

Inżynieria oprogramowania

Część 3: UML – Wstęp, diagramy przypadków użycia

| | | |
|------|---|----|
| 1. | UML | 3 |
| 2. | Diagramy..... | 3 |
| 2.1. | Diagramy struktur | 3 |
| 2.2. | Diagramy zachowań | 3 |
| 3. | Wykorzystanie UML..... | 4 |
| 4. | Diagram przypadków użycia | 4 |
| 4.1. | Aktorzy | 4 |
| | Zadanie 1 do zrealizowania | 5 |
| 4.2. | Przypadek użycia | 5 |
| | Zadanie 2 do zrealizowania | 5 |
| 4.3. | Związki..... | 6 |
| | Zadanie 3 do zrealizowania | 6 |
| | Zadanie 4 do zrealizowania | 9 |
| 4.4. | Modelowanie otoczenia systemu | 9 |
| | Zadanie 5 do zrealizowania | 10 |
| 4.5. | Modelowanie wymagań stawianych systemowi..... | 10 |
| | Zadanie 6 do zrealizowania | 11 |

1. UML

Otwarty format **UML** (ang. **Unified Modeling Language**, czyli Zunifikowany Język Modelowania) – język formalny wykorzystywany do modelowania różnego rodzaju systemów, stworzony przez Grady Boocha, Jima Rumbaugh'a oraz Ivara Jacobsona, obecnie rozwijany przez Object Management Group.



Służy do modelowania dziedziny problemu (opisywania-modelowania fragmentu istniejącej rzeczywistości – na przykład modelowanie tego, czym zajmuje się jakiś dział w firmie) – w przypadku stosowania go do analizy oraz do modelowania rzeczywistości, która ma dopiero powstać – tworzy się w nim głównie modele systemów informatycznych. UML jest głównie używany wraz z jego reprezentacją graficzną – jego elementom przypisane są symbole, które wiązane są ze sobą na diagramach.

Dla najnowszej wersji 2.2 języka UML wyróżnia się 14 diagramów głównych oraz 3 abstrakcyjne (struktur, zachowań i interakcji). Istnieją niestety pewne niejednoznaczności co do stosowanego polskiego tłumaczenia diagramów, np. ang. timing diagram jest tłumaczony jako diagram czasowy, zależności czasowych, harmonogramowania, uwarunkowań czasowych czy diagram przebiegów czasowych.

2. DIAGRAMY

2.1. DIAGRAMY STRUKTUR

- Klas (najczęściej spotykane, ang. class diagram)
- Obiektów (ang. object diagram)
- Komponentów (ang. component diagram)
- Wdrożenia (ang. deployment diagram)
- Struktur złożonych (ang. composite structure diagram)
- Pakietów (ang. package diagram)
- Profili (ang. profile diagram, nowość wprowadzona w UML 2.2)

2.2. DIAGRAMY ZACHOWAŃ

- Czynności (ang. activity diagram)
- Przypadków użycia (ang. use case diagram)
- Maszyny stanów (ang. state machine diagram) (dla UML 1.x Stanów, ang. statechart diagram)
- Interakcji (diagram abstrakcyjny)
- Komunikacji (ang. communication diagram) (dla UML 1.x Współdziałania, ang. collaboration diagram)
- Sekwencji (ang. sequence diagram)
- Czasowe (ang. timing diagram)
- Przeglądu interakcji (ang. interaction overview diagram)

W przypadku modelowania biznesowego można korzystać z pewnych modyfikacji wyżej wymienionych diagramów UML, np. diagramu biznesowych przypadków użycia (charakterystyczna cięciwa dla symbolów aktora i przypadku użycia).

3. WYKORZYSTANIE UML

W praktyce rzadko kiedy trzeba opracowywać wszystkie diagramy i w większości przypadków korzysta się z mniej niż połowy wyżej wymienionych. Nie powinno modelować się tylko dla samego modelowania, dlatego nie zawsze wszystkie rodzaje są potrzebne.

