



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

		kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	ELEC	Elektronik	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	FASH	Fashion	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	HERB	Bumbu Dapur	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	TOYS	Mainan	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	5	HOME	Perabotan Rumah	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	6	CML	Camilan	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	7	MNR	Minuman Ringan	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	14	PRF	Parfum	2025-03-18 08:17:31	2025-03-18 08:17:31

		barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
<input type="checkbox"/>	Edit Copy Delete	1	1	B001	TV Sumsang	3000000	4000000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	2	1	B002	Laptop Asus	5000000	6000000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	3	2	B003	Kaos	50000	60000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	4	2	B004	Sepatu Sneakers	120000	150000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	5	3	B005	Garam	3000	5000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	6	3	B006	Gula	7000	10000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	7	4	B007	Lego	500000	650000	NULL	NULL
<input type="checkbox"/>	Edit Copy Delete	8	4	B008	Mobil Remote	200000	350000	NULL	NULL

### Tampilan `m_kategori` dan `m_barang`



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

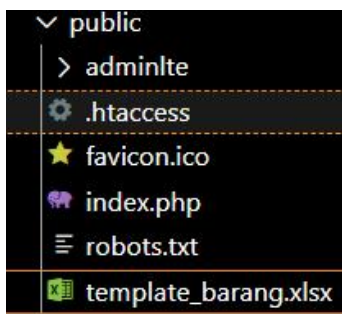
---

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.



Membuat file template excel dengan data sesuai pada database

- 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>
</section>
```



```
<table class="table table-bordered table-sm table-striped table-hover"
id="table-barang">
  <thead>
    <tr><th>No</th><th>Kode Barang</th><th>Kode Barang</th><th>Harga
Beli</th><th>Harga Jual</th><th>Kategori</th><th>Aksi</th></tr>
  </thead>
  <tbody></tbody>
</table>
</div>
</div>
<div id="myModal" class="modal fade animate shake" tabindex="-1" data-backdrop="static"
data-keyboard="false" data-width="75%"></div>

@endsection

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

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



```
        orderable: true,  
        searchable: false,  
        render: function(data, type, row){  
            return new Intl.NumberFormat('id-ID').format(data);  
        }  
    },  
    {  
        data: "kategori.kategori_nama",  
        className: "",  
        width: "14%",  
        orderable: true,  
        searchable: false  
    },  
    {  
        data: "aksi",  
        className: "text-center",  
        width: "14%",  
        orderable: false,  
        searchable: false  
    }  
    ]  
});  
  
$('#table-barang_filter input').unbind().bind().on('keyup',  
    function(e){ if(e.keyCode == 13){ // enter key  
        tableBarang.search(this.value).draw();  
    }  
});  
  
$('.filter_kategori').change(function(){ tableBarang.  
    draw();  
});  
});  
</script>  
@endpush
```

```
resources > views > barang > index.blade.php > {} "index.blade.php" > script > ready() callback > ajax > "type"  
1 @extends('layouts.template')  
2  
3 @section('section:content')  
4 <div class="card">  
5     <div class="card-header">  
6         <h3 class="card-title">Daftar barang</h3>  
7         <div class="card-tools">  
8             <button onclick="modalAction('{{ url(path: '/barang/import') }}')" class="btn btn-info">Import Barang</button>  
9             <button onclick="modalAction('{{ url(path: '/barang/create_ajax') }}')" class="btn btn-success">Tambah Data</button>  
10        </div>  
11    </div>  
12    <div class="card-body">  
13        <div id="filter" class="form-horizontal filter-date p-2 border-bottom mb-2">  
14            <div class="row">  
15                <div class="col-md-12">  
16                    <div class="form-group form-group-sm row text-sm mb-0">  
17                        <label for="filter_date" class="col-md-1 col-form-label">Filter</label>  
18                        <div class="col-md-3">  
19                            <select name="filter_kategori" class="form-control form-control-sm filter_kategori">  
20                                <option value="">- Semua -</option>  
21                                @foreach ($kategori as $1)  
22                                    <option value="{{ $1->kategori_id }}">{{ $1->kategori_nama }}</option>  
23                                </foreach>  
24                            </select>  
25                            <small class="form-text text-muted">Kategori Barang</small>  
26                        </div>  
27                    </div>  
28                </div>  
29            </div>  
30        </div>  
31        @if (session(key: 'success'))  
32            <div class="alert alert-success">{{ session(key: 'success') }}</div>  
33        @endif  
34        @if (session(key: 'error'))  
35            <div class="alert alert-danger">{{ session(key: 'error') }}</div>  
36        @endif  
37        <table class="table table-bordered table-sm table-striped table-hover" id="table-barang">  
38            <thead>
```

