

# Ruslan Gilmullin

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

## PROFESSIONAL SUMMARY

Senior Frontend Software Engineer with full-stack experience, focused on TypeScript and interactive frontends, with ~10 years across banking/fintech (internet-banking platforms, KYC, payment flows, financial calculators). Client - React, Next.js. Server - Fastify, NestJS, Apollo GraphQL. Auth - WebAuthn passkeys, JWT. Data - MySQL, Meilisearch. Infra - self-hosted on Docker, nginx, GitHub Actions on a single VPS. Based in Moscow. Most recently at Yandex Pay (Yandex Fintech) - owned the KYC frontend across three identification tiers, built the Apollo Server BFF for KYC/registration/OTP-auth, and established the design-system-level approach to webview integration. Before that, at Art. Lebedev Studio - financial calculators (hand-rolled annuity and differential amortization engines with full unit-test coverage), CMS page-block architecture, and animation-heavy product sites. Earlier, 6+ years at Compass Plus on the TranzAxis/TranzWare internet-banking platforms - ES3→ES6 client-core migration plus Java/Spring backend. Track record on legacy migrations and performance work - lazy loading at scale, Chromium memory-leak workaround. Maintain open-source TypeScript libraries (Turing and Post machine emulators with an interactive demo). Own work end-to-end - from architecture to deployment. Open to in-office, hybrid, and remote roles in Moscow.

## WORK EXPERIENCE

### Yandex Fintech (Yandex Pay)

Apr 2022 - Nov 2025

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

Yandex Fintech's payment platform and digital wallet - web app and native webview surfaces inside the Yandex super-app.

- Built the registration flow - entry point into the product, gating every downstream surface
- Owned the multi-level KYC frontend - the gateway every wallet user passes through to unlock a full-fledged banking product. Three tiers: basic, mobileID + Gosuslugi, and full identification - the last either via courier (Yandex Maps with delivery-zone polygons for picking an available meeting place) or via an online flow that hands off to the native app through a deeplink
- 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:** React, TypeScript, Apollo GraphQL, React Hook Form, Zod, Express, Next.js, NestJS

### Art. Lebedev Studio

Sep 2020 - Apr 2022

#### Frontend Developer · Moscow, Russia

Russia's best-known design studio - worked on several client products.

- **Alex Glushko Plastic Surgeon Website** ([alexglushko.ru](http://alexglushko.ru)) - CMS page-block architecture: clients compose pages from typed blocks with configurable image/video positions and dynamic SVG generation; content-driven popup architecture; Lottie + canvas animations; ResizeObserver-driven responsive SVGs; text along curve paths; lazy loading engineered around a Chromium memory leak. Announced by Art. Lebedev Studio. Stack: React, Redux, Sass
- **Financial Culture Website - Central Bank of Russia** ([fincult.info](http://fincult.info)) - hand-rolled loan and deposit calculator engines with full unit test coverage; true interest accrual with out-of-schedule payments; annuity and differential amortization; banker's rounding; built the inflation calculator UI - dynamic SVG inflation charts (with d3 helpers) and the "superfood" price-comparison mode (calculator engine on the backend, authored by another team); Leaflet maps; custom form controls. Announced by Art. Lebedev Studio. Stack: React, Redux
- **Agrobank Uzbekistan Website** ([agrobank.uz](http://agrobank.uz)) - multi-language banking site with i18next-driven Russian/Uzbek routing; Yandex Maps for branch lookup; client-side PDF generation via @react-pdf/renderer; Storybook component library. Announced by Art. Lebedev Studio. Stack: TypeScript, React, Redux Toolkit, Sass

**Frontend + Backend Developer** · Magnitogorsk, Russia

TranzAxis Internet Banking (TXIB) and TranzWare Internet Banking (TWIB) - client-side web access to banking products, plus adjacent server-side logic in Java.

- Co-designed a modular client-side component system with lazy loading and client-side rendering - the foundation the team kept extending without regressing first-load performance
- Implemented core internet-banking flows: accounts, cards and deposits, third-party card linking, vendor payments, transaction history
- Migrated the client app core from ES3 to ES6 and shipped a Rollup-based ES-module build pipeline; in parallel maintained the un-migrated ES3 component layer, keeping backwards compatibility across the migration
- Wrote backend logic in Java (Tomcat, Spring, Groovy): server-side components for banking operations and REST APIs for the frontend
- **Stack:** JavaScript (ES6, ES3), HTML5, CSS3, ESM, Rollup; Java, Spring, Tomcat, Groovy; REST APIs

