

HOLADOC

WEB PROGRAMMING

Kelompok 3:

2501966692 - Ricky Zaini

2501967921 - Farrell Hartono

2501966881 - Lorenzo Donovan

2501968924 - Alvian Daniel Susanto

2540131046 - Wen Sen Tan

2540131235 - Evan Jeremy Wijaya

Class Program: Computer Science - LG01

Lecturer:

Rani Puspita, S.Kom., M.Kom

UNIVERSITAS BINA NUSANTARA

COMPUTER SCIENCE

2023, ODD SEMESTER

DAFTAR ISI

DAFTAR ISI	2
Bab 1. Pendahuluan	3
1.1. Latar Belakang	3
1.2. Tujuan	3
1.3. Manfaat	3
1.4. Batasan	4
1.5. Tools yang Digunakan	4
Bab 2. Code	5
2.1. Controller	5
2.2. Views	9
2.3. Model	12
2.4. Route	16
2.5. Middleware	16
Bab 3. User Interface	18
3.1 Home Page	18
3.2 Category Page	20
3.3 Login/Register Page	20
3.4 Product Detail Page	22
3.5. Product Category Detail Page	23
3.6 Shopping Cart Page	24
3.7 Payment Method Page	25
3.8 Purchase Page	26
Bab 4. Kesimpulan & Saran	27
4.1. Kesimpulan	27
4.2. Saran	27
Daftar Pustaka	28

BAB 1. PENDAHULUAN

1.1. LATAR BELAKANG

Dalam era digital yang terus berkembang, kebutuhan akan pelayanan kesehatan yang efisien dan praktis semakin meningkat. Salah satu aspek penting dari pelayanan kesehatan adalah akses yang mudah dan cepat terhadap obat-obatan yang diperlukan. HolaDoc hadir sebagai solusi inovatif untuk memenuhi kebutuhan ini dengan memberikan platform aplikasi dalam bentuk website yang menyediakan layanan apotek secara online. Tujuan utamanya adalah memudahkan pelanggan untuk memesan obat dimana saja tanpa perlu mengantre secara langsung. HolaDoc juga bekerja sama secara langsung dengan produsen obat dan apotek-apotek untuk memastikan keamanan dan keaslian produk yang dijual.

1.2. TUJUAN

Tujuan dari website HolaDoc :

- Memudahkan Pembelian Obat: Memberikan pengguna kemudahan dalam memesan obat-obatan secara online dari apotik terdekat.
- Informasi Obat yang Lengkap: Menyediakan informasi lengkap mengenai obat-obatan, seperti deskripsi, dosis, efek samping, dan lainnya.
- Meningkatkan Efisiensi: Menghemat waktu dan usaha pengguna dengan layanan online yang mudah digunakan.

1.3. MANFAAT

Manfaat dari website HolaDoc :

- Memudahkan Pembelian Obat: Pengguna dapat membeli obat dengan lebih mudah tanpa harus mengunjungi apotek secara langsung.
- Informasi yang Lengkap: Pengguna dapat mengakses informasi lengkap tentang obat-obatan sebelum membeli.
- Menghemat Waktu: Pengguna dapat menghemat waktu dengan menghindari antrian di apotek.

1.4. BATASAN

Batasan-batasan yang ada agar website HolaDoc dapat berjalan:

- Hukum dan Regulasi: Mematuhi hukum dan regulasi yang berlaku terkait penjualan obat-obatan secara online, seperti izin resmi, pengaturan privasi, dan aturan FDA (Food and Drug Administration).
- Perlindungan Data Pengguna: Meningkatkan keamanan untuk melindungi informasi pribadi pengguna, termasuk informasi medis dan transaksi keuangan.
- Ketersediaan Stok Obat: Ketersediaan obat-obatan harus diperbarui secara teratur agar pengguna mendapatkan informasi yang akurat.

1.5. TOOLS YANG DIGUNAKAN

Tools yang digunakan untuk membuat website HolaDoc:

- Laravel

Laravel menjadi framework untuk membuat website HolaDoc. Laravel juga digunakan untuk membangun backend dari website Apotik Online, memungkinkan manajemen data dan autentikasi pengguna.

- Visual Studio Code

Visual Studio Code menjadi tempat untuk membuat dan mengembangkan kode untuk frontend (Bootstrap) dan backend (PHP) di website HolaDoc.

- MySQL

MySQL digunakan untuk menyimpan dan mengelola data HolaDoc seperti, informasi obat, stok obat, dan informasi pengguna, melalui XAMPP untuk membuat server lokal dan tempat pengelolaan MySQL.

- Bootstrap

Bootstrap digunakan sebagai pengembang frontend dari website HolaDoc agar website menjadi responsif dan menarik secara visual.

- Figma

Figma digunakan untuk membuat desain awal dari website HolaDoc dan menjadi referensi kami dalam membuat website HolaDoc secara desain.

- GitHub

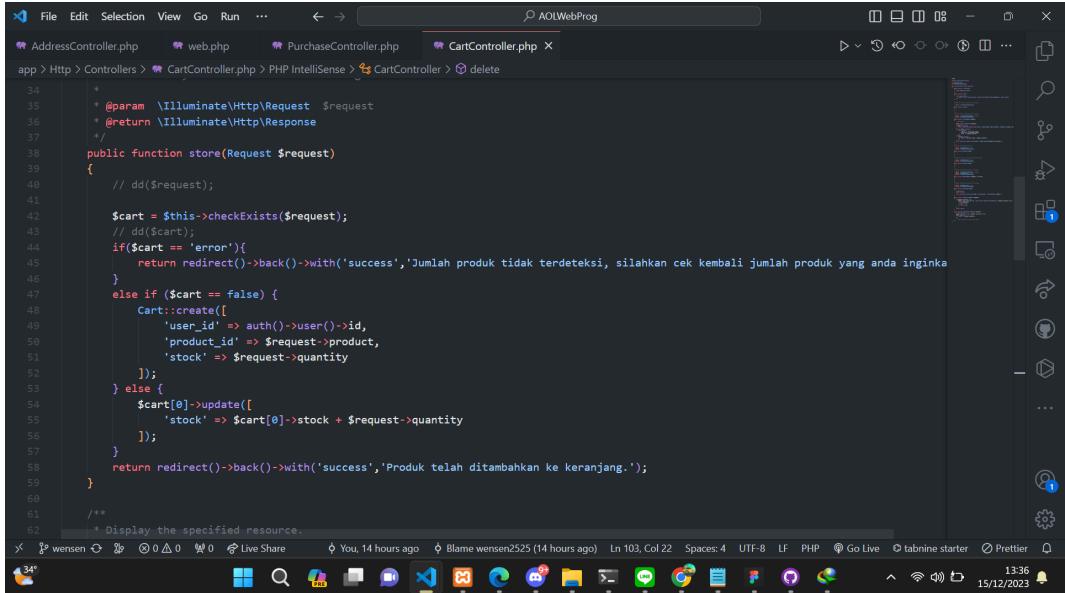
GitHub digunakan untuk membantu kami bekerjasama secara tim dalam pembuatan website HolaDoc.

BAB 2. CODE

2.1. Controller

CartController, AddressController, PurchaseController, ProductController, Auth(LoginController, RegisterController), CategoryController.

CartController (Bertugas untuk menangani bagian keranjang pelanggan, ada store, index, updateQuantity, delete)



```

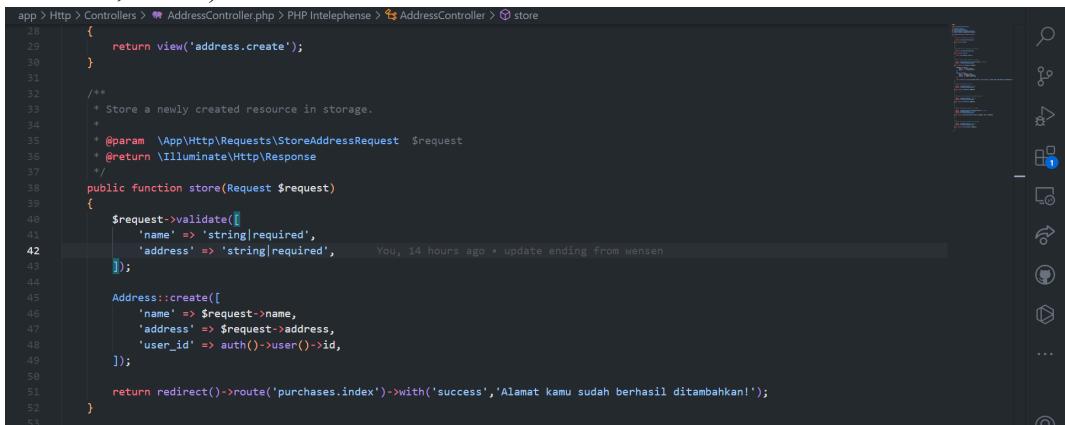
File Edit Selection View Go Run ... < > AOLWebProg
AddressController.php web.php PurchaseController.php CartController.php X
app > Http > Controllers > CartController.php > PHP IntelliSense > CartController > delete

34     *
35     * @param \Illuminate\Http\Request $request
36     * @return \Illuminate\Http\Response
37     */
38    public function store(Request $request)
39    {
40        // dd($request);
41
42        $cart = $this->checkExists($request);
43        // dd($cart);
44        if($cart == 'error'){
45            return redirect()->back()->with('success','Jumlah produk tidak terdeteksi, silahkan cek kembali jumlah produk yang anda inginkan');
46        }
47        else if ($cart == false) {
48            Cart::create([
49                'user_id' => auth()->user()->id,
50                'product_id' => $request->product,
51                'stock' => $request->quantity
52            ]);
53        } else {
54            $cart[0]->update([
55                'stock' => $cart[0]->stock + $request->quantity
56            ]);
57        }
58        return redirect()->back()->with('success','Produk telah ditambahkan ke keranjang.');
59    }
60
61 /**
62 * Display the specified resource.
63 */

```

You, 14 hours ago Blame wensen2525 (14 hours ago) Ln 103, Col 22 Spaces: 4 UTF-8 LF PHP Go Live tabnine starter Prettier 13:36 15/12/2023

AddressController (Bertugas untuk menangani bagian alamat pelanggan ada store, create)



```

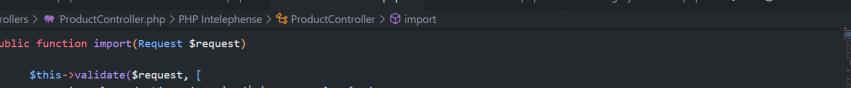
app > Http > Controllers > AddressController.php > PHP Intelephense > AddressController > store
28    {
29        return view('address.create');
30    }
31
32 /**
33 * Store a newly created resource in storage.
34 *
35 * @param \App\Http\Requests\StoreAddressRequest $request
36 * @return \Illuminate\Http\Response
37 */
38    public function store(Request $request)
39    {
40        $request->validate([
41            'name' => 'string|required',
42            'address' => 'string|required',
43        ]);
44
45        Address::create([
46            'name' => $request->name,
47            'address' => $request->address,
48            'user_id' => auth()->user()->id,
49        ]);
50
51        return redirect()->route('purchases.index')->with('success','Alamat kamu sudah berhasil ditambahkan!');
52    }
53