Menyesuaikan tampilan index barang dengan menambahkan button untuk menampilkan dialog popup untuk import file



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>
```



```
1 <form action="{{ url(path: '/barang/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" id="exampleModallabel">Import Data Barang</h5>
7 <button type="button" class="close" data-dismiss="modal" aria-label="Close"><span
8 | | aria-hidden="true">&times;</span></button>
9 </div>
10 <div class="modal-body">
11 <div class="form-group">
12 <label>Download Template</label>
13 <a href="{{ asset(path: 'template_barang.xlsx') }}" class="btn btn-info btn-sm" download><i
14 | | class="fa fa-file-excel"></i>Download</a>
15 <small id="error-kategori_id" class="error-text form-text text-danger"></small>
16 </div>
17 <div class="form-group">
18 <label>Pilih File</label>
19 <input type="file" name="file_barang" id="file_barang" class="form-control" required>
20 <small id="error-file_barang" class="error-text form-text text-danger"></small>
21 </div>
22 </div>
23 <div class="modal-footer">
24 <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
25 <button type="submit" class="btn btn-primary">Upload</button>
26 </div>
27 </div>
28 </div>
29 </form>
30 <script>
31 $(document).ready(function() {
32 $("#form-import").validate({
33 rules: {
34 file_barang: {
35 required: true,
36 extension: ".xlsx"
37 },
38 },
39 submitHandler: function(form) {
40 var formData = new FormData(form);
41
42 $.ajax({
43 url: form.action,
44 type: form.method,
```

Menampilkan halaman popup dialog form untuk melakukan import data dari file berformat xlsx

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
```

```
Route::delete(uri: '{id}/delete_ajax', action: [BarangController::class, 'delete_ajax']);  
Route::delete(uri: '{id}', action: [BarangController::class, 'destroy']); // menghapus data user  
Route::get(uri: '/import', action: [BarangController::class, 'import']);  
Route::post(uri: '/import_ajax', action: [BarangController::class, 'import_ajax']);
```

Menambahkan route import untuk menampilkan form, kemudian import ajax untuk melakukan importing data dari file yang telah di inputkan pada form

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

```
composer require phoffice/phpspreadsheet
```

```
PS C:\Users\Pongo\Documents\Kuliah\PWL_2025\Week8\PWL_POS> composer require phoffice/phpspreadsheet  
./composer.json has been updated  
Running composer update phoffice/phpspreadsheet  
Loading composer repositories with package information  
Updating dependencies  
Lock file operations: 5 installs, 0 updates, 0 removals
```

Menginstall library [phpoffice](#) untuk membantu dalam membaca data dalam file [xlsx](#)

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)  
{
```



```
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()
{

```



```
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(),
                    ];
                }
            }
        }

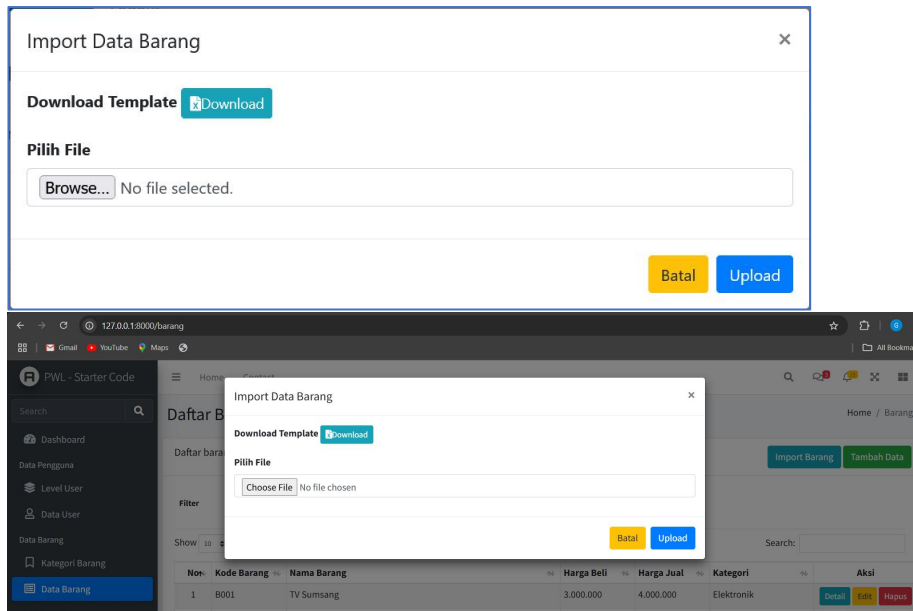
        if(count($insert) > 0){
            // insert data ke database, jika data sudah ada, maka diabaikan
            BarangModel::insertOrIgnore($insert);
        }

