



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

JOBSHEET 08

File Import dan Export ke PDF dan Excel pada Laravel

Sesuai dengan **Studi Kasus PWL.pdf**.

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

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

A. File Upload (import data) pada Laravel

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

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



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

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

Praktikum 1 – Implementasi Upload File untuk import data:

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

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

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

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



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

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

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">
        <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>
    </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>Kode Barang</th><th>Harga
Beli</th><th>Harga Jual</th><th>Kategori</th><th>Aksi</th></tr>
    </thead>
    <tbody></tbody>
</table>
</div>
<div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static"
data-keyboard="false" data-width="75%"></div>

@endsection

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

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



```
orderable: true,
searchable: false,
render: function(data, type, row){
    return new Intl.NumberFormat('id-ID').format(data);
}
},{
    data: "kategori.kategori_nama",
    className: "",
    width: "14%",
    orderable: true,
    searchable: false
},{
    data: "aksi",
    className: "text-center",
    width: "14%",
    orderable: false,
    searchable: false
}
]
});

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

$('.filter_kategori').change(function(){
    tableBarang.draw();
});
});
</script>
@endpush
```

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

```
<form action="{{ url('/barang/import_ajax') }}" method="POST" id="form-import"
enctype="multipart/form-data">
@csrf
<div id="modal-master" class="modal-dialog modal-lg" role="document">
    <div class="modal-content">
        <div class="modal-header">
            <h5 class="modal-title" id="exampleModalLabel">Import Data Barang</h5>
            <button type="button" class="close" data-dismiss="modal" aria-
label="Close"><span aria-hidden="true">&times;</span></button>
        </div>
        <div class="modal-body">
            <div class="form-group">
                <label>Download Template</label>
                <a href="{{ asset('template_barang.xlsx') }}" class="btn btn-info btn-
sm" download><i class="fa fa-file-excel"></i>Download</a>
                <small id="error-kategori_id" class="error-text form-text text-
danger"></small>
            </div>
            <div class="form-group">
                <label>Pilih File</label>
                <input type="file" name="file_barang" id="file_barang" class="form-
control" required>
                <small id="error-file_barang" class="error-text form-text text-
danger"></small>
            </div>
        </div>
        <div class="modal-footer">
```



```
        <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
        <button type="submit" class="btn btn-primary">Upload</button>
    </div>
</div>
</div>
</form>
<script>
$(document).ready(function() {
    $("#form-import").validate({
        rules: {
            file_barang: {required: true, extension: "xlsx"},
        },
        submitHandler: function(form) {
            var formData = new FormData(form); // Jadikan form ke FormData untuk
menghandle file

            $.ajax({
                url: form.action,
                type: form.method,
                data: formData, // Data yang dikirim berupa FormData
                processData: false, // setting processData dan contentType ke false,
untuk menghandle file
                contentType: false,
                success: function(response) {
                    if(response.status){ // jika sukses
                        $('#myModal').modal('hide');
                        Swal.fire({
                            icon: 'success',
                            title: 'Berhasil',
                            text: response.message
                        });
                        tableBarang.ajax.reload(); // reload datatable
                    }else{ // jika error
                        $('.error-text').text('');
                        $.each(response.msgField, function(prefix, val) {
                            $('#error-'+prefix).text(val[0]);
                        });
                        Swal.fire({
                            icon: 'error',
                            title: 'Terjadi Kesalahan',
                            text: response.message
                        });
                    }
                }
            });
            return false;
        },
        errorElement: 'span',
        errorPlacement: function (error, element) {
            error.addClass('invalid-feedback');
            element.closest('.form-group').append(error);
        },
        highlight: function (element, errorClass, validClass) {
            $(element).addClass('is-invalid');
        },
        unhighlight: function (element, errorClass, validClass) {
            $(element).removeClass('is-invalid');
        }
    });
});
</script>
```

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



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

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

```
composer require phoffice/phpspreadsheet
```

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

```
<?php  
namespace App\Http\Controllers;  
  
use App\Models\LevelModel;  
use App\Models\BarangModel;  
use App\Models\KategoriModel;  
use Illuminate\Http\Request;  
use Illuminate\Support\Facades\Hash;  
use Illuminate\Support\Facades\Validator;  
use PhpOffice\PhpSpreadsheet\IOFactory;  
use Yajra\DataTables\Facades\DataTables;  
  
class BarangController extends Controller  
{  
  
    public function index (){  
  
        $activeMenu = 'barang';  
        $breadcrumb = (object) [  
            'title' => 'Data Barang',  
            'list' => ['Home', 'Barang']  
        ];  
  
        $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();  
        return view('barang.index', [  
            'activeMenu' => $activeMenu,  
            'breadcrumb' => $breadcrumb,  
            'kategori' => $kategori  
        ]);  
    }  
  
    public function list(Request $request)  
    {  
        $barang = BarangModel::select('barang_id', 'barang_kode', 'barang_nama',  
        'harga_beli', 'harga_jual', 'kategori_id')->with('kategori');  
  
        $kategori_id = $request->input('filter_kategori');  
        if(!empty($kategori_id)){  
            $barang->where('kategori_id', $kategori_id);  
        }  
  
        return DataTables::of($barang)  
            ->addIndexColumn()  
            ->addColumn('aksi', function ($barang) { // menambahkan kolom aksi
```



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

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

public function store_ajax(Request $request)
{
    if($request->ajax() || $request->wantsJson()){
        $rules = [
            'kategori_id' => ['required', 'integer', 'exists:m_kategori,kategori_id'],
            'barang_kode' => ['required', 'min:3', 'max:20',
'unique:m_barang,barang_kode'],
            'barang_nama' => ['required', 'string', 'max:100'],
            'harga_beli' => ['required', 'numeric'],
            'harga_jual' => ['required', 'numeric'],
        ];
        $validator = Validator::make($request->all(), $rules);
        if($validator->fails()){
            return response()->json([
                'status' => false,
                'message' => 'Validasi Gagal',
                'msgField' => $validator->errors()
            ]);
        }
        BarangModel::create($request->all());
        return response()->json([
            'status' => true,
            'message' => 'Data berhasil disimpan'
        ]);
    }
    redirect('/');
}

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

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



```
if ($request->ajax() || $request->wantsJson()) {  
    $rules = [  
        'kategori_id' => ['required', 'integer', 'exists:m_kategori,kategori_id'],  
        'barang_kode' => ['required', 'min:3', 'max:20',  
        'unique:m_barang,barang_kode, . . $id .,barang_id'],  
        'barang_nama' => ['required', 'string', 'max:100'],  
        'harga_beli' => ['required', 'numeric'],  
        'harga_jual' => ['required', 'numeric'],  
    ];  
  
    // use Illuminate\Support\Facades\Validator;  
    $validator = Validator::make($request->all(), $rules);  
    if ($validator->fails()) {  
        return response()->json([  
            'status' => false, // respon json, true: berhasil, false: gagal  
            'message' => 'Validasi gagal.',  
            'msgField' => $validator->errors() // menunjukkan field mana yang error  
        ]);  
    }  
  
    $check = BarangModel::find($id);  
    if ($check) {  
        $check->update($request->all());  
        return response()->json([  
            'status' => true,  
            'message' => 'Data berhasil diupdate'  
        ]);  
    } else{  
        return response()->json([  
            'status' => false,  
            'message' => 'Data tidak ditemukan'  
        ]);  
    }  
    return redirect('/');  
}  
  
public function confirm_ajax($id)  
{  
    $barang = BarangModel::find($id);  
    return view('barang.confirm_ajax', ['barang' => $barang]);  
}  
  
public function delete_ajax(Request $request, $id)  
{  
    if($request->ajax() || $request->wantsJson()){  
        $barang = BarangModel::find($id);  
        if($barang){ // jika sudah ditemukan  
            $barang->delete(); // barang di hapus  
            return response()->json([  
                'status' => true,  
                'message' => 'Data berhasil dihapus'  
            ]);  
        }else{  
            return response()->json([  
                'status' => false,  
                'message' => 'Data tidak ditemukan'  
            ]);  
        }  
    }  
    return redirect('/');  
}  
  
public function import()  
{  
    return view('barang.import');  
}
```



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

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

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

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

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

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

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

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

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



Import Data Barang

Download Template [Download](#)

Pilih File

No file selected.

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

The screenshot shows a web-based application interface titled "PWL - Starter Code". On the left, there is a sidebar with navigation links: Dashboard, Data Pengguna, Level User, Data User, Data Barang, Kategori Barang, Data Barang (which is currently selected), Data Transaksi, Stok Barang, Transaksi Penjualan, and Logout. The main content area is titled "Daftar Barang" and contains a table with 10 rows of data. The columns are: No, Kode Barang, Nama Barang, Harga Beli, Harga Jual, Kategori, and Aksi. The data includes various items like Laptop, Smartphone, Kaos, Jaket, Roti, Kue, Teh Botol, Kopi, Panci, and Sapu, with their respective prices and categories. At the bottom of the table, it says "Showing 1 to 10 of 10 entries". Above the table, there are buttons for Import Barang, Tambah Data, and Tambah Data (Ajax). The footer of the page includes copyright information: "Copyright © 2014-2021 AdminLTE.io. All rights reserved." and "Version 3.2.0".