```

PurchaseController (Bertugas untuk menangani pembelian produk ada store, update, index, validateStock, checkOut)

```
47
48     $purchase = DB::transaction(function () use ($request) {
49
50         // SET PAYMENT DUE
51         $payment_due = Carbon::now()->addHours(5);
52
53         $purchase = Purchase::create([
54             'user_id' => auth()->user()->id,
55             'payment_due' => $payment_due,
56         ]);
57         $totalPricePurchase = 0;
58         $count_cart = count($request->input());
59         if (isset($request->product)) {
60             $product = Product::find($request->product);
61
62             $product->update([
63                 'stock' => $product->stock - $request->quantity,
64                 'stock_sold' => $product->stock_sold + $request->quantity,
65             ]);
66
67             // VALIDATE Stock
68             if ($this->validateStock($product, $request->quantity) == false)
69                 return redirect()->route('dashboard')->with('failed', 'This product is sold out!');
70
71             $purchaseDetail = PurchaseDetail::create([
72                 'purchase_id' => $purchase->id,
73                 'product_id' => $product->id,
74                 'price' => $product->price,
75                 'quantity' => $request->quantity
76             ]);
77         }
78     });
79
80     $purchase->update(['status' => 'PENDING']);
81
82     return redirect()->route('dashboard')->with('success', 'Your purchase has been submitted');
83 }
```

ProductController(Bertugas untuk menangani produk seperti import data dan menampilkan produkDetail per produknya)



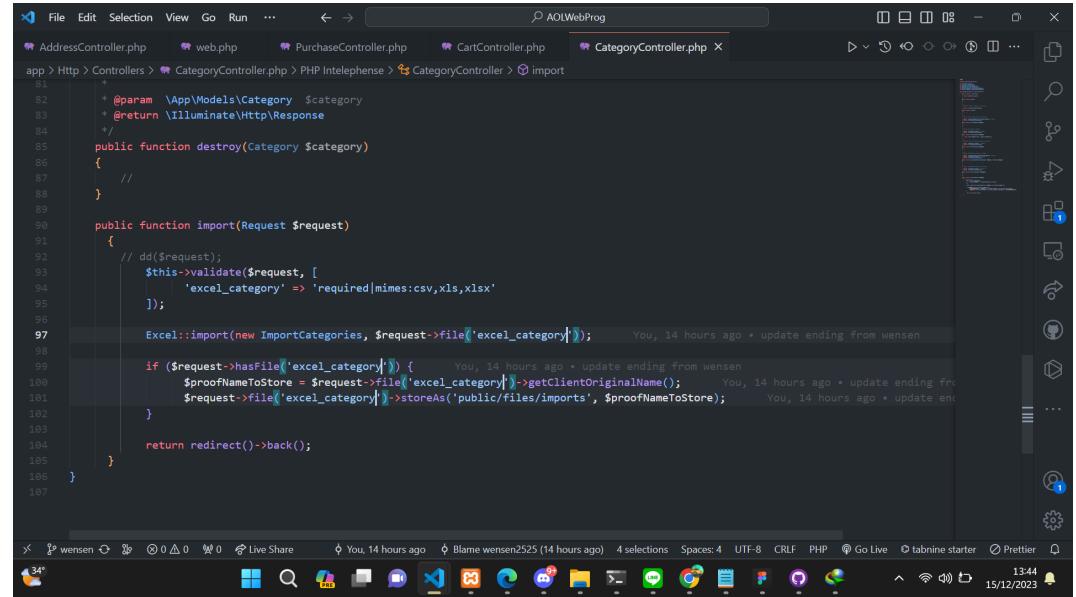
The screenshot shows a code editor with the file `ProductController.php` open. The code handles file imports, specifically CSV, XLS, and XLSX files. It includes validation for the file type and name, and stores the file in a public directory. A note from a user named 'wensen' is visible in the code area.

```
class ProductController extends Controller
{
    public function import(Request $request)
    {
        $this->validate($request, [
            'excel_product' => 'required|mimes:csv,xls,xlsx'
        ]);

        Excel::import(new ImportProducts, $request->file('excel_product'));
        $proofNameToStore = $request->file('excel_product')->getClientOriginalName();
        $request->file('excel_product')->storeAs('public/files/imports', $proofNameToStore);
    }

    return redirect()->back();
}
```

CategoryController (Bertugas untuk menampilkan halaman kategori, lalu produk-produk per kategori, lalu ada function import date kategori)

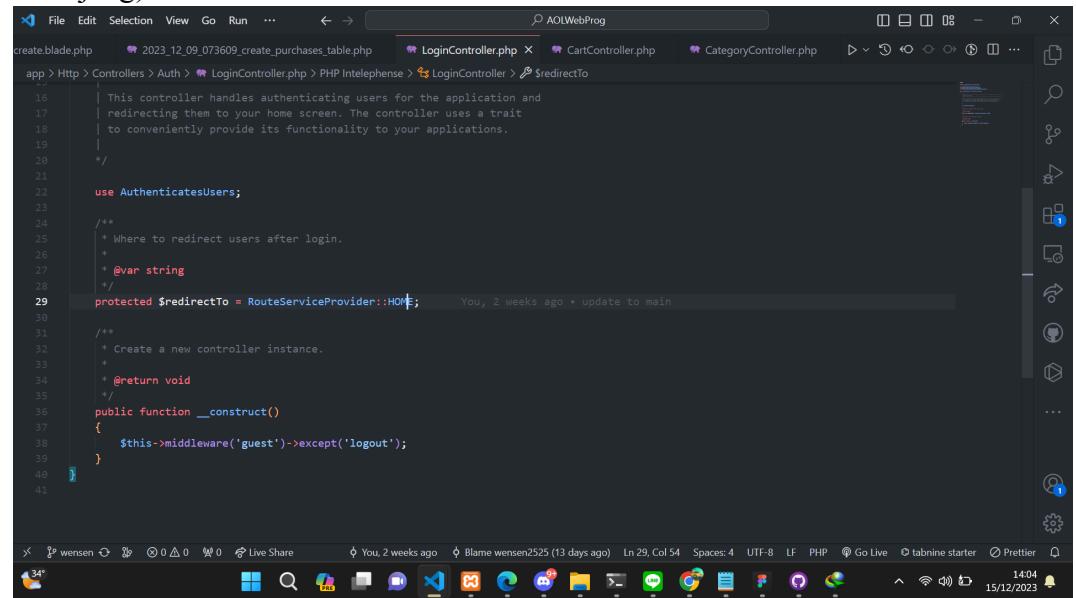


```

File Edit Selection View Go Run ... < > AOLWebProg
AddressController.php PurchaseController.php CartController.php CategoryController.php X
app > Http > Controllers > CategoryController.php > PHP Intelephense > CategoryController > import
1.  +
2.  + @param \App\Models\Category $category
3.  + @return Illuminate\Http\Response
4.  +
5.  public function destroy(Category $category)
6.  {
7.  }
8.  +
9.  public function import(Request $request)
10. {
11.     // dd($request);
12.     $this->validate($request, [
13.         'excel_category' => 'required|mimes:csv,xls,xlsx'
14.     ]);
15.
16.     Excel::import(new ImportCategories, $request->file('excel_category'));
17.     if ($request->hasFile('excel_category')) {
18.         $proofNameToStore = $request->file('excel_category')>getClientOriginalName();
19.         $request->file('excel_category')>storeAs('public/files-imports', $proofNameToStore);
20.     }
21.
22.     return redirect()->back();
23. }
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.
41.

```

LoginController (Bertugas untuk menangani login pelanggan, login ini dipakai saat mau membeli produk, atau menambahkan produk ke keranjang)

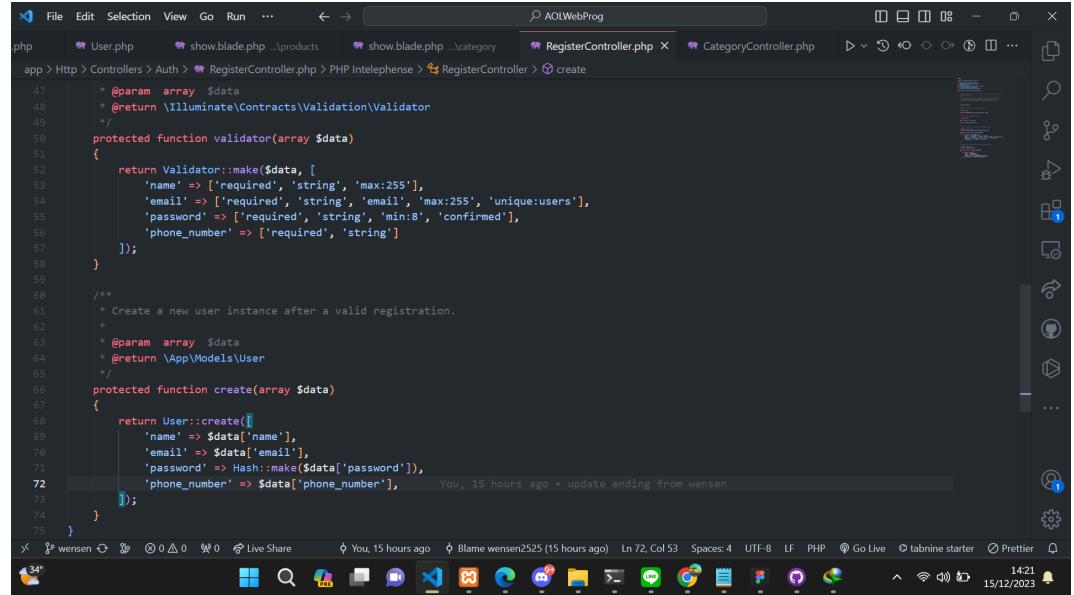


```

File Edit Selection View Go Run ... < > AOLWebProg
create.blade.php 2023_12_09_073609_create_purchases_table.php LoginController.php X CartController.php CategoryController.php
app > Http > Controllers > Auth > LoginController.php > PHP Intelephense > LoginController > $redirectTo
16. | This controller handles authenticating users for the application and
17. | redirecting them to your home screen. The controller uses a trait
18. | to conveniently provide its functionality to your applications.
19.
20.
21.
22. use AuthenticatesUsers;
23.
24. /**
25. * Where to redirect users after login.
26. *
27. * @var string
28. */
29. protected $redirectTo = RouteServiceProvider::HOME;
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.
40.
41.

```

RegisterController (Bertugas untuk menangani pendaftaran pelanggan)



```

File Edit Selection View Go Run ... < > AOLWebProg
php User.php show.blade.php ...products show.blade.php ...category RegisterController.php CategoryController.php D v O O P ...
app > Http > Controllers > Auth > RegisterController.php > PHP Intelephense > RegisterController > create
47 * @param array $data
48 * @return \Illuminate\Contracts\Validation\Validator
49 */
50 protected function validator(array $data)
51 {
52     return Validator::make($data, [
53         'name' => ['required', 'string', 'max:255'],
54         'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
55         'password' => ['required', 'string', 'min:8', 'confirmed'],
56         'phone_number' => ['required', 'string']
57     ]);
58 }
59 /**
60 * Create a new user instance after a valid registration.
61 *
62 * @param array $data
63 * @return \App\Models\User
64 */
65 protected function create(array $data)
66 {
67     return User::create([
68         'name' => $data['name'],
69         'email' => $data['email'],
70         'password' => Hash::make($data['password']),
71         'phone_number' => $data['phone_number'],
72     ]);
73 }
74 }
75 }

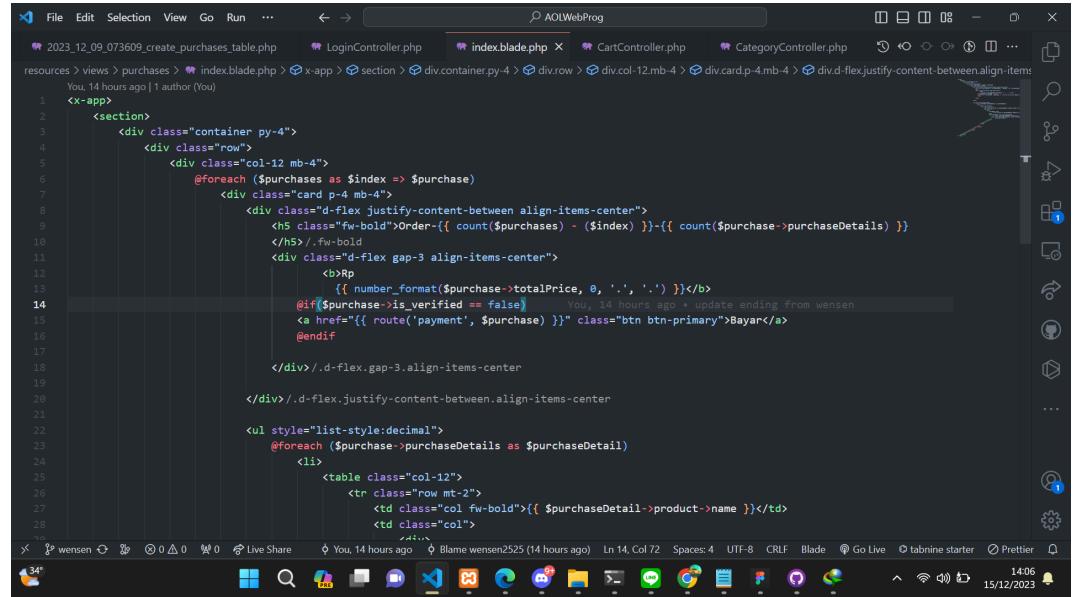
```

The screenshot shows a code editor window with the file `RegisterController.php` open. The code defines a controller for user registration. It includes validation rules for name, email, password, and phone number, and a method to create a new user instance using the `User::create()` method.

