



ОЛЕКСІЙ
МАЗУРЕНКО
FULL-STACK DEVELOPER

EMAIL newiqa@gmail.com
ТЕЛ. +380 63 588 3160
САЙТ oleksiimazurenko.dev
GITHUB github.com/oleksiimazurenko
LINKEDIN linkedin.com/in/oleksii-mazurenko
TELEGRAM @oleksii_mazurenko
ФОРМАТ Remote

НАВИЧКИ

Основний стек

Мови:

TypeScript, JavaScript

Розмітка і стилі:

HTML, CSS/SCSS

Ядро:

React 19 · Next.js — App Router, RSC, SSR/SSG/ISR, Server Actions, Route Handlers, proxy (middleware)

Стилі / UI:

TailwindCSS · shadcn / Base UI · Framer Motion · GSAP (прості анімації) · Lexical · dnd-kit

Стан і дані:

Zustand · SWR · TanStack Query · Zod · Prisma · MongoDB (Atlas) та PostgreSQL на рівні CRUD

Архітектура:

модульна структура під App Router · проєктування схеми даних (Prisma)

SEO / i18n:

sitemaps, robots.txt, метатеги, JSON-LD, canonical, aria, семантична розмітка · next-intl · Lingui

Аналітика:

Amplitude (навішування івентів) · власна система збору подій · Faro (Grafana) · oxlint · Vercel

AI-розробка:

Claude Code щодня, агенти зі скілами під рутину, MCP-інтеграції

Інтеграції

Solidgate · Stripe · Better Auth · Firebase Auth · GrowthBook · PostHog · Strapi CMS · Payload CMS · AWS (Lambda, SST + OpenNext) · Hetzner · OVH

Також працював з

Strapi 4/5 + плагіни · бїлд-скрипти та кодоген · Playwright (візуальна регресія) · rnpm · Turborepo · CI/CD · Solid.js · Svelte

ЯК Я ПРАЦЮЮ

1 · Спершу прибрати задачу, потім прискорювати

- Робить агент — задача більше не приходить ні до кого, агент виконує її сам: авторев'ю коду, аналітика сама падає в Slack.
- Робить замовник — задача не доходить до мене, замовник виконує її разом з AI: аустріч, платформа інфлюенсерів.
- Робить команда — плагін, через який команда змінює сама: редіректи в CMS для SEO-команди.

2 · Коли роблю сам

- Рутинний патерн — спершу пишу під нього скіл.
- Plan mode перед кодом: спершу межі задачі й план, потім реалізація — сам або з агентом, залежно від ціни помилки.

3 · Конвеєр

dev → build → юніт-тести + Playwright (агент, результат читаю я) → стейдж (мій ручний прохід) → тестувальник → PR (агенти зі скілами за зонами) → прод → фінальна перевірка

Основа: OOP · SOLID · DRY · KISS · YAGNI · CI/CD + сувора послідовність кроків, яка описана тут: [oleksiimazurenko.dev/pipeline](#)

ЗНАННЯ МОВ

Українська — рідна

English — A2: технічну документацію читаю, розмовну підтягаю щодня

FULL-STACK DEVELOPER

REACT / NEXT.JS

Full-Stack розробник, 5+ років у React / Next.js; серверна частина — у межах Next.js (Server Actions, Prisma). Оцінюю задачі за бізнес-цінністю і веду свою частину — від рішення до продакшену. Перший рік у Promova робив продукт руками, далі додав до роботи AI-агентів: рутину й частину реалізації пишу з ними, а постановка, архітектура, ревію і відповідальність за прод лишаються моїми.

ДОСВІД РОБОТИ

Promova (promova.com)

Кві 2025 — Сер 2026 · Remote

Ціль: тримати систему бізнес-воронки і публічних SEO-сторінок швидкою й керованою для команд; підтримувати web2app-шлях і автоматизувати рутину.

