

E-commerce Application

Overview

An E-commerce Application is a project which will be used by the customer for buying the products and admin for managing the products and order details.

Goals

Laravel: Creating an Admin panel with the understanding of all the concepts such as integration of themes, migrations, seeders, validations, ORM, API, Passport/ JWT token

Vue : Creating an Customer panel with the understanding of all the concepts with the integration of themes, use of API's

Specifications

Use the latest version of the frameworks and plugins.

You should be good in Laravel and Vue, JQuery and other required languages.

Your coding should meet the standards like indentation, comments, variable names, request type, parameter name, code reusability etc.

After every form action (Create/ Update/ Delete) there should be a notification.

Be more careful while choosing the type of DB column, it should be generic and logical don't use any type blindly.

The development should be on time and also consist of each aspect or module.

Prepare the document so that you can demo it to others as well.

Milestones

Admin theme setup

Download the theme from :

<https://github.com/ColorlibHQ/AdminLTE/archive/refs/tags/v3.1.0.zip>

Theme preview is from : <https://adminlte.io/themes/v3/>

Integrate it with the Laravel framework and set up Login and other pages.

Database setup

Setup your database using

1. Migration
2. Seeding

Role setup and admin login

Before starting with login and user management follow below steps

1. Create roles table using migration with following column

role_id [int][pk]

role_name [varchar(100)]

2. Add following hardcoded entry using seeding in above roles table

superadmin

admin

inventory manager

order manager

Customer

3. Add hardcoded entry in users table with following information [Refer table: users]

firstname = Admin

lastname = admin

email = {your_email}

password = admin123

Implement the admin login. other than customer user will able to login in admin. so you need to check role type of the user. Different types of user have different access rights.

User Management

It will be normal Add/Edit/Modify in admin section for Admin Users.

Form Fields:

Firstname - Textbox - Compulsory validation

Lastname - Textbox - Compulsory validation

Email - Textbox - Compulsory, Email validation

Password - Password - Compulsory, Alphanumeric, between 8 to 12 characters validation

Confirm Password - Compulsory, Same as above

Status - Radio Button (Active / Inactive) - Compulsory

Role - select box [values will come from roles tables] [default will be customer]

Refer table: users and roles

Configuration Management

Admin can define constants which can be used for shopping cart. These values need to be defined in admin so that admin can change those any point of time without changing any code. For ex: admin_email, notification_email. Use same CRUD package.

Refer table: configuration

Admin - Banner Management

Refer Banner table and use CRUD to create ADD / EDIT / DELETE operation.

Admin - Category Management

It's a multi level category system. Only admin can add categories. Use ACL management package and CRUD package for implementation.

Admin - Product Management

Admin can manage the product from admin site. Basic Product details will be saved in product table. Product images will be saved in product_images table. Product categories will be saved in product_categories table. Product parameters will be saved in product_attributes_assoc table.

Refer : product, product_categories, product_images, product_attributes_assoc

Admin - Coupon Management

Admin will be able to generate the coupon code which will get applied on grand total – shipping charges. This alphanumeric code admin will generate from admin which can be redeemed in frontend.

NOTE : YOU NEED TO USE STORED PROCEDURE FOR THIS

Refer table: coupons

Front End : Home Page - Frontend Design Setup

Create different Controller directory for Admin and Frontend as per HMVC

Design integration of Home page

<https://www.dropbox.com/s/j8bhi1qd2nyfine/e-shopper-best-free-ecommerce-html-template.zip?dl=0>

Use laravel FrontEnd scaffolding and Compiling Assets for minified and mix of CSS and Javascript (<https://laravel.com/docs/5.5/frontend>) (<https://laravel.com/docs/5.5/mix>)

Front End : User Registration and Login

Frontend users can register and login using any of the following three methods:

By filling up form

Facebook Connect

Twitter Connect

Google Connect

NOTE : Role type for front end user will be customer

Refer table: user

Front End : Shopping Cart

User can browse through categories, subcategories and product and add any of them in the cart. The session should be handle using session. We don't save this session data in ase tables. User will be able to redeem the coupon generated by admin in frontend.

Refer package : <https://packagist.org/packages/gloudemans/shoppingcart>

Front End : Checkout

User can checkout using following steps:

Login / Register – User will have to compulsorily login to the system to purchase anything

Billing address – Select from your address book or add address while checkout

Shipping address - Select from your address book or add address while checkout

Shipping Method – We need to implement shipping rule. If order size is above 500 then there will be free shipping that mean shipping charges will be 0. Else there will be \$50 shipping charges.

Payment Method – There will be two payment methods

COD – Cash on Delivery

Paypal IPN – Refer https://developer.paypal.com/docs/classic/ipn/gs_IPN/

Order Review – It should show all the details of above 5 steps.

Before redirecting control to Paypal IPN insert the records in table user_order and order_details with status pending. If we get success message update the transaction_id and make the status as Processing.

Front End : WishList

User will be able to add the product to his wish list. He should be login to the system in order to save this in wish list for future purchase.

Refer table: user_wish_list

Front End : My Account

User will be able to manage his profile information from this section.

Refer table: user

Front End : My Orders

User will be able to see the orders placed on the website in this tab. User will be able to see the current status of the order here.

Refer table: user_order, order_details, user, user_address, coupons, coupons_used

Front End : Track Order

In this page user will be able to enter his order id and email id and get the order status.

Refer table: user, user_orders

Front End : Change password

If user wants to change password user will be able to do that from this section.

Refer table: user

Front End : Contact US

User will be able to post the queries from frontend form. Admin will be able to see those in backend

Name

Email

Contact No

Message

Refer table: contact_us

Front End : Newsletters

If anyone subscribe for newsletter then add him directly to Mailchimp. Refer link

<http://apidocs.mailchimp.com/api/>. Create trial account on it.

Front End / Admin : Email Notification

Admin can manage the content of email sent via system using this module. Content will have the variables which will be replace dynamically.

Refer table: email_template

Whenever user register on the website email will be sent to user with login credentials.

Admin will get copy of above email without password.

Whenever customer places the order, all the order details will be sent to customer email id.

One copy of above will be sent to admin

Admin changes status of order then email will be sent to customer

Contact us form submission notification to admin

If Admin responds to contact us form send email to customer with note.

Send one consolidated email to admin with order placed within a day. It should be through cron.

Send one email to admin with user wish list once in a week.

For content refer: <https://www.dropbox.com/s/4cs7iil503i7nwl/Email%20Notification.docx?dl=0>

Admin - Contact Us

Admin will able to see all the message which sent from front end form

Admin : CMS

Admin can manage the content of static pages using this module.

Refer Table: cms

Admin : Order Management

Admin should be able to see the order listing and the details of each order. It should contain the following information:

1. Customer details with Address
2. Ordered products
3. Shipping Method
4. Payment gateway details.

Admin : Reports

Admin will be able to grab below reports from the admin section.

Sales Report

Customer Registered

Coupons used