Wzorce Strukturalne w Projekcie Alegro

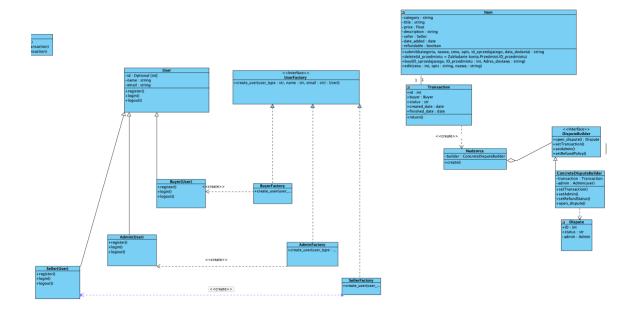
Wzorzec 1: Fasada (Facade)

Motywacja

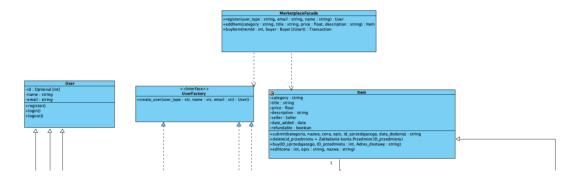
W systemie zarządzania marketplace'em pojawia się wiele operacji: rejestracja użytkowników, dodawanie przedmiotów, kupowanie produktów. Aby uprościć interakcję z tymi funkcjonalnościami i ukryć wewnętrzną złożoność modułów (fabryk użytkowników, modeli przedmiotów i transakcji), zastosowano wzorzec Fasady. Pozwala to wystawiać jeden spójny interfejs (MarketplaceFacade), który znacząco ułatwia korzystanie z systemu i zmniejsza sprzężenie między warstwami.

Konsekwencje: łatwiejsze utrzymanie kodu, uproszczone testowanie funkcji systemowych, możliwość wprowadzenia kolejnych zmian w logice wewnętrznej bez zmieniania interfejsu dla użytkownika.

Struktura przed wdrożeniem



Struktura po wdrożeniu



Współdziałanie

Użytkownik wywołuje metodę MarketplaceFacade.register(), a fasada wewnętrznie wywołuje odpowiednią fabrykę (BuyerFactory, SellerFactory, AdminFactory) i tworzy użytkownika. Podobnie addItem() i buyItem() operują na odpowiednich modułach w tle, zapewniając prosty interfejs.

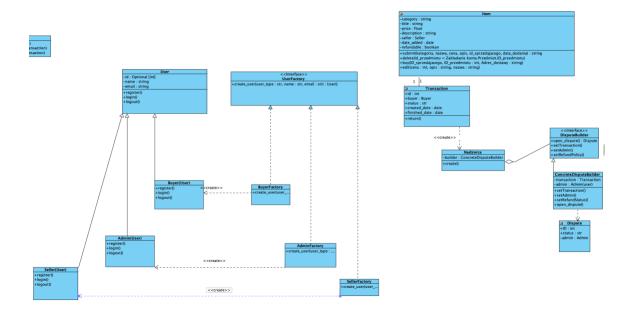
Wzorzec 2: Pełnomocnik (Proxy)

Motywacja

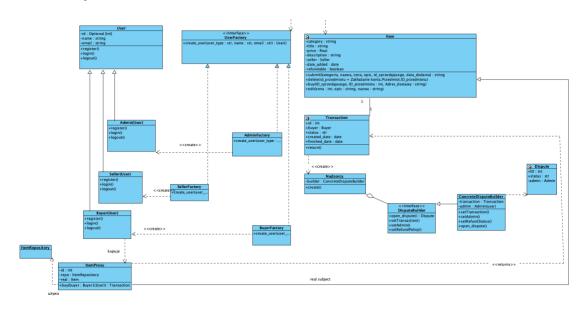
System zarządza dużą liczbą przedmiotów (Item). Ładowanie pełnych danych wszystkich ofert od razu byłoby nieefektywne. W celu optymalizacji pamięci i opóźnionego pobierania danych zastosowano wzorzec Proxy. ItemProxy reprezentuje obiekt Item i ładuje pełne dane dopiero, gdy są rzeczywiście potrzebne (np. przy zakupie).

Konsekwencje: mniejsze zużycie pamięci, krótszy czas odpowiedzi przy przeglądaniu oferty, lepsze skalowanie systemu.

Struktura przed wdrożeniem



Struktura po wdrożeniu



Współdziałanie

- 1. Użytkownik (Buyer) woła buy() na obiekcie ItemProxy.
- 2. ItemProxy sprawdza, czy ma załadowany prawdziwy Item.
- 3. Jeśli nie, pobiera go przez ItemRepository.
- 4. Następnie deleguje operację buy() do prawdziwego obiektu Item.
- 5. Item tworzy nowy Transaction.