



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						

	A	B	C	D	E	F	G	H	I	J	K
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.

Name	Date modified	Type	Size
adminlte	30/03/2025 13:54	File folder	
.htaccess	13/02/2024 09:23	HTACCESS File	1 KB
favicon.ico	13/02/2024 09:23	ICO File	0 KB
index.php	13/02/2024 09:23	PHP Source File	2 KB
robots.txt	13/02/2024 09:23	Text Source File	1 KB
template_barang.xlsx	30/03/2025 14:05	Microsoft Excel W...	10 KB

- 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
```

```
Minggu, 8 > Jobsheet8 > PWL_POS > resources > views > barang > index.blade.php > {} "index.blade.php"  
1 @extends('layouts.template')  
2  
3 @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('/barang/import') }}')" class="btn btn-info">Import Barang</button>  
9 <a href="{{ url('/barang/create') }}" class="btn btn-primary">Tambah  
10 Data</a>  
11 <button onclick="modalAction('{{ url('/barang/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>  
12 </div>  
13 </div>  
14 <div class="card-body">  
15 <!-- untuk Filter data -->  
16 <div id="filter" class="form-horizontal filter-date 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-  
21 label">Filter</label>  
22 <div class="col-md-3">  
23 <select name="filter_kategori" class="form-control form-  
24 control-sm filter_kategori">  
25 <option value="">- Semua -</option> @foreach($kategori as $1)  
26 <option value="{{ $1->kategori_id }}">{{ $1-&br/>27 <option value="{{ $1->kategori_id }}">{{ $1-&br/>28 </foreach>  
29 </select>  
30 <small class="form-text text-muted">Kategori Barang</small>  
31 </div>  
32 </div>  
33 </div>  
34 </div>  
35 <div> @if(session('success'))  
36 <div class="alert alert-success">{{ session('success') }}</div> @endif  
37 @if(session('error'))  
38 <div class="alert alert-danger">{{ session('error') }}</div> @endif  
39 <table class="table-bordered table-sm table-striped table-hover" id="table-barang">
```

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>
```




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>

```
Minggu, 8 > Jobsheet8 > PWL_POS > resources > views > barang > import.blade.php > script > ready() callback > submitHandler > success
1 <form action="{{ url('/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 aria-hidden="true">&times;</span></button>
8 </div>
9 <div class="modal-body">
10 <div class="form-group">
11 <label>Download Template</label>
12 <a href="{{ asset('template_barang.xlsx') }}" class="btn btn-info btn-sm download"><i class="fa fa-file-excel"></i>Download</a>
13 <small id="error-kategori_id" class="error-text form-text text-danger"></small>
14 </div>
15 <div class="form-group">
16 <label>Pilih File</label>
17 <input type="file" name="file_barang" id="file_barang" class="form-control" required>
18 <small id="error-file_barang" class="error-text form-text text-danger"></small>
19 </div>
20 </div>
21 <div class="modal-footer">
22 <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
23 <button type="submit" class="btn btn-primary">Upload</button>
24 </div>
25 </div>
26 </div>
27 </form>
28 <script>
29 $(document).ready(function() {
30 $("#form-import").validate({
31 rules: {
32 file_barang: {
33 required: true,
34 extension: ".xlsx"
35 }
36 },
37 });
38 </script>
```

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



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

```
192
193
194 Route::group(['prefix' => 'barang'], function () {
195     Route::middleware(['authorize:STF'])->group(function () {
196         Route::get('/', [BarController::class, 'index']); // menampilkan halaman awal barang
197         Route::post('/list', [BarController::class, 'list']); // menampilkan data barang dalam bentuk json untuk datatable
198         Route::get('/create', [BarController::class, 'create']); // menampilkan halaman form tambah barang
199         Route::post('/', [BarController::class, 'store']); // menyimpan data barang baru
200         Route::get('/create_ajax', [BarController::class, 'create_ajax']); // menampilkan halaman form tambah user Ajax
201         Route::post('/ajax', [BarController::class, 'store_ajax']); // menyimpan data user baru Ajax
202         Route::get('/{id}', [BarController::class, 'show']); // menampilkan detail barang
203         Route::get('/{id}/edit', [BarController::class, 'edit']); // menampilkan halaman form edit barang
204         Route::put('/{id}', [BarController::class, 'update']); // menyimpan perubahan data barang
205         Route::get('/{id}/edit_ajax', [BarController::class, 'edit_ajax']); // Menampilkan halaman form edit user Ajax
206         Route::put('/{id}/update_ajax', [BarController::class, 'update_ajax']); // Menyimpan perubahan data user Ajax
207         Route::get('/{id}/delete_ajax', [BarController::class, 'confirm_ajax']); // Untuk tampilan form confirm delete user Ajax
208         Route::delete('/{id}/delete_ajax', [BarController::class, 'delete_ajax']); // Untuk hapus data user Ajax
209         Route::get('/barang/import', [BarController::class, 'import']); // ajax form upload excel
210         Route::post('/barang/import_ajax', [BarController::class, 'import_ajax']); // ajax import excel
211         Route::delete('/{id}', [BarController::class, 'destroy']); // menghapus data barang
```

