

Roman Ermishin

Software Developer

Контакты

- **GitHub:** [@EternalRival](#)
- **E-Mail:** erdevelopment8@gmail.com
- **Discord:** [eternal_rival](#)
- **Telegram:** [@Eternal_Rival](#)

Обо мне

Увлёкся программированием в июле 2020 года в качестве хобби. Самостоятельно изучил некоторые основы, разработал приложение IQTestRPM (Raven's Progressive matrices) и выпустил его в Google Play (1,5k установок и 4.54★). Когда количество загрузок стало расти, я понял, что мне нравится создавать приложения, которыми пользуются люди. На некоторое время забросил это хобби по личным причинам.

В 2022 году вернулся к разработке, начал самостоятельно изучать JS, а затем и остальные технологии Frontend стека (и немного Backend).

Прошел курсы JS, Node и React в RSSchool (курсы от EPAM), участвовал в командных проектах и являюсь волонтером (помогаю новичкам с различными вопросами).

Технологический стек

- Git
- JavaScript / TypeScript / React
- Webpack / Vite / NextJS (Pages Router)
- TanstackRouter / React Router
- Zustand / Redux Toolkit
- TanstackQuery / Redux Toolkit Query
- RESTful Api / GraphQL
- CSS modules / SCSS / TailwindCSS / MUI
- Yup / Zod
- ESLint / Stylelint / Husky
- Jest / Vitest / Testing Library

Проекты

- **decision-making-tool**
 - Пет-проект
 - [source \(private\)](#) / [deploy \(obfuscated\)](#)
 - Приложение для стримеров, контентмейкеров и любителей настольных игр. Помогает совершить случайный выбор на основе веса варианта (вес увеличивает шанс выпадения варианта).
 - TS, Vite, CSS Modules

- **rsschool-tools**
 - Пет-проект
 - [source](#) / [deploy](#)
 - Приложение для студентов и активистов RSSchool. Помогает посчитать баллы за Crosscheck-проверки и искать деплои других студентов на github-pages.
 - TS, React, NextJS (Pages Router), TailwindCSS, SWR, Zod
- **rsschool-gratitude-summary**
 - Пет-проект
 - [source](#) / [demo](#)
 - Приложение для студентов и активистов RSSchool. Выполнено в виде расширения для Google Chrome. Помогает получить сводку по отправленным и полученным благодарностям в виде таблицы с фильтрацией и сортировкой. Все действия происходят на клиентской части приложения.
 - TS, Vue3, Vite, TailwindCSS, Tanstack Query, Zod
- **christmas-shop**
 - [source](#) / [deploy](#) / [figma](#)
 - Приложение для набора текста с помощью физической и/или виртуальной клавиатуры.
 - TS, Webpack, CSS Modules
- **virtual-keyboard**
 - [source](#) / [deploy](#)
 - Приложение для набора текста с помощью физической и/или виртуальной клавиатуры.
 - JS, Webpack, Sass
- **home-library-service**
 - [source](#)
 - Backend приложение для хранения данных об артистах, альбомах и песнях с возможностью редактирования списка "избранного". Реализован запуск приложения и базы данных в Docker-контейнерах и взаимодействие между ними. Реализована авторизация с использованием JWT. Реализованы маршруты для CRUD-операций через REST Api.
 - TS, NestJS, Docker, Docker Compose, TypeORM, PostgreSQL, JWT, Swagger
- **e-commerce-app**
 - [source](#) / [deploy](#)
 - Frontend приложение онлайн-магазина с использованием CommerceTools service (GraphQL Api) в качестве backend. Изначально это учебный командный проект RSSchool, но я выполнял его один, вне курса (в свободное время) ради расширения моего технологического стека. По завершению trial-доступа разработка и поддержка была остановлена.
 - TS, React, NextJS (Pages Router), Vitest, TailwindCSS, Material UI, Zustand, Tanstack Query, Zod, Axios, GraphQL, React Hook Form, Swiper, Feature Sliced Design, CommerceTools

Сертификаты

- RSSchool JSFE2022Q3 ([Certificate](#))
- RSSchool NodeJS2023Q2 ([Certificate](#))
- RSSchool React2023Q4 ([Certificate](#))