**Bài thực hành 09**

**Authorization**

**Mục tiêu**

* Bảo mật, phân quyền hành động thao tác dữ liệu người dùng.

**Cho CSDL gồm các bảng sau (Có thể kế thừa từ lab trước):**

categories, products, customers, comments.

Chi tiết cầu trúc các bảng

**Danh mục (categories)**

| **Tên cột** | **Kiểu DL** | **Độ dài** | **Ràng buộc** |
| --- | --- | --- | --- |
| **id** | int |  | Khóa chính, tự động tăng |
| name | varchar | 100 | Không rỗng, không trùng nhau |
| status | number |  |  |
| created\_at | date |  | Mặc định ngày hiện tại |
| updated\_at | timestamp |  | Mặc định ngày hiện tại |

**Sản phẩm (products)**

| **Tên cột** | **Kiểu DL** | **Độ dài** | **Ràng buộc** |
| --- | --- | --- | --- |
| **id** | Int |  | Khóa chính, tự động tăng |
| name | Varchar | 120 | Không rỗng |
| price | Float |  | Không rỗng |
| sale\_price | Float |  | Được phép rỗng, mạc định 0 |
| image | Varchar | 200 | Không rỗng |
| category\_id | Int |  | Không rỗng, khóa ngoại tới bảng categories |
| status | tinyint |  | Được phép rỗng, mạc định 1 |
| content | text |  | Không rỗng |
| created\_at | date |  | Mặc định ngày hiện tại |
| updated\_at | timestamp |  | Mặc định ngày hiện tại |

**Khách hàng (customers)**

| **Tên cột** | **Kiểu DL** | **Độ dài** | **Ràng buộc** |
| --- | --- | --- | --- |
| **id** | int |  | Khóa chính, tự động tăng |
| name | varchar | 100 | Không rỗng |
| email | varchar | 100 | Không rỗng, không trùng nhau |
| password | varchar | 100 | Không rỗng |
| phone | varchar | 50 | Không rỗng, không trùng |
| address | varchar | 250 | rỗng |
| created\_at | date |  | Mặc định ngày hiện tại |
| updated\_at | timestamp |  | Mặc định ngày hiện tại |

**Bình luận (comments)**

| **Tên cột** | **Kiểu DL** | **Độ dài** | **Ràng buộc** |
| --- | --- | --- | --- |
| **id** | int |  | Khóa chính, tự động tăng |
| customer\_id | Int |  | Không rỗng, khóa ngoại tới bảng customers |
| content | Text |  | Không rỗng |
| status | Tinyint |  | Mặc định 0 |
| created\_at | date |  | Mặc định ngày hiện tại |
| updated\_at | timestamp |  | Mặc định ngày hiện tại |

**Chú ý:** *Sinh viên thực hiện chức năng đăng nhập, đăng ký tương tự lab 08.2, nếu đã làm rồi thì bỏ qua*

*Thêm dữ liệu demo cho bảng categories và bảng products*

Review lại dự án trước đó có thể có

**Master view (view chính) trong thư mục resources/views/master/home.blade.php**

<!DOCTYPE htm>

<html lang="">

<head>

    <meta charset="utf-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Title Page</title>

    <!-- Bootstrap CSS -->

    <link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

    <nav class="navbar navbar-inverse">

        <div class="container">

            <a class="navbar-brand" href="#">My Shopping</a>

            <ul class="nav navbar-nav">

                <li><a href="{{ route('home.index') }}">Home</a></li>

                <li><a href="#">About</a></li>

                <li><a href="#">Blogs</a></li>

                <li><a href="#">Contact</a></li>

            </ul>

        </div>

    </nav>

    <div class="container">

        @yield('main')

    </div>

    <!-- jQuery -->

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>

    <!-- Bootstrap JavaScript -->

    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

</body>

</html>

# Bài 1: Tạo chức năng xem danh sách và chi tiết sản phẩm

**Bước 1:** Tạo danh sách sản phẩm ở trang chủ

Xem lại **routes/web.php**

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController;

/\*

|--------------------------------------------------------------------------

| Web Routes

|--------------------------------------------------------------------------

|

| Here is where you can register web routes for your application. These

| routes are loaded by the RouteServiceProvider within a group which

| contains the "web" middleware group. Now create something great!

|

\*/

Route::get('', [HomeController::class, 'index'])->name('home.index');

// route sẽ hiển thị form login

Route::get('login', [HomeController::class, 'login'])->name('home.login');

// route này validate dữ liệu khi submit form

Route::post('login', [HomeController::class, 'check\_login']);

Tạo Model Product (nếu chwua có), code tham khảo

Lưu ý: model **Product** sẽ tự map với bảng **products**

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

use App\Scopes\AncientScope;

class Product extends Model

{

    use HasFactory;

    protected $fillable = ['name','price','sale\_price','image','content','category\_id','status'];

}

Mở lại **HomeController**, tìm đến phương thức index (trang chủ) truy vấn danh sách sản phẩm để hiển thị tại trang chủ như sau

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Product;

class HomeController extends Controller

{

    public function index()

    {

        $products = Product::orderBy('id','desc')->limit(4)->get();

        return view('index', compact('products'));

    }

    public function login()

    {

        // Hiển thị form login

        return view('login');

    }

    public function check\_login(Request $req)