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

Menambahkan method import yang mengembalikan view form dan import ajax yang memiliki fungsi untuk importing data

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





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

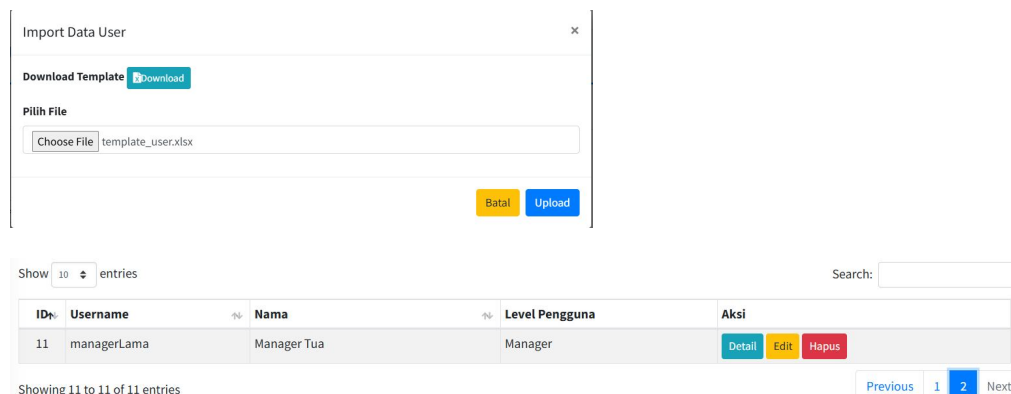
9	B009	Kursi Plastik	500.000	650.000	Perabotan Rumah	Detail	Edit	Hapus
10	B010	Lampu Belajar	35.000	40.000	Perabotan Rumah	Detail	Edit	Hapus
11	SBK-003	Telur Omega(10 butir)	22.000	25.000	Elektronik	Detail	Edit	Hapus
12	SNK-003	Sari Roti	11.500	12.500	Fashion	Detail	Edit	Hapus
13	MND-003	Shampoo Pantene	17.500	18.500	Bumbu Dapur	Detail	Edit	Hapus
14	BAY-003	Baju Bayi 2th	89.000	92.500	Mainan	Detail	Edit	Hapus
15	MNM-003	Cleo 600ml	3.750	4.300	Perabotan Rumah	Detail	Edit	Hapus

Data bertambah pada database melalui metode import file

## Tugas 1 – Implementasi File Upload untuk Import Data:

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

### User







## Level

Import Data Level

Download Template

Download

Pilih File

Choose File

template\_level.xlsx

Batal

Upload

4	CUS	Customer	<div>Detail</div>	<div>Edit</div>	<div>Hapus</div>
5	CEO	Chief Executive	<div>Detail</div>	<div>Edit</div>	<div>Hapus</div>

## Kategori

Import Data Barang

Download Template

Download

Pilih File

Choose File

template\_kategori.xlsx

Batal

Upload

8	PRF	Parfum	<div>Detail</div>	<div>Edit</div>	<div>Hapus</div>
9	TKR	Tikar	<div>Detail</div>	<div>Edit</div>	<div>Hapus</div>

## Stok

Import Data Stok

Download Template

Download

Pilih File

Choose File

template\_stok.xlsx

Batal

Upload

ID	Nama Barang	Nama User	Tanggal Stok	Jumlah Stok	Aksi
11	TV Sumsang	Administrator	0000-00-00 00:00:00	69	<div>Detail</div> <div>Edit</div> <div>Hapus</div>

- Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
- Submit kode untuk impementasi prakktikum 1 pada repository github kalian.



