

Nama : Dini Elminingtyas
Kelas/No.absen : SIB 2B / 07
NIM : 2341760180

PEMROGRAMAN WEB LANJUT
JOBSCHEET 08
FILE IMPORT AND EXPORT

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 Mamypoko	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

⇒ Database

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	KAT001	Electronics	NULL	2025-03-23 15:26:37
2	KAT002	Pakaian	NULL	NULL
3	KAT003	Alat Tulis	NULL	NULL
4	KAT004	Makanan	NULL	NULL
5	KAT005	Minuman	NULL	NULL
7	CML	Cemilan	NULL	NULL
8	MNR	Minuman Ringan	NULL	NULL
12	SPTU	Sepatu	2025-03-13 09:02:21	2025-03-13 09:02:21

barang_id	barang_kode	barang_nama	harga_beli	harga_jual	kategori_id	created_at	updated_at
1	BRG001	Laptop	7000000	10000000	1	NULL	2025-03-23 15:35:40
2	BRG002	Smartphone	3000000	4000000	1	NULL	NULL
3	BRG003	T-Shirt	50000	100000	2	NULL	NULL
4	BRG004	Celana Jeans	150000	250000	2	NULL	NULL
5	BRG005	Pulpen	5000	7500	3	NULL	NULL
6	BRG006	Buku Tulis	3000	5000	3	NULL	NULL
7	BRG007	Mie Instan	2500	3500	4	NULL	NULL
8	BRG008	Air Mineral	1500	3000	5	NULL	NULL
9	BRG009	Keyboard	150000	250000	1	NULL	NULL
10	BRG010	Mouse	100000	150000	1	NULL	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						

⇒ File excel

	A	B	C	D	E
1	kategori_id	kategori_kode	kategori_nama	created_at	updated_at
2	1	KAT001	Electronics	NULL	2025-03-23 15:26:37
3	2	KAT002	Pakaian	NULL	NULL
4	3	KAT003	Alat Tulis	NULL	NULL
5	4	KAT004	Makanan	NULL	NULL
6	5	KAT005	Minuman	NULL	NULL
7	7	CML	Cemilan	NULL	NULL
8	8	MNR	Minuman Ringan	NULL	NULL
9	12	SPTU	Sepatu	2025-03-13 09:02:21	2025-03-13 09:02:21
10					

A	B	C	D	E	F	G	H
barang_id	barang_kode	barang_nama	harga_beli	harga_jual	kategori_id	created_at	updated_at
1	BRG001	Laptop	7000000	10000000	1	NULL	2025-03-23 15:35:40
2	BRG002	Smartphone	3000000	4000000	1	NULL	NULL
3	BRG003	T-Shirt	50000	100000	2	NULL	NULL
4	BRG004	Celana Jeans	150000	250000	2	NULL	NULL
5	BRG005	Pulpen	5000	7500	3	NULL	NULL
6	BRG006	Buku Tulis	3000	5000	3	NULL	NULL
7	BRG007	Mie Instan	2500	3500	4	NULL	NULL
8	BRG008	Air Mineral	1500	3000	5	NULL	NULL
9	BRG009	Keyboard	150000	250000	1	NULL	NULL
10	BRG010	Mouse	100000	150000	1	NULL	NULL

3. Selanjutnya kita modifikasi view pada `barang/index.blade.php` untuk bisa menambahkan tombol menambah form untuk upload untuk import data barang

```

@extends('layouts.template')

@section('content')
    <div class="card">
        <div class="card-header">
            <h3 class="card-title">Daftar barang</h3>
            <div class="card-tools">
                <button onclick="modalAction('{{ url('/barang/import') }}'") class="btn btn-info">Import Barang</button>
                <a href="{{ url('/barang/create') }}" class="btn btn-primary">Tambah Data</a>
                <button onclick="modalAction('{{ url('/barang/create_ajax') }}'") class="btn btn-success">Tambah Data (Ajax)</button>
            </div>
        </div>
        <div class="card-body">
            <!-- untuk Filter data -->
            <div id="filter" class="form-horizontal filter-date p-2 border-bottom mb-2">
                <div class="row">
                    <div class="col-md-12">
                        <div class="form-group form-group-sm row text-sm mb-0">
                            <label for="filter_date" class="col-md-1 col-form-label">Filter</label>
                            <div class="col-md-3">
                                <select name="filter_kategori" class="form-control form-control-sm filter_kategori">
                                    <option value="">- Semua -</option>
                                    @foreach($kategori as $l)
                                        <option value="{{ $l->kategori_id }}>{{ $l->kategori_nama }}</option>
                                    @endforeach
                                </select>
                                <small class="form-text text-muted">Kategori Barang</small>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
            @if(session('success'))
                <div class="alert alert-success">{{ session('success') }}</div>
            @endif
            @if(session('error'))
                <div class="alert alert-danger">{{ session('error') }}</div>
            @endif
        </div>
    </div>

```

⇒ Source code

```

resources > views > barang > index.blade.php > () *1
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 Data</a>
10          <button onclick="modalAction('{{ url('/barang/create_ajax') }}')" class="btn btn-success">Tambah Data (Ajax)</button>
11        </div>
12      </div>
13
14      <div class="card-body">
15        <!-- Filter Data -->
16        <div id="filter" class="form-horizontal filter-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_kategori" class="col-md-1 col-form-label">Filter</label>
21                <div class="col-md-3">
22                  <select name="filter_kategori" class="form-control form-control-sm filter_kategori">
23                    <option value=""> Semua </option>
24                    @foreach($kategori as $l)
25                      <option value="{{ $l->kategori_id }}>{{ $l->kategori_nama }}</option>
26                    @endforeach
27                  </select>
28                  <small class="form-text text-muted">Kategori Barang</small>
29                </div>
30              </div>
31            </div>
32          </div>
33        </div>
34
35        @if(session('success'))
36          <div class="alert alert-success">{{ session('success') }}</div>
37        @endif
38        @if(session('error'))
39          <div class="alert alert-danger">{{ session('error') }}</div>
40        @endif

```

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">

```

⇒ [Source code](#)

