

Damian Tarnowski

Architekt systemów AI | Full-stack .NET

Kraków | hdtldr@gmail.com | github.com/DamianTarnowski | damiantarnowski.azurewebsites.net

PROFIL

Architekt systemów AI i full-stack developer .NET z 23-letnim doświadczeniem w tworzeniu oprogramowania. Projektuję i buduję od zera złożone, skalowalne platformy (SaaS) oparte o AI, łącząc perspektywę inżynierską z biznesową i produktową. Najczęściej pracuję w ekosystemie .NET (Blazor Server/WASM/Hybrid, MAUI) oraz integruję rozwiązania z mikroserwisami (Python, Rust) i usługami AI (RAG, orkiestracja LLM, Semantic Kernel, OpenAI Assistants API).

- Obecnie CTO i AI Architekt (ok. 1,5 roku) w startupie wdrażającym AI; organizuję pracę dwóch zespołów.
- Architektura end-to-end: UI, backend asynchroniczny, warstwa danych (PostgreSQL/pgvector), przechowywanie plików (Azure Blob Storage) + wdrożenia/hosting w Azure (Docker).
- Systemy real-time: SignalR, streaming odpowiedzi AI, WebRTC; interfejsy wzbogacane o JavaScript Interop.
- Wydajność i niezawodność: przetwarzanie równoległe, bezpieczna wielowątkowość, transakcyjny zapis danych (ACID).

KLUCZOWE OSIĄGNIĘCIA

- SaaS EdTech AI (eduedu.pl): generowanie kursów, quizów, fiszek i ćwiczeń.
- Real-time i UX: streaming odpowiedzi AI, SignalR/WebRTC, JavaScript Interop.
- Wydajność i niezawodność: Parallel.ForEachAsync + Interlocked + transakcyjny zapis (ACID).
- OCR dokumentów (faktury) i integracje z systemami księgowymi.

DOŚWIADCZENIE ZAWODOWE

CTO & AI Architekt | startup wdrażający AI | obecnie

- Prowadzenie 2 zespołów: priorytety, proces dostarczania, standardy jakości.
- Architektura i rozwój produktów AI end-to-end: .NET/Blazor, integracje modeli (Python), wdrożenia/hosting w Azure + Docker.

Freelance / konsulting | Architekt .NET / AI | wcześniej

- Budowa aplikacji full-stack i integracji AI dla klientów B2B/B2C.
- Projektowanie architektur, mentoring, optymalizacja wydajności.

WYBRANE PROJEKTY KOMERCYJNE

Platforma Edukacyjna AI (eduedu.pl) – EdTech SaaS

Stack: .NET, Blazor Server, PostgreSQL, Azure (hosting, Blob Storage), KaTeX, Plotly.js

- Stworzenie i wdrożenie do testów platformy do automatycznego tworzenia treści: kompletne kursy, interaktywne quizy, fiszki i ćwiczenia wieloetapowe z weryfikacją oraz feedbackiem AI w czasie rzeczywistym.
- Zaprojektowanie i implementacja architektury end-to-end: responsywny UI w Blazor Server, backend orkiestrujący wyspecjalizowane serwisy AI oraz warstwa danych z persystencją w PostgreSQL i składowaniem plików w Azure Blob Storage.
- Implementacja zaawansowanego systemu interakcji z AI: streaming odpowiedzi oraz renderowanie notacji matematycznych (KaTeX) i wizualizacji (Plotly.js) bezpośrednio w UI.

- Budowa wydajnego mechanizmu przetwarzania zapytań do AI: Parallel.ForEachAsync + bezpieczne wątkowo zliczanie postępu (Interlocked) + transakcyjny zapis wyników do bazy (ACID).
- Optymalizacja wydajności i zarządzania stanem w Blazor Server (m.in. IDbContextFactory, centralny serwis stanu AppState, wzorzec Template/Instance); hosting w Azure.

SensumAI – konsola do głębokiej analizy emocji w czasie rzeczywistym

Demo: vimeo.com/988776481

Stack: .NET, Blazor Server, SignalR, WebRTC, JavaScript Interop, Python (modele AI), FFMpegCore

- Zaprojektowanie aplikacji real-time opartej o .NET i Blazor Server, z dwukierunkową komunikacją SignalR (sygnalizacja WebRTC, czat, aktualizacje dashboardu bez odświeżania strony).
- Integracja wielomodalnych modeli AI (tekst, głos, wideo): analiza semantyczna tekstu z użyciem OpenAI Assistants API oraz modele w Pythonie (uruchamiane w kontenerach) do analizy ekspresji twarzy i tonu głosu w czasie rzeczywistym przez REST API.
- Opracowanie potoku przetwarzania audio/wideo: WebRTC do streamingu wideo, JS interop z faceapi.js (detekcja twarzy i punktów charakterystycznych) oraz Web Speech API (transkrypcja mowy na tekst).
- Przetwarzanie surowych danych audio po stronie serwera z użyciem FFMpegCore (konwersja, resampling) w celu przygotowania wejścia dla modeli AI.
- Stworzenie dynamicznego dashboardu do wizualizacji danych analitycznych (wskaźniki, wykresy), w tym niestandardowa wizualizacja korelacji na HTML Canvas; hosting na własnym serwerze.

Więcej projektów i case study na: damiantarnowski.azurewebsites.net

UMIEJĘTNOŚCI I TECHNOLOGIE

- **Architektura i inżynieria:** architektura mikroserwisowa, systemy rozproszone, projektowanie API, SOLID, pragmatyczne prototypowanie, embedded
- **.NET i backend:** .NET 10/9/Core, C#, ASP.NET Core, Blazor (Server/WASM/Hybrid), MAUI, REST API, SignalR, Entity Framework Core, programowanie asynchroniczne i wielowątkowe
- **AI/ML:** Microsoft Semantic Kernel, OpenAI Assistants API, RAG, orkiestracja LLM (GPT, Gemini, Claude, modele lokalne), prompt engineering, NLP, computer vision
- **Dane:** MS SQL Server, PostgreSQL (pgvector), Neo4j, SQLite, NoSQL
- **Frontend:** MudBlazor, HTML5, CSS3, JavaScript, integracja JS z Blazor, KaTeX, Plotly.js, WebRTC, Web Speech API, MediaRecorder API, HTML Canvas
- **DevOps i narzędzia:** Git, GitHub Actions (CI/CD), Azure (Blob Storage, hosting aplikacji), Docker, FFMpegCore, Visual Studio
- **Języki:** C#, Python, JavaScript, C, Assembler

ZAINTERESOWANIA

- aktywność fizyczna, zdrowy tryb życia, nowe technologie, tworzenie muzyki, prawo i finanse, biznes

KLAUZULA RODO (OPCJONALNIE)

Wyrażam zgodę na przetwarzanie moich danych osobowych zawartych w CV na potrzeby obecnego oraz przyszłych procesów rekrutacyjnych, zgodnie z obowiązującymi przepisami.