2.2. Views

Purchase, Checkout, Cart, Home, Kategori, Produk Detail, Produk-Produk per Kategori

Purchase (Bertugas untuk menampilkan daftar obat yang pernah di check out yang mencakup obat-obat yang sudah dibayar maupun belum dibayar)

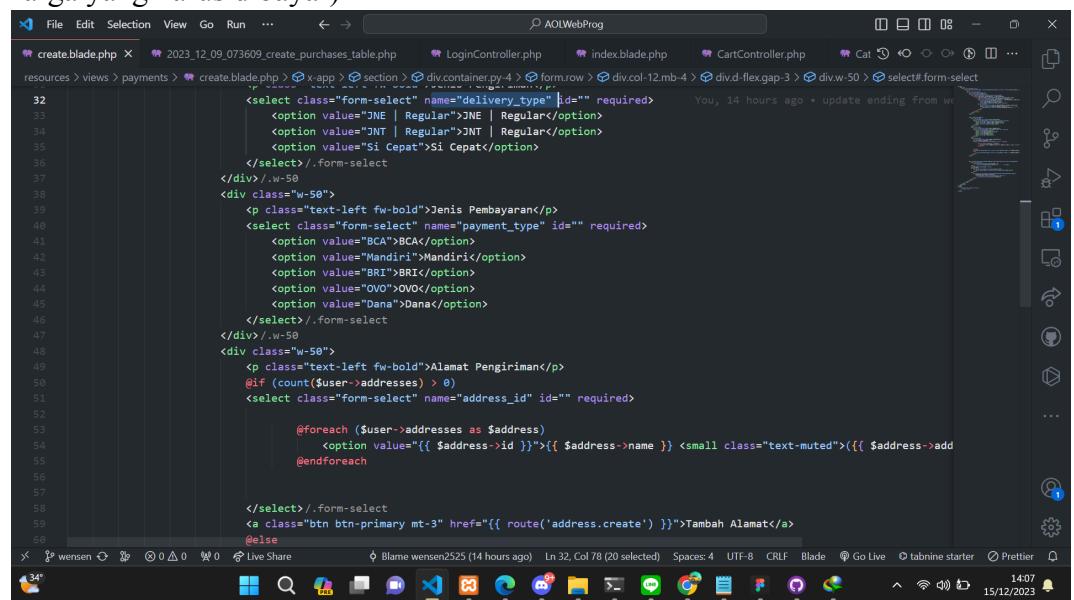


```

<x-app>
    <div class="container py-4">
        <div class="row">
            <div class="col-12 mb-4">
                @foreach ($purchases as $index => $purchase)
                    <div class="card p-4 mb-4">
                        <div class="d-flex justify-content-between align-items-center">
                            <h5 class="fw-bold">Order-{{ count($purchases) - ($index) }}-{{ count($purchase->purchaseDetails) }}</h5>
                            <h5 class="fw-bold">Total Price {{ number_format($purchase->totalPrice, 0, ',', '.') }}</h5>
                            @if($purchase->is_verified == false)
                                <a href="{{ route('payment', $purchase ) }}" class="btn btn-primary">Bayar</a>
                            @endif
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>

```

Checkout(Bertugas untuk menampilkan daftar obat yang ingin dibeli, jenis pengiriman, jenis pembayaran, alamat pengiriman, beserta dengan total harga yang harus dibayar)



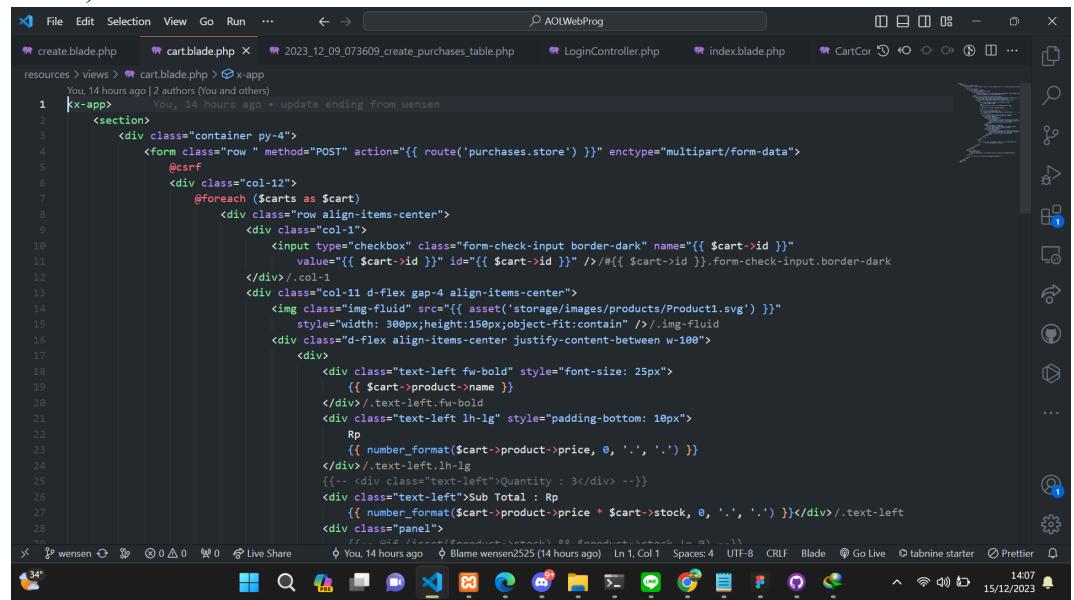
```

<div class="w-50">
    <p class="text-left fw-bold">Jenis Pembayaran</p>
    <select class="form-select" name="payment_type" id="" required>
        <option value="BCA">BCA</option>
        <option value="Mandiri">Mandiri</option>
        <option value="BRI">BRI</option>
        <option value="OVO">OVO</option>
        <option value="Dana">Dana</option>
    </select>
</div>
<div class="w-50">
    <p class="text-left fw-bold">Alamat Pengiriman</p>
    @if (count($user->addresses) > 0)
        <select class="form-select" name="address_id" id="" required>
            @foreach ($user->addresses as $address)
                <option value="{{ $address->id }}>{{ $address->name }}</option>
            @endforeach
        </select>
    @else
        <a class="btn btn-primary mt-3" href="{{ route('address.create') }}>Tambah Alamat</a>
    @endif

```

Cart(Bertugas untuk menampilkan obat-obatan yang ingin dibeli pada satu waktu)

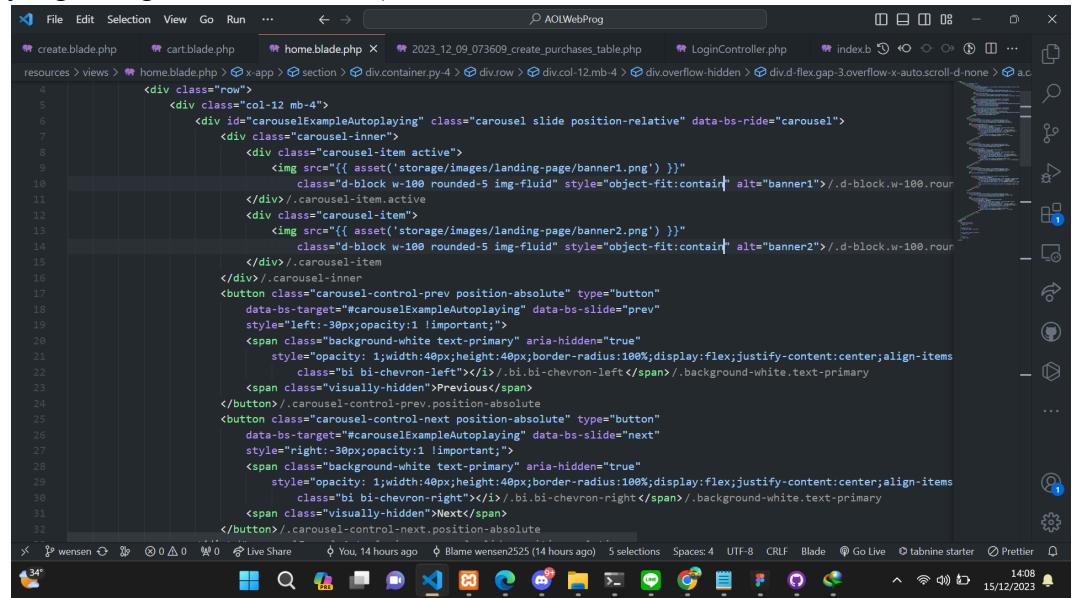
People
Innovation
Excellence



```

resources > views > cart.blade.php > x-app
You, 14 hours ago | 2 authors (You and others)
1 <x-app>
2 <section>
3   <div class="container py-4">
4     <form class="row" method="POST" action="{{ route('purchases.store') }}" enctype="multipart/form-data">
5       @csrf
6       <div class="col-12">
7         @foreach ($scarts as $cart)
8           <div class="row align-items-center">
9             <div class="col-1">
10               <input type="checkbox" class="form-check-input border-dark" name="{{ $cart->id }}"
11                 value="{{ $cart->id }}" id="{{ $cart->id }}"/>#{{ $cart->id }}.form-check-input.border-dark
12             </div>/.col-1
13             <div class="col-11 d-flex gap-4 align-items-center">
14               .img-fluid
15               <div class="d-flex align-items-center justify-content-between w-100">
16                 <div>
17                   <div class="text-left fw-bold" style="font-size: 25px">
18                     {{ $cart->product->name }}</div>/.text-left.fw-bold
19                   <div class="text-left lh-lg" style="padding-bottom: 10px">
20                     Rp
21                     {{ number_format($cart->product->price, 0, ',', '.') }}</div>/.text-left.lh-lg
22                   {{-- <div class="text-left">Quantity : 3</div> --}}
23                   <div class="text-left">Sub Total : Rp
24                     {{ number_format($cart->product->price * $cart->stock, 0, ',', '.') }}</div>/.text-left
25                   <div class="panel">
26                     {{-- <div class="panel-body">Rp {{ number_format($cart->product->price * $cart->stock, 0, ',', '.') }}</div> --}}
27                   {{-- <div class="panel-footer">--}}
28                 </div>/.panel
29             </div>/.d-flex.align-items-center.justify-content-between.w-100
30           </div>/.row.align-items-center
31         @endforeach
32       </div>/.col-12
33     </form>
34   </section>
35 
```