```

resources > views > barang > import.blade.php > script
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">
8             <span aria-hidden="true">&times;</span>
9           </button>
10        </div>
11        <div class="modal-body">
12          <div class="form-group">
13            <label>Download Template</label>
14            <a href="{{ asset('template_barang.xlsx') }}" class="btn btn-info btn-sm" download>
15              <i class="fa fa-file-excel"></i> Download
16            </a>
17            <small id="error-kategori_id" class="error-text form-text text-danger"></small>
18          </div>
19          <div class="form-group">
20            <label>Pilih File</label>
21            <input type="file" name="file_barang" id="file_barang" class="form-control" required>
22            <small id="error-file_barang" class="error-text form-text text-danger"></small>
23          </div>
24        </div>
25        <div class="modal-footer">
26          <button type="button" data-dismiss="modal" class="btn btn-warning">Batal</button>
27          <button type="submit" class="btn btn-primary">Upload</button>
28        </div>
29      </div>
30    </div>
31  </form>

```

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

```

⇒ Source code

```

147
148      //route ajax
149      Route::get('/create_ajax', [BarangController::class, 'create_ajax']);
150      Route::post('/ajax', [BarangController::class, 'store_ajax']);
151      Route::get('/{id}/edit_ajax', [BarangController::class, 'edit_ajax']);
152      Route::put('/{id}/update_ajax', [BarangController::class, 'update_ajax']);
153      Route::get('/{id}/delete_ajax', [BarangController::class, 'confirm_ajax']);
154      Route::delete('/{id}/delete_ajax', [BarangController::class, 'delete_ajax']);
155      Route::get('/import', [BarangController::class, 'import']); // ajax form upload excel
156      Route::post('/import_ajax', [BarangController::class, 'import_ajax']); // ajax import excel
157

```

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

```
composer require phoffice/phpspreadsheet
```

⇒ Source code

```
PS D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (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
- Locking composer/pcre (3.3.2)
- Locking maennchen/zipstream-php (3.1.1)
- Locking markbaker/complex (3.0.2)
- Locking markbaker/matrix (3.0.1)
- Locking phoffice/phpspreadsheet (4.1.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
```

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);
        }

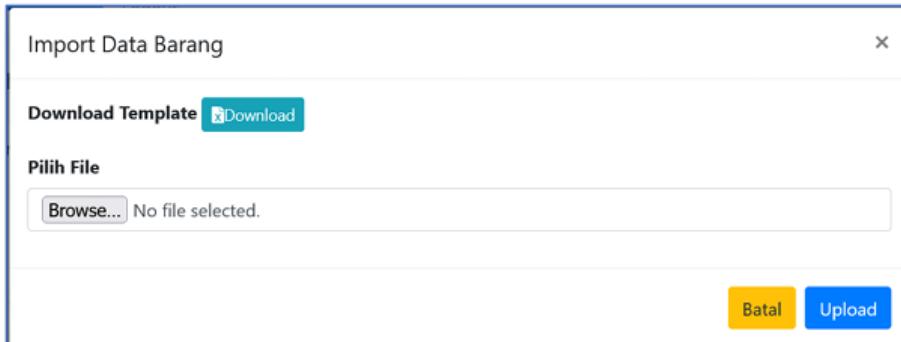
        return DataTables::of($barang)
            ->addIndexColumn()
            ->addColumn('aksi', function ($barang) { // menambahkan kolom aksi
```

⇒ Source code

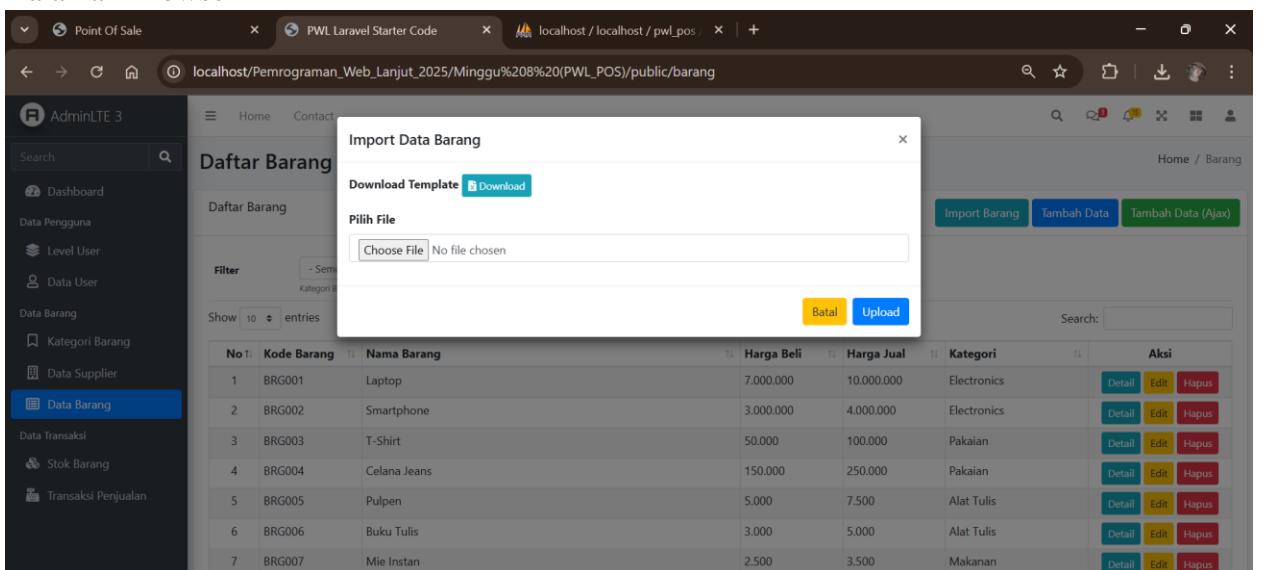


