

Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut

## 7.1. CRUD

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

### 7.1.1. Konfigurasi dan Skema

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=ecommerce_1301202398
DB_USERNAME=root
DB_PASSWORD=
```

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

```
haikal@TUF-hklfach MINGW64 ~/Documents/Kuliah/Semester 6/Aplikasi Berbasis Platform [GGM]/Modul 7 - Laravel Lanjut/ecommerce_1301202398 (master)
$ php artisan make:migration create_products_table

[INFO] Migration [C:\Users\haikal\Documents\Kuliah\Semester 6\Aplikasi Berbasis Platform [GGM]\Modul 7 - Laravel Lanjut\ecommerce_1301202398\database\migrations\2023_03_31_153104_create_products_table.php] created successfully.
```

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

```
Modul 7 - Laravel Lanjut > ecommerce_1301202398 > database > migrations > 2023_03_31_153104_create_products_table.php >
11      ~/  
12      public function up(): void  
13      {  
14          Schema::create('products', function (Blueprint $table) {  
15              $table->id();  
16              $table->string('name');  
17              $table->integer('price');  
18              $table->timestamps();  
19          });  
20      }
```

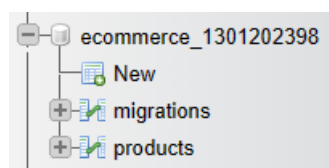
File <date\_created>\_create\_products\_table.php akan terbuat pada folder database/migrations. Kemudian ditambahkan kolom name dan price.

```
haikal@TUF-hklfach MINGW64 ~/Documents/Kuliah/Semester 6/Aplikasi Berbasis Platform [GGM]/Modul 7 - Laravel Lanjut/ecommerce_1301202398 (master)
$ php artisan migrate --path='database/migrations/2023_03_31_173140_create_products_table.php'

[INFO] Running migrations.

2023_03_31_173140_create_products_table ..... 192ms DONE
```

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



Hasil dari tabel yang telah dibuat pada database.

Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut  
**7.1.2.Model**

```
haikal@TUF-hklfach MINGW64 ~/Documents/Kuliah/Semester 6/Aplikasi Berbasis Platform [GGM]/Modul 7 - Laravel Lanjut/ecommerce_1301202398
(master)
$ php artisan make:model Product

[INFO] Model [C:\Users\haikal\Documents\Kuliah\Semester 6\Aplikasi Berbasis Platform [GGM]\Modul 7 - Laravel Lanjut\ecommerce_1301202398\app\Models/Product.php] created successfully.
```

```
app > Models > Product.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  0 references | 0 implementations
9  class Product extends Model
10 {
11     use HasFactory;
12 }
```

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

### 7.1.3.Controller

```
haikal@TUF-hklfach MINGW64 ~/Documents/Kuliah/Semester 6/Aplikasi Berbasis Platform [GGM]/Modul 7 - Laravel Lanjut/ecommerce_1301202398
(master)
$ php artisan make:controller ProductController --resource

[INFO] Controller [C:\Users\haikal\Documents\Kuliah\Semester 6\Aplikasi Berbasis Platform [GGM]\Modul 7 - Laravel Lanjut\ecommerce_1301202398\app\Http\Controllers/ProductController.php] created successfully.
```

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

```
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Product;

0 references | 0 implementations
class ProductController extends Controller
{
    /**
     * Display a listing of the resource.
     */
    0 references | 0 overrides
    public function index()
    {
        $prods = Product::get();
        return view('product.index', ['list' => $prods]);
    }

    /**
     * Show the form for creating a new resource.
     */
    0 references | 0 overrides
    public function create()
    {
        return view('product.form', [
            'title' => 'Tambah',
            'method' => 'POST',
            'action' => 'product'
        ]);
    }

    /**
     * Store a newly created resource in storage.
     */
    0 references | 0 overrides
    public function store(Request $request)
    {
        $prod = new Product;
        $prod->name = $request->name;
        $prod->price = $request->price;
        $prod->save();
        return redirect('/product')->with('msg', 'Tambah berhasil');
    }
}
```

```
    /**
     * Display the specified resource.
     */
    1 reference | 0 overrides
    public function show(string $id)
    {
        return Product::find($id);
    }

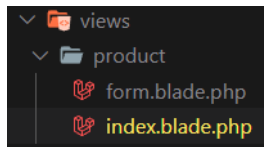
    /**
     * Show the form for editing the specified resource.
     */
    0 references | 0 overrides
    public function edit(string $id)
    {
        return view('product.form', [
            'title' => 'Edit',
            'method' => 'PUT',
            'action' => 'product/$id',
            'data' => Product::find($id)
        ]);
    }