Home(Bertugas untuk menampilkan beranda yang berisi banner, obat yang terlaris pada tahun tersebut, promo, kategori obat terlaris, dan obat-obatan yang sering dibeli sehari-hari)



```

resources > views > home.blade.php > x-app > section > div.container.py-4 > div.row > div.col-12.mb-4 > div.overflow-hidden > div.d-flex.gap-3.overflow-x-auto.scroll-d-none > a.c
1 <div class="row">
2   <div class="col-12 mb-4">
3     <div id="carouselExampleAutoplaying" class="carousel slide position-relative" data-bs-ride="carousel">
4       <div class="carousel-inner">
5         <div class="carousel-item active">
6           /.d-block.w-100.rour
7         </div>/.carousel-item.active
8         <div class="carousel-item">
9           /.d-block.w-100.rour
10        </div>/.carousel-item
11      </div>/.carousel-inner
12      <button class="carousel-control-prev position-absolute" type="button"
13        data-bs-target="#carouselExampleAutoplaying" data-bs-slide="prev"
14        style="left: -30px; opacity: 1 !important;">
15        <span class="background-white text-primary" aria-hidden="true"
16          style="width: 40px; height: 40px; border-radius: 100%; display: flex; justify-content: center; align-items: center;>
17            <i class="bi bi-chevron-left">/</i>/.bi.bi-chevron-left</span>/.background-white.text-primary
18        <span class="visually-hidden">Previous</span>
19      </button>/.carousel-control-prev.position-absolute
20      <button class="carousel-control-next position-absolute" type="button"
21        data-bs-target="#carouselExampleAutoplaying" data-bs-slide="next"
22        style="right: -30px; opacity: 1 !important;">
23        <span class="background-white text-primary" aria-hidden="true"
24          style="width: 40px; height: 40px; border-radius: 100%; display: flex; justify-content: center; align-items: center;>
25            <i class="bi bi-chevron-right">/</i>/.bi.bi-chevron-right</span>/.background-white.text-primary
26        <span class="visually-hidden">Next</span>
27      </button>/.carousel-control-next.position-absolute
28    </div>
29 
```



People
Innovation
Excellence

Kategori(Berguna untuk menampilkan kategori-kategori obat yang ada)

You, 14 hours ago | author (You)

```
1 <x-app>
2     <section>
3         <div class="container py-4">
4             <h3 class="fw-bold text-center">KATEGORI</h3>
5             <div class="row">
6                 @foreach ($categories as $category)
7                     <a class="col-4 p-3 scale text-decoration-none" href="{{ route('category.show', $category) }}>
8                         <div class="bg-outline-primary d-flex gap-5 justify-content-center align-items-center p-4 rounded-4">
9                             name }}"
12                                <div class="fs-4 fw-bold">
13                                    {{ $category->name }}
14                                </div>/.fs-4 fw-bold
15                            </div>/.bg-outline-primary.d-flex.gap-5.justify-content-center.align-items-center.p-4.rounded-4
16                        </a>/.col-4.p-3.scale.text-decoration-none
17                    @endforeach
18                </div>/.row
19            </div>/.container.py-4
20        </section>
21    </x-app>
22    <style>
23        You, 14 hours ago | author (You)
24        .scale:hover {
25            You, 14 hours ago + update ending from wensen
26            scale: 1.02;
27        }
28        You, 14 hours ago | author (You)
29        .bg-outline-primary{
30            You, 14 hours ago + update ending from wensen
31            border: 1px solid #007bff;
32            border-radius: 10px;
33            padding: 10px;
34            background-color: #fff;
35            color: #007bff;
36            font-weight: bold;
37            text-decoration: none;
38            transition: all 0.3s ease;
39        }
40        .bg-outline-primary:hover {
41            You, 14 hours ago + update ending from wensen
42            border: 1px solid #007bff;
43            border-radius: 10px;
44            padding: 10px;
45            background-color: #fff;
46            color: #007bff;
47            font-weight: bold;
48            text-decoration: none;
49            transition: all 0.3s ease;
50        }
51    </style>
52
```

ProdukDetail(Berguna untuk menampilkan halaman informasi-informasi terkait dengan produk obat tertentu seperti deskripsi, komposisi, cara pemakaian, dosis, efek samping. Selain itu halaman ini juga menampilkan produk lainnya yang mungkin akan cocok dengan pengguna)

```
File Edit Selection View Go Run ... ← → AOLWebProg  
resources > views > products > show.blade.php > x-app > section > div.container.py-4 > div.row > div.col-12.mb-4 > p  
62 </form> / d-flex gap-4 align-items-center justify-content-between  
63 </div> / col-9  
64 </div> / row  
65 </div> / col-12 mb-4  
66 <div class="col-12 mb-4">  
67 <p class="fw-bold">Deskripsi</p>  
68 <p>{{ $product->description }}</p>  
69 </div> / col-12 mb-4  
70 <div class="col-12 mb-4">  
71 <p class="fw-bold">Komposisi</p>  
72 <p>{{ $product->ingredients }}</p>  
73 </div> / .col-12 mb-4  
74 <div class="col-12 mb-4">  
75 <p class="fw-bold">Cara pemakaian</p>  
76 <p>{{ $product->howtouse }}</p>  
77 </div> / .col-12 mb-4  
78 <div class="col-12 mb-4">  
79 <p class="fw-bold">Dosis</p>  
80 <p>{{ $product->dose }}</p> You, 14 hours ago * update ending from wensen  
81 </div> / .col-12 mb-4  
82 <div class="col-12 mb-4">  
83 <p class="fw-bold">Efek Samping</p>  
84 <p>{{ $product->sideeffects }}</p>  
85 </div> / .col-12 mb-4  
86 <div class="col-12 mb-4">  
87 <!-- List Product Title -->  
88 @if(count($other_products) > 1)  
89 <div class="fs-4 fw-bold">Produk lainnya yang mungkin cocok kamu</div>  
90 @endif  
x wensen ↗ Live Share ↗ You, 14 hours ago ↗ Blame wensen2525 (14 hours ago) Ln 80, Col 41 Spaces: 4 UTF-8 CRLF Blade ⚡ Go Live tabnine starter ⚡ Prettier
```

Produk-Produk per Kategori (Berguna untuk menampilkan produk-produk yang sesuai dengan kategori)

People
Innovation
Excellence

```
File Edit Selection View Go Run ... < > AOLWebProg
CartController.php Cart.php User.php show.blade.php ...products show.blade.php ...category CategoryController.php
resources > views > category > show.blade.php > x-app > section > div.container.py-4 > div.row > div.col-12.mb-4 > div.overflow-hidden.py-3 > div.d-flex.flex-wrap.gap-3 overflow-x-auto
24         </div> /.row.align-items-center
25     </div> /.col-12.mb-4
26     <!-- End Header (Category Description) -->
27     <div class="col-12 mb-4">
28         <!-- List Product Title -->
29         <div class="fs-4 fw-bold">Product</div>
30         <!-- End List Product Title -->
31
32         <!-- List Item -->
33         <div class="overflow-hidden py-3">
34             <div class="d-flex flex-wrap gap-3 overflow-x-auto scroll-d-none">
35                 @foreach ($category->products as $other_product)
36                     <a class="card border-0 text-decoration-none position-relative" href="{{ route('products.show', $other_product) }}" style="min-width: 16rem;max-width:16rem"
37                         name }}"
38                         class="card-img-top rounded-3 img-fluid" style="height: 170px; width:100%;object-fit:cover"/>.card-img-top.rounded-3.img-fluid
39                     <div class="mt-2">
40                         <p class="fs-5 m-0 fw-bold">{{ $other_product->name }}</p>
41                         <p>Rp {{ number_format($other_product->price, 0, ',', '.') }}</p>
42                     </div> /.mt-2
43                 </a> /.card.border-0.text-decoration-none.position-relative
44             @endforeach
45         </div> /.d-flex.flex-wrap.gap-3.overflow-x-auto.scroll-d-none
46     </div> /.overflow-hidden.py-3
47     <div class="col-12.mb-4">
48         <div class="row">
49             You, 14 hours ago + update ending from wen
50         </div> /.row
51     </div> /.container.py-4
52 
```

2.3. Model

Address, Cart, Purchase, purchaseDetail, Category, User, Product

Address (address model ini mempunyai relationship dengan user dan purchase)

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** AOLWebProg
- Address Bar:** app > Models > Address.php > PHP Intelephense > Address.php > purchases
- Code Editor:** The main content area displays the `Address.php` file from a Laravel application. The code defines a `Address` model that belongs to the `User` model and has many `Purchase` models.
- Right Sidebar:** A vertical sidebar on the right contains several icons representing different tools and features, such as a file tree, search, and help.
- Bottom Taskbar:** The taskbar includes icons for File, Edit, Selection, View, Go, Run, and various system status indicators like battery level and signal strength.

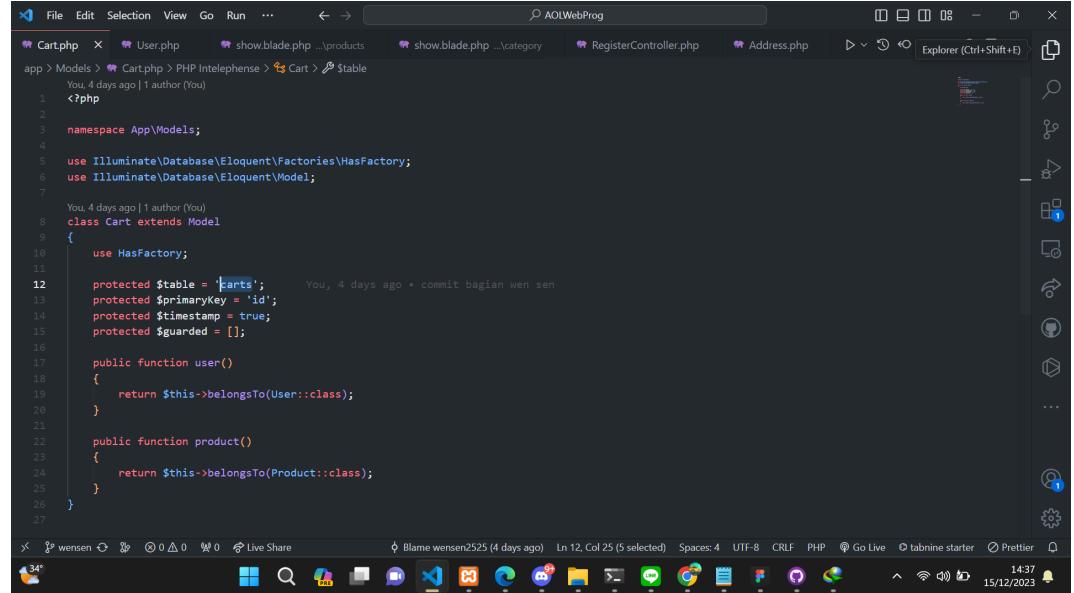
```
File Edit Selection View Go Run ... ← → AOLWebProg
app > Models > Address.php > PHP Intelephense > Address.php > purchases
namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
You, 15 hours ago | 1 author (You)
class Address extends Model
{
    use HasFactory;
    use HasFactory;

    protected $table = 'addresses';
    protected $primaryKey = 'id';
    protected $timestamp = true;
    protected $guarded = [];

    public function user()
    {
        return $this->belongsTo(User::class);
    }

    public function purchases()
    {
        return $this->hasMany(Purchase::class); You, 15 hours ago * update ending from wensen
    }
}
34% 14:36 15/12/2023
```

Cart (cart model ini mempunyai relationship dengan user dan product)



