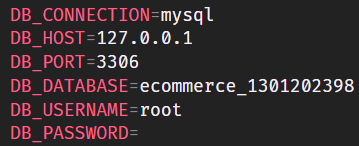
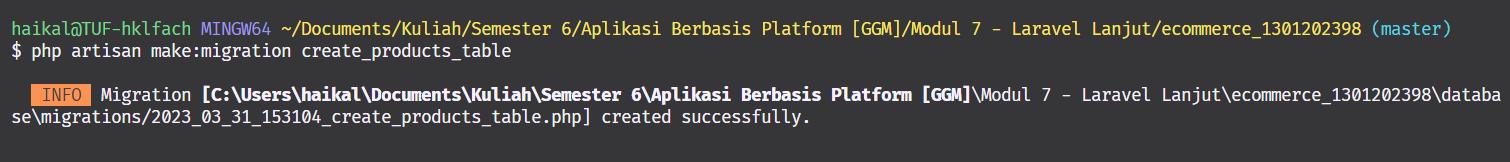
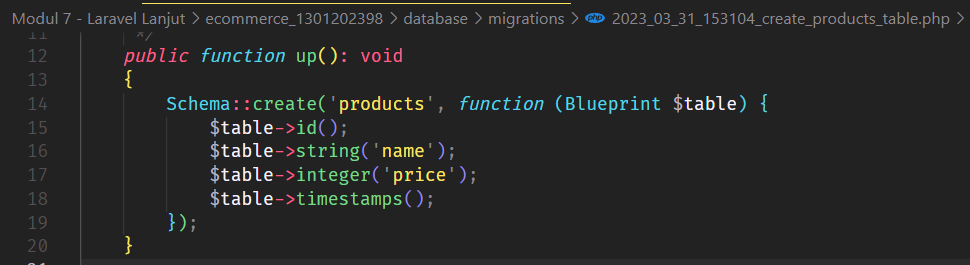
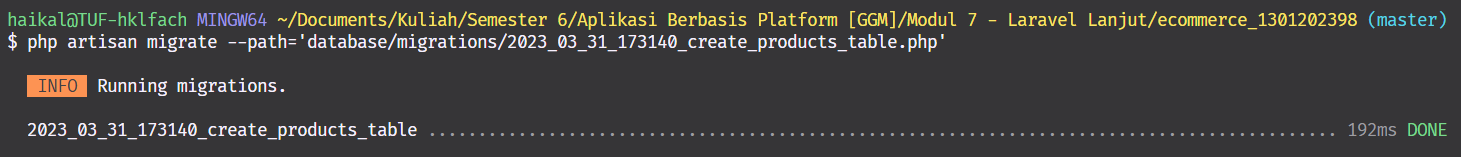
* 1. **CRUD**

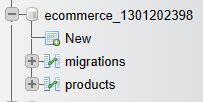
Akan dibuat aplikasi ecommerce yang dapat melakukan implementasi Create, Read, Update, dan Delete (CRUD) pada tabel produknya.

* + 1. **Konfigurasi dan Skema**

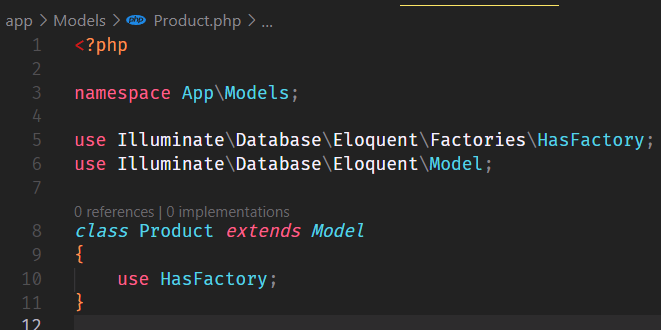
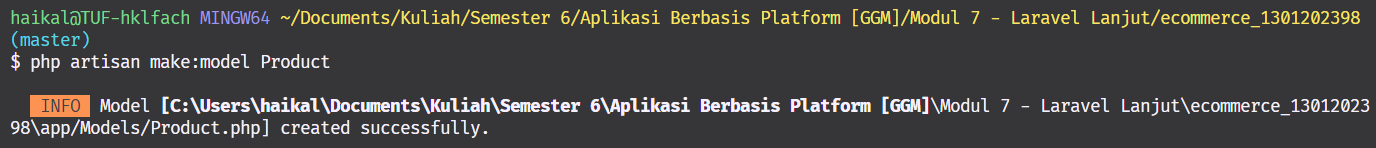
Konfigurasi file .env menyesuaikan dengan jenis database dan nama database yang digunakan.

Membuat table product menggunakan command “php artisan make:migration <nama\_table>”.

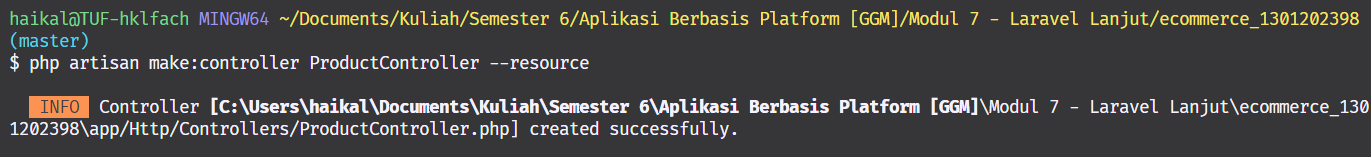
File <date\_created>\_create\_products\_table.php akan terbuat pada folder database/migrations. Kemudian ditambahkan kolom name dan price.

