

Rafał Eisen

Frontend Developer

React · TypeScript

KONTAKT

rafaleisen1@gmail.com

+48 536 260 650

Szczecin, Polska

linkedin.com/in/rafal-eisen

rafaleisen.pl

github.com/ajzyn

UMIEJĘTNOŚCI

Frontend

React (Hooks, Context), TypeScript, JavaScript (ES6+), Next.js, RR v7, HTML5, CSS3, Tailwind, MUI, Radix UI, ShadcnUI, Tanstack Table, Recharts, Angular, RWD

Stan i komunikacja

Redux Toolkit, RTK Query, Zustand, TanStack Query, React Router, TanStack Router, REST API, WebSocket

Testy i jakość

Vitest, React Testing Library, Jest, Playwright, Cypress, ESLint, Prettier, SonarQube, Code Review

DevOps i narzędzia

Git (GitHub/GitLab), GitHub Actions, CI/CD, Docker, Vite, Webpack, Vercel, AWS (S3)

AI i automatyzacja

AI-assisted dev (Claude Code), n8n, LLM (Gemini Realtime), LiveKit

Backend i metodyki

Node.js, NestJS, PostgreSQL, Supabase · Feature-Based Architecture, Component-Driven Dev, SOLID, A11y, Agile/Scrum

JĘZYKI

Polski (ojczysty) · Angielski (B2)

CERTYFIKATY

OWASP Top 10 · Sekurak

2020

WYKSZTAŁCENIE

Technikum Organizacji Reklamy / Ekonomiczne nr 1, Szczecin

09/2008 – 06/2012

Podsumowanie

Frontend Developer z ponad 4-letnim komercyjnym doświadczeniem w React i TypeScript, specjalizujący się w budowie wydajnych, reużywalnych komponentów i skalowalnej architektury frontendu dla aplikacji enterprise. Zaprojektowałem bibliotekę komponentów, która skróciła implementację nowych funkcjonalności o ~połowę, podejmuję decyzje architektoniczne przyjmowane przez zespół i wspieram rozwój innych deweloperów (mentoring, onboarding). Doświadczenie zdobyte wcześniej jako tester przekłada się na realną dbałość o jakość, edge case'y i pokrycie testami w codziennej pracy

Doświadczenie

Frontend Developer · Accenture

01/2022 – 06/2026 · Szczecin

Wewnętrzne aplikacje webowe (React/TS) dla klientów enterprise. Jedną aplikację zbudowałem od zera (architektura, setup, cały frontend) jako jedyny frontend developer, dwie rozwijałem; zespoły 1–5 osób, w bezpośredniej komunikacji ze stakeholderami.

- Zaprojektowałem i wdrożyłem bibliotekę ~50 reużywalnych komponentów React (Tailwind, MUI) oraz zestaw hooków do złożonych, spójnych tabel i formularzy – skróciło to implementację formularzy i tabel o ~połowę i zapewniło spójność UI przy mniejszej duplikacji kodu.
- Zintegrowałem 150+ endpointów REST z Redux Toolkit + RTK Query (cache, retry, granularne loading/error states) i zmodernizowałem legacy Redux do RTK; synchronizacja danych w czasie rzeczywistym przez WebSocket. Skala: aplikacja obsługująca miliony sygnałów.
- Zoptymalizowałem wydajność interfejsu przy dużych zbiorach danych - ograniczyłem re-rendery (React.memo, useMemo, useCallback) i wdrożyłem lazy loading danych; płynniejsza praca na rozbudowanych tabelach, różnica potwierdzona pomiarami.
- Napisałem ~500 testów automatycznych (Vitest + RTL, integracyjne i jednostkowe), skonfigurowałem CI (GitHub Actions) oraz analizę statyczną (SonarQube); prowadziłem code review 6–10 PR tygodniowo, egzekwując standardy clean code.
- Podejmowałem decyzje architektoniczne przyjmowane przez zespół (m.in. ujednolicony system tabel oparty na komponentach i hookach); mentorowałem dewelopera spoza projektu i wdrażałem nowe osoby (pair programming).

AI Automation Developer (freelance, B2B) · Agento

06/2026 – obecnie

Realizuję komercyjne zlecenia automatyzacji procesów biznesowych w n8n dla klientów agencji, równoległe do pracy w Accenture - w tym głosowych agentów AI obsługujących połączenia przychodzące oraz proste interfejsy webowe triggerujące webhooki. Stack: n8n, LiveKit, Telnyx, Gemini

Software Tester · IA

07/2019 – 12/2021 · Szczecin

- Z własnej inicjatywy przekonałem kierownictwo do wdrożenia testów automatycznych (Cypress) zamiast wyłącznie manualnego QA i zbudowałem pierwsze zestawy automatów, ograniczając powtarzalną pracę manualną i przyspieszając wykrywanie regresji.
- Projektowałem i utrzymywałem setki scenariuszy testowych (UAT + regresja), wychwytyując krytyczne błędy przed wdrożeniem na produkcję; praca w Agile/Scrum.

Projekty

ServiceFlow · własny projekt SaaS

03/2026 – obecnie

SaaS dla firm instalacyjno-serwisowych: CRM, zarządzanie zleceniami, planowanie, powiadomienia w czasie rzeczywistym, wysyłka przypomnień SMS do klientów. Buduję samodzielnie od ~3 miesięcy - od pomysłu i rozmów z użytkownikami, przez design, po pełną implementację; premiera 07/2026. Stack: Next.js, TypeScript, ShadcnUI, Tailwind, PostgreSQL, Supabase, GitHub Actions, Vercel. Biegle wykorzystuję AI-assisted development (Claude Code).

FleetPulse · projekt własny

github.com/ajzyn/fleet-pulse

Publicznie dostępna aplikacja, na której wdrażam i testuję nowe technologie frontendowe. Stack: React Router v7 (jako framework), TanStack Table, Radix UI Themes, Recharts, Playwright (testy end-to-end).