    /**
     * Update the specified resource in storage.
     */
    1 reference | 0 overrides
    public function update(Request $request, string $id)
    {
        $prod = Product::find($id);
        $prod->name = $request->name;
        $prod->price = $request->price;
        $prod->save();
        return redirect('/product')->with('msg', 'Edit berhasil');
    }

    /**
     * Remove the specified resource from storage.
     */
    1 reference | 0 overrides
    public function destroy(string $id)
    {
        Product::destroy($id);
        return redirect('/product')->with('msg', 'Delete berhasil');
    }
}
```

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

Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut  
**7.1.4.View**



Struktur direktori untuk interface product.

```
<!DOCTYPE html>
<html>
<head>
<title>Daftar Produk</title>
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css"
integrity="sha384-zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2ylzuuKfdKlfPg/eSrtxUkn" crossorigin="anonymous">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-FqybjgWLrvvRgtW6bflB7jazzrfsaBxjSOmm/tB9LTS80NXqgbR9W80Wht/ampnF" crossorigin="anonymous"></script>
</head>
<body style="width:95%">
<div class="container text-center">
<p>Praktikum ABP MOD 7</p>
<p>1301202398</p>
<p>Muhamad Fachri Haikal</p>
</div>
<div class="row justify-content-center" style="margin-top:13%">
<div class="col-4">
<span class="float-left">{{ session('msg') }}</span>
<a href="/product/create" class="btn btn-secondary" style="float:right">Tambah</a><br />
<table class="table table-bordered table-striped">
<tr>
<th>Nama</th>
<th>Harga</th>
<th>Aksi</th>
</tr>
<tr>
<td colspan="3">@foreach($list as $d)
<tr>
<td>{{ $d->name }}</td>
<td>{{ $d->price }}</td>
<td>
<a href="/product/{{ $d->id }}/edit" class="btn btn-primary">Edit</a>
<form method="post" action="/product/{{ $d->id }}" style="display:inline" onsubmit="return confirm('Yakin hapus?')">
@csrf
@method('DELETE')
<button class="btn btn-danger">Hapus</button>
</form>
</td>
</tr>
<tr>
<td colspan="3">@endforeach
</td>
</tr>
</table>
</div>
</div>
</body>
</html>
```

Isi file index.blade.php untuk product.

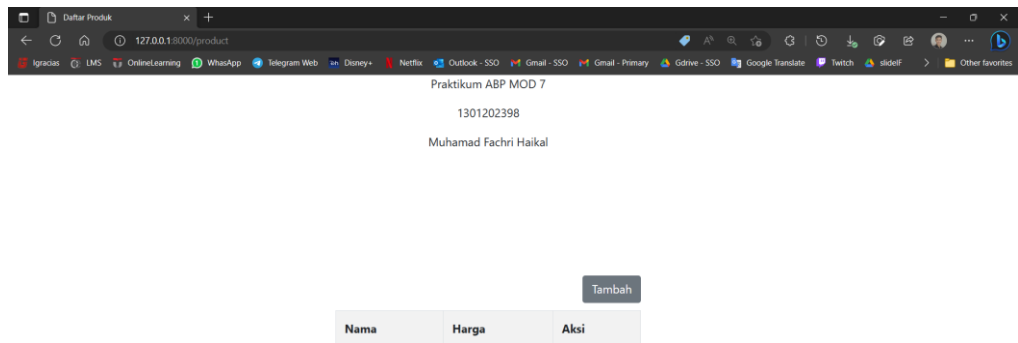
```
<!DOCTYPE html>
<html>
<head>
<title>Form {{ $title }} Produk</title>
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.css"
integrity="sha384-zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2ylzuuKfdKlfPg/eSrtxUkn" crossorigin="anonymous">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-FqybjgWLrvvRgtW6bflB7jazzrfsaBxjSOmm/tB9LTS80NXqgbR9W80Wht/ampnF" crossorigin="anonymous"></script>
</head>
<body style="width:95%">
<div class="container text-center">
<p>Praktikum ABP MOD 7</p>
<p>1301202398</p>
<p>Muhamad Fachri Haikal</p>
</div>
<div class="row justify-content-center" style="margin-top:13%">
<div class="col-3">
<h4>Form {{ $title }} Produk</h4>
<form class="border" style="padding:20px" method="POST" action="{{ $action }}">
@csrf
<input type="hidden" name="_method" value="{{ $method }}" />
<div class="form-group">
<label>Nama</label>
<input type="text" name="name" class="form-control" value="{{isset($data)?$data->name:'' }}" />
</div>
<div class="form-group">
<label>Harga</label>
<input type="number" name="price" class="form-control" value="{{isset($data)?$data->price:'' }}" />
</div>
<div style="text-align:center">
<button class="btn btn-success">Simpan</button>
</div>
</form>
</div>
</div>
</body>
</html>
```

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

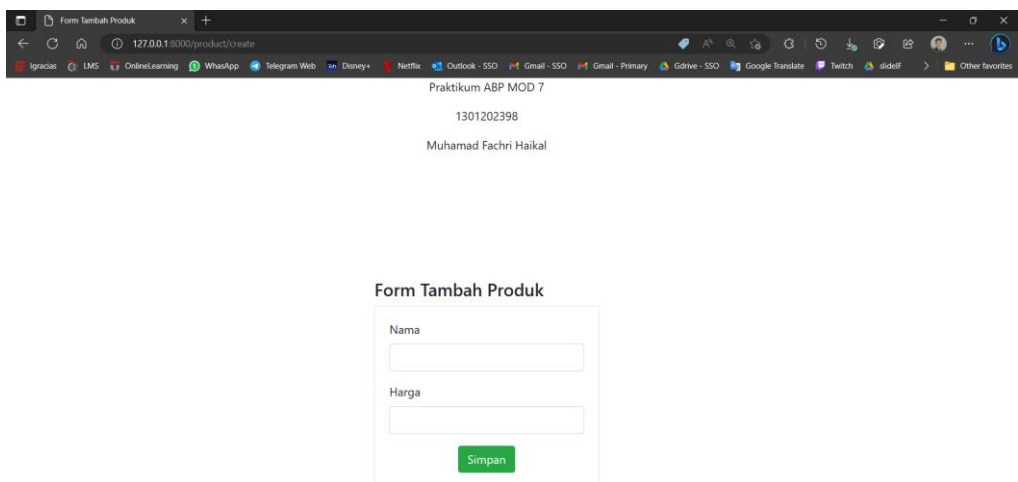
Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut  
**7.1.5. Tampilan Halaman**

