

Audyt sklepu WooCommerce: wyposażenie24.pl, wrzesień 2026

Sporządzono: 24 września 2026

Sklep	wyposazenie24.pl, sprzedaż w Polsce
Asortyment	wyposażenie gastronomii (4 574 z 7 330 produktów), warsztat i garaż, dom i ogród, elektronarzędzia, wagi, chemia przemysłowa i inne
Model sprzedaży	dropshipping: produkty, ceny i stany importowane z pliku hurtowni
Platforma	WordPress 6.8.1, WooCommerce 9.9.3, PHP 8.3
Motyw	Porto 7.5.3 z motywem potomnym Porto Child 1.1
Wtyczki	35, w tym 32 aktywne
Serwer	hosting współdzielony, serwer LiteSpeed
Okres działania	09.2024-29.07.2026; domena wygaśa 06.09.2026
Audyt	23.09.2026, wariant podstawowy

Raport przykładowy wariantu podstawowego. Sklep należał do wykonawcy audytu. Do audytu przywrócono go 23.09.2026 z kopii plików i bazy na tym samym serwerze, na osobnym adresie za hasłem, z zanonimizowanymi danymi klientów (dalej: kopia audytowa). Raport opisuje stan sklepu w chwili wyłączenia; pomiary wydajności pochodzą z kopii audytowej.

Wszystkie odstępstwa od zwykłego audytu są opisane w ostatniej części raportu. Przy każdym problemie stoi rekomendacja naprawy: wskazówka, co zmienić, bez instrukcji krok po kroku.

Priorytety w całym raporcie: **krytyczny** kosztuje zamówienia teraz albo otwiera sklep na atak; **wysoki** hamuje sprzedaż albo podnosi koszt utrzymania; **średni** to porządek i higiena; **niski** nie wpływa na zamówienia. Priorytety podano tak, jakby sklep działał. Sklep jest wyłączony, więc dziś realne ryzyko mogą stwarzać tylko klucze usług zapisane w jego plikach i bazie (punkt 3.4); pozostała kolejność dotyczy sklepu, który miałby wrócić do sprzedaży.

1. Skróót

Kasa działała bez typowych przeszkód, ale od czerwca 2025 r. sklep sprzedawał ofertę, której nie znał: import stanów z hurtowni zatrzymał się 15.06.2025 i nie został wznowiony do końca działania sklepu. Telefony dostawały wolniejszy sklep niż komputery, aktualizacje nie miały jak przyjść, a ostrzeżenia zapory nie trafiały do nikogo.

Stan	Obszar	Co ustalono
✗	Oferta	1 095 z 4 688 produktów oznaczonych jako dostępne (23%) hurtownia nie ma; 580, które ma, sklep ukrywa jako wyprzedane (punkt 4.1)
✗	Telefon	żadna strona nie trafia do pamięci podręcznej: serwer odpowiada w 0,62 s (mediana) wobec 0,06 s strony z pamięci; każdy nowy odwiedzający wczytuje pierwszą stronę dwa razy (2.2, 2.3)
✗	Aktualizacje	92 opublikowane luki w 19 składnikach, 45 do wykorzystania bez logowania, w tym krytyczna w rdzeniu WordPressa; automatyczne poprawki zablokowane przez repozytorium kodu w katalogu strony (3.1, 3.2)
✗	Pliki w katalogu strony	repozytorium <code>.git</code> i 5 998 spisów plików po kopiowaniu przez FTP; serwer wydaje pliki ukryte (3.5)
✗	Klucze usług	zapisane w kodzie i w bazie; klucz OpenAI był aktywny w dniu audytu, właściciel odwołał go 24.09.2026 (3.4)
✗	Zapora i logowanie	42 ostrzeżenia bez adresu do powiadomień; żadne konto nie ma weryfikacji dwuetapowej (3.3)
✗	Przelew	przykładowe dane konta z WooCommerce na stronie potwierdzenia i w mailu do klienta (4.2)
✓	Kasa	zakup bez konta, jeden krok, koszt wysyłki widoczny w koszyku; zamówienie testowe do strony potwierdzenia w 1,6 s (4.3)
✓	Podatki i ceny	VAT 23% wg adresu wysyłki; najniższa cena z 30 dni zapisywana dla wszystkich 7 330 produktów (4.5)
✓	Konta	rejestracja zamknięta, trzy konta, brak haseł aplikacji (3.6)

Dlaczego sprzedaż nie działała tak, jak mogła. Sama kasa nie odstraszała; sprzedaż hamowały rzeczy przed nią. Klient mógł zamówić jeden z 1 095 produktów, których hurtownia nie miała, i dostać anulowanie zamiast paczki; nie widział 580 produktów, które mógł kupić; na telefonie każda strona kategorii i produktu czekała na serwer. Porzuconych koszyków w wariantach podstawowych nie da się policzyć.

Co zrobić najpierw. Pełna lista z priorytetami i tym, kto wykonuje: [rozdział 7, Co poprawić, w jakiej kolejności](#).

1. Zaktualizować WordPress do 6.8.10 i usunąć z katalogu strony repozytorium kodu oraz spisy plików po kopiowaniu przez FTP.
2. Wznowić import stanów z hurtowni, z powiadomieniem o każdym nieudanym przebiegu.
3. Przywrócić pamięć podręczną dla telefonów i wyłączyć tryb gościa we wtyczce LiteSpeed Cache.
4. Usunąć klucz OpenAI z kodu wtyczki i odwołać pozostałe klucze usług zapisane w sklepie.
5. Zaplanować aktualizację wtyczek i WooCommerce jako osobny projekt z testem na kopii sklepu.

Czego audyt nie objął: zachowań klientów (wariant rozszerzony; sklep bez ruchu), treści i pozycjonowania, zgodności prawnej, dostarczalności poczty, kopii zapasowych po stronie hostingu oraz pięciu wtyczek, których nie ma w publicznych bazach luk.