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 phpoffice/phpspreadsheet
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS DEVICES
yajra/laravel-datatables-editor ..... DONE
yajra/laravel-datatables-fractal ..... DONE
yajra/laravel-datatables-html ..... DONE
yajra/laravel-datatables-oracle ..... DONE

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

Warning: Module "mysqli" is already loaded in Unknown on line 0
Warning: PHP Startup: Unable to load dynamic library 'php_pdo_sqlsrv_81_ts_64.dll' (tried: C:/laragon/bin/php/php-8.1.10-Min32-vs16-x64/ext/php_pdo_sqlsrv_81_ts_64.dll (The specified module could not be found), C:/laragon/bin/php/php-8.1.10-Min32-vs16-x64/ext/p
hp_pdo_sqlsrv_81_ts_64.dll (The specified module could not be found)) in Unknown on line 0
[Error] No publishable resources for tag [laravel-assets].

Found 1 security vulnerability advisory affecting 1 package.
Run "composer audit" for a full list of advisories.
Using version ^4.1 for phpoffice/phpspreadsheet
PS C:\laragon\www\PA_2023\Minggu_3\JobSheet3\PA1_P05>
```

7. Selanjutnya kita modifikasi file **BarController.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('/');
```



```
Minggu 8 > Jobsheet8 > PwL_POS > app > Http > Controllers > BarangController.php > ...
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\LevelModel;
6 use App\Models\BarangModel;
7 use App\Models\KategoriModel;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Validator;
10 use PhpOffice\PhpSpreadsheet\IOFactory;
11 use Vajra\DataTables\Facades\DataTables;
12
13 class BarangController extends Controller
14 {
15     public function index()
16     {
17         $activeMenu = 'barang';
18         $breadcrumb = (object) [
19             'title' => 'Data Barang',
20             'list' => ['Home', 'Barang']
21         ];
22
23         $kategori = KategoriModel::select('kategori_id', 'kategori_nama')->get();
24
25         return view('barang.index', compact('activeMenu', 'breadcrumb', 'kategori'));
26     }
27
28     public function list(Request $request)
29     {
30         $barang = BarangModel::with('kategori')
31             ->select('barang_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual', 'kategori_id');
32
33         if ($request->filled('filter_kategori')) {
34             $barang->where('kategori_id', $request->filter_kategori);
35         }
36
37         return DataTables::of($barang)
38             ->addIndexColumn()
39             ->addColumn('aksi', function ($barang) {
40                 return 'button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id . '/show_ajax') . '\')"' class="btn btn-info btn-sm">Detail</button>
41                     button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id . '/edit_ajax') . '\')"' class="btn btn-warning btn-sm">Edit</button>
42                     button onclick="modalAction(\'' . url('/barang/' . $barang->barang_id . '/delete_ajax') . '\')"' class="btn btn-danger btn-sm">Hapus</button>";
43             })
44             ->rawColumns(['aksi'])
45             ->make(true);
46     }
47
48     public function create_ajax()
49     {
50     }
51 }
```

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



Import Data Barang

Download Template

Download

Pilih File

Browse...

No file selected.