Projektując system informatyczny, rozpoczyna się przeważnie od tworzenia diagramów w następującej kolejności:

- ❖ Przypadków użycia
- ❖ Sekwencji
- ❖ Klas
- ❖ Aktywności

Są to najczęściej wykorzystywane diagramy. Pozostałe bywają pomijane, zwłaszcza przy budowaniu niedużych systemów informatycznych.

4. DIAGRAM PRZYPADKÓW UŻYCIA

Diagramy przypadków użycia służą do modelowania perspektywy przypadków użycia systemu, a w tym do opisywania otoczenia systemu, podsystemu lub klasy lub określania wymagań dotyczących zachowań tych bytów.

Diagramy przypadków użycia są szczególnie przydatne w obrazowaniu, specyfikowaniu i dokumentowaniu zachowania bytu. Dzięki nim systemy, podsystemy i klasy stają się bardziej przystępne i zrozumiałe.

Dobrze zbudowany diagram przypadków użycia

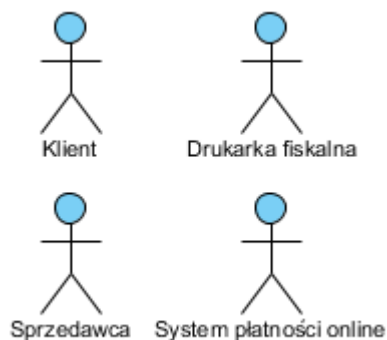
- uwypukla jeden statyczny aspekt perspektywy przypadków użycia systemu;
- uwzględnia tylko te przypadki użycia i tych aktorów, którzy są niezbędni do zrozumienia tego aspektu.
- uwzględnia szczegóły odpowiednie do przyjętego poziomu abstrakcji, z dodatkami (np. miejsca rozszerzania), które są niezbędne do zrozumienia tego, na czym Ci zależy;
- nie jest zbyt ogólny, a zatem czytelnik nie zostanie wprowadzony w błąd co do istotnego znaczenia.

Gdy rysujesz diagram przypadków użycia,

- nadaj mu nazwę, która określa jego przeznaczenie;
- tak ułóż elementy, żeby zminimalizować liczbę przecinających się linii
- poukładaj zbliżone znaczeniowo działania i role tak, żeby były także blisko siebie na płaszczyźnie;
- skorzystaj z notatek i kolorów, żeby zwrócić uwagę czytelnika na to, na czym Ci zależy;
- postaraj się nie umieszczać na nim zbyt wielu rodzajów związków; jeśli sieć związków zawierania i rozszerzania jest bardzo złożona, przenieś związane z nimi byty na odrębne diagramy.

4.1. AKTORZY

Aktorzy reprezentują spójny zbiór ról, jakie odgrywają użytkownicy przypadku użycia w czasie interakcji z danym przypadkiem użycia. Aktorzy mogą reprezentować stanowiska i funkcje w danej organizacji, mogą to być także systemy zewnętrzne aplikacji (podsystem, baza danych itd.) czy też urządzenia.



Rysunek 1: Aktorzy

Aktorzy są najczęściej prezentowani jako proste postacie, tak jak to pokazano na rysunku 1.

ZADANIE 1 DO ZREALIZOWANIA

Z wykorzystaniem programu do modelowania UML utworzyć projekt i dodać aktorów przedstawionych na rysunku 1. **(Wszystkie zadania z instrukcji realizować w ramach jednego projektu)**

- Sprzedawca
- Klient
- Drukarka fiskalna
- System płatności online

4.2. PRZYPADEK UŻYCIA

Opisani wyżej aktorzy, a więc użytkownicy projektowanego przez nas systemu, oczekują od niego, aby oferował on określoną funkcjonalność. Każdy z aktorów potrzebuje innej funkcjonalności systemu (jednak może się ona miejscami nakładać, a więc pewne funkcje mogą być potrzebne jednocześnie kilku aktorom). Te funkcjonalności to inaczej mówiąc zadania, jakie system musi spełniać.



