
        }
    }

    try {
        $file = $request->file('file_supplier');
        $reader = IOFactory::createReader('Xlsx');
        $reader->setReadDataOnly(true);
        $spreadsheet = $reader->load($file->getRealPath());
        $sheet = $spreadsheet->getActiveSheet();
        $data = $sheet->toArray(null, false, true, true);

        $insert = [];
        if (count($data) > 1) {
            foreach ($data as $row => $value) {
                if ($row > 1) { // Skip header row
                    $insert[] = [
                        'supplier_kode' => $value['A'],
                        'supplier_nama' => $value['B'],
                        'supplier_alamat' => $value['C'],
                        'supplier_telp' => $value['D'],
                        'created_at' => now(),
                    ];
                }
            }
            if (count($insert) > 0) {
                SupplierModel::insertOrIgnore($insert);
                return response()->json([
                    'status' => true,
                    'message' => 'Data berhasil diimport'
                ]);
            }
        }
        return response()->json([
            'status' => false,
            'message' => 'Tidak ada data yang diimport'
        ]);
    } catch (\Exception $e) {
        Log::error('Supplier Import Error: ' . $e->getMessage());
        return response()->json([
            'status' => false,
            'message' => 'Gagal mengunggah file: ' . $e->getMessage()
        ], 500);
    }
    return redirect('/');
}
```



f. Tampilannya

The screenshot shows the 'Daftar Pengguna' page with a table of user data. A modal window titled 'Import Data Pengguna' is open, containing a 'Download Template' button and a 'Choose File' input field with 'template_supplier.xlsx' selected. Below the modal is a success message: 'Berhasil' with a checkmark icon, followed by 'Data berhasil diimport' and an 'OK' button.

Daftar Pengguna

No	ID Pengguna	Username	Nama	Level	Aksi
1	1	admin	Administrator	Admin	[Detail, Edit, Hapus]
2	2	manager	Manager	Manager	[Detail, Edit, Hapus]
3	3	staff	Staf/Kasir	Staff	[Detail, Edit, Hapus]
4	4	customer-1	Customer 1	Customer	[Detail, Edit, Hapus]
5	7	customer-2	Yonanda Mayfa Ruspithy	Customer	[Detail, Edit, Hapus]

Import Data Pengguna

Download Template [Download](#)

Pilih File

Choose File template_supplier.xlsx

Batal Upload

Berhasil

Data berhasil diimport

OK

Daftar Supplier

No	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	PT. Elektronik Jaya	Jl. Industri No.123, Jakarta	[Detail, Edit, Hapus]
2	SUP002	CV. Fashion Indonesia	Jl. Teknologi No. 45, Bandung	[Detail, Edit, Hapus]
3	SUP003	UD. Makalan Sehat	Jl. Pasar Baru No. 78, Semarang	[Detail, Edit, Hapus]
4	SUP004	PT. Mewarna Prima	Jl. Industri Minuman Blok A5, Tangerang	[Detail, Edit, Hapus]
5	SUP005	PT. Farmasi Indonesia	Jl. Farmasi No. 98, Semarang	[Detail, Edit, Hapus]
6	SUP006	PT. Teknologi Nasional	Jl. Merdeka No. 88, Yogyakarta	[Detail, Edit, Hapus]
7	SUP007	CV. Sumber Makmur	Jl. Raya Bogor No. 15, Bogor	[Detail, Edit, Hapus]
8	SUP008	PT. Jaya Abadi	Jl. Sudirman No. 99, Medan	[Detail, Edit, Hapus]
9	SUP009	UD. Cahaya Baru	Jl. Diponegoro No. 22, Denpasar	[Detail, Edit, Hapus]
10	SUP010	PT. Sentosa Raya	Jl. Gant Saloko No. 77, Makassar	[Detail, Edit, Hapus]

➤ Level User

a. Membuat template_level.xls

level_kod	level_na
e	ma
GUE	Guest
SUP	Supervis
TRN	Trainee



b. Modifikasi level/index.blade

```
18 seconds ago | Author (You)
@extends('layouts.template')

@section('content')
    <div class="card card-outline card-primary">
        <div class="card-header">
            Import Data Level
        </div>
        <div class="card-tools">
            This view button for Importing levels ...
            <button onclick="modalAction('{{ url('/level/import') }}')" class="btn btn-info btn-sm mt-1">Import Level</button>
            <a class="btn btn-sm btn-primary mt-1" href="{{ url('/level/create') }}">Tambah</a>
            <button onclick="modalAction('{{ url('/level/create_ajax') }}')" class="btn btn-sm btn-success mt-1">Tambah Ajax</button>
        </div>
    </div>
    <div class="card-body">
        <!-- Filter Section (Dynamic Level Code) -->
        <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_kode" class="col-md-1 col-form-label">Filter</label>
                        <div class="col-md-3">
                            <select name="filter_kode" class="form-control form-control-sm filter_kode">
                                <option value=""> Semua --</option>
                                @foreach($levels as $l)
                                    <option value="{{ $l->level.kode }}>{{ $l->level.kode }}</option>
                                @endforeach
                            </select>
                            <small class="form-text text-muted">Kode Level</small>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

c. Membuat level/import.blade

```
<Form action="{{ url('/level/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 Level</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&amptimes</span>
                </button>
            </div>
            <div class="modal-body">
                <div class="form-group">
                    <label>Download Template</label>
                    <a href="{{ asset('template_level.xlsx') }}" class="btn btn-info btn-sm" download>
                        <i class="fa fa-file-excel"></i> Download
                    </a>
                    <small id="error-template" class="error-text form-text text-danger"></small>
                </div>
                <div class="form-group">
                    <label>Pilih File</label>
                    <input type="file" name="file_level" id="file_level" class="form-control" required>
                    <small id="error-file_level" 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_level: {
                    required: true,
                    extension: "xlsx|xls"
                },
                submitHandler: function (form) {
                    var formData = new FormData(form);

                    $.ajax({
                        url: form.action,
                        type: form.method,
                        data: formData,
                        processData: false,
                        success: function (response) {
                            if (response.error) {
                                $.each(response.error, function (key, value) {
                                    $('#error-' + key).text(value);
                                });
                            } else {
                                alert("Data berhasil diimport!");
                                window.location.href = response.url;
                            }
                        }
                    });
                }
            }
        });
    });
</script>
```




f. Tampilan

Daftar Level

ID	Kode	Nama	Aksi
1	ADM	Administrator	[Detail] [Edit] [Hapus]
2	MNG	Manager	[Detail] [Edit] [Hapus]
3	STF	Staf/Kasir	[Detail] [Edit] [Hapus]
4	CUS	Customer	[Detail] [Edit] [Hapus]

Showing 1 to 4 of 4 entries.

Import Data Level

Download Template [Download](#)

Pilih File

Choose File template_level.xlsx

Batal Upload

Berhasil

Data level berhasil diimport

OK

Daftar Level

ID	Kode	Nama	Aksi
1	ADM	Administrator	[Detail] [Edit] [Hapus]
2	MNG	Manager	[Detail] [Edit] [Hapus]
3	STF	Staf/Kasir	[Detail] [Edit] [Hapus]
4	CUS	Customer	[Detail] [Edit] [Hapus]
5	GUE	Guest	[Detail] [Edit] [Hapus]
6	SUP	Supervisor	[Detail] [Edit] [Hapus]
7	TRN	Trainee	[Detail] [Edit] [Hapus]

Showing 1 to 7 of 7 entries



➤ Kategori Barang

- a. Membuat template_kategori.xls

kategori_	kategori_
kode	nama
KTG006	Aksesoris
	Peralatan
KTG007	Rumah
	Tangga
KTG008	Alat Tulis

