Министерство науки и высшего образования Российской Федерации

Федерального государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

ПРАКТИЧЕСКИЕ РАБОТЫ

ПО «РАЗРАБОТКЕ ПРОГРАММНЫХ МОДУЛЕЙ»

Листов:

|  |  |
| --- | --- |
| Выполнил студент | Проверил преподаватель |
| Группы П50-4-22 | \_\_\_\_\_\_\_\_\_\_\_\_Д.В.Серяк |
| Юрченко Вячеслав Валерьевич | «\_\_\_» \_\_\_\_\_\_\_\_2024 года |

Москва 2024

ПРАКТИЧЕСКАЯ РАБОТА №13

***Цель работы:***

1. **Структура и логика приложения**:
   1. Проект должен быть полноценным веб-приложением, разработанным с использованием React.js.
   2. Должны быть страницы: «Главная», «Каталог», «Избранное», «Корзина», «О товаре», «Заказ».
   3. Header и Footer на каждой странице одинаковые
   4. Страница «Главная»:
      1. Краткая информация о магазине
      2. Форма обратной связи (с отправкой сообщений на почту)
      3. Небольшой список товаров ограниченный до 10 позиций
   5. Страница «Каталог»:
      1. Полный список товаров
      2. Возможность поиск товаров
      3. Возможность фильтрации по категориям (минимум 5)
      4. Возможность добавления товара в «Избранное»
      5. Переход на страницу «О товаре»
   6. Страница «Избранное/Корзина»:
      1. Выгрузка выбранных товаров
      2. Для страницы «Корзина» сделать итоговый подсчет стоимости
   7. Страница «О товаре»:
      1. Полная информация о выбранном товаре (меняется в зависимости от выбранной тематики)
      2. Возможность добавление товара в «Избранное»
      3. Возможность добавление товара в «Корзина»
   8. Страница «Заказ»:
      1. Должна быть форма для оформления заказа (Адрес, ФИО, способ оплаты)
      2. После заполнения всех полей должна появляться Recaptcha (Оформляйте ее по вашему усмотрению):
         1. При успешном прохождении заказ должен подтвердиться и отправиться письмом на почту
         2. При неудачном заполнении, выдать сообщение о неудаче.
2. **Обработка форм**
   1. Проект должен включать формы, созданные с использованием библиотеки React Hook Form. Эти формы должны быть функциональными и корректно обрабатывать ввод пользователя.
3. **Использование Анимаций**:
   1. Проект должен использовать библиотеку Framer Motion для анимаций минимум в проекте использовать 3 анимации
4. **Использование Json-Server**:
   1. Проект должен использовать Json-Server для создания мокового REST API. Полученные данные с сервера должны быть использованы в приложении.
5. **Качество кода**:
   1. Код должен быть чистым, хорошо организованным и легко читаемым. Используйте правила написания чистого кода.
6. **Дизайн и отзывчивость**:
   1. Интерфейс должен быть интуитивно понятным и удобным для пользователя.
   2. Дизайн должен быть современным и привлекательным визуально. Приложение должно корректно отображаться на различных устройствах и разрешениях экрана.(Полностью адаптивен)
   3. Все страницы приложения должны быть выполнены в единой стилистики.
   4. Для создания дизайна вы можете пользоваться и дополнительными библиотеками(Bootstrap, Materilize и т.д)

***Ход работы:***

1. Создать проект с помощью команды **npx create-react-app <название приложения>**
2. Теперь прописываем в терминале команду – npm start
3. Устанавливаем необходимые пакеты и библиотеки.

* npm i -d json-server.
* npm I -d concurrently

Открываем package.json, где прописываем две строчки для дальнейшего запуска json сервера.

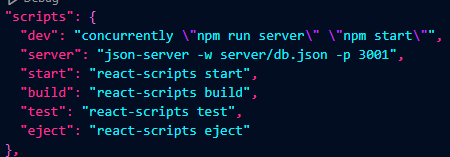


Рисунок 1 – конфигурируем package.json

В server/db.json следует указать в формате json’a элементы, которые будут продаваться в каталоге.