Batal

Upload

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Log Out

Logout

Home

Contact

Data Barang

Daftar Barang

Filter

Show 10 entries

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Keripik Kentang	4.500	7.000	Alat Tulis	Detail Edit Hapus
2	BRG002	Donat Coklat	6.000	8.500	Alat Tulis	Detail Edit Hapus
3	BRG003	Susu Kotak	3.500	5.000	Pakaian	Detail Edit Hapus
4	BRG004	Mie Instan	4.000	5.500	Pakaian	Detail Edit Hapus
5	BRG005	Headset Bluetooth	120.000	180.000	Peralatan Rumah	Detail Edit Hapus
6	BRG006	Monitor LED	300.000	450.000	Peralatan Rumah	Detail Edit Hapus
7	BRG007	Keset Karet	18.000	25.000	Makanan	Detail Edit Hapus

Home / Barang

[Barang](#) [Tambah Data](#) [Tambah Data \(Ajax\)](#)

Search:

Import Data Barang

Download Template

Download

Pilih File

Choose File

No file chosen

Batal

Upload

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

PWL - Starter Code

Search

Dashboard

Data Pengguna

Level User

Data User

Data Barang

Kategori Barang

Data Barang

Data Transaksi

Stok Barang

Transaksi Penjualan

Log Out

Logout

Home

Contact

Data Barang

Daftar Barang

Filter

Show 10 entries

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Keripik Kentang	4.500	7.000	Alat Tulis	Detail Edit Hapus
2	BRG002	Donat Coklat	6.000	8.500	Alat Tulis	Detail Edit Hapus
3	BRG003	Susu Kotak	3.500	5.000	Pakaian	Detail Edit Hapus
4	BRG004	Mie Instan	4.000	5.500	Pakaian	Detail Edit Hapus
5	BRG005	Headset Bluetooth	120.000	180.000	Peralatan Rumah	Detail Edit Hapus
6	BRG006	Monitor LED	300.000	450.000	Peralatan Rumah	Detail Edit Hapus
7	BRG007	Keset Karet	18.000	25.000	Makanan	Detail Edit Hapus

Home / Barang

[Import Barang](#) [Tambah Data](#) [Tambah Data \(Ajax\)](#)

Search:

Berhasil

Data berhasil diimport

OK



Tugas 1 – Implementasi File Upload untuk Import Data:

1. Silahkan implementasikan praktikum 1 pada project kalian masing-masing untuk semua menu
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - a. Membuat data barang sesuai dengan tabel m_barang di excel dengan format xlsx kemudian disimpan di folder public pada project kita
 - b. Melakukan modifikasi tampilan view pada index.blade pada folder barang untuk menambahkan tombol upload untuk import barang
 - c. Kemudian membuat file blade baru di folder barang dengan nama import.blade untuk menampilkan tampilan pada saat import file
 - d. Melakukan install library phpoffice/phpspreadsheet melalui terminal
 - e. Selanjutnya melakukan modif pada file barangcontroller untuk mengatur logika pada saat import file
 - f. Selanjutnya mencoba untuk import file melalui halaman data barang kemudian klik import barang menggunakan file excel yang sudah kita buat diawal tadi
3. Submit kode untuk impementasi praktikum 1 pada repository github kalian.

B. Export Excel di Laravel

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

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

Praktikum 2 – Export Data ke Excel

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

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

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

Kita ubah dengan

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



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

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
```

```
121 Route::delete('/id/delete_ajax',[BarangController::class,'delete_ajax']); // untuk hapus data user Ajax
122 Route::get('/import',[BarangController::class,'import']);
123 Route::post('/import_ajax',[BarangController::class,'import_ajax']); // ajax import excel
124 Route::get('/export_excel',[BarangController::class,'export_excel']); // export excel
125 Route::delete('/id/delete',[BarangController::class,'delete']); // hapus data barang
```

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();
```

```
187 public function export_excel() {
188
189     // ambil data barang yang akan di export
190     $barang = BarangModel::select('kategori_id','barang_kode','barang_nama','harga_beli','harga_jual')
191         ->orderBy('kategori_id')
192         ->with('kategori')
193         ->get();
194 }
```

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
```

```
197 // Load library excel
198
199 $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
200 $sheet = $spreadsheet->getActiveSheet(); // ambil sheet yang aktif
201
202 $sheet->setCellValue('A1','No');
203 $sheet->setCellValue('B1','Kode Barang');
204 $sheet->setCellValue('C1','Nama Barang');
205 $sheet->setCellValue('D1','Harga Beli');
206 $sheet->setCellValue('E1','Harga Jual');
207 $sheet->setCellValue('F1','Kategori');
208
209 $sheet->getStyle('A1:F1')->getFont()->setBold(true); // bold header
210
```



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 }
```