## B. Export Excel di Laravel

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

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

### Praktikum 2 – Export Data ke Excel

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

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

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

Kita ubah dengan

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

```
<div class="card-tools">  
  <button onclick="modalAction('{{ url(path: '/barang/import') }}')" class="btn btn-info">Import Barang</button>  
  <a href="{{ url(path: '/barang/export_excel') }}" class="btn btn-primary"><i class="fa fa-file-excel"></i> Export Barang</a>  
  <button onclick="modalAction('{{ url(path: '/barang/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>  
</div>
```

Mengganti tombol create (bukan ajax) dengan export data barang



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
```

```
Route::get(uri: '/import', action: [BarangController::class, 'import']);
Route::post(uri: '/import_ajax', action: [BarangController::class, 'import_ajax']);
Route::get(uri: '/barang/export_excel', action: [BarangController::class, 'export_excel']);
```

#### Menambahkan method untuk 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();
```

```
public function export_excel(): never
{
    // ambil data barang yang akan di export
    $barang = BarangModel::select(columns: 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
        ->orderBy(column: 'kategori_id')
        ->with(relations: 'kategori')
        ->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
```

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

$sheet->setCellValue('A1', 'No');
$sheet->setCellValue('B1', 'Kode Barang');
$sheet->setCellValue('C1', 'Nama Barang');
$sheet->setCellValue('D1', 'Harga Beli');
$sheet->setCellValue('E1', 'Harga Jual');
$sheet->setCellValue('F1', 'Kategori');

$sheet->getStyle('A1:F1')->getFont()->setBold(true); // bold header
```



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

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

```
$no = 1; // nomor data dimulai dari 1
$baris = 2; // baris data dimulai dari baris ke 2

foreach ($barang as $key => $value) {
    $sheet->setCellValue('A' . $baris, $no);
    $sheet->setCellValue('B' . $baris, $value->barang_kode);
    $sheet->setCellValue('C' . $baris, $value->barang_nama);
    $sheet->setCellValue('D' . $baris, $value->harga_beli);
    $sheet->setCellValue('E' . $baris, $value->harga_jual);
    $sheet->setCellValue('F' . $baris, $value->kategori->kategori_nama); // ambil nama kategori
    $baris++;
    $no++;
}
```

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 }
```