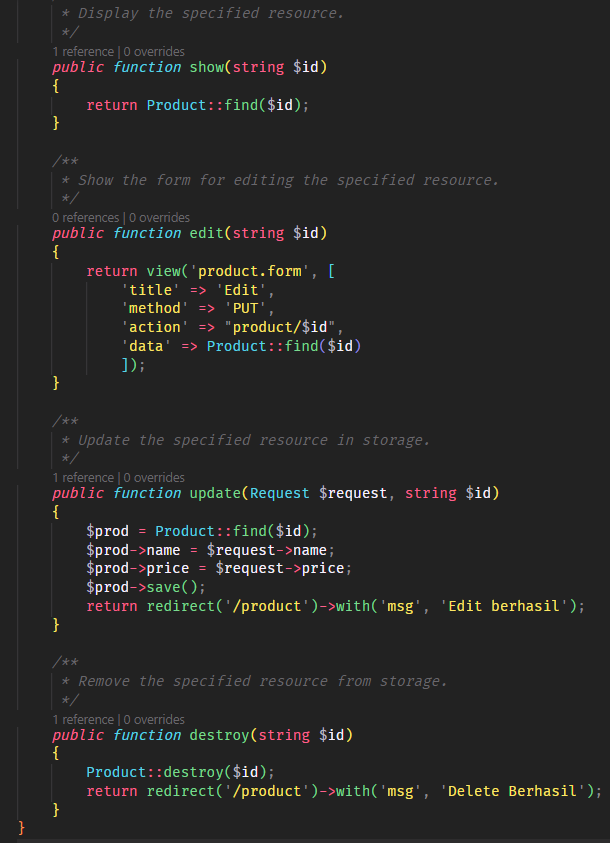
Membuat table ke database dengan command “php artisan migrate”.

Hasil dari tabel yang telah dibuat pada database.

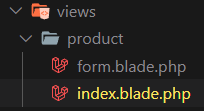
* + 1. **Model**

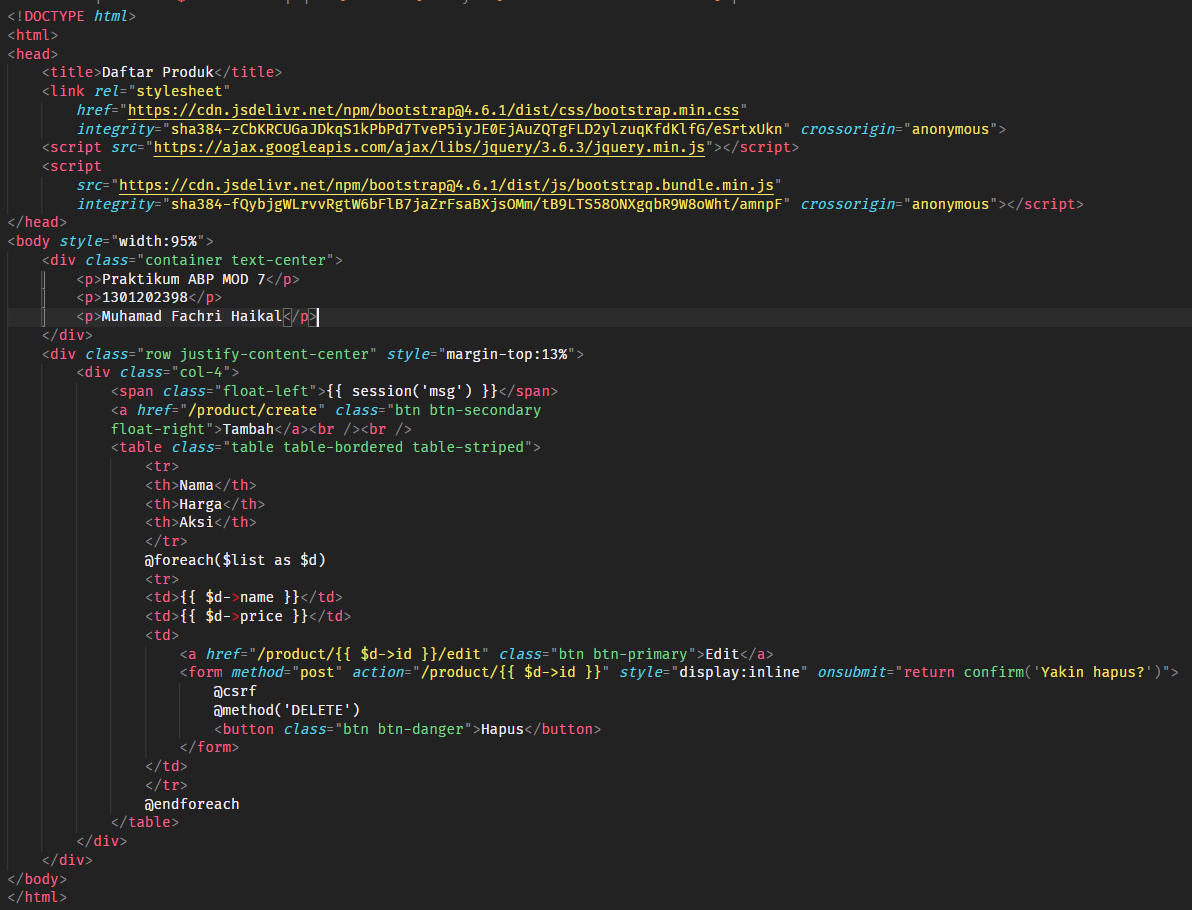
Membuat model product dengan command “php artisan make:model <nama\_model>”.

* + 1. **Controller**

****Membuat controller product dengan command “php artisan make:controller <nama\_controller> --resource”, opsi resource digunakan untuk membuat method-method yang berguna agar memudahkan nantinya.