- b. Modifikasi kategori/index.blade

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

@section('content')
    <div class="card card-outline card-primary">
        <div class="card-header">
            <h3 class="card-title">{{ $page->title }}</h3>
            <div class="card-tools">
                <button onclick="modalAction('{{ url('/kategori/import') }}'" class="btn btn-info">Import Kategori</button>
                <a href="{{ url('/kategori/create') }}" class="btn btn-primary">Tambah Data</a>
                <button onclick="modalAction('{{ url('/kategori/create_ajax') }}'" class="btn btn-success">Tambah Data (Ajax)</button>
            </div>
        </div>
        <div class="card-body">
            <!-- Filter Section --> You, 19 minutes ago * Uncommitted changes
            <div id="filter" class="form-horizontal filter-date p-2 border-bottom mb-2">
                <div class="row">
                    <div class="col-md-12">
                        <div class="form-group form-group-sm row text-sm mb-0">
                            <label for="filter_kategori" class="col-md-1 col-form-label">Filter</label>
                            <div class="col-md-3">
                                <select name="filter_kategori" class="form-control form-control-sm filter_kategori">
                                    <option value=""> Semua -</option>
                                    @foreach($kategoris as $kategori)
                                        <option value="{{ $kategori->kategori_nama }}>{{ $kategori->kategori_nama }}</option>
                                    @endforeach
                                </select>
                                <small class="form-text text-muted">Nama Kategori</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>

```

- c. Membuat kategori/import.blade



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI MALANG

JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
<form action="{{ url('kategori/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 Kategori</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&amptimes</span>
                </button>
            </div>
            <div class="modal-body">
                <div class="form-group">
                    <label>Download Template</label>
                    <a href="{{ asset('template_kategori.xlsx') }}" class="btn btn-info btn-sm" download>
                        <i class="fa fa-file-excel"></i> Download
                    </a>
                    <small id="error-template" class="error-text form-text text-danger"></small>
                </div>
                <div class="form-group">
                    <label>Pilih File</label>
                    <input type="file" name="file_kategori" id="file_kategori" class="form-control" required>
                    <small id="error-file_kategori" 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_kategori: {
                required: true,
                extension: "xlsx|xls"
            },
            submitHandler: function (form) {
                var formData = new FormData(form);

                $.ajax({
                    url: form.action,
                    type: form.method,
                    data: formData,
                    processData: false,
                    contentType: false,
                    success: function (response) {
                        if (response.status) {
                            $('#myModal').modal('hide');

                            Swal.fire({
                                icon: 'success',
                                title: 'Berhasil',
                                text: response.message
                            });
                            if (window.tableKategori) {
                                window.tableKategori.ajax.reload();
                            }
                            else if (window.parent && window.parent.tableKategori) {
                                window.parent.tableKategori.ajax.reload();
                            }
                            else {
                                $('#table_kategori').DataTable().ajax.reload();
                            }
                        } else {
                            Swal.fire({
                                icon: 'error',
                                title: 'Terjadi Kesalahan',
                                text: response.message
                            });
                            $('.error-text').text('');
                            $.each(response.msgField, function (prefix, val) {
                                $('#error-' + prefix).text(val[0]);
                            });
                            Swal.fire({
                                icon: 'error',
                                title: 'Terjadi Kesalahan',
                                text: response.message
                            });
                        }
                    },
                    error: function (xhr, status, error) {
                        console.log('AJAX Error:', xhr.responseText);
                        Swal.fire({
                            icon: 'error',
                            title: 'Kesalahan Server',
                            text: 'Terjadi kesalahan saat mengunggah file: ' + (xhr.responseJSON).message || 'Silakan coba lagi.'
                        });
                    }
                });
                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>
```



- d. Modifikasi route/web.php untuk mengakomodir proses upload file pada menu kategori

```
Route::middleware(['authorize:ADM,MNG'])->group(function () {
    // Tambahkan route grup untuk kategori
    Route::group(['prefix' => 'kategori'], function () {
        Route::get('/', [KategoriController::class, 'index']); // Menampilkan halaman tabel kategori
        Route::post('/list', [KategoriController::class, 'list']); // Mengambil data kategori untuk DataTables
        Route::get('/create', [KategoriController::class, 'create']); // Menampilkan form tambah kategori
        Route::post('/create_ajax', [KategoriController::class, 'create_ajax']); // Menampilkan form tambah kategori ajax
        Route::post('/ajax', [KategoriController::class, 'store_ajax']); // Menyimpan data kategori baru ajax
        Route::post('/', [KategoriController::class, 'store']); // Menyimpan data kategori baru
        Route::get('/{id}', [KategoriController::class, 'show']); // Menampilkan detail kategori
        Route::get('/{id}/edit', [KategoriController::class, 'edit']); // Menampilkan form edit kategori
        Route::get('/{id}/edit_ajax', [KategoriController::class, 'edit_ajax']); // Menampilkan form edit kategori ajax
        Route::put('/{id}/update_ajax', [KategoriController::class, 'update_ajax']); // Menyimpan perubahan kategori ajax
        Route::get('/{id}/delete_ajax', [KategoriController::class, 'confirm_ajax']); // Menampilkan konfirmasi hapus kategori ajax
        Route::delete('/{id}/delete_ajax', [KategoriController::class, 'delete_ajax']); // Menghapus kategori ajax
    });
});
```

- e. Tambahkan function import dan import_ajax pada KategoriController

```
/*
 * Menampilkan form import kategori
 */
public function import(): FactoryView
{
    return view('kategori.import');
}

/**
 * Import kategori dari Excel file
 */
public function import_ajax(Request $request): JsonResponse|Redirect\RedirectResponse
{
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'file_kategori' => ['required', 'mimes:xlsx,xls', 'max:1024']
        ];

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

        try {
            $file = $request->file('file_kategori');
            $reader = IOFactory::createReader('xlsx');
            $reader->setReadDataOnly(true);
            $spreadsheet = $reader->load($file->getRealPath());
            $sheet = $spreadsheet->getActiveSheet();
            $data = $sheet->toArray(null, false, true, true);

            $insert = [];
            if (count($data) > 1) {
                foreach ($data as $row => $value) {
                    // Skip header row
                    if ($row > 1) {
                        // Skip completely empty rows
                        if (empty($value['A']) && empty($value['B'])) {
                            continue;
                        }

                        // Only insert rows where both values are present
                        if (empty($value['A']) && empty($value['B'])) {
                            $insert[] = [
                                'kategori_kode' => $value['A'],
                                'kategori_nama' => $value['B'],
                                'created_at' => now(),
                            ];
                        }
                    }
                }
            }

            if (count($insert) > 0) {
                KategoriModel::insertOrIgnore($insert);
                return response()->json([
                    'status' => true,
                    'message' => 'Data kategori berhasil diimport (' . count($insert) . ' data)'
                ]);
            }
        } catch (Exception $e) {
            Log::error('Kategori Import Error: ' . $e->getMessage());
            return response()->json([
                'status' => false,
                'message' => 'Gagal mengunggah file: ' . $e->getMessage()
            ], 500);
        }
    }

    return redirect('/kategori');
}
```



f. Tampilan

The screenshot shows the PWL - Starter Code application interface. On the left is a sidebar with navigation links: Home, Contact, Search, Dashboard, Data Pengguna, Level User, Data User, Data Barang, and Kategori Barang (which is highlighted in blue). The main content area displays a table titled 'Daftar Kategori barang' with columns for ID, Kode, Nama, and Aksi (Detail, Edit, Hapus). The table contains five entries: KTG001 (Elektronik), KTG002 (Pakaian), KTG003 (Makanan), KTG004 (Minuman), and KTG005 (Obat-obatan). Below the table, a message says 'Showing 1 to 5 of 5 entries'. At the top right of the main area are buttons for Import Kategori, Tambah Data, and Tambah Data (Ajax).

A modal window titled 'Import Data Kategori' is open. It includes a 'Download Template' button with a 'Download' link, a 'Pilih File' section with a 'Choose File' input field containing 'template_kategori.xlsx', and two buttons at the bottom: 'Batal' (Cancel) and 'Upload'.

Below the modal, a confirmation dialog box titled 'Berhasil' (Successful) is shown. It features a large green checkmark icon, the text 'Data kategori berhasil diimport (3 data)', and an 'OK' button.

At the very bottom of the screenshot, another instance of the 'Daftar Kategori' table is visible, showing 8 entries. The sidebar on the left of this second table also has 'Kategori Barang' selected.



