

Ruslan Gilmullin

+7 *** ** ** | mellonis@yandex.ru | linkedin.com/in/mellonis | mellonis.ru | Moscow, Russia

PROFESSIONAL SUMMARY

Senior Frontend Developer with ~10 years of experience building TypeScript applications for banking and fintech: internet-banking platforms, KYC, payment flows, financial calculators. Production experience across React, Next.js (SSR), and Angular with RxJS. Built and owned a GraphQL BFF (Apollo Server) alongside the frontend at Yandex Pay; currently building a seller-portal application with React at Wildberries. Track record on legacy migrations (ES3 to ES6 client core) and web performance: lazy loading at scale, a Chromium memory-leak workaround. Maintains open-source TypeScript libraries with a live demo at demo.machines.mellonis.ru; portfolio at mellonis.ru. Owns work end-to-end, from architecture to deployment.

CORE COMPETENCIES

- React & Next.js (SSR)
- Angular & RxJS
- TypeScript / JavaScript (ES6+)
- GraphQL - Apollo Client & Server (BFF)
- Redux Toolkit & React Hook Form
- Design Systems & Figma Token Pipeline
- Web Performance & Lazy Loading
- Testing: Jest, Playwright

WORK EXPERIENCE

Wildberries

June 2026 - Present

Senior Frontend Developer | Moscow, Russia (hybrid/remote)

- Developing one of the seller-portal applications with React for the largest e-commerce marketplace in Russia and the CIS
- Stack: React, Angular

Yandex Fintech (Yandex Pay)

April 2022 - November 2025

Senior Frontend Engineer (incl. BFF/SSR) | Moscow, Russia (hybrid/remote)

- Owned the multi-level KYC frontend (React, Next.js) - the gateway every wallet user passes through to unlock a full banking product. Three identification tiers: basic, mobileID + Gosuslugi, and full identification via courier (Yandex Maps with delivery-zone polygons) or an online flow handing off to the native app through a deeplink
- Built the registration flow - entry point into the product, gating every downstream surface
- Built the BFF layer (Apollo Server / GraphQL) for the registration, KYC, and OTP-auth flows - composed internal-service calls into a single graph consumed by the frontend
- Established a design-system-level approach to webview integration with safe-area handling (iPhone notch, Dynamic Island) so product teams stopped duplicating mobile-specific layout fixes
- Led dark-theme integration across the web surfaces; designed the Figma-token-driven CSS generation pipeline so design tokens flowed into code without manual translation
- Stack: Next.js, React, TypeScript, Apollo GraphQL, React Hook Form, Zod, Express, NestJS, AbortController

Art. Lebedev Studio

September 2020 - April 2022

Frontend Developer | Moscow, Russia

- Financial Culture Website - Central Bank of Russia (fincult.info):** hand-rolled loan and deposit calculator engines with full unit-test coverage - annuity and differential amortization, out-of-schedule payments, banker's rounding; inflation calculator UI with dynamic SVG charts (d3 helpers). React, Redux
- "World of Dostoevsky" Igroteka (dostoevskyworld.ru):** Angular + RxJS mini-game collection - live-feel chat with "Fyodor Dostoevsky", drag-to-sort chronology, audio transcription with character-by-character diff; configurable game flow and per-game question sets. Angular, RxJS, TypeScript
- Alex Glushko Plastic Surgeon Website (alexglushko.ru):** CMS page-block architecture with typed blocks and dynamic SVG generation; Lottie + canvas animations; ResizeObserver-driven responsive SVGs; lazy loading engineered around a Chromium memory leak. React, Sass, Redux

- **Agrobank Uzbekistan Website** (agrobank.uz): multi-language banking site with i18next Russian/Uzbek routing; client-side PDF generation via @react-pdf/renderer; Storybook component library. React, Redux Toolkit, TypeScript, Sass

Compass Plus

July 2014 - September 2020

Frontend + Backend Developer | Magnitogorsk, Russia

- Co-designed a modular client-side component system with lazy loading and client-side rendering for the TranzAxis/TranzWare internet-banking platforms - the foundation the team kept extending without regressing first-load performance
- Migrated the client app core from ES3 to ES6 and shipped a Rollup-based ES-module build pipeline while maintaining backwards compatibility with the un-migrated ES3 component layer
- Implemented core internet-banking flows: accounts, cards and deposits, third-party card linking, vendor payments, transaction history
- Wrote backend logic in Java (Tomcat, Spring, Groovy): server-side components for banking operations and REST APIs for the frontend

PROJECTS

Poetry Site

Personal full-stack project: poems and prose with bookmarks, comments, voting, and full-text search. Migrated a long-running PHP site to a multi-stack architecture. Frontend in Next.js / React; Fastify + TypeScript + Zod API with JWT and WebAuthn passkeys, rate-limiting, structured logging; MySQL 8 + Meilisearch. Self-hosted on a single VPS: Docker, nginx, Let's Encrypt, GitHub Actions CI/CD.

Next.js | React | Fastify | TypeScript | MySQL | Meilisearch | Docker

machines-demo

Interactive in-browser Turing & Post machine playground (Svelte 5). User code runs in a Web Worker sandbox - the worker boundary doubles as the security boundary, with terminate-on-timeout; the main thread replays the worker's command stream onto an animated tape step by step. Companion open-source npm libraries: @turing-machine-js/machine and @post-machine-js/machine.

Svelte 5 | Vite | CodeMirror 6 | Web Workers | Playwright e2e

Minesweeper

Framework-agnostic vanilla-TypeScript Minesweeper engine (no DOM/canvas in core), plus a canvas renderer with offscreen sprites and a color-keyed pick canvas for geometry-free hit-testing. devicePixelRatio-aware, Jest tests.

TypeScript | Canvas | Jest

CERTIFICATIONS

React Developer - [Yandex Practicum](#) (2021) | React, Redux/RTK, React Router, TypeScript, WebSocket, unit & E2E testing

TypeScript Intensive - [Ilya Kantor / javascript.ru](#) (2021) | generics, utility types, type-driven design

RxJS Intensive - [Ilya Kantor / javascript.ru](#) (2021) | hot/cold observables, schedulers, operator composition

Dive into Python - [MIPT, Mail.Ru Group / Coursera](#) (2019) | OOP, async/await, multithreading

SKILLS

Frontend: React, Next.js, Angular, Svelte, TypeScript, JavaScript, Redux Toolkit, RxJS, Apollo GraphQL (client), i18next, Storybook, Sass

Backend: Node.js, Fastify, NestJS, Express, GraphQL/Apollo Server, REST APIs, rate-limiting, structured logging

Animations & UI: Lottie, canvas, SVG generation, Swiper **Forms & Documents:** React Hook Form, Zod, Inputmask, @react-pdf/renderer

Browser APIs: Web Workers, ResizeObserver, IntersectionObserver, AbortSignal

Security/Auth: WebAuthn passkeys, JWT, OTP, Ed25519, Web Worker sandboxing **Data:** MySQL 8, Meilisearch

Build & DevOps: Vite, Rollup, Webpack, Docker, nginx, GitHub Actions CI/CD, self-hosted VPS **Testing:** Jest, Playwright

Maps: Yandex Maps, Leaflet, Google Maps **Languages:** Russian (native), English (working proficiency)