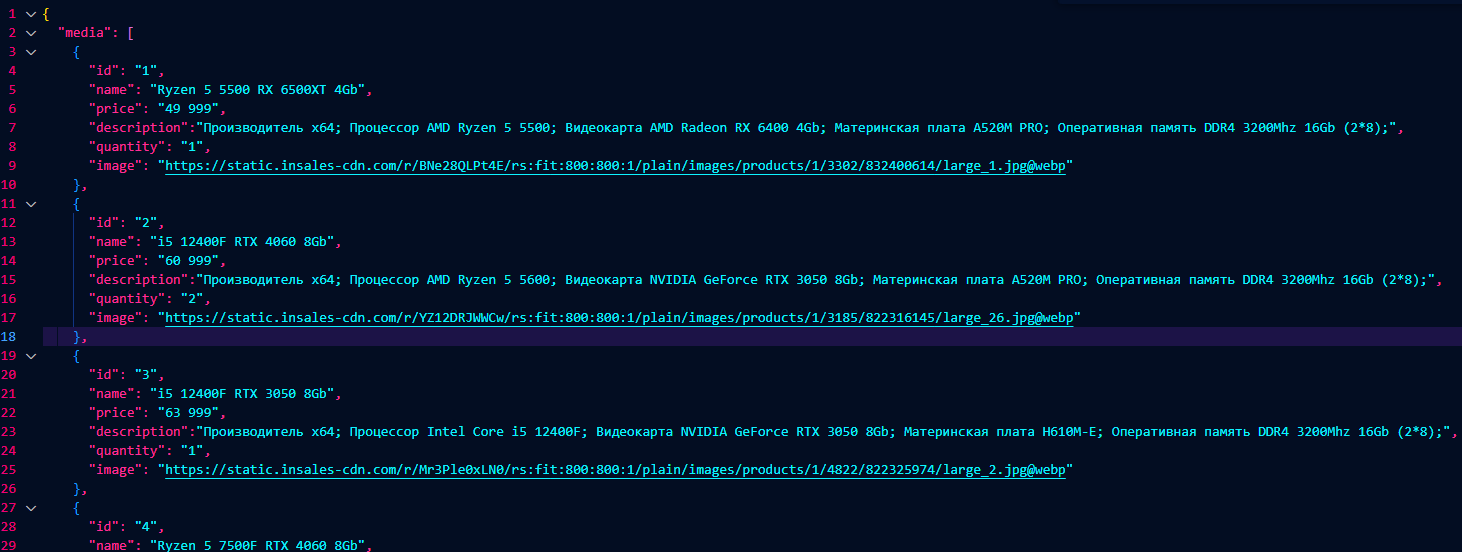


Рисунок 2 – db.json

1. Теперь перейдём к вёрстке всех страниц и их контекстных элементов.

Для страницы каталога указываем эти библиотеки в качестве импортов.