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

Jawab:

- a. Mengimport lib yg diperlukan. Library PhpSpreadsheet digunakan untuk membaca dan memproses file Excel yang akan diupload oleh pengguna
 - b. Menambahkan route ntuk menangani proses import
 - c. Membuat form import. Method import() pada KategoriController dibuat untuk menampilkan form import. Form ini berisi input file untuk memilih file Excel dan tombol submit untuk mengunggah file tersebut.
 - d. Pembuatan Tampilan Form Import. File import.blade.php dibuat dengan konten form upload yang mencakup:
 - Input untuk memilih file Excel
 - Tombol untuk mendownload template Excel
 - Informasi format data yang diharapkan
 - Tombol untuk melakukan upload
 - e. Method import_ajax() pada KategoriController menangani proses import data
3. Submit kode untuk implemetasi praktikum 1 pada repository github kalian.

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>
```



```
<button onclick="modulAction('{{ url('/barang/import') }}')>Import Barang</button>
<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('/import', [BarangController::class, 'import']); // ajax form upload excel
Route::post('/import_ajax', [BarangController::class, 'import_ajax']); // AJAX import excel
Route::get('/export_excel', [BarangController::class, 'export_excel']); // ajax form download 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)

```
1 reference | 0 overrides
public function export_excel(): never {
    // 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();
```

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

```
// load lib excel
$spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
$sheet = $spreadsheet->getActiveSheet(); //ambil sheet yg aktif

$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); //bold header
```

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

```
$no = 1;
$baris = 2;
foreach ($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); //ambil nama kategori
    $baris++;
    $no++;
}
```

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

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



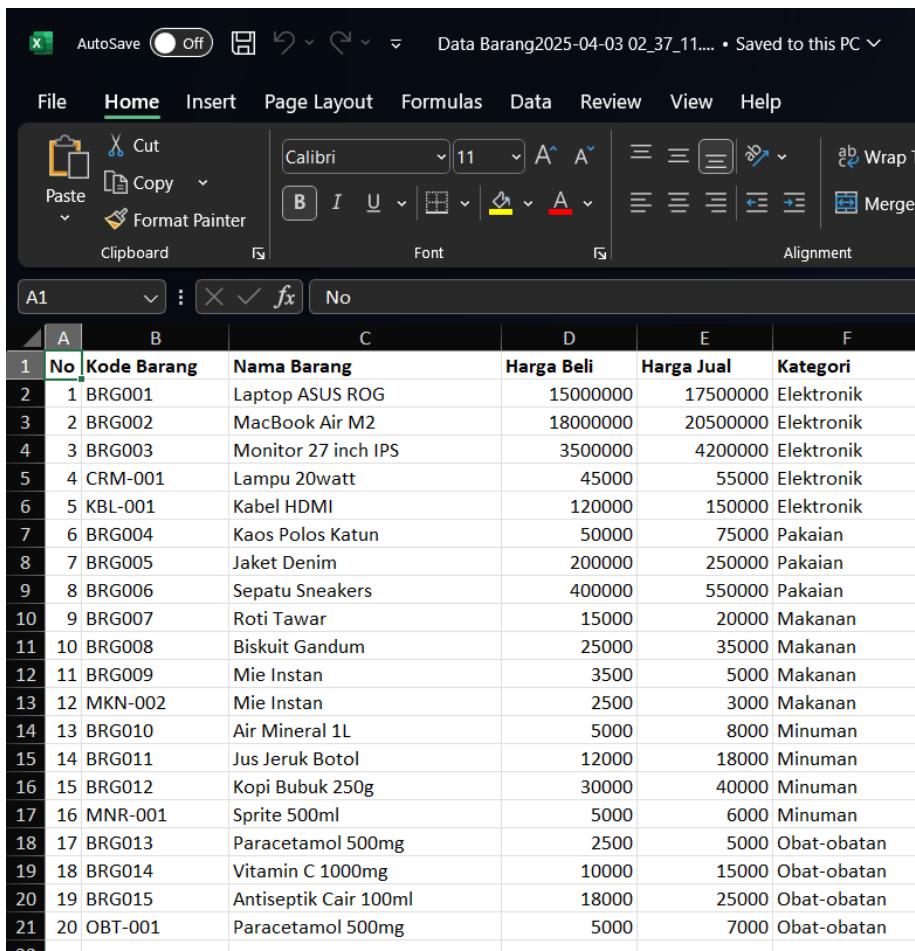
8. Bagian akhir proses export excel adalah kita set nama sheet, dan proses untuk dapat di download oleh pengguna

```
$sheet->setTitle('Data Barang'); //set title sheet
$writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
$fileName = 'Data Barang' . date(format: 'Y-m-d H:i:s') . '.xlsx';

// Set headers for file download
header(header: 'Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
header(header: 'Content-Disposition: attachment;filename="' . $fileName . '"');
header(header: 'Cache-Control: max-age=0');
header(header: 'Cache-Control: max-age=1');
header(header: 'Expires: Mon, 26 Jul 1997 05:00:00 GMT');
header(header: 'Last-Modified: ' . gdate(format: 'D, d M Y H:i:s') . ' GMT');
header(header: 'Pragma: public');

$writer->save('php://output');
exit; You, 2 seconds ago * Uncommitted changes
```

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



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Laptop ASUS ROG	15000000	17500000	Elektronik
2	BRG002	MacBook Air M2	18000000	20500000	Elektronik
3	BRG003	Monitor 27 inch IPS	3500000	4200000	Elektronik
4	CRM-001	Lampu 20watt	45000	55000	Elektronik
5	KBL-001	Kabel HDMI	120000	150000	Elektronik
6	BRG004	Kaos Polos Katun	50000	75000	Pakaian
7	BRG005	Jaket Denim	200000	250000	Pakaian
8	BRG006	Sepatu Sneakers	400000	550000	Pakaian
9	BRG007	Roti Tawar	15000	20000	Makanan
10	BRG008	Biskuit Gandum	25000	35000	Makanan
11	BRG009	Mie Instan	3500	5000	Makanan
12	MKN-002	Mie Instan	2500	3000	Makanan
13	BRG010	Air Mineral 1L	5000	8000	Minuman
14	BRG011	Jus Jeruk Botol	12000	18000	Minuman
15	BRG012	Kopi Bubuk 250g	30000	40000	Minuman
16	MNR-001	Sprite 500ml	5000	6000	Minuman
17	BRG013	Paracetamol 500mg	2500	5000	Obat-obatan
18	BRG014	Vitamin C 1000mg	10000	15000	Obat-obatan
19	BRG015	Antiseptik Cair 100ml	18000	25000	Obat-obatan
20	OBT-001	Paracetamol 500mg	5000	7000	Obat-obatan



Tugas 2 – Implementasi File Export Excel:

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

Jawab:

➤ Level User

- a. Modifikasi level/index.blade.php

```
<button onclick="modalAction('{{ url('/level/import') }}')>Import</button>
|<a href="{{ url('/level/export_excel') }}>Export Level</a>
|<button onclick="modalAction('{{ url('/level/create_ajax') }}')>Create</button>
```

- b. Tambahkan route pada route/web.php untuk bisa memproses export excel

```
Route::post('/import_ajax', [LevelController::class, 'import_ajax']); // handle file upload
|Route::get('/export_excel', [LevelController::class, 'export_excel']); // ajax.form.download.excel
|Route::put('/{id}', [LevelController::class, 'update']); // Menyimpan perubahan level
```