2. Wydajność

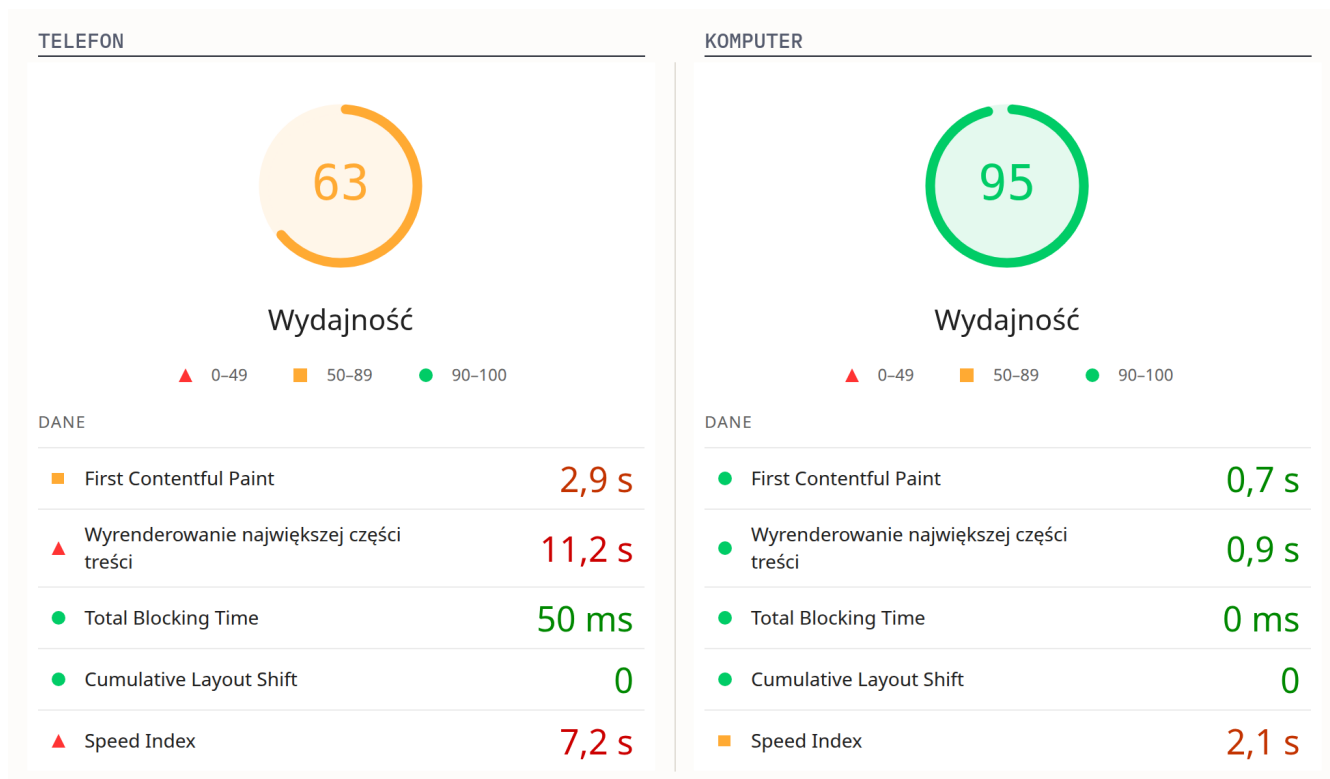
2.1 Pomiar

Lighthouse 12.8.2 w przeglądarce Chromium na komputerze audytora, profil telefonu i komputera, ograniczenie łącza i procesora symulowane (ustawienia domyślne), trzy przebiegi na stronę, mediana; 23.09.2026 13:30-13:54 UTC+2. PageSpeed Insights nie był możliwy, bo kopia audytowa jest za hasłem; porównanie tego zestawu z PageSpeed Insights na stronie publicznej (telefon: różnica kilku punktów w granicach rozrzutu, komputer: ten sam wynik) jest w „Warunkach audytu i odstępstwach”.

Pamięć podręczna stron jest na kopii wyłączona (odpowiadała przed sprawdzeniem hasła), więc każdy pomiar pokazuje stronę wygenerowaną przez PHP. Dla telefonów to jest zwykły stan sklepu (punkt 2.2), dla komputera stan przy pierwszym wejściu po wyczyszczeniu pamięci podręcznej. Koszyk mierzono pusty.

Urządzenie	Strona	Wynik (3 przebiegi)	FCP	LCP	TBT	Odpowiedź serwera	Waga	Żądania
Telefon	główna	63 (56, 65, 63)	2,92 s	11,19 s ¹	54 ms	605 ms	4 268 KB ¹	161
Telefon	kategoria	71 (57, 71, 72)	3,11 s	5,04 s ¹	50 ms	677 ms	4 657 KB ¹	164
Telefon	produkt	66 (67, 66, 66)	3,11 s	8,74 s ¹	84 ms	511 ms	4 762 KB ¹	171
Telefon	koszyk	86 (86, 86, 87)	2,96 s	3,33 s	54 ms	422 ms	1 390 KB	134
Komputer	główna	95 (95, 94, 95)	0,73 s	0,91 s ¹	0 ms	774 ms	4 991 KB ¹	166
Komputer	kategoria	90 (81, 93, 90)	0,94 s	1,38 s ¹	0 ms	883 ms	4 855 KB ¹	166
Komputer	produkt	96 (94, 96, 97)	0,75 s	0,84 s ¹	0 ms	735 ms	7 496 KB ¹	189
Komputer	koszyk	96 (98, 96, 95)	0,82 s	0,95 s	0 ms	433 ms	1 390 KB	136

¹ Zdjęcia na kopii audytowej pochodzą z plików na serwerze i z pliku hurtowni, część odpowiada błędem 404, bo zasobnik ze zdjęciami sklepu przestał działać razem z domeną. Waga i LCP stron ze zdjęciami nie opisują sklepu. Przesunięcie układu (CLS) na wszystkich stronach 0-0,043, czyli w normie.



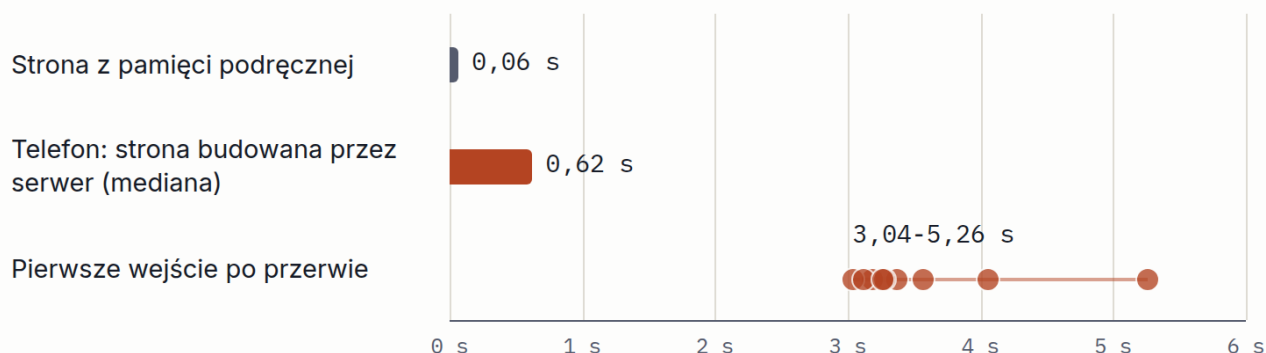
Strona główna w Lighthouse, przebieg z medianą wyniku: z lewej telefon (63), z prawej komputer (95). Czas największego elementu na tej stronie zależy od zdjęć, więc nie opisuje sklepu (przypis ¹). Kopia audytowa, 23.09.2026.

Czas odpowiedzi serwera, seria `curl` z komputera audytora (strona główna i produkt, profil telefonu i komputera, 10 próbek, UTC+2: dziewięć 23.09.2026 w godz. 14:12-22:32 co 53-68 minut, dziesiąta 24.09.2026 w godz. 08:39-08:43):

Urządzenie	Strona	Mediana	Najkrótszy	Najdłuższy	Pierwsze żądanie po przerwie
Telefon	główna	0,65 s	0,54 s	0,73 s	3,18 s i 3,37 s
Telefon	produkt	0,57 s	0,50 s	0,66 s	4,06 s, 3,04 s i 3,57 s
Komputer	główna	0,83 s	0,72 s	1,00 s	3,12 s i 3,28 s
Komputer	produkt	0,72 s	0,61 s	1,85 s	3,27 s i 5,26 s

Mediana, najkrótszy i najdłuższy czas liczone z żądań wysłanych 1-2 minuty po poprzednim, 7-8 na wiersz. Ostatnia kolumna to pierwsze żądanie próbki, po 38-64 minutach bez ruchu na kopii (w dziesiątej próbce po 10 godzinach, pierwsze żądanie tego dnia).

Dla porównania strona podana z pamięci podręcznej: 0,061 s (główna) i 0,062 s (produkt), pomiary pojedyncze 23.09.2026 13:17-13:19 UTC+2, zanim pamięć podręczną na kopii wyłączono. Pierwsze wejście po kilkunastu minutach bez ruchu trwało 3,28 s (14:05 UTC+2) i 3,64 s (13:17 UTC+2, pierwsze żądanie po uruchomieniu kopii).



Czas odpowiedzi serwera na kopii audytowej: strona z pamięci podręcznej 0,06 s, strona budowana przez serwer na telefonie 0,62 s (mediana obu stron z tabeli), pierwsze wejście po przerwie 3,04-5,26 s (dziewięć pomiarów z ostatniej kolumny tabeli).

2.2 Telefony nie dostają stron z pamięci podręcznej ani kodu po optymalizacji priorytet: wysoki

Wtyczka LiteSpeed Cache 7.1 ma włączoną osobną pamięć dla telefonów. Działa to tylko wtedy, gdy w pliku `.htaccess` stoi jej reguła oznaczająca żądania z telefonów. Bez tej reguły wtyczka oznacza każde żądanie z telefonu jako niepodlegające pamięci podręcznej (kod wtyczki: `src/control.cls.php`, linie 738-758), a strona niepodlegająca pamięci podręcznej nie przechodzi też optymalizacji plików CSS i JS.

