### МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕУЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

### Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Курсовая работа**

По дисциплине: Базы данных

Тема: Создание прототипа интернет-магазина

Автор работы \_\_\_\_\_\_ Маркевич Александр Александрович

(подпись)

Руководитель проекта \_\_\_\_\_\_ Панченко Максим Владимирович

(подпись)

Оценка \_\_\_\_\_\_

Белгород 2024

**Содержание**

1. Введение……………………………………………...3-4

1.1 Выбор подхода………………………………………5-6

2. Разработка Backend.........................................................7

2.1. ER-модель ………………………………………....8-11

2.2. Маршруты ...................................................................12

2.3 Описание функционала..........................................13-14

3. Разработка Frontend..................................................15-23

4. Заключение………………………………………….....24

5. Список литературы……………………………………25

6. Приложение с кодом…………………………………..25

**Глава 1. Введение**

Современный мир невозможно представить без Интернета. Он стал неотъемлемой частью жизни миллиардов людей по всему миру и существенно изменил способ, которым мы общаемся, работаем, учимся и развлекаемся. Интернет является глобальной сетью, объединяющей множество компьютеров и устройств, и предоставляющей доступ к огромному объему информации, услуг и возможностей.

Он применяется в различных целях, начиная от развлечений, заканчивая ведением бизнеса. Уже не секрет, что в интернете можно купить что угодно. Зачастую нам не сильно хочется идти в магазин и покупать, к примеру, одежду которой там может и не быть. Для этого и были созданы интернет-магазины — формы электронной коммерции, которые позволяют потребителям напрямую покупать товары или услуги у продавца.

Создание интернет-магазина стало популярным направлением в сфере разработки веб-приложений. В современном мире все больше людей предпочитают покупать товары онлайн, и интернет-магазины становятся неотъемлемой частью коммерческой деятельности.

Цель данной курсовой работы заключается в разработке интернет-магазина с использованием Laravel и React. Laravel — это прикладной фреймворк, который позволяет создавать мощные и масштабируемые веб-приложения. React — JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

Объект исследования является процесс разработки интернет-магазина и построения правильных базы данных.

Предмет исследования – языки программирования php и Typescript, язык разметки и стилей — HTML5 и CSS.

Цель – программно реализовать взаимодействие с базами данных, а также разработать прототип интернет-магазина.

Для достижения поставленной цели необходимо решить следующие задачи:

* Применить на практике необходимый стэк технологий.
* Подготовка (подключение пакетов и библиотек, необходимых сервисов для правильной работы с базами данных, связать backend с frontend и т. д.) .
* Связать правильно таблицы между собой
* Разработать алгоритмы и сервисы в проекте
* Сделать клиентскую часть приложения
* Протестировать работоспособность интернет-магазина

**Глава 1.1 Выбор подхода**

Создание сайтов является интересным и актуальным занятием, в нем решает не только алгоритмическая составляющая, но и творческая. Воплощение интернет-магазина позволит проработать все навыки и шаблоны, так необходимые разработчикам.

Выбор пал на разработку Web-приложения через фреймворки Laravel (Web api), React, Базу данных PostgreSQL:

Laravel:

Это популярный фреймворк для веб-приложений, написанный на языке PHP. Он предназначен для упрощения процесса разработки путем предоставления различных инструментов и библиотек, которые помогают создавать чистый, масштабируемый и поддерживаемый код.

React:

Это библиотека для создания пользовательских интерфейсов. Она позволяет разработчикам создавать многоразовые компоненты и управлять состоянием приложений в эффективном и предсказуемом стиле.

PostgreSQL:

База данных, которая поддерживает JSON и другие функции NoSQL, надежность и целостность данных. Открытое ПО.

**Стек и версии используемых технологий: PHP - 8.3.13, TypeScript –** версия5.5, React **–** 19 , Laravel **– 11**, **PostgreSQL ~16**

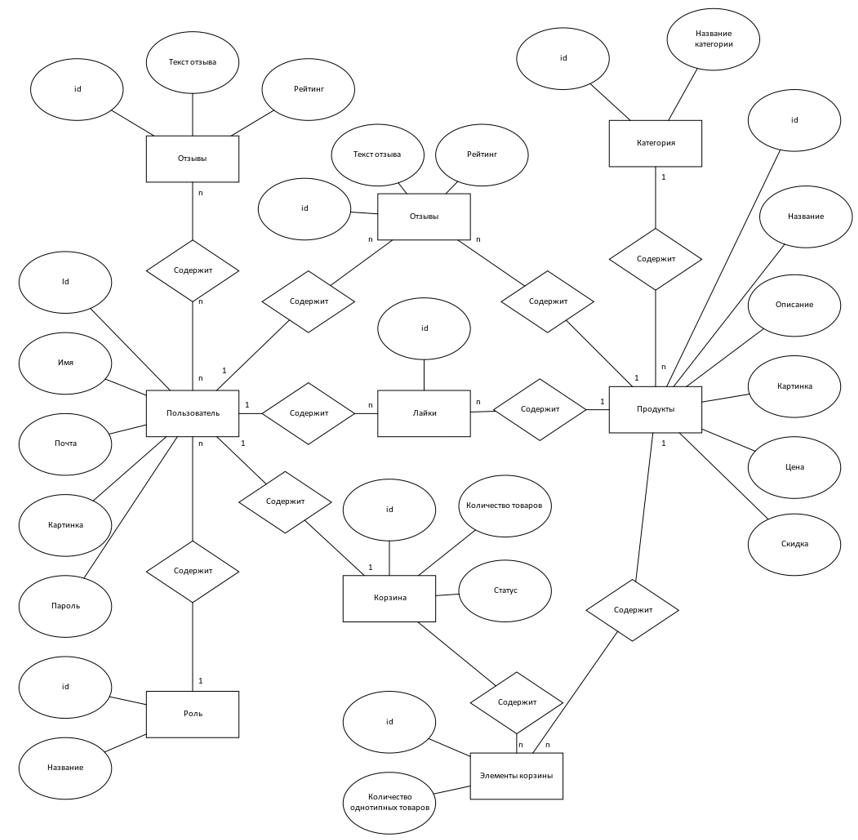
Глава 2. Разработка Backend

Бэкенд играет критически важную роль в обеспечении функциональности интернет-магазина: он отвечает за обработку запросов, работу с базами данных, аутентификацию пользователей и реализацию всех необходимых бизнес-процессов, таких как обработка заказов, управление товарами и аналитика.

В данной главе будет рассмотрен процесс проектирования и реализации RESTful API, который обеспечивает взаимодействие между фронтенд и бэкенд-частью приложения.

Глава 2.1. ER-модель

В данной подглаве мы будем рассматривать ER-модель (Entity-Relationship Model), которая является важным инструментом для проектирования баз данных в контексте разработки интернет-магазина. ER-модель позволяет визуализировать связь между различными сущностями, а также определить атрибуты каждой сущности и их взаимосвязи. Это важный этап проектирования, так как правильно построенная модель способствует более эффективной организации данных и упрощает процесс их управления.  
ER:

Рисунок 1

**Сущности:**

**Пользователь:** содержит информацию о пользователе: уникальный идентификатор (id), Имя, Почта, Пароль, Картинку(Аватар) идентификатор роли в приложении (role\_id)

**Роль пользователя:** содержит информацию о роли пользователя на уровне приложения: уникальный идентификатор (id), Название

**Отзывы:** содержит информацию о отзывах: уникальный идентификатор (id), Текст отзыва, Рейтинг, идентификатор пользователя в приложении(user\_id), идентификатор продукта в приложении(product\_id)

**Лайки:** содержит информацию о лайках: уникальный идентификатор (id), идентификатор пользователя в приложении(user\_id), идентификатор продукта в приложении(product\_id)

**Корзина:** содержит информацию о корзине: уникальный идентификатор (id), Количестве товара в корзине, Статус об том содержится ли товары в корзине, идентификатор пользователя в приложении(user\_id)

**Продукты в корзине:** содержит информацию о продуктах в корзине: уникальный идентификатор (id), идентификатор продукта в приложении(product\_id), идентификатор корзины в приложении(cart\_id), Количество повторяющихся товаров

**Продукты:** содержит информацию о продукте: уникальный идентификатор (id), Название, Описание, Картинка товара, Цена,

Скидка, идентификатор категории в приложении(category\_id)

**Категория:** содержит информацию о категории: уникальный идентификатор (id), Название

**Связи:**

Пользователь авторизуется под ролью. Для конкретной роли доступен разный функционал приложения.