Rysunek 2: Przypadki użycia

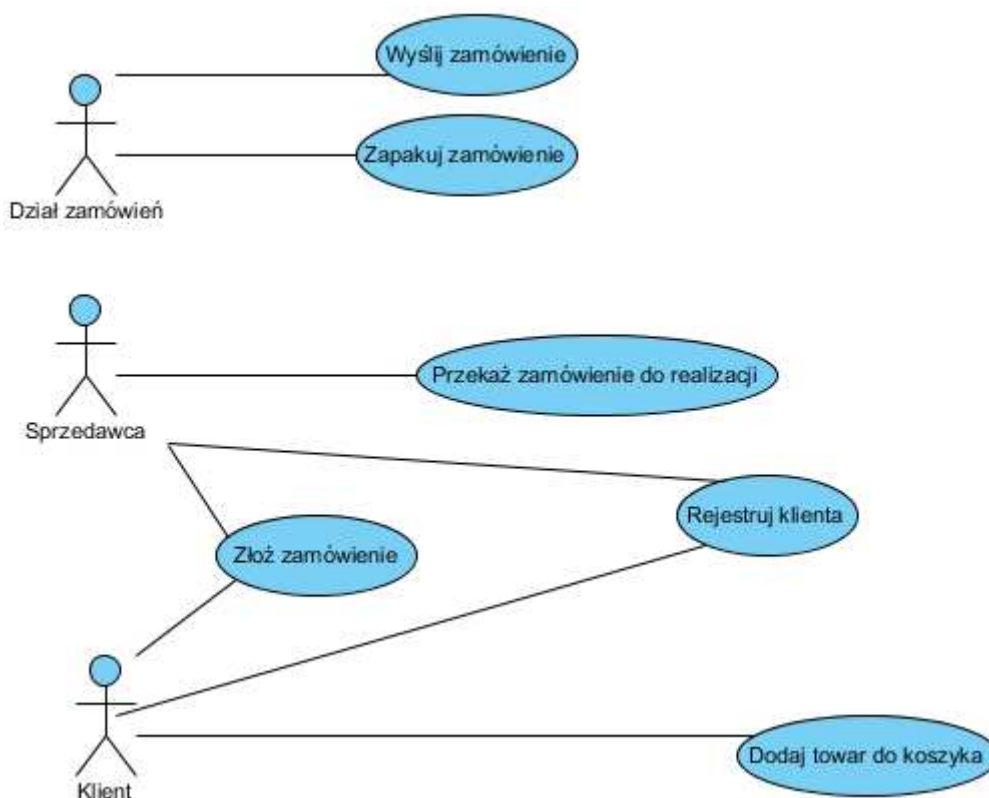
Zadania to jednocześnie nasze przypadki użycia. Oficjalnie przypadek użycia jest specyfikacją akcji i ich wariantów, które poprzez interakcje z aktorami systemu, system może wykonać. Najprościej rzecz ujmując, jest on działaniem, jakie realizuje system w odpowiedzi na aktywność aktora. Przypadki użycia na diagramach UML prezentuje się w postaci elips z umieszczonymi w środku nazwami (zobacz rysunek 2).

ZADANIE 2 DO ZREALIZOWANIA

Dodać do diagramu przypadki użycia przedstawione na rysunku 2.

4.3. ZWIĄZKI

Głównym związkiem jest asocjacja. Asocjacja jest podstawowym rodzajem związków między klasami. Oznacza ona istnienie trwałego powiązania pomiędzy nimi. Mówi ona o wystąpieniu dwukierunkowej komunikacji pomiędzy przypadkiem użycia a aktorem. Jeśli komunikacja ta przebiega tylko w jednym kierunku, można kierunek ten zaznaczyć strzałką. W przypadku diagramów użycia, związkom nie nadaje się nazw.