**PROJECTS**

---

**Yandex Pay** · [pay.yandex.ru](http://pay.yandex.ru)

Web app and native-app webviews for the payment service. Built the registration flow; owned most of the frontend for the multi-level KYC - Basic, mobileID + Gosuslugi, and full identification via either offline courier (Yandex Maps polygons for delivery zones) or online selfie. Built the Apollo Server BFF for registration, KYC, and OTP-auth flows - composed internal-service calls into a single graph consumed by the frontend. Led dark-theme integration and the Figma-token CSS pipeline; engineered webview safe-area handling on mobile devices

**Poetry Site** · [poetry.mellonis.ru](http://poetry.mellonis.ru)

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

**machines-demo** · [demo.machines.mellonis.ru](http://demo.machines.mellonis.ru)

Interactive in-browser Turing & Post machine playground (Svelte 5). User code runs in a Web Worker sandbox - `unsafe-eval` is scoped to the worker, so the worker boundary doubles as the security boundary; only plain data crosses `postMessage`. Terminate-on-timeout. The main thread holds the UI plus a mirror TuringMachine that replays the worker's command stream onto an animated tape step by step. Vite, CodeMirror 6 with Lezer-based syntax preflight, Playwright e2e.

**@turing-machine-js/machine** · [github.com/mellonis/turing-machine-js](https://github.com/mellonis/turing-machine-js)

npm monorepo: core (State, Tape, TapeBlock, TuringMachine) with Mermaid round-trip state-graph serialization. Two binary-arithmetic libs share ops - 5-symbol marker vs 3-symbol bare alphabets - making the alphabet-size vs state-graph-size trade-off explicit.

**@post-machine-js/machine** · [github.com/mellonis/post-machine-js](https://github.com/mellonis/post-machine-js)

Post-machine builder that compiles down to a `@turing-machine-js/machine` state graph; the two libs share one core via peer-dependency. Supports subroutines (`ca11`), inline grouped instructions, and branching via `check`.

**15-puzzle** · [15-puzzle.mellonis.ru](http://15-puzzle.mellonis.ru)

Sliding-puzzle SPA (vanilla JS + Vite) paired with a zero-runtime-dep Node service. Levels form a chain - boards are deterministically generated (seeded mulberry32 + Fisher-Yates + parity-solvability check) and the server replays the submitted move list before signing each level transition with Ed25519. Each signed blob points only to the next level, and level URLs are unguessable 16-char hex hashes, so the chain is enforced at the URL layer too.

**Minesweeper** · [mines.mellonis.ru](http://mines.mellonis.ru)

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.

**TXIB - TranzAxis Internet Banking**

Compass Plus's web-based internet-banking platform for retail/SMB customers (2014-2020) - accounts, cards, deposits, third-party card linking, vendor payments, transaction history. Modular client-side component system with lazy loading and client-side rendering; integration with a Java/Spring backend via REST APIs.

**TWIB / IB6 - TranzWare Internet Banking**

Compass Plus's next-generation internet-banking platform (2014-2020) - new architecture relative to TWIB 5. Built on the migrated ES6 + ES-module foundation.

## CERTIFICATIONS

---

React Developer - <a href="#">Yandex Practicum</a> · React, Redux/RTK, React Router, TypeScript, WebSocket, unit & E2E testing, capstone project	2021
TypeScript Intensive - <a href="#">Ilya Kantor / javascript.ru</a> · generics, utility types, type-driven design, build tool integration	Jun 2021
RxJS Intensive - <a href="#">Ilya Kantor / javascript.ru</a> · hot/cold observables, schedulers, operator composition, error handling	Jul 2021
Dive into Python - <a href="#">MIPT, Mail.Ru Group / Coursera</a> · 44 h · 100% · OOP, async/await, multithreading	Jan 2019

## SKILLS

---

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

**Animations & UI:** Lottie, canvas, SVG generation, Swiper

**Forms & Documents:** React Hook Form, Zod, Inputmask, @react-pdf/renderer

**Design-system:** Figma-token CSS pipeline, safe-area-aware webview integration

**Browser APIs:** Web Workers, ResizeObserver, IntersectionObserver, AbortSignal

**Maps:** Yandex Maps, Leaflet, Google Maps

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

**Security/Auth:** WebAuthn passkeys, JWT, OTP, Ed25519, Web Worker sandboxing

**JVM (historic):** Java, Spring, Tomcat, Groovy

**Data:** MySQL 8, Meilisearch

**Build:** Vite, Rollup, Webpack

**DevOps:** Docker, DDEV, Dozzle, nginx, Let's Encrypt, GitHub Actions CI/CD, self-hosted VPS, npm monorepo publishing

**Testing:** Jest, Playwright

**Other:** Python (OOP, asyncio, multithreading - certified)

**Languages:** Russian (native), English (B1 - Pearson)