

Aplikacje bazodanowe – projekt aplikacji

Analiza biznesowa

Aplikacja służyć ma do obsługi towarowego portu morskiego. Port składa się z terminali różnych typów (masowe, kontenerowe, naftowe, ...), choć może być także kilka terminali tego samego typu. W skład terminala wchodzi co najmniej jeden dok, w którym może przebywać w danym momencie jeden statek. W dokach odbywają się także załadunki i wyładunki ładunków. Oprócz doków, w skład terminala mogą wchodzić także magazyny, mogące przechowywać ładunki.

Każdy ładunek jest jakimś towarem w konkretnej ilości i miejscu. Ponadto, ładunek posiada nadawcę, który jest jego pierwotnym właścicielem oraz odbiorcę, który odbiera go z portu.

Istnieje możliwość przeładunku ładunków między statkami, między magazynami oraz między magazynami a statkami. Ładunek może także przybyć spoza portu oraz zostać wytransportowany poza port. W takim przypadku nie interesują nas wcześniejsze i późniejsze losy ładunku – śledzimy jego „cykl życiowy” wyłącznie na terenie naszego portu.

Za przechowywanie ładunków w magazynach oraz przebywanie statków w dokach pobierane są opłaty portowe. Opłatę za dok płaci armator, który jest właścicielem statku. Opłatę za przetrzymywanie ładunku płaci odbiorca ładunku.

Niektóre z ładunków, które przybywają do portu podlegają kontroli celnej. Takie ładunki muszą być najpierw skontrolowane przez inspektora celnego. Inspektor celny może zatwierdzić kontrolę celną, naliczając cło, albo ją odrzucić i skonfiskować towar lub wstrzymać jego rozładunek ze statku. Jeśli kontrola celna jest negatywna, ale ładunek nie został skonfiskowany, można przeprowadzić ponowną kontrolę. Jeśli natomiast ładunek został skonfiskowany traktujemy go tak jakby został rozładowany poza port.

Wymagania użytkownika

- system powinien być dostosowany do specyficznych potrzeb użytkowników: Administratora Systemu, Administratora Portu, Administratora Ładunków oraz Inspektora Celnego (role poszczególnych użytkowników zaprezentowane zostały w diagramie przypadków użycia)
- system ma uwzględniać możliwość rozbudowy portu o dodatkowe terminale, magazyny, doki.
- system ma pozwalać na wyświetlanie statków przebywających aktualnie w porcie oraz

statków przebywających w nim historycznie

- dla każdego ładunku, system ma pozwalać na podglądanie jego historii, tzn. śledzenie wszystkich jego przeładunków od przybycia do portu do jego opuszczenia
- dla każdego magazynu system powinien wyświetlać wszystkie przebywające w nim aktualnie ładunki, oraz ładunki przebywające w nim historycznie w danym okresie
- system powinien kontrolować kompatybilność statków, ładunków i terminali, np. nie pozwalając na załadunek kontenerów na tankowiec ani na dokowanie kontenerowców w doku dla masowców itd.
- system nie powinien zezwalać na przekroczenie maksymalnej ładowności statku ani pojemności magazynu
- system nie powinien pozwalać na wprowadzenie rozładunku ładunku podlegającego kontroli celnej, dla którego nie ma pozytywnej kontroli celnej. Ponadto, statek na którego pokładzie jest taki ładunek, nie może opuścić portu.
- system powinien automatycznie naliczać opłaty portowe i celne, umożliwiając także ręczne naliczenie opłat dodatkowych
- powinna być możliwość ustalenia, który użytkownik wykonał daną akcję w systemie (zadokowanie statku, zgłoszenie rozładunku, itp.)
- system ma mieć ergonomiczny interfejs użytkownika