Rysunek 3: Związki - asocjacja

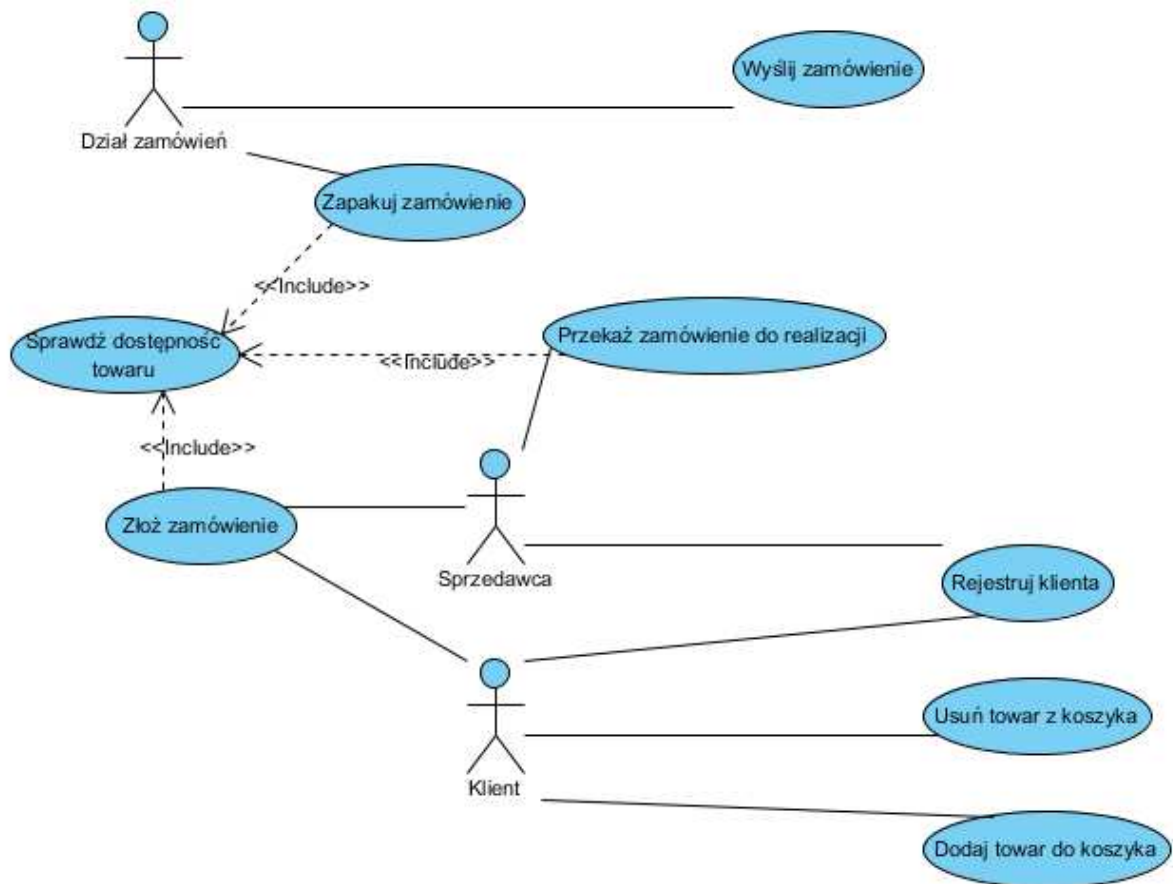
Na rysunku 3 zaprezentowany został przykładowy diagram użycia. Aktorzy klient oraz sprzedawca dzielą między sobą część przypadków użycia. Dzięki temu obsługa sklepu on-line może za klienta złożyć zamówienie w przypadku, gdy klient skontaktuje się z obsługą telefonicznie. Innym związkiem, już o wiele rzadziej spotykanym w przypadkach użycia, jest zawieranie. Zawierany przypadek użycia nie jest wykonywany samodzielnie.

ZADANIE 3 DO ZREALIZOWANIA

Na nowym diagramie użycia utworzyć diagram przedstawiony na rysunku 3.

