

Dokumentacja

Projektowanie aplikacji internetowych

Autor: Miłosz Nierodzik

1. Opis projektu

404 Style to aplikacja e-commerce służąca do sprzedaży odzieży i akcesoriów modowych. System umożliwia zarządzanie produktami, kategoriami oraz zamówieniami, a użytkownikom przeglądanie oferty, dodawanie produktów do koszyka oraz składanie zamówień z płatnościami online.

Aplikacja składa się z dwóch głównych części:

- **Frontend** – interfejs użytkownika zbudowany w React przy użyciu Vite
- **Backend** – API REST zbudowane w Node.js z Express i MySQL

Główne funkcje aplikacji:

- Przeglądanie produktów z filtrowaniem i wyszukiwaniem
- System koszyka oraz składania zamówień
- Płatności online przez Stripe
- Panel administracyjny do zarządzania produktami i zamówieniami
- System komentarzy z paginacją i głosowaniem
- Personalizacja motywów kolorystycznych
- Autoryzacja użytkowników (email/hasło oraz OAuth Google/GitHub)
- System ról (user, moderator, admin)

2. Wymagania funkcjonalne

2.1 Użytkownik (User)

- Rejestracja i logowanie do systemu
- Przeglądanie produktów z filtrowaniem według kategorii
- Wyszukiwanie produktów po nazwie i opisie
- Dodawanie produktów do koszyka
- Składanie zamówień z danymi wysyłki
- Przeglądanie historii własnych zamówień
- Dodawanie komentarzy do produktów
- Głosowanie na komentarze (upvote / downvote)
- Zmiana motywu kolorystycznego strony
- Odpowiadanie na komentarze (replies)

2.2 Moderator

- Wszystkie uprawnienia użytkownika
- Moderacja komentarzy w przypisanych kategoriach (zatwierdzanie / odrzucanie)
- Zarządzanie produktami w przypisanych kategoriach

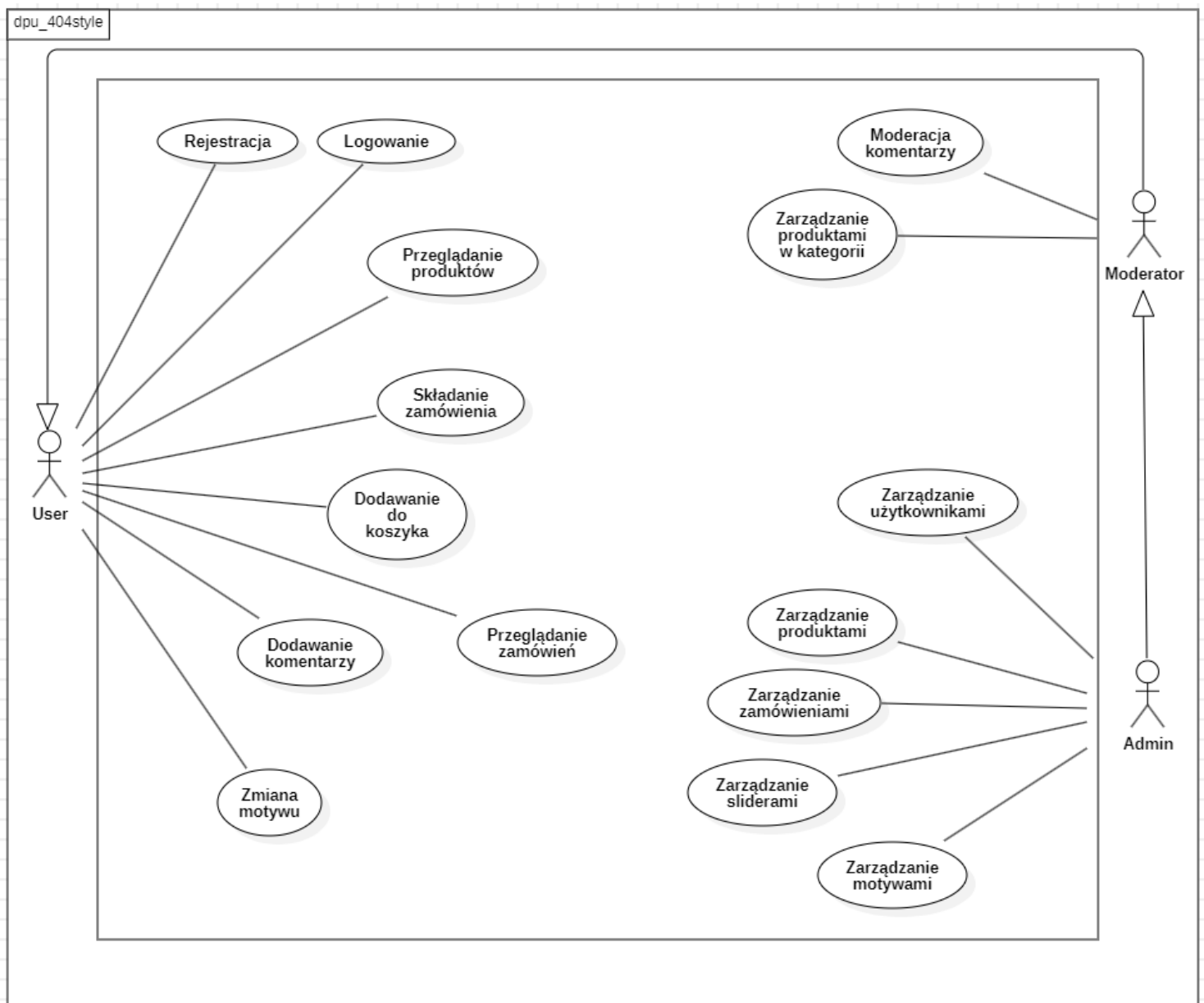
2.3 Administrator

- Wszystkie uprawnienia moderatora
- Zarządzanie użytkownikami (zmiana ról, usuwanie)
- Dodawanie, edycja i usuwanie produktów
- Zarządzanie kategoriami
- Zarządzanie zamówieniami (zmiana statusów)
- Zarządzanie sliderami na stronie głównej
- Zarządzanie motywami kolorystycznymi
- Przeglądanie statystyk sprzedaży
- Przypisywanie moderatorów do kategorii