- c. Tambahkan fungsi export_excel() pada file LevelController.php

```
reference(0 overrides
public function export_excel(): never
{
    // Get level data to export
    $levels = LevelModel::select('level_id', 'level_kode', 'level_nama')
        ->orderBy('level_id')
        ->get();

    // Create new spreadsheet
    $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
    $sheet = $spreadsheet->getActiveSheet();

    // Set column headers
    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'ID Level');
    $sheet->setCellValue('C1', 'Kode Level');
    $sheet->setCellValue('D1', 'Nama Level');

    // Make headers bold
    $sheet->getStyle('A1:D1')->getFont()->setBold(true);

    // Populate data
    $no = 1;
    $baris = 2;
    foreach ($levels as $level) {
        $sheet->setCellValue('A' . $baris, $no);
        $sheet->setCellValue('B' . $baris, $level->level_id);
        $sheet->setCellValue('C' . $baris, $level->level_kode);
        $sheet->setCellValue('D' . $baris, $level->level_nama);
        $baris++;
        $no++;
    }

    // Auto-size columns
    foreach (range(start: 'A', end: 'D') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    // Set sheet title
    $sheet->setTitle('Data Level');

    // Create writer and set filename
    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $fileName = 'Data Level' . date(format: 'Y-m-d_H-i-s') . '.xlsx';

    // Set headers for file download
    header(header: 'Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
    header(header: 'Content-Disposition: attachment;filename=' . $fileName . '');
    header(header: 'Cache-Control: max-age=0');
    header(header: 'Cache-Control: max-age=1');
    header(header: 'Expires: Mon, 26 Jul 1997 05:00:00 GMT');
    header(header: 'Last-Modified: ' . gmdate(format: 'D, d M Y H:i:s') . ' GMT');
    header(header: 'Pragma: public');

    $writer->save('php://output');
    exit;
}
```



d. Download file Export

	A	B	C	D	E	F
1	No	ID Level	Kode Level	Nama Level		
2	1	1	ADM	Administrator		
3	2	2	MNG	Manager		
4	3	3	STF	Staf/Kasir		
5	4	4	CUS	Customer		
6	5	50	GUE	Guest		
7	6	51	SUP	Supervisor		
8	7	52	TRN	Trainee		

➤ Data User

a. Modifikasi user/index.blade.php

```
<a href="{{ url('/user/export_excel') }}" class="btn btn-primary">Export User</a>
```

b. Tambahkan route pada route/web.php untuk bisa memproses export excel

```
Route::get('/export_excel', [UserController::class, 'export_excel']); // ajax form download excel
```

c. Tambahkan fungsi export_excel() pada file UserController.php



```
reference [0 overrides]
public function export_excel(): never
{
    // Get user data to export with level relationship
    $users = UserModel::with('level')
        ->select('user_id', 'username', 'nama', 'level_id')
        ->orderBy('user_id')
        ->get();

    // Create new spreadsheet
    $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
    $sheet = $spreadsheet->getActiveSheet();

    // Set column headers
    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'ID User');
    $sheet->setCellValue('C1', 'Username');
    $sheet->setCellValue('D1', 'Nama Lengkap');
    $sheet->setCellValue('E1', 'Level');

    // Make headers bold
    $sheet->getStyle('A1:E1')->getFont()->setBold(true);

    // Populate data
    $no = 1;
    $baris = 2;
    foreach ($users as $user) {
        $sheet->setCellValue('A' . $baris, $no);
        $sheet->setCellValue('B' . $baris, $user->user_id);
        $sheet->setCellValue('C' . $baris, $user->username);
        $sheet->setCellValue('D' . $baris, $user->nama);
        $sheet->setCellValue('E' . $baris, $user->level->level_nama ?? ''); // Use level relationship
        $baris++;
        $no++;
    }

    // Auto-size columns
    foreach (range(start: 'A', end: 'E') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

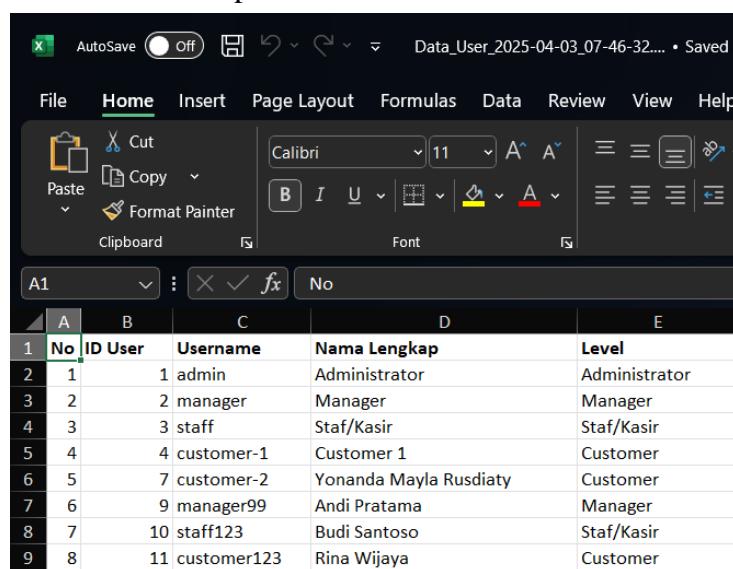
    // Set sheet title
    $sheet->setTitle('Data User');

    // Create writer and set filename
    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $fileName = 'Data_User_' . date(format: 'Y-m-d_H-i-s') . '.xlsx';

    // Set headers for file download
    header(header: 'Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
    header(header: 'Content-Disposition: attachment;filename=' . $fileName . '');
    header(header: 'Cache-Control: max-age=0');
    header(header: 'Cache-Control: max-age=1');
    header(header: 'Expires: Mon, 26 Jul 1997 05:00:00 GMT');
    header(header: 'Last-Modified: ' . gdate(format: 'D, d M Y H:i:s') . ' GMT');
    header(header: 'Pragma: public');

    $writer->save('php://output');
    exit;
}
```

d. Download file Export



No	ID User	Username	Nama Lengkap	Level
1	1	admin	Administrator	Administrator
2	2	manager	Manager	Manager
3	3	staff	Staf/Kasir	Staf/Kasir
4	4	customer-1	Customer 1	Customer
5	5	customer-2	Yonanda Mayla Rusdiyat	Customer
6	6	manager99	Andi Pratama	Manager
7	7	staff123	Budi Santoso	Staf/Kasir
8	8	customer123	Rina Wijaya	Customer
9				



➤ Kategori Barang

- Modifikasi kategori/index.blade.php

```
<a href="{{ url('/kategori/export_excel') }}" class="btn btn-primary">Export Kategori</a>
```

- Tambahkan route pada route/web.php untuk bisa memproses export excel

```
Route::get('/export_excel', [KategoriController::class, 'export_excel']); // ajax form download excel
```

- Tambahkan fungsi export_excel() pada file KategoriController.php

```
function () {
    public function export_excel(): never
    {
        // Get kategori data to export
        $kategoris = KategoriModel::select('kategori_id', 'kategori_kode', 'kategori_nama')
            ->orderBy('kategori_id')
            ->get();

        // Create new spreadsheet
        $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
        $sheet = $spreadsheet->getActiveSheet();

        // Set column headers
        $sheet->setCellValue('A1', 'No');
        $sheet->setCellValue('B1', 'ID Kategori');
        $sheet->setCellValue('C1', 'Kode Kategori');
        $sheet->setCellValue('D1', 'Nama Kategori');

        // Make headers bold
        $sheet->getStyle('A1:D1')->getFont()->setBold(true);

        // Populate data
        $no = 1;
        $baris = 2;
        foreach ($kategoris as $kategori) [
            $sheet->setCellValue('A' . $baris, $no);
            $sheet->setCellValue('B' . $baris, $kategori->kategori_id);
            $sheet->setCellValue('C' . $baris, $kategori->kategori_kode);
            $sheet->setCellValue('D' . $baris, $kategori->kategori_nama);
            $baris++;
            $no++;
        ]

        // Auto-size columns
        foreach (range(start: 'A', end: 'D') as $columnID) {
            $sheet->getColumnDimension($columnID)->setAutoSize(true);
        }

        // Set sheet title
        $sheet->setTitle('Data Kategori');

        // Create writer and set filename
        $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
        $fileName = 'Data_Kategori_' . date(format: 'Y-m-d_H-i-s') . '.xlsx';

        // Set headers for file download
        header(header: 'Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
        header(header: 'Content-Disposition: attachment;filename=' . $fileName . "'");
        header(header: 'Cache-Control: max-age=0');
        header(header: 'Cache-Control: max-age=1');
        header(header: 'Expires: Mon, 26 Jul 1997 05:00:00 GMT');
        header(header: 'Last-Modified: ' . gmdate(format: 'D, d M Y H:i:s') . ' GMT');
        header(header: 'Pragma: public');

        $writer->save('php://output');
        exit;
    }
}
```



d. Download file Export

No	ID Kategori	Kode Kategori	Nama Kategori
1	KTG001	Elektronik	
2	KTG002	Pakaian	
3	KTG003	Makanan	
4	KTG004	Minuman	
5	KTG005	Obat-obatan	
6	KTG006	Aksesoris	
7	KTG007	Peralatan Rumah Tangga	
8	KTG008	Alat Tulis	