    {

        // Khai báo các quy ràng buộc xác thực

        $rules = [

            'email' => 'required|email',

            'password' => 'required'

        ];

        // customize các tin nhắn

        $message = [

            'email.required' => 'Địa chỉ email không được để trống',

            'email.email'    => 'Địa chỉ email không đúng định dạng',

            'password.required' => 'Mật khẩu không được để trống'

        ];

        // nếu Các ràng buộc đã hợp lệ, thì xử lý tiếp

        $req->validate($rules, $message);

    }

}

Mở lại file **views/index.blade.php (chưa có thì tạo mới)**

@extends('master.home')

@section('main')

<h2>Sản phẩm mới nhất</h2>

<div class="row products">

    @foreach($products as $pro)

    <div class="col-md-3">

        <div class="thumbnail">

            <img src="https://image.kacana.vn/images/product/gio-xach-dep-hang-hieu-tui-nu-fsm-2303-4013216754.jpg"

                alt="">

            <div class="caption">

                <h3>{{ $pro->name }}</h3>

                <p>

                    <s>Giá gốc: {{ $pro->price }}</s>

                    <b>Giá mới: {{ $pro->sale\_price }}</b>

                </p>

                <p>

                    <a href="#" class="btn btn-primary">View</a>

                    <a href="#" class="btn btn-default">Add To Cart</a>

                </p>

            </div>

        </div>

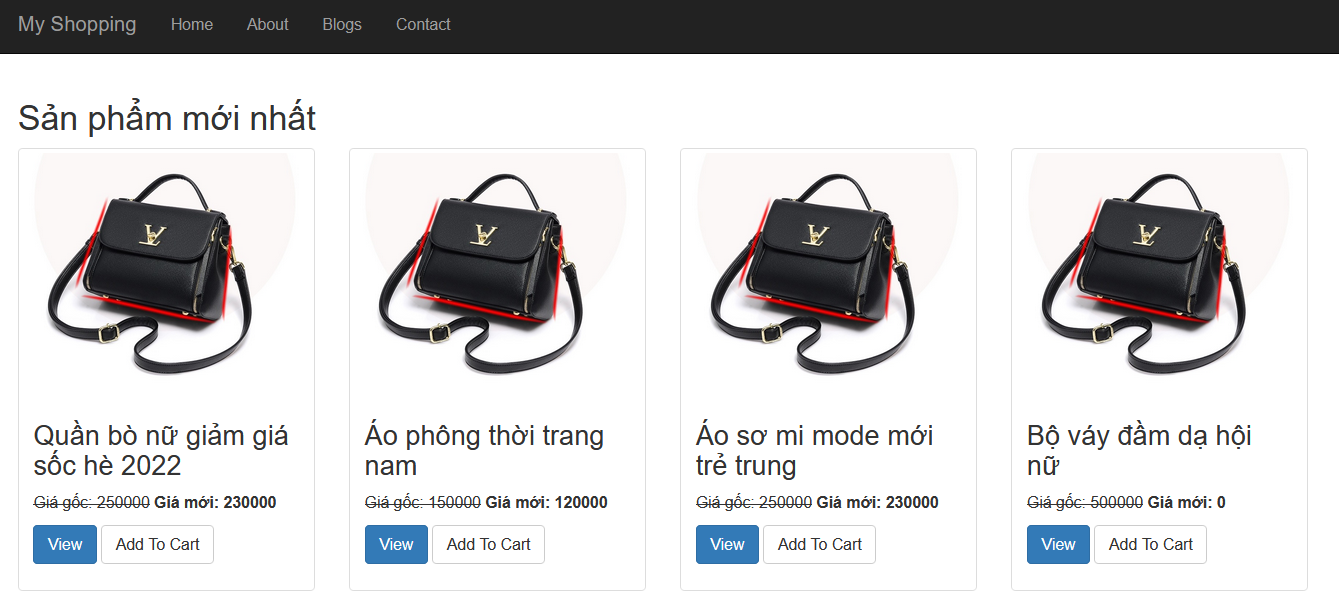
    </div>

    @endforeach

</div>

@stop()

Kết quả tương tự như sau



**Bước 2:** Tạo chức năng xem chi tiết sản phẩm

Mở file **routes/web.php** thêm route xem chi tiết sản phẩm

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController;

/\*

|--------------------------------------------------------------------------

| Web Routes

|--------------------------------------------------------------------------

|

| Here is where you can register web routes for your application. These

| routes are loaded by the RouteServiceProvider within a group which

| contains the "web" middleware group. Now create something great!

|

\*/

Route::get('', [HomeController::class, 'index'])->name('home.index');

Route::get('product/{product}', [HomeController::class, 'product'])->name('home.product\_detail');

// route sẽ hiển thị form login

Route::get('login', [HomeController::class, 'login'])->name('home.login');

// route này validate dữ liệu khi submit form

Route::post('login', [HomeController::class, 'check\_login']);

Tiếp theo mở lại **HomeController** thêm phương thức product và code tương tự như sau

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Product;

class HomeController extends Controller

{

    public function index()

    {

        $products = Product::orderBy('id','desc')->limit(4)->get();

        return view('index', compact('products'));

    }

    public function product(Product $product)

    {

        return view('product-detail', compact('product'));

    }

    public function login()

    {

        // Hiển thị form login

        return view('login');

