

Oryginalne wymagania dotyczące serwisu

Serwis realizuje obsługę katalogu produktów oraz koszyka. Klient serwisu powinien móc:

- dodać produkt do katalogu,
- usunąć produkt z katalogu,
- wyświetlić produkty z katalogu jako stronicowaną listę o co najwyżej 3 produktach na stronie,
- utworzyć koszyk,
- dodać produkt do koszyka, przy czym koszyk może zawierać maksymalnie 3 produkty,
- usunąć produkt z koszyka,
- wyświetlić produkty w koszyku, wraz z ich całkowitą wartością.

Kod, który masz przed sobą, stara się implementować te wymagania z pomocą `Symfony 6.0`.

Twoje zadanie

Scenariusz zadania rekrutacyjnego jest następujący:

Masz przed sobą serwis działający produkcyjnie z dużą liczbą aktywnych użytkowników. Użytkownicy i testerzy serwisu zgłosili następujące problemy i prośby:

- * Od użytkownika: Chcemy móc dodawać do koszyka ten sam produkt kilka razy, o ile nie zostanie przekroczony sumaryczny limit sztuk produktów. Teraz to nie działa.
- * Od testera: Limit koszyka nie zawsze działa. Wprawdzie, gdy podczas naszych testów dodajemy czwarty produkt do koszyka to dostajemy komunikat `Cart is full.`, ale pomimo tego i tak niektóre koszyki w bazie danych mają po cztery produkty.
- * Od użytkownika i testera: Najnowsze (ostatnio dodane) produkty powinny być dostępne na początkowych stronach listy produktów w katalogu.
- * Od użytkownika: Musimy mieć możliwość edycji produktów. Czasami w nazwach są literówki, innym razem cena jest nieaktualna.

Prosimy o naprawienie / implementację.

PS1. Prawdziwym celem zadania jest oczywiście kawałek kodu, który możemy ocenić, a potem porozmawiać o nim w czasie interview "twarzą w twarz".

Przy czym pamiętaj, że liczy się nie tylko napisany kod PHP, ale także umiejętność przedstawienia czytelnego rozwiązania, użycia odpowiednich narzędzi (choćby systemu wersjonowania), udowodnienia poprawności rozwiązania (testy), dobrej oceny tego jak zmiany wpłyną na dotychczasowych użytkowników itd.

PS2. To Twoja okazja na pokazanie umiejętności, więc jeśli uważasz, że w kodzie jest coś nie tak, widzisz więcej błędów, coś powinno być zaimplementowane inaczej, możesz do listy zadań dodać opcjonalny refactoring, albo krótko wynotować swoje spostrzeżenia (może przeprowadzić coś w rodzaju code review?).

Ale robiąc to nie zapominaj o PS1!

Powodzenia!