```
Route::get('/product', [\App\Http\Controllers\ProductController::class, 'index']);  
Route::get('/product/create', [\App\Http\Controllers\ProductController::class, 'create']);  
Route::get('/product/{id}/edit', [\App\Http\Controllers\ProductController::class, 'edit']);  
Route::get('/product/{id}', [\App\Http\Controllers\ProductController::class, 'show']);  
Route::put('/product/{id}', [\App\Http\Controllers\ProductController::class, 'update']);  
Route::post('/product', [\App\Http\Controllers\ProductController::class, 'store']);  
Route::delete('/product/{id}', [\App\Http\Controllers\ProductController::class, 'destroy']);
```

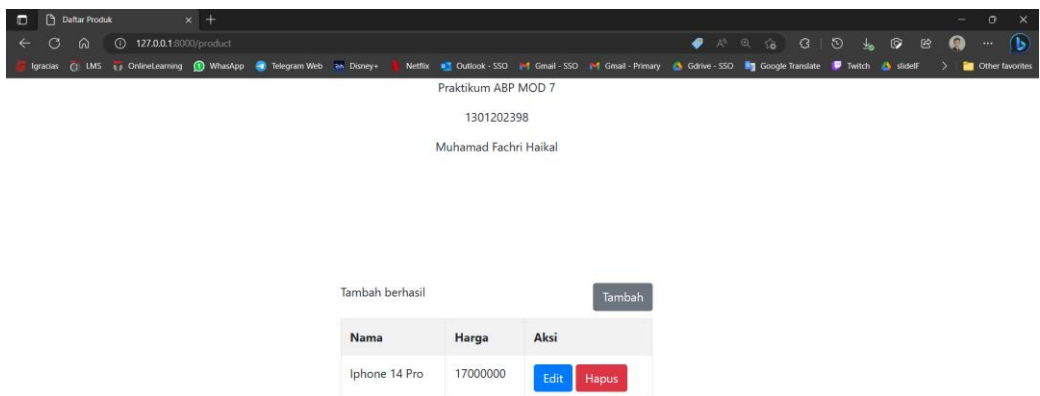
Tambahkan routing pada file web.php



- <http://127.0.0.1:8000/product>



- <http://127.0.0.1:8000/product/create>



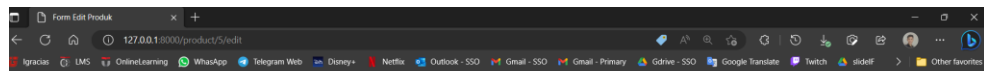
Nama: Muhamad Fachri Haikal

NIM : 1301202398

Kelas : IF4401

Praktikum ABP MOD 7 Laravel Lanjut

- <http://127.0.0.1:8000/product> saat telah menambah produk.



Praktikum ABP MOD 7

1301202398

Muhamad Fachri Haikal

#### Form Edit Produk

Nama

Iphone 14 Pro

Harga

17000000

Simpan

- <http://127.0.0.1:8000/product/{id}/edit>



Praktikum ABP MOD 7

1301202398

Muhamad Fachri Haikal

Delete Berhasil

Tambah

Nama

Harga

Aksi

- <http://127.0.0.1:8000/product> saat delete.

## 7.2. 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:

```
<tr>
<th>Nama</th>
<th>Harga</th>
<th>Aksi</th>
</tr>
@foreach($list as $d)
<tr>
<td>{{ $d->name }}</td>
<td>{{ $d->price }}</td>
<td>
```

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

Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut

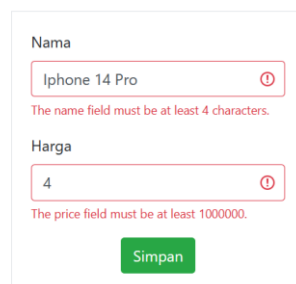
### 7.3. Form Validation

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

```
public function update(Request $request)
{
    $this->validate($request, [
        'name' => 'required|min:4',
        'price' => 'required|integer|min:1000000'
    ]);
    $prod = Product::find($id);
    $prod->name = $request->name;
    $prod->price = $request->price;
    $prod->save();
    return redirect('/product')->with('msg', 'Edit berhasil');
}
```

```
public function store(Request $request)
{
    $this->validate($request, [
        'name' => 'required|min:4',
        'price' => 'required|integer|min:1000000'
    ]);
    $prod = new Product;
    $prod->name = $request->name;
    $prod->price = $request->price;
    $prod->save();
    return redirect('/product')->with('msg', 'Tambah Berhasil');
}
```

#### Form Edit Produk



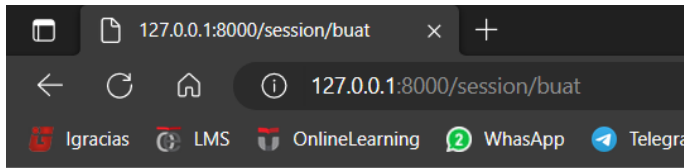
### 7.4. Session

```
app > Http > Controllers > SessionController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  0 references | 0 implementations
8  class SessionController extends Controller
9  {
10     // menampilkan isi session
11     0 references | 0 overrides
12     public function tampilkanSession(Request $request) {
13         if($request->session()->has('nama')){
14             echo $request->session()->get('nama');
15         }else{
16             echo 'Tidak ada data dalam session.';
17         }
18     }
19
20     // membuat session
21     0 references | 0 overrides
22     public function buatSession(Request $request) {
23         $request->session()->put('nama','Muhamad Fachri Haikal / 1301202398');
24         echo "Data telah ditambahkan ke session.";
25     }
26
27     // menghapus session
28     0 references | 0 overrides
29     public function hapusSession(Request $request) {
30         $request->session()->forget('nama');
31         echo "Data telah dihapus dari session.";
32     }
33 }
```

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

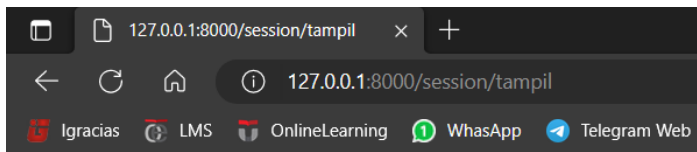
```
Route::get('/session/tampil',[App\Http\Controllers\SessionController::class, 'tampilkanSession']);
Route::get('/session/buat',[App\Http\Controllers\SessionController::class, 'buatSession']);
Route::get('/session/hapus',[App\Http\Controllers\SessionController::class, 'hapusSession']);
```

Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut  
Membuat routingnya.



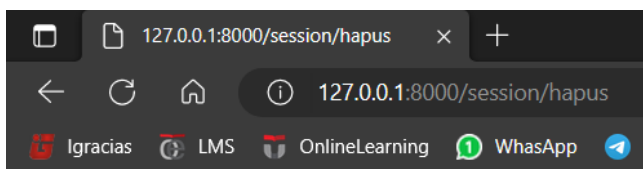
Data telah ditambahkan ke session.

Membuat session.



Muhamad Fachri Haikal / 1301202398

Menampilkan isi session.



Data telah dihapus dari session.

Menghapus session.

## 7.5. Middleware