    }

    public function check\_login(Request $req)

    {

        // Khai báo các quy ràng buộc xác thực

        $rules = [

            'email' => 'required|email',

            'password' => 'required'

        ];

        // customize các tin nhắn

        $message = [

            'email.required' => 'Địa chỉ email không được để trống',

            'email.email'    => 'Địa chỉ email không đúng định dạng',

            'password.required' => 'Mật khẩu không được để trống'

        ];

        // nếu Các ràng buộc đã hợp lệ, thì xử lý tiếp

        $req->validate($rules, $message);

    }

}

**Bước 3:** Tạo views/product-detail.blade.php và soạn code hiển thị chi tiết sản phẩm tương tự như sau

@extends('master.home')

@section('main')

<h2>Chi tiết sản phẩm</h2>

<div class="row products">

    <div class="col-md-5">

        <img src="https://image.kacana.vn/images/product/gio-xach-dep-hang-hieu-tui-nu-fsm-2303-4013216754.jpg" alt=""

            style="width:100%">

    </div>

    <div class="col-md-7">

        <h3>{{ $product->name }}</h3>

        <p style="font-size: 18px">

            <s>Giá gốc: {{ $product->price }}</s>

            <b>Giá mới: {{ $product->sale\_price }}</b>

        </p>

        <p>

        <form action="" method="POST" class="form-inline" role="form">

            <div class="form-group">

                <input class="form-control" name="quantity" placeholder="Input quantity">

            </div>

            <button type="submit" class="btn btn-success">Add To Cart</button>

        </form>

        </p>

    </div>

</div>

@stop()

**Bước 4:** Mở lại file views/index.blade.php tìm đến nút bấm xem chi tiết đặt link như sau

@extends('master.home')

@section('main')

<h2>Sản phẩm mới nhất</h2>

<div class="row products">

    @foreach($products as $pro)

    <div class="col-md-3">

        <div class="thumbnail">

            <img src="https://image.kacana.vn/images/product/gio-xach-dep-hang-hieu-tui-nu-fsm-2303-4013216754.jpg"

                alt="">

            <div class="caption">

                <h3>{{ $pro->name }}</h3>

                <p>

                    <s>Giá gốc: {{ $pro->price }}</s>

                    <b>Giá mới: {{ $pro->sale\_price }}</b>

                </p>

                <p>

                    <a href**="{{ route('home.product\_detail', $pro->id) }}"** class="btn btn-primary">View</a>

                    <a href="#" class="btn btn-default">Add To Cart</a>

                </p>

            </div>

        </div>

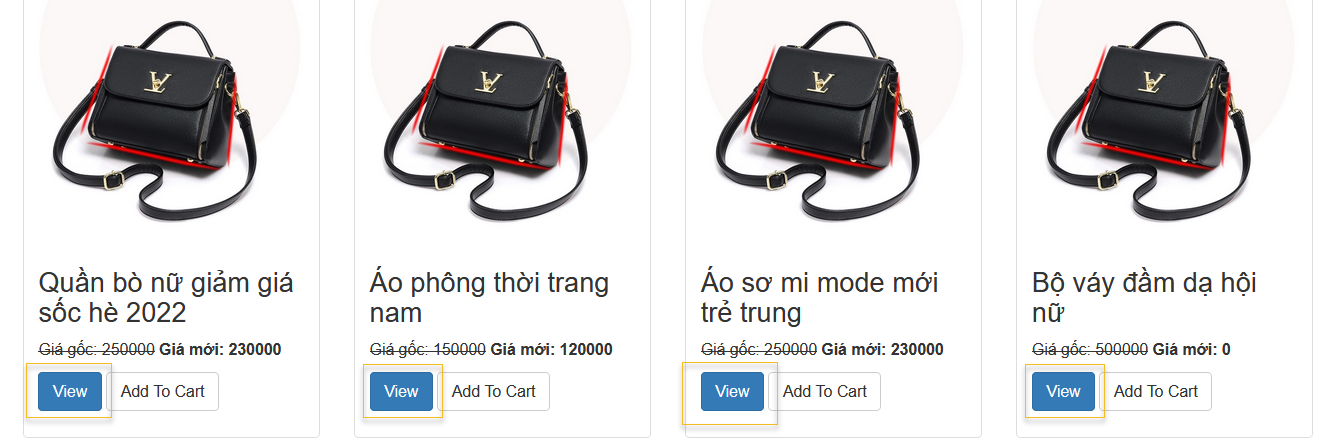
    </div>

    @endforeach

</div>

@stop()

Kết quả khi click vào nút xem sẽ chuyển hướng đến trang chi tiết sản phẩm



Chi tiết sản phẩm



# Bài 2: Tạo chức năng bình luận sản phẩm

**Bước 1:** Model Comment

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class Comment extends Model

{

    use HasFactory;

    protected $fillable = ['content','customer\_id','product\_id'];

}

**Bước 2:** Tạo route cho bình luận

Mở lại file routes/web.php và soạn code sau

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController;

/\*

|--------------------------------------------------------------------------

| Web Routes

|--------------------------------------------------------------------------

|

| Here is where you can register web routes for your application. These

| routes are loaded by the RouteServiceProvider within a group which

| contains the "web" middleware group. Now create something great!

|

\*/

Route::get('', [HomeController::class, 'index'])->name('home.index');