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



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

```
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
```

```
$sheet->setTitle('Data Barang'); // set title sheet

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

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

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

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

AutoSave





### Hasil export xlsx

	A	B	C	D	E	F
1	No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
2	1	B001	TV Sumsang	3000000	4000000	Elektronik
3	2	B002	Laptop Asus	5000000	6000000	Elektronik
4	3	B003	Kaos	50000	60000	Fashion
5	4	B004	Sepatu Sneakers	120000	150000	Fashion
6	5	B005	Garam	3000	5000	Bumbu Dapur
7	6	B006	Gula	7000	10000	Bumbu Dapur
8	7	B007	Lego	500000	650000	Mainan
9	8	B008	Mobil Remote	200000	350000	Mainan
10	9	B009	Kursi Plastik	500000	650000	Perabotan Rumah
11	10	B010	Lampu Belajar	35000	40000	Perabotan Rumah
12						

## Tugas 2 – Implementasi File Export Excel:

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

### User

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Home Contact

Daftar User

Daftar user yang terdaftar dalam sistem

Import User Export Barang Tambah Data

	A	B	C	D	E	F
1	No	Nama	Username	Level		
2	1	Administrator	admin	Administrator		
3	2	Manager	manager	Manager		
4	3	Staff/Kasir	staff	Staff/Kasir		
5	4	Pelanggan Pertama	customer-1	Customer		
6	5	Manager 2	manager_dua	Manager		
7	6	Manager 3	manager_tiga	Manager		
8	7	Manager Dua Dua	manager22	Manager		
9	8	Manager Tiga Tiga	manager33	Manager		
10	9	Manager Lima Lima	manager56	Manager		
11	10	test1	testing	Customer		
12	11	Manager Tua	managerLama	Manager		

### Level

PWL - Starter Code

Search

Dashboard

Data Pengguna

Home Contact

Daftar Level

Daftar level yang terdaftar dalam sistem

Import Level Export Level Tambah Data

Data Level 2025-04-14 09\_28\_15.xlsx  
6.3 KB • Done

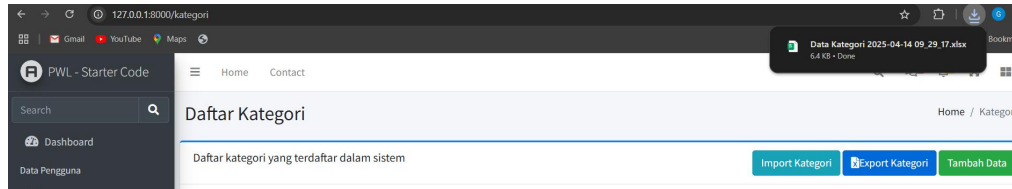
Data User 2025-04-14 09\_27\_29.xlsx  
6.5 KB • Done

Data Barang 2025-04-14 09\_15\_25.xlsx  
6.6 KB • 13 minutes ago

	A	B	C
1	No	Kode Level	Nama Level
2	1	ADM	Administrator
3	2	MNG	Manager
4	3	STF	Staff/Kasir
5	4	CUS	Customer
6	5	CEO	Chief Executive

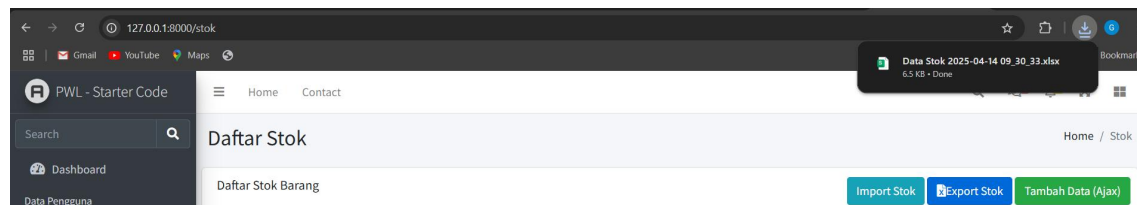


## Kategori



	A	B	C
1	No	Kode Kategori	Nama Kategori
2	1	ELEC	Elektronik
3	2	FASH	Fashion
4	3	HERB	Bumbu Dapur
5	4	TOYS	Mainan
6	5	HOME	Perabotan Rumah
7	6	CML	Camilan
8	7	MNR	Minuman Ringan
9	8	PRF	Parfum
10	9	TKR	Tikar

## Stok



	A	B	C	D	E
1	No	Nama Barang	Nama User	Tanggal	Jumlah Stok
2	1	TV Sumsang		2025-03-05 09:05:42	63
3	2	Laptop Asus		2025-03-05 09:05:42	20
4	3	Kaos		2025-03-05 09:05:42	35
5	4	Sepatu Sneakers		2025-03-05 09:05:42	10
6	5	Garam		2025-03-05 09:05:42	67
7	6	Gula		2025-03-05 09:05:42	20
8	7	Lego		2025-03-05 09:05:42	16
9	8	Mobil Remote		2025-03-05 09:05:42	46
10	9	Kursi Plastik		2025-03-05 09:05:42	26
11	10	Lampu Belajar		2025-03-05 09:05:42	10
12	11	TV Sumsang		0000-00-00 00:00:00	69

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
3. Submit kode untuk impementasi prakktikum 2 pada repository github kalian.



### C. Export PDF di Laravel

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

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

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

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

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

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

```
PS C:\Users\Pongo\Documents\Kuliah\PWL_2025\Week8\PWL_POS> composer require barryvdh/laravel-dompdf
./composer.json has been updated
Running composer update barryvdh/laravel-dompdf
Loading composer repositories with package information
```

Meninstall library **dompdf** untuk menggenerate pdf

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>
```

```
<a href="{{ url(path: '/barang/export_excel') }}" class="btn btn-primary"><i class="fa fa-file-excel"></i>Export Barang</a>
<a href="{{ url(path: '/barang/export_pdf') }}" class="btn btn-warning"><i class="fa fa-file-pdf"></i> Export Barang</a>
<button onclick="modalAction('{{ url(path: '/barang/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
</div>
```