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

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

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



```
222 foreach (range('A', 'F') as $columnID) {  
223     $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size untuk kolom  
224 }  
225
```

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
```

```
222 $sheet->setTitle('Data Barang'); // set title sheet  
223  
224 $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');  
225 $filename = 'Data Barang '.date('Y-m-d H:i:s').'.xlsx';  
226  
227 header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');  
228 header('Content-Disposition: attachment;filename="'.$filename.'");  
229 header('Cache-Control: max-age=0');  
230 header('Cache-Control: max-age=1');  
231 header('Expires: Mon, 26 Jul 1997 05:00:00 GMT');  
232 header('Last-Modified: ' . gmdate('D, d M Y H:i:s') . ' GMT');  
233 header('Cache-Control: cache, must-revalidate');  
234 header('Pragma: public');  
235  
236 $writer->save('php://output');  
237 exit;
```

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

	A	B	C	D	E	F
1	No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
2	1	SBK-001	Beras Cap Jago (5kg)	65000	68000	Sembako
3	2	SBK-002	Beras Bramo Cap Lele	80000	83000	Sembako
4	3	SBK-003	Telur Omega (10 butir)	22000	25000	Sembako
5	4	SNK-001	Happy Toss	10500	11000	Makanan Ringan
6	5	SNK-002	Oreo	7200	7800	Makanan Ringan
7	6	SNK-003	Sari Roti	11500	12500	Makanan Ringan
8	7	MND-001	Sabun Lifebouy	4250	5000	Peralatan Mandi
9	8	MND-002	Pasta Gigi Pepsoden	6750	7500	Peralatan Mandi
10	9	MND-003	Shampo Pantene	17500	18500	Peralatan Mandi
11	10	BAY-001	Susu SGM Coklat 900gr	92500	95000	Keperluan Bayi
12	11	BAY-002	Popok Mamy Poko	56000	58000	Keperluan Bayi
13	12	BAY-003	Baju Bayi 2th	89000	92500	Keperluan Bayi
14	13	MNM-001	Aqua 600ml	3700	4500	Minuman
15	14	MNM-002	Le Mineral	3500	4000	Minuman
16	15	MNM-003	Cleo 600ml	3750	4300	Minuman



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>

The screenshot shows an Excel spreadsheet with the following data:

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Keripik Kentang	4500	7000	Alat Tulis
2	BRG002	Donat Coklat	6000	8500	Alat Tulis
3	BRG012	Bubur Ayam	10000	13000	Alat Tulis
4	BRG014	Mie Ayam	8000	12000	Alat Tulis