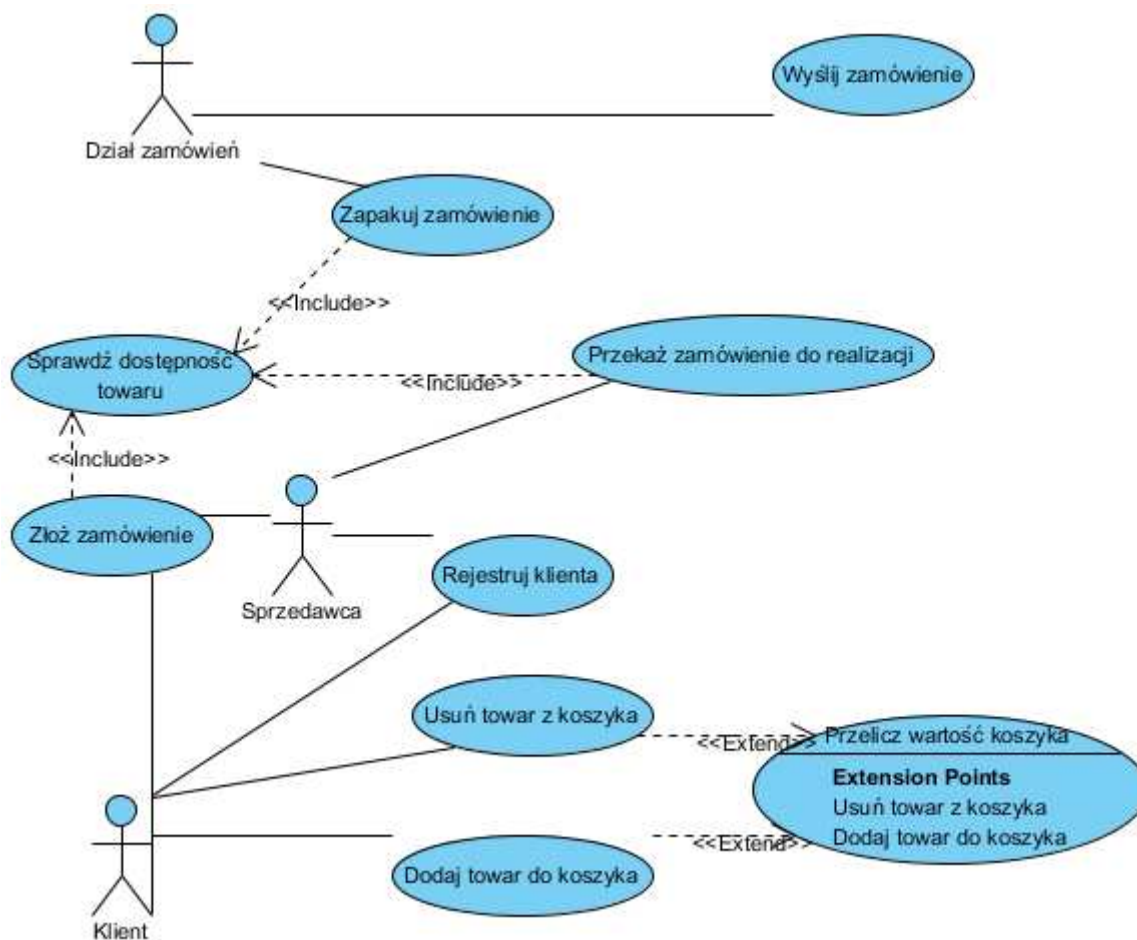
Związek zawierania ma postać przerywanej strzałki ze stereotypem <<include>>, biegnącej od przypadku użycia zawierającego do zawieranego. Związku zawierania używa się wówczas, gdy z kilku innych przypadków użycia można wydzielić pewną część wspólną. Na rysunku 4 - zamiast przypadków użycia "Sprawdź dostępność towarów, a następnie zapakuj zamówienie", "Sprawdź dostępność towarów a następnie złóż zamówienie" i "Sprawdź dostępność towarów, a następnie przełącz zamówienie do realizacji" mamy "Zapakuj zamówienie",

"Złóż zamówienie" i "Przełącz zamówienie do realizacji" z wydzielonym osobnym przypadkiem użycia "Sprawdź dostępność towaru".



