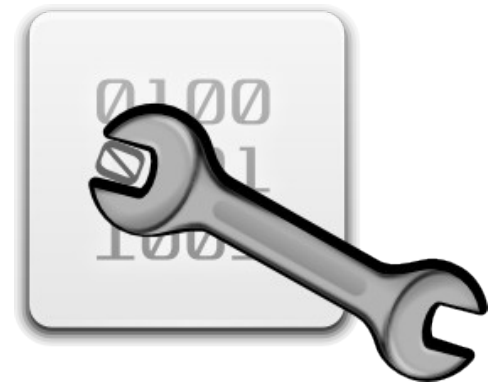


Podstawy inżynierii oprogramowania



Modelowanie.
Podstawy notacji UML

Aleksander Lamża
ZKSB · Instytut Informatyki
Uniwersytet Śląski w Katowicach

aleksander.lamza@us.edu.pl

- Czym jest UML?
- Wybrane diagramy
- Jak przedstawić budowę systemu, zależności i działanie?

Zastanówmy się jak, na podstawie zebranych wymagań, utworzyć **model** tworzonego systemu (aplikacji).

Na tym etapie może pomóc **UML**.

UML – *Unified Modeling Language*, czyli zunifikowany język modelowania.

Język ten powstał w latach 90. XX wieku. Celem jego opracowania było ujednolicenie metod modelowania obiektowego.

Jest to graficzny język modelowania, na który składa się kilkanaście typów **diagramów**.

Diagram nie jest projektem.

Diagram to **reprezentacja modelu projektu** (lub części projektu), która przedstawia wybrany aspekt tego projektu.

Diagramy dynamiczne (behawioralne)

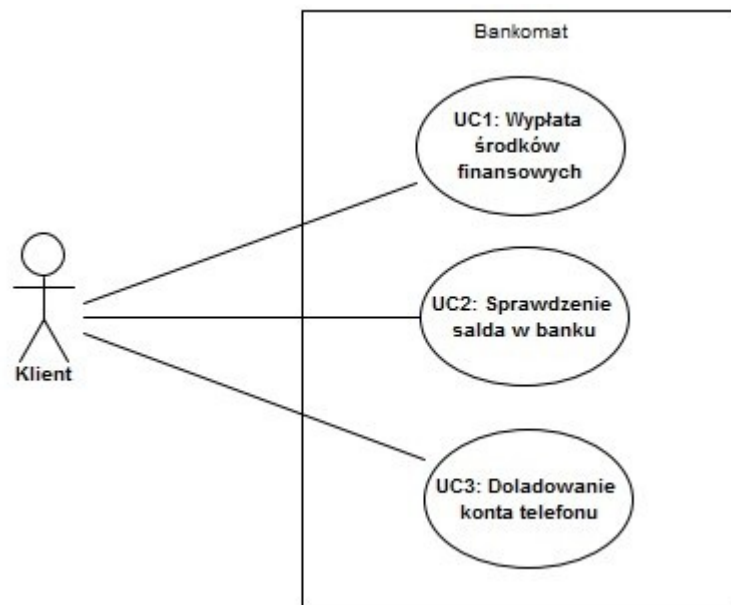
- przypadków użycia
- aktywności (czynności)
- maszyny stanów
- sekwencji
- ...

Diagramy statyczne (struktur)

- klas
- obiektów
- komponentów
- wdrożenia, pakietów, struktur...