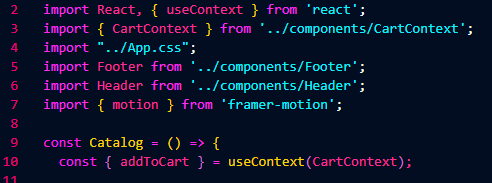


Рисунок 3 – импорты каталога

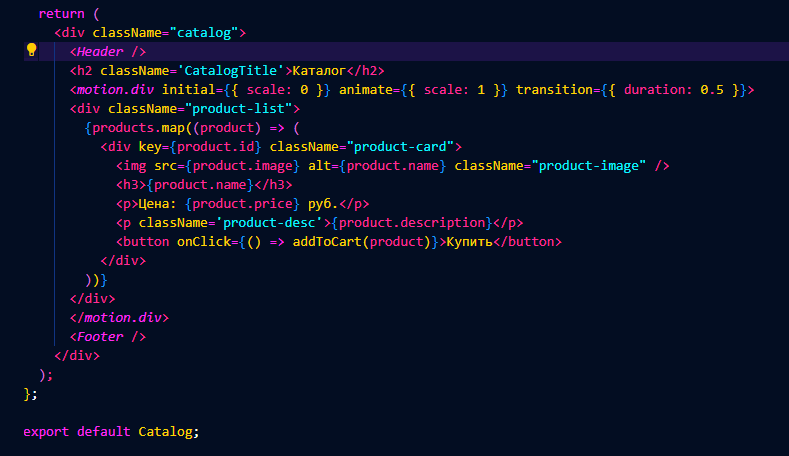


Рисунок 4 – реакт вёрстка страницы каталога

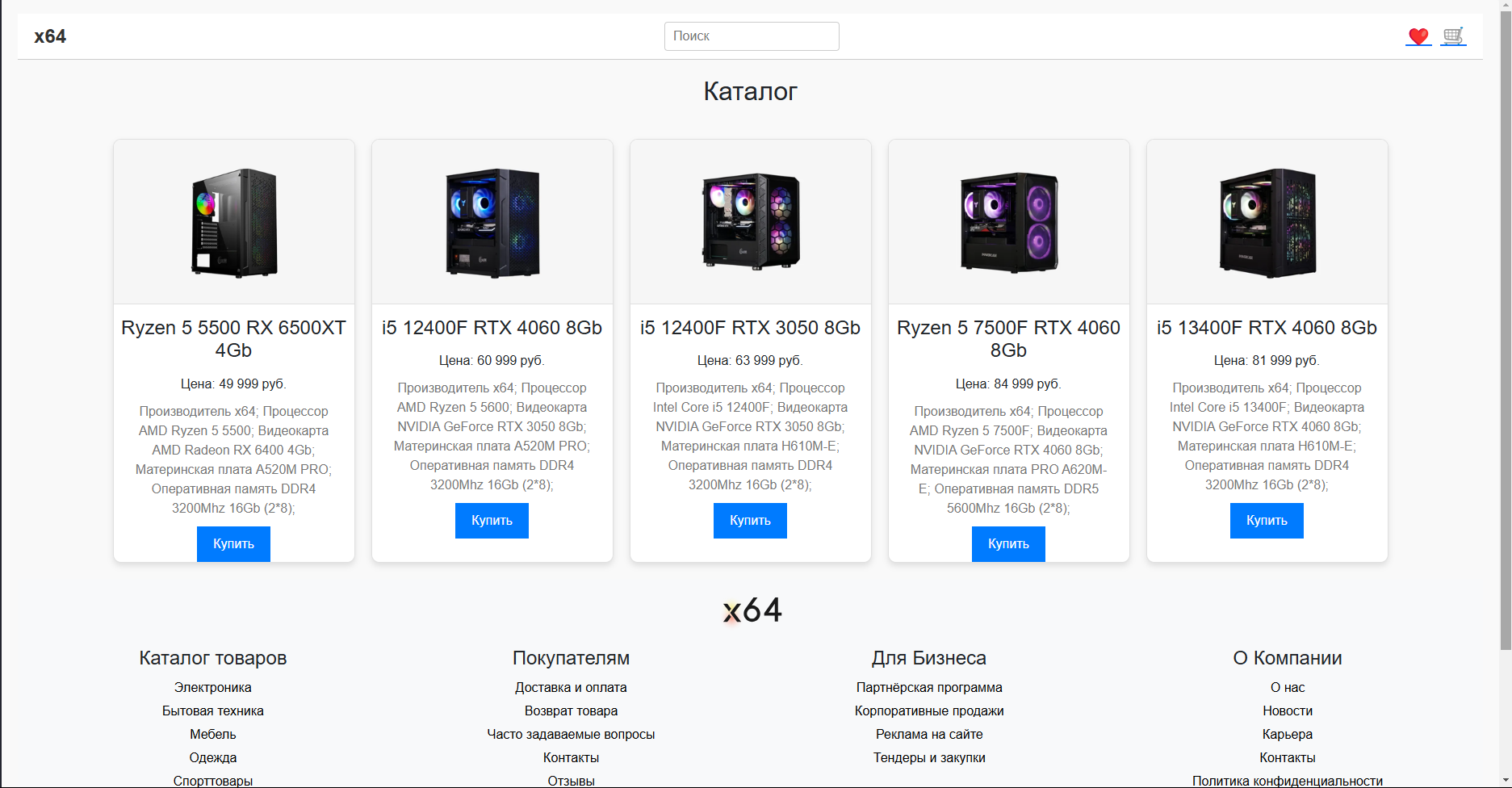


Рисунок 5 - результат