```
app > Http > Controllers > BarangController.php > BarangController > import_ajax
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\LevelModel;
6 use App\Models\BarangModel;
7 use App\Models\KategoriModel;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Hash;
10 use Yajra\DataTables\Facades\DataTables;
11 use PhpOffice\PhpSpreadsheet\IOFactory;
12 use Illuminate\Support\Facades\Validator;
13
14 class BarangController extends Controller
15 {
16     public function index()
17     {
18         $breadcrumb = (object) [
19             'title' => 'Daftar Barang',
20             'list' => ['Home', 'Barang']
21         ];
22
23         $page = (object) [
24             'title' => 'Daftar barang yang terdaftar dalam sistem'
25         ];
26
27         $activeMenu = 'barang';
28         $kategori = KategoriModel::all();
29
30         return view('barang.index', ['breadcrumb' => $breadcrumb, 'page' => $page, 'kategori' => $kategori, 'activeMenu' => $activeMenu]);
31     }
32     public function list(Request $request)
33     {
34         $barangs = BarangModel::select('barang_id', 'kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')->with('kategori');
35         if (!empty($request->kategori_id)) {
36             $barangs->where('kategori_id', $request->kategori_id);
37         }
38     }
39 }
```

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



⇒ Halaman Browser



No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Laptop	7.000.000	10.000.000	Electronics	[Detail] [Edit] [Hapus]
2	BRG002	Smartphone	3.000.000	4.000.000	Electronics	[Detail] [Edit] [Hapus]
3	BRG003	T-Shirt	50.000	100.000	Pakaian	[Detail] [Edit] [Hapus]
4	BRG004	Celana Jeans	150.000	250.000	Pakaian	[Detail] [Edit] [Hapus]
5	BRG005	Pulpen	5.000	7.500	Alat Tulis	[Detail] [Edit] [Hapus]
6	BRG006	Buku Tulis	3.000	5.000	Alat Tulis	[Detail] [Edit] [Hapus]
7	BRG007	Mie Instan	2.500	3.500	Makanan	[Detail] [Edit] [Hapus]

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

⇒ Halaman Browser

The screenshot shows a web application interface for managing inventory. On the left, there's a sidebar with navigation links like Dashboard, Data Pengguna, Level User, Data User, Data Barang, Kategori Barang, Data Supplier, and Data Barang (which is currently selected). The main content area is titled 'Daftar Barang' and shows a table of products with columns for ID, Kode Barang, Nama Barang, Harga Jual, Kategori, and Aksi (Actions). A modal window is overlaid on the page, indicating that data has been successfully imported. The table data is as follows:

No	Kode Barang	Nama Barang	Harga Jual	Kategori	Aksi
1	BRG001	Laptop	10.000.000	Electronics	[Detail] [Edit] [Hapus]
2	BRG002	Smartphone	4.000.000	Electronics	[Detail] [Edit] [Hapus]
3	BRG003	T-Shirt	100.000	Pakaian	[Detail] [Edit] [Hapus]
4	BRG004	Celana Jeans	250.000	Pakaian	[Detail] [Edit] [Hapus]
5	BRG005	Pulpen	7.500	Alat Tulis	[Detail] [Edit] [Hapus]
6	BRG006	Buku Tulis	5.000	Alat Tulis	[Detail] [Edit] [Hapus]
7	BRG007	Mie Instan	3.500	Makanan	[Detail] [Edit] [Hapus]
8	BRG008	Air Mineral	1.500	Minuman	[Detail] [Edit] [Hapus]
9	BRG009	Keyboard	150.000	Electronics	[Detail] [Edit] [Hapus]
10	BRG010	Mouse	100.000	Electronics	[Detail] [Edit] [Hapus]

Tugas 1 - Implementasi File Upload untuk Import Data:

1. Silahkan implementasikan praktikum 1 pada project kalian masing-masing untuk semua menu
- ⇒ Implementasi project POS
- a. Level User

The screenshot shows a web application interface for managing user levels. On the left, there's a sidebar with navigation links like Dashboard, Data Pengguna, Level User (which is currently selected), Data User, Data Barang, Kategori Barang, Data Supplier, and Data Barang. The main content area is titled 'Daftar Level' and shows a table of level data with columns for ID, Kode Level, and Name. A modal window is overlaid on the page, titled 'Import Data Level', with a file selection input field and 'Batal' and 'Upload' buttons. The table data is as follows:

ID	Kode Level	Name	Actions
1	ADM	Manager	[Detail] [Edit] [Delete]
2	MNG	Manager	[Detail] [Edit] [Delete]
3	STF	Staff/Kasir	[Detail] [Edit] [Delete]
4	CUST	Customer	[Detail] [Edit] [Delete]

b. Data User

The screenshot shows the 'Daftar User' (User List) page. On the left, there's a sidebar with various menu items like Admin, Dashboard, Data Pengguna, Level User, Data User (which is selected), Data Barang, Kategori Barang, Data Supplier, Data Transaksi, Stok Barang, and Transaksi Penjualan. The main area displays a table with columns: ID, Username, Nama, Level Pengguna, and Aksi (Actions). The table contains 5 entries:

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	[Detail, Edit, Delete]
2	manager	Manager	Staff/Kasir	[Detail, Edit, Delete]
3	staff	Staff/Kasir	Staff/Kasir	[Detail, Edit, Delete]
4	customer	Customer	Customer	[Detail, Edit, Delete]
5	cashier	kasir	Manager	[Detail, Edit, Delete]

Below the table, there's a message 'Showing 1 to 5 of 5 entries'. Overlaid on the page is a modal titled 'Import Data User' with a 'Choose File' input field and 'Upload' button.

c. Kategori Barang

The screenshot shows the 'Daftar Kategori' (Category List) page. The sidebar is identical to the previous one. The main area displays a table with columns: ID, Kode Kategori, and Nama. The table contains 4 entries:

ID	Kode Kategori	Nama
1	CTGY01	
2	CTGY02	Beauty & Health
3	CTGY03	Home Care
4	CTGY04	Baby & Kids

Below the table, there's a message 'Showing 1 to 4 of 4 entries'. Overlaid on the page is a modal titled 'Import Data Kategori' with a 'Choose File' input field and 'Upload' button.

d. Data Supplier

The screenshot shows the 'Daftar Supplier' (Supplier List) page. The sidebar is identical to the previous ones. The main area displays a table with columns: ID, Kode Supplier, Supplier, and Alamat. The table contains 5 entries:

ID	Kode Supplier	Supplier	Alamat
1	SUP001	Supplier b	Jl. Pahlawan No. 5, Surabaya
2	SUP002	Supplier C	Jl. Raya No. 10, Bandung
3	SUP003	Supplier D	Jl. Sejahtera No. 8, Yogyakarta
4	SUP004	Supplier E	Jl. Sentosa No. 3, Bali
5	SUP005		

Below the table, there's a message 'Showing 1 to 5 of 5 entries'. Overlaid on the page is a modal titled 'Import Data Supplier' with a 'Choose File' input field and 'Upload' button.

e. Data Barang

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BRG001	Nasi Goreng Spesial	15000	20000	Food & Beverage	Detail Edit Delete
2	BRG002	Mie Ayam	12000	16000	Food & Beverage	Detail Edit Delete
3	BRG003	Es Teh Manis	5000	7000	Food & Beverage	Detail Edit Delete
4	BRG004	Sabun Mandi	8000	12000	Beauty & Health	Detail Edit Delete

f. Penjualan

ID	Kode Penjualan	Pembeli	Tanggal	Aksi
1	PJLAN001	Newt	2024-02-27 08:00:00	Detail Edit Delete
2	PJLAN002	Thomas	2024-02-27 08:30:00	Detail Edit Delete
3	PJLAN003	Minho	2024-02-27 09:00:00	Detail Edit Delete
4	PJLAN004	Chuck	2024-02-27 09:15:00	Detail Edit Delete
5	PJLAN005	Alby	2024-02-27 10:00:00	Detail Edit Delete
6	PJLAN006	Gally	2024-02-27 10:30:00	Detail Edit Delete
7	PJLAN007	Teresa Tie	2024-02-27 11:00:00	Detail Edit Delete
8	PJLAN008	Brenda	2024-02-27 11:30:00	Detail Edit Delete
9	PJLAN009	Aris Jones	2024-02-27 12:00:00	Detail Edit Delete
10	PJLAN010	Harriet	2024-02-27 12:30:00	Detail Edit Delete

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - ⇒ Langkah pertama yang dilakukan adalah membuat file view untuk form import data Excel pada masing-masing menu, misalnya import.blade.php di folder views/barang. Setelah itu, file index pada setiap menu dimodifikasi dengan menambahkan tombol *Import* yang memanggil modal import via AJAX. Kemudian, dua function ditambahkan di controller, yaitu import() untuk menampilkan form, dan import_ajax() untuk memproses file yang diunggah menggunakan PhpSpreadsheet. Terakhir, instal library phpoffice/phpspreadsheet melalui perintah composer require phpoffice/phpspreadsheet, dan mendaftarkan route import dan import_ajax di file web.php untuk mengaktifkan fitur tersebut.
3. Submit kode untuk implementasi praktkikum 1 pada repository github kalian.
 - ⇒ Github



Praktikum 2 – Export Data ke Excel

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>
```

