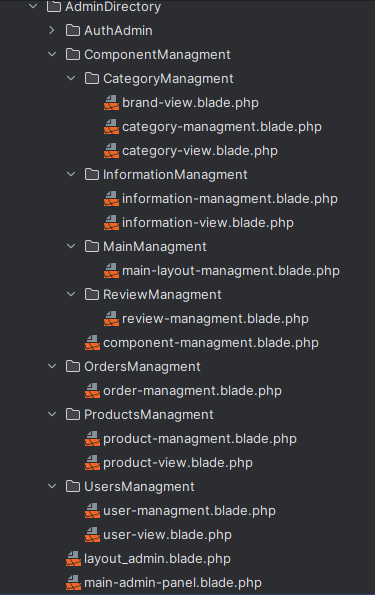
С английским у меня плохо, поэтому некоторые переменные и названия могут быть ну уж очень плохо названы, по типу Backet как корзина (место card или managment место нормального management \*до последнего думал, что я писал правильно\*)

К дополнению по комментариям на всякий случай добавляю этот файл.

Задание было решено не полностью, некоторые детали я сделать не смог, какие именно описаны в «Тестовое задание\_ Реализация интернет-магазина фиксирование прохождения.docx»



Для начала описание части администратора.

«AuthAdmin» - папка с view для авторизации в панели администратора.

«ComponentManagment» - папка для «компонентов», а именно:

1)\*CategoryManagment\* - Категории/Бренды и их вывода (brand-view; category-view; category-management)

2) \*InformationManagment\* - для карточек информации и их вывода (information-management; information-view)

3)\*MainManagment\* - для управления главной страницей

4)\*ReviewManagment\* - для управления отзывами

5)component-managment – файл «родитель» для остальных в этой категории

«OrdersManagment» - папка для управления заказами

«ProductsManagment» - папка для управления товарами:

1)product-managment – для просмотра всех товаров

2)product-view – для управления определенного товара

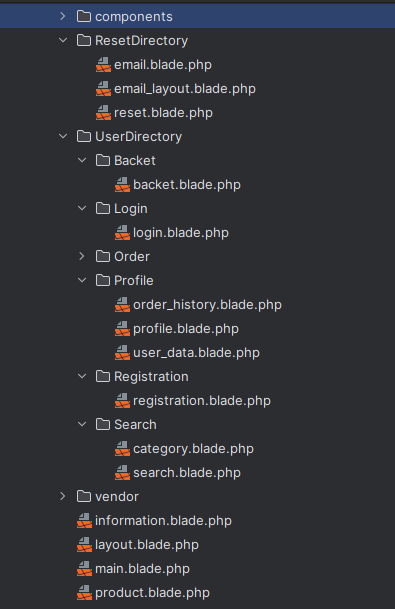
«UsersManagment» - папка для управления пользователями

1)user-managment – просмотр пользователей

2)user-view – для просмотра определенного пользователя

«layout\_admin» - файл конструктор для административной части в целом

«main-admin-panel» - главная страница административной части



«components» - попытался реализовать «хлебные крошки» но не получилось.

«ResetDirectory» - папка для восстановления пароля по почте с конструктором

«UserDirectory» - папка для пользовательской части:

1. «Backet» - папка для корзины
2. «Login» - для авторизации в пользовательской части
3. «Order» - для создания и подтверждения заказа
   1. «create\_order» - форма для создания заказа
   2. «final\_order» - форма для подтверждения созданного заказа
4. «Profile» - папка для профиля авторизированного пользователя
   1. «order\_history» - просмотр всех заказов
   2. «profile» - главная страница профиля
   3. «user\_data» - форма для редактирования персональных данных
5. «Registration» - папка для регистрации пользователя
6. «Search» - папка для поиска по категориям и для обычного поиска
   1. «category» - для вывода категории
   2. «search» - для поиска

«information» - форма для вывода карточки информации

«layout» - форма конструктор для пользовательской части

«main» - форма для главной страницы

«product» - форма для вывода товара  
  
  
  
найденные ошибки:  
1. Ошибка при запуске миграции

2. В задании было указано, что надо было найти готовый дизайн (не готовый шаблон) интернет магазина и его сверстать при помощи tailwind, по факту сделан голый css с элементами tailwind и то верстка очень сильно хромает... (даже админ панель сверстана, когда было разрешено взять готовый шаблон)

3. Проверка авторизации сделана через отдельный middleware, когда для проверки авторизации есть отдельный, встроенный middleware ->middleware('auth')

4. Наименования не только переменных, а вообще всего, хромает... Все можно назвать намного проще и лаконичнее. Также очень большие проблемы с форматироваанием кода, очень тяжело читается, потому что много элементов идут слошняком пелиной

5. Жесткое забивание ID-шников ролей не есть хорошо, очень не хорошо, вообще в системах делать что-либо опирающееся на ID настоятельно не рекомендуется, надо было придумать другое решение

6. Есть проверки на вывод кнопок на определенные роли, а вот если перейти на запрещенную для, например, менеджера ссылку создания пользователь, то она откроется и можно спокойно делать свои дела, надо было сделать отдельный middleware и применять, где необходимо

7. Очень много повторяющегося кода, который можно было вынести в заглавные контроллеры, потому что если что-то надо будет поменять, то придется не в 1-ом месте менять, а в каждом контроллере, где это используется

8. Метод create для редактирования и для создания (например пользователя, который не работает) делать нельзя, лучше делать все строго по канонам MVC, т.е. для CRUD систем обязательно наличие (можно что-то исключить, добавлять, НО НЕ ОБЪЕДИНЯТЬ обязательные в одно) index, create, store, edit, update, destroy методов, для таких методов как create и edit можно использовать одну и туже форму

9. edit и update делаются через невидимый input id, это делать очень не правильно, в админ панелях для вывода определенной записи из БД, надо id размещать в URL (/show/{id} - взято из документации Laravel) и после при помощи специальной записи Laravel сам найдет запись или выведет 404

10. Очень все напутано и плохо названо в роутах, черт ногу сломит что за что отвечает

11. Очень большая ошибка, нет валидаций на приходящие данные, есть только на некоторые методы, а надо на все!!! Также не надо было бы перебирать все input в котроллерах, тебе бы валидация выдала данные, которые тебе нужны для БД, не надо было бы писать километровый с получением данных из input-ов

12. Очень странно сделаны фильтрации, можно было сделать 1 where, в котором функция и она проверяет какие параметры фильтров пришли и применять их

13. В товарах используется поле manufacturer, когда нет никакого раздела где я могу их добавить/редактировать, а без него мне не дает создать товар

14. Какие-то траблы с таблицей статусов, везде жалуется, что он не знает, что за таблица statuses (таблица названа не верно, в единственном числе)

15. Я не могу попасть в админ панель никак, хотя бы фабрику сделал 1 тестовую запись, пришлось комментировать проверки на авторизацию