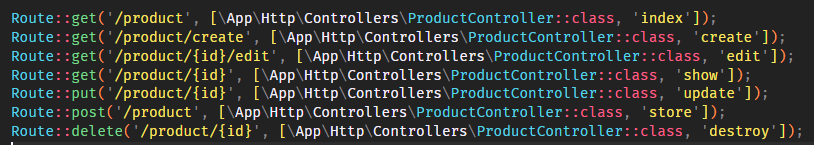
Isi dari method-method yang ada dari controller product adalah implementasi dari CRUD.

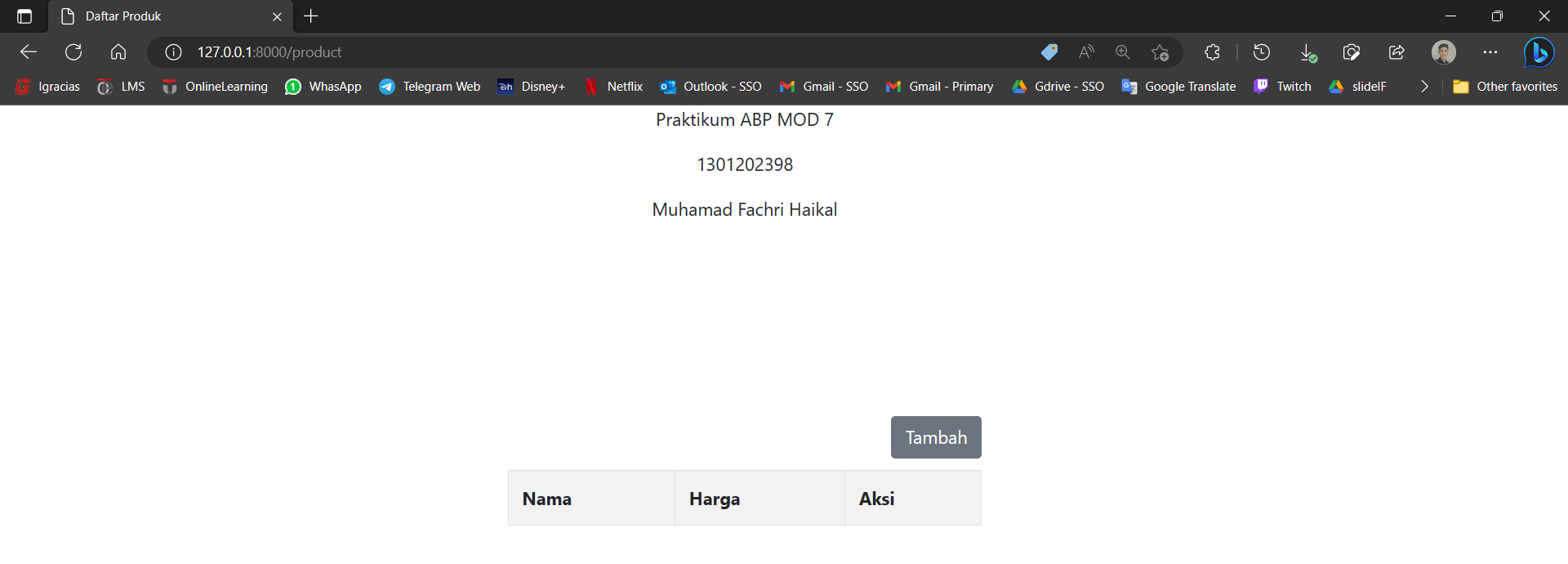
* + 1. **View**

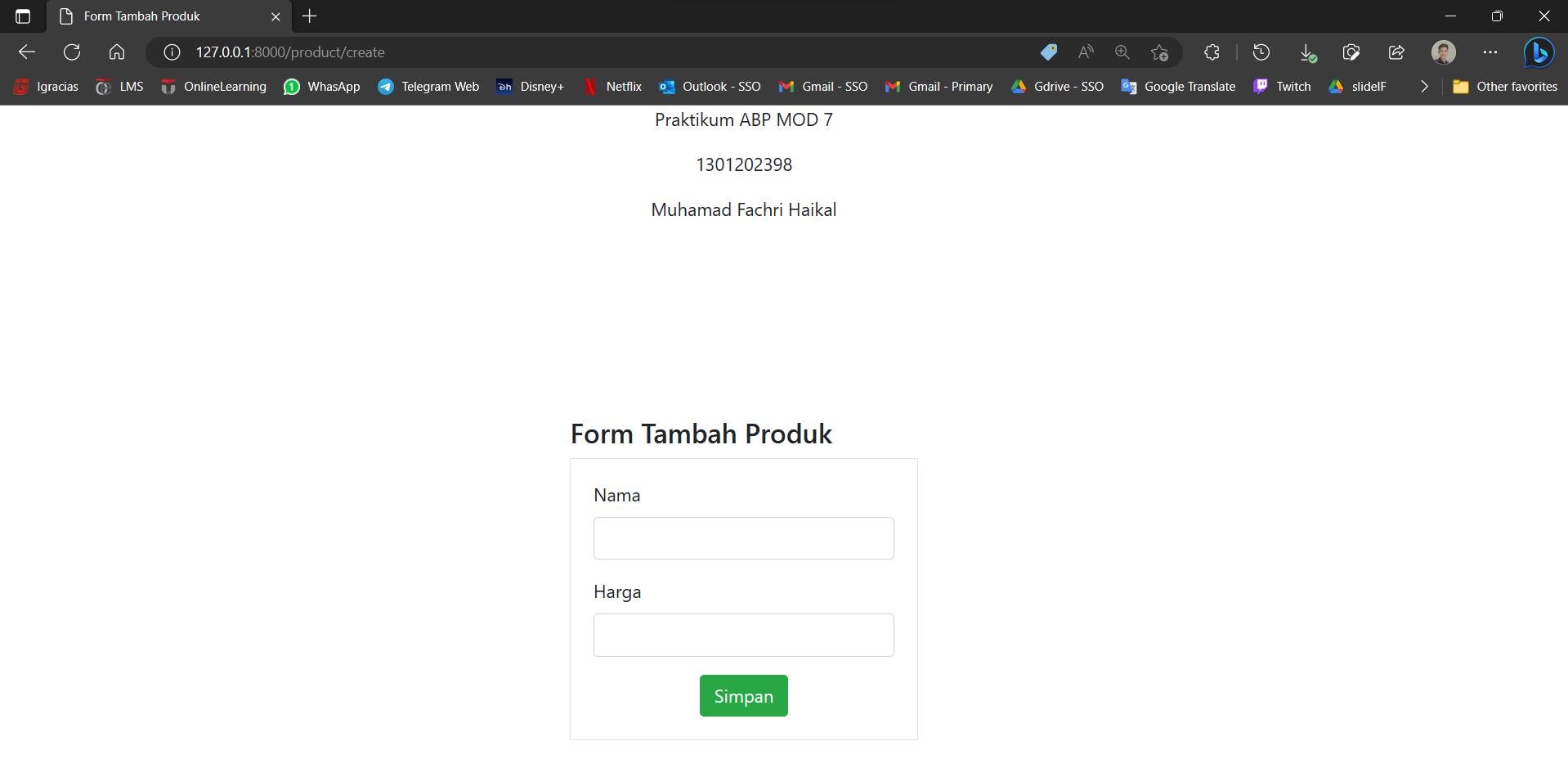
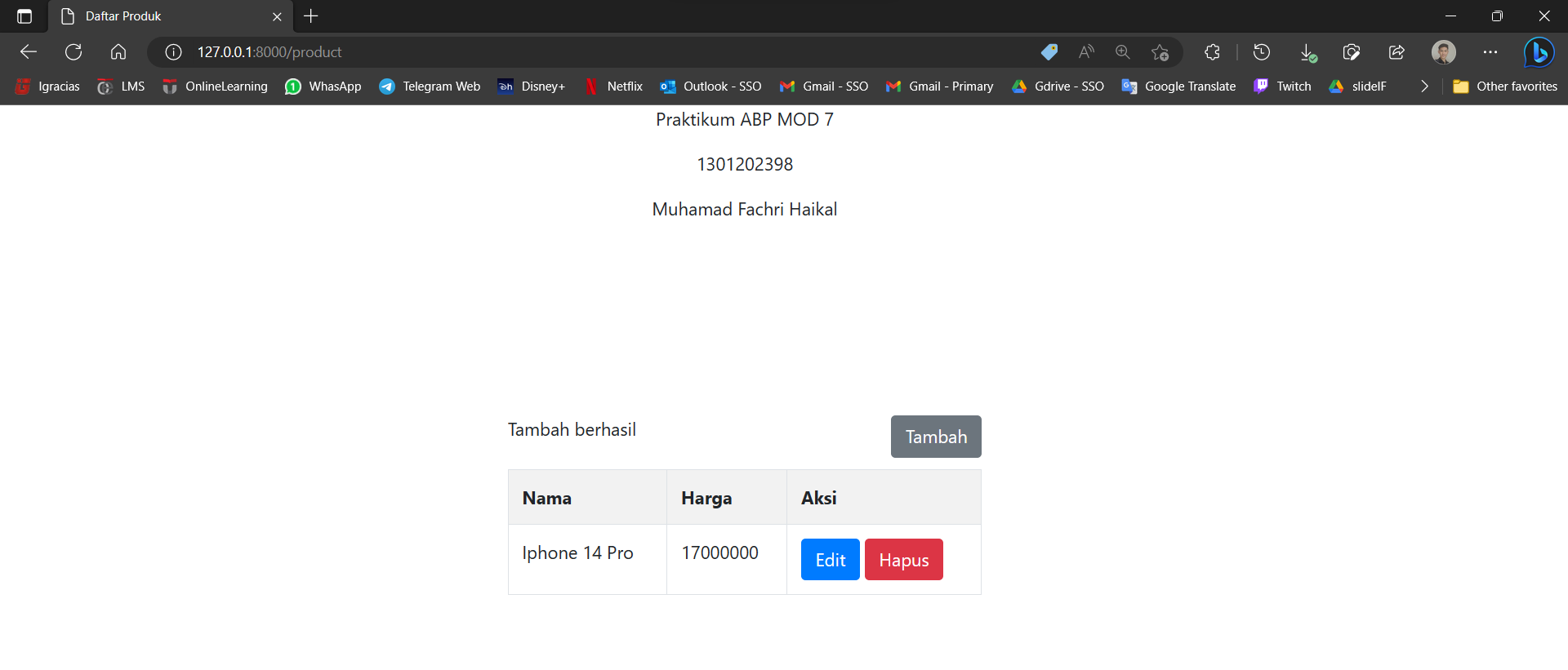