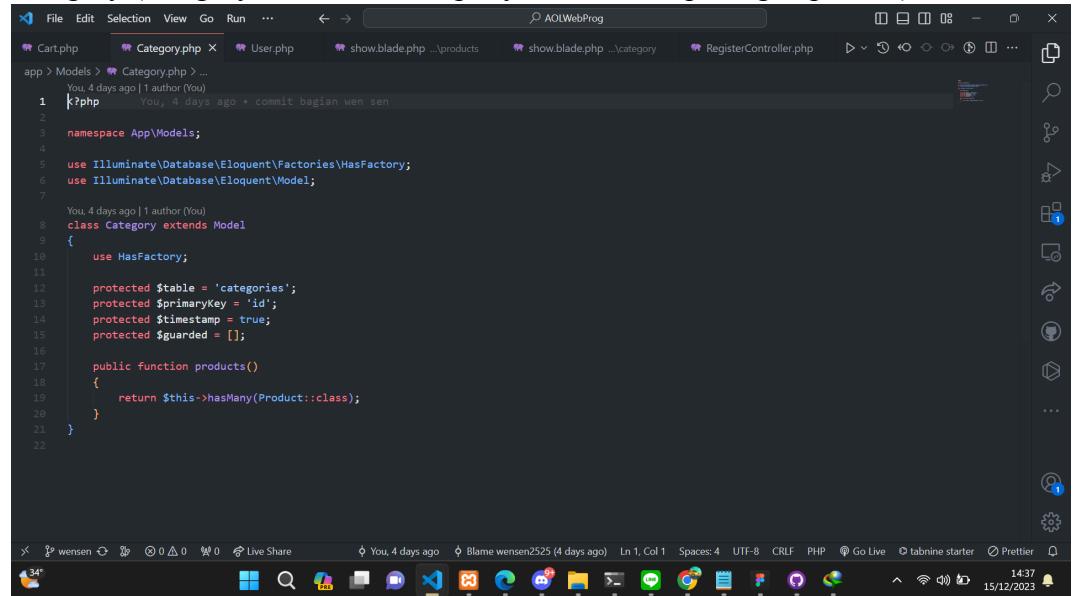
```

File Edit Selection View Go Run ... < > AOLWebProg
Cart.php User.php show.blade.php ...products show.blade.php ...category RegisterController.php Address.php Explorer (Ctrl+Shift+E)
app > Models > Cart.php > PHP Intelphense > Cart > $table
You, 4 days ago | 1 author (You)
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 You, 4 days ago | 1 author (You)
9 class Cart extends Model
{
10     use HasFactory;
11
12     protected $table = 'carts'; You, 4 days ago + commit bagian wen sen
13     protected $primaryKey = 'id';
14     protected $timestamp = true;
15     protected $guarded = [];
16
17     public function user()
18     {
19         return $this->belongsTo(User::class);
20     }
21
22     public function product()
23     {
24         return $this->hasMany(Product::class);
25     }
26 }

```

Live Share Blame wensen2525 (4 days ago) Ln 12, Col 25 (5 selected) Spaces: 4 UTF-8 CRLF PHP Go Live tabnine starter Prettier 14:37 15/12/2023

Category (category model ini mempunyai relationship dengan produk)



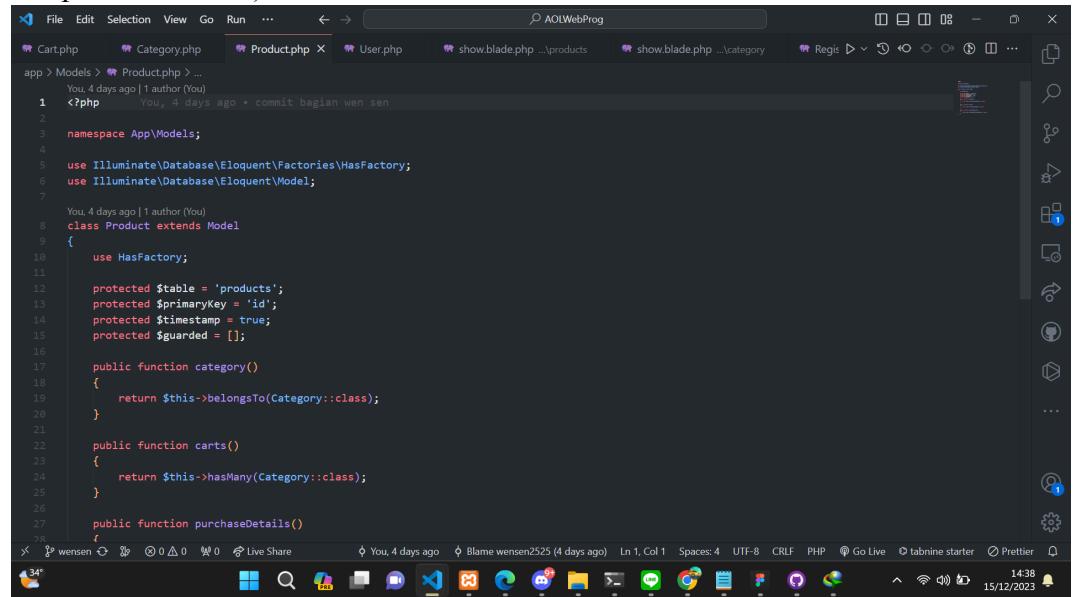
```

File Edit Selection View Go Run ... < > AOLWebProg
Cart.php User.php show.blade.php ...products show.blade.php ...category RegisterController.php ... ...
app > Models > Category.php > ...
You, 4 days ago | 1 author (You)
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 You, 4 days ago | 1 author (You)
9 class Category extends Model
{
10     use HasFactory;
11
12     protected $table = 'categories';
13     protected $primaryKey = 'id';
14     protected $timestamp = true;
15     protected $guarded = [];
16
17     public function products()
18     {
19         return $this->hasMany(Product::class);
20     }
21 }

```

Live Share You, 4 days ago Blame wensen2525 (4 days ago) Ln 1, Col 1 Spaces: 4 UTF-8 CRLF PHP Go Live tabnine starter Prettier 14:37 15/12/2023

Product (product model ini mempunyai relationship dengan category, cart, dan purchaseDetail)

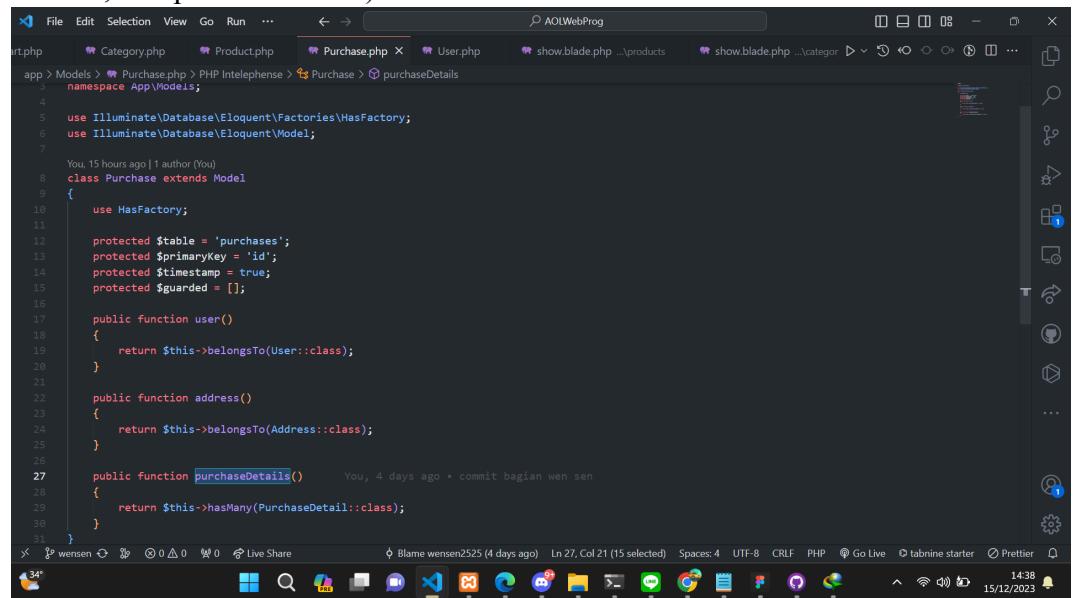


```

File Edit Selection View Go Run ... ↵ → 🔍 AOLWebProg
Cart.php Category.php Product.php X User.php show.blade.php ...products show.blade.php ...category Regis D... O... C... I... P... S... E... M... F... T... H... L... N... X...
app > Models > Product.php > ...
You, 4 days ago | 1 author (You)
1 <?php
2     You, 4 days ago * commit bagian wen sen
3
4     namespace App\Models;
5
6     use Illuminate\Database\Eloquent\Factories\HasFactory;
7     use Illuminate\Database\Eloquent\Model;
8
9     You, 4 days ago | 1 author (You)
10    class Product extends Model
11    {
12        use HasFactory;
13
14        protected $table = 'products';
15        protected $primaryKey = 'id';
16        protected $timestamp = true;
17        protected $guarded = [];
18
19        public function category()
20        {
21            return $this->belongsTo(Category::class);
22        }
23
24        public function carts()
25        {
26            return $this->hasMany(Category::class);
27        }
28
29        public function purchaseDetails()
30        {
31            You, 4 days ago * commit bagian wen sen
32            return $this->hasMany(PurchaseDetail::class);
33        }
34    }

```

Purchase (purchase model ini mempunyai relationship dengan user, address, dan purchaseDetail)



```

File Edit Selection View Go Run ... ↵ → 🔍 AOLWebProg
Cart.php Category.php Product.php Purchase.php X User.php show.blade.php ...products show.blade.php ...category Regis D... O... C... I... P... S... E... M... F... T... H... L... N... X...
app > Models > Purchase.php > PHP Intelphense > Purchase > purchaseDetails
You, 15 hours ago | 1 author (You)
1 <?php
2     namespace App\Models;
3
4     use Illuminate\Database\Eloquent\Factories\HasFactory;
5     use Illuminate\Database\Eloquent\Model;
6
7     You, 15 hours ago | 1 author (You)
8     class Purchase extends Model
9     {
10         use HasFactory;
11
12         protected $table = 'purchases';
13         protected $primaryKey = 'id';
14         protected $timestamp = true;
15         protected $guarded = [];
16
17         public function user()
18         {
19             return $this->belongsTo(User::class);
20         }
21
22         public function address()
23         {
24             return $this->belongsTo(Address::class);
25         }
26
27         public function purchaseDetails()      You, 4 days ago * commit bagian wen sen
28         {
29             return $this->hasMany(PurchaseDetail::class);
30         }
31     }

```

PurchaseDetail (purchaseDetail model ini mempunyai relationship dengan purchase dan produk)

The screenshot shows a Microsoft Edge browser window with the following details:

- Title Bar:** AOLWebProg
- Address Bar:** AOLWebProg
- Left Sidebar:** Shows navigation links: File, Edit, Selection, View, Go, Run, ...; and a search bar.
- Right Sidebar:** Includes icons for copy, paste, find, refresh, and other developer tools.
- Content Area:** A code editor with the following PHP code:

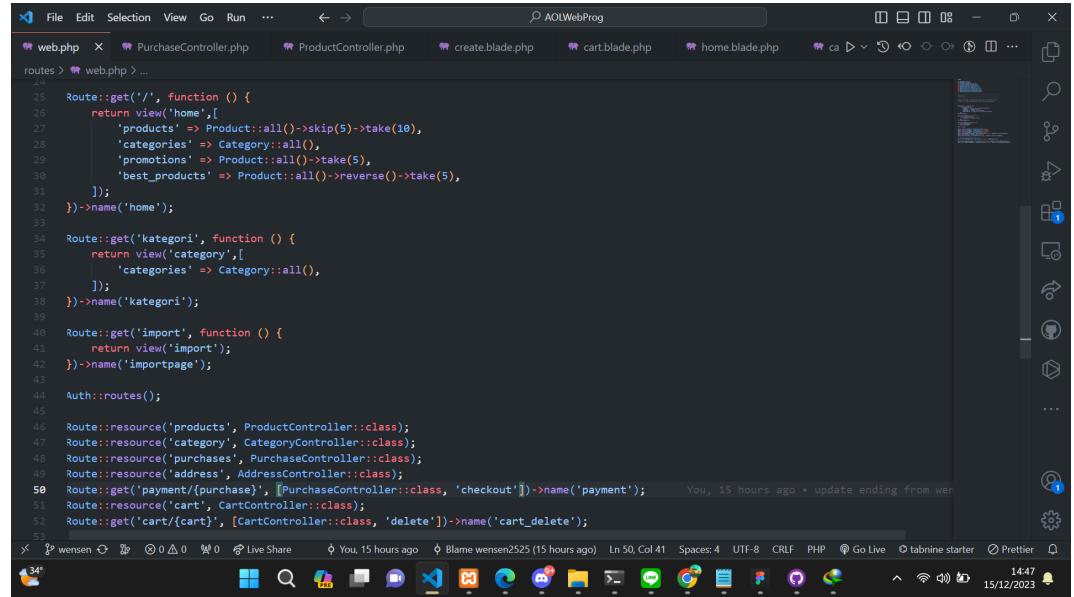
```
You, 4 days ago | author (You)
1  krphp You, 4 days ago • commit bagian wen sen
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 You, 4 days ago | author (You)
9 class PurchaseDetail extends Model
10 {
11     use HasFactory;
12
13     protected $table = 'purchase_details';
14     protected $primaryKey = 'id';
15     protected $timestamp = true;
16     protected $guarded = [];
17
18     public function purchase()
19     {
20         return $this->belongsTo(Purchase::class);
21     }
22
23     public function product()
24     {
25         return $this->belongsTo(Product::class);
26     }
27 }
```

The code is a Laravel Eloquent model named `PurchaseDetail`. It uses the `HasFactory` trait and extends the `Model` class. The table name is `purchase_details`, and the primary key is `id`. The model has two relationships: `purchase` (belonging to `Purchase`) and `product` (belonging to `Product`).

User (user model ini mempunyai relationship dengan purchase, address, dan cart)

2.4. Route

Routing di sini mencakup semua route yang digunakan di dalam project kali ini, ada yang memakai resource, lalu ada juga yang memanggil langsung tanpa menggunakan controller, dan ada juga yang passing data lewat routing ini juga.



```

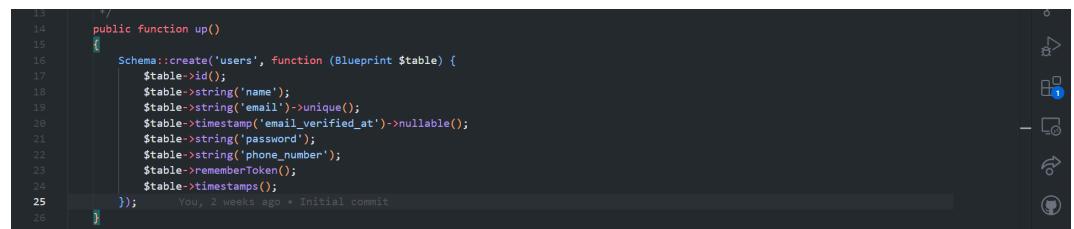
File Edit Selection View Go Run ... < > AOLWebProg
routes > web.php > ...
24
25 Route::get('/', function () {
26     return view('home',[ ...
27         'products' => Product::all()->skip(5)->take(10),
28         'categories' => Category::all(),
29         'promotions' => Product::all()->take(5),
30         'best_products' => Product::all()->reverse()->take(5),
31     ]);
32 })->name('home');
33
34 Route::get('kategori', function () {
35     return view('category',[ ...
36         'categories' => Category::all(),
37     ]);
38 })->name('kategori');
39
40 Route::get('import', function () {
41     return view('import');
42 })->name('importpage');
43
44 Auth::routes();
45
46 Route::resource('products', ProductController::class);
47 Route::resource('category', CategoryController::class);
48 Route::resource('purchases', PurchaseController::class);
49 Route::resource('address', AddressController::class);
50 Route::get('payment/{purchase}', [PurchaseController::class, 'checkout'])->name('payment');
51 Route::resource('cart', CartController::class);
52 Route::get('cart/{cart}', [CartController::class, 'delete'])->name('cart_delete');
53
54
You, 15 hours ago + update ending from wensen
You, 15 hours ago Blame wensen2525 (15 hours ago) Ln 50, Col 41 Spaces: 4 UTF-8 CRLF PHP Go Live tabnine starter Prettier
14:47 15/12/2023

```

2.5. Middleware

Middleware di sini merupakan field field yang bakal dipakai di project ini

users



```

13
14     public function up()
15     {
16         Schema::create('users', function (Blueprint $table) {
17             $table->id();
18             $table->string('name');
19             $table->string('email')->unique();
20             $table->timestamp('email_verified_at')->nullable();
21             $table->string('password');
22             $table->string('phone_number');
23             $table->rememberToken();
24             $table->timestamps();
25         });
26     }
You, 2 weeks ago + Initial commit

```

addresses



```

14
15     public function up()
16     {
17         Schema::create('addresses', function (Blueprint $table) {
18             $table->id();
19             $table->string('name');
20             $table->longText('address');
21             $table->unsignedBigInteger('user_id');
22             $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
23             $table->timestamps();
24         });
25     }

```

purchases

```

14     public function up()
15     {
16         Schema::create('purchases', function (Blueprint $table) {
17             $table->id();
18             $table->unsignedBigInteger('user_id');
19             $table->foreign('user_id')->references('id')->onDelete('cascade');
20             $table->unsignedBigInteger('address_id')->nullable();
21             $table->foreign('address_id')->references('id')->on('addresses')->onDelete('cascade');
22             $table->string('payment_due');
23             $table->string('delivery_type')->nullable();
24             $table->string('payment_type')->nullable();
25             $table->integer('totalPrice')->nullable();
26             $table->boolean('is_verified')->default(false);
27             $table->timestamps();
28         });
29     }
30

```

People
Innovation
Excellence

categories

```

14     public function up()
15     {
16         Schema::create('categories', function (Blueprint $table) {
17             $table->id();
18             $table->string('name');
19             $table->string('image');
20             $table->timestamps();
21         });
22     }
23

```

products

```

14     public function up()
15     {
16         Schema::create('products', function (Blueprint $table) {
17             $table->id();
18             $table->string('name');
19             $table->unsignedBigInteger('category_id');
20             $table->foreign('category_id')->references('id')->onDelete('cascade');
21             $table->longText('description')->nullable();
22             $table->longText('ingredients')->nullable();
23             $table->longText('howtouse')->nullable();
24             $table->longText('doe')->nullable();
25             $table->longText('sideeffects')->nullable();
26             $table->string('price');
27             $table->string('image');
28             $table->integer('stock')->default(100);
29             $table->integer('stock_sold')->default(0);
30             $table->boolean('is_open')->default(true);
31             $table->timestamp();
32         });
33     }
34

```

purchase_details

```

14     public function up()
15     {
16         Schema::create('purchase_details', function (Blueprint $table) {
17             $table->id();
18             $table->unsignedBigInteger('purchase_id');
19             $table->foreign('purchase_id')->references('id')->on('purchases')->onDelete('cascade');
20             $table->unsignedBigInteger('product_id');
21             $table->foreign('product_id')->references('id')->on('products')->onDelete('cascade');
22             $table->integer('price');
23             $table->integer('quantity');
24             $table->integer('totalPrice')->nullable();
25             $table->timestamps();
26         });
27     }
28

```

carts

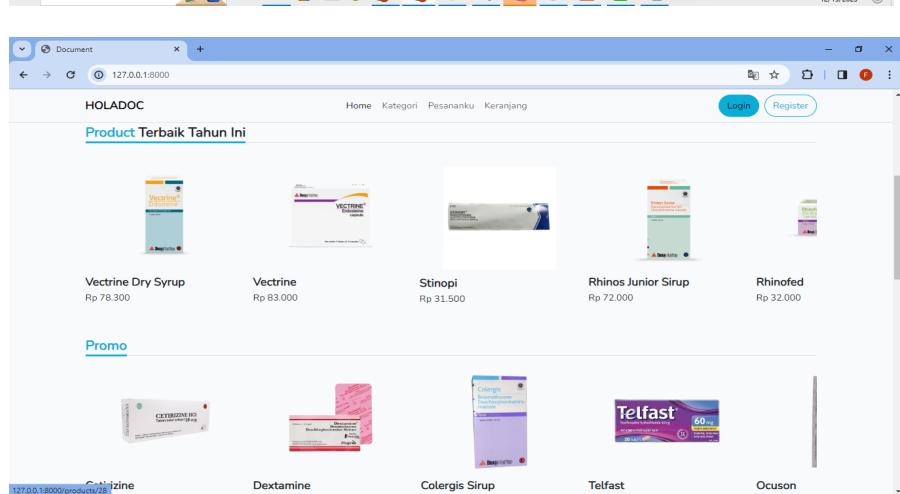
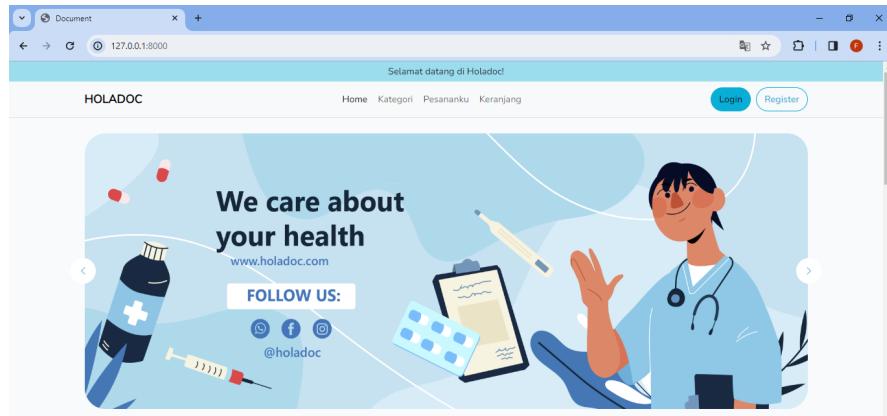
```

14     public function up()
15     {
16         Schema::create('carts', function (Blueprint $table) {
17             $table->id();
18             $table->unsignedBigInteger('product_id');
19             $table->foreign('product_id')->references('id')->on('products')->onDelete('cascade');
20             $table->unsignedBigInteger('user_id');
21             $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
22             $table->integer('stock');
23             $table->timestamps();
24         });
25     }
26

```

BAB 3. USER INTERFACE

3.1 HOME PAGE



The screenshot shows the HOLADOC website interface. At the top, there is a header with the BINUS UNIVERSITY GRADUATE logo. Below the header, the main content area is titled "HOLADOC". It features a "Promo" section displaying five products: Cetirizine, Dextamine, Colergis Sirup, Telfast, and Ocuson, each with a small image and price. Below this is a "Kategori Terlaris" section with five categories: Alergi, Pencernaan, Suplemen & Vitamin, Asma, and Batuk, Pilek, & Flu, each represented by a circular icon. At the bottom of this section, it says "Obat Sehari-Hari".