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	ELK001	Laptop	5.000.000	5.500.000	Elektronik	Detail Edit Hapus
2	ELK002	Smartphone	3.000.000	3.500.000	Elektronik	Detail Edit Hapus
3	PAK001	Kaos	50.000	75.000	Pakaian	Detail Edit Hapus
4	PAK002	Jaket	150.000	200.000	Pakaian	Detail Edit Hapus
5	MAK001	Roti	10.000	15.000	Makanan	Detail Edit Hapus
6	MAK002	Kue	20.000	25.000	Makanan	Detail Edit Hapus
7	MIN001	Teh Botol	5.000	7.000	Minuman	Detail Edit Hapus
8	MIN002	Kopi	10.000	12.000	Minuman	Detail Edit Hapus
9	PRR001	Panci	50.000	75.000	Peralatan Rumahh	Detail Edit Hapus
10	PRR002	Sapu	25.000	30.000	Peralatan Rumahh	Detail Edit Hapus

Import Data Barang

Download Template [Download](#)

Pilih File

template_barang.xlsx



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
11	CML001	Keripik kentang	15.000	18.000	Cemilan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
12	CML002	Popcorn	15.000	17.000	Cemilan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
13	CML003	Biskuit	10.000	13.000	Cemilan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
14	MNR001	Ale-ale	1.500	2.500	Minuman Ringan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Import berhasil

Tugas 1 – Implementasi File Upload untuk Import Data:

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

Tabel _user

Modifikasi pada index.blade



```
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card">
5          <div class="card-header">
6              <h3>Daftar User</h3>
7          <div class="card-tools">
8              <button onclick="modalAction('{{ url('/user/import') }}')" class="btn btn-info">Import User</button>
9              <a href="{{ url('/user/create') }}" class="btn btn-primary">Tambah Data</a>
10             <button onclick="modalAction('{{ url('/user/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
11         </div>
12     </div>
13
14     <div class="card-body">
15         <!-- Filter Data -->
16         <div id="filter" class="form-horizontal filter-data p-2 border-bottom mb-2">
17             <div class="row">
18                 <div class="col-md-12">
19                     <div class="form-group form-group-sm row text-sm mb-0">
20                         <label for="filter_date" class="col-md-1 col-form-label">Filter</label>
21                         <div class="col-md-3">
22                             <select name="filter_level" class="form-control form-control-sm filter_level">
23                                 <option value=""> Semua </option>
24                                 @foreach($level as $l)
25                                     <option value="{{ $l->level_id }}>{{ $l->level_nama }}</option>
26                                 @endforeach
27                             </select>
28                             <small class="form-text text-muted">Level Pengguna</small>
29                         </div>
30                     </div>
31                 </div>
32             </div>
33         </div>
34
35         @if(session('success'))
36             <div class="alert alert-success">{{ session('success') }}</div>
37         @endif
38
39         @if(session('error'))
40             <div class="alert alert-danger">{{ session('error') }}</div>
41         @endif
42
43     <table class="table table-bordered table-sm table-striped table-hover" id="table-user">
44         <thead>
45             <tr>
46                 <th>No.</th>
47                 <th>Username</th>
48                 <th>Nama</th>
49                 <th>Level Pengguna</th>
50                 <th>Aksi</th>
51             </tr>
52         </thead>
53         <tbody></tbody>
54     </table>
55     </div>
56     </div>
57     <div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static" data-keyboard="false" data-width="75%"></div>
58     @endsection
59
60     @push('js')
61         <script>
62             function modalAction(url = '') {
63                 $('#myModal').load(url, function() {
64                     $('#myModal').modal('show');
65                 });
66             }
67
68             var tableUser;
69             $(document).ready(function() {
70                 tableUser = $('#table-user').DataTable({
71                     processing: true,
72                     serverSide: true,
73                     ajax: {
74                         url: "{{ url('user/list') }}",
75                         type: "POST",
76                         dataType: "json",
77                         data: function(d) {
78                             d.filter_level = $('.filter_level').val();
79                         }
80                     },
81                     columns: [
82                         {
83                             data: "DT_RowIndex",
84                             className: "text-center",
85                             width: "5%",
86                             orderable: false,
87                             searchable: false
88                         },
89                         {
90                             data: "username",
91                             className: "",
92                             width: "25%",
93                             orderable: true,
94                             searchable: true
95                         },
96                         {
97                             data: "name",
98                             className: "",
99                             width: "35%",
100                            orderable: true,
101                            searchable: true
102                         },
103                         {
104                             data: "level.level_nama",
105                             className: "",
106                             width: "20%",
107                             orderable: true,
108                             searchable: false
109                         },
110                         {
111                             data: "aksi",
112                             className: "text-center",
113                             width: "15%",
114                             orderable: false,
115                             searchable: false
116                         }
117                     ],
118                 });
119
120                 $('#table_user_filter input').unbind().bind().on('keyup', function(e) {
121                     if (e.keyCode == 13) { // Enter key
122                         tableUser.search(this.value).draw();
123                     }
124                 });
125
126                 $('.filter_level').change(function() {
127                     tableUser.draw();
128                 });
129             });
130         </script>
131     @endpush
```



Membuat view baru import.blade

```
1 <form action="{{ url('/user/import_ajax') }}" method="POST" id="form-import" enctype="multipart/form-data">
2     @csrf
3     <div id="modal-master" class="modal-dialog modal-lg" role="document">
4         <div class="modal-content">
5             <div class="modal-header">
6                 <h5 class="modal-title">Import Data User</h5>
7                 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8                     <span aria-hidden="true"></span>
9                 </button>
10            </div>
11            <div class="modal-body">
12                <div class="form-group">
13                    <label>Download Template</label>
14                    <a href="{{ asset('template_user.xlsx') }}" class="btn btn-info btn-sm" download>
15                        <i class="fa fa-file-excel"></i> Download
16                    </a>
17                    <small id="error-kategori_id" class="error-text form-text text-danger"></small>
18                </div>
19                <div class="form-group">
20                    <label>Pilih File</label>
21                    <input type="file" name="file_user" id="file_user" class="form-control required">
22                    <small id="error-file_user" class="error-text form-text text-danger"></small>
23                </div>
24            </div>
25            <div class="modal-footer">
26                <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
27                <button type="submit" class="btn btn-primary">Upload</button>
28            </div>
29        </div>
30    </form>
31
32
33 <script>
34     $(document).ready(function() {
35         $("#form-import").validate({
36             rules: {
37                 file_user: {
38                     required: true,
39                     extension: "xlsx"
40                 }
41             },
42             submitHandler: function(form) {
43                 var formData = new FormData(form);
44
45                 $.ajax({
46                     url: form.action,
47                     type: form.method,
48                     data: formData,
49                     processData: false,
50                     contentType: false,
51                     success: function(response) {
52                         if (response.status) {
53                             $('#myModal').modal('hide');
54                             Swal.fire({
55                                 icon: 'success',
56                                 title: 'Berhasil',
57                                 text: response.message
58                             });
59                             tableUser.ajax.reload();
60                         } else {
61                             $('.error-text').text('');
62                             $.each(response.msgField, function(prefix, val) {
63                                 $('#error-' + prefix).text(val[0]);
64                             });
65                             Swal.fire({
66                                 icon: 'error',
67                                 title: 'Terjadi Kesalahan',
68                                 text: response.message
69                             });
70                         }
71                     },
72                     error: function(xhr, status, error) {
73                         Swal.fire({
74                             icon: 'error',
75                             title: 'Error AJAX',
76                             text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
77                         });
78                         console.log('Error:', xhr.responseText);
79                     }
80                 });
81                 return false;
82             },
83             errorElement: 'span',
84             errorPlacement: function(error, element) {
85                 error.addClass('invalid-feedback');
86                 element.closest('.form-group').append(error);
87             },
88             highlight: function(element) {
89                 $element.addClass('is-invalid');
90             },
91             unhighlight: function(element) {
92                 $element.removeClass('is-invalid');
93             }
94         });
95     </script>
```



Mofifikasi pada userContoller menambahkan fungsi import

```
 1  public function import()
 2  {
 3      return view('user.import');
 4  }
 5
 6  public function import_ajax(Request $request)
 7  {
 8      if ($request->ajax() || $request->wantsJson()) {
 9          $rules = [
10              'file_user' => ['required', 'mimes:xlsx', 'max:1024']
11          ];
12
13          $validator = Validator::make($request->all(), $rules);
14          if ($validator->fails()) {
15              return response()->json([
16                  'status' => false,
17                  'message' => 'Validasi Gagal',
18                  'msgField' => $validator->errors()
19              ]);
20          }
21
22          $file = $request->file('file_user');
23
24          $reader = IOFactory::createReader('Xlsx');
25          $reader->setReadDataOnly(true);
26          $spreadsheet = $reader->load($file->getRealPath());
27          $sheet = $spreadsheet->getActiveSheet();
28
29          $data = $sheet->toArray(null, false, true, true);
30
31          $insert = [];
32          if (count($data) > 1) {
33              foreach ($data as $baris => $value) {
34                  if ($baris > 1) {
35                      $insert[] = [
36                          'level_id' => $value['A'],
37                          'username' => $value['B'],
38                          'nama' => $value['C'],
39                          'password' => bcrypt($value['D']),
40                          'created_at' => now(),
41                      ];
42                  }
43              }
44
45              if (count($insert) > 0) {
46                  UserModel::insertOrIgnore($insert);
47              }
48
49              return response()->json([
50                  'status' => true,
51                  'message' => 'Data user berhasil diimport'
52              ]);
53          } else {
54              return response()->json([
55                  'status' => false,
56                  'message' => 'Tidak ada data yang diimport'
57              ]);
58          }
59      }
60      return redirect('/');
61  }
```



Menambahkan route untuk import

```
Route::get('/import', [UserController::class, 'import']);  
Route::post('/import_ajax', [UserController::class, 'import_ajax']);
```

implementasi

No	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrasi	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	Manager11	Manager111	Manager	[Detail] [Edit] [Hapus]
9	Rusgart	Bayu Triwibowo	Manager	[Detail] [Edit] [Hapus]

Import Data User

Download Template [Download](#)

Pilih File

Choose File template_user.xlsx

Batal Upload

10	Artemis	Adam	Staff/Kasir	Detail
11	Asmodeus	Nicho	Staff/Kasir	Detail

Tabel m_kategori

Modifikasi pada index.blade



```
1  @extends('layouts.template')
2
3  @section('content')
4      <div class="card">
5          <div class="card-header">
6              <h3 class="card-title">Daftar Kategori</h3>
7              <div class="card-tools">
8                  <button onclick="modalAction('{{ url('/kategori/import') }}')" class="btn btn-info">Import Kategori</button>
9                  <button onclick="modalAction('{{ url('/kategori/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
10             </div>
11         </div>
12
13         <div class="card-body">
14             @if(session('success'))
15                 <div class="alert alert-success">{{ session('success') }}</div>
16             @endif
17
18             @if(session('error'))
19                 <div class="alert alert-danger">{{ session('error') }}</div>
20             @endif
21
22             <table class="table table-bordered table-sm table-striped table-hover" id="table-kategori">
23                 <thead>
24                     <tr>
25                         <th>No</th>
26                         <th>Kode Kategori</th>
27                         <th>Nama Kategori</th>
28                         <th>Aksi</th>
29                     </tr>
30                 </thead>
31                 <tbody></tbody>
32             </table>
33         </div>
34     <div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static" data-keyboard="false" data-width="75%"></div>
35     @endsection
36
37     @push('js')
38         <script>
39             function modalAction(url = '') {
40                 $('#myModal').load(url, function() {
41                     $('#myModal').modal('show');
42                 });
43             }
44
45             var tableKategori;
46             $(document).ready(function() {
47                 tableKategori = $('#table-kategori').DataTable({
48                     processing: true,
49                     serverSide: true,
50                     ajax: {
51                         url: "{{ url('/kategori/list') }}",
52                         type: "POST",
53                         dataType: "json"
54                     },
55                     columns: [
56                         {
57                             data: "DT_RowIndex",
58                             className: "text-center",
59                             width: "5%",
60                             orderable: false,
61                             searchable: false
62                         },
63                         {
64                             data: "kategori_kode",
65                             className: "",
66                             width: "30%",
67                             orderable: true,
68                             searchable: true
69                         },
70                         {
71                             data: "kategori_nama",
72                             className: "",
73                             width: "45%",
74                             orderable: true,
75                             searchable: true
76                         },
77                         {
78                             data: "aksi",
79                             className: "text-center",
80                             width: "20%",
81                             orderable: false,
82                             searchable: false
83                         }
84                     ]
85                 });
86             });
87             $('#table-kategori_filter input').unbind().bind().on('keyup', function(e) {
88                 if (e.keyCode == 13) { // Enter key
89                     tableKategori.search(this.value).draw();
90                 }
91             });
92         });
93     </script>
94     @endpush
```