Kopia pliku `.htaccess` z 22.06.2025 ma tę regułę; plik działający do wyłączenia sklepu (ostatnia zmiana 04.09.2025, przy konfiguracji zapory Wordfence) już jej nie ma. Na tym samym serwerze i z tymi samymi regułami kopia audytowa odpowiada tak (23.09.2026 14:09-14:10 UTC+2):

- profil komputera: nagłówek `x-litespeed-cache-control: public,max-age=604800`, 57 plików CSS z katalogu optymalizacji wtyczki, w kodzie strony adnotacja wtyczki o optymalizacji i zapisie do pamięci;
- profil telefonu: `x-litespeed-cache-control: no-cache` i `cache-control: no-cache, no-store`, 57 oryginalnych plików CSS, adnotacja „Page uncached”.

Skutki na telefonie: każda strona kategorii i produktu jest generowana od nowa (czasy w punkcie 2.1); Lighthouse widzi 9-12 plików blokujących pierwsze wyświetlenie wobec 2-8 na komputerze i szacuje, że usunięcie ich z tej drogi skróci pierwsze wyświetlenie treści o 1,7-2,15 s; nagłówek `no-store` wyłącza przeglądarkę pamięć stron, więc powrót przyciskiem „wstecz” z produktu do kategorii pobiera i generuje kategorię od nowa.

Rekomendacja naprawy.

Zapisać ponownie ustawienia LiteSpeed Cache, żeby wtyczka odtworzyła regułę dla telefonów, i sprawdzić, że druga wizyta telefonem na tej samej stronie dostaje ją z pamięci podręcznej.

2.3 Każdy nowy odwiedzający wczytuje pierwszą stronę dwa razy priorytet: wysoki

W LiteSpeed Cache włączony jest tryb gościa. Przeglądarka bez ciasteczka wtyczki dostaje stronę, po czym skrypt wtyczki pyta serwer o ciasteczko i przeładowuje stronę (kod: `lib/guest.cls.php`, linie 49-81, i `assets/js/guest.min.js`). Przeglądarki przedstawiające się jako narzędzia pomiarowe (Lighthouse, GTmetrix, Google, Pingdom, roboty) są z tego wyłączone, więc testy tego nie pokazują.

Potwierdzenie w pomiarze: dwa pierwsze przebiegi Lighthouse, jeszcze bez ciasteczka wtyczki, wczytały dokument strony dwa razy, razem z zapytaniem o ciasteczko; po jego ustawieniu kolejne przebiegi wczytywały stronę raz. Pierwsze wejście z reklamy albo z wyszukiwarki to więc dwa pełne wczytania, na telefonie oba generowane przez serwer.

Rekomendacja naprawy.

Wyłączyć w LiteSpeed Cache tryb gościa i optymalizację gościa, a potem sprawdzić wizytą z nowego profilu przeglądarki, że strona wczytuje się raz.

2.4 Cache obiektowy ustawiony, ale nie działa priorytet: wysoki

W LiteSpeed Cache wybrany jest cache obiektowy Redis, ale w katalogu `wp-content` nie ma pliku `object-cache.php`, bez którego WordPress go nie używa. Na koncie hostingu rozszerzenie PHP `redis` jest załadowane i jest katalog usługi Redis (odczyt z serwera 23.09.2026). Bez cache obiektowego strony poza pamięcią podręczną (telefony, koszyk, kasa, konto) za każdym razem wykonują pełen zestaw zapytań do bazy. Czy usługa Redis działa i jaki ma limit pamięci, rozstrzygnie test połączenia w ustawieniach wtyczki.

Rekomendacja naprawy.

Włączyć cache obiektowy z adresem Redis podanym przez hosting i porównać czas odpowiedzi serwera przed zmianą i po niej.

2.5 Waga kodu strony priorytet: wysoki, projekt

Strona produktu na telefonie bez zdjęć to 148 plików i ok. 1,26 MB: skrypty 575 KB (59 plików), fonty 393 KB (24), style 246 KB (61), HTML 47 KB. Na każdej stronie ładuje się 56-61 plików CSS i 50-60 plików JS; Lighthouse wskazuje 134-158 KB nieużywanego CSS i 74-168 KB nieużywanego JS na stronę. Dwa pliki ikon Font Awesome ważą razem 254 KB i ładują się wszędzie.

Fonty pochodzą z serwerów Google (8-9 plików) i dodatkowo z lokalnych kopii Elementora (Roboto, Poppins, Roboto Slab); na kopii audytowej kolejne 9-11 plików fontów wskazuje starą domenę i się nie wczytuje, więc w sklepie fontów było więcej. Największy pojedynczy skrypt to Google Tag Manager (172 KB).

Skrypt banera zgody (CookieYes, 34 KB) blokuje wyświetlenie każdej strony: ok. 1,07 s w symulacji telefonu i ok. 0,26 s na komputerze. Wtyczka wstawia go jako pierwszy skrypt celowo, żeby mogła zatrzymać inne skrypty przed zgodą, więc nie należy go odraczać; zysk da dopiero przegląd narzędzia zgody. Strony mają 1 266-2 927 elementów, a przetwarzanie na telefonie zajmuje procesorowi 1,7-3,1 s.

▲ Wyeliminuj zasoby blokujące renderowanie — Szacunkowe przyspieszenie: 1900 ms	▼
▲ Ogranicz nieużywany kod CSS — Szacunkowe zmniejszenie rozmiaru: 147 KiB	▼
▲ Minifikuj CSS — Szacunkowe zmniejszenie rozmiaru: 8 KiB	▼
▲ Zminimalizuj aktywność głównego wątku — 2,6 s	▼
▲ Minifikuj JavaScript — Szacunkowe zmniejszenie rozmiaru: 32 KiB	▼
▲ Zmień rozmiar obrazów — Szacunkowe zmniejszenie rozmiaru: 5 KiB	▼
▲ Unikaj wyświetlania starszych skryptów JavaScript w nowoczesnych przeglądarkach — Szacunkowe zmniejszenie rozmiaru: 37 KiB	▼
▲ Ogranicz nieużywany JavaScript — Szacunkowe zmniejszenie rozmiaru: 100 KiB	▼
▲ Strona uniemożliwiła przywrócenie pamięci podręcznej stanu strony internetowej — 4 przyczyny niepowodzenia	▼
■ Wyświetlaj zasoby statyczne, stosując efektywne zasady pamięci podręcznej — Znaleziono 121 zasobów	▼

Lighthouse, strona produktu na telefonie, przebieg z medianą wyniku (66): zalecenia z szacowaną oszczędnością. Na górze pliki blokujące pierwsze wyświetlenie, dalej m.in. nieużywany CSS i JavaScript oraz 121 plików bez długiego czasu przechowywania (punkt 2.6). Kopia audytowa, 23.09.2026.

Rekomendacja naprawy.

Odchudzić kod strony jako osobny projekt: zasoby motywu i wtyczek ładować tylko tam, gdzie są używane, zostawić jedną bibliotekę ikon i podawać fonty z własnego serwera (rozdział 7).

2.6 Pozostałe

- Brak reguł pamięci przeglądarki dla plików statycznych priorytet: średni: Lighthouse znajduje 106-121 plików na stronę bez długiego czasu przechowywania. Blok tych reguł zniknął z `.htaccess` razem z blokiem z punktu 2.2.
- Zdjęcie największego elementu strony głównej na telefonie ma atrybut `loading="lazy"` priorytet: średni, więc przeglądarka pobiera je później, niż mogłaby.
- Zapytania o koszyk przy każdej stronie priorytet: średni: wtyczka koszyka bocznego (Side Cart WooCommerce 2.6.6) po wczytaniu każdej strony pyta serwer o zawartość koszyka, także gdy koszyk jest pusty; na stronie głównej, w koszyku i na stronie potwierdzenia drugie takie zapytanie wysyła WooCommerce. To zapytania POST, których pamięć podręczna stron nie obsługuje, więc nawet strona podana z pamięci kosztuje serwer jedno albo dwa wywołania PHP. Na kopii audytowej trwały 0,18-1,0 s (mediana 0,26 s, 30 zapytań w czterech przejściach ścieżki zakupu), najdłużej przy pierwszym wejściu na komputerze.

Rekomendacja naprawy.

