PROGRAM SZKOLENIA FRAMEWORKI

Szkolenie Node.js

Zbudujesz w Node.js serwer z bazą danych, uwierzytelnianiem i testami, gotowy do wdrożenia.

CZAS TRWANIA	POZIOM	FORMA
3 dni	**średniozaawansowany**	**warsztat dla zespołu, zdalnie lub stacjonarnie**

Jeśli piszesz frontend w JavaScript, tym samym językiem i tymi samymi narzędziami zbudujesz serwer. Kłopot w tym, że wiele poradników o Node.js opisuje stan sprzed kilku lat: `nodemon`, `dotenv`, `ts-node` i `axios` tam, gdzie dziś wystarcza sam Node.js.

Na szkoleniu budujesz jedną aplikację od pustego katalogu: serwer HTTP z bazą danych, logowaniem użytkowników, obsługą błędów i testami, na aktualnym Node.js LTS. Poznasz mechanizmy, które decydują o tym, czy serwer wytrzyma produkcję: event loop, strumienie, logi i łagodne zamykanie.

PROGRAM

3 dni

- Nowoczesny Node.js i projekt w npm
  - Wersje LTS i roczny cykl wydań: którą wersję wybrać do projektu
  - TypeScript bez kompilacji (type stripping): tylko składnia usuwalna, bez `enum` i parametrów konstruktora jako pól
  - `--env-file`, `--watch`, `node --run` i `fetch` zamiast paczek
  - Lockfile, `npm ci` i aktualizacje zależności
- Jak działa Node.js
  - Moduły ES i CommonJS
  - Event loop: co blokuje serwer i jak to wykryć
  - Promise, `async` / `await` i `EventEmitter`
- Strumienie i praca z plikami
  - Stream i Buffer: duże pliki bez zapełniania pamięci
  - `pipeline` i obsługa błędów w strumieniach
  - Ciężkie obliczenia w `worker_threads`
- Serwer HTTP
  - `node:http` bez frameworka
  - Express 5, Fastify i Hono: czym się różnią
  - Routing, middleware i obsługa błędów
  - Walidacja danych wejściowych z Zod

- Baza danych
  - PostgreSQL przez ORM (Drizzle), zapytania SQL tam, gdzie ORM przeszkadza
  - Migracje schematu i dane testowe
  - Transakcje i pula połączeń
- Uwierzytelnianie i autoryzacja
  - Sesje w ciasteczkach a tokeny JWT
  - Hasła: wbudowany `crypto.argon2`, zapis parametrów i soli, porównanie przez `timingSafeEqual`
  - Role i uprawnienia do zasobów
  - OAuth 2 i OpenID Connect w skrócie
- Bezpieczeństwo
  - `npm audit` i ryzyko w zależnościach
  - Limity zapytań i nagłówki bezpieczeństwa
  - Model uprawnień Node.js (`--permission`): ograniczenie dostępu do plików i procesów
- Błędy, logi i zamykanie serwera
  - `unhandledRejection` i `uncaughtException`: jak na nie reagować
  - Logi strukturalne (pino)
  - Łagodne zamykanie serwera na `SIGTERM`
  - Debugowanie przez `--inspect` w edytorze i Chrome DevTools
- Testowanie
  - Wbudowany runner `node:test`, mocki i pokrycie kodu
  - Testy jednostkowe i integracyjne
  - Testy API na prawdziwej bazie w kontenerze

Projektowanie API (REST, GraphQL, OpenAPI, webhooki) i OAuth 2 omawiam szerzej na szkoleniu API: REST i GraphQL. Jeśli kupujesz to szkolenie razem z nim i ze szkoleniem Bezpieczeństwo WWW, moduły o uwierzytelnianiu i bezpieczeństwie skracam do przeglądu, a czas oddaję na rozwój projektu.

DLA KOGO?

- Dla programistów frontendu, którzy chcą pisać backend w JavaScript.
- Dla programistów innych języków, którzy przechodzą na Node.js.
- Dla zespołów, które utrzymują starszą aplikację w Node.js i chcą ją unowocześnić.

CO MUSISZ UMIEĆ?

- Programować w JavaScript: moduły, klasy, `async` / `await`.
- Pracować z terminalem i Gitem.
- Znać podstawy SQL (`SELECT`, `INSERT`, `JOIN`).
- Znać podstawy TypeScriptu (nie są konieczne, ale ułatwią pracę).
- Uruchamiać kontenery w Dockerze, bo testy korzystają z bazy w kontenerze. Na komputerze potrzebujesz działającego Dockera.

CZEGO SIĘ NAUCZYSZ?

- Uruchomisz projekt w TypeScript i z plikiem `.env` bez dodatkowych narzędzi.
- Zrozumiesz event loop i znajdziesz kod, który blokuje serwer.
- Przetworzysz duży plik strumieniem bez zapełniania pamięci.
- Wybierzesz framework HTTP i zbudujesz na nim serwer z walidacją danych.
- Połączysz aplikację z PostgreSQL i napiszesz migracje schematu.
- Zabezpieczysz aplikację logowaniem, rolami i limitami zapytań.
- Skonfigurujesz logi i łagodne zamykanie serwera przed wdrożeniem.
- Przetestujesz aplikację testami jednostkowymi i integracyjnymi w `node:test`.

OPINIE UCZESTNIKÓW

> *„Gruntownie, krok po kroku stworzyliśmy aplikację sklepu internetowego, po drodze pisząc testy jednostkowe, e2e, web komponenty, autorski backend oparty na Node.js + Express.js, jak również wykorzystaliśmy narzędzia CI.”*
>
> Łukasz, JavaScript Developer (WarsawJS MasterClass)

> *„Piotr przekazywał wiedzę na temat tworzenia odpornych na błędy aplikacji internetowych. Podczas szkolenia tworzyliśmy testy jednostkowe, testy end-to-end aplikacji po stronie klienta oraz testy backendu napisanego w Node.js.”*
>
> Dominik, Frontend Developer (WarsawJS MasterClass)

TRENER

Piotr Kowalski

W branży IT od 2008 roku. Od 2014 jest mentorem dla wielu developerów. Jako trener spędził na sali szkoleniowej ponad 1000 godzin, a na LinkedIn ma 52 rekomendacje. Każde jego szkolenie to dawka solidnej wiedzy oraz humoru.

Więcej: piecioshka.pl/piotr-kowalski

ZORGANIZUJMY TO SZKOLENIE W TWOIM ZESPOLE

Program dopasuję do Waszego projektu i poziomu grupy. Napisz, z czym pracujecie, a odpowiem z propozycją terminu i zakresu.

KONTAKT **piecioshka.pl/kontakt**
E-MAIL **piecioshka+blog [małpka] gmail.com**
SZKOLENIE **piecioshka.pl/szkolenia/nodejs/**