⇒ Source code

```
8     <button onClick="modalAction('{{ url('/barang/import') }}')>Import Barang</button>
9     <a href="{{ url('/barang/export_excel') }}" class="btn btn-primary btn-sm py-1 px-2 mt-1"><i class="fa fa-file-excel"></i> Export Barang</a>
10    <button onClick="modalAction('{{ url('/barang/create_ajax') }}')>Tambah Data (Ajax)</button>
11
```

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
```

⇒ Source code

```
156     Route::post('/import_ajax', [BarangController::class, 'import_ajax']); // ajax import excel
157     Route::get('/export_excel', [BarangController::class, 'export_excel']); // export excel
158
```

3. Selanjutnya kita tambahkan fungsi `export_excel()` pada file `BarangController.php`

⇒ Source code

```
366
369     public function export_excel()
370     {
371
372     }
```

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();
```

⇒ Source code

```
369
370     public function export_excel()
371     {
372         // ambil data barang yang akan di export
373         $barang = BarangModel::select('kategori_id', 'barang_kode', 'barang_nama', 'harga_beli', 'harga_jual')
374             ->orderBy('kategori_id')
375             ->with('kategori')
376             ->get();
377     }
```

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

```

⇒ Source code

```

377     // Load Library excel
378     $spreadsheet = new \PhpOffice\PhpSpreadsheet\Spreadsheet();
379     $sheet = $spreadsheet->getActiveSheet(); // ambil yang active
380
381     $sheet->setCellValue('A1', 'No');
382     $sheet->setCellValue('B1', 'Kode Barang');
383     $sheet->setCellValue('C1', 'Nama Barang');
384     $sheet->setCellValue('D1', 'Harga Beli');
385     $sheet->setCellValue('E1', 'Harga Jual');
386     $sheet->setCellValue('F1', 'Kategori');
387
388     $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     }

```

⇒ Source code

```

390     $no = 1;
391     $baris = 2;
392     foreach ($barang as $key => $value) {
393         $sheet->setCellValue('A' . $baris, $no);
394         $sheet->setCellValue('B' . $baris, $value->barang_kode);
395         $sheet->setCellValue('C' . $baris, $value->barang_nama);
396         $sheet->setCellValue('D' . $baris, $value->harga_beli);
397         $sheet->setCellValue('E' . $baris, $value->harga_jual);
398         $sheet->setCellValue('F' . $baris, $value->kategori->kategori_nama); // ambil nama kategori
399         $baris++;
400         $no++;
401     }

```

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     }
```

⇒ Source code

```
03 |     foreach (range('A', 'F') as $columnID) {  
04 |         $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size kolom  
05 |     }
```