Przy ponownym zapisie ustawień LiteSpeed Cache (punkt 2.2) włączyć też pamięć przeglądarki. Pierwsze zdjęcia strony głównej wyłączyć z leniwego ładowania. Koszyk odświeżać tylko wtedy, gdy klient coś do niego dodał, z jednego źródła.

Czego nie sprawdzono: wyników PageSpeed Insights z serwerów Google (kopia za hasłem); danych z prawdziwych urządzeń (Chrome UX Report, Search Console; domena wygaśła); wagi i formatu zdjęć sklepu (zasobnik niedostępny); trafień w pamięć podręczną przy ruchu (jedna para pomiarów); zachowania pod obciążeniem; zapytań do bazy na poszczególnych stronach.

3. Bezpieczeństwo

3.1 Wersje i opublikowane luki priorytet: krytyczny dla rdzenia, wysoki dla wtyczek

WordPress 6.8.1 (aktualna poprawka tej linii: 6.8.10, najnowsza wersja: 7.1.2), WooCommerce 9.9.3 (najnowsza 11.1.2), PHP 8.3 (tylko poprawki bezpieczeństwa, do 31.12.2027). Przegląd publicznych baz luk (Wordfence, Patchstack, WPScan, NVD, GitHub Security Advisories) wobec zainstalowanych wersji 38 składników (35 wtyczek, rdzeń, motyw Porto, PHP), stan 23.09.2026, bez żadnych żądań do sklepu: 92 luki w 19 składnikach, 14 składników bez znanych luk dla tej wersji, 5 nie do sprawdzenia (brak w publicznych bazach: wtyczki płatne, pisane na zamówienie albo mało znane). Pasma CVSS: krytyczne 2, wysokie 27, średnie 55, niskie 8. Bez logowania do wykorzystania 45, z czego 10 wymaga działania ofiary (kliknięcia w przygotowany link).

Składnik	Zainstalowana	Najnowsza	Luki	W tym bez logowania
Rank Math SEO	1.0.246	1.0.279	13	4
WordPress (rdzeń)	6.8.1	6.8.10 / 7.1.2	12	3
Customer Reviews for WooCommerce	5.78.1	5.122.0	11	8
Fluent Forms	6.0.4	6.2.14	10	4
CTX Feed	6.5.66	8.0.26	6	1
YITH WooCommerce Wishlist	4.6.0	4.18.1	5	5
LiteSpeed Cache	7.1	7.9.1	5	4
WooCommerce	9.9.3	11.1.2	5	1
pozostałe 11 składników			25	15

Dwie luki krytyczne:

- Rdzeń WordPressa, CVE-2026-87902** (CVSS 4.0: 9.2, bez logowania): obejście ścieżki przy wyborze szablonu strony. Poprawka w 6.8.10 z 22.09.2026; według Patchstack pierwsza próba wykorzystania pojawiła się tego samego dnia o 19:44 UTC+2, kilka godzin po publikacji poprawki. Według opublikowanego opisu wykonanie kodu wymaga motywu z katalogiem `page-*` i dostępnego na serwerze pliku `pearcmd.php`. Żaden z zainstalowanych motywów (Porto, motyw potomny, cztery motywy domyślne) nie ma takiego katalogu (sprawdzone w plikach), więc ta droga jest tu zamknięta. Aktualizacja i tak jest pierwsza, bo to poprawka w tej samej linii wersji, a atakujący już próbują tę lukę wykorzystać.
- CTX Feed, CVE-2026-66709** (CVSS 9.1): wykonanie kodu z konta z rolą kierownika sklepu. W sklepie jest jedno takie konto, bez weryfikacji dwuetapowej.

Rekomendacja naprawy.
Od razu zaktualizować WordPress do 6.8.10. Wtyczki, motyw i WooCommerce zaktualizować jako osobny projekt z testem ścieżki zakupu na kopii sklepu, zaczynając od wtyczek z najpoważniejszymi lukami (rozdział 7).

3.2 Aktualizacje nie miały jak przyjść priorytet: wysoki

25 z 27 wtyczek z wordpress.org jest w tyle, m.in. Elementor 3.29.2 (120 nowszych wydań), WooCommerce 9.9.3 (98), CTX Feed 6.5.66 (93), Customer Reviews (60), Rank Math (42), Fluent Forms (37), wtyczka płatności Stripe (36). Daty katalogów w kopii plików pokazują, że od przeprowadzki 17.06.2025 zmieniły się tylko dwie wtyczki (FluentSMTP 03.09.2025, faktury 04.09.2025). Automatyczne aktualizacje są wyłączone dla wszystkich 35 wtyczek.

Rdzeń WordPressa domyślnie instaluje poprawki bezpieczeństwa sam, ale tu nie mógł: w katalogu strony leży repozytorium kodu `.git`, a WordPress uznaje wtedy stronę za prowadzoną przez programistę i pomija wszystkie aktualizacje w tle (`wp-admin/includes/class-wp-automatic-updater.php`, funkcja `is_vcs_checkout`). Informuje o tym tylko ekran „Zdrowie witryny”. W bazie nie ma śladu, żeby automat kiedykolwiek zakończył aktualizację.

Rekomendacja naprawy.

Usunąć repozytorium kodu z katalogu strony (punkt 3.5), żeby rdzeń znów sam instalował poprawki bezpieczeństwa, a wtyczki aktualizować według stałego harmonogramu z testem na kopii.

3.3 Zapora bez powiadomień, logowanie bez drugiego składnika priorytet: wysoki

Wordfence 8.0.5 (wersja darmowa) nie ma wpisanego adresu do powiadomień. Licznik ostrzeżeń wtyczki pokazuje 42; wtyczka zwiększa go, zanim sprawdzi adres, a przy pustym adresie kończy wysyłkę bez błędu (`lib/wordfenceClass.php`, linie 7442-7445). W archiwum poczty właściciela nie ma ani jednego ostrzeżenia Wordfence z tego sklepu.

Ostatnie skanowanie (szybkie, 12.06.2026 06:11 UTC+2) zgłosiło 26 nowych problemów; otwartych jest 27 (12 krytycznych aktualizacji wtyczek, 10 średnich, 3 motywów, 1 WordPressa, 1 pominięta ścieżka). Porównywanie plików wtyczek i motywów z oryginałami z wordpress.org jest wyłączone, więc zmiana w ich kodzie nie zostałaby zgłoszona, o ile nie pasuje do znanego wzorca złośliwego kodu.

Weryfikacji dwuetapowej nie ma żadne konto. W dzienniku logowań (09.2024-05.2026) jest 19 udanych logowań i 156 nieudanych, wszystkie na nieistniejące nazwy użytkownika (148 razy „admin”). Na stronie logowania wystąpiły dwa błędy krytyczne PHP (01.05.2026 12:11 i 03.05.2026 18:46 UTC+2): Wordfence nie obsłużył adresu IP zawierającego niedozwolony znak; dotyczyły pojedynczych żądań (źródło: wiadomości trybu odzyskiwania WordPressa).

Rekomendacja naprawy.

Wpisać w Wordfence adres powiadomień, który ktoś czyta, włączyć weryfikację dwuetapową dla administratora i kierownika sklepu, blokować od razu próby logowania na nieistniejące nazwy i włączyć porównywanie plików wtyczek i motywów.

3.4 Klucze usług w kodzie i w bazie priorytet: wysoki

Klucz API OpenAI jest zapisany jawnie w kodzie wtyczki pisanej na zamówienie („WooCommerce SEO Optimizer”, `wp-content/plugins/woo-seo-optimizer/index.php`, linia 163) i w dwóch ustawieniach w bazie; to ten sam klucz (porównanie skrótów, bez wypisywania wartości). W dniu audytu klucz był aktywny (sprawdzone zapytaniem tylko do odczytu o listę modeli), więc każdy, kto miał kopię plików albo bazy, mógł go używać na koszt właściciela konta; to był jedyny priorytet krytyczny w tym punkcie. Właściciel odwołał klucz 24.09.2026, a to samo zapytanie zwróciło wtedy błąd autoryzacji (09:07 UTC+2).

W bazie leżą też klucze produkcyjne Stripe, token konta CookieYes, tokeny dostępu Rank Math do usług Google i hasło skrzynki SMTP, a w `wp-config.php` klucze zasobnika Cloudflare R2.