Menambahkan button untuk export pdf

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
```

```
Route::get(uri: '/export_excel', action: [BarangController::class, 'export_excel'])
Route::get(uri: '/export_pdf', action: [BarangController::class, 'export_pdf']);
```

Menambahkan route untuk 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');
}

public function export_pdf(): Response
{
    $barang = BarangModel::select(columns: 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
        ->orderBy(column: 'kategori_id')
        ->orderBy(column: 'barang_kode')
        ->with(relations: '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", false); // set true jika ada gambar dari url
    $pdf->render();

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

Menambahkan method export pdf untuk melakukan export data bertipe pdf

- 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.

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



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

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
  <style>
    body{
      font-family: "Times New Roman", Times, serif;
      margin: 6px 20px 5px 20px;
      line-height: 15px;
    }
    table {
      width:100%;
      border-collapse: collapse;
    }
    td, th {
      padding: 4px 3px;
    }
    th{
      text-align: left;
    }
    .d-block{
      display: block;
    }
    img.image{
      width: auto;
      height: 80px;
      max-width: 150px;
      max-height: 150px;
    }
    .text-right {
      text-align: right;
    }
  }
```



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

<h3 class="text-center">LAPORAN DATA BARANG</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>
</tbody>
</table>
```



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

```
1 <html>
2
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <style>
6 body {
7     font-family: "Times New Roman", Times, serif;
8     margin: 6px 20px 5px 20px;
9     line-height: 15px;
10 }
11
12 table {
13     width: 100%;
14     border-collapse: collapse;
15 }
16
17 td,
18 th {
19     padding: 4px 3px;
20 }
21
22 th {
23     text-align: left;
24 }
25
26 .d-block {
27     display: block;
28 }
29
30 img.image {
31     width: auto;
32     height: 80px;
33     max-width: 150px;
34     max-height: 150px;
35 }
36
37 .text-right {
38     text-align: right;
39 }
40
41 .text-center {
42     text-align: center;
43 }
44
```

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

### Hasil export pdf

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

**LAPORAN DATA BARANG**

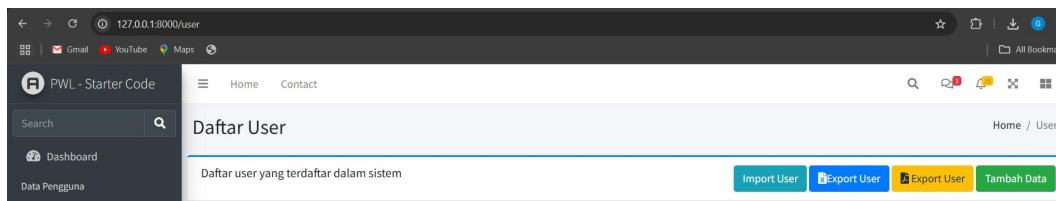
No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	B001	TV Sumsang	3.000.000	4.000.000	Elektronik
2	B002	Laptop Asus	5.000.000	6.000.000	Elektronik
3	B003	Kaos	50.000	60.000	Fashion
4	B004	Sepatu Sneakers	120.000	150.000	Fashion
5	B005	Garam	3.000	5.000	Bumbu Dapur
6	B006	Gula	7.000	10.000	Bumbu Dapur
7	B007	Lego	500.000	650.000	Mainan
8	B008	Mobil Remote	200.000	350.000	Mainan
9	B009	Kursi Plastik	500.000	650.000	Perabotan Rumah
10	B010	Lampu Belajar	35.000	40.000	Perabotan Rumah



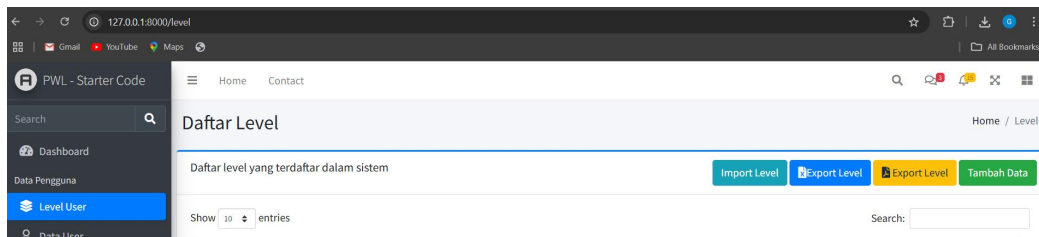
### Tugas 3 – Implementasi Export PDF pada Laravel :

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

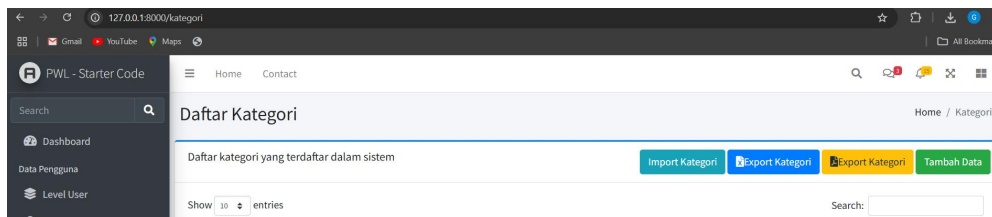
#### User



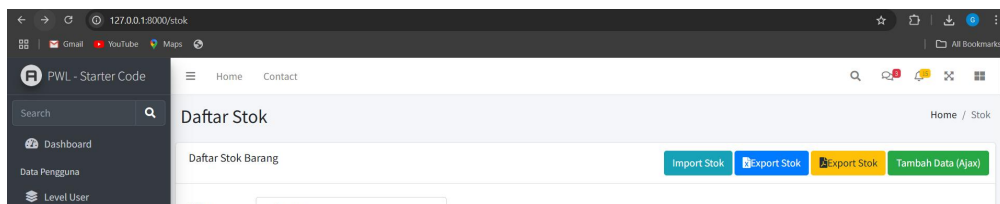
#### Level



#### Kategori



#### Stok



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



#### Tugas 4 – Implementasi Upload File Gambar :

1. Silahkan implementasikan fitur **upload file** untuk mengubah **foto profile** di project web kalian
2. Jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan

#### Menambahkan kolom photo pada migrations