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
```

⇒ Source code

```
403 |     foreach (range('A', 'F') as $columnID) {  
404 |         $sheet->getColumnDimension($columnID)->setAutoSize(true); // set auto size kolom  
405 |     }  
406  
407 |     $sheet->setTitle('Data Barang'); // set title sheet  
408 |     $writer = IOFactory::createWriter($spreadsheet, 'Xlsx');  
409 |     $filename = 'Data Barang ' . date('Y-m-d H:i:s') . '.xlsx';  
410 |     header('Content-Type: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet');  
411 |     header('Content-Disposition: attachment;filename="'. $filename .'"');  
412 |     header('Cache-Control: max-age=0');  
413 |     header('Cache-Control: max-age=1');  
414 |     header('Expires: Mon, 26 Jul 1997 05:00:00 GMT');  
415 |     header('Last-Modified: ' . gmdate('D, d M Y H:i:s') . ' GMT');  
416 |     header('Cache-Control: cache, must-revalidate');  
417 |     header('Pragma: public');  
418  
419 |     $writer->save('php://output');  
420 |     exit;  
421 } // end function export_excel
```

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

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

⇒ Hasil download

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Laptop	7.000.000	10.000.000	Electronics	[Detail] [Edit] [Hapus]
2	BRG002	Smartphone	3.000.000	4.000.000	Electronics	[Detail] [Edit] [Hapus]
3	BRG003	T-Shirt	50.000	100.000	Pakaian	[Detail] [Edit] [Hapus]
4	BRG004	Celana Jeans	150.000	250.000	Pakaian	[Detail] [Edit] [Hapus]
5	BRG005	Pulpen	5.000	7.500	Alat Tulis	[Detail] [Edit] [Hapus]
6	BRG006	Buku Tulis	3.000	5.000	Alat Tulis	[Detail] [Edit] [Hapus]
7	BRG007	Mie Instan	2.500	3.500	Makanan	[Detail] [Edit] [Hapus]
8	BRG008	Air Mineral	1.500	3.000	Minuman	[Detail] [Edit] [Hapus]
9	BRG009	Keyboard	150.000	250.000	Electronics	[Detail] [Edit] [Hapus]
10	BRG010	Mouse	100.000	150.000	Electronics	[Detail] [Edit] [Hapus]

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Laptop	700000	1000000	Electronics
2	BRG002	Smartphone	3000000	4000000	Electronics
3	BRG009	Keyboard	150000	250000	Electronics
4	BRG010	Mouse	100000	150000	Electronics
5	BRG003	T-Shirt	50000	100000	Pakaian
6	BRG004	Celana Jeans	150000	250000	Pakaian
7	BRG005	Pulpen	5000	7500	Alat Tulis
8	BRG006	Buku Tulis	3000	5000	Alat Tulis
9	BRG007	Mie Instan	2500	3500	Makanan
10	BRG008	Air Mineral	1500	3000	Minuman
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					

Tugas 2 - Implementasi File Export Excel:

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

⇒ **Implementasi project POS**

- Level User

ID	Kode Level	Nama Level	Aksi
1	ADM	Administrator	[Detail] [Edit] [Delete]
2	MNG	Manager	[Detail] [Edit] [Delete]
3	STF	Staff/Kasir	[Detail] [Edit] [Delete]
4	CUST	Customer	[Detail] [Edit] [Delete]

b. Data User

Daftar User

Daftar user yang terdaftar dalam sistem

ID	Username	Nama	Level Pengguna	Aksi
1	admin	Administrator	Administrator	[Detail] [Edit] [Delete]
2	manager	Manager	Staff/Kasir	[Detail] [Edit] [Delete]
3	staff	Staff/Kasir	Staff/Kasir	[Detail] [Edit] [Delete]
4	customer	Customer	Customer	[Detail] [Edit] [Delete]
5	cashier	kasir	Manager	[Detail] [Edit] [Delete]

Showing 1 to 5 of 5 entries

localhost/Pemrograman_Web_Lanjut_2025/POS/public/user AdminLTE.io All rights reserved. Version 3.2.0

c. Kategori Barang

Daftar Kategori

Daftar kategori yang terdaftar dalam sistem

ID	Kode Kategori	Nama Kategori	Aksi
1	CTGY01	Food & Beverage	[Detail] [Edit] [Delete]
2	CTGY02	Beauty & Health	[Detail] [Edit] [Delete]
3	CTGY03	Home Care	[Detail] [Edit] [Delete]
4	CTGY04	Baby & Kids	[Detail] [Edit] [Delete]

Showing 1 to 4 of 4 entries

localhost/Pemrograman_Web_Lanjut_2025/POS/public/kategori AdminLTE.io All rights reserved. Version 3.2.0

d. Data Supplier

Daftar Supplier

Daftar supplier yang terdaftar dalam sistem

ID	Kode Supplier	Nama Supplier	Alamat Supplier	Aksi
1	SUP001	Supplier A	Jl. Merdeka No. 1, Jakarta	[Detail] [Edit] [Delete]
2	SUP002	Supplier b	Jl. Pahlawan No. 5, Surabaya	[Detail] [Edit] [Delete]
3	SUP003	Supplier C	Jl. Raya No. 10, Bandung	[Detail] [Edit] [Delete]
4	SUP004	Supplier D	Jl. Sejahtera No. 8, Yogyakarta	[Detail] [Edit] [Delete]
5	SUP005	Supplier E	Jl. Sentosa No. 3, Bali	[Detail] [Edit] [Delete]

Showing 1 to 5 of 5 entries

localhost/Pemrograman_Web_Lanjut_2025/POS/public/supplier AdminLTE.io All rights reserved. Version 3.2.0

e. Data Barang

ID	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori Barang	Aksi
1	BRG001	Nasi Goreng Spesial	15000	20000	Food & Beverage	[Detail] [Edit] [Delete]
2	BRG002	Mie Ayam	12000	16000	Food & Beverage	[Detail] [Edit] [Delete]
3	BRG003	Es Teh Manis	5000	7000	Food & Beverage	[Detail] [Edit] [Delete]
4	BRG004	Sabun Mandi	8000	12000	Beauty & Health	[Detail] [Edit] [Delete]
5	BRG005	Shampoo	15000	20000	Beauty & Health	[Detail] [Edit] [Delete]
6	BRG006	Pasta Gigi	10000	15000	Beauty & Health	[Detail] [Edit] [Delete]
7	BRG007	Detergen	20000	25000	Home Care	[Detail] [Edit] [Delete]
8	BRG008	Pembersih Lantai	25000	30000	Home Care	[Detail] [Edit] [Delete]
9	BRG009	Pewangi Ruangan	12000	15000	Home Care	[Detail] [Edit] [Delete]
10	BRG010	Popok Bayi	30000	40000	Baby & Kids	[Detail] [Edit] [Delete]

f. Stok Barang

ID	Jumlah	Tanggal	Barang ID	Supplier ID	Aksi
1	100	2024-02-27 08:00:00	1	1	[Detail] [Edit] [Delete]
2	120	2024-02-27 08:00:00	2	2	[Detail] [Edit] [Delete]
3	10	2024-02-27 08:00:00	3	3	[Detail] [Edit] [Delete]
4	30	2024-02-27 08:00:00	4	4	[Detail] [Edit] [Delete]
5	50	2024-02-27 08:00:00	5	5	[Detail] [Edit] [Delete]
6	75	2024-02-27 08:00:00	6	1	[Detail] [Edit] [Delete]
7	200	2024-02-27 08:00:00	7	2	[Detail] [Edit] [Delete]
8	300	2024-02-27 08:00:00	8	3	[Detail] [Edit] [Delete]
9	45	2024-02-27 08:00:00	9	4	[Detail] [Edit] [Delete]
10	60	2024-02-27 08:00:00	10	5	[Detail] [Edit] [Delete]

g. Penjualan

ID	Kode Penjualan	Pembeli	Tanggal	Aksi
1	PJAN001	Newt	2024-02-27 08:00:00	[Detail] [Edit] [Delete]
2	PJAN002	Thomas	2024-02-27 08:30:00	[Detail] [Edit] [Delete]
3	PJAN003	Minho	2024-02-27 09:00:00	[Detail] [Edit] [Delete]
4	PJAN004	Chuck	2024-02-27 09:15:00	[Detail] [Edit] [Delete]
5	PJAN005	Alby	2024-02-27 10:00:00	[Detail] [Edit] [Delete]
6	PJAN006	Gally	2024-02-27 10:30:00	[Detail] [Edit] [Delete]
7	PJAN007	Teresa Tie	2024-02-27 11:00:00	[Detail] [Edit] [Delete]
8	PJAN008	Brenda	2024-02-27 11:30:00	[Detail] [Edit] [Delete]
9	PJAN009	Aris Jones	2024-02-27 12:00:00	[Detail] [Edit] [Delete]
10	PJAN010	Harriet	2024-02-27 12:30:00	[Detail] [Edit] [Delete]

- Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
 - Langkah pertama dimulai dengan mengedit file index pada setiap menu untuk menambahkan tombol *Export* sebagai fitur tambahan. Setelah itu, route baru ditambahkan di web.php untuk menangani proses export ke Excel. Kemudian, dibuat function export() di masing-masing controller yang bertugas mengambil data dari database dan mengubahnya menjadi file Excel menggunakan PhpSpreadsheet. Saat tombol *Export* ditekan, data langsung diunduh dalam bentuk file Excel oleh pengguna.
- Submit kode untuk implementasi praktkikum 2 pada repository github kalian.

⇒ Github

Minggu 8 (PWL_POS)	add exorrt function	now
POS	add exorrt function	now

Praktikum 3 - Implementasi Export PDF di Laravel dengan dompdf

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

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

⇒ Source code

