

Гильмуллин Руслан

+7 *** ** * | mellonis@yandex.ru | linkedin.com/in/mellonis | mellonis.ru | Москва, Россия

ПРОФЕССИОНАЛЬНОЕ РЕЗЮМЕ

Senior Frontend Developer, ~10 лет разработки TypeScript-приложений для банкинга и финтех: платформы интернет-банкинга, KYC, платёжные флоу, финансовые калькуляторы. Продакшен-опыт: React, Next.js (SSR), Angular с RxJS. В Яндекс Пэй построил и вёл GraphQL BFF (Apollo Server) вместе с фронтендом; сейчас в Wildberries разрабатывает одно из приложений портала продавцов на React. Опыт legacy-миграций (ядро клиента с ES3 на ES6) и работы с производительностью: ленивая загрузка в масштабе, обход утечки памяти Chromium. Поддерживает open-source TypeScript-библиотеки с живым демо на demo.machines.mellonis.ru; портфолио - mellonis.ru. Ведёт задачи end-to-end, от архитектуры до прода.

КЛЮЧЕВЫЕ КОМПЕТЕНЦИИ

- React и Next.js (SSR)
- Angular и RxJS
- TypeScript / JavaScript (ES6+)
- GraphQL - Apollo Client и Server (BFF)
- Redux Toolkit и React Hook Form
- Дизайн-системы и Figma-token pipeline
- Веб-производительность и ленивая загрузка
- Тестирование: Jest, Playwright

ОПЫТ РАБОТЫ

Wildberries

июнь 2026 - настоящее время

Senior Frontend Developer | Москва, Россия (гибрид/удалённо)

- Разрабатываю одно из приложений портала продавцов крупнейшего e-commerce-маркетплейса России и СНГ на React
- Стек: React, Angular

Яндекс Финтех (Яндекс Пэй)

апрель 2022 - ноябрь 2025

Senior Frontend Engineer (incl. BFF/SSR) | Москва, Россия (гибрид/удалённо)

- Владел фронтендом многоуровневого KYC (React, Next.js) - воронки, через которую каждый пользователь кошелька получает доступ к полноценному банковскому продукту. Три уровня: базовый, mobileID + Госуслуги, полная идентификация - через курьера на Яндекс Картах с полигонами зон доставки либо через онлайн-флоу с переходом в нативное приложение по диплинку
- Сделал флоу регистрации - точку входа в продукт, через которую проходит каждый дальнейший сценарий
- Реализовал BFF-слой (Apollo Server / GraphQL) для регистрации, KYC и OTP-авторизации - собирал вызовы внутренних сервисов в единый граф для фронтенда
- Разработал подход интеграции в вебвью с учётом safe-area на уровне дизайн-системы (iPhone notch, Dynamic Island) - продуктовые команды перестали дублировать мобильные хаки в каждом проекте
- Вёл интеграцию тёмной темы по веб-поверхностям; спроектировал пайплайн генерации CSS из Figma-токенов - токены попадают в код без ручного переноса
- Стек: Next.js, React, TypeScript, Apollo GraphQL, React Hook Form, Zod, Express, NestJS, AbortController

Студия Артемия Лебедева

сентябрь 2020 - апрель 2022

Фронтенд-разработчик | Москва, Россия

- Сайт «Финансовая культура» - Банк России (fincult.info): сам написал движки кредитного и депозитного калькуляторов с полным покрытием юнит-тестами - аннуитетная и дифференцированная амортизация, внеплановые погашения, банковское округление; UI инфляционного калькулятора с динамическими SVG-графиками (d3-хелперы). React, Redux
- Игротека «Мир Достоевского» (dostoevskyworld.ru): сборник мини-игр на Angular + RxJS - чат с «Фёдором Достоевским» с эффектом живого общения, сортировка хронологии перетаскиванием, стенографирование под аудиодорожку с посимвольным диффом; настраиваемый флоу игры и наборы вопросов. Angular, RxJS, TypeScript
- Сайт пластического хирурга Alex Glushko (alexglushko.ru): CMS-архитектура страниц-блоков с типизированными блоками и динамической генерацией SVG; анимации Lottie + canvas; адаптивные SVG на ResizeObserver; ленивая загрузка с обходом утечки памяти Chromium. React, Sass, Redux