Пользователь может лайкать товары.

Пользователь может добавлять товары в корзину.

Пользователь может добавлять отзывы в товаре.

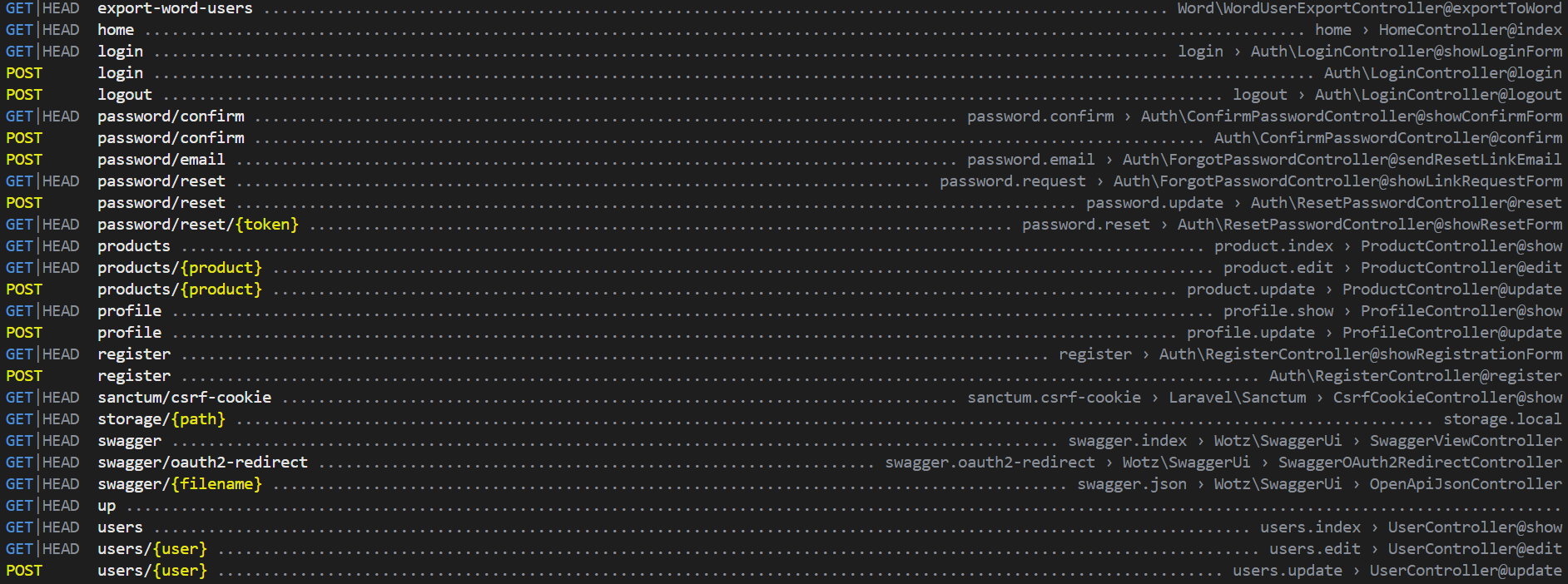
У одной категории может быть много товаров.

В корзине у пользователя может быть много товаров. Которые связана с продуктами.

Глава 2.2. Маршруты

В данной главе мы рассмотрим маршруты, которые представляют собой важную часть архитектуры веб-приложения интернет-магазина. Маршруты позволяют организовать обработку запросов пользователей и управление навигацией внутри приложения.

Рисунок 2

Рисунок 3

Глава 2.3. Описание функционала

**Со стороны API (для клиентской части) было реализовано:**

1) Регистрация, авторизация, выход из системы, проверка токена на валидность, обновление токена - пользователя.

2) Получение всех лайков на те товары, которые пользователь отставил свою реакцию. Поставить лайк на товар (если товар уже был выбран, то лайк автоматически уберется с товара).

3) Получение всех товаров, которые были добавлены в корзину. Удаление товаров из корзины. Уменьшение количества товаров. Добавление товаров (можно добавлять один и тот же товар).

4) Получение всех отзывов на товар, добавление/редактирование /удаления отзыва.

5) Получение среднего рейтинга продукта.

6) Получения всех продуктов.

7) Получение всех категорий.

8) Имеются фильтры, сортировки и неточный поиск для товаров.

**Со стороны API (для CRM клиентской части) было реализовано:**

1) Регистрация, авторизация, выход из системы, смены пароля - администратора.

**2)Добавление, Просмотр, изменение, удаление пользователей.**

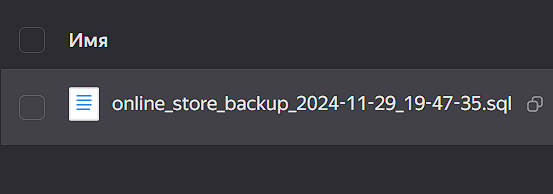
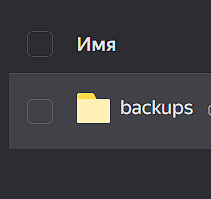
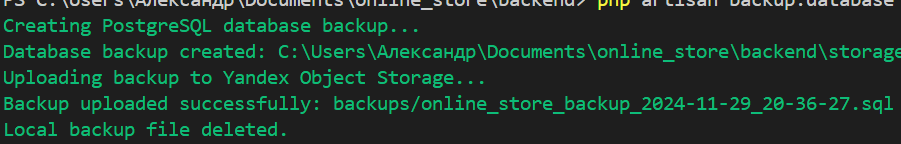
3) **Добавление, Просмотр, изменение, удаление категорий.**