Rekomendacja naprawy.

Odwołać pozostałe klucze u dostawców (w działającym sklepie: wymienić), usunąć klucz OpenAI z kodu wtyczki i z bazy, a klucze trzymać w konfiguracji serwera, nie w kodzie.

3.5 Pliki w katalogu strony, których tam nie powinno być priorytet: krytyczny

- Repozytorium `.git`: 64 zapisy z okresu 05.09.2024-13.06.2025, ok. 400 MB. W historii jest m.in. kod wtyczki pisanej na zamówienie (plik zmieniany w 5 zapisach); nie ma w niej `wp-config.php` ani zrzutów bazy. Plik `.htaccess` sklepu do dnia wyłączenia nie blokował dostępu do `/.git`, a serwer wydaje pliki ukryte (następny punkt), więc repozytorium mogło być do pobrania z zewnątrz. Nie da się tego potwierdzić, bo logi dostępu z okresu działania sklepu już nie istnieją.
- 5 998 plików `.listing` po kopiowaniu przez FTP; każdy zawiera spis plików swojego katalogu z rozmiarami i datami. Serwer wydaje je każdemu: `/.listing` na kopii audytowej odpowiada kodem 200 (2 313 bajtów, 23.09.2026 14:08 UTC+2). To spis zawartości katalogów mimo wyłączzonego listowania.
- `wp-content/debug.log` (111 KB, z 04.03.2025), `cf-images.log` w katalogu wgrywanych plików (4,8 MB, 104 629 linii), kopie `.htaccess`, pliki konfiguracyjne edytora kodu.

Rekomendacja naprawy.

Usunąć wszystko z listy z katalogu strony (repozytorium, jeśli potrzebne, trzymać poza nim) i zostawić regułę serwera blokującą pliki ukryte i kopie.

3.6 Konfiguracja serwera i WordPressa

- Brak pliku `.htaccess` w `wp-content/uploads` blokującego wykonanie PHP priorytet: wysoki: plik PHP wgrany przez lukę w którejś z wtyczek uruchomi się.
- Brak nagłówków bezpieczeństwa (HSTS, `Content-Security-Policy`, `X-Frame-Options`, `X-Content-Type-Options`, `Referrer-Policy`, `Permissions-Policy`); nagłówek `x-powered-by` podaje wersję PHP priorytet: średni. Żadna z wersji pliku `.htaccess` sklepu nie ustawia nagłówków, a odpowiedź kopii audytowej na tym samym serwerze ich nie ma (23.09.2026 14:09 UTC+2).
- Edycja plików PHP z panelu WordPressa włączona (brak `DISALLOW_FILE_EDIT`) priorytet: średni.
- W porządku: rejestracja kont zamknięta, trzy konta (administrator, kierownik sklepu, klient), brak haseł aplikacji, unikalne klucze i sole w `wp-config.php`, katalog faktur zamknięty regułą serwera, w katalogu drugiej wtyczki faktur brak plików PDF.

Rekomendacja naprawy.

Dodać w katalogu `wp-content/uploads` regułę blokującą wykonanie PHP, ustawić nagłówki bezpieczeństwa i ukryć wersję PHP, a edycję plików z panelu wyłączyć stałą `DISALLOW_FILE_EDIT`.

3.7 Kopie zapasowe priorytet: wysoki

W WordPressie jest jeden wpis Duplicatora z 27.09.2024, a plik tej kopii nie istnieje. Innej wtyczki kopii nie ma. Kopii po stronie hostingu nie sprawdzano.

Rekomendacja naprawy.

Potwierdzić u hostingu zakres i czas przechowywania kopii, odtworzyć jedną na osobnym adresie i usunąć martwy wpis Duplicatora.

Czego nie sprawdzono: kopii zapasowych i kont w panelu hostingu i FTP; pięciu wtyczek, których nie ma w publicznych bazach luk; czy klucz OpenAI jest w historii repozytorium; dostępności XML-RPC i ujawniania nazw użytkowników; logów dostępu z okresu działania (nie istnieją). Luk nie testowano żadaniami: ustalenia pochodzą z odczytu wersji wobec opublikowanych komunikatów.

4. Konfiguracja sklepu

4.1 Stany magazynowe i import z hurtowni priorytet: krytyczny

Produkty i stany przychodzą z hurtowni przez WP All Import Pro, a import uruchamia zewnętrzny harmonogram (cron-job.org). Ostatni zakończony import ruszył 15.06.2025 o 00:00 UTC+2, a przebieg trwał ok. 1 h 52 min (6 308 produktów zaktualizowanych, 9 157 wierszy pominiętych).

Tego samego dnia o 15:02 UTC+2 harmonogram wyłączył zadanie przetwarzania importu po 26 kolejnych błędach serwera (HTTP 500) i wysłał o tym e-mail. Import uruchomiony o północy 16.06.2025 nie został przetworzony i do końca działania sklepu pozostał w stanie „uruchomiony”. Zadanie uruchamiające import harmonogram wyłączył w nocy z 7 na 8.07.2026 (08.07.2026 00:01 UTC+2, błąd 404). W ten sam sposób harmonogram wyłączał zadanie przetwarzania już wcześniej (30.12.2024, 30.03.2025). Przyczyny błędów 500 nie da się dziś ustalić, bo logi serwera z tamtego okresu nie istnieją; zbiega się to z przeprowadzką sklepu na nowy serwer (12-17.06.2025).

Stany sklepu wobec pliku hurtowni pobranego 23.09.2026 ok. 08:45 UTC+2 (dopasowanie po SKU, 7 330 produktów):

Stan w sklepie	Hurtownia ma na stanie	Hurtownia ma 0 szt.	Brak w ofercie hurtowni	Razem
dostępny	3 593	484	611	4 688
brak (ukryty)	580	560	1 502	2 642

1 095 z 4 688 produktów oznaczonych jako dostępne (23%) hurtownia dziś nie ma; 580 produktów, które ma, sklep ukrywa (ustawienie „ukryj produkty niedostępne” włączone). Porównanie mierzy rozjazd po 15 miesiącach bez synchronizacji, nie stan z dnia wyłączenia. Porównanie obejmuje tylko dostępność, bo sklep przelicza ceny przy imporcie (m.in. dolicza koszt wysyłki produktom wysyłanym osobno).

Dostępne w sklepie: 4 688

hurtownia ma na stanie	0 szt.	poza ofertą
3 593	484	611

Ukryte w sklepie jako niedostępne: 2 642

na stanie	0 szt.	poza ofertą
580	560	1 502

■ czerwone: stan w sklepie rozjechany ze stanem hurtowni

Ta sama tabela jako wykres. Czerwone odcinki to rozjazd, który widzi klient: 1 095 produktów oznaczonych jako dostępne, których hurtownia nie ma, i 580 ukrytych, które hurtownia ma na stanie.

Rekomendacja naprawy.

Ustalić przyczynę błędów przetwarzania, uruchamiać import z cronu serwera i wysyłać powiadomienie na czytany adres, gdy import nie zakończy się w ciągu doby.

4.2 Stan w dniu wyłączenia

Tryb „wkrótce” (coming soon) włączony, wszystkie metody płatności wyłączone. To zgodne z zamknięciem sklepu, nie usterka. W historii 11 zamówień (08.2024-09.2025) klienci wybierali płatność za pobraniem (7) i Przelewy24 obsługiwane przez wtyczkę Stripe (4). Wtyczka Stripe pozostała w trybie produkcyjnym z kluczami w bazie (punkt 3.4).

W ustawieniach przelewu zostały dane przykładowe z WooCommerce: „Przykładowe konto bankowe”, numer 555 555 555, bez numeru IBAN **priorytet: krytyczny w chwili włączenia przelewu**. Przy wyłączonym przelewie nie ma to skutku, ale po jego włączeniu klient dostaje te dane na stronie potwierdzenia zamówienia i w mailu, więc nie wie, dokąd wpłacić (sprawdzone zamówieniem testowym, punkt 4.3).

Rekomendacja naprawy.

Przed włączeniem przelewu wpisać rachunek sklepu i sprawdzić mail do klienta zamówieniem testowym.

