

Ретроспектива развития redux взаимодействия между фронтендом и бекендом

Панфилов Александр

Обо мне



- 6 лет в IT разработке, последние 3 года на позиции фронтенд разработчика
- Любимые технологии: React, Next, Nest, CI/CD, MongoDB, MySQL

О чем доклад

- Что такое redux и какие проблемы он решает?
- Этапы развития redux:
 - Темные времена
 - Война за выживание
 - Ренессанс
 - Рассвет
- Дальнейшее развитие redux
- Выводы

Что такое redux

Redux — это state менеджер глобального состояния приложения. Он позволяет следить за состоянием приложения и перерисовывать его в тех местах, где было изменено состояние

Middleware

- saga
- thunk

Темные века

- Рабочий процесс включал работу с тремя файлами: actions, reducer, saga|thunk

Проблемы redux:

- Много однотипного кода, который приходилось писать вручную;
- Нет единого подхода в обработке апи и обрывом запроса (abort signal);
- Тяжело масштабировать проект (последнее относится сугубо ко мне);

Война за выживание

- Переосмысливание работы с экшенами, рождение redux-actions;
- Появление хуков useDispatch, useSelector;
- Начало использования typescript.

Проблемы

- Плохая типизация redux и redux-actions;
- Конкуренция со стороны mobx и ему подобных state менеджеров.

Ренессанс

- Redux переписывают на typescript;
- Thunk окончательно вытесняет saga и входит в состав redux;
- Появление slice и adapter;
- Создание своих селекторов.

Проблемы

- Отсутствие единообразия работы по api с бекендом.

Рассвет

- Появление rtk query и решение всех проблем по работе апи с бекендом.

Проблемы

- Нельзя модифицировать кешированные данные (об этом заявляют сами разработчики);
- Нет четкой структуры файлов (сугубо мое мнение).

Развитие redux в дальнейшем

- Улучшение работы с кешированными данными;
- Сделают гайд или рекомендации для построения четкой структуры файлов;
- Возможность работы с серверными компонентами и экшенами.

Спасибо за внимание



Блог



Канал на дзене



Канал в телеграме