Для корзины указываем данную реакт вёрстку



Рисунок 6 – вёрстка корзины товаров

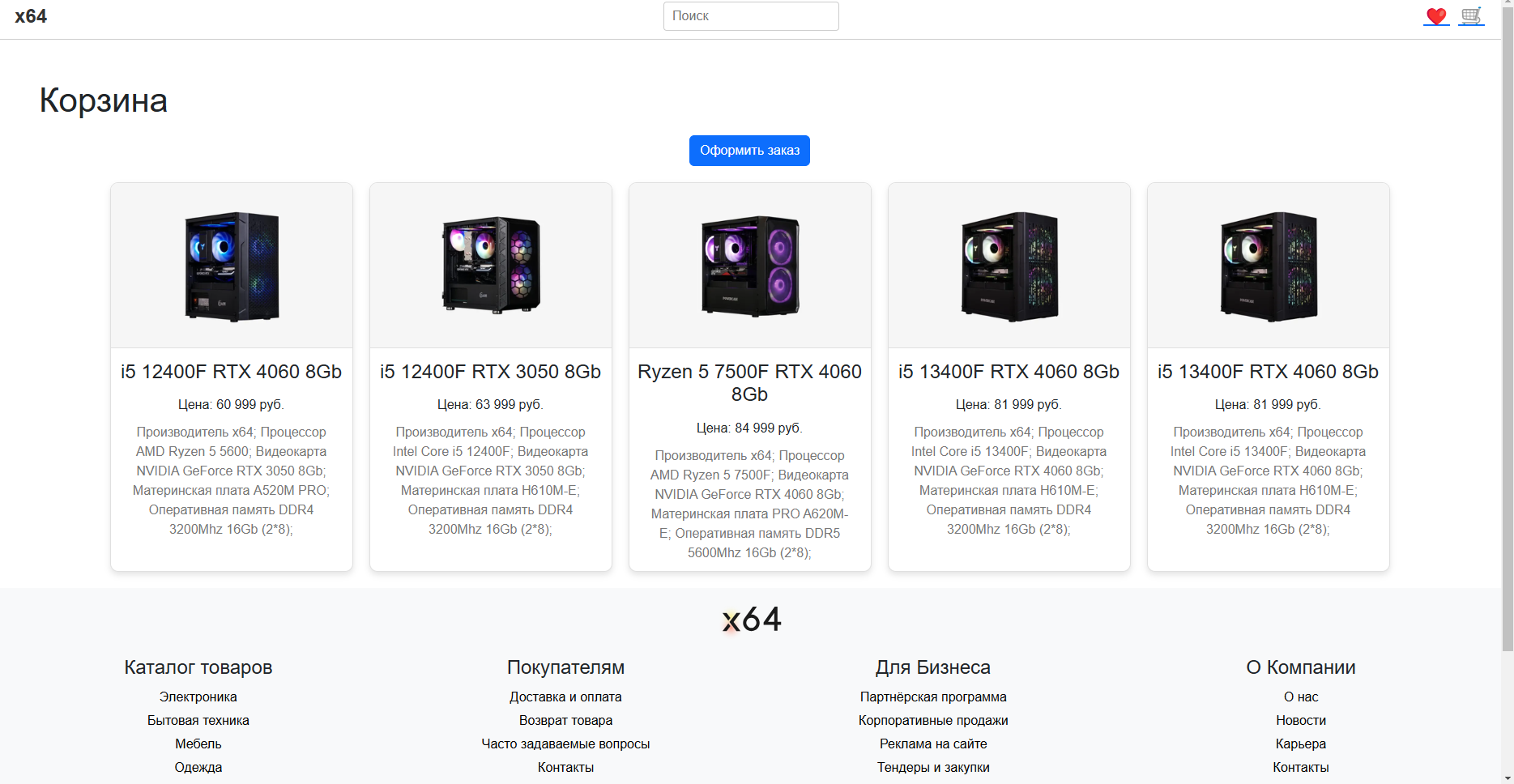


Рисунок 7 - результат

Для страницы избранных товаров используется данная реакт вёрстка с конкретными импортами