5	BRG003	Susu Kotak	3500	5000	Pakaian
6	BRG004	Mie Instan	4000	5500	Pakaian
7	BRG013	Sarung	20000	30000	Pakaian
8	BRG005	Headset Bluetooth	120000	180000	Peralatan Rumah
9	BRG006	Monitor LED	300000	450000	Peralatan Rumah
10	BRG015	Chocolatos	5000	7000	Peralatan Rumah
11	BRG007	Keset Karet	18000	25000	Makanan
12	BRG008	Ember Plastik	25000	35000	Makanan
13	BRG016	Semangka	25000	30000	Makanan
14	BRG009	Celana Jeans	60000	75000	Minuman
15	BRG010	Sweater Rajut	120000	160000	Minuman

Tugas 2 – Implementasi File Export Excel:

1. Silahkan implementasikan praktikum 2 pada project kalian masing-masing untuk semua menu
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - a. Melakukan modifikasi pada tampilan index.blade di folder barang dengan menambahkan tombol export barang dan melakukan hapus tombol pada tambah data karena kita sudah menggunakan tombol tambah data ajax
 - b. Kemudian kita melakukan tambah pada folder route dibagian web untuk menambahkan url untuk mengarah ke export file
 - c. Lalu kita menambahkan fungsi export_excel() pada file barang controller
 - d. Kemudian kita masukkan library spreadsheet pada bagian header di file barang controller
 - e. Selanjutnya kita menambahkan kode looping data yang berfungsi untuk mendapatkan data dari database
 - f. Melakukan set lebar pada tiap kolom di excel
 - g. Selanjutnya melakukan tambah kode untuk proses export excel dan proses untuk dapat di download oleh user
 - h. Jika telah selesai pada data barang bisa melakukan export file berupa file excel berisikan data barang yang ada
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.

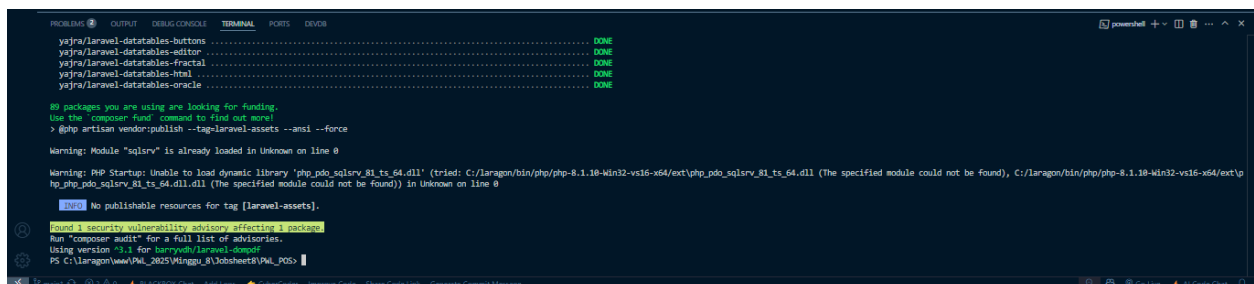
Untuk mengimplementasikan export ke PDF di Laravel, salah satu paket yang paling populer adalah **dompdf**. Dengan menggunakan paket **dompdf**, Kita dapat dengan mudah mengubah tampilan HTML menjadi PDF yang dapat disesuaikan sesuai kebutuhan. Fitur ini sangat bermanfaat dalam berbagai konteks, seperti 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:

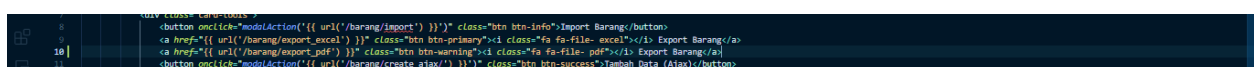
1. Kita lakukan proses instalasi library dompdf terlebih dahulu dengan mengetikkan perintah pada terminal/CMD

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



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
```