➤ Data Supplier

a. Modifikasi supplier/index.blade.php

```
<a href="{{ url('/supplier/export_excel') }}" class="btn btn-primary">Export Supplier</a>
```

b. Tambahkan route pada route/web.php untuk bisa memproses export excel

```
Route::post('/import_ajax', [SupplierController::class, 'import_ajax']); // Handles the file upload
Route::get('/export_excel', [SupplierController::class, 'export_excel']); // ajax form download excel
```

c. Tambahkan fungsi export_excel() pada file SupplierController.php



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI

POLITEKNIK NEGERI MALANG

JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141

Telp. (0341) 404424 – 404425, Fax (0341) 404420

<http://www.polinema.ac.id>

```
public function export_excel() {
    // Get supplier data to export
    $suppliers = SupplierModel::select('supplier_id', 'supplier.kode', 'supplier.nama', 'supplier.alamat', 'supplier.telp')
        ->orderBy('supplier_id')
        ->get();

    // Create new spreadsheet
    $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
    $sheet = $spreadsheet->getActiveSheet();

    // Set column headers
    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'ID Supplier');
    $sheet->setCellValue('C1', 'Kode Supplier');
    $sheet->setCellValue('D1', 'Nama Supplier');
    $sheet->setCellValue('E1', 'Alamat');
    $sheet->setCellValue('F1', 'Telepon');

    You, 1 second ago * Uncommitted changes
    // Make headers bold
    $sheet->getStyle('A1:F1')->getFont()->setBold(true);

    // Populate data
    $no = 1;
    $baris = 2;
    foreach ($suppliers as $supplier) {
        $sheet->setCellValue('A' . $baris, $no);
        $sheet->setCellValue('B' . $baris, $supplier->supplier_id);
        $sheet->setCellValue('C' . $baris, $supplier->supplier.kode);
        $sheet->setCellValue('D' . $baris, $supplier->supplier.nama);
        $sheet->setCellValue('E' . $baris, $supplier->supplier.alamat);
        $sheet->setCellValue('F' . $baris, $supplier->supplier.telp);
        $baris++;
        $no++;
    }

    // Auto size columns
    foreach (range(start: 'A', end: 'F') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    // Set sheet title
    $sheet->setTitle('Data Supplier');

    // Create writer and set filename
    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $fileName = 'Data_Supplier_' . date(format: 'Y-m-d_H-i-s') . '.xlsx';

    // Set headers for file download
    header(header: 'Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');
    header(header: 'Content-Disposition: attachment;filename=' . $fileName . '');
    header(header: 'Cache-Control: max-age=0');
    header(header: 'Cache-Control: max-age=1');
    header(header: 'Expires: Mon, 26 Jul 1997 05:00:00 GMT');
    header(header: 'Last-Modified: ' . gmdate(format: 'D, d M Y H:i:s') . ' GMT');
    header(header: 'Pragma: public');

    $writer->save('php://output');
    exit;
}
```

d. Download file Export

No	ID Supplier	Kode Supplier	Nama Supplier	Alamat	Telepon
1	1	SUP001	PT. Elektronik Jaya	Jl. Industri No. 123, Jakarta	021-5551234
2	2	SUP002	CV. Fashion Indonesia	Jl. Tekstil No. 45, Bandung	022-6667890
3	3	SUP003	UD. Makanan Sehat	Jl. Pasar Baru No. 78, Surabaya	031-7772345
4	4	SUP004	PT. Minuman Prima	Jl. Industri Minuman Blok A5, Tangerang	021-8883456
5	5	SUP005	PT. Farma Indonesia	Jl. Farmasi No. 56, Semarang	024-9994567
6	6	SUP006	PT. Teknologi Nusantara	Jl. Merdeka No. 88, Yogyakarta	
7	7	SUP007	CV. Sumber Makmur	Jl. Raya Bogor No. 15, Bogor	
8	8	SUP008	PT. Jaya Abadi	Jl. Sudirman No. 99, Medan	
9	9	SUP009	UD. Cahaya Baru	Jl. Diponegoro No. 22, Denpasar	
10	10	SUP010	PT. Sentosa Raya	Jl. Gatot Subroto No. 77, Makassar	

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

Jawab:

- Kita modifikasi inde.blade untuk mengganti button Tambah Data menjadi Export Excel
- Setelah itu kita arahkan routenya agar ke Controller dengan memproses function export_excel
- Di Controller kita akan membuat function export_excel. Pertama kita akan



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

mengambil data barang yg akan diexport. Lalu kita load lib excel yg dibutuhkan, dan mengatur header untuk download

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



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

```
PS D:\Apps\laragon\www\PWL2025\week 8\jobsheet> 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 (v3.1.1)
- Locking dompdf/dompdf (v3.1.0)

89 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

INFO: No publishable resources for tag [laravel-assets].

No security vulnerability advisories found.
Using version ^3.1 for 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_excel') }}" class="btn btn-primary"><i class="fa fa-file-excel"></i> Export Excel
<a href="{{ url('/barang/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Pdf</a>
<button onclick="modalAction('{{ url('/barang/create_ajax') }}'" class="btn btn-success">Tambah Data (Ajax)</button>
```

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

```
Route::get('/export_excel', [BarangController::class, 'export_excel']); // ajax form download excel
Route::get('/export_pdf', [BarangController::class, 'export_pdf']); // ajax form download pdf
```

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



```
1 reference | 0 overrides
public function export_pdf(): Response {
    $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 url
    $pdf ->render();

    return $pdf->stream('Data Barang'.date(format: '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 dompdf 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;
        }

        .text-center {
            text-align: center;
        }

        .p-i {
            padding: 5px 1px 5px 1px;
        }
    </style>
</head>
<body>
```



```
.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>


|                                                                                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <span class="text-center d-block font-11 font-bold mb-1">KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN</span> <span class="text-center d-block font-13 font-bold mb-1">TEKNOLOGI</span> <span class="text-center d-block font-13 font-bold mb-1">POLITEKNIK NEGERI</span> <span class="text-center d-block font-13 font-bold mb-1">MALANG</span> <span class="text-center d-block font-10">Jl. Soekarno-Hatta No. 9 Malang</span> <span class="text-center d-block font-10">65141</span> <span class="text-center d-block font-10">Telepon (0341) 404424 Pes. 101-</span> <span class="text-center d-block font-10">105, 0341-404428, Fax. (0341) 404420</span> <span class="text-center d-block font-10">Laman: <a href="http://www.polinema.ac.id">www.polinema.ac.id</a></span> |
|--------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|


</body>
```

```
<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(num: $b->harga_beli, decimals: 0, decimal_separator: ',', thousands_separator: '.') }}</td>
            <td class="text-right">{{ number_format(num: $b->harga_jual, decimals: 0, decimal_separator: ',', thousands_separator: '.') }}</td>
            <td>{{ $b->kategori->kategori_nama }}</td>
        </tr>
        @endforeach
    </tbody>
</table>
</body>
</html>
```



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

Jawab:

Berikut adalah hasil nya:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Laptop ASUS ROG	15.000.000	17.500.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
2	BRG002	MacBook Air M2	18.000.000	20.500.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
3	BRG003	Monitor 27 inch IPS	3.500.000	4.200.000	Elektronik	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
4	BRG004	Kaos Polos Katun	50.000	75.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
5	BRG005	Jaket Denim	200.000	250.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
6	BRG006	Sepatu Sneakers	400.000	550.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
7	BRG007	Roti Tawar	15.000	20.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
8	BRG008	Biskuit Gandum	25.000	35.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
9	BRG009	Mie Instan	3.500	5.000	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
10	BRG010	Air Mineral 1L	5.000	8.000	Minuman	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

