YOORshop SAS
RCS 817 466 147 — 254 Rue Vendôme, 69003 Lyon, France — yoorhosting.com
[dane firmy do uzupełnienia]

Ref. raportu: YH-DEMO-2026-02
Data analizy: 30 października 2026

● DIAGNOZA I KLUCZOWE ROZWIĄZANIA

RAPORT INŻYNIERII APLIKACJI I BAZY DANYCH • REZULTAT NR 2

Optymalizacja wydajności strony internetowej

Analiza skoków CPU do 70 %, blokad pamięci podręcznej menu i przepisanie zapytania SQL skanującego 787 504 wiersze.

KLIENT DOCELOWY

Firma: SARL Moto Pièces Express (Fikcyjna)
Sklep online: https://boutique-moto.example (PrestaShop 8.2.3)
Serwer dedykowany: srv-demo.yoorhosting.com (Xeon 16 rdzeni, 64 GB RAM)
Środowisko: PHP-FPM pool motodemo_prod (48 workerów) • MariaDB

OBCIĄŻENIE CPU
70 % do 1 270 %
45 skoków podczas importu katalogu

WOLNE ZAPYTANIA PHP
540 / 976
Zablokowane w preg_grep menu

PRZYSPIESZENIE SQL
x 4 000
Z 1,60 s do 0,0004 s na wywołanie

CZAS ZAPYTANIA O NOWOŚCI
0,0004 s
Bezpośrednie porównanie po indeksie

ZGŁOSZENIE KLIENTA

„Wobec poważnych spowolnień podczas regularnej aktualizacji cen i stanów przez automatyczne moduły importu oraz powtarzających się przeciążeń procesora (skoki do 1 270 % łącznego CPU na 16 rdzeniach), klient zlecił YOORshop **wskazanie dokładnej przyczyny blokad** oraz opracowanie **planu optymalizacji bez negatywnego wpływu na katalog**."

ZIDENTYFIKOWANY WSPÓLNY MECHANIZM PRZECIĄŻENIA

Analiza porównawcza dziennika wolnych zapytań PHP (*PHP slowlog*) i znaczników czasu modyfikacji w bazie (`date_upd`) ujawniła strukturalne wąskie gardło: **każdy zapis produktu czyści jednocześnie dwie ciężkie pamięci podręczne aplikacji**, które są natychmiast przeliczane w pętli podczas synchronizacji:

Potwierdzona korelacja czasowa: Modyfikacja 22 produktów w ciągu godziny pokrywa się co do sekundy z wysypem wolnych zapytań (20:44, 20:47, 20:49, 20:50 oraz 20:52–20:56). Łącznie zarejestrowano 45 skoków obciążenia wywołanych żądaniami POST z adresu IP operatora `198.51.100.22` .

1 Problem nr 1: Nadmierna przebudowa menu górnego (Moduł `ps_mainmenu`)

PUNKT ANALIZY	USTALENIA TECHNICZNE I ZATWIERDZONE ROZWIĄZANIE
Zbędne podpięcie hooków	Moduł jest niepotrzebnie podpięty pod `actionObjectProductAddAfter` , `actionObjectProductUpdateAfter` i `actionObjectProductDeleteAfter` (tabela `psdemo_hook_module`). Hooki te wymuszają `clearMenuCache()` (ps_mainmenu.php:1007), usuwając cały cache JSON.
Koszt przebudowy	**540 z 976 wolnych żądań** PHP zatrzymuje się na `preg_grep` (ps_mainmenu.php:796). Przebudowa porównuje **9 788 kategorii z 26 968 obrazami**. Dwa procesy równolegle przeliczają ten sam plik (kolizja zapisu o 21:02:01 i 21:02:05).
Natychmiastowa naprawa	**Odpięcie `ps_mainmenu` od trzech hooków produktowych** w panelu PrestaShop (*Wygląd* > *Pozycje*). Zmiana w 100 % bezpieczna: menu nie zawiera pojedynczych produktów (`MOD_BLOCKTOPMENU_ITEMS = CAT77,CAT71,LNK1,LNK3`). Hooki kategorii i stron CMS pozostają aktywne.

2 Problem nr 2: Nieindeksowane zapytanie bloku „Nowe produkty” (`ps_newproducts`)

Moduł `ps_newproducts` po każdej aktualizacji wywołuje `Product::getNewProducts()`. Oryginalne zapytanie wykonuje funkcję bezpośrednio na indeksowanej kolumnie, co uniemożliwia MariaDB użycie indeksu:

Zapytanie oryginalne (bez indeksu) **1,60 sekundy**

`WHERE DATEDIFF(date_add, NOW()) < 20`

Skutek w MariaDB: Indeks `date_add` jest pomijany. Pełny skan (*Full Table Scan*) **787 504 wierszy** w `psdemo_category_product`, tabela tymczasowa na dysku i ciężkie sortowanie.

✓ Zapytanie zoptymalizowane (override) **0,0004 sekundy**

`WHERE date_add > '2026-10-07'`

Skutek w MariaDB: Natychmiastowe użycie indeksu B-Tree. Odczyt ograniczony do 8 wymaganych rekordów. **Zmierzone przyspieszenie: 4 000 razy**.

3 Dobór wydajności puli PHP-FPM

Aby obsłużyć równoległy ruch bez błędów HTTP 503, pula `motodemo_prod` została indywidualnie skonfigurowana przez nasz zespół:

MAKS. PROCESÓW (CHILDREN)
48 workerów
Standard współdzielony: 15

PRZYDZIELONA PAMIĘĆ
1 024 MB / worker
Obsługa dużych importów katalogu

ŚREDNIE OBCIĄŻENIE CRON
57 % rdzenia
Stabilna praca co minutę

ODBUDOWA CACHE
Natychmiastowa
Brak kolizji procesów

4 Podsumowanie działań i uzyskanych efektów

#	ZALECANE DZIAŁANIE	SPOSÓB WDROŻENIA	ZMIERZONY EFEKT
1	Odpięcie modułu `ps_mainmenu`	Back-office (Wygląd > Pozycje)	**-540 wolnych zapytań** Usunięcie blokady `preg_grep`.
2	Wdrożenie nadpisania SQL `getNewProducts`	Plik w `override/classes/Product.php`	**Czas krótszy 4 000x** Koniec skanu 787 504 wierszy MariaDB.
3	Usunięcie sond monitorujących	Oczyszczenie `/root/cpuwatch/`	**Zakończono** Sondy zatrzymane 28.10 o 12:44.

Zatwierdzenie techniczne ekspertyzy:
Dział Inżynierii i Wydajności Web — YOORshop SAS

✓ Plan Optymalizacji Zatwierdzony