4.3 Kasa i ścieżka zakupu

Zakup bez konta włączony, kasa w jednym kroku (klasyczna, na krótkim kodzie WooCommerce), koszt wysyłki widoczny w koszyku (przykład: produkt 290,00 zł, wysyłka 19,00 zł, razem 309,00 zł z 23% VAT).

Ścieżka zakupu (strona główna, sklep, produkt, dodanie do koszyka, koszyk, kasa) sprawdzona na kopii audytowej: dwa przejścia na telefonie (390 px) i dwa na komputerze (1440 px), 23.09.2026 15:12-15:25 UTC+2. Wszystkie strony odpowiadają, serwer oddaje dokument strony w 0,38-0,94 s (bez pamięci podręcznej, która na kopii jest wyłączona; jeden wyjątek: pierwsza strona przejścia z zamówieniem testowym, 3,2 s), dodanie do koszyka działa bez przeładowania strony. W każdym przejściu pierwsza strona wczytała się dwa razy (punkt 2.3). W stanie z dnia wyłączenia kasa pokazuje komunikat „Przykro nam, ale nie jest udostępniona żadna metoda płatności” (punkt 4.2).

Koszyk > **Zamówienie**

> Zamówienie zakończone

Powracający klient? **Logowanie**

Masz kupon? **WPROWADŹ SWÓJ KOD**

Dane płatności

Imię *

Nazwisko *

Nasze dane bankowe

Przykładowe konto bankowe:

• Numer konta: **555 555 555**

Twoje zamówienie

PRODUKT

Bezprzewodowy
głośnik Bluetooth
40W Bang SE czarny
x 1

290,00 zł

Kwota:

290,00 zł

Z lewej kasa na telefonie (390 px): jeden krok, zakup bez zakładania konta. Z prawej strona potwierdzenia zamówienia testowego z przykładowymi danymi konta z WooCommerce (punkt 4.2). Kopia audytowa, 23.09.2026.

Na obu urządzeniach jedyny błąd JavaScript pochodzi ze skryptu banera zgody (CookieYes), przypisanego do starej domeny; po zablokowaniu tego skryptu strona nie zgłasza błędów JavaScript. To skutek kopii, nie usterka sklepu.

4.4 Wysyłka

W ustawieniach WooCommerce jest jedna strefa (Polska) i „Kurier” za 1,00 zł, ale faktyczny koszt liczy kod motywu potomnego (porto-child/functions.php , filtry woocommerce_package_rates , linie 180 i 808): 19 zł za paczkę (własne pole w ustawieniach), koszt indywidualny z pola produktu, paczki łączone i osobne. Produktom wysyłanym osobno import dolicza koszt wysyłki do ceny (linia 144) i oznacza je jako „darmowa dostawa”. Działa, ale ustawienia WooCommerce pokazują coś innego niż płaci klient, a zmiana cennika wysyłki wymaga programisty `priority: średni, rozdział 5`. Etykieta w koszyku „Kurier (1 łączone paczki)” jest niegramatyczna `priority: niski`.

PRODUKT	CENA	ILOŚĆ	KWOTA
 Bezprzewodowy głośnik Bluetooth 40W Bang SE czarny	290,00 zł	- 1 +	290,00 zł

Kod kuponu

WYKORZYSTAJ KUPON

ZAKTUALIZUJ KOSZYK

PODSUMOWANIE KOSZYKA

Kwota 290,00 zł

Szczegóły wysyłki

Całkowita waga 2.5 kg

Paczki łączone 1

Koszt paczek łączonych 19,00 zł

Szacowany koszt wysyłki 19,00 zł

Wysyłka

Kurier (1 łączone paczki): 19,00 zł

Metody wysyłki zostaną zaktualizowane podczas składania zamówienia.

Polska

Koszyk z produktem za 290,00 zł. Szczegóły wysyłki (waga, paczki łączone, 19,00 zł) liczy kod motywu potomnego, nie ustawienia WooCommerce; pod nimi etykieta „Kurier (1 łączone paczki)”. Kopia audytowa, 23.09.2026.

Rekomendacja naprawy.

Przy przenoszeniu logiki sklepu do wtyczki (punkt 5.5) przenieść cennik wysyłki do ustawień, które pokazują to, co płaci klient, i poprawić odmianę w etykiecie koszyka.

4.5 Podatki, maile, ceny

- Podatki: VAT 23%, ceny z podatkiem, liczony wg adresu wysyłki. Bez uwag.
- Maile transakcyjne: szablony domyślne, wszystkie włączone. Nadawca w ustawieniach WooCommerce to adres w innej domenie niż sklep, a wtyczka FluentSMTP wysyła przez skrzynkę na serwerze hostingu z adresu w domenie sklepu `priority: niski; ujednolicić`. Dziennik FluentSMTP ma jeden wpis (01.06.2026), więc nie mówi, czy maile do klientów dochodziły.
- Najniższa cena z 30 dni (Omnibus): wtyczka Price History, 30 dni, wyświetlanie przy promocji, historia cen dla wszystkich 7 330 produktów. W sklepie nie było produktów w promocji, więc wyświetlenia nie dało się sprawdzić.

Rekomendacja naprawy.

Ustawić ten sam adres nadawcy w domenie sklepu w WooCommerce i we wtyczce SMTP.

4.6 Analityka i zgody

Tagi Google Analytics 4 i Microsoft Advertising (UET) wstawione w kod strony i ładowane przez mechanizm opóźniania skryptów LiteSpeed Cache; motyw potomny dodaje konwersje rozszerzone Bing w nagłówku i zaproszenie do opinii Google w stopce, oba tylko na stronie potwierdzenia zamówienia. Moduł analityki Rank Math nie wstawia drugiego kodu GA (w porządku). Baner zgody: CookieYes.

- Zakup w Google Analytics 4 **priorytet: średni**: tag GA4 w motywie potomnym ma samą konfigurację, bez zdarzeń sklepu, więc GA4 nie dostaje zakupu ani wartości zamówienia i nie pokaże przychodu ani tego, które źródło ruchu sprzedaje. Wtyczki, które mogłyby te zdarzenia wysyłać, są wyłączone albo bez identyfikatorów (moduł Rank Math bez wstawiania kodu, śledzenie w CTX Feed bez numeru konta). Czy zakup zdefiniowano regułą w panelu GA4 (i tak bez wartości zamówienia), nie sprawdzono.
- Konwersje rozszerzone Bing **priorytet: średni**: kod przekazuje skrót adresu e-mail i numeru telefonu klienta. Numer miał trafiać w formacie międzynarodowym, ale warunek dopisujący „+” nigdy się nie spełnia, a numeru kierunkowego kraju kod nie dodaje, więc dla numerów wpisanych bez „+48” dopasowanie po telefonie prawdopodobnie nie działa.

Rekomendacja naprawy.

Wysyłać do GA4 zdarzenia sklepu z wartością zamówienia, zgodnie z ustawieniem zgody.

W konwersjach Bing poprawić format numeru telefonu i sprawdzić dopasowanie w panelu Microsoft Advertising.

Czego nie sprawdzono: czy tagi czekają na zgodę (baner CookieYes jest przypisany do starej domeny i na kopii audytowej nie działa); dostarczania maili do skrzynek (na kopii poczta jest przechwytywana do pliku); płatności online w działaniu (bramki Stripe wyłączone, klucze usunięte z kopii; przelew sprawdzono zamówieniem testowym); aktualności feedów produktowych CTX Feed (Google Shopping i drugi feed).

5. Porządek w instalacji

5.1 Wtyczki i motywy **priorytet: średni**

35 wtyczek: 32 aktywne, 3 nieaktywne (WP MailHog SMTP, narzędzie programisty do przechwytywania poczty; Better Search Replace; Preserve Page and Taxonomy Hierarchy 0.1 spoza wordpress.org). Filenames to Latin nie ma wydania od 09.08.2020. Dwie wtyczki formularzy (Contact Form 7 i Fluent Forms), obie używane (zgłoszenia zapisane w obu).

Wtyczka pisana na zamówienie „WooCommerce SEO Optimizer” (nagłówek z szablonu „Author: Your Name”) wysyła tytuły i opisy produktów do API OpenAI; klucz w kodzie, punkt 3.4. Cztery nieaktywne motywy domyślne (Twenty Twenty-Two do Twenty Twenty-Five).

