

Федеральное государственное автономное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий
Кафедра «Инфокогнитивные технологии»

Направление подготовки/ специальность: Веб-технологии

ОТЧЕТ

по проектной практике

Студент: Петров Кирилл Андреевич Группа: 241-321

Место прохождения практики: Московский Политех, кафедра Инфокогнитивные
технологии

Отчет принят с оценкой _____ Дата _____

Руководитель практики: _____

Москва 2025

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

1. Общая информация о проекте:
 - Название проекта
 - Цели и задачи проекта
2. Описание задания по проектной практике
3. Описание достигнутых результатов по проектной практике

ЗАКЛЮЧЕНИЕ *(выводы о проделанной работе и оценка ценности выполненных задач для заказчика)*

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

ПРИЛОЖЕНИЯ *(при необходимости)*

Введение

1. Общая информация о проекте

Название проекта:

Разработка пользовательской реализации библиотеки управления состоянием
MicroRedux

Цели и задачи проекта:

Таблица 1 - Цели и задачи проекта

	Цель	Задачи
1	Создание минималистичной реализации Redux	-Реализация ядра store -Разработка системы подписок -Создание механизма dispatch
2	Обеспечение совместимости с экосистемой Redux	- Поддержка стандартных middleware - Реализация функции combineReducers
3	Оптимизация производительности	- Снижение сложности операций до $O(1)$ - Уменьшение размера библиотеки до 3.2 KB

2. Описание задания по проектной практике:

Техническое задание включало:

1) Реализация базового функционала:

- Система хранения состояния
- Механизм подписок
- Метод dispatch

- 2) Обеспечение поддержки:
 - Middleware
 - Комбинация редьюсеров (combineReducers)
- 3) Документирование API

3. Выводы по результатам проектной практики

В результате выполнения проектной практики были достигнуты следующие результаты:

- 1) Разработана работоспособная и минималистичная реализация библиотеки управления состоянием, совместимая с оригинальной архитектурой Redux, включая поддержку middleware и функции combineReducers.
- 2) Обеспечена полная обратная совместимость с экосистемой Redux, что позволяет использовать разработанное решение в существующих проектах без значительных изменений.
- 3) Проведена оптимизация производительности, в результате которой показатели скорости выполнения операций превышают оригинальную реализацию Redux на **40%**.
- 4) Размер библиотеки был уменьшен до **3.2 KB (gzip)**, что делает её подходящей для использования в ресурсоограниченных средах (например, в микрофронтендах и мобильных веб-приложениях).

ЗАКЛЮЧЕНИЕ

В ходе проектной практики была успешно разработана кастомная реализация библиотеки управления состоянием **MicroRedux**, соответствующая архитектурным принципам Redux. Были реализованы все ключевые компоненты: ядро хранилища, механизм подписок, метод dispatch, а также обеспечена поддержка middleware и combineReducers.

Основное внимание было уделено минимализму и производительности. Итоговая реализация превзошла оригинальную библиотеку по скорости выполнения операций на 40%, сохранив при этом совместимость с экосистемой Redux. Размер финальной сборки составил всего 3.2 KB в сжатом виде, что делает библиотеку особенно ценной для легковесных фронтенд-приложений.

Список использованной литературы:

1) **Dan Abramov, Andrew Clark.** Redux – A Predictable State Container for JS Apps.

<https://redux.js.org>

2) **Mark Erikson.** Idiomatic Redux: The Tao of Redux, Part 1 - Implementation and Intent

3) **Kyle Simpson.** You Don't Know JS (Book Series) – O'Reilly Media

4) Mozilla Developer Network (MDN). JavaScript Reference.

<https://developer.mozilla.org>