**4) Добавление, Просмотр, изменение, удаление продуктов.**

**5) Экспорт в PDF и WORD: пользователей, продуктов, категорий**

**Дополнительный функционал:**

**1) Резервное копирование данных на жесткий диск раз в 2 недели в облако и на локальном сервере**



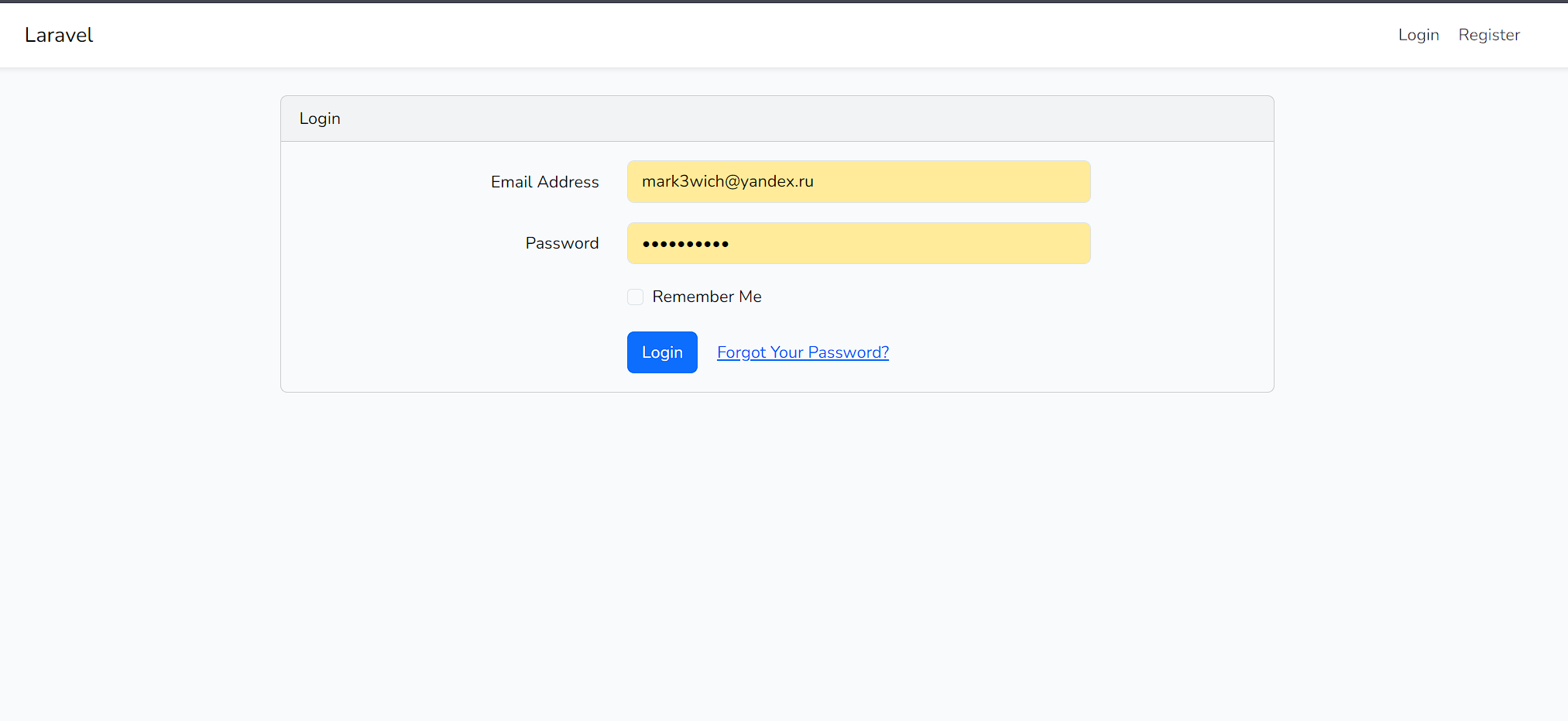
**2) Автоматическая генерация пользователей, отзывов, категорий, и т.д**

**Глав 3 Frontend**

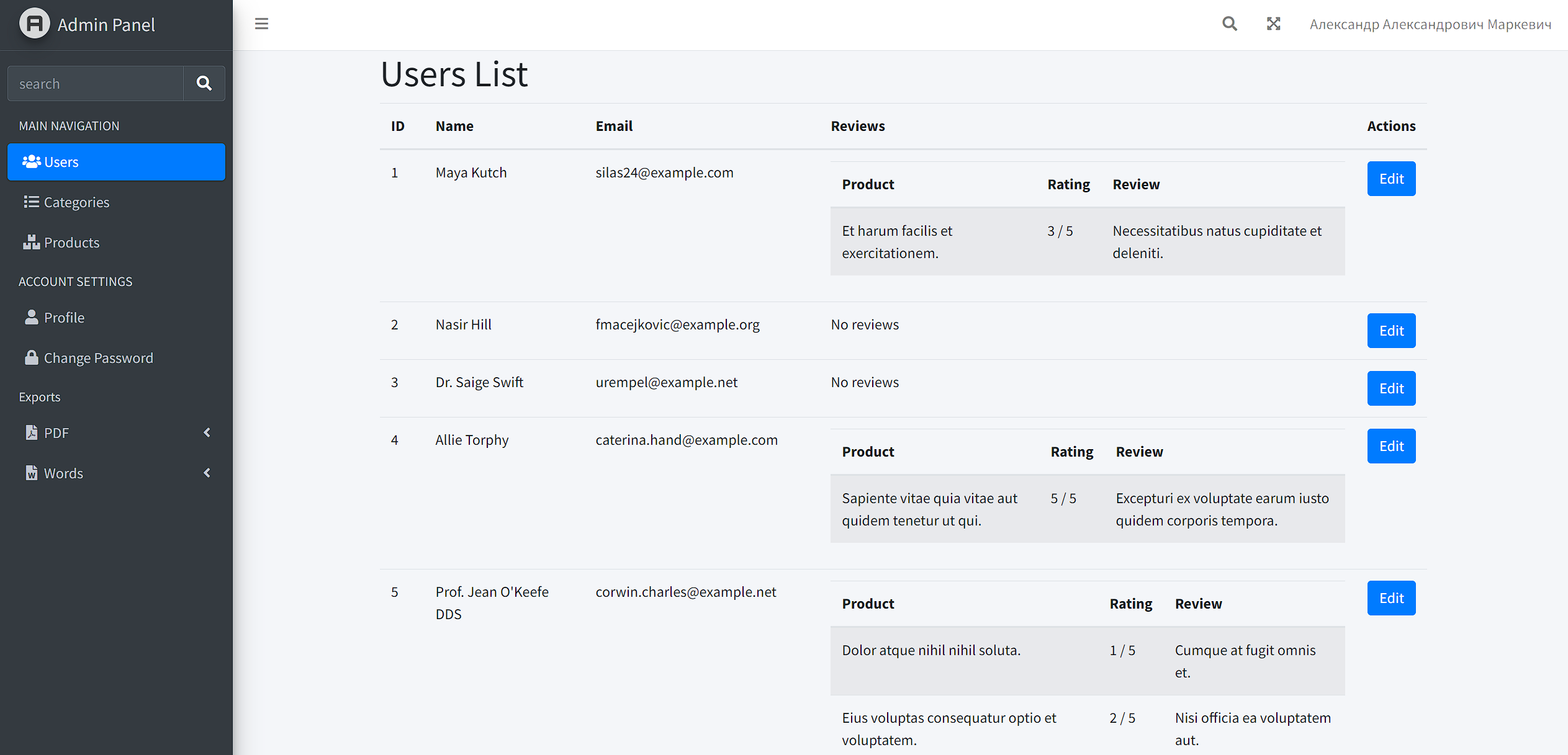
В данной главе будет описываться взаимодействие с клиентской частью приложения.