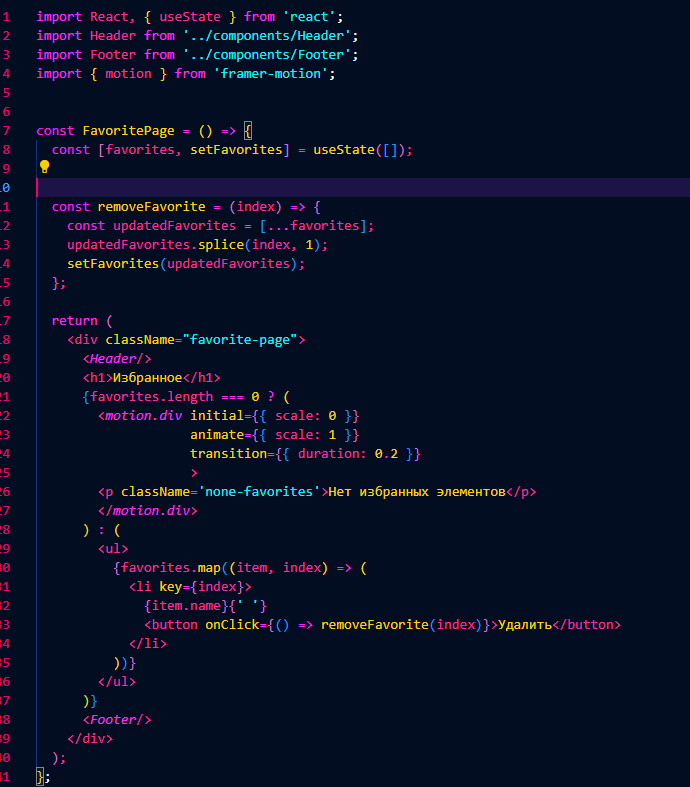


Рисунок 8 – вёрстка избранных товаров

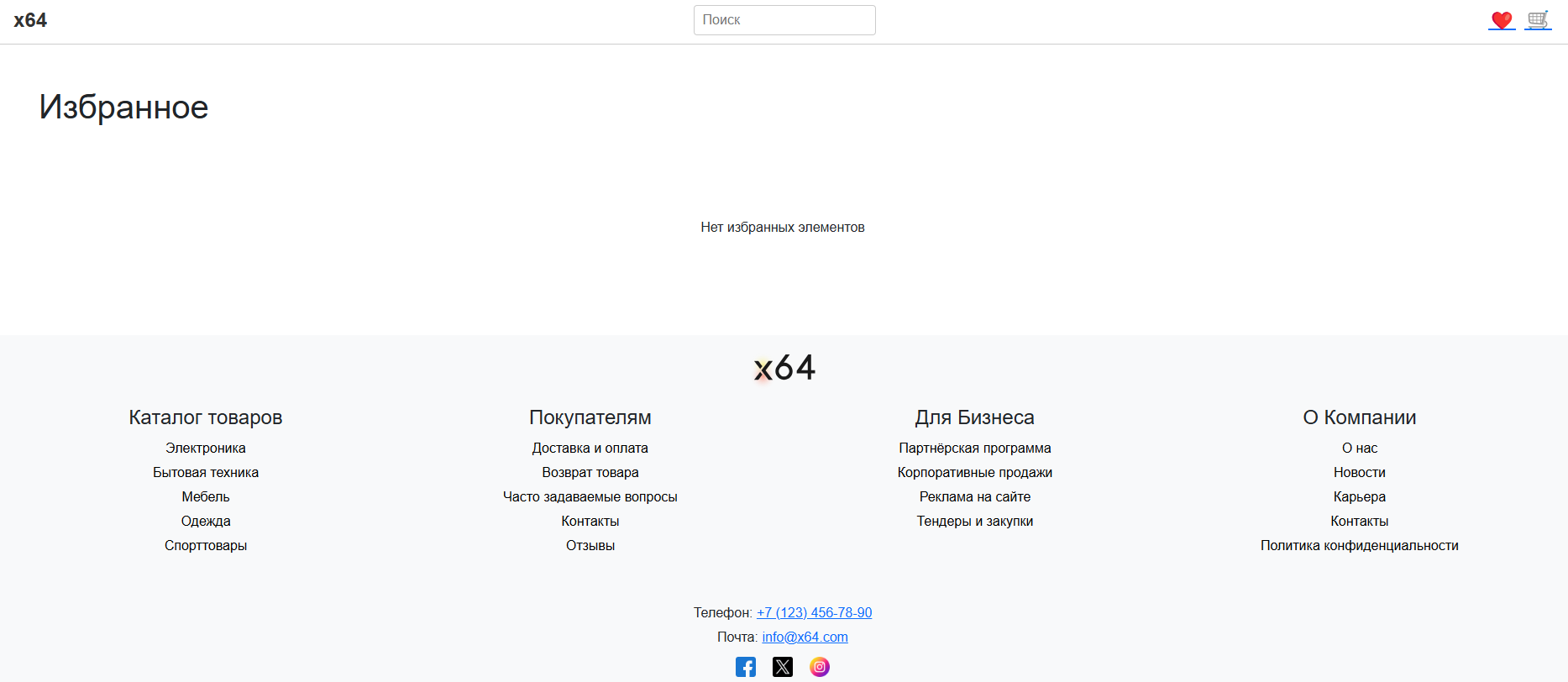


Рисунок 9 - результат

Карточка описания проекта сборок ПК

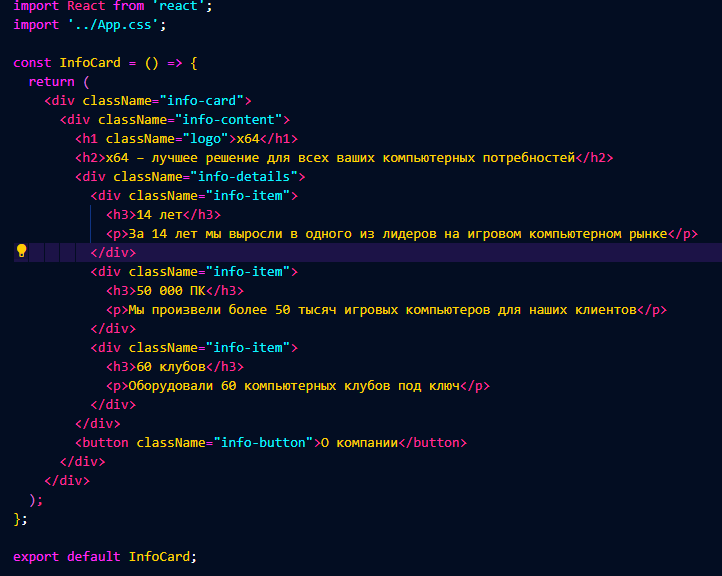


Рисунок 10 – вёрстка карточки о компании

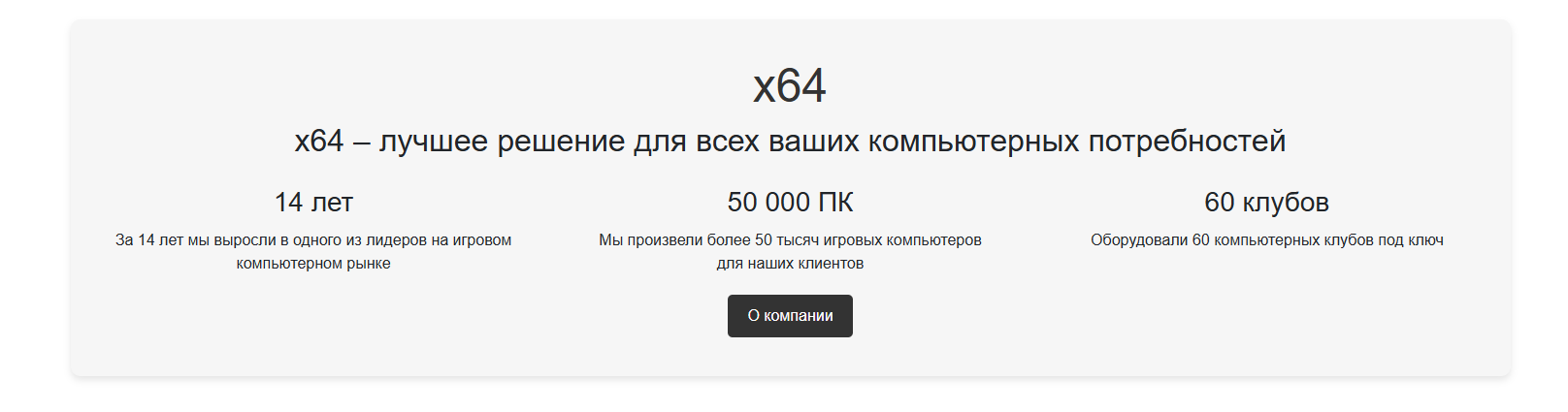


Рисунок 11 – результат

Вёрстка главной страницы выполнена данным образом:



Рисунок 12 – вёрстка главной страницы

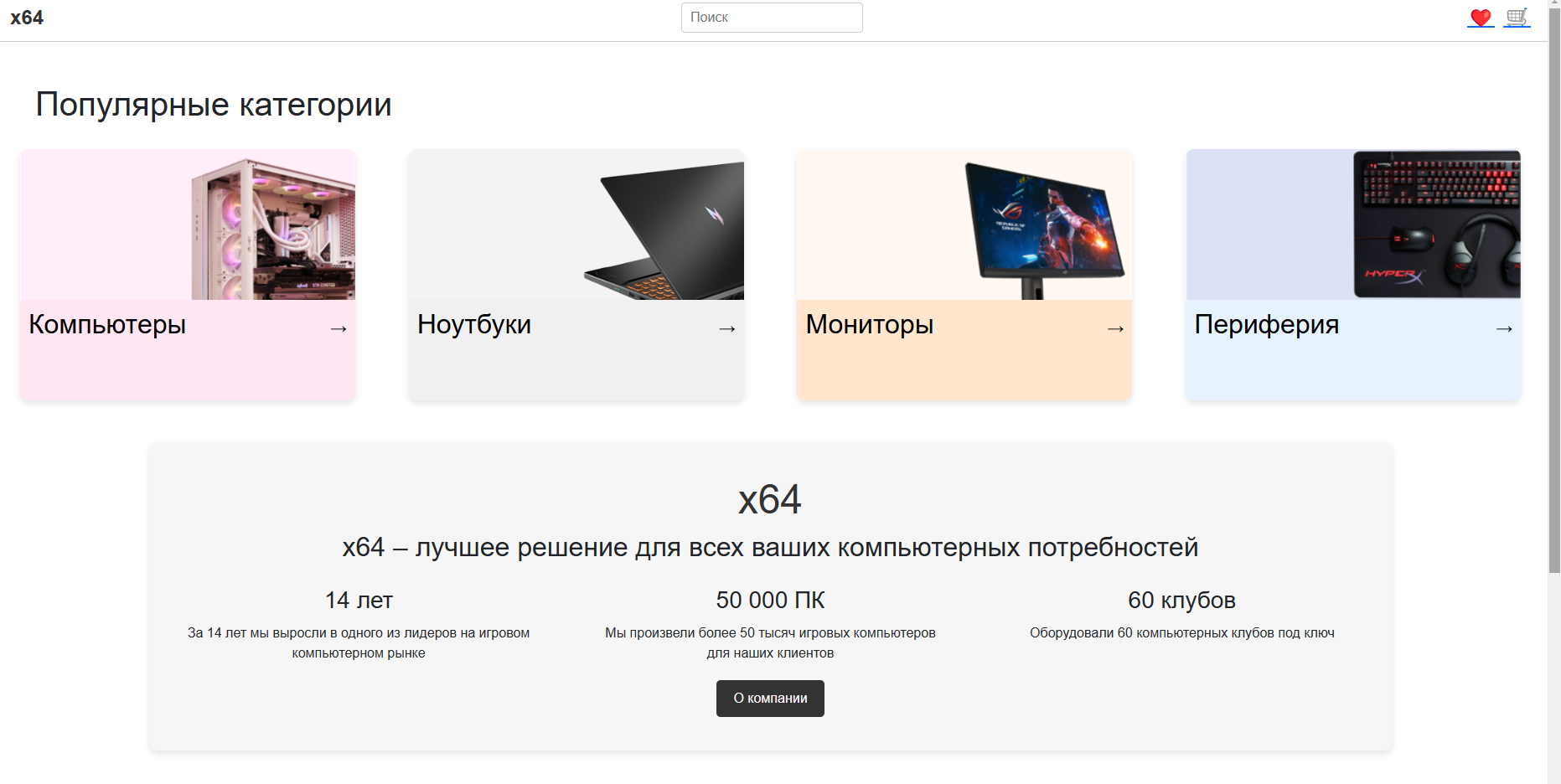


Рисунок 13 – результат (1)