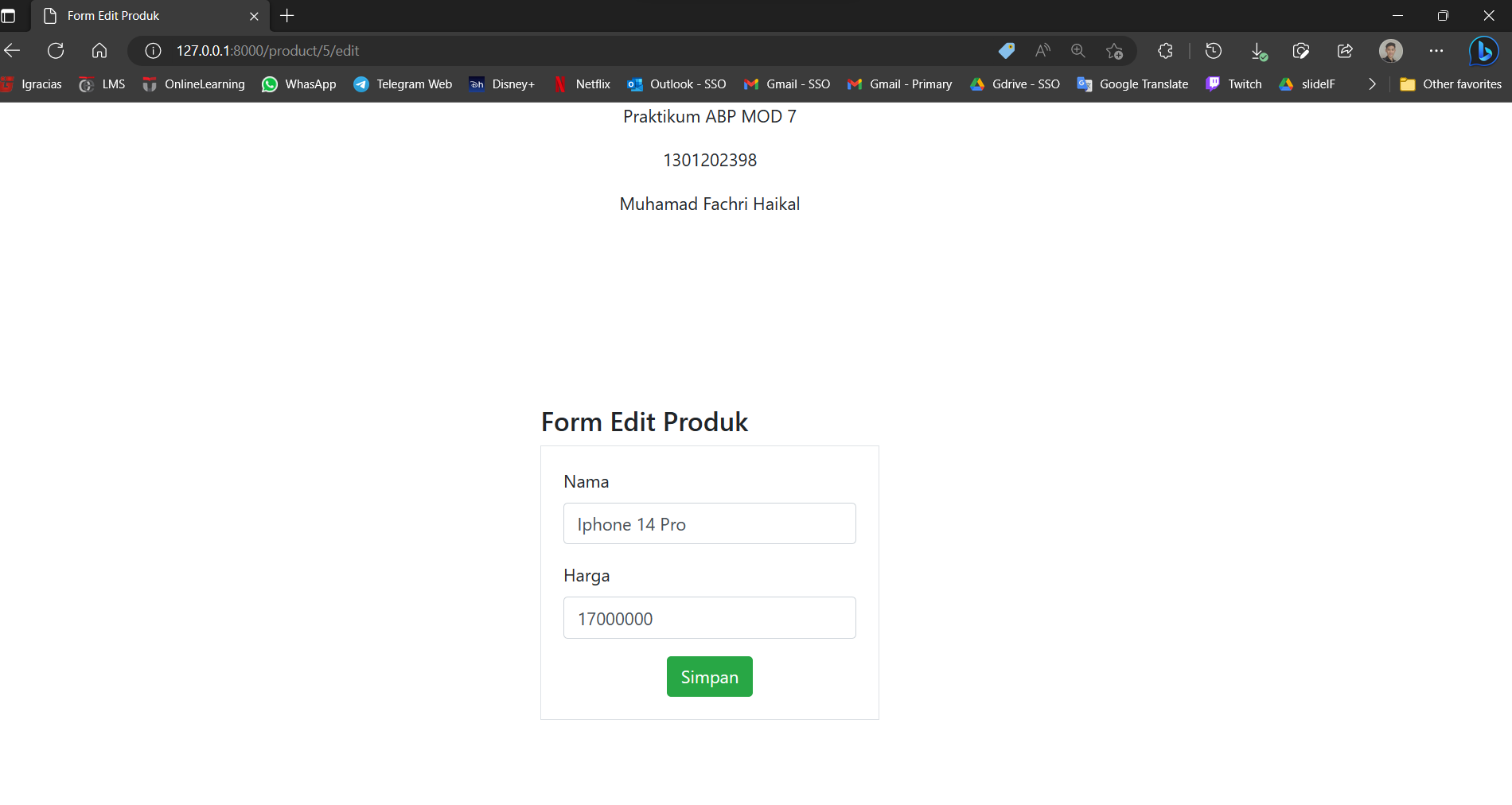
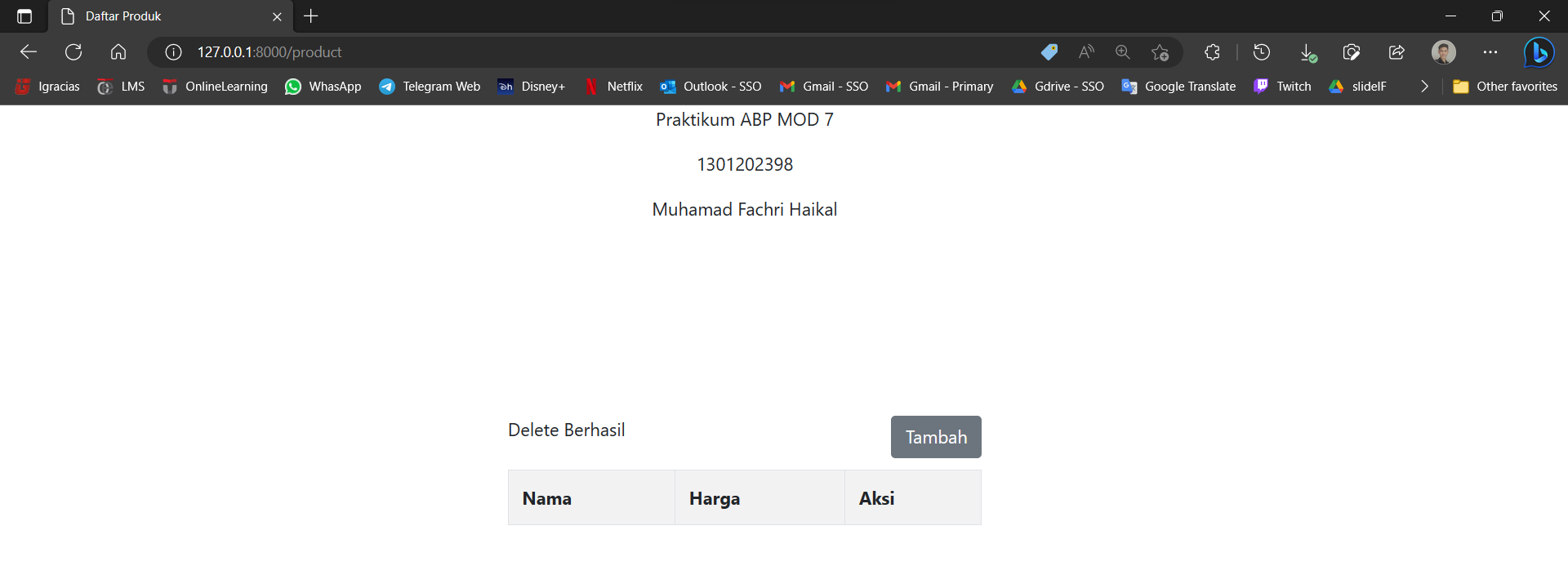
****Struktur direktori untuk interface product.