```
PS D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (PWL_POS)> composer require barryvdh/laravel-dompdf
./composer.json has been updated
Running composer update barryvdh/laravel-dompdf
Loading composer repositories with package information
Updating dependencies
Lock file operations: 6 installs, 0 updates, 0 removals
- Locking barryvdh/laravel-dompdf (v3.1.1)
- Locking dompdf/dompdf (v3.1.0)
- Locking dompdf/php-font-lib (1.0.1)
- Locking dompdf/php-svg-lib (1.0.0)
- Locking masterminds/html5 (2.9.0)
- Locking sabberworm/php-css-parser (v8.8.0)
Writing lock file
```

- 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>
```

⇒ Source code

```
<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
```

⇒ Source code

```
156     Route::post('/import_ajax',[BarangController::class,'import_ajax']); // ajax import excel
157     Route::get('/export_excel',[BarangController::class,'export_excel']); // export excel
158     Route::get('/export_pdf',[BarangController::class,'export_pdf']); // export pdf
159
```

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');
}
```

⇒ Source code

```
424     public function export_pdf()
425     {
426         $barang = BarangModel::select('kategori_id','barang_kode','barang_nama','harga_beli','harga_jual')
427             ->orderBy('kategori_id')
428             ->orderBy('barang_kode')
429             ->with('kategori')
430             ->get();

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

436         return $pdf->stream('Data Barang '.date('Y-m-d H:i:s').'.pdf');
437     }
438 }
```

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.

⇒ Source code

```
resources > views > barang > 📄 export_pdf.blade.php > ⚙️ html
1  <!DOCTYPE html>
2  <html lang="id">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <style>
7          body {
8              font-family: "Times New Roman", Times, serif;
9              margin: 6px 20px 5px 20px;
10             line-height: 15px;
11         }
12         table {
13             width: 100%;
14             border-collapse: collapse;
15         }
16         td, th {
17             padding: 4px 3px;
18         }
19         th {
20             text-align: left;
21         }
22         .d-block {
23             display: block;
24         }
25         img.image {
26             width: auto;
27             height: 80px;
28             max-width: 150px;
29             max-height: 150px;
30         }
31         .text-right {
32             text-align: right;
33         }
34         .text-center {
35             text-align: center;
36         }
37         .p-1 {
38             padding: 5px 1px;
39         }
40         .font-10 {
41             font-size: 10pt;
42         }
43         .font-11 {
44             font-size: 11pt;
45         }
46         .font-12 {
47             font-size: 12pt;
48         }
49         .font-13 {
50             font-size: 13pt;
51         }
```

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

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

```

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

⇒ Halaman Browser

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori	Aksi
1	BRG001	Laptop	7.000.000	10.000.000	Electronics	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
2	BRG002	Smartphone	3.000.000	4.000.000	Electronics	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
3	BRG003	T-Shirt	50.000	100.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
4	BRG004	Celana Jeans	150.000	250.000	Pakaian	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
5	BRG005	Pulpen	5.000	7.500	Alat Tulis	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
6	BRG006	Buku Tulis	3.000	5.000	Alat Tulis	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
7	BRG007	Mie Instan	2.500	3.500	Makanan	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
8	BRG008	Air Mineral	1.500	3.000	Minuman	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
9	BRG009	Keyboard	150.000	250.000	Electronics	<button>Detail</button> <button>Edit</button> <button>Hapus</button>
10	BRG010	Mouse	100.000	150.000	Electronics	<button>Detail</button> <button>Edit</button> <button>Hapus</button>

Showing 1 to 10 of 10 entries

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

LAPORAN DATA BARANG

No	Kode Barang	Nama Barang	Harga Beli	Harga Jual	Kategori
1	BRG001	Laptop	7.000.000	10.000.000	Electronics
2	BRG002	Smartphone	3.000.000	4.000.000	Electronics
3	BRG009	Keyboard	150.000	250.000	Electronics
4	BRG010	Mouse	100.000	150.000	Electronics
5	BRG003	T-Shirt	50.000	100.000	Pakaian
6	BRG004	Celana Jeans	150.000	250.000	Pakaian
7	BRG005	Pulpen	5.000	7.500	Alat Tulis
8	BRG006	Buku Tulis	3.000	5.000	Alat Tulis
9	BRG007	Mie Instan	2.500	3.500	Makanan
10	BRG008	Air Mineral	1.500	3.000	Minuman