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 BARANG

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Laptop ASUS ROG	15.000.000	17.500.000	Elektronik
2	BRG002	MacBook Air M2	18.000.000	20.500.000	Elektronik
3	BRG003	Monitor 27 inch IPS	3.500.000	4.200.000	Elektronik
4	CRM-001	Lampu 20watt	45.000	55.000	Elektronik
5	KBL-001	Kabel HDMI	120.000	150.000	Elektronik
6	BRG004	Kaos Polos Katun	50.000	75.000	Pakaian
7	BRG005	Jaket Denim	200.000	250.000	Pakaian
8	BRG006	Sepatu Sneakers	400.000	550.000	Pakaian
9	BRG007	Roti Tawar	15.000	20.000	Makanan
10	BRG008	Biskuit Gandum	25.000	35.000	Makanan
11	BRG009	Mie Instan	3.500	5.000	Makanan
12	MKN-002	Mie Instan	2.500	3.000	Makanan
13	BRG010	Air Mineral 1L	5.000	8.000	Minuman
14	BRG011	Jus Jeruk Botol	12.000	18.000	Minuman
15	BRG012	Kopi Bubuk 250g	30.000	40.000	Minuman
16	MNR-001	Sprite 500ml	5.000	6.000	Minuman
17	BRG013	Paracetamol 500mg	2.500	5.000	Obat-obatan



Tugas 3 – Implementasi Export PDF pada Laravel :

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

Jawab:

➤ Level User

- a. Modifikasi index.blade untuk menambahkan button Export Pdf

```
<a href="{{ url('/level/export_excel') }}" class="btn btn-primary">Export Excel</a>
<a href="{{ url('/level/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Pdf</a>
```

- b. Menambahkan route export_pdf pada web.php

```
Route::get('/export_excel', [LevelController::class, 'export_excel']); // ajax form download excel
Route::get('/export_pdf', [LevelController::class, 'export_pdf']); // ajax form download pdf
```

- c. Membuat function export_pdf pada controller

```
1 reference | 0 overrides
public function export_pdf(): Response
{
    // Get level data to export
    $levels = LevelModel::select('level_id', 'level_kode', 'level_nama')
        ->orderBy('level_id')
        ->orderBy('level_kode')
        ->get();

    // use Barryvdh\DomPDF\Facade\PDF
    $pdf = Pdf::loadView('level.export_pdf', ['levels' => $levels]);
    $pdf->setPaper('a4', 'portrait'); // Set paper size and orientation
    $pdf->setOption("isRemoteEnabled", true); // Enable remote images
    $pdf->render();

    return $pdf->stream('Data Level' . date(format: 'Y-m-d_H-i-s') . '.pdf');
}
```

- d. Membuat view baru yaitu 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: 0px 20px 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;
        }

        .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;
        }
    </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 Faks. 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 LEVEL</h3>
    <table class="border-all">
        <thead>
            <tr>
                <th class="text-center">No:</th>
                <th>ID Level</th>
                <th>Kode Level</th>
                <th>Nama Level</th>
            </tr>
        </thead>
        <tbody>
            @foreach($levels as $level)
            <tr>
                <td class="text-center">{{ $loop->iteration }}</td>
                <td>{{ $level->level_id }}</td>
                <td>{{ $level->level_kode }}</td>
                <td>{{ $level->level_nama }}</td>
            </tr>
            @endforeach
        </tbody>
    </table>
</body>
</html>
```



e. Tampilan pdfnya

The screenshot shows a browser window with the URL 127.0.0.1:8000/levels/export_pdf. The page title is "export.pdf". At the top, there is a header with the logo of Politeknik Negeri Malang, the name of the institution, address, phone number, and website. Below the header, the title "LAPORAN DATA LEVEL" is displayed. A table follows, showing the data levels:

No	ID Level	Kode Level	Nama Level
1	1	ADM	Administrator
2	2	MNG	Manager
3	3	STF	Staf/Kasir
4	4	CUS	Customer
5	50	GUE	Guest
6	51	SUP	Supervisor
7	52	TRN	Trainee

➤ Data User

a. Modifikasi index.blade untuk menambahkan button Export Pdf

```
<a href="{{ url('user/export_excel') }}" class="btn btn-primary">Export Excel</a>
<a href="{{ url('/user/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Pdf</a>
```

b. Menambahkan route export_pdf pada web.php

```
Route::get('/export_pdf', [UserController::class, 'export_pdf']); // ajax form download pdf
```

c. Membuat function export_pdf pada controller

```
1 reference | 0 overrides
public function export_pdf(): Response
{
    // Get user data to export with level relationship
    $users = UserModel::with('level')
        ->select('user_id', 'username', 'nama', 'level_id')
        ->orderBy('user_id')
        ->orderBy('level_id')
        ->get();

    // use Barryvdh\DomPDF\Facade\PDF
    $pdf = Pdf::loadView('user.export_pdf', ['users' => $users]);
    $pdf->setPaper('a4', 'portrait'); // Set paper size and orientation
    $pdf->setOption("isRemoteEnabled", true); // Enable remote images
    $pdf->render();

    return $pdf->stream('Data User'.date(format: 'Y-m-d_H-i-s').'.pdf');
}
```

d. Membuat view baru yaitu export_pdf.blade.php



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI

Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65114
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

```
<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;
        }

        .text-center {
            text-align: center;
        }

        .p-1 {
            padding: 5px 1px 5px 1px;
        }

        .font-10 {
            font-size: 10pt;
        }
    </style>
</head>
<body>
    <table>
        <tr>
            <td>
                <img alt="Image placeholder" class="image" />
            </td>
            <td>
                <p>Text content goes here</p>
            </td>
        </tr>
    </table>
</body>
</html>
```

```
.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 width="15%" class="text-center">
        <td></td>
    </tr>
    <tr width="85%">
        <td>
            <span class="text-center d-block font-11 font-bold mb-1">KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET&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-18">Jl. Soekarno-Hatta No. 9 Malang 65145</span>
            <span class="text-center d-block font-18">Telepon : (0341) 404424 Per. 181-185, 0341-484428, Fax. (0341) 404428</span>
            <span class="text-center d-block font-18">Laman : www.polinema.ac.id</span>
        </td>
    </tr>
</table>
```

<h3 class="text-center">LAPORAN DATA USER</h3>

```
<table class="border-all">
    <thead>
        <tr>
            <th class="text-center">Nom</th>
            <th>ID User</th>
            <th>Username</th>
            <th>Nama Lengkap</th>
            <th>Level</th>
        </tr>
    </thead>
    <tbody>
        @foreach($users as $user)
            <tr>
                <td class="text-center" style="vertical-align: top;">({$loop->iteration})</td>
                <td>{ $user->user_id }</td>
                <td>{ $user->username }</td>
                <td>{ $user->nama }</td>
                <td>{ $user->level->level_nama ?? '-' }</td>
            </tr>
        @endforeach
    </tbody>
</table>
```



- e. Tampilan pdfnya

No	ID User	Username	Nama Lengkap	Level
1	1	admin	Administrator	Administrator
2	2	manager	Manager	Manager
3	3	staff	Staf/Kasir	Staf/Kasir
4	4	customer-1	Customer 1	Customer
5	7	customer-2	Yonanda Mayla Rusdiyat	Customer
6	9	manager99	Andi Pratama	Manager
7	10	staff123	Budi Santoso	Staf/Kasir
8	11	customer123	Rina Wijaya	Customer

➤ Kategori Barang

- a. Modifikasi index.blade untuk menambahkan button Export Pdf

```
<a href="{{ url('/kategori/export_excel') }}" class="btn btn-primary">Export Excel</a>
<a href="{{ url('/kategori/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Pdf</a>
button onclick="modalAction('{{ url('/kategori/create_ajax') }}'" class="btn btn-success">Tambah Data Ajax</button>
```

- b. Menambahkan route export_pdf pada web.php

```
Route::get('/export_pdf', [KategoriController::class, 'export_pdf']); // ajax form download pdf
```

- c. Membuat function export_pdf pada controller

```
1 reference | 0 overrides
public function export_pdf(): Response
{
    // Get kategori data to export
    $kategoris = KategoriModel::select('kategori_id', 'kategori_kode', 'kategori_nama')
        ->orderBy('kategori_id')
        ->orderBy('kategori_kode')
        ->get();