- Продуктова робота (перший рік — сам). Міграція на App Router, Lingui, TanStack Query, оптимізація запитів і зайвих рендерів. Воронки, верстка секцій, поточні задачі й фікси.
- Web2app-перехід. Сторінка-транзит між рекламою і App Store: перехоплював результат AppsFlyer Smart Script (OneLink) через assessor-пастку на window — редірект у момент запису, без опитування, плюс фолбек-таймер і власне складання атрибуції.
- Аналітика. Події Amplitude по всій воронці — від входу з реклами до оплати й переходу в застосунок; аналіз, де саме відвалюються користувачі.
- Платежі. Інтеграція Solidgate і Stripe за документацією — підключив і довів до робочого стану.
- Часті UI-сутності на чистий CSS. Ініціював ресерч: модалка, акордеон, слайдер, fit-text — що з цього можна без JS. Підхід ліг в основу опенсорс-пакета (див. React Patterns).
- Статичні сторінки для динамічних роутів. Скрипт, що генерує їх перед деплоєм. Вирішило проблему Turbopack: сторінка тягне тільки свої чанки CSS і JS, а не все підряд.
- A/B-тести з layout у проксі — щоб сторінка лишалась статичною.
- Черга подій аналітики. Між завантаженням сторінки й ініціалізацією Amplitude юзер устигав клікнути — подія губилась. Запропонував чергу, що накопичує івенти й відтворює після ініціалізації.
- Рев'ю PR за зонами проєкту (агенти Claude). Агент визначає, якої частини системи торкається зміна — оптимізація, платежі, аналітика — і застосовує відповідний набір правил перевірки.
- Аустріч інфлюенсерам — автоматизована підготовка й розсилка листів замість ручної роботи.
- Yarn → rnpm (карантин, ізоляція пакетів, швидкість) · Redux → Zustand · Strapi-плагіни — визначав потребу, вів реалізацію, тестував.
- Результат: від виконання задач перейшов до постановки й ведення рішень. Вибудував процеси, де контент-команда й AI-автоматизація роблять максимум самостійно.

Southern Core (Outsource, NDA)

Вер 2024 — Лют 2025 · Remote

- Ціль: привести проєкти до сучасного стеку й продакшн-якості — архітектура, SEO, стабільність.
- Міграція Pages Router → App Router; архітектура на Feature-Sliced Design.
- Обробка помилок — додав адекватний хендлінг там, де його не було. UX — виправляв недоліки, відновлював логіку зрозумілого інтерфейсу. API Request Batching — мінімізація запитів клієнт↔сервер.
- Технічне SEO та доступність: robots.txt, sitemaps, метатеги, JSON-LD, canonical, alt і title, aria. Локалізація — сам впроваджував next-intl.
- Code review для молодших розробників — інкапсуляція сутностей, модульність, баланс перевикористання компонентів.
- Інфраструктура: Next.js (ISR) + Directus на OVH; деплой через SST + OpenNext на AWS Lambda; CRON-автоматизація; A/B через PostHog; CI/CD.
- Результат: швидкі сторінки (ISR + batching), низька вартість інфраструктури, контент керується через Directus без розробника.

Landee (landee.app)

Сер 2023 — Сер 2024 · Remote

- Ціль: конструктор персональних сайтів (MVP) з нуля.
- Архітектура. Планував структуру коду та схему БД; MVP на Next.js (App Router), масштабована FSD-архітектура.
- Блоковий редактор на Lexical (rich-text: списки, таблиці, код) з drag-and-drop (dnd-kit, GSAP, shadcn/Radix).
- Дані. MongoDB Atlas через Server Actions + Prisma; повний CRUD; офлайн-стійка синхронізація — черга змін із відкатом. Власна аналітична система — збір кліків і показів, власні графіки.
- Публікація та онбординг. Публікація сторінок (ISR), онбординг із шаблонами; автентифікація (Clerk). Локалізація — сам впроваджував next-intl (uk/en).
- Серверна частина. Cron-роути на Vercel; монорепо на Turborepo; інтеграція зі сторонніми API (фонові фото й відео).
- Фронтенд і якість. TailwindCSS; анімації GSAP (простої складності); UI/UX. Обробка помилок; мінімізація useEffect. SEO: sitemaps, robots.txt, метатеги, JSON-LD.
- Результат: запущений MVP конструктора — редактор, публікація, онбординг, аналітика.

ОСОБИСТІ ПРОЄКТИ · oleksiimazurenko.dev

- Sortune — менеджер бібліотеки YouTube Music: AI-агенти через MCP класифікують жанри й самі розкладають плейлисти за сезоном і стилем. Спроєктував і запустив.
- Flint — платний macOS-застосунок у продажу: звільняє місце на диску від хмарних сервісів (iCloud, Google Drive, OneDrive, Dropbox).
- React Patterns — опенсорс-бібліотека в npm (@oleksiimazurenko/react-patterns): accordion, dialog, popover, slider, parallax, reveal, scroll-progress, sticky-shrink, fit-text, analytics. CSS-first, server-first, делегування подій.
- oleksiimazurenko.dev — власний сайт на Solid.js: мінімум JS, чистий CSS, швидко завантаження.

ВОЛОНТЕРСТВО

- Poludne (poludne.com) — платформа про українську літературу: цитати, автори, каталог творів. Розробка на Svelte, у команді, на волонтерських засадах.