Membuat view import.blade

```
● ○ ●
1 <form action="{{ url('/kategori/import_ajax') }}" method="POST" id="form-import" enctype="multipart/form-data">
2     @csrf
3     <div id="modal-master" class="modal-dialog modal-lg" role="document">
4         <div class="modal-content">
5             <div class="modal-header">
6                 <h5 class="modal-title">Import Data Kategori</h5>
7                 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8                     <span aria-hidden="true" style="color: #ccc;">×
9                 </button>
10            </div>
11            <div class="modal-body">
12                <div class="form-group">
13                    <label>Download Template</label>
14                    <a href="{{ asset('template_kategori.xlsx') }}" class="btn btn-info btn-sm" download>
15                        <i class="fa fa-file-excel"></i> Download
16                    </a>
17                </div>
18                <div class="form-group">
19                    <label>Pilih File:</label>
20                    <input type="file" name="file_kategori" id="file_kategori" class="form-control" required>
21                    <small id="error-file_kategori" class="error-text form-text text-danger"></small>
22                </div>
23            </div>
24            <div class="modal-footer">
25                <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
26                <button type="submit" class="btn btn-primary">Upload</button>
27            </div>
28        </div>
29    </div>
30 </form>
31
32 <script>
33     $(document).ready(function() {
34         $("#form-import").validate({
35             rules: {
36                 file_kategori: {
37                     required: true,
38                     extension: "xlsx"
39                 }
40             },
41             submitHandler: function(form) {
42                 var formData = new FormData(form);
43
44                 $.ajax({
45                     url: form.action,
46                     type: form.method,
47                     data: formData,
48                     processData: false,
49                     contentType: false,
50                     success: function(response) {
51                         if (response.status) {
52                             $('#myModal').modal('hide');
53                             Swal.fire({
54                                 icon: 'success',
55                                 title: 'Berhasil',
56                                 text: response.message
57                             });
58                             tableKategori.ajax.reload();
59                         } else {
60                             $('.error-text').text('');
61                             $.each(response.msgField, function(prefix, val) {
62                                 $('#error-' + prefix).text(val[0]);
63                             });
64                             Swal.fire({
65                                 icon: 'error',
66                                 title: 'Terjadi Kesalahan',
67                                 text: response.message
68                             });
69                         }
70                     },
71                     error: function(xhr, status, error) {
72                         Swal.fire({
73                             icon: 'error',
74                             title: 'Error AJAX',
75                             text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
76                         });
77                         console.log('Error:', xhr.responseText);
78                     }
79                 });
80                 return false;
81             },
82             errorElement: 'span',
83             errorPlacement: function(error, element) {
84                 error.addClass('invalid-feedback');
85                 element.closest('.form-group').append(error);
86             },
87             highlight: function(element) {
88                 $(element).addClass('is-invalid');
89             },
90             unhighlight: function(element) {
91                 $(element).removeClass('is-invalid');
92             }
93         });
94     });
95 </script>
```



Modifikasi pada kategoriController menambahkan fungsi import

```
● ● ●

1 public function import()
2 {
3     return view('kategori.import');
4 }
5
6 public function import_ajax(Request $request)
7 {
8     if ($request->ajax() || $request->wantsJson()) {
9         $rules = [
10             'file_kategori' => ['required', 'mimes:xlsx', 'max:1024']
11         ];
12
13         $validator = Validator::make($request->all(), $rules);
14         if ($validator->fails()) {
15             return response()->json([
16                 'status' => false,
17                 'message' => 'Validasi Gagal',
18                 'msgField' => $validator->errors()
19             ]);
20         }
21
22         $file = $request->file('file_kategori');
23
24         $reader = IOFactory::createReader('Xlsx');
25         $reader->setReadDataOnly(true);
26         $spreadsheet = $reader->load($file->getRealPath());
27         $sheet = $spreadsheet->getActiveSheet();
28         $data = $sheet->toArray(null, false, true, true);
29
30         $insert = [];
31         if (count($data) > 1) {
32             foreach ($data as $baris => $value) {
33                 if ($baris > 1) {
34                     $insert[] = [
35                         'kategori_kode' => $value['A'],
36                         'kategori_nama' => $value['B'],
37                         'created_at' => now(),
38                     ];
39                 }
40             }
41
42             if (count($insert) > 0) {
43                 KategoriModel::insertOrIgnore($insert);
44             }
45
46             return response()->json([
47                 'status' => true,
48                 'message' => 'Data kategori berhasil diimport'
49             ]);
50         } else {
51             return response()->json([
52                 'status' => false,
53                 'message' => 'Tidak ada data yang diimport'
54             ]);
55         }
56     }
57     return redirect('/');
58 }
```



Menambahkan route untuk import

```
Route::get('/import', [KategoriController::class, 'import']);  
Route::post('/import_ajax', [KategoriController::class, 'import_ajax']);
```

Implementasi

No	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	[Detail] [Edit] [Hapus]
2	PAK	Pakaihan	[Detail] [Edit] [Hapus]
3	MAK	Makanan	[Detail] [Edit] [Hapus]
4	MIN	Minuman	[Detail] [Edit] [Hapus]
5	PRR	Peralatan Rumahh	[Detail] [Edit] [Hapus]
6	CML	Cemilan	[Detail] [Edit] [Hapus]
7	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]

Import Data Kategori

Download Template [Download](#)

Pilih File

Choose File template_kategori.xlsx

Batal Upload

No	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	[Detail] [Edit] [Hapus]
2	PAK	Pakaihan	[Detail] [Edit] [Hapus]
3	MAK	Makanan	[Detail] [Edit] [Hapus]
4	MIN	Minuman	[Detail] [Edit] [Hapus]
5	PRR	Peralatan Rumahh	[Detail] [Edit] [Hapus]
6	CML	Cemilan	[Detail] [Edit] [Hapus]
7	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]
8	S	Bahan Pokok	[Detail] [Edit] [Hapus]
9	9	Obat-obatan	[Detail] [Edit] [Hapus]



Tabel m_level

Memodifikasi index.blade

```
1 @extends('layouts.template')
2
3 @section('content')
4     <div class="card">
5         <div class="card-header">
6             <h3>Daftar Level</h3>
7             <div class="card-tools">
8                 <button onclick="modalAction('{{ url('/level/import') }}'" class="btn btn-info">Import Level</button>
9                 <button onclick="modalAction('{{ url('/level/create_ajax') }}'" class="btn btn-success">Tambah Data (Ajax)</button>
10            </div>
11        </div>
12
13        <div class="card-body">
14            @if(session('success'))
15                <div class="alert alert-success">{{ session('success') }}</div>
16            @endif
17
18            @if(session('error'))
19                <div class="alert alert-danger">{{ session('error') }}</div>
20            @endif
21
22            <table class="table table-bordered table-sm table-striped table-hover" id="table-level">
23                <thead>
24                    <tr>
25                        <th>No</th>
26                        <th>Kode Level</th>
27                        <th>Nama Level</th>
28                        <th>Aksi</th>
29                    </tr>
30                </thead>
31                <tbody></tbody>
32            </div>
33        </div>
34        <div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static" data-keyboard="false" data-width="75%"></div>
35    @endsection
36
37    @push('js')
38        <script>
39            function modalAction(url = '') {
40                $('#myModal').load(url, function() {
41                    $('#myModal').modal('show');
42                });
43            }
44        }
45
46        var tableLevel;
47        $(document).ready(function() {
48            tableLevel = $('#table-level').DataTable({
49                processing: true,
50                serverSide: true,
51                ajax: {
52                    url: "{{ url('level/list') }}",
53                    type: "POST",
54                    dataType: "json"
55                },
56                columns: [
57                    {
58                        data: "DT_RowIndex",
59                        className: "text-center",
60                        width: "5%",
61                        orderable: false,
62                        searchable: false
63                    },
64                    {
65                        data: "level_kode",
66                        className: "",
67                        width: "30%",
68                        orderable: true,
69                        searchable: true
70                    },
71                    {
72                        data: "level_nama",
73                        className: "",
74                        width: "45%",
75                        orderable: true,
76                        searchable: true
77                    },
78                    {
79                        data: "aksi",
80                        className: "text-center",
81                        width: "20%",
82                        orderable: false,
83                        searchable: false
84                    }
85                ]
86            });
87
88            $('#table-level_filter input').unbind().bind().on('keyup', function(e) {
89                if (e.keyCode == 13) {
90                    tableLevel.search(this.value).draw();
91                }
92            });
93        });
94    @endpush
95
```



Membuat view import.blade

```
● ○ ●
1 <form action="{{ url('/level/import_ajax') }}" method="POST" id="form-import" enctype="multipart/form-data">
2     @csrf
3     <div id="modal-master" class="modal-dialog modal-lg" role="document">
4         <div class="modal-content">
5             <div class="modal-header">
6                 <h5 class="modal-title">Import Data Level</h5>
7                 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
8                     <span aria-hidden="true"></span>
9                 </button>
10            </div>
11            <div class="modal-body">
12                <div class="form-group">
13                    <label>Download Template</label>
14                    <a href="{{ asset('template_level.xlsx') }}" class="btn btn-info btn-sm" download>
15                        <i class="fa fa-file-excel"></i> Download
16                    </a>
17                </div>
18                <div class="form-group">
19                    <label>Pilih File</label>
20                    <input type="file" name="file_level" id="file_level" class="form-control" required>
21                    <small id="error-file_level" class="error-text form-text text-danger"></small>
22                </div>
23            </div>
24            <div class="modal-footer">
25                <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
26                <button type="submit" class="btn btn-primary">Upload</button>
27            </div>
28        </div>
29    </div>
30 </form>
31
32 <script>
33     $(document).ready(function() {
34         $("#form-import").validate({
35             rules: {
36                 file_level: {
37                     required: true,
38                     extension: "xlsx"
39                 }
40             },
41             submitHandler: function(form) {
42                 var formData = new FormData(form);
43
44                 $.ajax({
45                     url: form.action,
46                     type: form.method,
47                     data: formData,
48                     processData: false,
49                     contentType: false,
50                     success: function(response) {
51                         if (response.status) {
52                             $('#myModal').modal('hide');
53                             Swal.fire({
54                                 icon: 'success',
55                                 title: 'Berhasil',
56                                 text: response.message
57                             });
58                             tableLevel.ajax.reload();
59                         } else {
56
57                             $('.error-text').text('');
58                             $.each(response.msgField, function(prefix, val) {
59                                 $('#erron-' + prefix).text(val[0]);
60                             });
61                             Swal.fire({
62                                 icon: 'error',
63                                 title: 'Terjadi Kesalahan',
64                                 text: response.message
65                             });
66                         }
67                     },
68                 });
69             },
70             error: function(xhr, status, error) {
71                 Swal.fire({
72                     icon: 'error',
73                     title: 'Error AJAX',
74                     text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
75                 });
76                 console.log('Error:', xhr.responseText);
77             }
78         });
79     });
80     return false;
81 },
82     errorElement: 'span',
83     errorPlacement: function(error, element) {
84         error.addClass('invalid-feedback');
85         element.closest('.form-group').append(error);
86     },
87     highlight: function(element) {
88         $element.addClass('is-invalid');
89     },
90     unhighlight: function(element) {
91         $element.removeClass('is-invalid');
92     }
93 });
94 });
95 </script>
```