Tugas 3 - Implementasi Export PDF pada Laravel :

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

⇒ **Implementasi project POS**

- Level User**

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
Logo Polinema
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 LEVEL USER

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

b. Data User

Daftar user yang terdaftar dalam sistem

Import Export pdf Tambah +

localhost/Pemrograman_Web_Lanjut_2025/POS/public/user/export.pdf

1 / 1 - 100% + ⌂ ⌂

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
Logo Polinema 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 USER

No	Nama	Username	Level Pengguna
1	Administrator	admin	Administrator
2	Manager	manager	Staff/Kasir
3	Staff/Kasir	staff	Staff/Kasir
4	Customer	customer	Customer
5	kasir	cashier	Manager

c. Kategori Barang

Daftar kategori yang terdaftar dalam sistem

Import Export pdf Tambah +

localhost/Pemrograman_Web_Lanjut_2025/POS/public/kategori/export.pdf

1 / 1 - 100% + ⌂ ⌂ Reset

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

LAPORAN DATA KATEGORI BARANG

No	Kode Kategori	Nama Kategori
1	CTGY01	Food & Beverage
2	CTGY02	Beauty & Health
3	CTGY03	Home Care
4	CTGY04	Baby & Kids

d. Data Supplier

Daftar supplier yang terdaftar dalam sistem

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
Logo Polinema
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 SUPPLIER

No	Kode Supplier	Nama Supplier	Alamat
1	SUP001	Supplier A	Jl. Merdeka No. 1, Jakarta
2	SUP002	Supplier b	Jl. Pahlawan No. 5, Surabaya
3	SUP003	Supplier C	Jl. Raya No. 10, Bandung
4	SUP004	Supplier D	Jl. Sejahtera No. 8, Yogyakarta
5	SUP005	Supplier E	Jl. Sentosa No. 3, Bali

e. Data Barang

Daftar Barang

KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
Logo Polinema
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	Nasi Goreng Spesial	15.000	20.000	Food & Beverage
2	BRG002	Mie Ayam	12.000	16.000	Food & Beverage
3	BRG003	Es Teh Manis	5.000	7.000	Food & Beverage
4	BRG004	Sabun Mandi	8.000	12.000	Beauty & Health
5	BRG005	Shampoo	15.000	20.000	Beauty & Health
6	BRG006	Pasta Gigi	10.000	15.000	Beauty & Health
7	BRG007	Detergen	20.000	25.000	Home Care
8	BRG008	Pembersih Lantai	25.000	30.000	Home Care
9	BRG009	Pewangi Ruangan	12.000	15.000	Home Care
10	BRG010	Popok Bayi	30.000	40.000	Baby & Kids
11	BRG011	Susu Formula	50.000	60.000	Baby & Kids
12	BRG012	Mainan Edukatif	40.000	50.000	Baby & Kids

f. Stok Barang

The screenshot shows a PDF document titled "LAPORAN DATA STOK BARANG" generated by the application. The document header includes the logo of Polinema and its address: Jl. Soekarno-Hatta No. 9 Malang 65141, Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420, Laman: www.polinema.ac.id. The main content is a table listing various items with their quantities and suppliers.

No	Jumlah	Tanggal	Barang	Supplier
1	100	2024-02-27 08:00:00	Nasi Goreng Spesial	Supplier A
2	120	2024-02-27 08:00:00	Mie Ayam	Supplier b
3	10	2024-02-27 08:00:00	Es Teh Manis	Supplier C
4	30	2024-02-27 08:00:00	Sabun Mandi	Supplier D
5	50	2024-02-27 08:00:00	Shampoo	Supplier E
6	75	2024-02-27 08:00:00	Pasta Gigi	Supplier A
7	200	2024-02-27 08:00:00	Detergen	Supplier b
8	300	2024-02-27 08:00:00	Pembersih Lantai	Supplier C
9	45	2024-02-27 08:00:00	Pewangi Ruangan	Supplier D
10	60	2024-02-27 08:00:00	Popok Bayi	Supplier E

g. Penjualan

The screenshot shows a PDF document titled "LAPORAN DATA PENJUALAN" generated by the application. The document header includes the logo of Polinema and its address: Jl. Soekarno-Hatta No. 9 Malang 65141, Telepon (0341) 404424 Pes. 101-105, 0341-404420, Fax. (0341) 404420, Laman: www.polinema.ac.id. The main content is a table listing sales transactions with their details.

No	Kode Penjualan	Pembeli	Tanggal
1	PJLAN001	Newt	2024-02-27 08:00:00
2	PJLAN002	Thomas	2024-02-27 08:30:00
3	PJLAN003	Minho	2024-02-27 09:00:00
4	PJLAN004	Chuck	2024-02-27 09:15:00
5	PJLAN005	Alby	2024-02-27 10:00:00
6	PJLAN006	Gally	2024-02-27 10:30:00
7	PJLAN007	Teresa Tie	2024-02-27 11:00:00
8	PJLAN008	Brenda	2024-02-27 11:30:00
9	PJLAN009	Aris Jones	2024-02-27 12:00:00
10	PJLAN010	Harriet	2024-02-27 12:30:00
11	PJLAN011	Jorge	2024-02-27 13:00:00
12	PJLAN012	Vince	2024-02-27 13:30:00

2. Amati dan jelaskan tiap tahapan yang kalian kerjakan, dan jabarkan dalam laporan
- ⇒ Langkah pertama adalah menambahkan tombol Export Pdf di halaman index.blade.php, biasanya diletakkan di bagian atas tabel data agar mudah diakses oleh pengguna. Setelah itu, kita mendefinisikan route baru pada web.php yang mengarah ke function export_pdf() di controller terkait. Kemudian kita membuat function export_pdf() di controller untuk mengambil data dari database dan me-render-nya ke file PDF menggunakan library seperti dompdf. Terakhir, kita membuat file view khusus untuk PDF, misalnya pdf.blade.php, dengan desain sederhana dan styling CSS manual agar hasil PDF terlihat rapi dan sesuai kebutuhan.