- **Сайт Агробанка (Узбекистан) (agrobank.uz):** многоязычный банковский сайт с ru/uz-роутингом на i18next; клиентская генерация PDF через @react-pdf/renderer; библиотека компонентов в Storybook. React, Redux Toolkit, TypeScript, Sass

Compass Plus

июль 2014 - сентябрь 2020

Frontend + Backend Developer | Магнитогорск, Россия

- Спроектировали с коллегами модульную систему клиентских компонентов с ленивой загрузкой и client-side рендером для платформ интернет-банкинга TranzAxis/TranzWare - основу, на которой команда наращивала функциональность без деградации первичной загрузки
- Перевёл ядро клиентского приложения с ES3 на ES6 и поднял Rollup-сборку под ES-модули, сохраняя обратную совместимость с перенесённым ES3-слоем компонентов
- Реализовал ключевые флоу интернет-банкинга: счета, карты и депозиты, привязка карт сторонних банков, платежи вендорам, история операций
- Писал backend-логику на Java (Tomcat, Spring, Groovy): серверные компоненты банковских операций и REST API для фронтенда

ПРОЕКТЫ

[Poetry Site](#)

Личный фулстек-проект: стихи и проза с закладками, комментариями, голосованием и полнотекстовым поиском. Перенёс долгоживущий PHP-сайт в мульти-стек-архитектуру. Фронтенд на Next.js / React; API на Fastify + TypeScript + Zod с JWT и WebAuthn passkeys, rate-limiting, структурированным логированием; MySQL 8 + Meilisearch. Self-hosted на одном VPS: Docker, nginx, Let's Encrypt, GitHub Actions CI/CD.

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

[machines-demo](#)

Интерактивный браузерный эмулятор машин Тьюринга и Поста (Svelte 5). Пользовательский код выполняется в Web Worker-песочнице - граница воркера служит и границей безопасности, с terminate-on-timeout; главный поток шаг за шагом воспроизводит команды воркера на анимированной ленте. Открытые npm-библиотеки: @turing-machine-js/machine и @post-machine-js/machine.

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

[Minesweeper](#)

Сапёр на ванильном TypeScript: игровое ядро без фреймворков, DOM и canvas; отдельный canvas-рендерер с оффскрин-спрайтами и risk-канвасом для определения попаданий по цвету. С учётом devicePixelRatio; тесты на Jest.

TypeScript | Canvas | Jest

СЕРТИФИКАТЫ

React-разработчик - [Яндекс Практикум](#) (2021) | React, Redux/RTK, React Router, TypeScript, WebSocket, юнит- и E2E-тесты

TypeScript Intensive - [Илья Кантор / javascript.ru](#) (2021) | дженерики, utility-типы, типо-ориентированный дизайн

RxJS Intensive - [Илья Кантор / javascript.ru](#) (2021) | hot/cold observables, шедулеры, композиция операторов

Погружение в Python - [МФТИ, Mail.Ru Group / Coursera](#) (2019) | ООП, async/await, многопоточность

НАВЫКИ

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

Бэкенд: Node.js, Fastify, NestJS, Express, GraphQL/Apollo Server, REST API, rate-limiting, структурированное логирование

Анимации и UI: Lottie, canvas, генерация SVG, Swiper **Формы и документы:** 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 **Данные:** MySQL 8, Meilisearch

Сборка и DevOps: Vite, Rollup, Webpack, Docker, nginx, GitHub Actions CI/CD, self-hosted VPS **Тестирование:** Jest, Playwright

Карты: Яндекс Карты, Leaflet, Google Maps **Языки:** русский (родной), английский (рабочий уровень)