Menambahkan fungsi baru pada controller

```
 1  public function import()
 2  {
 3      return view('level.import');
 4  }
 5
 6  public function import_ajax(Request $request)
 7  {
 8      if ($request->ajax() || $request->wantsJson()) {
 9          $rules = [
10              'file_level' => ['required', 'mimes:xlsx', 'max:1024']
11          ];
12
13          $validator = Validator::make($request->all(), $rules);
14          if ($validator->fails()) {
15              return response()->json([
16                  'status' => false,
17                  'message' => 'Validasi Gagal',
18                  'msgField' => $validator->errors()
19              ]);
20          }
21
22          $file = $request->file('file_level');
23
24          $reader = IOFactory::createReader('Xlsx');
25          $reader->setReadDataOnly(true);
26          $spreadsheet = $reader->load($file->getRealPath());
27          $sheet = $spreadsheet->getActiveSheet();
28          $data = $sheet->toArray(null, false, true, true);
29
30          $insert = [];
31          if (count($data) > 1) {
32              foreach ($data as $baris => $value) {
33                  if ($baris > 1) {
34                      $insert[] = [
35                          'level_kode' => $value['A'],
36                          'level_nama' => $value['B'],
37                          'created_at' => now(),
38                      ];
39                  }
40              }
41
42              if (count($insert) > 0) {
43                  LevelModel::insertOrIgnore($insert);
44              }
45
46              return response()->json([
47                  'status' => true,
48                  'message' => 'Data level berhasil diimport'
49              ]);
50          } else {
51              return response()->json([
52                  'status' => false,
53                  'message' => 'Tidak ada data yang diimport'
54              ]);
55          }
56      }
57      return redirect('/');
58  }
```



Menambahkan route untuk import

```
Route::get('/import', [LevelController::class, 'import']);  
Route::post('/import_ajax', [LevelController::class, 'import_ajax']);
```

Implementasi

Import Data Level

Download Template [Download](#)

Pilih File

Choose File template_level.xlsx

Batal Upload

Daftar Level

No	Kode Level	Nama Level	Aksi
1	ADM	Administrasi	Detail Edit Hapus
2	MNG	Manager	Detail Edit Hapus
3	STF	Staff/Kasir	Detail Edit Hapus
4	SPR	Supervisor	Detail Edit Hapus
5	OWN	Owner	Detail Edit Hapus

Tabel t_stok

Memodifikasi index.blade



```
1 <!-- resources/views/stok/index.blade.php -->
2 @extends('layouts.template')
3
4 @section('content')
5   <div class="card">
6     <div class="card-header">
7       <h3>Daftar Stok Barang</h3>
8       <div class="card-tools">
9         <button onclick="modalAction('{{ url('/stok/import') }}')" class="btn btn-info">Import Stok</button>
10        <button onclick="modalAction('{{ url('/stok/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
11      </div>
12    </div>
13
14    <div class="card-body">
15      <div class="row">
16        <div class="col-12 col-md-12">
17          <div class="form-group row">
18            <div class="col-3 control-label col-form-label">Filter:</div>
19            <div class="col-3">
20              <select class="form-control" id="user_id" name="user_id">
21                <option value=""> Semua</option>
22                @foreach ($user as $item)
23                  <option value="{{ $item->id }}">{{ $item->name }}</option>
24                @endforeach
25              </select>
26              <small class="form-text text-muted">Nama User</small>
27            </div>
28          </div>
29        </div>
30        @if(session('success'))
31          <div class="alert alert-success">{{ session('success') }}</div>
32        @endif
33
34        @if(session('error'))
35          <div class="alert alert-danger">{{ session('error') }}</div>
36        @endif
37
38        <table class="table table-bordered table-sm table-striped table-hover" id="table-stok">
39          <thead>
40            <tr>
41              <th>ID</th>
42              <th>Nama Barang</th>
43              <th>Nama User</th>
44              <th>Tanggal Stok</th>
45              <th>Jumlah Stok</th>
46              <th>Aksi</th>
47            </tr>
48          </thead>
49          <tbody></tbody>
50        </table>
51      </div>
52    </div>
53    <div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static" data-keyboard="false" data-width="75%></div>
54  @endsection
55
56  @push('js')
57    <script>
58      function modalAction(url) {
59        $('#myModal').load(url, function() {
60          $('#myModal').modal('show');
61        });
62      }
63    </script>
64
65    var tableStok;
66    $(document).ready(function() {
67      tableStok = $('#table-stok').DataTable({
68        processing: true,
69        serverSide: true,
70        ajax: {
71          url: "{{ url('stok/list') }}",
72          type: "POST",
73          dataType: "json",
74          data: function(d) {
75            d.filter_barang = $('.filter_barang').val();
76          }
77        },
78        columns: [
79          {
80            data: 'Df_RowIndex',
81            className: "text-center",
82            width: "25",
83            orderable: false,
84            searchable: false
85          },
86          {
87            data: 'barang.barang_nama',
88            className: "",
89            width: "25",
90            orderable: true,
91            searchable: true
92          },
93          {
94            data: 'user.nama',
95            className: "",
96            width: "25",
97            orderable: false,
98            searchable: false
99          },
100          {
101            data: 'stok_tanggal',
102            className: "",
103            width: "20%",
104            orderable: true,
105            searchable: true
106          },
107          {
108            data: 'stok_jumlah',
109            className: "text-center",
110            width: "20%",
111            orderable: false,
112            searchable: false
113          },
114          {
115            data: 'aksi',
116            className: "text-center",
117            width: "20%",
118            orderable: false,
119            searchable: false
120          }
121        ],
122      );
123
124      $('#table-stok .filter input').unbind().bind().on('keyup', function(e) {
125        if (e.keyCode == 13) {
126          tableStok.search(this.value).draw();
127        }
128      });
129
130      $('.filter_barang').change(function() {
131        tableStok.draw();
132      });
133    });
134  </script>
135  @endpush
```



Membuatkan view baru import.blade

```
1 <!-- resources/views/stok/import_ajax.blade.php -->
2 <form action="{{ url('stok/import_ajax') }}" method="POST" id="form-import" enctype="multipart/form-data">
3     @csrf
4     <div id="modal-master" class="modal-dialog modal-lg" role="document">
5         <div class="modal-content">
6             <div class="modal-header">
7                 <h5 class="modal-title">Import Data Stok</h5>
8                 <button type="button" class="close" data-dismiss="modal" aria-label="Close">
9                     <span aria-hidden="true"></span>
10                </button>
11            </div>
12            <div class="modal-body">
13                <div class="form-group">
14                    <label>Download Template</label>
15                    <a href="{{ asset('template_stok.xlsx') }}" class="btn btn-info btn-sm" download>
16                        <i class="fa fa-file-excel"></i> Download
17                    </a>
18                </div>
19                <div class="form-group">
20                    <label>Pilih File</label>
21                    <input type="file" name="file_stok" id="file_stok" class="form-control" required>
22                    <small id="error-file_stok" class="error-text form-text text-danger"></small>
23                </div>
24            </div>
25            <div class="modal-footer">
26                <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
27                <button type="submit" class="btn btn-primary">Upload</button>
28            </div>
29        </div>
30    </form>
31
32
33 <script>
34     $(document).ready(function() {
35         $("#form-import").validate({
36             rules: {
37                 file_stok: {
38                     required: true,
39                     extension: "xlsx"
40                 }
41             },
42             submitHandler: function(form) {
43                 var formData = new FormData(form);
44
45                 $.ajax({
46                     url: form.action,
47                     type: form.method,
48                     data: formData,
49                     processData: false,
50                     contentType: false,
51                     success: function(response) {
52                         if (response.status) {
53                             $("#myModal").modal('hide');
54                             Swal.fire({
55                                 icon: 'success',
56                                 title: 'Berhasil',
57                                 text: response.message
58                             });
59                             tableStok.ajax.reload();
60                         } else {
61                             $('.error-text').text('');
62                             $.each(response.msgfield, function(prefix, val) {
63                                 $('#error-' + prefix).text(val[0]);
64                             });
65                             Swal.fire({
66                                 icon: 'error',
67                                 title: 'Terjadi Kesalahan',
68                                 text: response.message
69                             });
70                         }
71                     },
72                     error: function(xhr, status, error) {
73                         Swal.fire({
74                             icon: 'error',
75                             title: 'Error AJAX',
76                             text: 'Terjadi kesalahan: ' + xhr.status + ' - ' + error
77                         });
78                         console.log('Error:', xhr.responseText);
79                     }
80                 });
81                 return false;
82             },
83             errorElement: 'span',
84             errorPlacement: function(error, element) {
85                 error.addClass('invalid-feedback');
86                 element.closest('.form-group').append(error);
87             },
88             highlight: function(element) {
89                 $(element).addClass('is-invalid');
90             },
91             unhighlight: function(element) {
92                 $(element).removeClass('is-invalid');
93             }
94         });
95     });
96 </script>
```