Route::get('product/{product}', [HomeController::class, 'product'])->name('home.product\_detail');

Route::post('product-comment/{product}', [HomeController::class, 'productComment'])->name('home.product\_comment');

// route sẽ hiển thị form login

Route::get('login', [HomeController::class, 'login'])->name('home.login');

// route này validate dữ liệu khi submit form

Route::post('login', [HomeController::class, 'check\_login']);

**Bước 3:** Tạo form bình luận

Mở lại file chi tiết sản phẩm views/product-detail.blade.php

Sau đó tạo form như sau

@extends('master.home')

@section('main')

<h2>Chi tiết sản phẩm</h2>

<div class="row products">

    <div class="col-md-5">

        <img src="https://image.kacana.vn/images/product/gio-xach-dep-hang-hieu-tui-nu-fsm-2303-4013216754.jpg" alt=""

            style="width:100%">

    </div>

    <div class="col-md-7">

        <h3>{{ $product->name }}</h3>

        <p style="font-size: 18px">

            <s>Giá gốc: {{ $product->price }}</s>

            <b>Giá mới: {{ $product->sale\_price }}</b>

        </p>

        <p>

        <form action="" method="POST" class="form-inline" role="form">

            <div class="form-group">

                <input class="form-control" name="quantity" placeholder="Input quantity">

            </div>

            <button type="submit" class="btn btn-success">Add To Cart</button>

        </form>

        </p>

    </div>

</div>

<hr>

<!-- Kiểm tra xem người dùng đã đăng nhập chưa  -->

@if (auth()->guard('customer')->check())

<form action="{{ route('home.product\_comment', $product->id) }}" method="POST" role="form">

 <input type="hidden" name="product\_id" value="{{$product->id}}">

    <legend>Bình luận sản phẩm</legend>

    @csrf

    @if (Session::has('ok'))

    <div class="alert alert-success">

        <button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</button>

        <strong>{{ Session::get('ok') }}!</strong>

    </div>

    @endif

    <div class="form-group">

        <label for="">Nội dung bình luận</label>

        <textarea name="content" class="form-control" placeholder="Nội dung bình luận"></textarea>

        @error('content') {{ $message }} @enderror

    </div>

    <button type="submit" class="btn btn-primary">Gửi bình luận</button>

</form>

@else

<!-- Nếu chwua đăng nhập thì thông báo  -->

<div class="alert">

    <button type="button" class="close" data-dismiss="alert" aria-hidden="true">&times;</button>

    <strong>Chưa đăng nhập</strong> bạn hãy <a href="{{ route('home.login') }}">click vào đây</a> để đăng nhập