```
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create(table: 'm_user', callback: function (Blueprint $table): void {
            $table->id(column: 'user_id');
            $table->unsignedBigInteger(column: 'level_id')->index(); //indexing untuk ForeignKey
            $table->string(column: 'username', length: 20)->unique(); //unique untuk memastikan tidak ada username yang sama
            $table->string(column: 'name', length: 100);
            $table->string(column: 'password');
            $table->timestamps();
            $table->string(column: 'photo')->nullable(); // Menyimpan nama file foto profil

            // Mendefinisikan Foreign Key pada kolom level_id mengacu pada kolom level_id di tabel m_level
            $table->foreign(columns: 'level_id')->references(columns: 'level_id')->on(table: 'm_level');
        });
    }
}
```

#### Menambahkan inputan foto profile form create maupun update

```
<div class="form-group">
    <label>Foto Profil</label>
    <input type="file" name="photo" id="photo" class="form-control" accept="image/*">
    <small id="error-photo" class="error-text form-text text-danger"></small>
</div>
```

#### Menambahkan inputan baik di from create maupun edit

```
{
    name: {
        required: true,
        minlength: 3,
        maxlength: 100
    },
    password: {
        required: true,
        minlength: 6,
        maxlength: 20
    },
    photo: {
        required: false, // optional, bisa true kalau wajib
        extension: "jpg|jpeg|png",
        filesize: 2048000 // maksimal 2MB
    }
},
submitHandler: function(form) {
```



## Menambahkan kode untuk menyimpan gambar pada storage laravel untuk method store maupun update

```
public function store_ajax(Request $request): JsonResponse|mixed|Redirector|RedirectResponse {
    if ($request->ajax() || $request->wantsJson()) {
        $rules = [
            'level_id' => 'required|integer',
            'username' => 'required|string|min:3|unique:m_user,username',
            'name' => 'required|string|max:100',
            'password' => 'required|min:5',
            'photo' => 'nullable|image|mimes:jpg,jpeg,png|max:2048', // max 2MB
        ];

        $validator = Validator::make($request->all(), $rules);

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

        // Simpan file foto jika ada
        $photoName = null;
    }
}
```

## Hasil

Edit Data User

Level Pengguna

Manager

Username

manager

Nama

Manager

Password

Abaikan jika tidak ingin ubah password

Foto Profil

Choose File Portrait Of a Confident Young Smart Looking Man \_ Premium AI-generated image.jpeg

ID	Foto	Username	Nama	Level Pengguna	Aksi
1		admin	Administrator	Administrator	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
2		manager	Manager	Manager	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
3		staff	Staff/Kasir	Staff/Kasir	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

Showing 1 to 3 of 3 entries

[Previous](#) [1](#) [Next](#)



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

---

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

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