Menambahkan fungsi baru pada controller

```
1 public function import()
2 {
3     return view('stok.import');
4 }
5
6 public function import_ajax(Request $request)
7 {
8     if ($request->ajax() || $request->wantsJson()) {
9         $rules = [
10             'file_stok' => ['required', 'mimes:xlsx', 'max:1024']
11         ];
12
13         $validator = Validator::make($request->all(), $rules);
14         if ($validator->fails()) {
15             return response()->json([
16                 'status' => false,
17                 'message' => 'Validasi Gagal',
18                 'msgField' => $validator->errors()
19             ]);
20         }
21
22         $file = $request->file('file_stok');
23
24         $reader = IOFactory::createReader('Xlsx');
25         $reader->setReadDataOnly(true);
26         $spreadsheet = $reader->load($file->getRealPath());
27         $sheet = $spreadsheet->getActiveSheet();
28         $data = $sheet->toArray(null, false, true);
29
30         $insert = [];
31         if (count($data) > 1) {
32             foreach ($data as $baris => $value) {
33                 if ($baris > 1) {
34                     $insert[] = [
35                         'barang_id' => $value['A'],
36                         'user_id' => $value['B'],
37                         'stok_tanggal' => $value['C'],
38                         'stok_jumlah' => $value['D'],
39                         'created_at' => now(),
40                     ];
41                 }
42             }
43
44             if (count($insert) > 0) {
45                 StokModel::insertOrIgnore($insert);
46             }
47
48             return response()->json([
49                 'status' => true,
50                 'message' => 'Data stok berhasil diimport'
51             ]);
52         } else {
53             return response()->json([
54                 'status' => false,
55                 'message' => 'Tidak ada data yang diimport'
56             ]);
57         }
58     }
59     return redirect('/');
60 }
```



Menambahkan route untuk import

```
Route::get('/import', [StokController::class, 'import']);  
Route::post('/import_ajax', [StokController::class, 'import_ajax']);
```

Implementasi

ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
11	Panci	Staff/Kasir	0000-00-00 00:00:00	35	Detail Edit Hapus
12	Sapu	Staff/Kasir	0000-00-00 00:00:00	25	Detail Edit Hapus

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - Memodifikasi pada index untuk menjalankan import
 - Membuatkan view baru import.blade untuk menjalankan import
 - Menambahkan function baru untuk import
 - Menambahkan route untuk import
3. Submit kode untuk impementasi praktkikum 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.



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

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

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

Kita ubah dengan

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



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

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

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

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

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

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

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

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

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

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

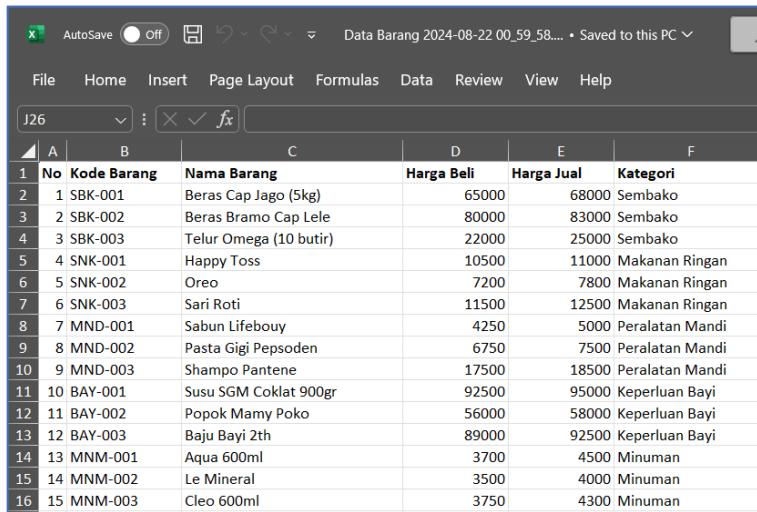
```
266 |     foreach(range('A','F') as $columnID) {
267 |         $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size untuk kolom
268 |     }
```



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

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

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



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

Hasilnya



Screenshot of a web application showing a list of items in a table. The table has columns: No, Kode Barang, Nama Barang, Harga Beli, Harga Jual, Kategori, and Aksi (Detail, Edit, Hapus). The data is as follows:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	ELK001	Laptop	5.000.000	5.500.000	Elektronik	[Detail] [Edit] [Hapus]
2	ELK002	Smartphone	3.000.000	3.500.000	Elektronik	[Detail] [Edit] [Hapus]
3	PAK001	Kaos	50.000	75.000	Pakaian	[Detail] [Edit] [Hapus]
4	PAK002	Jaket	150.000	200.000	Pakaian	[Detail] [Edit] [Hapus]
5	MAK001	Roti	10.000	15.000	Makanan	[Detail] [Edit] [Hapus]
6	MAK002	Kue	20.000	25.000	Makanan	[Detail] [Edit] [Hapus]
7	MIN001	Teh Botol	5.000	7.000	Minuman	[Detail] [Edit] [Hapus]
8	MIN002	Kopi	10.000	12.000	Minuman	[Detail] [Edit] [Hapus]
9	PRR001	Panci	50.000	75.000	Peralatan Rumahh	[Detail] [Edit] [Hapus]
10	PRR002	Sapu	25.000	30.000	Peralatan Rumahh	[Detail] [Edit] [Hapus]

Screenshot of a web application showing a list of items in a table. The table has columns: No, Kode Barang, Nama Barang, Harga Beli, Harga Jual, Kategori, and Aksi (Detail, Edit, Hapus). The data is as follows:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
11	CML001	Keripik kentang	15.000	18.000	Cemilan	[Detail] [Edit] [Hapus]
12	CML002	Popcorn	15.000	17.000	Cemilan	[Detail] [Edit] [Hapus]
13	CML003	Biskuit	10.000	13.000	Cemilan	[Detail] [Edit] [Hapus]
14	MNR001	Ale-ale	1.500	2.500	Minuman Ringan	[Detail] [Edit] [Hapus]

1	No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
2	1	ELK001	Laptop	5000000	5500000	Elektronik
3	2	ELK002	Smartphone	3000000	3500000	Elektronik
4	3	PAK001	Kaos	50000	75000	Pakaian
5	4	PAK002	Jaket	150000	200000	Pakaian
6	5	MAK001	Roti	10000	15000	Makanan
7	6	MAK002	Kue	20000	25000	Makanan
8	7	MIN001	Teh Botol	5000	7000	Minuman
9	8	MIN002	Kopi	10000	12000	Minuman
10	9	PRR001	Panci	50000	75000	Peralatan Rumahh
11	10	PRR002	Sapu	25000	30000	Peralatan Rumahh
12	11	CML001	Keripik kentang	15000	18000	Cemilan
13	12	CML002	Popcorn	15000	17000	Cemilan
14	13	CML003	Biskuit	10000	13000	Cemilan
15	14	MNR001	Ale-ale	1500	2500	Minuman Ringan

Tugas 2 – Implementasi File Export Excel:

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



menu

Tabel m_user

Memodifikasi tombol untuk export

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

Menambahkan function export pada controller

```
public function export_excel()
{
    $user = UserModel::select('level_id', 'username', 'nama', 'password')
        ->orderBy('level_id')
        ->with('level')
        ->get();

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

    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'Username');
    $sheet->setCellValue('C1', 'Nama');
    $sheet->setCellValue('D1', 'Password');
    $sheet->setCellValue('E1', 'Level Pengguna');

    $sheet->getStyle('A1:E1')->getFont()->setBold(true);

    $no = 1;
    $baris = 2;
    foreach ($user as $key => $value) {
        $sheet->setCellValue('A'.$baris, $no);
        $sheet->setCellValue('B'.$baris, $value->username);
        $sheet->setCellValue('C'.$baris, $value->nama);
        $sheet->setCellValue('D'.$baris, $value->password);
        $sheet->setCellValue('E'.$baris, $value->level->level_nama);
        $baris++;
        $no++;
    }

    foreach (range('A', 'E') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    $sheet->setTitle('Data User');

    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $filename = 'Data User '.date('Y-m-d H:i:s').'.xlsx';

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

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

Menambahkan route untuk export

```
Route::get('/export_excel', [UserController::class, 'export_excel']);
});
```



Screenshot of a web application interface titled "PWL - Starter Code". The main page shows a table of users with columns: No, Username, Name, Level Pengguna, and Aksi (Actions). The "Data User" menu item is highlighted in blue. The table data is as follows:

No	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrasi	[Detail] [Edit] [Hapus]
2	manager	Manager	Manager	[Detail] [Edit] [Hapus]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Hapus]
4	customer-1	Pelanggan Pertama	Staff/Kasir	[Detail] [Edit] [Hapus]
5	manager33	Manager Tiga Tiga	Manager	[Detail] [Edit] [Hapus]
6	manager56	Manager55	Manager	[Detail] [Edit] [Hapus]
7	manager12	Manager11	Manager	[Detail] [Edit] [Hapus]
8	Manager11	Manager11	Manager	[Detail] [Edit] [Hapus]
9	Rusgart	Bayu Triwibowo	Manager	[Detail] [Edit] [Hapus]
10	Artemis	Adam	Staff/Kasir	[Detail] [Edit] [Hapus]
11	Asmodeus	Nicho	Staff/Kasir	[Detail] [Edit] [Hapus]

Below the main table, there is a smaller table showing details for user ID 11:

No	Username	Nama	Level Pengguna	Aksi
11	Asmodeus	Nicho	Staff/Kasir	[Detail] [Edit] [Hapus]

At the bottom, there is a large table containing all user data again:

No	Username	Nama	Level Pengguna	Level Pengguna
2	1 admin	Administrator	\$2y\$12\$7ITX8kFP2pT/uUsW.9Z/7ezRt5wwT0kGD92f/D/3mOhCvVV1Cbz56	Administrasi
3	2 manager	Manager	\$2y\$12\$Zqat/wNKOJ7TNdGCgo0xSe2/WxNazN2iHnShMAH77A.i3NoMiq.a	Manager
4	3 manager33	Manager Tiga Tiga	\$2y\$12\$qElinnq5Onva0mlJ.JEim.hE3brimXWtLg0PYi5OA0SqUSQg38LW	Manager
5	4 manager56	Manager55	\$2y\$12\$kgzh13.4ZB49eMTJt8droOWvbboVtD3qWOA7e3Mi3d.w1aSK./HNPO	Manager
6	5 manager12	Manager11	\$2y\$12\$t6g3UbkGrE4XYe23iZ2wze9r8bYGdjJBtr5a/BvloCa3dKFgiVEyG	Manager
7	6 Manager11	Manager111		123456 Manager
8	7 Rusgart	Bayu Triwibowo	\$2y\$12\$SFFk0Mp8U9dzlpwuthyq0Oij8asUtcMWPx79Q4.jtvUVhSZPCDOwC	Manager
9	8 staff	Staff/Kasir	\$2y\$12\$Nn7aSo0AWhr.4pyiG1/6zON2zUZm6i6QpkP4FVKHDG14ZtTs2gc2u	Staff/Kasir
10	9 customer-1	Pelanggan Pertama	\$2y\$12\$cKHuQKpfqb2CbMWgnNcBGepKAV1126GVcOtYoiUUNToCmpHgRXSEK	Staff/Kasir
11	10 Artemis	Adam	\$2y\$12\$zOb0gGIHNV/.fbpJac5C9OT6h0Vnc6fDU4gbT14nUDz9mU4YIK3m6	Staff/Kasir
12	11 Asmodeus	Nicho	\$2y\$12\$L3KzqTl3rOO7A3.fe/fjOPDwg6MWZABg/Mw40YzKhscCrV2lusPK	Staff/Kasir

Tabel m_level

Memodifikasi tombol untuk export

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

Menambahkan function export pada controller



