

## КОНТАКТЫ







## ТЕХНОЛОГИИ

React

GIT GIT

✓ Rest API/Axios

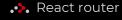
🏟 React-Native 💲 Storybook







Js JavaScript



Radix

N NextJS

MaterialUI

React hook form

Formik

Redux/RTK/SAGA/Thunk/Query

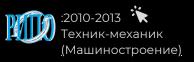
### **ОБРАЗОВАНИЕ**



(React/Redux/Typescript/JS)



:2014-2019 Инженер-механик (Машиностроение)



# КЕВЛИЧ НИКОЛАИ

### FRONTEND PA3PABOTYUK

Frontend-разработчик с 2+ годами опыта в создании и оптимизации Single Page Applications (SPA) с использованием React, Redux, Next.js и TypeScript.

- Имею успешный опыт улучшения производительности и качества кода.
- Реализовал сложную валидацию форм (React Hook Form, Zod) и эффективно управлял состоянием приложения (Redux/RTK/Thunk).
- Разрабатывал переиспользуемые UI-компоненты, работал с дизайнерами над UI/UX.
- Интегрировал приложения с REST API.
- Увлечен оптимизацией кода, достижением измеримых результатов и изучением новых технологий.

#### ОПЫТ РАБОТЫ

FRONTEND DEVELOPER

(Июнь 2022 – настоящее время)

Somegram code 🕻 click

Роль: Team Lead



Команде из 9 человек

Социальная сеть, похожая на Instagram, с настройкой профиля, созданием постов, обработкой изображений, общением в чате и

платными услугами

Training Complex code 🕻 click



Команде из 4 человек

Предназначен для обучения на основе карточек с вопросами и ответами, можно создавать собственные колоды или использовать уже существующие

Task Flow code



Команде из 1 человека

Приложение для создания, редактирования, удаления и отображения задач и списков

Network Communication code





Команде из 1 человека

Приложение для социальной сети, в которой пользователи могут добавлять друзей и общаться.

## ЗАДАЧИ

- **Авторизационный flow:** Регистрация, логинизация, восстановление пароля.
- Обработка и валидация форм ввода.
- Разработка кастомных хуков и React компонентов.
- Пагинация, поиск, сортировка и фильтрация.
- Работа с сервером: REST API, CRUD-операции.
- Роутинг: Реализация навигации, приватных роутов и редиректы.
- Загрузка файлов с клиента.
- Работа с модальными окнами.
- Обработка ошибок: Централизованная обработка и отображение ошибок.
- Тестирование: Unit-тесты с Jest.
- Рефакторинг: Оптимизация кода, контроль дублирования и устранение багов.
- Оптимизация: Использование memo, useMemo, useCallback.
- Командная работа: Git, GitHub, GitFlow.
- **Архитектурные решения:** Применение (UL, BLL, DAL) FSD.

English: A2 (Pre-Intermediate), изучаю для уровня В1