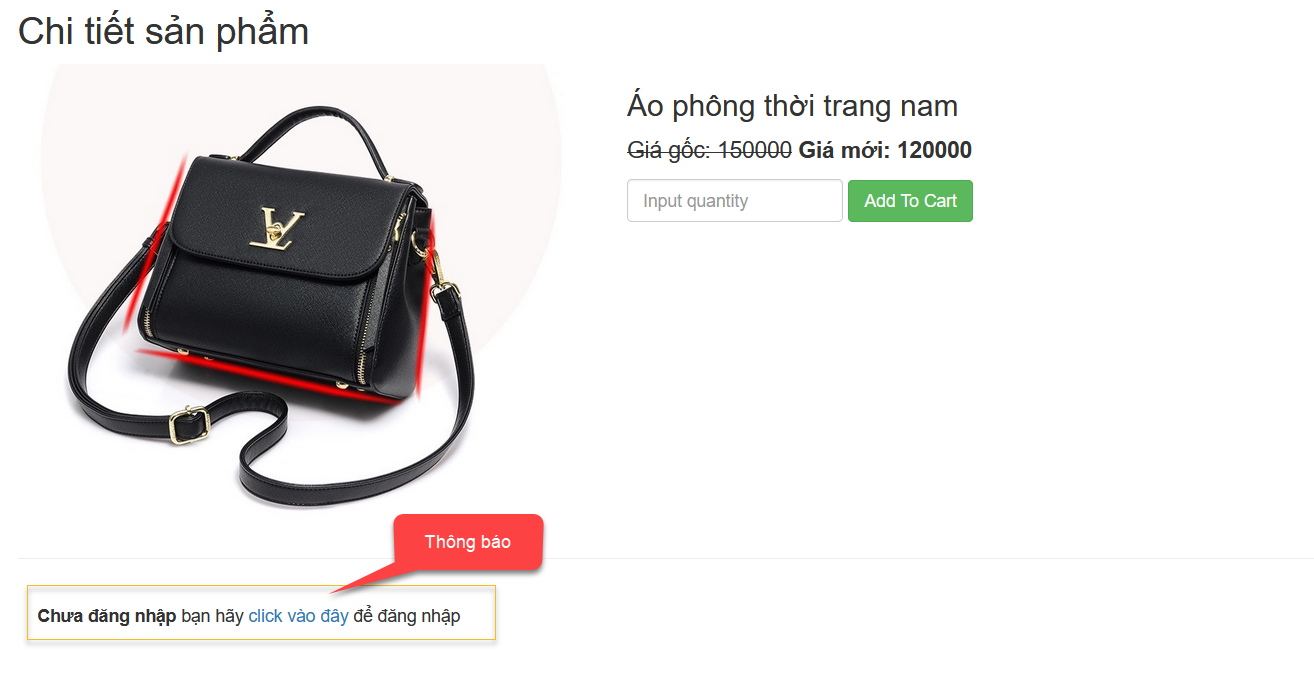
</div>

@endif

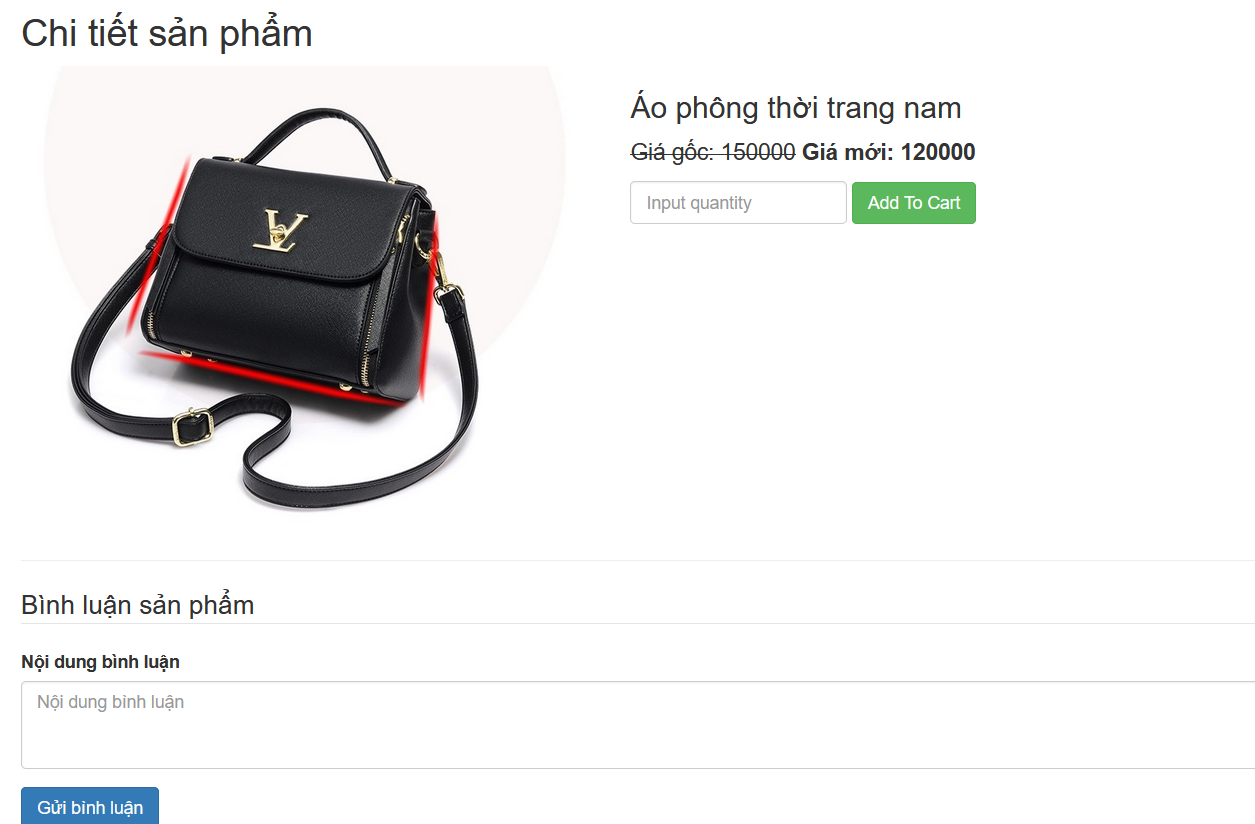
@stop()

Kết quả bước này sẽ cho các trường hợp sau

**Trường hợp 1:** Chưa đăng nhập tì sẽ thông báo lỗi và yêu cầu đăng nhập



**Trường hợp 2:** Đã đăng nhập thì hiển thị form cho phép nhập nội dung bình luận



**Bước 4:** Mở HomeController lên tạo phương thức productComment để xử lý logic khi người dùng nhấn nút Gửi bình luận

public function productComment(Request $req)

{

    $req->validate([

        'content' => 'required|min:10|max:500'

    ], [

        'content.requird' => 'Nội dung bình luận không được để trống',

        'content.min' => 'Nội dung bình luận tối thiểu là 10 ký tự',

        'content.requird' => 'Nội dung bình luận tối đa là 500 ký tự'

    ]);

    Comment::create([

        'customer\_id' => auth()->guard('customer')->id(),

'product\_id' => $req->product\_id,

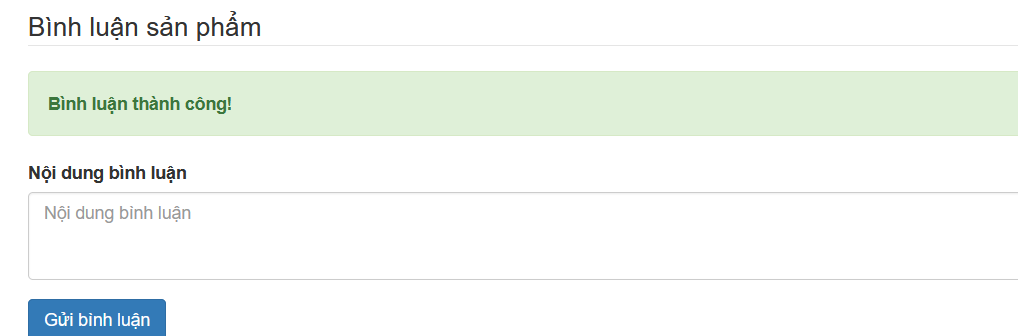
        'content' => $req->content

    ]);

    return redirect()->back()->with('ok','Bình luận thành công');

