

Итоговая работа

Задание

Реализовать веб-приложение (сайт), содержащее следующие страницы:

1. Главная страница
 - a. Список товаров (карточек), содержащих изображение, заголовок, описание и кнопку действия: добавить в корзину / удалить из корзины.
 - b. Поле ввода для поиска
 - c. Список категорий (при нажатии выполняющих фильтрацию товаров)
2. Страница товара
 - a. Фотография товара
 - b. Заголовок
 - c. Описание товара
 - d. Кнопка действия (добавить в корзину / удалить из корзины)
3. Страница корзины
 - a. Список товаров (карточек), содержащих изображение, заголовок, описание и кнопку действия: удалить из корзины. При нажатии на кнопку удаления из корзины товар пропадает из списка.
 - b. Кнопка “Оформить заказ”, которая выводит список всех добавленных товаров в консоль.
4. Страница обратной связи
 - a. Поле ввода email
 - b. Поле ввода имя
 - c. Поле ввода текста сообщения
 - d. Кнопка отправить, выводящая в консоль email, имя и текст сообщения в виде объекта.

Критерии оценивания

Работа оценивается по 4х-балльной системе: неудовлетворительно, удовлетворительно, хорошо, отлично.

Оценка напрямую зависит от количества выполненных требований (описаны ниже), а также от качества кода.

Если, например, выполнены все требования на оценку “отлично”, то плохое качество кода может снизить оценку до “хорошо”.

Общие требования

1. Проект должен получать данные из API или внешнего JSON-файла. Получение данных должно работать на GitHub Pages
2. Адаптивная вёрстка: всё должно выглядеть хорошо на любом разрешении экрана, на мобилке (минимальный размер – 375px)
3. Необходимо соблюдать семантическую вёрстку, хороший стиль кода - отступы, названия переменных и т.п.
4. Можно использовать UI-фреймворки (типа Material UI) - но не злоупотреблять. Свои стили в проекте тоже должны быть.

Требования на оценку “удовлетворительно” или “хорошо”

1. Стек: Чистые HTML/CSS/JS или React (на выбор)
2. Свёрстаны 3 страницы задания (без учёта логики): главная страница, форма обратной связи и страница товара (на карточках товаров и на странице товара кнопка “Добавить в корзину” должна присутствовать в вёрстке)
3. При нажатии на элемент товара должен быть выполнен переход на страницу конкретного товара.
4. На странице формы должна быть выполнена валидация с помощью JS: заполнение обязательных полей и проверка корректности email.

Требования на оценку “хорошо” или “отлично”

1. Стек: React
2. Свёрстаны 3 страницы задания (без учёта логики): главная страница, форма обратной связи и страница товара (на карточках товаров и на странице товара кнопка “Добавить в корзину” должна присутствовать в вёрстке)
3. Реализован поиск товаров и фильтрация по категориям на главной странице
4. При нажатии на элемент товара должен быть выполнен переход на страницу конкретного товара.
5. Форма обратной связи должна содержать дополнительные элементы: чекбокс, радиокнопки, загрузку файла.
6. На странице формы должна быть выполнена валидация с помощью JS (валидация всех реализованных полей)

Требования на оценку “отлично”

1. Стек: React
2. Свёрстаны все 4 страницы задания (без учёта логики).
3. Реализован поиск товаров и фильтрация по категориям на главной странице
4. При нажатии на элемент товара должен быть выполнен переход на страницу конкретного товара.
5. Форма обратной связи должна содержать дополнительные элементы: чекбокс, радиокнопки, загрузку файла.
6. На странице формы должна быть выполнена валидация с помощью JS (валидация всех реализованных полей)
7. Реализован механизм добавления и удаления из корзины.

Будет плюсом

1. Реализация страницы авторизации. Без авторизации доступ на остальные страницы проекта закрыт
2. Реализация добавления нескольких одинаковых товаров в корзину. После нажатия кнопки “Добавить в корзину” должен появляться счётчик и две кнопки, с помощью которых можно увеличить/уменьшить количество товаров в корзине. Страница корзины должна так же отражать количество одинаковых добавленных товаров.
3. Реализация пагинации: разбиение всех товаров на страницы, если на странице с учётом фильтров поиска и категории появляется более, чем 20 товаров.
4. Использование Redux для хранения общих данных приложения (возможно, тема оформления, данные пользователя и т.п.). Возможно в связке с localStorage.

Размещение проекта.

1. Код проекта должен быть размещён на GitHub в отдельном от домашних работ репозитории.
2. Проект необходимо разместить на GitHub Pages (по желанию - на любом другом сервере или хостинге, главное условие - возможность посмотреть проект всем по ссылке).