Rekomendacja naprawy.

Usunąć nieaktywne wtyczki i trzy z czterech motywów domyślnych, zostawić jedną wtyczkę formularzy i zastąpić porzuconą wtyczkę aktualnym odpowiednikiem.

5.2 Baza danych **priorytet: średni**

Baza ma 153 tabele i ok. 0,9 GB; ok. dwie trzecie to dane wtyczki Offload Media: tabela `wp_acoofm_items` (ok. 490 MB) i 71 wygasłych transientów `acoofm_cached_data_*` w tabeli opcji (razem ok. 163 MB, wygasły 12-13.06.2026). Tabela `wp_postmeta` ma ok. 194 MB.

Opcje ładowane przy każdym żądaniu (autoload) zajmują ok. 1,4 MB wobec progu 800 KB, od którego WordPress ostrzega w „Zdrowiu witryny” (wp-admin/includes/class-wp-site-health.php, linia 2648). Dwie największe, po ok. 0,5 MB, to transjenty bez terminu ważności: `_transient_dirsize_cache` (pamięć rozmiarów katalogów) i `_transient_ptk_patterns` (wzorce bloków pobierane przez WooCommerce). WordPress 6.8 zapisze pierwszy ponownie z terminem ważności, już bez ładowania przy każdym żądaniu; drugi WooCommerce 9.9.3 zapisuje bez terminu, więc po usunięciu wróci (do sprawdzenia po aktualizacji WooCommerce). Rewizji 97, osieroconych metadanych 0.

Rekomendacja naprawy.

Po kopii bazy usunąć wygasłe transjenty i `_transient_dirsize_cache`; porządek w tabeli Offload Media zrobić razem z decyzją z punktu 5.3.

5.3 Zdjęcia produktów priorytet: średni, projekt

Załączników jest 27 542, pliki na serwerze ma ok. 29% z nich. Resztę wtyczka Offload Media przeniosła do zewnętrznego zasobnika (Cloudflare R2) pod adresem w domenę sklepu, który przestał odpowiadać z wygaśnięciem domeny (brak rekordu DNS, 23.09.2026). Zdjęcia zależą więc od domeny, zasobnika i licencji jednej wtyczki. Czy sam zasobnik nadal istnieje, rozstrzygnie konto Cloudflare. Plik hurtowni daje zdjęcia dla 5 217 produktów (71%); dla 2 113 produktów spoza jej oferty zdjęcia są tylko w zasobniku albo w plikach na serwerze.

Rekomendacja naprawy.

Zdecydować, czy zdjęcia zostają w zasobniku, czy wracają na serwer, i przenieść je tak, żeby nie zależały od domeny ani od licencji jednej wtyczki.

5.4 Zadania w tle priorytet: średni

Na koncie hostingu nie ma crona systemowego dla sklepu (odczyt 23.09.2026), więc zadania WordPressa uruchamiają się tylko przy wizytach. Action Scheduler: 250 zadań nieudanych z okresu 09.2024-02.2026, z czego 110 w czerwcu 2025, głównie przebudowa tabel wyszukiwania produktów (125) i czyszczenie pamięci powiązanych produktów (84); 5 zadań oczekuje od 12.06.2026; 22 844 wiersze dziennika.

Rekomendacja naprawy.

Uruchamiać zadania WordPressa z crona serwera co kilka minut zamiast przy wizytach, a nieudane zadania przejrzeć i wyczyścić.

5.5 Logika sklepu w motywie potomnym priorytet: średni, projekt

Plik `porto-child/functions.php` ma 1 239 linii logiki sklepu: wysyłka (punkt 4.4), doliczanie kosztów przy imporcie, teksty stanów, oznaczenia darmowej wysyłki, konwersje Bing, zaproszenia do opinii Google. Zmiana albo uszkodzenie motywu potomnego zmienia ceny wysyłki.

Rekomendacja naprawy.

Przenieść logikę sklepu do osobnej wtyczki z ustawieniami i testem kalkulacji wysyłki.

5.6 Drobne

Ustawienie wtyczki Debug Log Manager wskazuje ścieżkę poprzedniego hostingu (ostrzeżenie w logu PHP); pliki robocze w katalogu strony: punkt 3.5.

Rekomendacja naprawy.

Poprawić ścieżkę w ustawieniach wtyczki Debug Log Manager albo usunąć wtyczkę, jeśli nie jest używana.

Czego nie sprawdzono: które funkcje każdej wtyczki są faktycznie używane; tabel po usuniętych wtyczkach poza tymi z listy największych.

6. Zachowania klientów, w wariancie rozszerzonym

Nie był objęty: raport przykładowy pokrywa wariant podstawowy, a sklep w dniu audytu nie miał ruchu. W wariancie rozszerzonym ten rozdział opisuje miesiąc pomiaru: gdzie klienci klikają, na którym kroku kasy odpadają, z jakich stron trafiają do koszyka i na jakich urządzeniach, każde ustalenie z liczbą sesji.

7. Co poprawić, w jakiej kolejności

Do kliknięcia dziś, bez wykonawcy

- Wordfence: w ustawieniach powiadomień e-mail wpisać adres, który ktoś czyta; w module logowania (Login Security) włączyć weryfikację dwuetapową dla administratora i kierownika sklepu.
- LiteSpeed Cache: na karcie Cache kliknąć „Zapisz zmiany” bez zmieniania ustawień (wtyczka zapisze ponownie swoje reguły serwera, w tym tę dla telefonów); w ustawieniach ogólnych wyłączyć tryb gościa.
- Wtyczki: usunąć trzy nieaktywne (WP MailHog SMTP, Better Search Replace, Preserve Page and Taxonomy Hierarchy).
- Usługa uruchamiająca import (cron-job.org): włączyć ponownie zadania importu i sprawdzić w historii zadania, czy przebieg kończy się bez błędu.

Proste (do zrobienia w tydzień, bez projektu):

Poprawka	Priorytet	Obszar	Co zmieni	Do wyceny przez
Zaktualizować WordPress 6.8.1 do 6.8.10, po kopii plików i bazy	krytyczny	Bezpieczeństwo	Zamyka 12 luk rdzenia, w tym krytyczną, którą atakujący próbują wykorzystać od 22.09.2026	administrator WordPress
Usunąć z katalogu strony repozytorium .git, 5 998 plików .listing, kopie .htaccess, logi i pliki edytora; reguła blokująca pliki ukryte	krytyczny	Bezpieczeństwo	Nikt nie pobierze historii kodu ani spisu plików serwera	administrator WordPress
Wznowić import stanów: przyczyna błędów przetwarzania, uruchamianie z crona serwera, powiadomienie o nieudanym przebiegu	krytyczny	Konfiguracja sklepu	Klient nie zamawia 1 095 produktów, których nie ma; 580 dostępnych wraca do oferty	programista WordPress
Przelew: wpisać rachunek sklepu w miejsce przykładowych danych WooCommerce, zanim metoda wróci do kasy	krytyczny	Konfiguracja sklepu	Klient płacący przelewem dostaje prawdziwy numer konta na stronie potwierdzenia i w mailu	właściciel (sam)
Usunąć klucz OpenAI z kodu wtyczki i z bazy (u dostawcy odwołany 24.09.2026); odwołać pozostałe klucze usług (Stripe, Google, Cloudflare, CookieYes, SMTP)	wysoki	Bezpieczeństwo	Nikt z kopią plików albo bazy nie używa usług na koszt właściciela	właściciel (odwołanie), programista WordPress (kod)
Wordfence: adres powiadomień, weryfikacja dwuetapowa dla administratora i kierownika sklepu, blokada nieistniejących loginów	wysoki	Bezpieczeństwo	Ostrzeżenia docierają; przejęcie konta wymaga drugiego składnika	właściciel (sam)
LiteSpeed Cache: ponowny zapis ustawień (z pamięcią przeglądarki), wyłączenie trybu gościa, sprawdzenie x-litespeed-cache: hit na telefonie	wysoki	Wydajność	Telefony dostają strony z pamięci podręcznej; pierwsza wizyta bez podwójnego wczytania; pliki statyczne z długim czasem przechowywania	właściciel (sam), sprawdzenie: administrator
Włączyć cache obiektowy Redis z testem połączenia	wysoki	Wydajność	Mniej zapytań do bazy na stronach bez pamięci podręcznej (koszyk, kasa, konto)	administrator WordPress