}

Test thử kết quả khi bình luận thành công sẽ có thông báo kiểu như sau



# Bài 3: Hiển thị danh sách bình luận có phân quyền người dùng

**Bước 1:** Mở model Product tạo phương thức trả về danh sách bình luận của sản phẩm

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

use App\Scopes\AncientScope;

class Product extends Model

{

    use HasFactory;

    protected $fillable = ['name','price','sale\_price','image','content','category\_id','status'];

    public function comments()

    {

        return $this->hasMany(Comment::class, 'product\_id', 'id');

    }

}

**Bước 2:** Mở model Comment, truy vấn thông tin của người dùng bình luận

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;

use Illuminate\Database\Eloquent\Model;

class Comment extends Model

{

    use HasFactory;

    protected $fillable = ['content','customer\_id','product\_id'];

    public function user()

    {

        return $this->hasOne(Customer::class, 'id','customer\_id');

    }

}

**Bước 3:** Tại giao diện trang chi tiết sản phẩm views/product-detail.blade.php, duyệt dữ liệu các bình luận cho sản phẩm đó.

<hr>

<div class="comments">

    @foreach($product->comments as $comment)

    <div class="media">

        <a class="pull-left" href="#">

            <img width="60" class="media-object" src="https://haycafe.vn/wp-content/uploads/2022/03/Avatar-TikTok-anh-dai-dien-TikTok.jpg" alt="Image">

        </a>

        <div class="media-body">

            <h4 class="media-heading">{{$comment->user->name}}</h4>

            <p>{{$comment->content}}</p>

            <a href="" class="label label-danger">Xóa</a>

            <a href="" class="label label-default">Sửa</a>

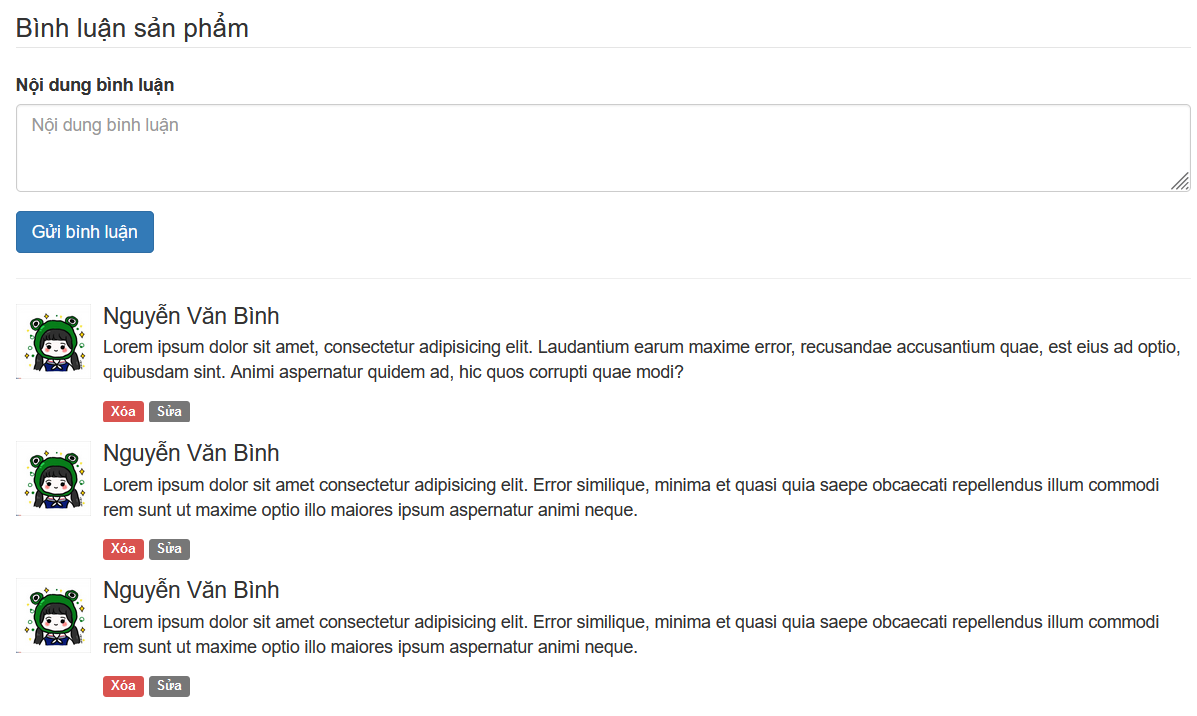
        </div>

    </div>

    @endforeach

</div>

Kết quả ta có danh sách bình luận



**Bước 4:** Thêm Gate kiểm tra tài khoản nào mới được hiển thị nút xóa, sửa comment

Mở file app/providers/AuthServiceProvider, tìm đến hàm boot và định nghĩa Gate như sau

<?php

namespace App\Providers;

use Illuminate\Foundation\Support\Providers\AuthServiceProvider as ServiceProvider;

use Illuminate\Support\Facades\Gate;

use App\Models\Customer;

use App\Models\Comment;

class AuthServiceProvider extends ServiceProvider

{

    /\*\*

     \* The policy mappings for the application.