Koncepcja architektury systemu

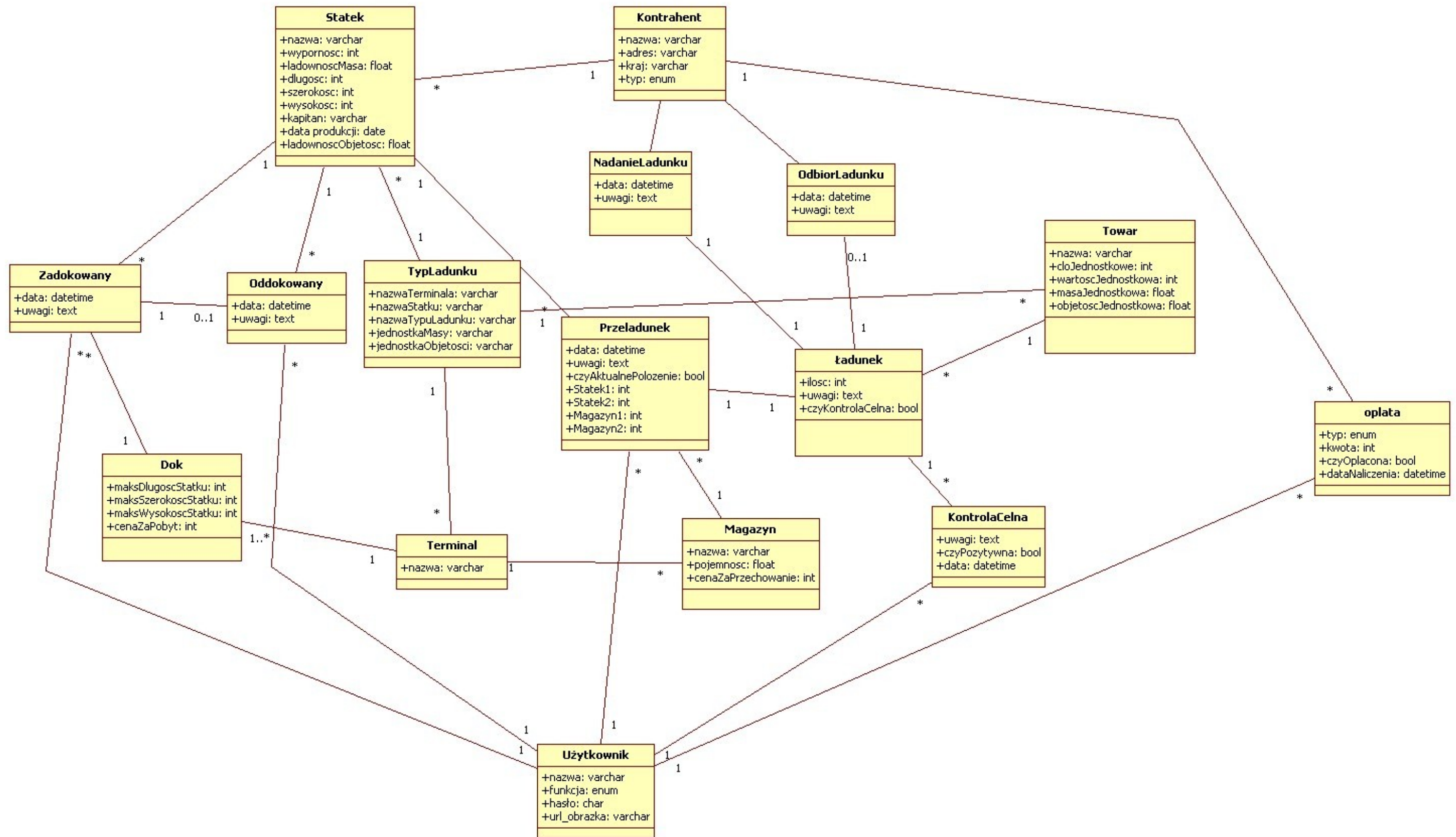
Aplikacja składać się będzie z dwóch warstw:

1. Warstwa bazy danych
2. Warstwa aplikacji

Warstwę bazy danych stanowić będzie baza MySQL. Generalnie nie będzie się w niej odbywać żadne przetwarzanie danych, z wyjątkiem prostego automatycznego naliczania opłat za pomocą wyzwalaczy.

Warstwa aplikacji będzie aplikacją webową napisaną w PHP. Składać się będzie z dwóch podwarstw: warstwy przetwarzania danych oraz interfejsu użytkownika. W warstwie przetwarzania danych odbywać się będzie komunikacja z bazą danych i przetwarzanie pobranych wyników. Natomiast interfejs użytkownika będzie jedynie wyświetlał pobrane dane w oknie przeglądarki internetowej oraz pobierał od użytkownika dane wejściowe.

UML – diagram schematu bazy danych



UML – diagram przypadków użycia

