

LAPORAN PRAKTIKUM PEMROGRAMAN WEB LANJUT

Nama : Ahmad Farieq

NIM : 231240001397

Mata kuliah : Pemrograman Web Lanjut

Dosen Pengampu : Harminto Mulyo, S.Kom. M.Kom.

1. Link Github

[famusproject/hypekas](https://github.com/famusproject/hypekas)

2. Dokumentasi

a) Struktur Folder Proyek (Laravel)

- **app/**
Berisi kode utama aplikasi (controller, model, middleware, dll).
- **Http/Controllers/**: Tempat controller (logika request/response, misal: SaleController.php, AuthController.php).
- **Models/**: Tempat model (representasi tabel database, misal: Product.php, Sale.php).
- **resources/views/**
Berisi file tampilan (Blade template) untuk frontend, misal:
 - **umkm/sales/**: Tampilan fitur penjualan.
 - **umkm/products/**: Tampilan produk.
 - **umkm/layouts/**: Layout utama.
- **routes/web.php**
Tempat mendefinisikan URL/route aplikasi dan menghubungkannya ke controller.
- **database/migrations/**
Berisi file migrasi untuk membuat/ubah struktur tabel database.
- **public/**
Berisi file yang bisa diakses publik (index.php, asset, dll).
- **config/**
Berisi file konfigurasi aplikasi.

b) Alur Login hingga ke Dashboard

1. **User mengakses halaman login (/login).**
2. **User mengisi username/email & password** lalu submit.
3. **AuthController** memproses login:
 - Mengecek kredensial ke database.
 - Jika benar, user diarahkan ke dashboard (/umkm/dashboard).
 - Jika salah, muncul pesan error.

4. Setelah login berhasil, user bisa mengakses seluruh fitur yang membutuhkan autentikasi (penjualan, produk, laporan, dll).

c) Alur CRUD (Create, Read, Update, Delete)

a. Menambah Data (Create)

- User klik tombol **Tambah** (misal: "Tambah Penjualan").
- Muncul form input data.
- User mengisi form dan submit.
- Data dikirim ke controller (misal: SaleController@store), divalidasi, lalu disimpan ke database.
- User diarahkan kembali ke daftar data dengan pesan sukses.

b. Melihat Data (Read)

- User melihat daftar data di halaman utama (misal: daftar penjualan).
- Data diambil dari database dan ditampilkan dalam tabel.

c. Mengedit Data (Update)

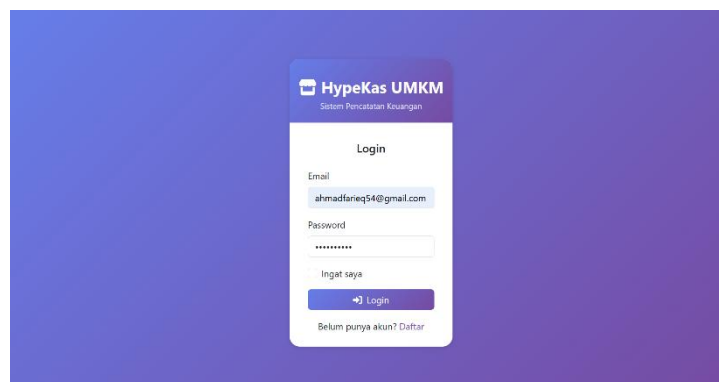
- User klik tombol **Edit** pada data yang ingin diubah.
- Muncul form edit dengan data lama terisi.
- User mengubah data lalu submit.
- Data dikirim ke controller (misal: SaleController@update), divalidasi, lalu diupdate di database.
- User diarahkan kembali ke daftar data dengan pesan sukses.

d. Menghapus Data (Delete)

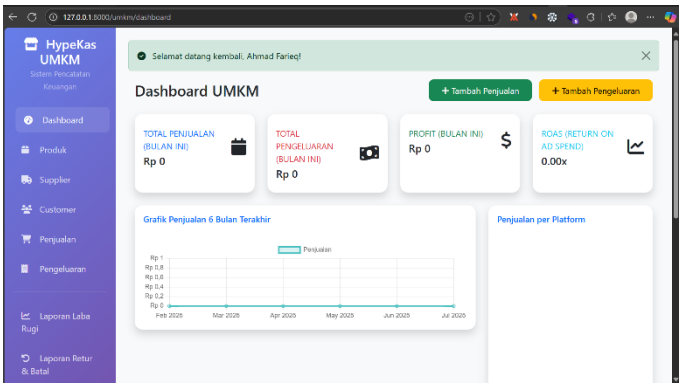
- User klik tombol **Hapus** pada data yang ingin dihapus.
- Muncul konfirmasi.
- Jika setuju, data dihapus oleh controller (misal: SaleController@destroy) dari database.
- User diarahkan kembali ke daftar data dengan pesan sukses.

d) Lampiran

a. Tampilan Login



b. Tampilan Dashboard



c. Tampilan Tabel

Nama Produk	SKU	Supplier	Harga Beli	Harga Jual	Stok	Status	Aksi
celana diamond knit oke	AD1	Agus	Rp 83.000	Rp 150.000	86	Aktif	Edit Hapus Detail
celana panjang ok	AD3	Agus	Rp 400.000	Rp 210.000	12345	Aktif	Edit Hapus Detail

d. Tampilan create

Tambah Produk Baru

Form fields include:

- Nama Produk *
- SKU *
- Harga Beli * (Rp)
- Harga Jual * (Rp)
- Supplier (Dropdown: Pilih Supplier)
- Stok *
- Deskripsi
- ☒ Aktifkan Produk

Buttons: [Batal](#), [Simpan Produk](#), [Kembali](#)

e. Tampilan Edit

Edit Produk

Form fields include:

- Nama Produk *
- SKU *
- Supplier (Dropdown: Agus)
- Stok *
- Harga Beli * (Rp 83000.00)
- Harga Jual * (Rp 150000.00)
- Deskripsi
- ☒ Produk Aktif

Buttons: [Batal](#), [Update Produk](#), [Kembali](#)

f. Tampilan delete