     \*

     \* @var array<class-string, class-string>

     \*/

    protected $policies = [

        // 'App\Models\Model' => 'App\Policies\ModelPolicy',

    ];

    /\*\*

     \* Register any authentication / authorization services.

     \*

     \* @return void

     \*/

    public function boot()

    {

        $this->registerPolicies();

        Gate::define('change-comment', function (Customer $customer, Comment $comment) {

            return $customer->id === $comment->customer\_id;

        });

    }

}

**Bước 5:** Kiểm tra quyền để hiển thị hoặc ẩn nút bấm Xóa, sửa trên giao diện comment

Mở lại file views/product-detail.blade.php tìm đến danh sách bình luận

<hr>

<div class="comments">

    @foreach($product->comments as $comment)

    <div class="media">

        <a class="pull-left" href="#">

            <img width="60" class="media-object" src="https://haycafe.vn/wp-content/uploads/2022/03/Avatar-TikTok-anh-dai-dien-TikTok.jpg" alt="Image">

        </a>

        <div class="media-body">

            <h4 class="media-heading">{{$comment->user->name}}</h4>

            <p>{{$comment->content}}</p>

            @if (auth('customer')->check() && auth('customer')->user()->can('change-comment', $comment))

                <a href="" class="label label-danger">Xóa</a>

                <a href="" class="label label-default">Sửa</a>

            @endif

        </div>

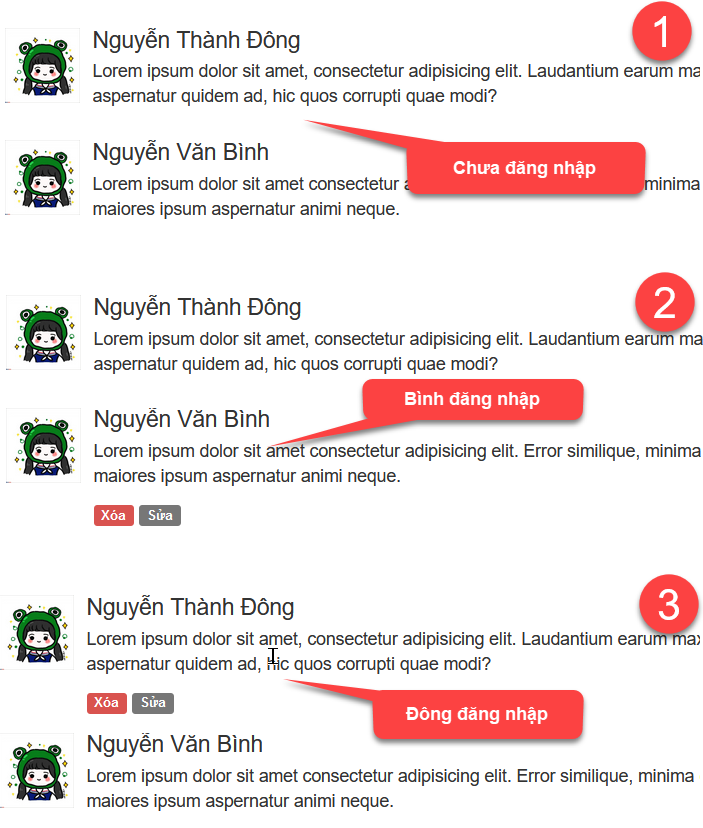
    </div>

    @endforeach

</div>

<hr>

Xem lại kết quả, các user sẽ chỉ có quyền Sửa, xóa bình luận của chính mình đăng



**Bước 6:** Tạo route xóa bình luận và kiểm tra quyền xóa

Mở routes/web.php

<?php

use Illuminate\Support\Facades\Route;

use App\Http\Controllers\HomeController;

/\*

|--------------------------------------------------------------------------

| Web Routes

|--------------------------------------------------------------------------

|

| Here is where you can register web routes for your application. These

| routes are loaded by the RouteServiceProvider within a group which

| contains the "web" middleware group. Now create something great!

|

\*/

Route::get('', [HomeController::class, 'index'])->name('home.index');

Route::get('product/{product}', [HomeController::class, 'product'])->name('home.product\_detail');

Route::post('product-comment/{product}', [HomeController::class, 'productComment'])->name('home.product\_comment');

Route::get('delete-comment/{comment}', [HomeController::class, 'deleteComment'])->name('home.deleteComment');

// route sẽ hiển thị form login

Route::get('login', [HomeController::class, 'login'])->name('home.login');

// route này validate dữ liệu khi submit form

Route::post('login', [HomeController::class, 'check\_login']);

**Bước 7:** Mở HomeController và tạo, hoặc tìm đến phương thức xóa sửa bình luận thêm code như sau

public function deleteComment(Comment $comment)

{

    if (auth('customer')->check() && auth('customer')->user()->can('change-comment', $comment)) {

        // thực hiện xóa

        $comment->delete();

        return redirect()->back()->with('ok','Xóa bình luận thành công');