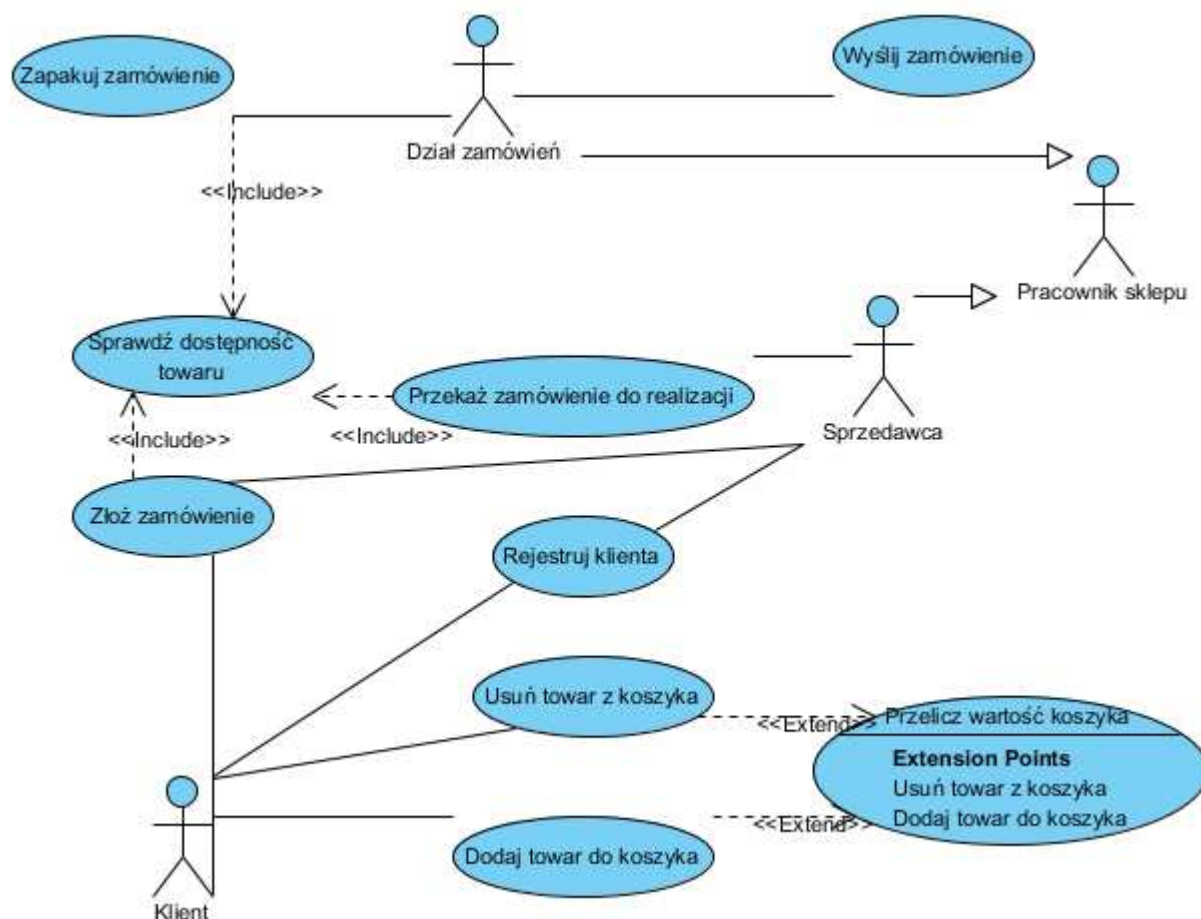
Rysunek 4: Związki - zawieranie

Związek rozszerzanie pozwala na wydzielenie przypadku użycia, który w pewnych sytuacjach może zostać wzbogacony o dodatkowe opcje. Wygląda on tak samo jak związek zawierania, jednak ma stereotyp <<extend>>. Na rysunku 5 widać, że przypadki użycia "Dodaj towar do koszyka" oraz "Usuń towar z koszyka" rozszerzają przypadek użycia "Przelicz wartość koszyka".



