

Работа с данными, полученными из API

Задача: разработать приложение, выводящее посты из API с возможностью взаимодействия с ними.

Требования:

- Использовать React.js, Redux-toolkit
- Использовать компонентный подход для создания приложения
- Приложение должно быть отзывчивым и хорошо выглядеть на разных устройствах (мобильные телефоны, планшеты, настольные компьютеры)
- Работа со стилями на свое усмотрение (без использования библиотек)

В качестве API использовать сервис [JSONPlaceholder - Free Fake REST API \(typicode.com\)](https://typicode.com/).

Описание приложения:

«Посты»

Метод API: /posts

Необходимо вывести все посты из API постранично, по умолчанию 10 штук на странице. Должна быть возможность изменить количество выводимых постов (10, 20, 50, 100, все). При перезагрузке изменения должны сохраниться. (использовать localStorage)

Каждый пост состоит из названия, имени пользователя (*метод API: /users*), его добавившего, основного текста, кнопок «Комментарии», «Редактировать», «Удалить» и «В избранное» (сделать иконками), а также чекбокса, позволяющего выделять несколько постов.

В API есть методы добавления, удаления и редактирования, их нужно использовать. Данные на сервере при этом не изменятся, но на странице изменения должны отображаться.

Описание кнопок

«Комментарии»: кнопка должна иметь два состояния. По умолчанию кнопка неактивна. При нажатии на неё, она становится активной, а под постом появляются комментарии. У каждого комментария отображается имя отправителя, его e-mail и текст самого комментария. При повторном нажатии кнопки, блок с комментариями скрывается, кнопка снова становится неактивной. *Метод API: /comments*.

«Редактировать»: при нажатии на кнопку должна появиться возможность изменить текст поста, его название и пользователя, от чьего имени опубликован пост. В режиме редактирования пользователь может сохранить или же отменить внесённые изменения.

«Удалить»: появляется всплывающее окно с подтверждением удаления. При утвердительном ответе пост убирается со страницы, при отрицательном ничего не происходит.

«В избранное»: пост должен как-либо выделиться внешне. Если нажать на кнопку повторно – пост удаляется из избранных.

Чекбокс: если хотя бы один пост отмечен чекбоксом, появляются две кнопки (к примеру, снизу) – «Удалить» и «В избранное». При нажатии на каждую из них должно всплыть окно с подтверждением. После утвердительного ответа происходит соответствующее массовое действие (удалить или добавить в избранное все отмеченные посты). После отрицательного – ничего не происходит, чекбоксы при этом остаются активными.

Фильтры и сортировка

Сверху должны присутствовать три фильтра: по названию поста, по имени пользователя, по нахождению в списке избранных. Фильтр по имени пользователя реализовать в виде выпадающего списка.

Пользователь должен иметь возможность сортировать посты в обе стороны по ID, названию, имени пользователя и по нахождению в списке избранных.

Также, в верхней части необходимо разместить кнопку добавления нового поста. При нажатии на неё появляется всплывающее окно, в котором пользователь может указать название нового поста, его текст, а также выбрать пользователя, от чьего имени пост будет размещён. При сохранении пост добавляется на страницу.

Оценка задания

Задание будет оцениваться по следующим критериям:

- Качество кода и его оформление (читаемость, структурированность, использование современных подходов и фреймворков)
- Работа с API (правильность запросов, обработка ошибок)
- Функциональность и соответствие требованиям задания
- Дизайн и пользовательский интерфейс (выбор цветовой схемы, компоновка элементов, адаптивность)

Успехов в выполнении задания!