Poprawka	Priorytet	Obszar	Co zmieni	Do wyceny przez
Zablokować wykonywanie PHP w <code>wp-content/uploads</code>	wysoki	Bezpieczeństwo	Plik wgrany przez lukę nie uruchomi się	administrator WordPress
Potwierdzić kopie hostingu próbą odtworzenia; usunąć martwy wpis Duplicatora	wysoki	Bezpieczeństwo	Wiadomo, z czego sklep da się odtworzyć	administrator WordPress
Pierwsze zdjęcie strony głównej bez leniwego ładowania	średni	Wydajność	Przeglądarka pobiera największy element strony od razu	programista WordPress
Cron systemowy zamiast zadań uruchamianych wizytami; wyczyścić 250 nieudanych zadań	średni	Porządek	Zadania w tle wykonują się o czasie, niezależnie od ruchu	administrator WordPress
Nagłówki bezpieczeństwa, ukrycie wersji PHP, <code>DISALLOW_FILE_EDIT</code>	średni	Bezpieczeństwo	Mniej informacji dla skanerów; brak edycji kodu z panelu	administrator WordPress
Usunąć 3 nieaktywne wtyczki i 3 motywy domyślne; zastąpić Filenames to Latin	średni	Porządek	Mniej kodu do aktualizowania i mniej miejsc na lukę	administrator WordPress
Baza: wygasłe transjenty i <code>_transient_dirsize_cache</code>	średni	Porządek	Baza mniejsza o ok. 163 MB; opcje ładowane przy każdym żądaniu mniejsze o ok. 0,5 MB (kolejne 0,5 MB po aktualizacji WooCommerce, do sprawdzenia)	administrator WordPress
Koszyk boczny: odświeżać zawartość koszyka tylko przy niepustym koszyku, jedno źródło odświeżania	średni	Wydajność	Strona podana z pamięci podręcznej nie wywołuje PHP przy każdym wejściu	programista WordPress
Analityka: zdarzenie zakupu z wartością zamówienia w GA4; format numeru telefonu w konwersjach rozszerzonych Bing	średni	Konfiguracja sklepu	GA4 pokazuje przychód i źródła sprzedaży; Microsoft dopasowuje konwersje także po telefonie	programista WordPress
Etykieta wysyłki w koszyku, jeden nadawca maili w WooCommerce i SMTP, ścieżka starego hostingu w ustawieniach	niski	Konfiguracja sklepu	Porządek w tym, co widzi klient i administrator	administrator WordPress

Większe (projekt do wyceny):

Poprawka	Priorytet	Obszar	Co zmienia	Do wyceny przez
Aktualizacja wtyczek, motywu i WooCommerce na kopii z testem ścieżki zakupu, potem produkcja; kolejność wg luk (CTX Feed, Customer Reviews, Rank Math, Fluent Forms), osobno skoki głównych wersji (Elementor 3 do 4, WooCommerce 9 do 11)	wysoki	Bezpieczeństwo	Dotyczy 80 z 92 opublikowanych luk (wtyczki i motyw; dla jednej luki motywu Porto WPScan nie podaje poprawki); sklep wraca do wersji wspieranych	programista WordPress
Odcudzenie kodu strony: moduły Elementora i Porto, zasoby wtyczek tylko na stronach, które ich używają, jedna biblioteka ikon, fonty z własnego serwera, jedna wtyczka formularzy	wysoki	Wydajność	Mniej plików blokujących wyświetlenie; Lighthouse wskazuje 134-158 KB nieużywanego CSS i 74-168 KB nieużywanego JS na stronę	programista WordPress (front-end)
Przeniesienie logiki sklepu z motywu potomnego (1 239 linii) do wtyczki z ustawieniami i testami; ustawienia wysyłki zgodne z tym, co płaci klient	średni	Porządek	Aktualizacja albo zmiana motywu nie zmienia cen wysyłki; cennik zmienia się bez programisty	programista WordPress
Zdjęcia: decyzja zasobnik czy serwer, porządek po Offload Media	średni	Porządek	Zdjęcia nie zależą od domeny i licencji jednej wtyczki; baza mniejsza o ok. 490 MB	programista WordPress

Warunki audytu i odstępowstwa

- **Sklep:** własność wykonawcy audytu, wyłączony 29.07.2026, domena wygaśa 06.09.2026. Od 12-13.06.2026 w bazie nie ma śladów pracy WordPressa (zadania w tle, skanowanie zapory); przyczyny nie ustalano.
- **Kopia audytowa:** pliki sklepu i zrzut jego bazy z 23.09.2026, przywrócone tego dnia na tym samym serwerze (hosting współdzielony, serwer LiteSpeed, PHP 8.3) na osobnym adresie za hasłem, z nagłówkiem i plikiem `robots.txt` zakazującymi indeksowania. Kopia zostanie usunięta z serwera po zakończeniu prac nad raportem. Część odczytów kodu i bazy wykonano na drugiej kopii (ten sam kod i ta sama baza) na komputerze audytora; ścieżkę zakupu i zamówienie testowe wykonano na kopii audytowej.
- **Dane klientów:** zanonimizowane przed wystawieniem na serwer (11 zamówień i 3 zwroty, 8 rekordów klientów, kontakty, zgłoszenia formularzy, notatki zamówień); dzienniki z adresami IP wyczyszczone. Kontrola po anonimizacji: 0 adresów e-mail spoza domen sklepu i jego właściciela w całej bazie (przed: 306 wierszy). Klucze usług usunięte z bazy i kodu kopii. Ustalenia dotyczące danych osobowych klientów przekazano właścicielowi osobno; nie ma ich w tej wersji raportu.
- **Zmiany na kopii:** wyłączony tryb „wkrótce”, żeby sklep dało się mierzyć; poczta przechwytywana do pliku; połączenia z usługami zewnętrznymi zablokowane; zadania zaplanowane WordPressa wyłączone. Na czas zamówienia testowego (23.09.2026 15:17-15:20 UTC+2) włączono przelew; zamówienie anulowano, stan magazynowy wrócił, przelew wyłączono. W przeglądarce, którą przechodzono ścieżkę zakupu, zablokowano wysyłanie danych do Google Analytics i Microsoft Advertising, żeby zamówienie testowe nie trafiło do statystyk sklepu. Rdzenia, wtyczek i motywu nie aktualizowano: zaniechanie jest materiałem audytu.
- **Zdjęcia i fonty:** zdjęcia z plików na serwerze (ok. 29% załączników) i z pliku hurtowni; pozostałe odpowiadają błędem 404. Część fontów wskazuje adres starej domeny i się nie wczytuje (punkt 2.5), więc strony kopii są lżejsze niż w sklepie.

- **Pomiary:** Lighthouse zamiast PageSpeed Insights (kopia za hasłem), z komputera audytora; pamięć podręczna stron wyłączona na kopii, bo odpowiadała przed sprawdzeniem hasła. Zestaw pomiarowy porównano z PageSpeed Insights 23.09.2026 na publicznej stronie wykonawcy audytu, po trzy przebiegi: wynik na telefonie 89 wobec 93 (przebiegi 82-95 wobec 84-98), na komputerze 99 w obu. Różnica median na telefonie mieści się w rozrzucie przebiegów obu narzędzi, więc wyniki z kopii należy czytać z tolerancją kilku punktów. Pojedyncze metryki różnią się w obie strony (telefon: TBT 189 ms wobec 71 ms, LCP 2,86 s wobec 3,08 s), a PageSpeed Insights używa nowszej wersji Lighthouse (13.5 wobec 12.8). Strona porównawcza jest lekka, więc zgodności dla stron z dużą ilością skryptów nie sprawdzono. Ustalenia o kodzie wtyczek pochodzą z odczytu kodu zainstalowanych wersji, o stanie danych z bazy sklepu, o serwerze z dostępu tylko do odczytu na koncie hostingu.
- **Rozdział 6:** nie istnieje (wariant podstawowy, sklep bez ruchu).