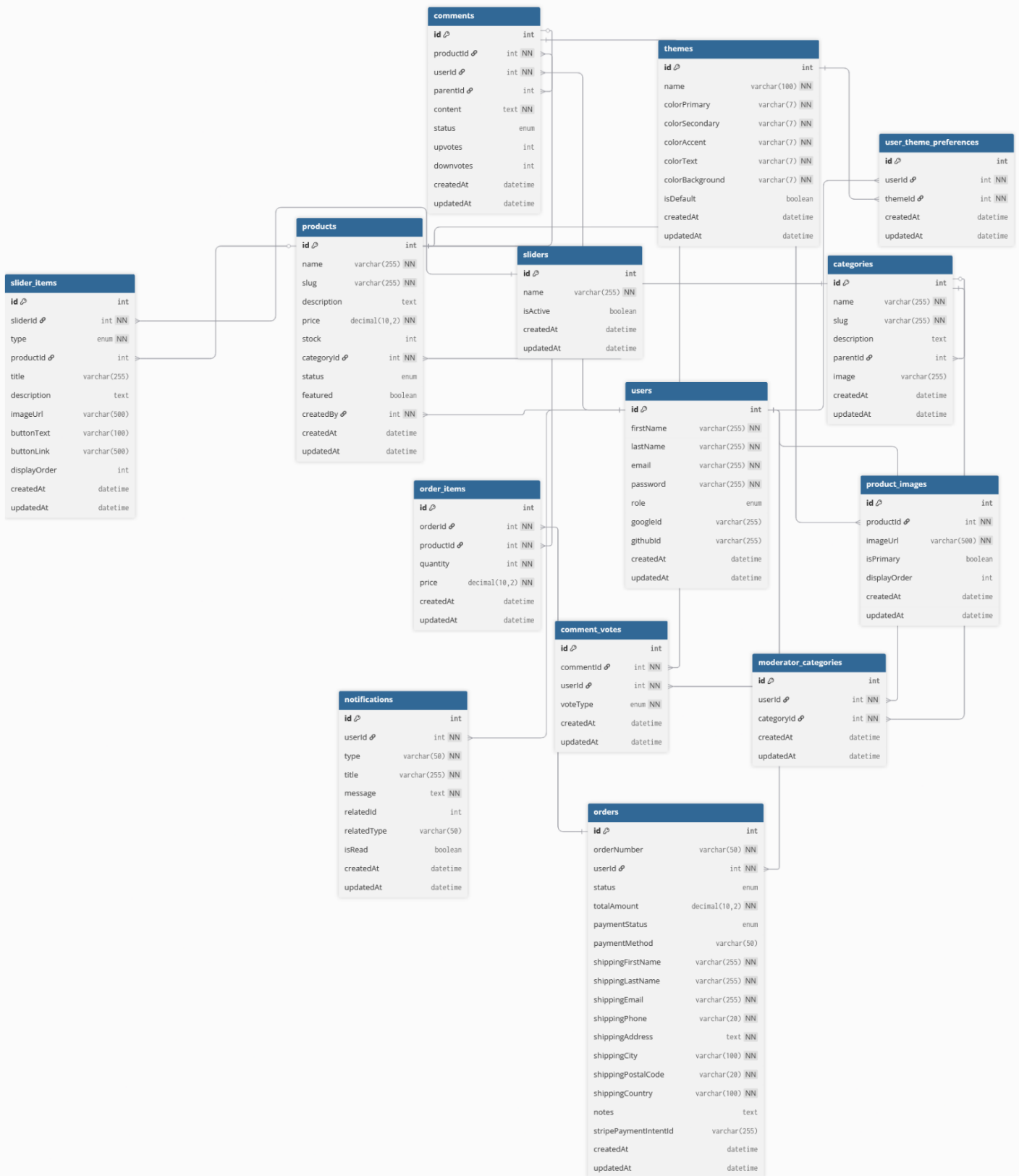
3. Wymagania niefunkcjonalne

- **Bezpieczeństwo** – hasła hashowane, JWT do autoryzacji
- **Responsywność** – interfejs dopasowany do urządzeń mobilnych
- **Walidacja** – sprawdzanie poprawności danych wejściowych
- **Obsługa błędów** – czytelne komunikaty dla użytkownika
- **Paginacja** – ograniczanie liczby danych na stronę

4. Diagram przypadków użycia



5. Schemat bazy danych



6. Technologie

Backend:

- Node.js v22.20.0
- Express 5.1.0
- MySQL 8.0
- Sequelize ORM: 6.37.7
- JWT 9.0.2
- Stripe 19.2.0

Frontend:

- React 19
- Vite 7.1.7
- Tailwind CSS 3.4.18

7. Instrukcja uruchomienia

Szczegółowa instrukcja znajduje się w pliku **README.md** w głównym katalogu projektu.