```
public function export_excel()
{
    $level = LevelModel::select('level_kode', 'level_nama')
        ->orderBy('level_kode')
        ->get();

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

    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'Kode Level');
    $sheet->setCellValue('C1', 'Nama Level');

    $sheet->getStyle('A1:C1')->getFont()->setBold(true);

    $no = 1;
    $baris = 2;
    foreach ($level as $key => $value) {
        $sheet->setCellValue('A'.$baris, $no);
        $sheet->setCellValue('B'.$baris, $value->level_kode);
        $sheet->setCellValue('C'.$baris, $value->level_nama);
        $baris++;
        $no++;
    }

    foreach (range('A', 'C') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    $sheet->setTitle('Data Level');

    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $filename = 'Data Level '.date('Y-m-d H:i:s').'.xlsx';

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

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

Menambahkan route untuk export

```
Route::get('/export_excel', [LevelController::class, 'export_excel']);
```



No	Kode Level	Nama Level
2	1 ADM	Administrasi
3	2 MNG	Manager
4	3 OWN	Owner
5	4 SPR	Supervisor
6	5 STF	Staff/Kasir

Tabel m_kategori

Memodifikasi tombol untuk export

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

Menambahkan function export pada controller



```
public function export_excel()
{
    $level = LevelModel::select('level_kode', 'level_nama')
        ->orderBy('level_kode')
        ->get();

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

    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'Kode Level');
    $sheet->setCellValue('C1', 'Nama Level');

    $sheet->getStyle('A1:C1')->getFont()->setBold(true);

    $no = 1;
    $baris = 2;
    foreach ($level as $key => $value) {
        $sheet->setCellValue('A' . $baris, $no);
        $sheet->setCellValue('B' . $baris, $value->level_kode);
        $sheet->setCellValue('C' . $baris, $value->level_nama);
        $baris++;
        $no++;
    }

    foreach (range('A', 'C') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    $sheet->setTitle('Data Level');

    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $filename = 'Data Level '.date('Y-m-d H:i:s').'.xlsx';

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

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

Menambahkan route untuk export

```
Route::post('/import_ajax', [KategoriController::class, 'import_ajax']);
Route::get('/export_excel', [KategoriController::class, 'export_excel']);
});
```



No	Kode Kategori	Nama Kategori	Aksi
1	ELK	Elektronik	[Detail] [Edit] [Hapus]
2	PAK	Pakaian	[Detail] [Edit] [Hapus]
3	MAK	Makanan	[Detail] [Edit] [Hapus]
4	MIN	Minuman	[Detail] [Edit] [Hapus]
5	PRR	Peralatan Rumahh	[Detail] [Edit] [Hapus]
6	CML	Cemilan	[Detail] [Edit] [Hapus]
7	MNR	Minuman Ringan	[Detail] [Edit] [Hapus]
8	8	Bahan Pokok	[Detail] [Edit] [Hapus]
9	9	Obat-obatan	[Detail] [Edit] [Hapus]

No	Kode Kategori	Nama Kategori
2	1	8 Bahan Pokok
3	2	9 Obat-obatan
4	3 CML	Cemilan
5	4 ELK	Elektronik
6	5 MAK	Makanan
7	6 MIN	Minuman
8	7 MNR	Minuman Ringan
9	8 PAK	Pakaian
10	9 PRR	Peralatan Rumahh

Tabel t_stok

Memodifikasi tombol untuk export

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

Menambahkan function export pada controller



```
public function export_excel()
{
    $stok = StokModel::select('barang_id', 'user_id', 'stok_tanggal', 'stok_jumlah')
        ->orderBy('user_id')
        ->with('barang', 'user')
        ->get();

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

    $sheet->setCellValue('A1', 'No');
    $sheet->setCellValue('B1', 'Nama Barang');
    $sheet->setCellValue('C1', 'Nama User');
    $sheet->setCellValue('D1', 'Tanggal');
    $sheet->setCellValue('E1', 'Jumlah Stok');

    $sheet->getStyle('A1:E1')->getFont()->setBold(true);

    $no = 1;
    $baris = 2;
    foreach ($stok as $key => $value) {
        $sheet->setCellValue('A' . $baris, $no);
        $sheet->setCellValue('B' . $baris, $value->barang->barang_nama);
        $sheet->setCellValue('C' . $baris, $value->user->nama);
        $sheet->setCellValue('D' . $baris, $value->stok_tanggal);
        $sheet->setCellValue('E' . $baris, $value->stok_jumlah);
        $baris++;
        $no++;
    }

    foreach (range('A', 'E') as $columnID) {
        $sheet->getColumnDimension($columnID)->setAutoSize(true);
    }

    $sheet->setTitle('Data Stok');

    $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');
    $filename = 'Data Stok '.date('Y-m-d H:i:s').'.xlsx';

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

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

Menambahkan route untuk export

```
Route::get('/export_excel', [StokController::class, 'export_excel']);
```



127.0.0.1:8000/stok

PWL - Starter Code

Home Contact

Daftar Stok

Daftar Stok Barang

Import Stok Export Stok Tambah Data (Ajax)

Filter: - Semua Nama User

Show 10 entries Search:

ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
1	Laptop	Administrator	2025-03-24 00:00:00	50	[Detail] [Edit] [Hapus]
2	Smartphone	Administrator	2025-03-04 18:26:17	25	[Detail] [Edit] [Hapus]
3	Kaos	Manager	2025-03-04 18:26:17	20	[Detail] [Edit] [Hapus]
4	Jaket	Manager	2025-03-04 18:26:17	15	[Detail] [Edit] [Hapus]
5	Roti	Manager	2025-03-04 18:26:17	10	[Detail] [Edit] [Hapus]
6	Kue	Staff/Kasir	2025-03-04 18:26:17	50	[Detail] [Edit] [Hapus]
7	Teh Botol	Staff/Kasir	2025-03-04 18:26:17	40	[Detail] [Edit] [Hapus]
8	Kopi	Staff/Kasir	2025-03-04 18:26:17	35	[Detail] [Edit] [Hapus]
9	Panci	Administrator	2025-03-04 18:26:17	45	[Detail] [Edit] [Hapus]
10	Laptop	Manager55	2025-03-10 00:00:00	50	[Detail] [Edit] [Hapus]

Showing 1 to 10 of 12 entries Previous 1 2 Next

1 ID Nama Barang Nama User Tanggal Stok Jumlah Stok Aksi

11	Panci	Staff/Kasir	0000-00-00 00:00:00	35	[Detail] [Edit] [Hapus]
12	Sapu	Staff/Kasir	0000-00-00 00:00:00	25	[Detail] [Edit] [Hapus]

1	No	Nama Barang	Nama User	Tanggal	Jumlah Stok
2	1	Laptop	Administrator	2025-03-24 00:00:00	50
3	2	Smartphone	Administrator	2025-03-04 18:26:17	25
4	3	Panci	Administrator	2025-03-04 18:26:17	45
5	4	Kaos	Manager	2025-03-04 18:26:17	20
6	5	Jaket	Manager	2025-03-04 18:26:17	15
7	6	Roti	Manager	2025-03-04 18:26:17	10
8	7	Kue	Staff/Kasir	2025-03-04 18:26:17	50
9	8	Teh Botol	Staff/Kasir	2025-03-04 18:26:17	40
10	9	Kopi	Staff/Kasir	2025-03-04 18:26:17	35
11	10	Panci	Staff/Kasir	0000-00-00 00:00:00	35
12	11	Sapu	Staff/Kasir	0000-00-00 00:00:00	25
13	12	Laptop	Manager55	2025-03-10 00:00:00	50

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - Mengubah tombol untuk export
 - Menambahkan function untuk export
 - Menambahkan route untuk export
3. Submit kode untuk implementasi praktikum 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

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

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

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

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

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

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



```
public function export_pdf()
{
    $barang = BarangModel::select('kategori_id','barang_kode','barang_nama','harga_beli','harga_jual')
        ->orderBy('kategori_id')
        ->orderBy('barang_kode')
        ->with('kategori')
        ->get();

    // use Barryvdh\DomPDF\Facade\Pdf;
    $pdf = Pdf::loadView('barang.export_pdf', ['barang' => $barang]);
    $pdf->setPaper('a4', 'portrait'); // set ukuran kertas dan orientasi
    $pdf->setOption("isRemoteEnabled", true); // set true jika ada gambar dari url
    $pdf->render();

    return $pdf->stream('Data Barang '.date('Y-m-d H:i:s').'.pdf');
}
```

5. Selanjutnya, kita buat view untuk dijadikan pdf dari layout html. File bisa kita buat dengan nama [barang/export_pdf.blade.php](#)

INFO

Perlu diingat, pada export_pdf, kita gunakan style CSS secara manual, dan tidak menggunakan framework CSS seperti bootstrap. Karena library 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;
        }
    </style>
</head>
<body>
```



```
.text-center {
    text-align: center;
}
.p-1{
    padding: 5px 1px 5px 1px;
}
.font-10{
    font-size: 10pt;
}
.font-11{
    font-size: 11pt;
}
.font-12{
    font-size: 12pt;
}
.font-13{
    font-size: 13pt;
}
.border-bottom-header{
    border-bottom: 1px solid;
}
.border-all, .border-all th, .border-all td{
    border: 1px solid;
}

```

```
</style>
</head>
<body>
    <table class="border-bottom-header">
        <tr>
            <td width="15%" class="text-center">
                <span class="text-center d-block font-11 font-bold mb-1">KEMENTERIAN
                PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI</span>
                <span class="text-center d-block font-13 font-bold mb-1">POLITEKNIK NEGERI
                MALANG</span>
                <span class="text-center d-block font-10">Jl. Soekarno-Hatta No. 9 Malang
                65141</span>
                <span class="text-center d-block font-10">Telepon (0341) 404424 Pes. 101-
                105, 0341-404420, Fax. (0341) 404420</span>
                <span class="text-center d-block font-10">Laman: www.polinema.ac.id</span>
            </td>
        </tr>
    </table>