    // use Barryvdh\DomPDF\Facade\PDF
    $pdf = Pdf::loadView('kategori.export_pdf', ['kategoris' => $kategoris]);
    $pdf->setPaper('a4', 'portrait'); // Set paper size and orientation
    $pdf->setOption("isRemoteEnabled", true); // Enable remote images
    $pdf->render();

    return $pdf->stream('Data_Kategori_' . date(format: 'Y-m-d_H-i-s') . '.pdf');
}
```

- d. Membuat view baru yaitu 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;
        }

        .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;
        }
    </style>

```

```
.border-bottom-header {
    border-bottom: 1px solid;
}

.border-all,
.border-all th,
.border-all td {
    border: 1px solid;
}

</head>
<body>
    <table class="border-bottom-header">
        <tr width="15%" class="text-center">
            <td><img alt="Logo of Polinema-bw.png" /></td>
            <td width="85%">
                <span class="text-center d-block font-bold mb-1">KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN<br/>TEKNOLOGI</span>
                <span class="text-center d-block font-bold mb-1">POLITEKNIK NEGERI<br/>MALANG</span>
                <span class="text-center d-block font-10">Jl. Soekarno-Hatta No. 9 Malang<br/>65141</span>
                <span class="text-center d-block font-10">Telepon (0341) 404424 Pes. 101-105, 0341-404426, 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 KATEGORI</h3>
    <table class="border-all">
        <thead>
            <tr>
                <th class="text-center">No</th>
                <th>Kategori</th>
                <th>Code Kategori</th>
                <th>Nama Kategori</th>
            </tr>
        </thead>
        <tbody>
            @foreach($kategoris as $kategori)
            <tr>
                <td class="text-center" data-loop="iteration">{{ $loop }}</td>
                <td>{{ $kategori->kategori_id }}</td>
                <td>{{ $kategori->kategori_kode }}</td>
                <td>{{ $kategori->kategori_nama }}</td>
            </tr>
            @endforeach
        </tbody>
    </table>
</body>
</html>
```



- e. Tampilan pdfnya



The screenshot shows a PDF document titled 'LAPORAN DATA KATEGORI' from the Politeknik Negeri Malang website. The document includes the university's logo and contact information at the top. Below the title is a table with 8 rows of data:

No	ID Kategori	Kode Kategori	Nama Kategori
1	1	KTG001	Elektronik
2	2	KTG002	Pakaian
3	3	KTG003	Makanan
4	4	KTG004	Minuman
5	5	KTG005	Obat-obatan
6	12	KTG006	Aksesoris
7	13	KTG007	Peralatan Rumah Tangga
8	14	KTG008	Alat Tulis

➤ Data Supplier

- a. Modifikasi index.blade untuk menambahkan button Export Pdf

```
<form> {{-->}} {{-->}} {{-->}} {{-->}} {{-->}} {{-->}}
```

- b. Menambahkan route export_pdf pada web.php

```
Route::get('/export_pdf', [SupplierController::class, 'export_pdf']); // ajax form download pdf
```

- c. Membuat function export_pdf pada controller

```
public function export_pdf(): Response
{
    // Get supplier data to export
    $suppliers = SupplierModel::select('supplier_id', 'supplier_kode', 'supplier_nama', 'supplier_alamat', 'supplier_telp')
        ->orderBy('supplier_id')
        ->orderBy('supplier_kode')
        ->get();

    // use Barryvdh\DomPDF\Facade\PDF
    $pdf = Pdf::loadView('supplier.export_pdf', ['suppliers' => $suppliers]);
    $pdf->setPaper('a4', 'portrait'); // Set paper size and orientation
    $pdf->setOption("isRemoteEnabled", true); // Enable remote images
    $pdf->render();

    return $pdf->stream('Data_Supplier_'. date(format: 'Y-m-d_H-i-s') . '.pdf');
}
```

- d. Membuat view baru yaitu 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: 0px 20px 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;
        }

        .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;
        }
    </style>

```

```
<body>
    <style>
        .border-bottom-header {
            border-bottom: 1px solid;
        }

        .border-all,
        .border-all th,
        .border-all td {
            border: 1px solid;
        }
    </style>
    <table class="border-bottom-header">
        <tr>
            <td width="15%" class="text-center">
                
            </td>
            <td width="85%">
                <span class="text-center d-block font-13 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 Pers. 101-
                    105, 0341-404426, Fax. (0341) 404428</span>
                <span class="text-center d-block font-10">E-mail: www.polinema.ac.id</span>
            </td>
        </tr>
    </table>
    <h3 class="text-center">LAPORAN DATA SUPPLIER</h3>
    <table class="border-all">
        <thead>
            <tr>
                <th class="text-center">No</th>
                <th>ID Supplier</th>
                <th>Kode Supplier</th>
                <th>Nama Supplier</th>
                <th>Alamat</th>
                <th>Telepon</th>
            </tr>
        </thead>
        <tbody>
            @foreach($suppliers as $supplier)
            <tr>
                <td class="text-center">{{ $loop->iteration }}</td>
                <td>{{ $supplier->supplier_id }}</td>
                <td>{{ $supplier->supplier_kode }}</td>
                <td>{{ $supplier->supplier_nama }}</td>
                <td>{{ $supplier->supplier_alamat }}</td>
                <td>{{ $supplier->supplier_telp }}</td>
            </tr>
            @endforeach
        </tbody>
    </table>
</body>
</html>
```



e. Tampilan pdfnya

No	ID Supplier	Kode Supplier	Nama Supplier	Alamat	Telepon
1	1	SUP001	PT. Elektronik Jaya	Jl. Industri No. 123, Jakarta	(021) 5551234
2	2	SUP002	CV. Fashion Indonesia	Jl. Tekstil No. 45, Bandung	(022) 6667890
3	3	SUP003	UD. Makanan Sehat	Jl. Pasar Baru No. 78, Surabaya	(031) 7772345
4	4	SUP004	PT. Minuman Prima	Jl. Industri Minuman Blok A5, Tangerang	(021) 5883456
5	5	SUP005	PT. Farma Indonesia	Jl. Faransi No. 56, Semarang	(024) 9994567
6	51	SUP006	PT. Teknologi Nusantara	Jl. Merdeka No. 88, Yogyakarta	
7	52	SUP007	CV. Sumber Mandiri	Jl. Raya Bogor No. 15, Bogor	
8	53	SUP008	PT. Jaya Abadi	Jl. Sudirman No. 99, Medan	
9	54	SUP009	UD. Cahaya Baru	Jl. Diponegoro No. 22, Denpasar	
10	55	SUP010	PT. Sentosa Raya	Jl. Guter Subrosti No. 77, Makassar	

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
Jawab:
 - a. Kita tambahkan button untuk export pdf nya dahulu pada barang/index.blade
 - b. Lalu kita atur route nya agar saat button di klik dia kaan mengarahkan ke function export_pdf pada BarangController. Route berada dalam grup middleware authorize:ADM,MNG yang memastikan hanya admin dan manager yang dapat mengakses fungsi ini
 - c. Lalu membuat function export_pdf Memanggil facade Pdf untuk membuat object PDF, dengan konfigurasi pdf untuk: mengatur ukuran kertas a4, mengatur orrientasi potrait, dan mengaktifkan dukungan gambar
 - d. Membuat view export_pdf yang mendefinisikan font, margin, dan line-height untuk dokumen, mengatur format tabel dengan border, mendefinisikan class untuk text alignment (center, right), dan mengatur style untuk gambar dan header
3. Submit kode untuk implemetasi 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

Jawab:

- a. Memodifikasi UserModel dengan menambahkan var ‘foto_profile’ pada \$fillable

```
0 references
protected $fillable = ['user_id', 'username', 'password', 'nama', 'level_id', 'created_at', 'updated_at', 'foto_profile'];
```

- b. Membuat migration untuk Menambahkan Field Foto Profile dan mengisi function up() serta down()



```
PS D:\Apps\laragon\www\PWL2025\week_8\jobsheet> php artisan make:migration add_foto_profile_to_users_table --table=users
[INFO] Migration [D:\Apps\laragon\www\PWL2025\week_8\jobsheet\database\migrations\2025_04_04_055215_add_foto_profile_to_users_table.php] created successfully.
```