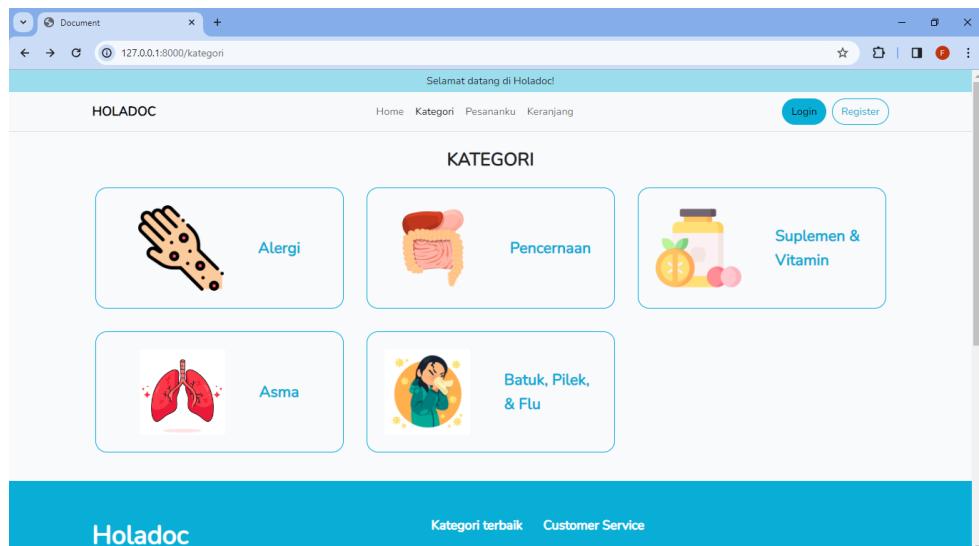
People
Innovation
Excellence

This screenshot shows the HOLADOC website again, but this time the "Vitamin" and "Flu" categories are selected. The "Obat Sehari-Hari" section is still present at the top. Below it, there are five product images: Bufacomb Cream, New Diatabs, Oralit, Polysilane Suspensi, and Promag, each with its name and price.

Pada tampilan pertama, user akan masuk ke tampilan home di home ini akan menampilkan header yang terdapat banner holadoc, dan dibawahnya terdapat list dari produk terbaik pada tahun ini, setelah itu di home terdapat produk yang sedang promo.

Dan terdapat kategori dari produk yang terlaris, dan adanya obat sehari-hari yang ditampilkan.

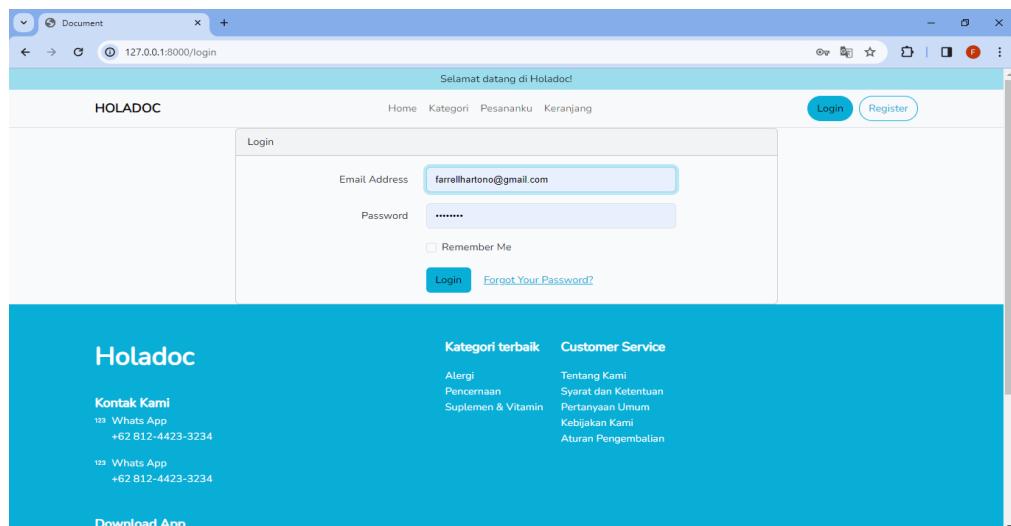
3.2 CATEGORY PAGE



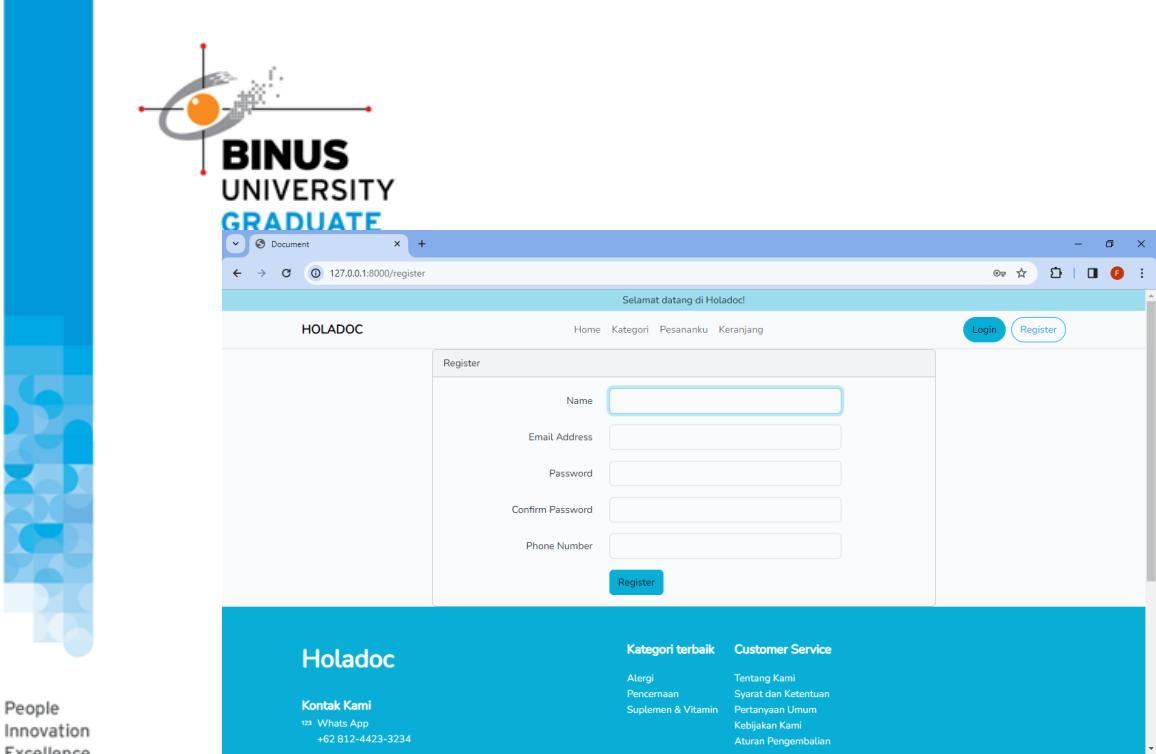
People
Innovation
Excellence

Pada tampilan ini terdapat page category yang telah kami buat. Kami memakai elemen bootstrap pada tampilan category, supaya website lebih responsive

3.3 LOGIN/REGISTER PAGE



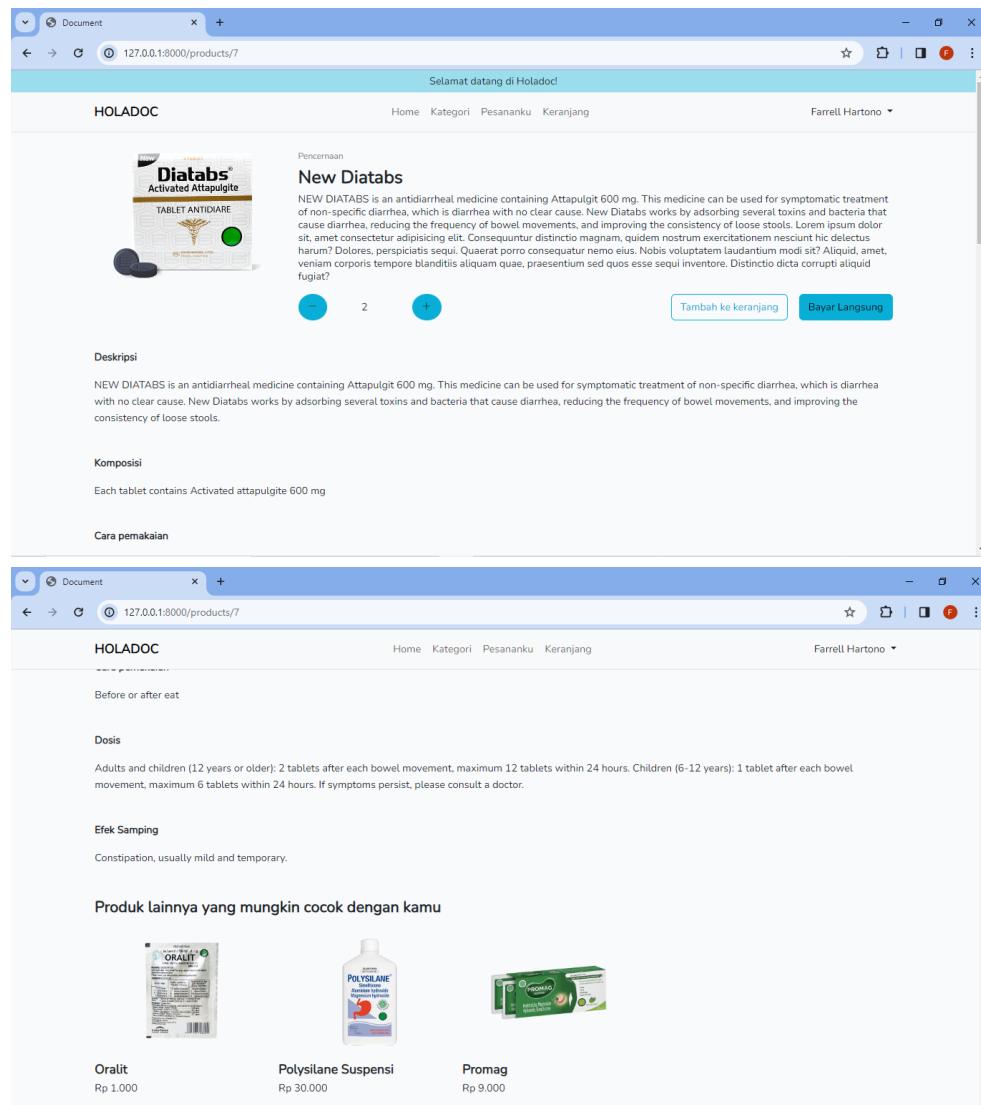
Berikut merupakan login page dari website kami, login page ini dibutuhkan jika membutuhkan user id, seperti ke page pesananku, dan keranjang. Di kedua page itu membutuhkan user id, maka jika user harus login terlebih dahulu, sebelum bisa mengakses pesananku, dan keranjang.



Ini merupakan register page yang dibuat dengan framework bootstrap.

3.4 PRODUCT DETAIL PAGE

People
Innovation
Excellence



Selamat datang di Holadoc!

HOLADOC

Pencarian

New Diatabs

NEW DIATABS is an antidiarrheal medicine containing Attapulgite 600 mg. This medicine can be used for symptomatic treatment of non-specific diarrhea, which is diarrhea with no clear cause. New Diatabs works by adsorbing several toxins and bacteria that cause diarrhea, reducing the frequency of bowel movements, and improving the consistency of loose stools.

2

Tambah ke keranjang Bayar Langsung

Deskripsi
NEW DIATABS is an antidiarrheal medicine containing Attapulgite 600 mg. This medicine can be used for symptomatic treatment of non-specific diarrhea, which is diarrhea with no clear cause. New Diatabs works by adsorbing several toxins and bacteria that cause diarrhea, reducing the frequency of bowel movements, and improving the consistency of loose stools.

