FRONTEND — DEVELOPER
Petr Murashko

📞 +7(903)176-77-88

✈ @Peter_Muraschko

🔗 https://www.linkedin.com/in/petr-murashko-646463297

✉ petr@ocumare.ru

PORTFOLIO

ABOUT ME

Team work, constantly learning and improving my skills through practice.

TECH STACK

- ❖ HTML, CSS (flex, grid), Emotion (styled)
- ❖ TypeScript, JavaScript, PostgreSQL
- ❖ ReactJS, NextJS
- ❖ Redux, Redux-toolkit, RTK-Query, React Query
- ❖ SCSS, CSS Modules, Tailwind CSS, Framer Motion
- ❖ Jest, React-testing-library
- ❖ Webpack, Vite
- ❖ Rest API, GraphQL
- ❖ CI/CD, Git, GitHub, GitLab, Docker, VS Code
- ❖ SOLID, DRY, KISS, YAGNI

LANGUAGES

English *C1 (advanced)*
Russian *Native*
German *A2 (pre-intermediate)*

EXPERIENCE
3 years 5 months

EDUCATION
National Research University "Moscow Power Engineering Institute»

Institute of Information and Computing Technologies,

- ***Bachelor's Degree***   (2020 - 2024)

EXPERIENCE DESCRIPTION

G-Lab LLC

Moscow www.g-lab.ru/

Information Technology, System Integration, Internet

- *Software Development*
- *System Integration, Automation of Enterprise Technological and Business Processes, IT Consulting*

Middle Frontend Developer   *June 2024 — July 2025 (1 year 2 months)*

1) Medical Information System (MIS)

○ Developed 100% of UI components, including complex interactive elements (tables, forms, modal windows), ensuring a unified design system and code reusability.

○ Implemented a dynamic form builder with Drag-and-Drop (DnD) using the dnd-kit library, allowing users to flexibly customize interfaces (dragging and dropping fields: text, numeric, checkbox matrices, etc.). Improved UX for system administrators.

○ Collaborated closely with backend developers:
- Worked with APIs via Swagger and OpenAPI, suggesting improvements for endpoints.
- Designed the interaction contract with the backend.

○ Wrote unit tests (Jest), achieving 95% coverage for critical components.

2) Official website «Association of Legal Education»

○ Developed responsive interfaces based on Figma mockups (Pixel Perfect), including a mobile version using the Tailwind CSS library.

3) Point-Rating System for MSAL University

○ Performed a comprehensive audit of the codebase, leading to a 30% reduction in development time and a 25% decrease in production bugs.

○ Replaced architectural solutions with modern counterparts (RTK Query instead of redux-thunk), reducing backend load by 20% through query optimization and caching. The number of dependencies was reduced from 58 to 35.

○ Conducted refactoring to eliminate anti-patterns (global variables, logic duplication) and implemented typed solutions (TypeScript). Result: a 25% reduction in production incidents (Sentry metrics), saving the team 15-20 hours of work monthly on maintenance.

_____

Thmoon

Moscow

Information Technology, System Integration, Internet

- *Software Development*
- *System Integration, Automation of Enterprise Technological and Business Processes, IT Consulting*

Trainee/Junior Frontend Developer   *March 2022 — June 2024 (2 year 4 months)*

- Developed and implemented 15+ high-performance SPAs for business clients, achieving a 98% Lighthouse performance score and reducing load time by 45%:

  - Complete transition to functional components with TypeScript;

  - Advanced work with hooks (useReducer for complex business logic, useImperativeHandle for component APIs);

  - Optimized contexts (Context API + useMemo) for multi-level applications;

  - Implementation of Compound Components and Render Props patterns for flexible UI kits.

- Configured an intelligent routing system:

  - Dynamic route-based code splitting (React.lazy + Suspense fallbacks);

  - Custom navigation middleware (analytics, NProgress bars);

  - Protected routes with JWT validation via Higher-Order Components.

- Developed a 3-layer abstraction for API interactions:

  - Base Axios client with interceptors (401/403 handling, JWT refresh flow);

  - Automated API layer based on TypeScript Generics;

  - Optimized caching strategy via RTK Query.

- Conducted 120+ code reviews focusing on:

  - Performance (memoization, preventing memory leaks);

  - Architectural cleanliness (adherence to FSD, Feature-Sliced Design).