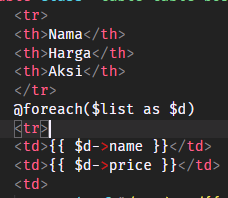
Isi file index.blade.php untuk product.

Isi file form.blade.php untuk form create dan update product.

* + 1. **Tampilan Halaman**

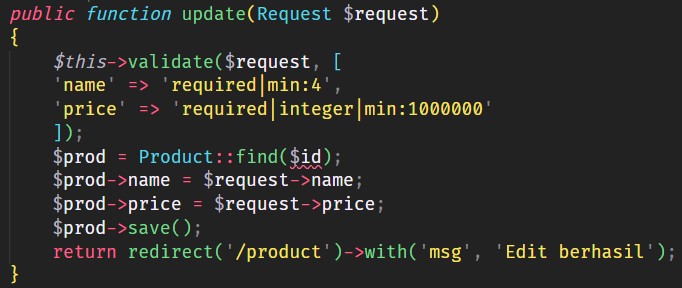
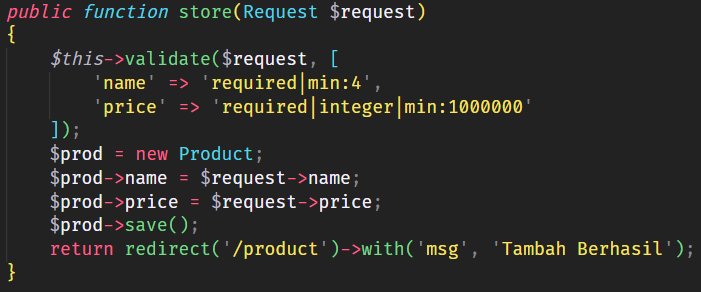
Tambahkan routing pada file web.php

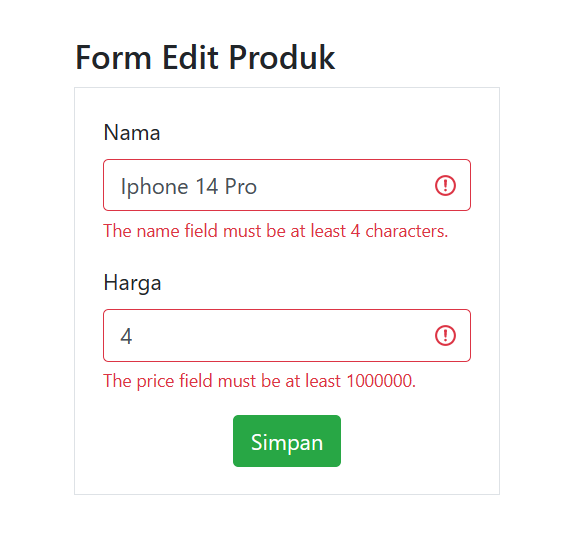
* <http://127.0.0.1:8000/product>
* http://127.0.0.1:8000/product/create
* <http://127.0.0.1:8000/product> saat telah menambah produk.
* [http://127.0.0.1:8000/product/{id}/edit](http://127.0.0.1:8000/product/%7bid%7d/edit)
* <http://127.0.0.1:8000/product> saat delete.
  1. **Templating Halaman**

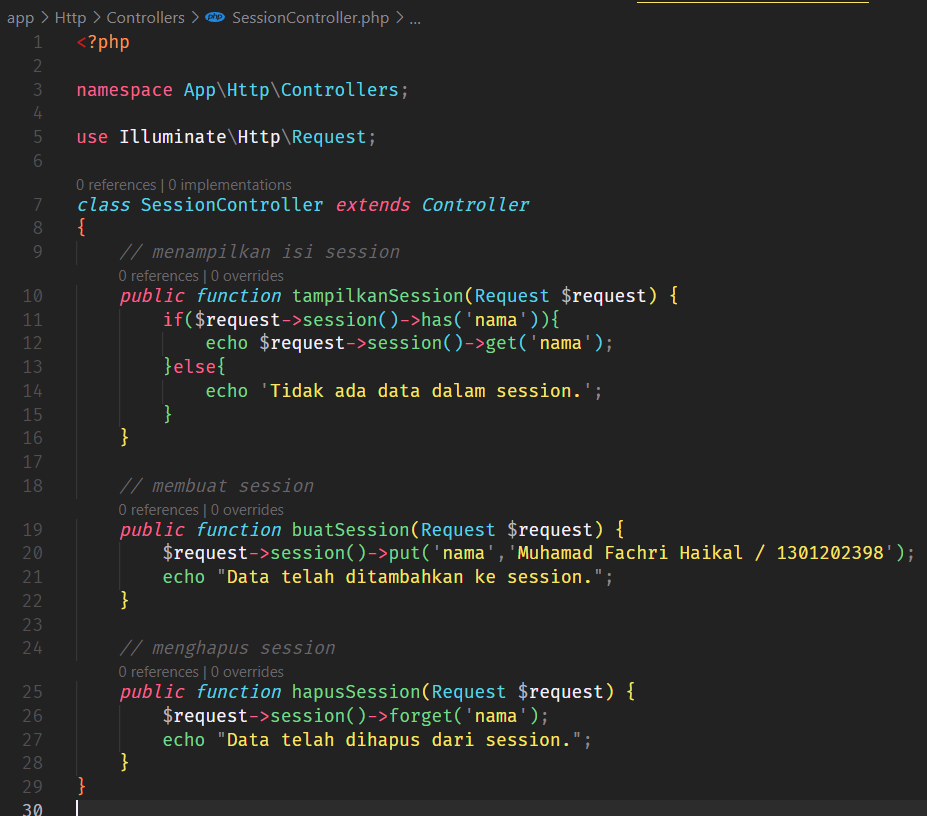
Laravel menyediakan fitur templating yang disebut dengan Blade. Blade memungkinkan pengembang web untuk membuat template yang dapat digunakan berulang kali dan dapat diperluas dengan mudah. Contohnya adalah berikut:

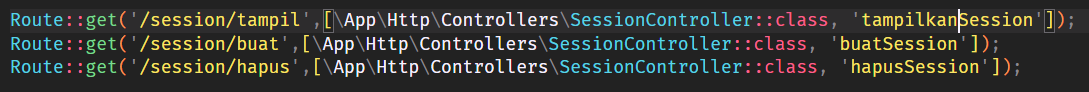
Digunakan directive @foreach untuk melakukan perulangan pada array $list. Setiap item pada array akan diakses sebagai variabel $d pada setiap iterasi perulangan.

* 1. **Form Validation**

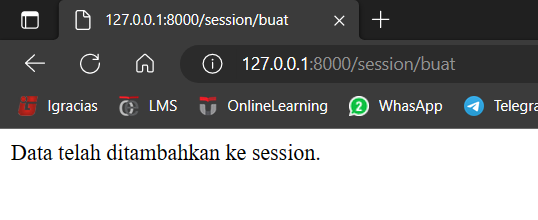
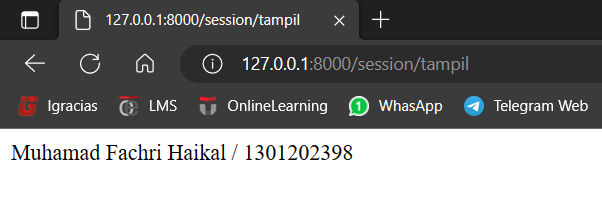
Penggunaan validasi pada request biasanya digunakan pada method store atau update biasanya untuk memvalidasi inputan dari user.

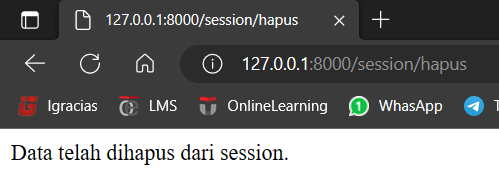


* 1. **Session**

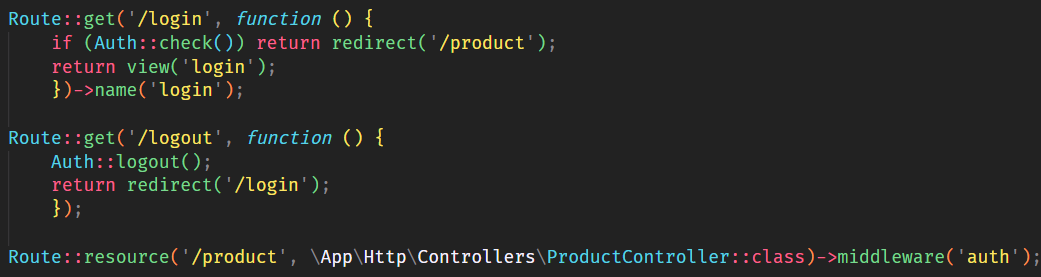
Membuat session controller untuk membuat, menampilkan, dan menghapus sebuah session.

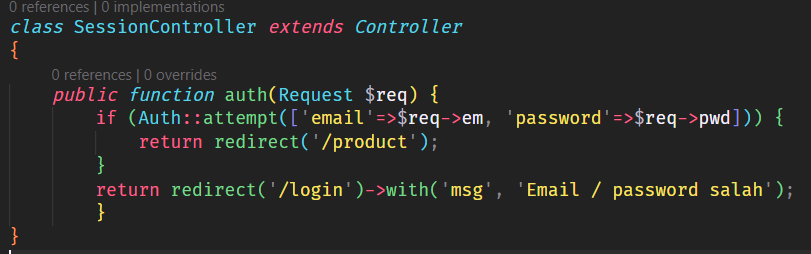
Membuat routingnya.

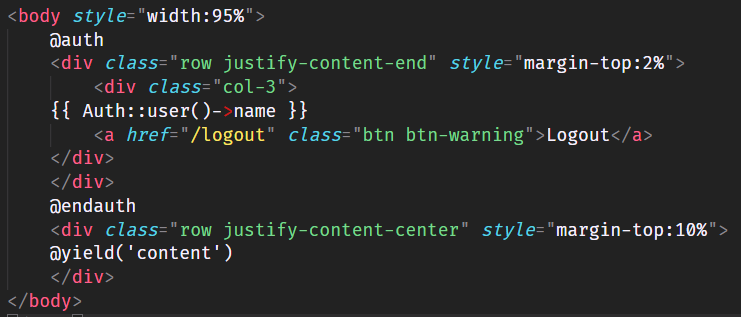
Membuat session.