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>

```
123 Route::post('/import_ajax', [BarangController::class, 'import_ajax']); // ajax import excel
124 Route::get('/export_excel', [BarangController::class, 'export_excel']); // export excel
125 | Route::get('/export_pdf', [BarangController::class, 'export_pdf']); // export pdf
126 Route::delete('/{id}', [BarangController::class, 'destroy']); // menghapus data barang
```

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`



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;
    }
  </style>
</head>
<body>
  <table border="1">
    <tr>
      <td>
        <div class="d-block">
          <img alt="Logo of Politeknik Negeri Malang" data-bbox="125 53 244 141"/>
        </div>
      </td>
      <td>
        KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
        <br>
        POLITEKNIK NEGERI MALANG
        <br>
        JURUSAN TEKNOLOGI INFORMASI
        <br>
        Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
        <br>
        Telp. (0341) 404424 – 404425, Fax (0341) 404420
        <br>
        http://www.polinema.ac.id
      </td>
    </tr>
  </table>
  <div class="text-right">
    <div class="d-block">
      <img alt="Logo of Politeknik Negeri Malang" data-bbox="125 53 244 141"/>
    </div>
  </div>
</body>
</html>
```



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



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

```
1 <html>
2 <head>
3
4 <style>
5 border-all,
6
7
8 </style>
9 </head>
10
11 <body>
12 <table class="border-bottom-header">
13 <tr>
14 <td width="15%" class="text-center">
15 </td>
16 <td width="85%">
17 <span class="text-center d-block font-11 font-bold mb-1">KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI</span>
18 <span class="text-center d-block font-13 font-bold mb-1">POLITEKNIK NEGERI
19 MALANG</span>
20 <span class="text-center d-block font-10">Jl. Soekarno-Hatta No. 9 Malang
21 65142</span>
22 <span class="text-center d-block font-10">Telepon (0341) 404424 Pes. 101-
23 105, 0341-404420, Fax. (0341) 404420</span>
24 <span class="text-center d-block font-10">Laman: www.polinema.ac.id</span>
25 </td>
26 </tr>
27 </table>
28 <h3 class="text-center">LAPORAN DATA BARANG</h3>
29 <table class="border-all">
30 <thead>
31 <tr>
32 <th class="text-center">No</th>
33 <th class="text-center">Kode Barang</th>
34 <th class="text-center">Nama Barang</th>
35 <th class="text-right">Harga Beli</th>
36 <th class="text-right">Harga Jual</th>
37 <th class="text-center">Kategori</th>
38 </tr>
39 </thead>
40 <tbody>
41 @foreach($barang as $b)
42 <tr>
43 <td class="text-center">{{ $loop->iteration }}</td>
44 <td class="text-center">{{ $b->barang_kode }}</td>
45 <td class="text-center">{{ $b->barang_nama }}</td>
46 <td class="text-right">{{ number_format($b->harga_beli, 0, ',', '.') }}</td>
47 <td class="text-right">{{ number_format($b->harga_jual, 0, ',', '.') }}</td>
48 <td class="text-center">{{ $b->kategori->kategori_nama }}</td>
49 </tr>
50 @endforeach
```

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



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

LAPORAN DATA BARANG

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Keripik Kentang	4.500	7.000	Alat Tulis
2	BRG002	Donat Coklat	6.000	8.500	Alat Tulis
3	BRG012	Bubur Ayam	10.000	13.000	Alat Tulis
4	BRG014	Mie Ayam	8.000	12.000	Alat Tulis
5	BRG003	Susu Kotak	3.500	5.000	Pakaian
6	BRG004	Mie Instan	4.000	5.500	Pakaian
7	BRG013	Sarung	20.000	30.000	Pakaian
8	BRG005	Headset Bluetooth	120.000	180.000	Peralatan Rumah
9	BRG006	Monitor LED	300.000	450.000	Peralatan Rumah
10	BRG015	Chocolatos	5.000	7.000	Peralatan Rumah
11	BRG007	Keset Karet	18.000	25.000	Makanan
12	BRG008	Ember Plastik	25.000	35.000	Makanan
13	BRG016	Semangka	25.000	30.000	Makanan
14	BRG009	Celana Jeans	60.000	75.000	Minuman
15	BRG010	Sweater Rajut	120.000	160.000	Minuman



Tugas 3 – Implementasi Export PDF pada Laravel :

1. Silahkan implementasikan export pdf pada project kalian masing-masing untuk semua menu
2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - a. Melakukan instalasi library dompdf pada terminal
 - b. Melakukan modif di file index.blade untuk tombol export barang
 - c. Selanjutnya modif di file route untuk menambahkan url export_pdf
 - d. Kemudian menambahkan fungsi export_pdf pada barang controller untuk mengatur logika nya
 - e. Selanjutnya membuat file baru pada folder barang yaitu export_pdf.blade untuk melakukan edit kode untuk tampilan export_pdf nantinya
 - f. Selanjutnya kita coba untuk melakukan unduh pada export_barang berupa file pdf dan didalamnya berisi data barang yang ada pada data barang
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

The screenshot displays a web application interface. On the left, a sidebar menu is visible with options like 'Dashboard', 'Data Pengguna', 'Level User', 'Data User' (highlighted), 'Data Barang', 'Kategori Barang', 'Data Barang', 'Data Transaksi', 'Stok Barang', 'Transaksi Penjualan', 'Log Out', and 'Logout'. The main content area shows a 'Daftar' (List) view of users. A modal window titled 'Edit Data User' is open, allowing editing of user details. The modal includes a dropdown for 'Level Pengguna' (set to Administrator), text inputs for 'Username' (admin) and 'Nama' (Administrator), and a password field. Below the password field is a note: 'Abaikan jika tidak ingin ubah password'. The 'Foto Profil' section features a 'Choose File' button and the text 'No file chosen'. At the bottom of the modal are 'Batal' (Cancel) and 'Simpan' (Save) buttons. The background table lists users with columns for ID, Name, Username, Password, and Profile Photo, with actions like 'Detail', 'Edit', and 'Hapus' (Delete) for each row.

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

*** Sekian, dan selamat belajar ***