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



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

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

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	ELK001	Laptop	5.000.000	5.500.000	Elektronik
2	ELK002	Smartphone	3.000.000	3.500.000	Elektronik
3	PAK001	Kaos	50.000	75.000	Pakaian
4	PAK002	Jaket	150.000	200.000	Pakaian
5	MAK001	Roti	10.000	15.000	Makanan
6	MAK002	Kue	20.000	25.000	Makanan
7	MIN001	Teh Botol	5.000	7.000	Minuman
8	MIN002	Kopi	10.000	12.000	Minuman
9	PRR001	Panci	50.000	75.000	Peralatan Rumah
10	PRR002	Sapu	25.000	30.000	Peralatan Rumah
11	CML001	Keripik Kentang	15.000	18.000	Cemilan
12	CML002	Popcorn	15.000	17.000	Cemilan
13	CML003	Biskuit	10.000	13.000	Cemilan
14	MNR001	Ale-ale	1.500	2.500	Minuman Ringan

Tugas 3 – Implementasi Export PDF pada Laravel :

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

Tabel m_user

Membuatkan view baru untuk export



```
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Laporan Data User</title>
7  <style>
8      body {
9          font-family: "Times New Roman", Times, serif;
10         margin: 6px 20px 5px 20px;
11         line-height: 15px;
12     }
13     table {
14         width: 100%;
15         border-collapse: collapse;
16     }
17     td, th {
18         padding: 4px 3px;
19     }
20     th {
21         text-align: left;
22     }
23     .d-block {
24         display: block;
25     }
26     img.image {
27         width: auto;
28         height: 80px;
29         max-width: 150px;
30         max-height: 150px;
31     }
32     .text-right {
33         text-align: right;
34     }
35     .text-center {
36         text-align: center;
37     }
38     .p-1 {
39         padding: 5px 1px;
40     }
41     .font-10 { font-size: 10pt; }
42     .font-11 { font-size: 11pt; }
43     .font-12 { font-size: 12pt; }
44     .font-13 { font-size: 13pt; }
45     .border-bottom-header {
46         border-bottom: 1px solid;
47     }
48     .border-all, .border-all th, .border-all td {
49         border: 1px solid;
50     }
51  </style>
52 </head>
53 <body>
54     <table class="border-bottom-header">
55         <tr>
56             <td width="15%" class="text-center">
57                 
58             </td>
59             <td width="85%">
60                 <span class="text-center d-block font-11 font-bold mb-1">
61                     KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
62                 </span>
63                 <span class="text-center d-block font-13 font-bold mb-1">
64                     POLITEKNIK NEGERI MALANG
65                 </span>
66                 <span class="text-center d-block font-10">
67                     Jl. Soekarno-Hatta No. 9 Malang 65141
68                 </span>
69                 <span class="text-center d-block font-10">
70                     Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420
71                 </span>
72                 <span class="text-center d-block font-10">
73                     Laman: www.polinema.ac.id
74                 </span>
75             </td>
76         </tr>
77     </table>
78
79     <h3 class="text-center">LAPORAN DATA USER</h3>
80
81     <table class="border-all">
82         <thead>
83             <tr>
84                 <th class="text-center">No</th>
85                 <th>Username</th>
86                 <th>Nama</th>
87                 <th>Level Pengguna</th>
88             </tr>
89         </thead>
90         <tbody>
91             @foreach($user as $u)
92                 <tr>
93                     <td class="text-center">{{ $loop->iteration }}</td>
94                     <td>{{ $u->username }}</td>
95                     <td>{{ $u->nama }}</td>
96                     <td>{{ $u->level->level_nama }}</td>
97                 </tr>
98             @endforeach
99         </tbody>
100    </table>
101 </body>
102 </html>
```



Menambahkan function export pada controller

```
public function export_pdf()
{
    $user = UserModel::select('level_id', 'username', 'nama', 'password')
        ->orderBy('level_id')
        ->with('level')
        ->get();

    // use Barryvdh\DomPDF\Facade\Pdf;
    $pdf = Pdf::loadView('user.export_pdf', ['user' => $user]);
    $pdf->setPaper('a4', 'portrait');
    $pdf->setOption("isRemoteEnabled", true);
    $pdf->render();

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

Menambahkan route untuk memproses export

```
Route::get('/export_pdf', [UserController::class, 'export_pdf']);
```

LAPORAN DATA USER

No	Username	Nama	Level Pengguna
1	admin	Administrator	Administrasi
2	manager	Manager	Manager
3	manager33	Manager Tiga Tiga	Manager
4	manager56	Manager55	Manager
5	manager12	Manager11	Manager
6	Manager11	Manager111	Manager
7	Rusgart	Bayu Triswibowo	Manager
8	staff	StaffKasir	Staff Kasir
9	customer-1	Pelanggan Pertama	Staff Kasir
10	Artemis	Adam	Staff Kasir
11	Asmodeus	Nicho	Staff Kasir

Tabel m_kategori

Membuatkan view baru untuk export



```
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Laporan Data Kategori</title>
7      <style>
8          body {
9              font-family: "Times New Roman", Times, serif;
10             margin: 6px 20px 5px 20px;
11             line-height: 15px;
12         }
13         table {
14             width: 100%;
15             border-collapse: collapse;
16         }
17         td, th {
18             padding: 4px 3px;
19         }
20         th {
21             text-align: left;
22         }
23         .d-block {
24             display: block;
25         }
26         img.image {
27             width: auto;
28             height: 80px;
29             max-width: 150px;
30             max-height: 150px;
31         }
32         .text-right {
33             text-align: right;
34         }
35         .text-center {
36             text-align: center;
37         }
38         .p-1 {
39             padding: 5px 1px;
40         }
41         .font-10 { font-size: 10pt; }
42         .font-11 { font-size: 11pt; }
43         .font-12 { font-size: 12pt; }
44         .font-13 { font-size: 13pt; }
45         .border-bottom-header {
46             border-bottom: 1px solid;
47         }
48         .border-all, .border-all th, .border-all td {
49             border: 1px solid;
50         }
51     </style>
52  </head>
53  <body>
54      <table class="border-bottom-header">
55          <tr>
56              <td width="15%" class="text-center">
57                  
58              </td>
59              <td width="85%">
60                  <span class="text-center d-block font-11 font-bold mb-1">
61                      KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
62                  </span>
63                  <span class="text-center d-block font-13 font-bold mb-1">
64                      POLITEKNIK NEGERI MALANG
65                  </span>
66                  <span class="text-center d-block font-10">
67                      Jl. Soekarno-Hatta No. 9 Malang 65141
68                  </span>
69                  <span class="text-center d-block font-10">
70                      Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420
71                  </span>
72                  <span class="text-center d-block font-10">
73                      Laman: www.polinema.ac.id
74                  </span>
75              </td>
76          </tr>
77      </table>
78
79      <h3 class="text-center">LAPORAN DATA KATEGORI</h3>
80
81      <table class="border-all">
82          <thead>
83              <tr>
84                  <th class="text-center">No</th>
85                  <th>Kode Kategori</th>
86                  <th>Nama Kategori</th>
87              </tr>
88          </thead>
89          <tbody>
90              @foreach($kategori as $k)
91                  <tr>
92                      <td class="text-center">{!! $loop->iteration !!}</td>
93                      <td>{!! $k->kategori_kode !!}</td>
94                      <td>{!! $k->kategori_nama !!}</td>
95                  </tr>
96              @endforeach
97          </tbody>
98      </table>
99  </body>
100 </html>
```



Menambahkan function export pada controller

```
public function export_pdf()
{
    $kategori = KategoriModel::select('kategori_kode', 'kategori_nama')
        ->orderBy('kategori_kode')
        ->get();

    // use Barryvdh\DomPDF\Facade\Pdf;
    $pdf = Pdf::loadView('kategori.export_pdf', ['kategori' => $kategori]);
    $pdf->setPaper('a4', 'portrait');
    $pdf->setOption("isRemoteEnabled", true);
    $pdf->render();

    return $pdf->stream('Data kategori '.date('Y-m-d H:i:s').'.pdf');
}
```

Menambahkan route untuk memproses export

```
Route::get('/export_pdf', [KategoriController::class, 'export_pdf']);
```

No	Kode Kategori	Nama Kategori
1	S	Hakan Paklok
2	9	Obat-obatan
3	CML	Cemilan
4	ELK	Elektronik
5	MAK	Makanan
6	MIN	Minuman
7	MNR	Minuman Ringan
8	PAK	Pakaian
9	PRR	Peralatan Rumah

Tabel m_level

Membuatkan view baru untuk export



```
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Laporan Data Level</title>
7      <style>
8          body {
9              font-family: "Times New Roman", Times, serif;
10             margin: 6px 20px 5px 20px;
11             line-height: 15px;
12         }
13         table {
14             width: 100%;
15             border-collapse: collapse;
16         }
17         td, th {
18             padding: 4px 3px;
19         }
20         th {
21             text-align: left;
22         }
23         .d-block {
24             display: block;
25         }
26         img.image {
27             width: auto;
28             height: 80px;
29             max-width: 150px;
30             max-height: 150px;
31         }
32         .text-right {
33             text-align: right;
34         }
35         .text-center {
36             text-align: center;
37         }
38         .p-1 {
39             padding: 5px 1px;
40         }
41         .font-10 { font-size: 10pt; }
42         .font-11 { font-size: 11pt; }
43         .font-12 { font-size: 12pt; }
44         .font-13 { font-size: 13pt; }
45         .border-bottom-header {
46             border-bottom: 1px solid;
47         }
48         .border-all, .border-all th, .border-all td {
49             border: 1px solid;
50         }
51     </style>
52 </head>
53 <body>
54     <table class="border-bottom-header">
55         <tr>
56             <td width="15%" class="text-center">
57                 
58             </td>
59             <td width="85%">
60                 <span class="text-center d-block font-11 font-bold mb-1">
61                     KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
62                 </span>
63                 <span class="text-center d-block font-13 font-bold mb-1">
64                     POLITEKNIK NEGERI MALANG
65                 </span>
66                 <span class="text-center d-block font-10">
67                     Jl. Soekarno-Hatta No. 9 Malang 65141
68                 </span>
69                 <span class="text-center d-block font-10">
70                     Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420
71                 </span>
72                 <span class="text-center d-block font-10">
73                     Laman: www.polinema.ac.id
74                 </span>
75             </td>
76         </tr>
77     </table>
78     <h3 class="text-center">LAPORAN DATA LEVEL</h3>
79     <table class="border-all">
80         <thead>
81             <tr>
82                 <th class="text-center">No</th>
83                 <th>Kode Level</th>
84                 <th>Nama Level</th>
85             </tr>
86         </thead>
87         <tbody>
88             @foreach($level as $l)
89                 <tr>
90                     <td class="text-center">{{ $loop->iteration }}</td>
91                     <td>{{ $l->level_kode }}</td>
92                     <td>{{ $l->level_nama }}</td>
93                 </tr>
94             @endforeach
95         </tbody>
96     </table>
97     </body>
98 </html>
```



Menambahkan function export pada controller

```
public function export_pdf()
{
    $level = LevelModel::select('level_kode', 'level_nama')
        ->orderBy('level_kode')
        ->get();