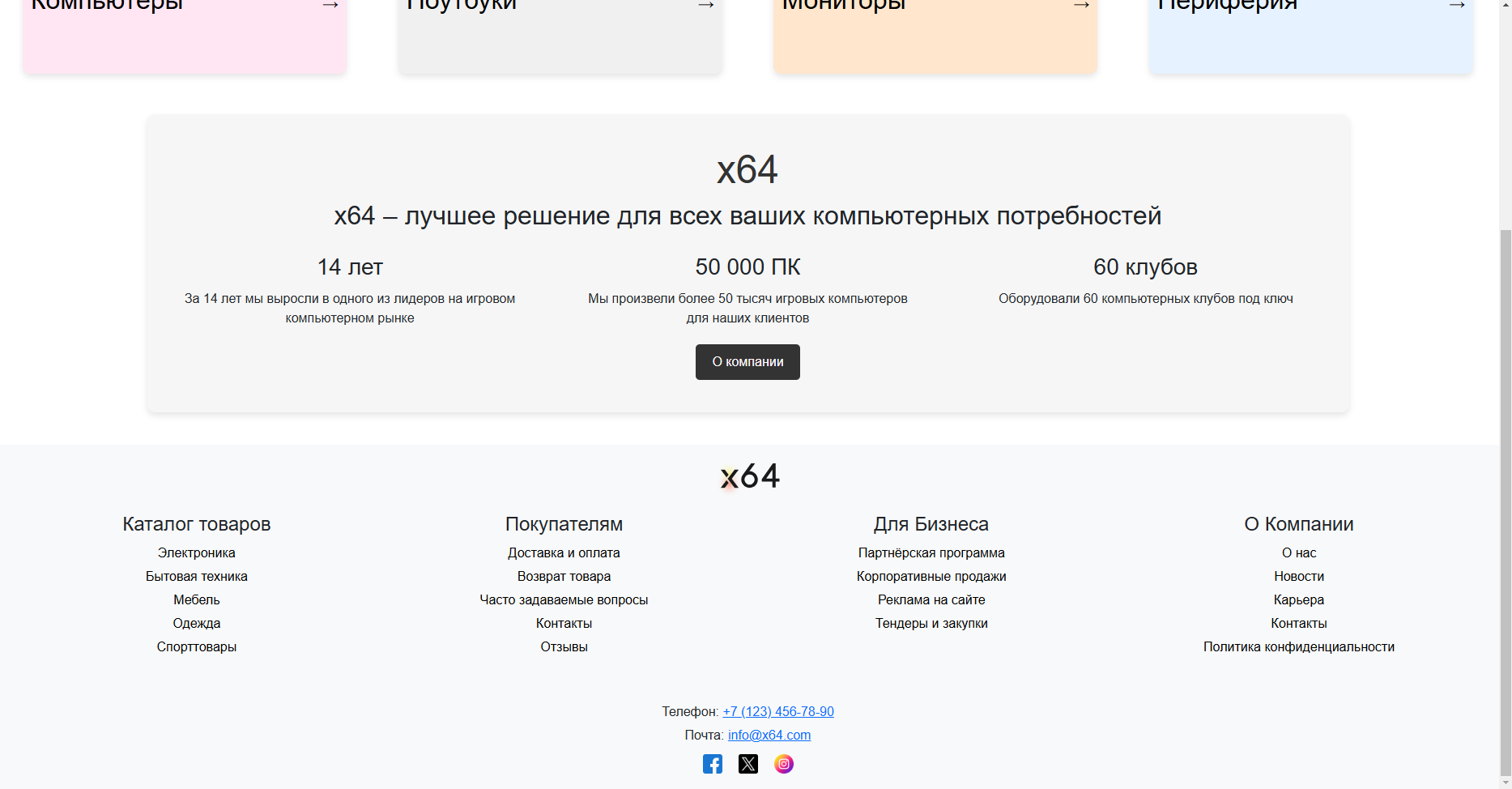


Рисунок 14 – результат (2)

Пример анимации в каталоге:

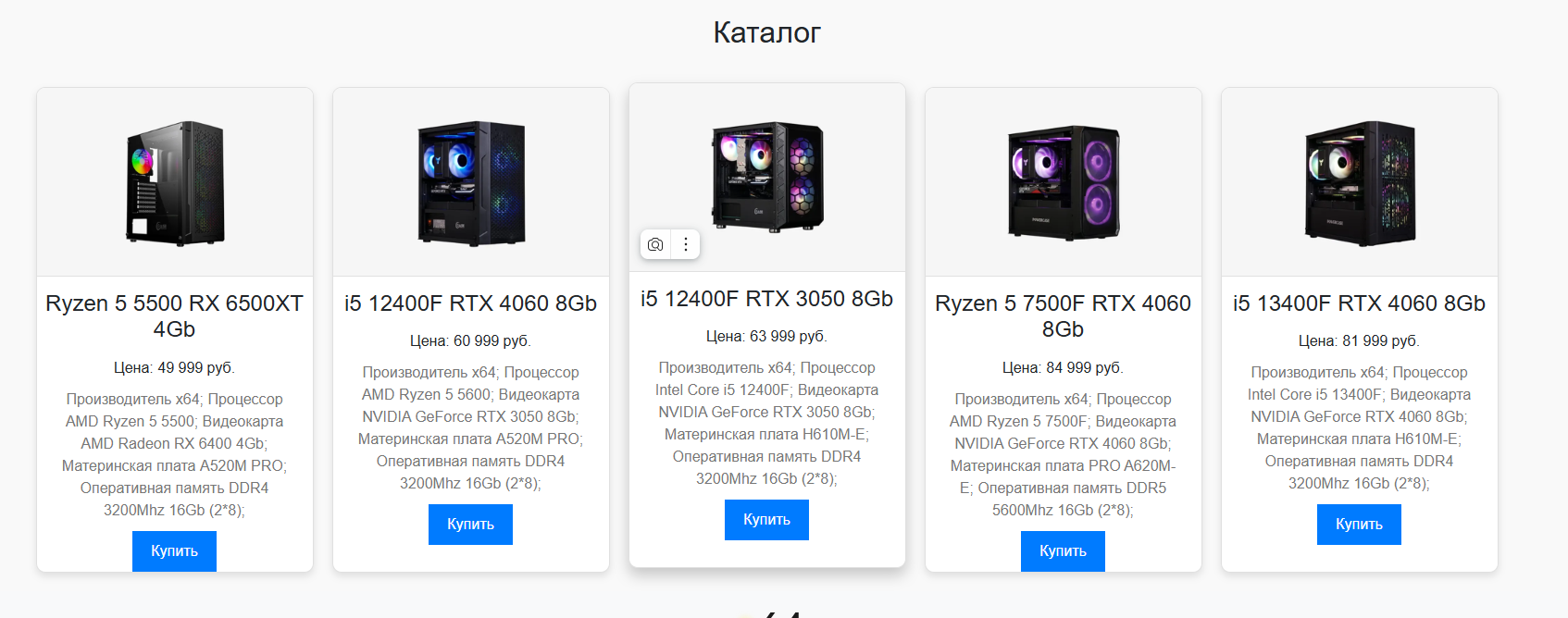


Рисунок 15 – пример анимации карточек в каталоге

Контекстная механика работы корзины:

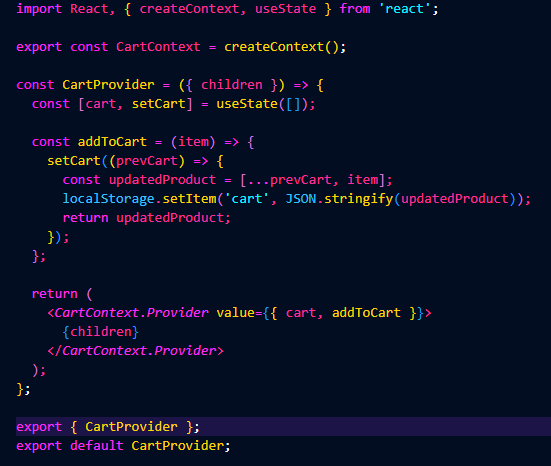


Рисунок 16 – корзина (2)

Контекстная механика работы избранного списка товаров:

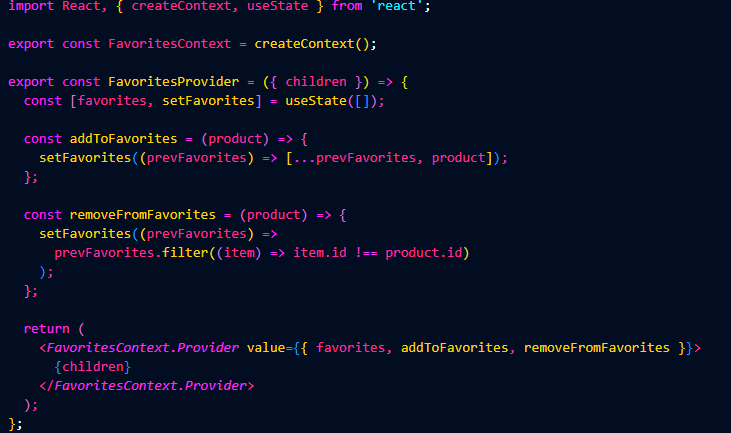


Рисунок 17 – избранные товары (2)

Внутреннее устройство футера страниц:

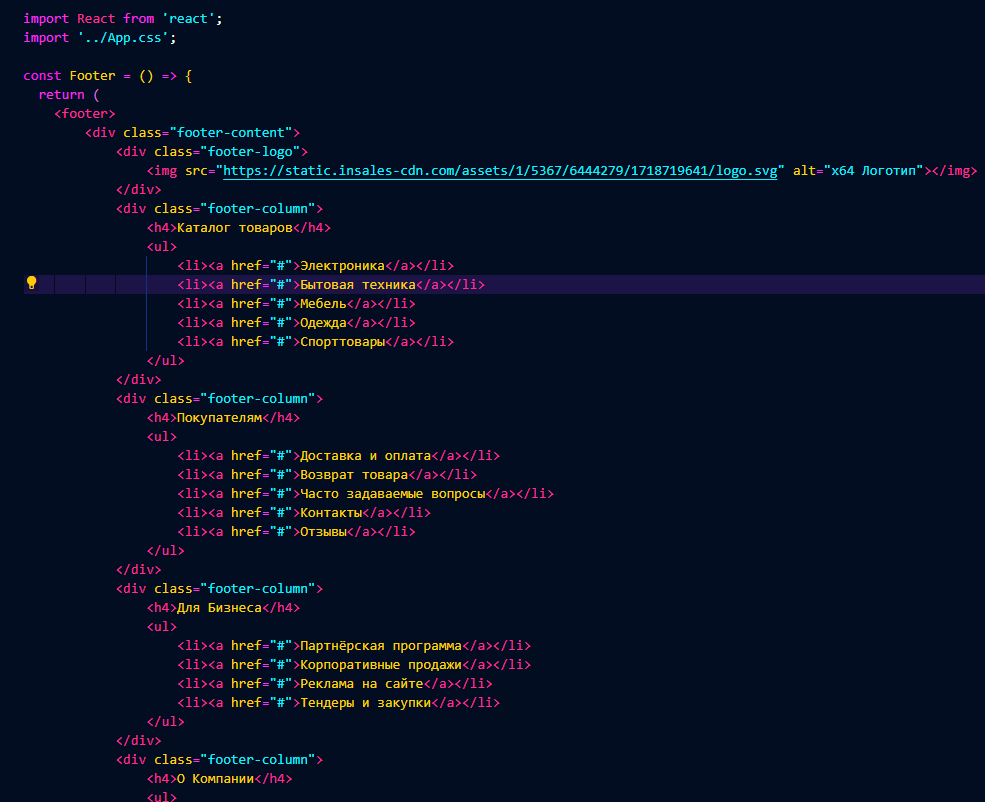


Рисунок 18 – вёрстка футера

Вёрстка шапки страниц сайта:



Рисунок 19 – шапка сайта (1)



Рисунок 20 – шапка сайта (2)

Для более подробного изучения рекомендуется просмотреть репозиторий гитхаба, к дополнению можно ознакомиться со стилями в файле «**App.css**».

***Вывод:*** выполнены некоторые пункты, наличие которых определяет сайт как интернет-магазин. В наличии механика просмотра каталога товаров и добавление их в корзину. Из корзины существует переход на страницу оформления заказа. Весь сайт расписан в одной стилистике и выполнен в таком формате, чтобы было проще пользоваться новичкам. В футере существуют контакты для связи с владельцами сайта. Проект разработан на реакте с использованием json сервера. В наличии сайта так же есть некоторые анимации с помощью использования библиотеки frame-motion.