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

⇒ Github

	add extract pdf function	now
	add extract pdf function	now

-harusnya commitnya add export pdf function :\|

Tugas 4 - Implementasi Upload File Gambar :

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

⇒ Source code

```

138     <div class="dropdown-menu dropdown-menu-right">
139         <a class="dropdown-item" href="{{ url('/user/profile') }}>
140             <i class="fas fa-user-circle"></i> Profile <!-- Ikon profil -->
141         </a>
142         <a class="dropdown-item" href="#" onclick="event.preventDefault(); document.getElementById('logout-form').submit();">
143             <i class="fas fa-sign-out-alt"></i> Logout <!-- Ikon logout -->
144         </a>
145         <form id="logout-form" action="{{ url('logout') }}" method="POST" style="display: none;">
146             @csrf
147         </form>
148     </div>

```

```

resources > views > user > profile.blade.php > ...
1  @extends('layouts.template')
2
3  @section('content')
4  @if (session('success'))
5      <div class="alert alert-success alert-dismissible fade show" role="alert">
6          {{ session('success') }}
7          <button type="button" class="close" data-dismiss="alert" aria-label="Close">
8              <span aria-hidden="true">&times;</span>
9          </button>
10     </div>
11     @endif
12
13     @if (session('error'))
14         <div class="alert alert-danger alert-dismissible fade show" role="alert">
15             {{ session('error') }}
16             <button type="button" class="close" data-dismiss="alert" aria-label="Close">
17                 <span aria-hidden="true">&times;</span>
18             </button>
19         </div>
20     @endif
21
22     <div class="card shadow">
23         <div class="card-header bg-primary text-white">
24             <h3 class="card-title">Informasi Profil</h3>
25         </div>
26         <div class="card-body">
27             <div class="row">
28                 <div class="col-md-4 text-center">
29                     <div class="mb-4">
30                         
32                     </div>
33                 </div>

```

```
1 class UserModel extends Authenticatable
2 {
3     use HasFactory;
4
5     protected $table = 'm_user';
6     protected $primaryKey = 'user_id';
7     protected $fillable = ['username', 'password', 'nama', 'level_id', 'created_at', 'updated_at', 'user_profile_picture'];
8     protected $hidden = ['password']; // jangan di tampilkan saat select
9     protected $casts = ['password' => 'hashed']; // casting password agar otomatis di hash
10
11 database > migrations > 2025_04_12_170026_update_m_user_table.php > ...
12
13     <?php
14
15     use Illuminate\Database\Migrations\Migration;
16     use Illuminate\Database\Schema\Blueprint;
17     use Illuminate\Support\Facades\Schema;
18
19     return new class extends Migration
20     {
21         /**
22          * Run the migrations.
23          */
24         public function up(): void
25         {
26             Schema::table('m_user', function (Blueprint $table) {
27                 $table->string('user_profile_picture')->nullable()->after('password');
28             });
29         }
30
31         /**
32          * Reverse the migrations.
33          */
34         public function down(): void
35         {
36             Schema::table('m_user', function (Blueprint $table) {
37                 $table->dropColumn('user_profile_picture');
38             });
39         }
40     };
41 }
```

```
app > Http > Controllers > UserController.php > ...
12     class UserController extends Controller
13
14     public function profile_page()
15     {
16         $user = auth()->user();
17
18         $breadcrumb = (object) [
19             'title' => 'User Profile',
20             'list' => ['Home', 'Profile']
21         ];
22
23         $page = (object) [
24             'title' => 'User Profile'
25         ];
26
27         $activeMenu = 'profile';
28
29         return view('user.profile', ['user' => $user, 'breadcrumb' => $breadcrumb, 'page' => $page, 'activeMenu' => $activeMenu]);
30     }
31
32     public function update_picture(Request $request)
33     {
34         // Validasi file
35         $request->validate([
36             'user_profile_picture' => 'required|image|mimes:jpeg,png,jpg|max:2048',
37         ]);
38
39         try {
40             $user = auth()->user();
41
42             if (!$user) {
43                 return redirect('/login')->with('error', 'Silahkan login terlebih dahulu');
44             }
45
46             $userId = $user->user_id;
47
48             $userModel = UserModel::find($userId);
49
50             if (!$userModel) {
51                 return redirect('/login')->with('error', 'User tidak ditemukan');
52             }
53
54             // Menghapus foto jika sudah ada
55             if ($userModel->user_profile_picture && file_exists(storage_path('app/public/' . $userModel->user_profile_picture))) {
56                 Storage::disk('public')->delete($userModel->user_profile_picture);
57             }
58
59             $fileName = 'profile_' . $userId . '_' . time() . '.' . $request->user_profile_picture->extension();
60             $path = $request->user_profile_picture->storeAs('profiles', $fileName, 'public');
61
62             UserModel::where('user_id', $userId)->update([
63                 'user_profile_picture' => $path
64             ]);
65         }
66     }
67 }
```

```

49 |     Route::get('/user/profile', [UserController::class, 'profile_page'])->name('user.profile');
50 |     Route::post('/user/update_picture', [UserController::class, 'update_picture'])->name('user.update_picture');
51 |

```

PS D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (PWL_POS)> php artisan storage:link
INFO The [D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (PWL_POS)\public\storage] link has been connected to [D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (PWL_POS)\public\storage].

PS D:\laragon\www\Pemrograman_Web_Lanjut_2025\Minggu 8 (PWL_POS)> php artisan migrate
INFO Running migrations.
2025_04_12_170026_update_m_user_table 193ms DONE

⇒ Halaman Browser

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

⇒ Pertama, ditambahkan menu profile di header navbar yang mengarah ke halaman profil user. Kemudian, dibuat migration untuk menambahkan kolom user_profile pada tabel users. Setelah itu, model User diperbarui dengan menambahkan field tersebut ke dalam \$fillable. Selanjutnya, dibuat view profile.blade.php untuk menampilkan dan mengedit data user. Lalu, ditambahkan fungsi profile() dan updateProfile() di controller untuk menangani tampilan dan update data. Route baru ditambahkan untuk akses dan penyimpanan data profil. Terakhir, dijalankan perintah php artisan storage:link untuk membuat folder penyimpanan foto profil.

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

⇒ Github

Minggu 8 (PWL_POS)	add profile & upload image file function	now
POS	add profile & upload image file function	now