    // use Barryvdh\DomPDF\Facade\Pdf;
    $pdf = Pdf::loadView('level.export_pdf', ['level' => $level]);
    $pdf->setPaper('a4', 'portrait');
    $pdf->setOption("isRemoteEnabled", true);
    $pdf->render();

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

Menambahkan route untuk memproses export

```
Route::get('/export_pdf', [LevelController::class, 'export_pdf']);
```

Laporan Data Level

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

LAPORAN DATA LEVEL

No	Kode Level	Nama Level
1	ADM	Administrasi
2	MNG	Manager
3	OWN	Owner
4	SPR	Supervisor
5	STF	Staff/Kasir

Tabel t_stok

Membuatkan view baru untuk export



```
 1 <!DOCTYPE html>
 2 <html lang="id">
 3 <head>
 4     <meta charset="UTF-8">
 5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
 6     <title>Laporan Data Stok</title>
 7     <style>
 8         body {
 9             font-family: "Times New Roman", Times, serif;
10             margin: 6px 20px 5px 20px;
11             line-height: 15px;
12         }
13         table {
14             width: 100%;
15             border-collapse: collapse;
16         }
17         td, th {
18             padding: 4px 3px;
19         }
20         th {
21             text-align: left;
22         }
23         .d-block {
24             display: block;
25         }
26         img.image {
27             width: auto;
28             height: 80px;
29             max-width: 150px;
30             max-height: 150px;
31         }
32         .text-right {
33             text-align: right;
34         }
35         .text-center {
36             text-align: center;
37         }
38         .p-1 {
39             padding: 5px 1px;
40         }
41         .font-10 { font-size: 10pt; }
42         .font-11 { font-size: 11pt; }
43         .font-12 { font-size: 12pt; }
44         .font-13 { font-size: 13pt; }
45         .border-bottom-header {
46             border-bottom: 1px solid;
47         }
48         .border-all, .border-all th, .border-all td {
49             border: 1px solid;
50         }
51     </style>
52 </head>
53 <body>
54     <table class="border-bottom-header">
55         <tr>
56             <td width="15%" class="text-center">
57                 
58             </td>
59             <td width="85%">
60                 <span class="text-center d-block font-11 font-bold mb-1">
61                     KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
62                 </span>
63                 <span class="text-center d-block font-13 font-bold mb-1">
64                     POLITEKNIK NEGERI MALANG
65                 </span>
66                 <span class="text-center d-block font-10">
67                     Jl. Soekarno-Hatta No. 9 Malang 65141
68                 </span>
69                 <span class="text-center d-block font-10">
70                     Telepon (0341) 404424 Pes. 101-105, 0341-404428, Fax. (0341) 404420
71                 </span>
72                 <span class="text-center d-block font-10">
73                     Laman: www.polinema.ac.id
74                 </span>
75             </td>
76         </tr>
77     </table>
78     <h3 class="text-center">LAPORAN DATA STOK</h3>
79
80     <table class="border-all">
81         <thead>
82             <tr>
83                 <th class="text-center">No</th>
84                 <th>Nama Barang</th>
85                 <th>Nama User</th>
86                 <th>Tanggal Stok</th>
87                 <th class="text-right">Jumlah Stok</th>
88             </tr>
89         </thead>
90         <tbody>
91             @foreach($stok as $s)
92             <tr>
93                 <td class="text-center">{{ $loop->iteration }}</td>
94                 <td>{{ $s->barang->barang_nama }}</td>
95                 <td>{{ $s->user->username }}</td>
96                 <td>{{ $s->stok_tanggal }}</td>
97                 <td class="text-right">{{ number_format($s->stok_jumlah, 0, ',', '.') }}</td>
98             </tr>
99             @endforeach
100        </tbody>
101    </table>
102
103 </body>
104 </html>
```



Menambahkan function export pada controller

```
public function export_pdf()
{
    $stok = StokModel::select('barang_id', 'user_id', 'stok_tanggal', 'stok_jumlah')
        ->orderBy('user_id')
        ->with('barang', 'user')
        ->get();

    // use Barryvdh\DomPDF\Facade\Pdf;
    $pdf = Pdf::loadView('stok.export_pdf', ['stok' => $stok]);
    $pdf->setPaper('a4', 'portrait');
    $pdf->setOption("isRemoteEnabled", true);
    $pdf->render();

    return $pdf->stream('Data Stok '.date('Y-m-d H:i:s').'.pdf');
}
```

Menambahkan route untuk memproses export

```
Route::get('/export_pdf', [StokController::class, 'export_pdf']);
```

The screenshot shows a browser window with the URL `127.0.0.1:8000/stok/export_pdf`. The page title is "Laporan Data Stok". On the left, there is a thumbnail preview of the PDF document. On the right, the PDF content is displayed, featuring the Politeknik Negeri Malang logo at the top, followed by the text "KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI POLITEKNIK NEGERI MALANG" and its address. Below this is a table titled "LAPORAN DATA STOK" with 12 rows of data.

No	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok
1	Laptop	admin	2025-03-24 00:00:00	50
2	Smartphone	admin	2025-03-04 18:26:17	25
3	Panci	admin	2025-03-04 18:26:17	45
4	Kaos	manager	2025-03-04 18:26:17	20
5	Jaket	manager	2025-03-04 18:26:17	15
6	Roti	manager	2025-03-04 18:26:17	10
7	Kue	staff	2025-03-04 18:26:17	50
8	Teh Botol	staff	2025-03-04 18:26:17	40
9	Kopi	staff	2025-03-04 18:26:17	35
10	Panci	staff	0000-00-00 00:00:00	35
11	Sapu	staff	0000-00-00 00:00:00	25
12	Laptop	manager56	2025-03-10 00:00:00	50

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - Membuatkan view baru untuk export
 - Menambahkan function export pada controller untuk memproses export
 - Menambahkan route untuk memproses export
3. Submit kode untuk implementasi export pdf pada repository github kalian.

Tugas 4 – Implementasi Upload File Gambar :

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

Membuat migration baru untuk menambahkan kolom baru pada tabel m_user



```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('t_penjualan_detail', function (Blueprint $table) {
15             $table->id('detail_id');
16             $table->unsignedBigInteger('penjualan_id');
17             $table->unsignedBigInteger('barang_id');
18             $table->integer('harga');
19             $table->integer('jumlah');
20             $table->timestamps();
21
22             $table->foreign('penjualan_id')->references('penjualan_id')->on('t_penjualan');
23             $table->foreign('barang_id')->references('barang_id')->on('m_barang');
24         });
25     }
26
27     /**
28      * Reverse the migrations.
29      */
30     public function down(): void
31     {
32         Schema::dropIfExists('t_penjualan_detail');
33     }
34 };
```

```
protected $table = 'm_user';
protected $primaryKey = 'user_id';
protected $fillable = ['username', 'password', 'nama', 'level_id', 'created_at', 'updated_at', 'photo_profile'];
```

menambahkan fuction baru untuk profile



```
public function profile()
{
    $user = auth()->user();

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

    $breadcrumb = (object) [
        'title' => 'Profile User',
        'list' => ['Home', 'Profile']
    ];

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

    $activeMenu = 'profile';

    return view('user.photo_profile', compact('user', 'breadcrumb', 'page', 'activeMenu'));
}
```

```
public function update_photo(Request $request)
{
    // Validasi file
    $request->validate([
        'photo_profile' => 'required|image|mimes:jpeg,png,jpg|max:2048',
    ]);

    try {
        $user = auth()->user();

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

        $userId = $user->user_id;

        $userModel = UserModel::find($userId);

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

        // Menghapus foto jika sudah ada
        if ($userModel->photo_profile && file_exists(storage_path('app/public/' . $userModel->photo_profile))) {
            Storage::disk('public')->delete($userModel->photo_profile);
        }

        $fileName = 'profile_' . $userId . '_' . time() . '.' . $request->photo_profile->extension();
        $path = $request->photo_profile->storeAs('profiles', $fileName, 'public');

        UserModel::where('user_id', $userId)->update([
            'photo_profile' => $path
        ]);

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

membuat route baru



```
Route::get('/profile', [UserController::class, 'profile']);
Route::post('/update-photo', [UserController::class, 'update_photo']);
```

membuat view baru untuk profile

```
1  @extends('layouts.template')
2
3  @section('content')
4  <div class="card">
5      <div class="card-header">
6          <h3 class="card-title">Profile User</h3>
7      </div>
8      <div class="card-body">
9          <div class="row">
10             <div class="col-md-4 text-center">
11                 
12                 <img alt="User Image" style="width: 200px; height: 200px; object-fit: cover;">
13
14                 <form action="{{ url('/update-photo') }}" method="POST" enctype="multipart/form-data" class="mt-3">
15                     @csrf
16                     <div class="form-group">
17                         <div class="input-group">
18                             <div class="custom-file">
19                                 <input type="file" class="custom-file-input" id="photo_profile" name="photo_profile" accept="image/*">
20                                 <label class="custom-file-label" for="photo_profile">Pilih Foto</label>
21                             </div>
22                         </div>
23                         @error('photo_profile')
24                             <small class="text-danger">{{ $message }}</small>
25                         @enderror
26                     </div>
27                     <button type="submit" class="btn btn-primary">Upload Foto</button>
28                 </form>
29             </div>
30             <div class="col-md-8">
31                 <h4>Data User</h4>
32                 <table class="table">
33                     <tr>
34                         <th width="30%">Username</th>
35                         <td>{{ $user->username }}</td>
36                     </tr>
37                     <tr>
38                         <th>Nama</th>
39                         <td>{{ $user->name }}</td>
40                     </tr>
41                     <tr>
42                         <th>Level</th>
43                         <td>{{ $user->level->level_nama ?? '-' }}</td>
44                     </tr>
45                 </table>
46             </div>
47         </div>
48     </div>
49     @endsection
50
51     @push('scripts')
52     <script>
53         $(document).ready(function() {
54             $('#photo_profile').on('change', function() {
55                 var fileName = $(this).val().split('\\').pop();
56                 $(this).next('.custom-file-label').html(fileName);
57
58                 if (this.files && this.files[0]) {
59                     var reader = new FileReader();
60                     reader.onload = function(e) {
61                         $('img.img-circle').attr('src', e.target.result);
62                     }
63                     reader.readAsDataURL(this.files[0]);
64                 }
65             });
66         });
67     </script>
68     @endpush
69 
```



2. Jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - Membuat migration baru untuk menambahkan kolom baru pada tabel m_user
 - Membuatkan function baru pada controller untuk memproses update profile
 - Menambahkan route baru
 - Membuatka view baru untuk tampilan
3. Submit kode untuk impementasi export pdf pada repository github kalian.

*** Sekian, dan selamat belajar ***