Rysunek 5: Związki - rozszerzanie

Ostatnim związkiem będzie uogólnienie. Ma on na celu uogólnienie aktorów bądź przypadków użycia, przy czym obiekt uogólniany posiada wszystkie cechy obiektu ogólnego. Uogólnienie ma postać strzałki z linią ciągłą i zamkniętym grotem. Uogólnienie zostało zaprezentowane na rysunku 6 - aktorzy Dział zamówień oraz Sprzedawca on-line są specjalizacjami aktora Pracownik sklepu.



Rysunek 6: Związki – uogólnienie

ZADANIE 4 DO ZREALIZOWANIA

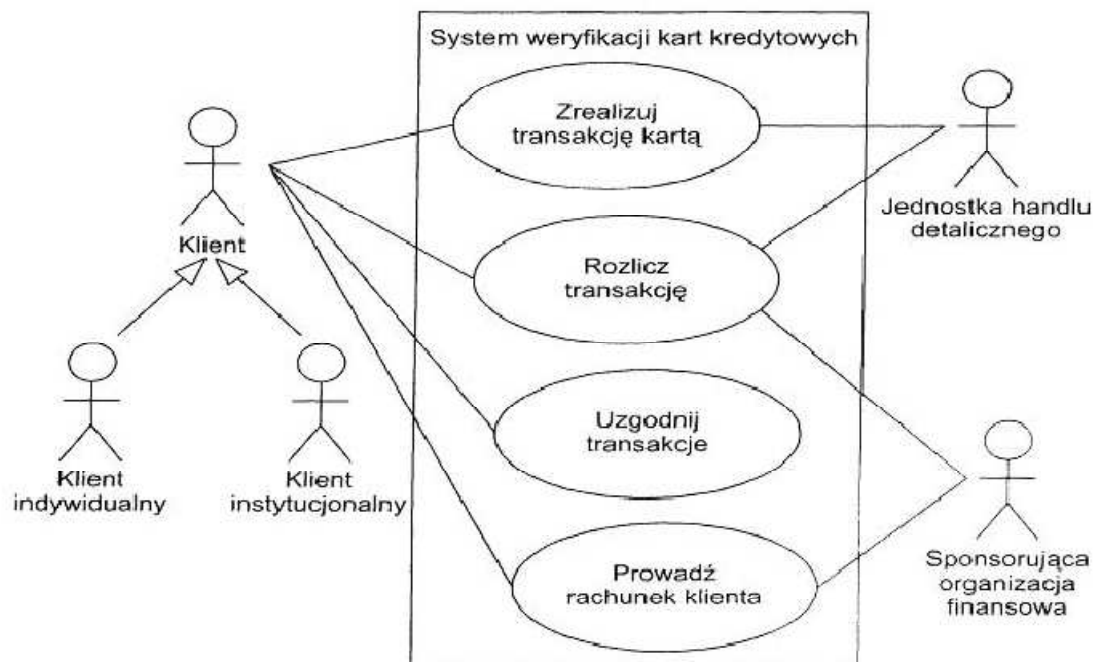
Na nowym diagramie użycia utworzyć diagram przedstawiony na rysunku 6.

4.4. MODELOWANIE OTOCZENIA SYSTEMU

To zadanie polega między innymi na wyznaczeniu granicy wokół całego systemu i na wskazaniu leżących poza nią aktorów, którzy wchodzi w interakcję z systemem. Diagramy przypadków użycia służą w tym wypadku do zdefiniowania aktorów i znaczenia ich ról.

Wytyczne:

- Zidentyfikuj aktorów działających wokół systemu. W tym celu rozważ, które grupy potrzebują pomocy systemu do realizacji swoich zadań, są niezbędne do realizacji funkcji systemu, są w interakcji z urządzeniami zewnętrznymi lub innymi systemami informatycznymi, wypełniają drugorzędne funkcje, takie jak zarządzanie i pielęgnacja.
- Uporządkuj podobnych aktorów za pomocą uogólnień.
- Aby zwiększyć czytelność modelu, dodaj - jeśli trzeba - stereotypy do aktorów.