    }

    return abort(403); // trả về 403 không có quyền

}

Đặt link cho nút xóa

@if (auth('customer')->check() && auth('customer')->user()->can('change-comment', $comment))

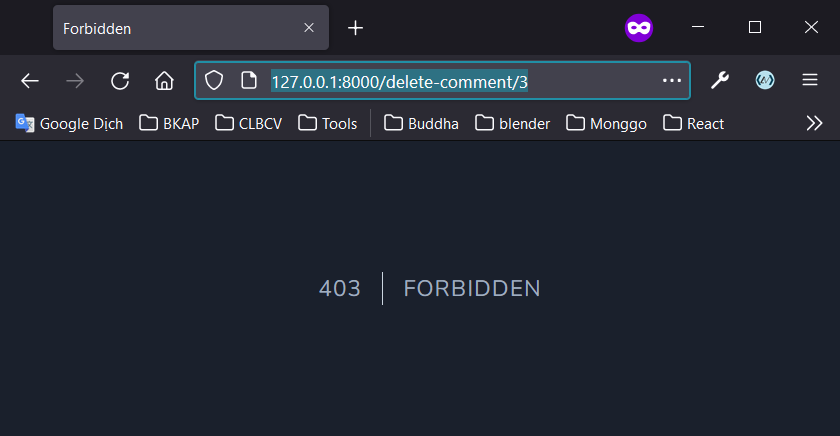
    <a href="{{ route('home.deleteComment', $comment->id) }}" class="label label-danger">Xóa</a>

    <a href="" class="label label-default">Sửa</a>

@endif

Khi cố tình truy cập vào link xóa comment mà chưa đăng nhập hoặc không thuộc quyền của user đó

[http://127.0.0. 1:8000/delete-comment/3](http://127.0.0.1:8000/delete-comment/3)



# Bài 4: Tạo policy, kết hợp Gate và policy

**Bước 1:** Tạo CommentPolicy với lệnh sau

php artisan make:policy CommentPolicy

au khi tạo, file CommentPolicy nằm trong thư mục app/Policies

**Bước 2:** Khai báo phương thức xử lý logic kiemr tra quyền truy cập comment nhứ au

Mở file: **app/Policies/CommentPolicy.php**

<?php

namespace App\Policies;

use App\Models\Customer;

use App\Models\Comment;

use Illuminate\Auth\Access\HandlesAuthorization;

class CommentPolicy

{

    use HandlesAuthorization;

    /\*\*

     \* Create a new policy instance.

     \*

     \* @return void

     \*/

    public function changeComment(Customer $customer, Comment $comment)

    {

        return $customer->id === $comment->customer\_id;

    }

}

**Bước 3:** Mở lại file app/Providers/ AuthServiceProvider

Đăng ký policy và áp dụng phương thưc trong CommentPolicy như sau

<?php

namespace App\Providers;

use Illuminate\Foundation\Support\Providers\AuthServiceProvider as ServiceProvider;

use Illuminate\Support\Facades\Gate;

use App\Models\Customer;

use App\Models\Comment;

class AuthServiceProvider extends ServiceProvider

{

    /\*\*

     \* The policy mappings for the application.

     \*

     \* @var array<class-string, class-string>

     \*/

    protected $policies = [

        'App\Models\Customer' => 'App\Policies\CustomerPolicy',

    ];

    /\*\*

     \* Register any authentication / authorization services.

     \*

     \* @return void

     \*/

    public function boot()

    {

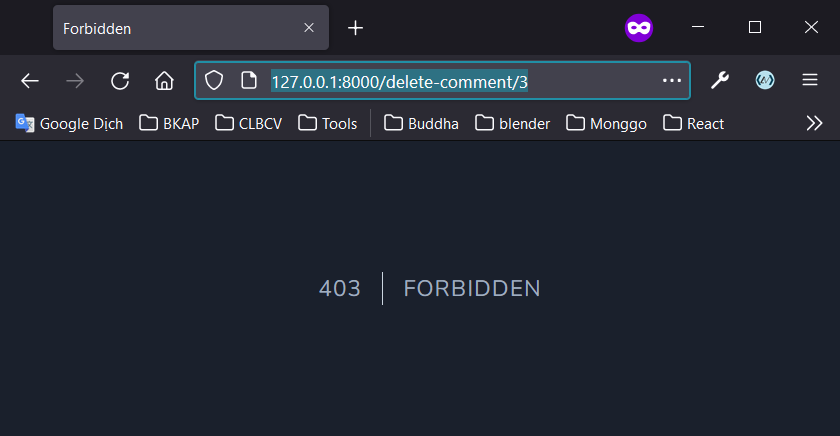
        $this->registerPolicies();

        Gate::define('change-comment', [CustomerPolicy::class,'changeComment']);

    }

}

Test lại kết quả khi truy cập route xóa, sửa comment mà không đăng nhập hoặc không có quyền thì báo lỗi



# Bài tập làm thêm

Sinh viên dựa vào bảng users tài khoản đăng nhập admin

Hãy thêm cột role, Các tài khoản khác nhau sẽ có các **role** như

Admin: toàn quyền

Product: chỉ có quyền quản lý sản phẩm

… các quyền khác làm tương tự

Nếu tài khoản đăng nhập vào mục mà không có quyền thì sẽ báo lỗi 403