CMR:

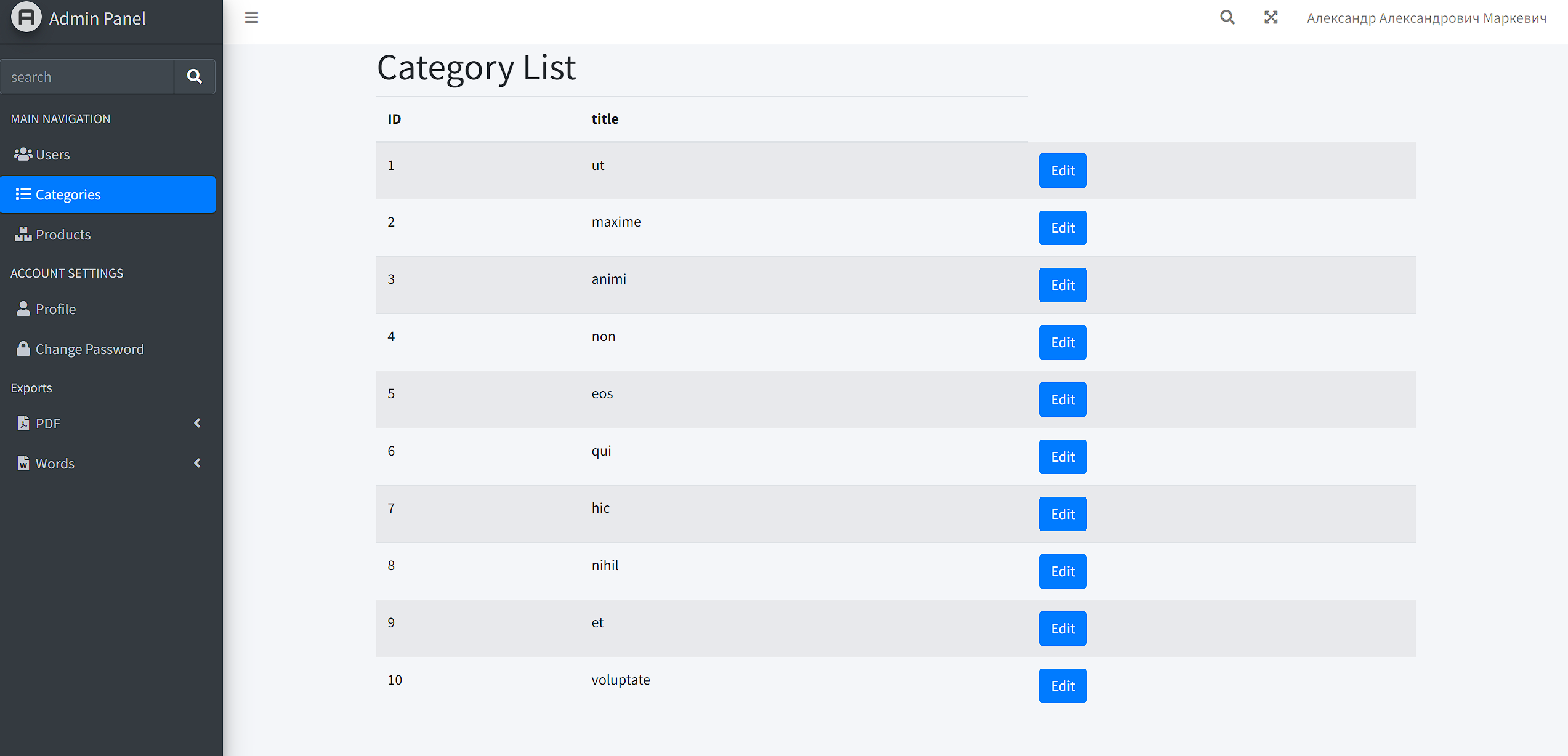
Страница для авторизации и регистрации.

Рисунок 4

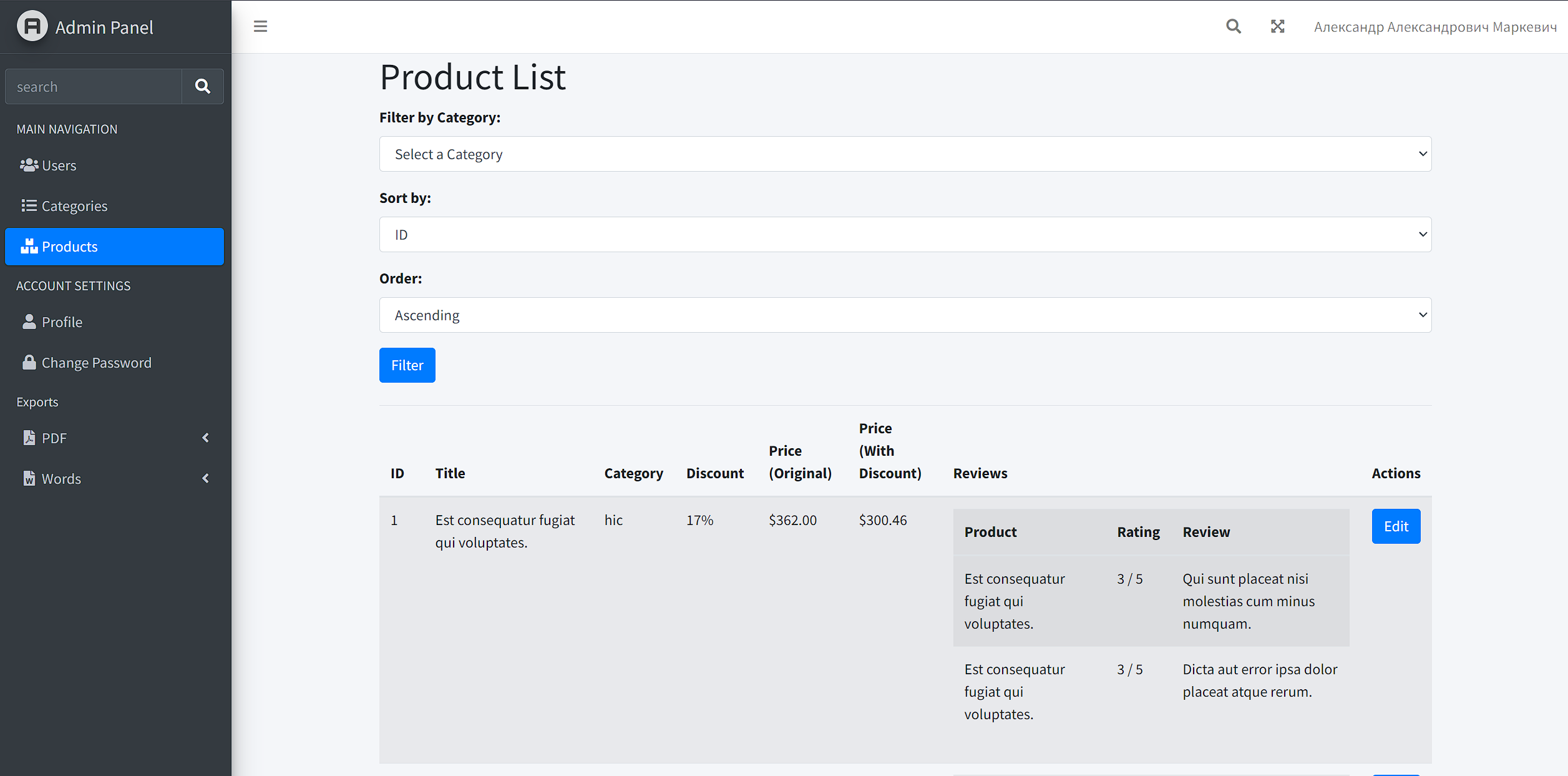
Страница с пользователями:

 Рисунок 5

Страница с категориями:

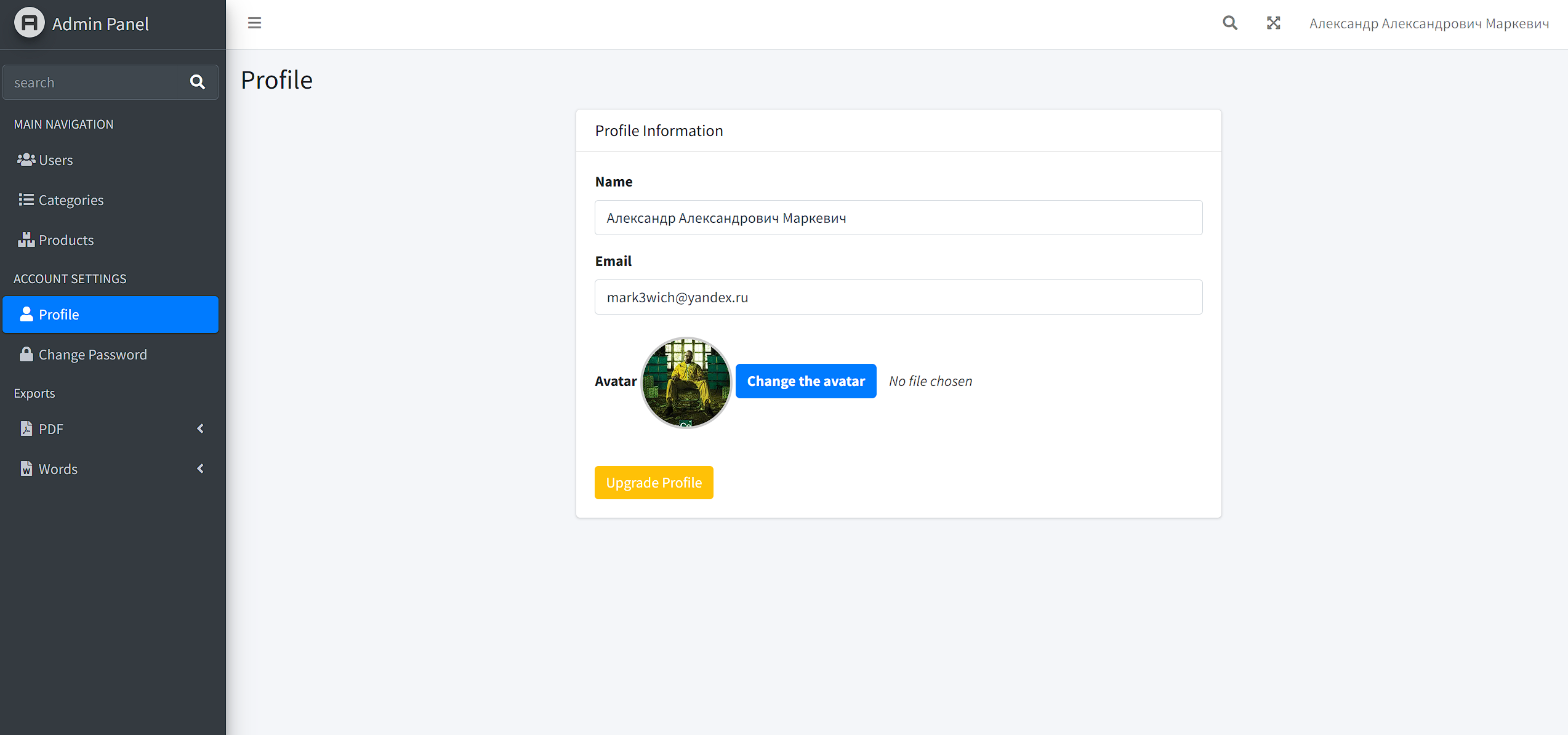
Рисунок 6

Страница с продуктами:

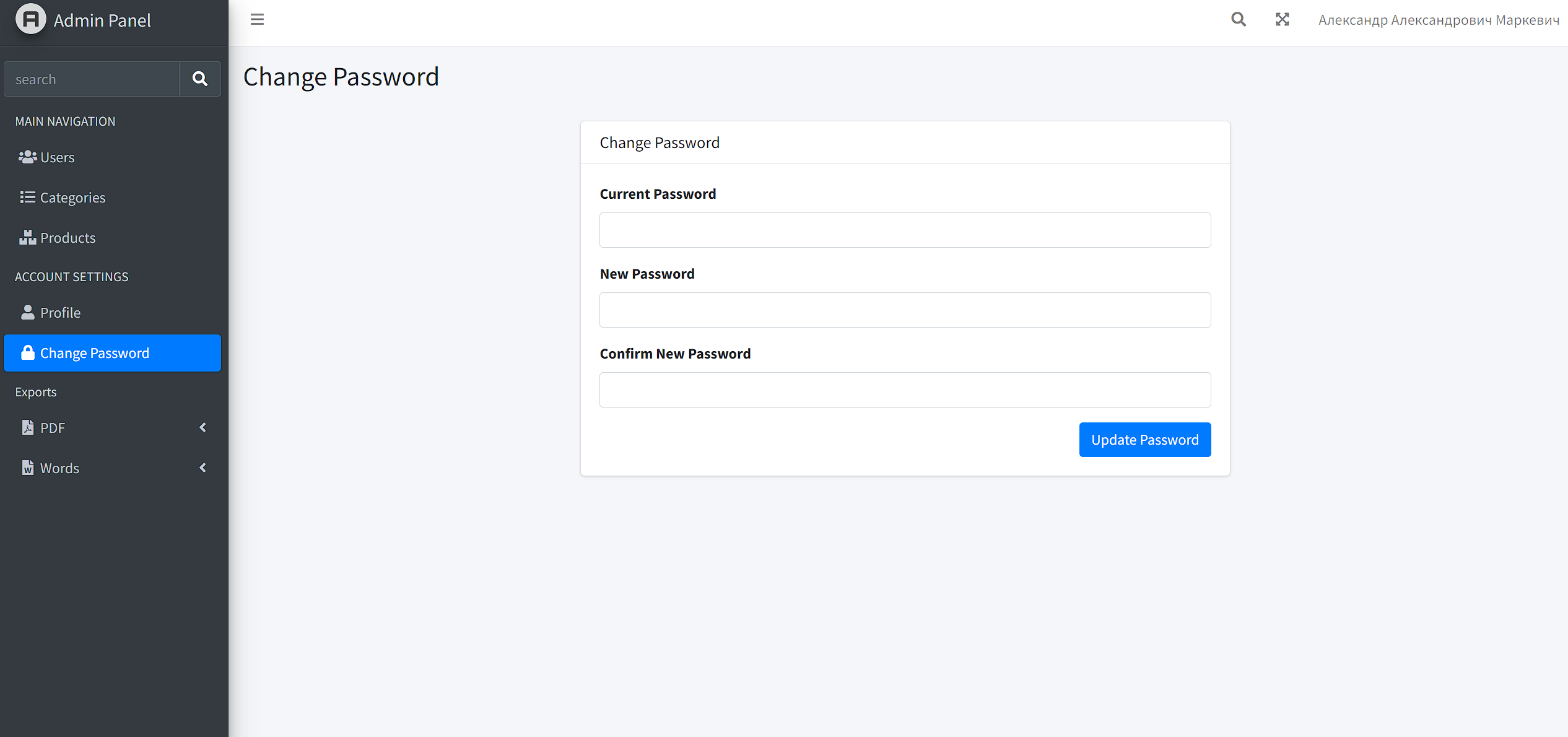
Рисунок 7

Возможны 3 сортировки

Страница с профилем администратора:

Рисунок 8

Страница с изменением пароля:

Рисунок 9

Выбор что экспортировать:

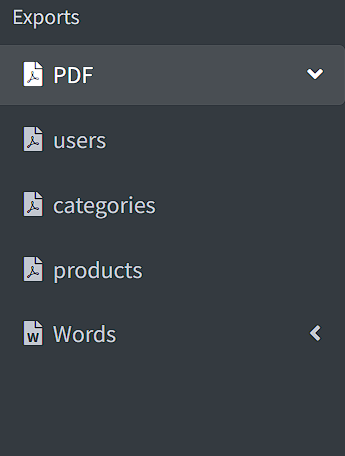
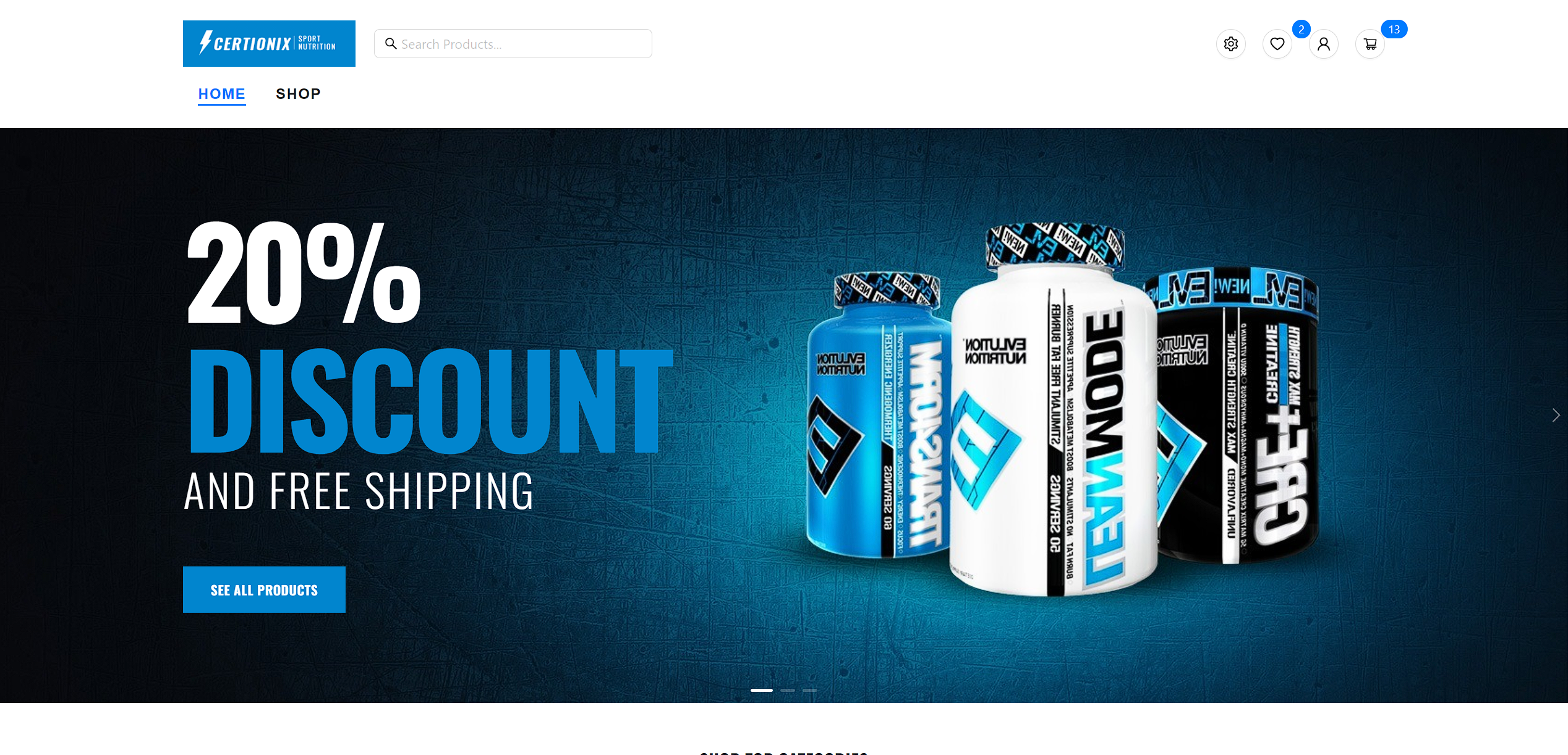


Рисунок 10

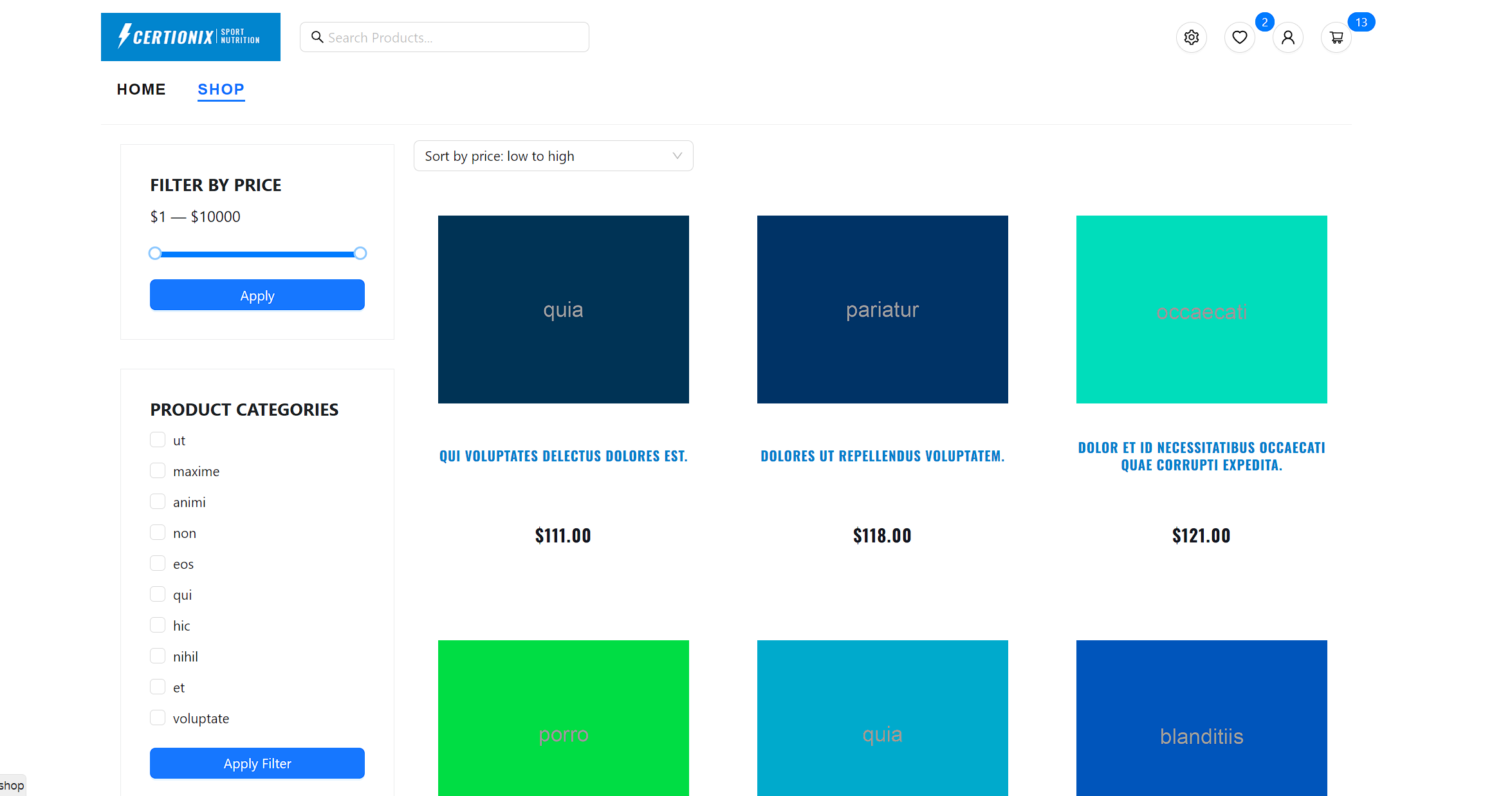
React:

Главная страница:

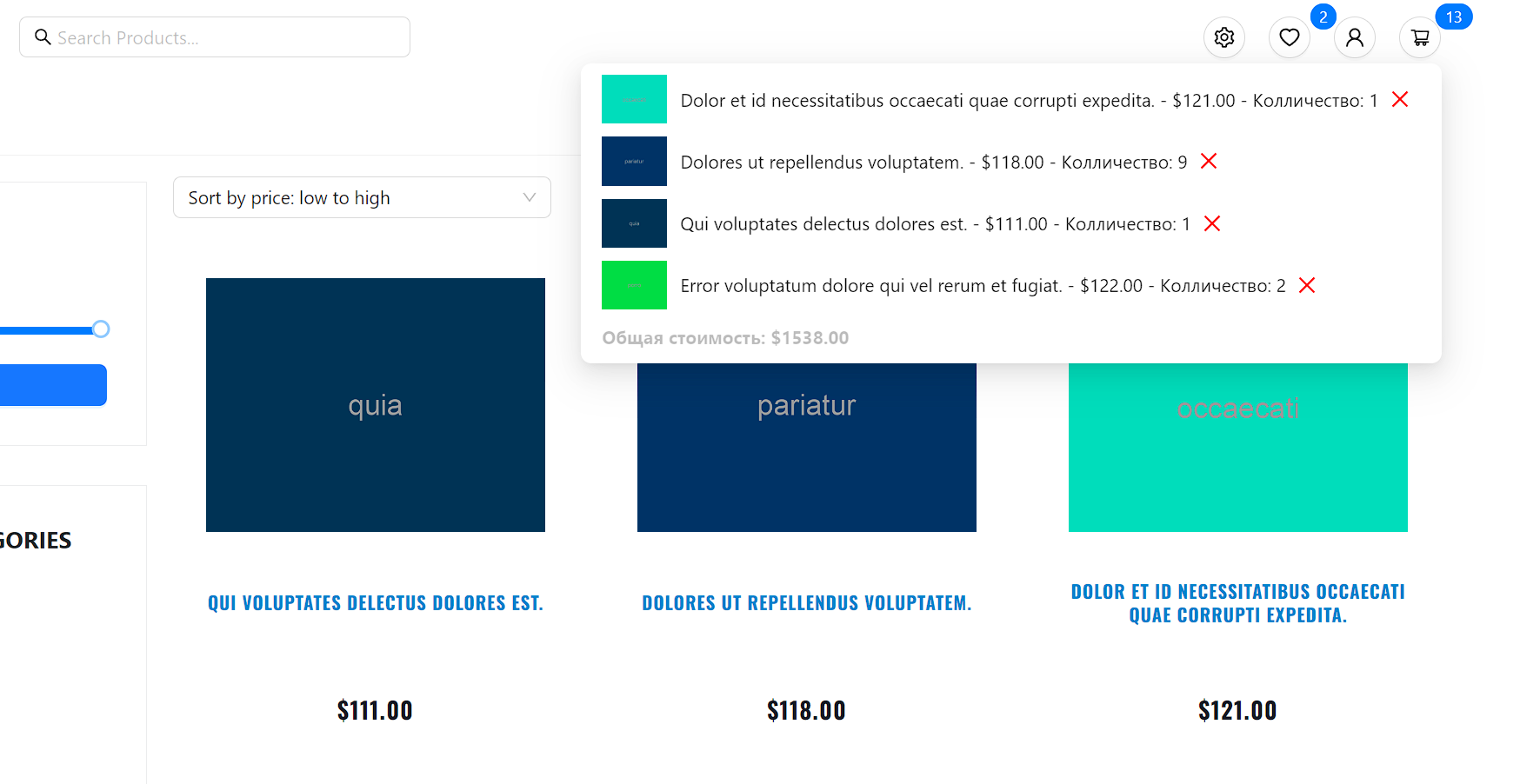
Рисунок 11

Имеется слайдер и верстка снизу

Страница магазина:

Рисунок Рисунок 12

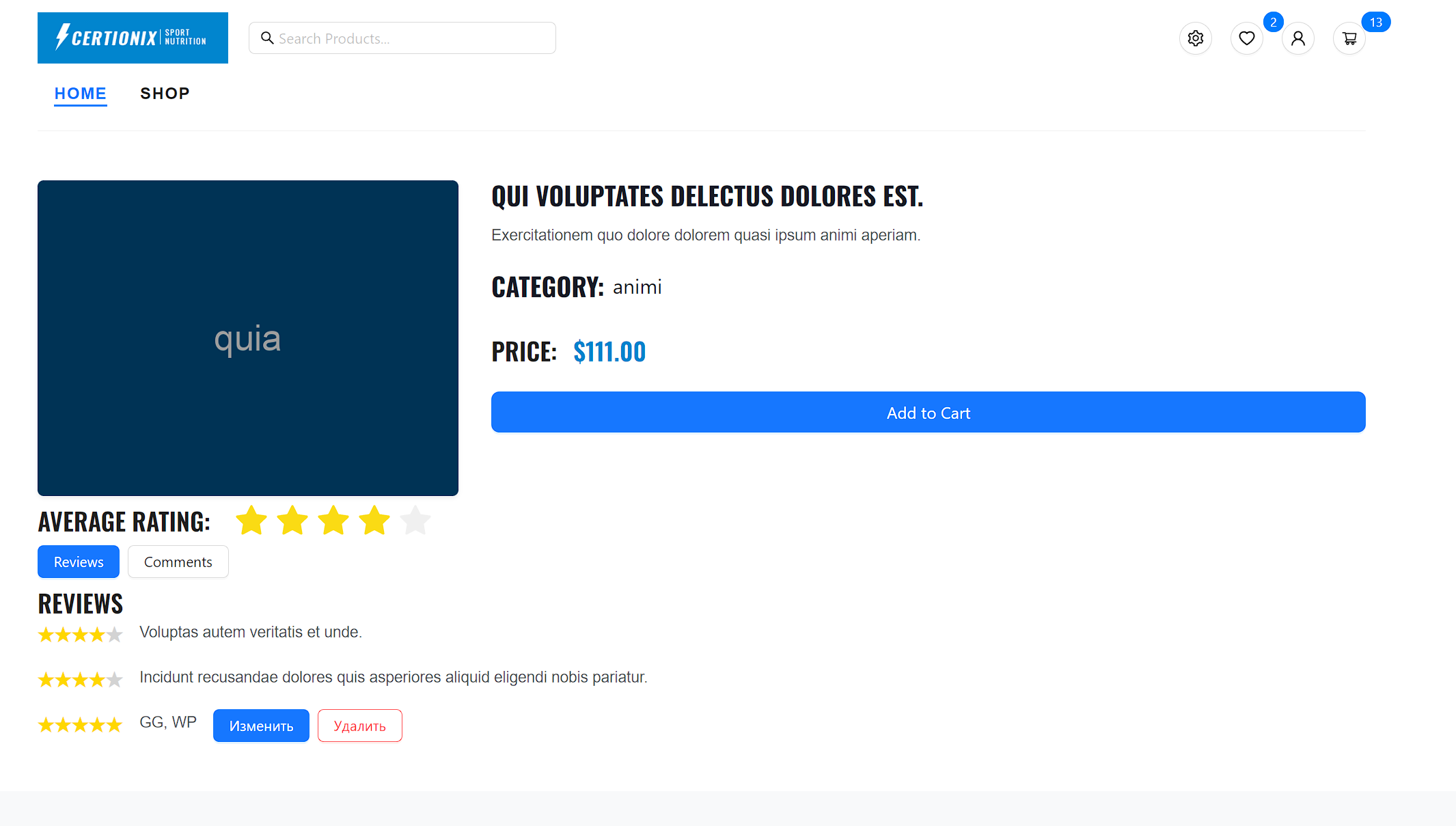
Корзина:

Рисунок 13

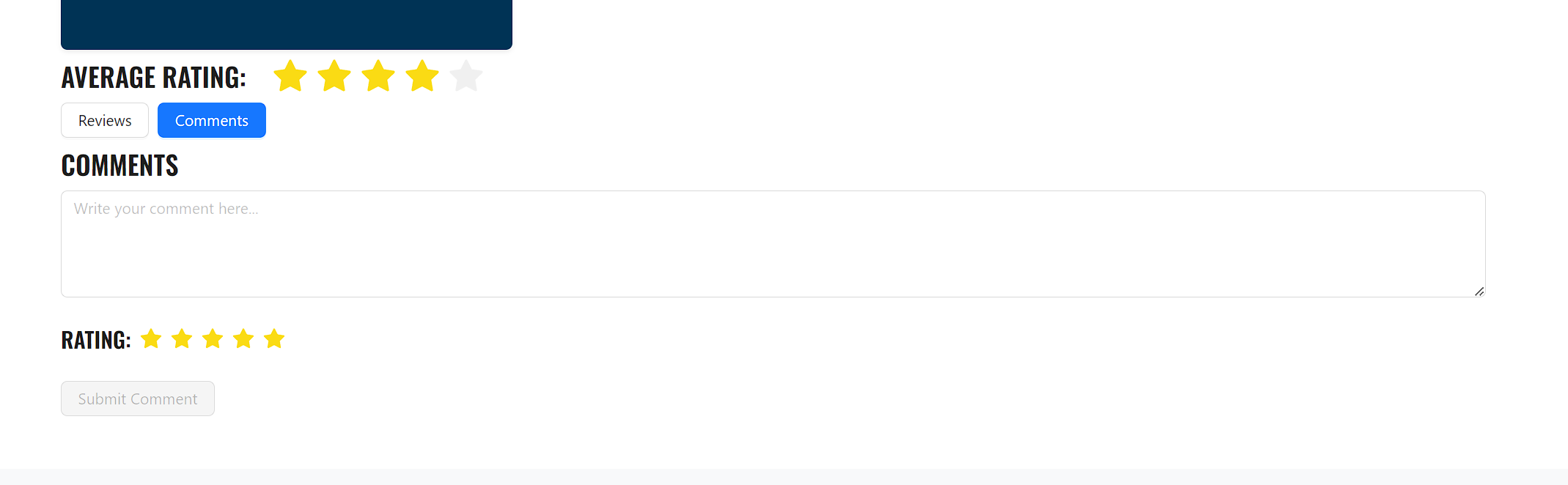
Лайки:

Рисунок 14

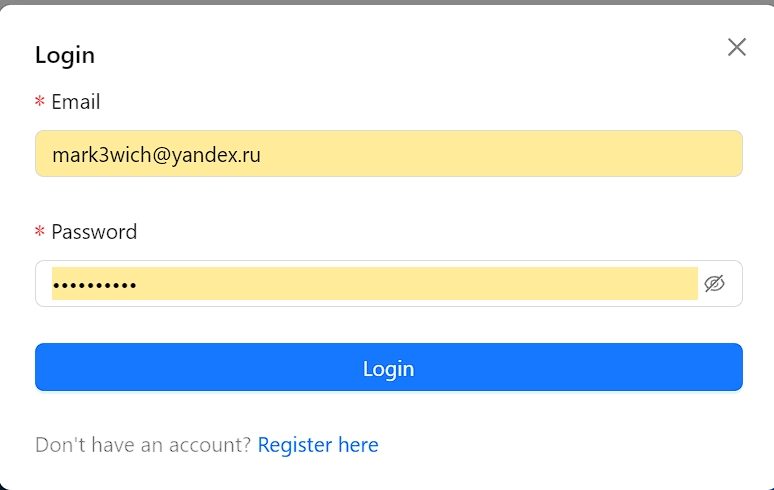
Страница с продуктом:

Рисунок 15

Блок отправки отзыва:

Рисунок 16

Авторизация:

Рисунок 17

Регистрация:

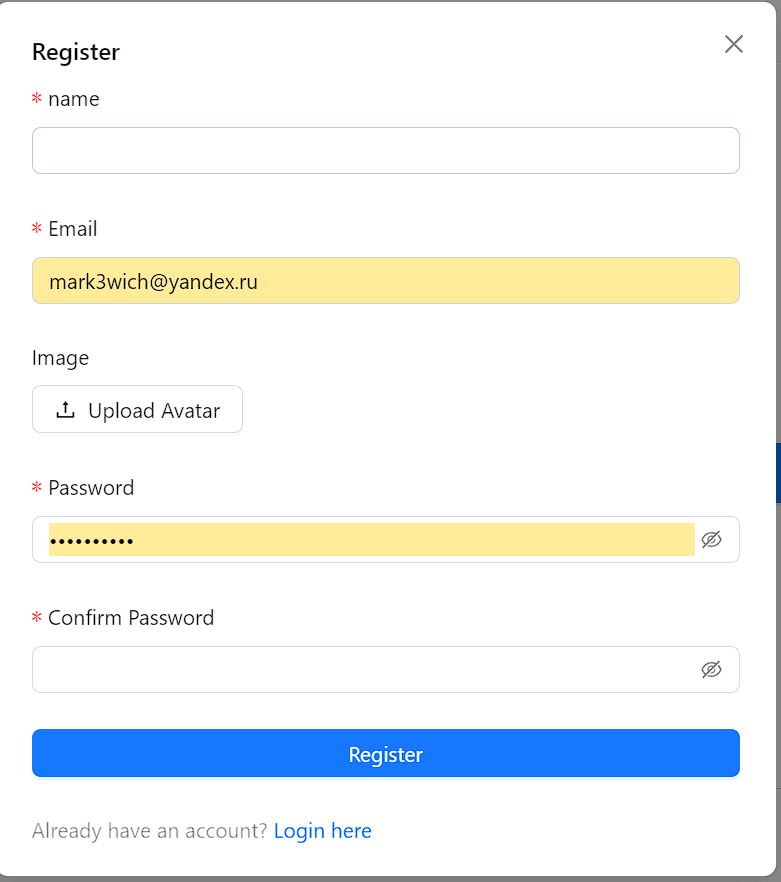


Рисунок 18

**Глава 4 Заключение**

Спроектированы базы данных. Программно реализован прототип интернет-магазина frontend и backend его часть.

**Глава 5 Список источников и литературы**

**1. Документация Laravel** [Электронный ресурс]. Дата обращения:10.11.2024. Режим доступа: <https://laravel.com/docs/11.x/documentation>

2. **Документация React** [Электронный ресурс]. Дата обращения:9.11.2024. Режим доступа: https://react.dev/learn

3. Документация Postgress [Электронный ресурс]. Дата обращения:11.11.2024. Режим доступа: <https://www.postgresql.org/>

**Приложение с кодом:**

Ссылка на github репозиторий: https://github.com/Mark3vich/online\_store