```
Route::get('/login', function () {  
    if (Auth::check()) return redirect('/product');  
    return view('login');  
})->name('login');  
  
Route::get('/logout', function () {  
    Auth::logout();  
    return redirect('/login');  
});  
  
Route::resource('/product', \App\Http\Controllers\ProductController::class)->middleware('auth');
```

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.

```
0 references | 0 implementations  
class SessionController extends Controller  
{  
    0 references | 0 overrides  
    public function auth(Request $req) {  
        if (Auth::attempt(['email'=>$req->em, 'password'=>$req->pwd])) {  
            return redirect('/product');  
        }  
        return redirect('/login')->with('msg', 'Email / password salah');  
    }  
}
```

Nama: Muhamad Fachri Haikal

NIM : 1301202398

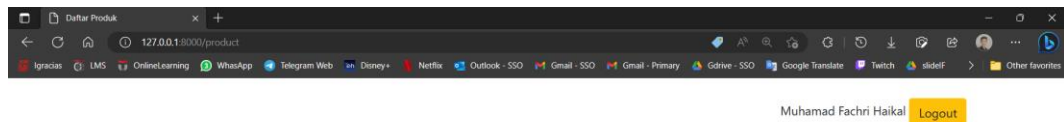
Kelas : IF4401

Praktikum ABP MOD 7 Laravel Lanjut

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

```
<body style="width:95%">
  @auth
    <div class="row justify-content-end" style="margin-top:2%">
      <div class="col-3">
        {{ Auth::user()->name }}
        <a href="/logout" class="btn btn-warning">Logout</a>
      </div>
    </div>
  @endauth
  <div class="row justify-content-center" style="margin-top:10%">
    @yield('content')
  </div>
</body>
```

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



Tambah		
Nama	Harga	Aksi
Iphone 14 Pro	4	<a href="#">Edit</a> <a href="#">Hapus</a>
Samung S23 Ultra	20000000	<a href="#">Edit</a> <a href="#">Hapus</a>

## 7.6. Model Relasi

```
haikal@TUF-hk1fach MINGW64 ~/Documents/Kuliah/Semester 6/Aplikasi Berbasis Platform [GGM]/Modul 7 - Laravel Lanjut/ecommerce_1301202398 (master)
$ php artisan make:model Variant -m

[INFO] Model [C:\Users\haikal\Documents\Kuliah\Semester 6\Aplikasi Berbasis Platform [GGM]\Modul 7 - Laravel Lanjut\ecommerce_1301202398\app\Models\Variant.php] created successfully.

[INFO] Migration [C:\Users\haikal\Documents\Kuliah\Semester 6\Aplikasi Berbasis Platform [GGM]\Modul 7 - Laravel Lanjut\ecommerce_1301202398\database\Migrations\2023_04_01_063146_create_variants_table.php] created successfully.
```

Membuat model dan table variant.

```
Schema::create('variants', function (Blueprint $table) {
    $table->id();
    $table->string('name');
    $table->text('description');
    $table->string('processor');
    $table->string('memory');
    $table->string('storage');
    $table->foreignId('product_id')->constrained();
    $table->timestamps();
});
```

Model variant berelasi dengan model product.



Nama: Muhamad Fachri Haikal  
NIM : 1301202398  
Kelas : IF4401  
Praktikum ABP MOD 7 Laravel Lanjut

```
0 references | 0 implementations
class Product extends Model
{
    use HasFactory;
    0 references
    protected $fillable = [
        'id',
        'name',
        'price',
        'created_at',
        'updated_at',
    ];
    0 references | 0 overrides
    public function variants() {
        return $this->hasMany(Variant::class);
    }
}
```

```
class Variant extends Model
{
    use HasFactory;
    0 references | 0 overrides
    public function product() {
        return $this->belongsTo(Product::class);
    }
}
```

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.