```
You, 5 minutes ago | 1 author (You)
<php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

You, 5 minutes ago | 1 author (You)
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::table('m_user', function (Blueprint $table): void {
            $table->string('foto_profile')->nullable()->after('level_id');
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::table('m_user', function (Blueprint $table): void {
            $table->dropColumn('foto_profile');
        });
    }
};
```

```
PS D:\Apps\laragon\www\PWL2025\week_8\jobsheet> php artisan migrate
INFO  Running migrations.
2025_04_04_055215_add_foto_profile_to_users_table ..... 20ms DONE
```

c. Membuat symbolic link

```
PS D:\Apps\laragon\www\PWIL2025\week_8\jobsheet> php artisan storage:link
[INFO] The [D:\Apps\laragon\www\PWIL2025\week_8\jobsheet\public\storage] link has been connected to [D:\Apps\laragon\www\PWIL2025\week_8\jobsheet\storage\app\public].
```

d. Membuat halaman profile /user/profile.blade



```
@push('scripts')
<script>
    // Script untuk menampilkan nama file yang dipilih
    $(document).ready(function() {
        $('#foto_profile').on('change', function() {
            // Dapatkan nama file
            var fileName = $(this).val().split('\\').pop();
            // Ganti label dengan nama file
            $(this).next('.custom-file-label').html(fileName);

            // Preview image
            if (this.files && this.files[0]) {
                var reader = new FileReader();
                reader.onload = function(e) {
                    $('#img-img-circle').attr('src', e.target.result);
                }
                reader.readAsDataURL(this.files[0]);
            }
        });
    });
</script>
@endpush
```

- e. Membuat method controller pada UserController untuk profile() dan updatePhoto()

```
/** 
 * Show the user profile page
 */
reference |0 overrides
public function profile(): Factory|mixed|RedirectResponse|View
{
    // Use Laravel's auth system to get the current user
    $user = auth()->user();

    if (!$user) {
        return redirect('/login')->with('error', 'Silahkan login terlebih dahulu');
    }

    // Define breadcrumb as an OBJECT (note the (object) cast)
    $breadcrumb = (object) [
        'title' => 'Profile User',
        'list' => ['Home', 'Profile']
    ];
    You, 11 minutes ago + Uncommitted changes

    // Define page information
    $page = (object) [
        'title' => 'Profil Pengguna'
    ];

    // Set active menu
    $activeMenu = 'profile';

    return view('user.profile', compact(var_name: 'user', var_names: 'breadcrumb', 'page', 'activeMenu'));
}
```

```
/** 
 * Update user profile photo
 */
reference |0 overrides
public function updatePhoto(Request $request)
{
    // Validasi file
    $request->validate([
        'foto_profile' => 'required|image|mimes:jpeg,png,jpg|max:2048',
    ]);

    try {
        // Get the authenticated user
        $user = auth()->user();

        if (!$user) {
            return redirect('/login')->with('error', 'Silahkan login terlebih dahulu');
        }

        // Get the user ID
        $userId = $user->user_id;

        // Get the proper UserModel instance
        $userModel = UserModel::find($userId);

        if (!$userModel) {
            return redirect('/login')->with('error', 'User tidak ditemukan');
        }

        // Delete old photo if exists
        if ($userModel->foto_profile && file_exists(storage_path('app/public/' . $userModel->foto_profile))) {
            Storage::disk('public')->delete($userModel->foto_profile);
        }

        // Store new photo
        $fileName = 'profile_' . $userId . '_' . time() . '.' . $request->foto_profile->extension();
        $path = $request->foto_profile->storeAs('profiles', $fileName, 'public');

        // Update user record using update() method
        $userModel::where('user_id', $userId)->update([
            'foto_profile' => $path
        ]);

        return redirect()->back()->with('success', 'Foto profile berhasil diperbarui');
    } catch (\Exception $e) {
        return redirect()->back()->with('error', 'Gagal mengupload foto: ' . $e->getMessage());
    }
}
```



f. Konfigurasi route pada web.php

```
// Dashboard yang dapat diakses oleh semua role yang login
Route::middleware(['auth'])->group(function () {
    Route::get('/', [WelcomeController::class, 'index'])->name('dashboard');
    Route::get('/profile', [UserController::class, 'profile']);
    Route::post('/user/update-photo', [UserController::class, 'updatePhoto']);
});
You, last week ·feat: initial commit
```

g. Memodifikasi /layouts/sidebar.blade.php untuk menambahkan button profile

```
<li class="nav-item">
    <a href="{{ url('/profile') }}" class="nav-link">
        <i class="nav-icon fas fa-user"></i>
        <p>Profile</p>
    </a>
</li>
```

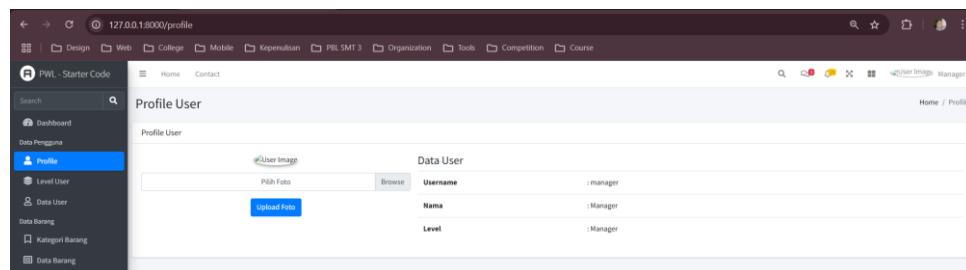
h. Modifikasi header.blade.php untuk menampilkan profile foto dan level nya

```
<!-- Left navbar links -->
<ul class="navbar-nav">
    <li class="nav-item">
        <a class="nav-link" data-widget="pushmenu" href="#" role="button"><i class="fas fa-bars"></i></a>
    </li>
```

```
<!-- User profile dropdown -->
<li class="nav-item dropdown user-menu">
    <a href="#" class="nav-link dropdown-toggle" data-toggle="dropdown">
        
        <span class="d-none d-md-inline">{{ auth()>user()>nama }}</span>
    </a>
    <ul class="dropdown-menu dropdown-menu-lg dropdown-menu-right">
        <!-- User image -->
        <li class="user-header bg-primary">
            
            <p>
                {{ auth()>user()>nama }}
                <small>{{ auth()>user()>level->level_nama ?? 'User' }}</small>
            </p>
        </li>
        <!-- Menu Footer-->
        <li class="user-footer">
            You, 21 minutes ago * Uncommitted changes
            <a href="{{ url('/profile') }}" class="btn btn-default btn-flat">Profile</a>
            <a href="#" class="btn btn-default btn-flat float-right"
                onclick="event.preventDefault(); document.getElementById('logout-form').submit();">
                Sign out
            </a>
            <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display: none;">
                @csrf
            </form>
        </li>
    </ul>
</li>
```

i. Tampilan.

- Disini saya login sebagai manager. Maka tampilan awalnya adalah sbg berikut





-
- Setelah diunggah foto profil, maka profil akan terupdate

The top screenshot shows the 'Profile User' page of a web application. On the left is a sidebar with navigation links: PWL - Starter Code, Dashboard, Data Pengguna, Profile (which is highlighted in blue), Level User, Data User, Data Barang, Kategori Barang, Data Barang, Data Supplier, and Data Supplier. The main area has a heading 'Profile User' and a placeholder for a profile picture with a red cartoon character. Below it are fields for 'Username' (manager), 'Nama' (Manager), and 'Level' (Manager). There are 'Pilih Foto' and 'Browse' buttons for uploading a photo, and a 'Upload Foto' button. The bottom screenshot shows a dashboard with a large circular placeholder for a profile picture with a red cartoon character, labeled 'Manager'. Below it are buttons for 'Profile' and 'Sign out'. At the top of the dashboard, there are icons for notifications (3), messages, and a search bar, followed by a 'Manager' link.

2. Jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
3. Submit kode untuk implementasi export pdf pada repository github kalian.

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