Rysunek 7: Modelowanie otoczenia systemu

ZADANIE 5 DO ZREALIZOWANIA

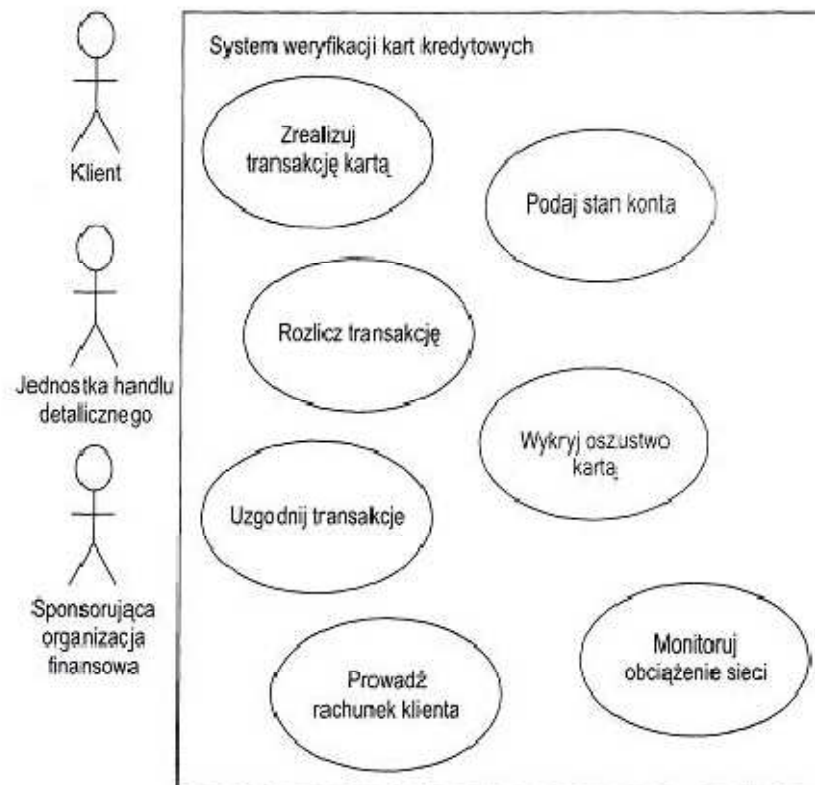
Na nowym diagramie użycia utworzyć diagram przedstawiony na rysunku 7.

4.5. MODELOWANIE WYMAGAŃ STAWIANYCH SYSTEMOWI

To zadanie polega na określeniu, co system powinien robić (z punktu widzenia jego otoczenia) - niezależnie od tego, jak ma to zrobić. Diagramy przypadków użycia służą w tym wypadku do zdefiniowania oczekiwanego działania systemu. Możesz do systemu podejść jak do czarnej skrzynki - znane jest otoczenie i sposób porozumienia się z bytami leżącymi poza nim, ale nie to, co dzieje się w jego wnętrzu.

Wytyczne:

- Określ otoczenie systemu. W tym celu zidentyfikuj okalających go aktorów.
- W wypadku każdego aktora rozważ działania, których on oczekuje lub wymaga od systemu.
- Zapisz te działania w postaci przypadków użycia.
- Wyłącz powtarzające się fragmenty działań i utwórz z nich nowe przypadki użycia, które będą dołączane przez inne przypadki użycia.
- Wydziel warianty działań i umieść je w nowych przypadkach użycia, które rozszerzają główne ciągi zdarzeń innych przypadków użycia.
- Uwzględnij te przypadki użycia, aktorów i związki między nimi na diagramie przypadków użycia.



Rysunek 8: Modelowanie wymagań stawianych systemowi

ZADANIE 6 DO ZREALIZOWANIA

Na nowym diagramie użycia utworzyć diagram przedstawiony na rysunku 8.