Тестовое задание для React Developer

Необходимо разработать react приложение "Управление заказами", которое будет использоваться для отображения, редактирования и управления заказами клиентов. Использовать следующее: - Язык разработки typescript - Менеджер пакетов yarn - Библиотеку компонентов Material UI - Инструмент получения данных Redux Toolkit RTK Query - Библиотека для форм react-hook-form - Любой подходящий fake API

Далее описание требуемого функционала

Авторизация и аутентификация: Создайте форму входа. Используйте фиктивные данные для авторизации (например, username: admin, password: admin). После успешного входа пользователь должен быть перенаправлен на основную страницу приложения.

Универсальный компонент для табличного отображения данных: Разработайте компонент для отображения данных в табличном виде с использованием компонентов Material UI (Table, TableRow, TableCell и других). Компонент должен поддерживать: - Сортировку по колонкам. - Фильтрацию данных. - Пагинацию. - Раскрывающиеся строки для отображения дополнительной информации (например, подробности заказа или вложенные элементы). - Редактирование данных внутри таблицы. Компонент должен быть максимально универсальным, чтобы его можно было использовать для отображения различных наборов данных в других частях приложения.

Управление заказами: Используйте разработанный компонент для отображения списка заказов. Таблица заказов должна поддерживать сортировку, фильтрацию, пагинацию и раскрытие строк для отображения детальной информации о заказе, включая продукты, количество и цену.

Создание компонента "Фильтры": Разработайте компонент для фильтрации списка заказов. Компонент должен включать несколько параметров: - Фильтр по статусу заказа (например, "Ожидает оплаты", "Отправлен", "Доставлен"). - Фильтр по дате создания заказа (диапазон дат). - Поле поиска для фильтрации по имени клиента или номеру заказа. - Кнопка "Сбросить фильтры", которая возвращает таблицу в исходное состояние.

Манипуляции с данными в модальных окнах: Любые операции по добавлению, редактированию и удалению данных должны производиться через модальные окна. Для создания и редактирования заказов используйте react-hook-form для управления и валидации форм.

Интеграция с REST API: Используйте RTK Query для взаимодействия с REST API. Приложение должно поддерживать следующие запросы: - Получение списка заказов. - Получение детальной информации о заказе. - Обновление информации о заказе. - Удаление заказа. Реализуйте правильную обработку состояний загрузки, ошибок и успешных запросов (используя соответствующие сообщения и спиннеры).

Уведомления: После успешного выполнения любой операции (создание, редактирование, удаление) должно отображаться уведомление об успешности операции (используйте Material UI Snackbar).

Результат нужно загрузить в гитхаб и развернуть на одном из бесплатных хостингов.