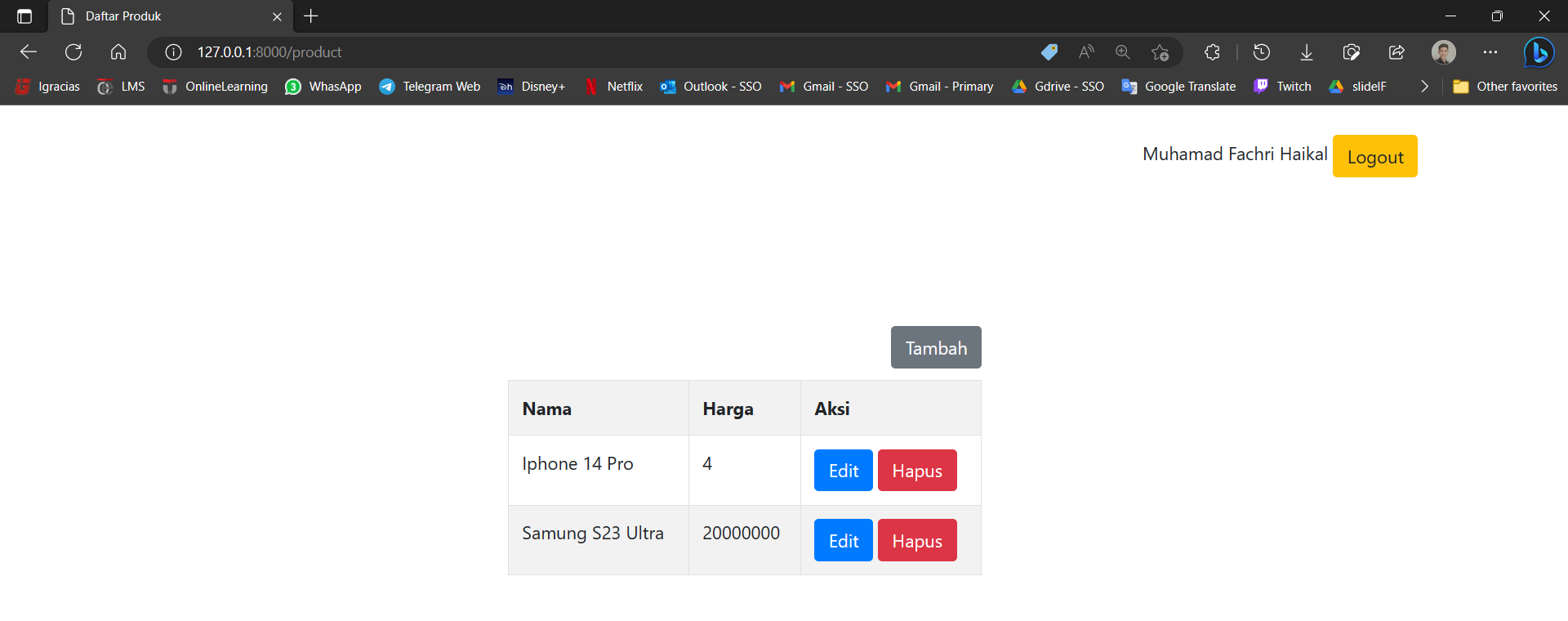
Menampilkan isi session.

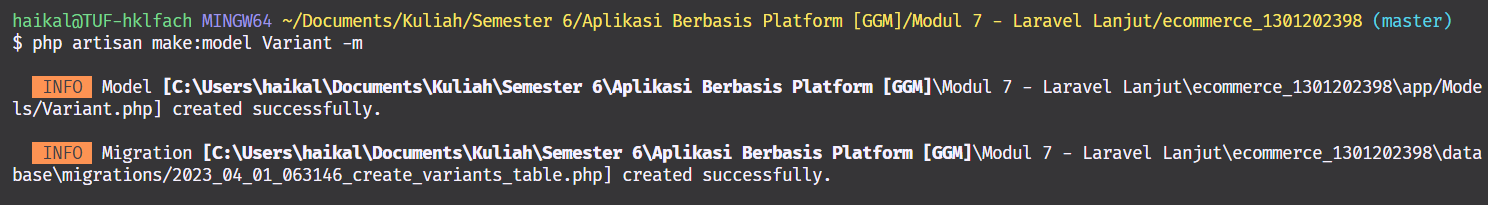
Menghapus session.

* 1. **Middleware**

****Membuat routing untuk login, logout, dan product. Auth::check() berguna untuk mengecek apakah user terautentikasi. Jika terautentikasi maka redirect ke halaman product. Auth::logout() berguna untuk menghapus autentikasi.

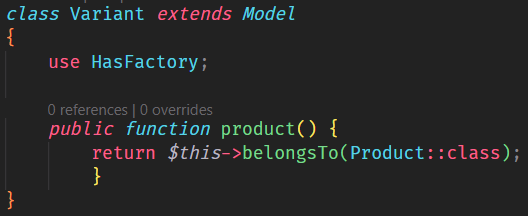
****Fungsi Auth::attempt() digunakan untuk proses autentikasi, yaitu apakah email dan password yang di-submit ada dalam database pada tabel users.

****directive @auth untuk pengecekan apakah user telah terautentikasi, dan menggunakan Auth::user() untuk mengakses data user yang login.

* 1. **Model Relasi**

****Membuat model dan table variant.

Model variant berelasi dengan model product.

****Membuat relasi pada setiap model agar saling terhubung.

Tiap data product dapat mengakses semua data variant-nya dengan fungsi variants() dan tiap variant dapat mengakses product yang berelasi dengannya dengan menggunakan atribut product.