Komposisi
Each tablet contains Activated attapulgite 600 mg

Cara pemakaian
Before or after eat

Dosis
Adults and children (12 years or older): 2 tablets after each bowel movement, maximum 12 tablets within 24 hours. Children (6-12 years): 1 tablet after each bowel movement, maximum 6 tablets within 24 hours. If symptoms persist, please consult a doctor.

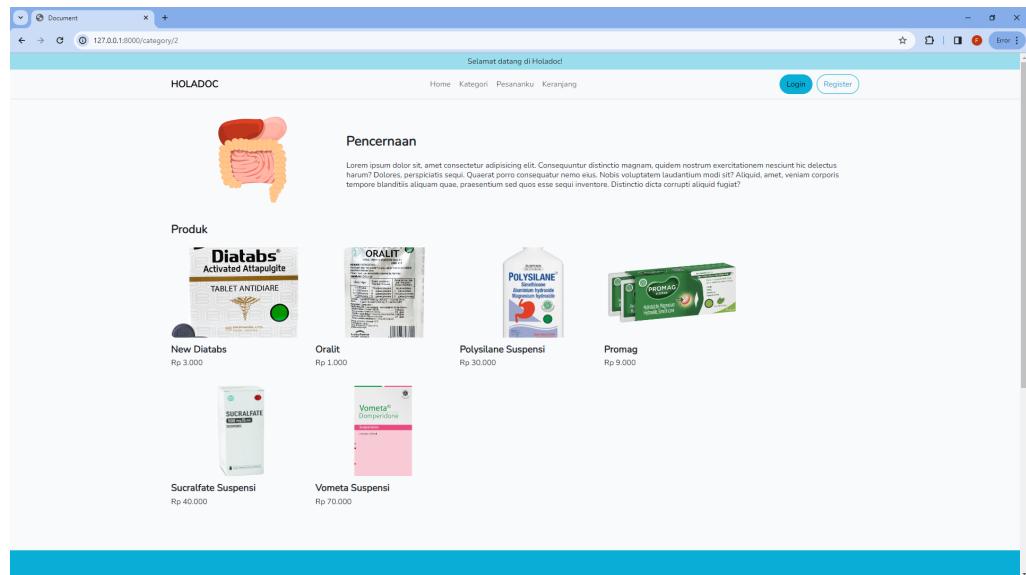
Efek Samping
Constipation, usually mild and temporary.

Produk lainnya yang mungkin cocok dengan kamu

Produk	Harga
Oralit	Rp 1.000
Polysilane Suspensi	Rp 30.000
Promag	Rp 9.000

merupakan tampilan detail dari produk, ada fitur untuk menambah jumlah barang yang diinginkan, di website ini ada penjelasan tentang produknya ada deskripsi, komposisi, cara pemakaian, dosis, efek samping serta ada tampilan produk lainnya

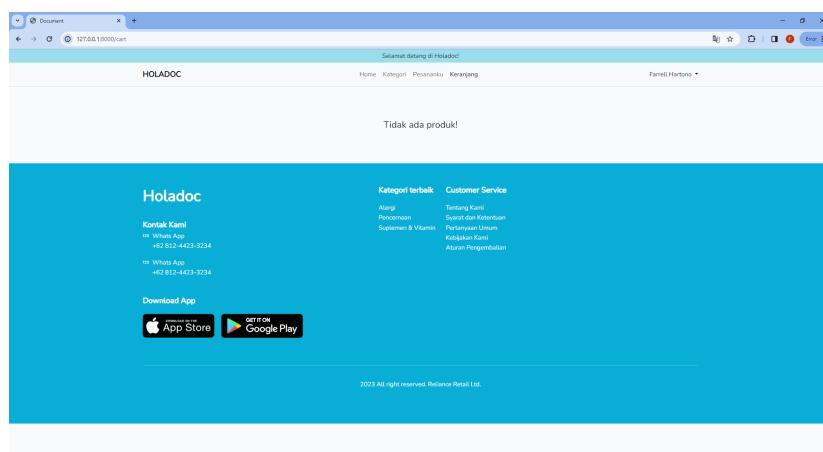
3.5. PRODUCT CATEGORY DETAIL PAGE



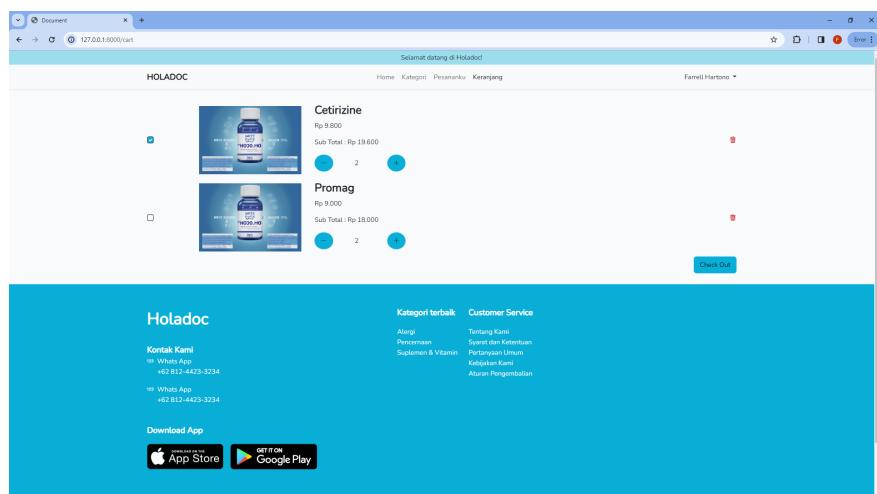
Setelah memencet kategori, akan muncul page product category, yang isinya ada gambar dari kategorinya sendiri dan produk dari kategori tersebut.

3.6 SHOPPING CART PAGE

People
Innovation
Excellence

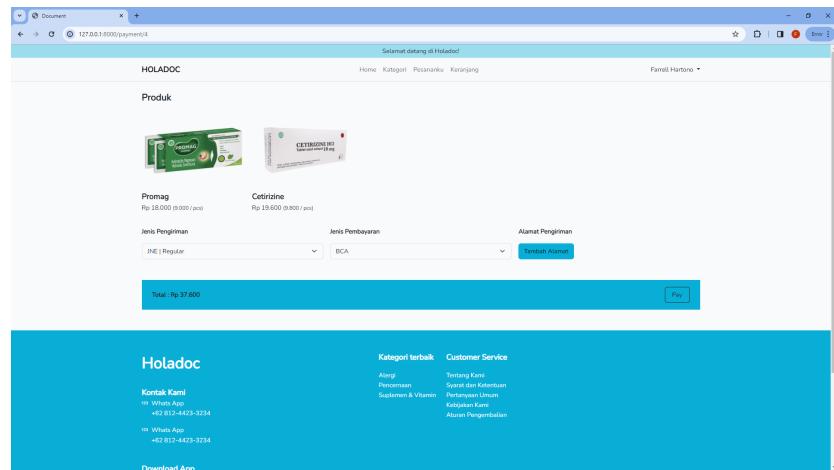


Dalam page ini, jika tidak ada produk yang dimasukkan ke keranjang maka page akan menampilkan tidak ada produk.



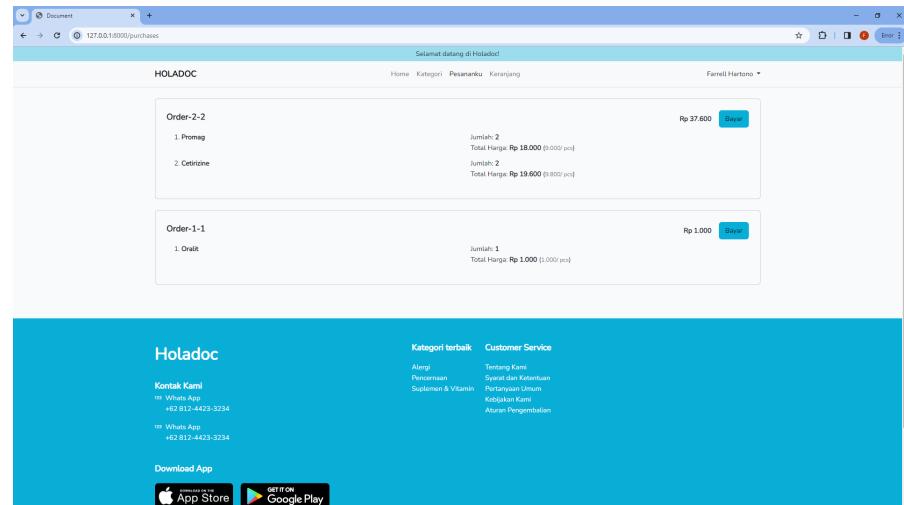
Setelah itu jika kita sudah menambahkan produk ke keranjang, maka tampilan page akan seperti berikut.

3.7 PAYMENT METHOD PAGE



Dalam page payment method, terdapat hasil check out dari keranjang yang dipilih, serta metode pembayaran yang dapat dipilih berupa, BCA, Mandiri, BRI, OVO, DANA, dan jenis pengiriman berupa JNE, JNT, dan Si cepat

3.8 PURCHASE PAGE



The screenshot shows a web browser displaying the HOLADOC purchase page. The page header includes the BINUS UNIVERSITY GRADUATE PROGRAM logo, the word "HOLADOC", and a user profile for "Farrell Hartono". Below the header, there are two sections for orders:

- Order-2-2:** Contains items 1. Promog and 2. Cetirizine. The total price is Rp 37.600.
- Order-1-1:** Contains item 1. Oralit. The total price is Rp 1.000.

At the bottom of the page, there is a footer with links for "Kontak Kami", "Download App" (links to both Apple App Store and Google Play), and "Customer Service" (links to Tertulis Kami, Syarat dan Ketentuan, Pertanyaan Umum, Kolaborkan Kami, and Alasan Pengembalian).

Terdapat page purchase yang isinya, dari keranjang-keranjang yang sudah dipilih, jika menklik tombol pay maka akan masuk ke page payment method

BAB 4. KESIMPULAN & SARAN

4.1. KESIMPULAN

HolaDoc merupakan e-commerce yang menjual berbagai jenis obat-obatan dengan kenyamanan dan aksesibilitas tinggi bagi konsumen.HolaDoc memberikan solusi yang efektif dalam penjualan obat-obatan. Dengan interface atau tampilan yang sederhana dan menarik, pengguna dapat dengan mudah memahami, mencari, membandingkan, dan membeli berbagai jenis obat. pilihan produk yang luas dan sudah dikategorisasi beserta deskripsi produk yang singkat, padat, dan jelas memberikan kemudahan dan fleksibilitas dalam memilih varian obat-obat yang sesuai.

4.2. SARAN

Dalam pembuatan website ini, kami berhasil mencapai berbagai macam tujuan yang telah kami utarakan pada pada bab pendahuluan. Namun, keterbatasan sumber daya dan waktu, kami menyadari bahwa masih banyak kekurangan pada website kami. Oleh karena itu, pada pengembangan selanjutnya, berbagai fitur dapat ditambahkan antara lain:

- **Fitur konsultasi** dengan profesional berlisensi untuk memberikan nasihat dan merespon kebutuhan konsumen seputar obat-obatan
- **Pengingat Obat** untuk membantu konsumen mengingat jadwal untuk mengonsumsi obat secara teratur.
- **Artikel Kesehatan** untuk dibaca pengguna agar pengguna dapat lebih teredukasi mengenai kesehatan dan obat-obatan.

DAFTAR PUSTAKA

[Halodoc - Beli Obat, Tanya Dokter, Cek Lab Terpercaya](#)

People
Innovation
Excellence