Diagram przypadków użycia

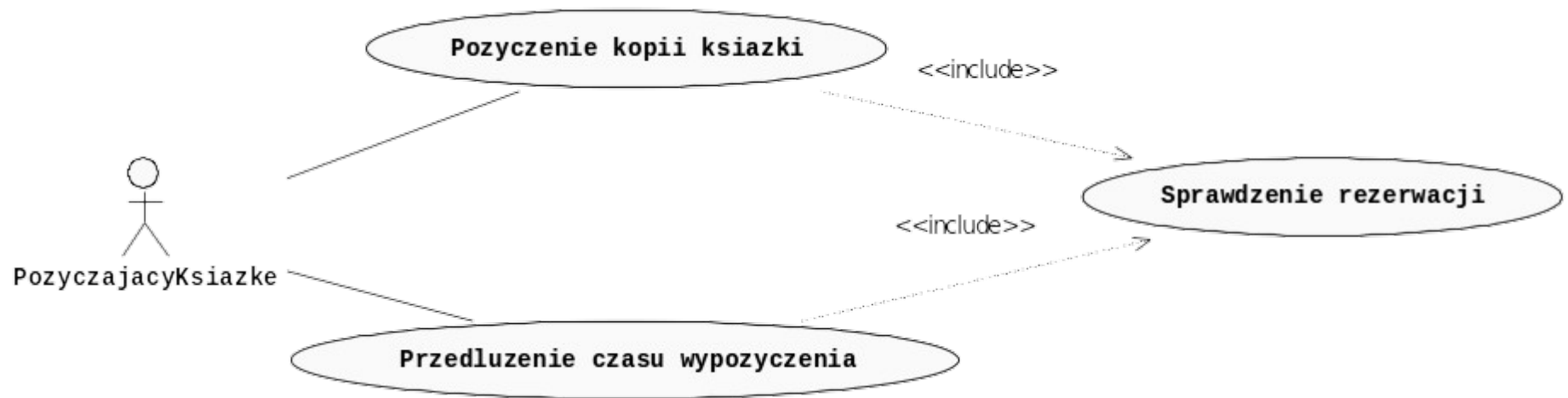
Przedstawia związki łączące **przypadki użycia** (scenariusze działania) z **aktorami** (np. użytkownikami odgrywającymi daną rolę).



UML – przegląd wybranych diagramów

Diagram przypadków użycia cd.

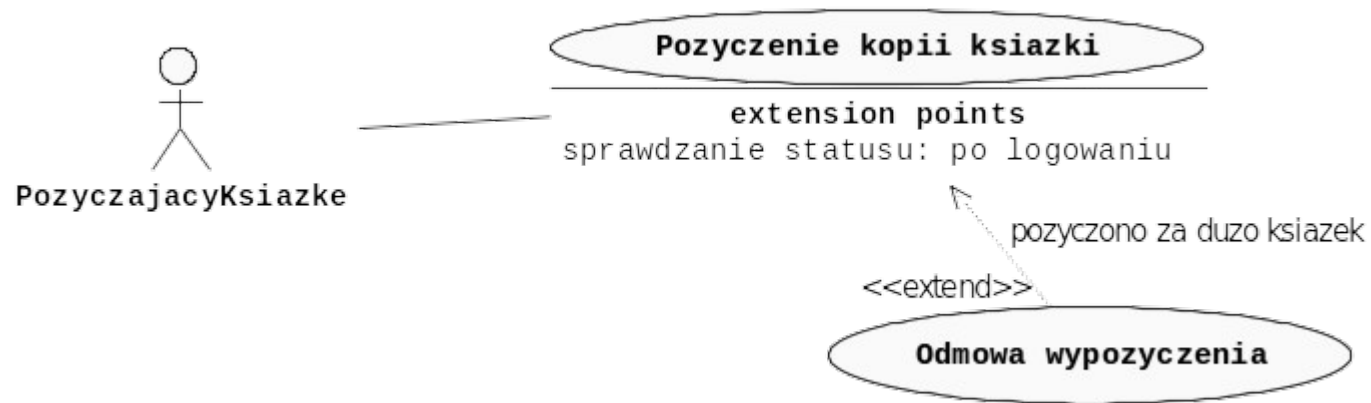
Zawieranie innych przypadków użycia («include»):



UML – przegląd wybranych diagramów

Diagram przypadków użycia cd.

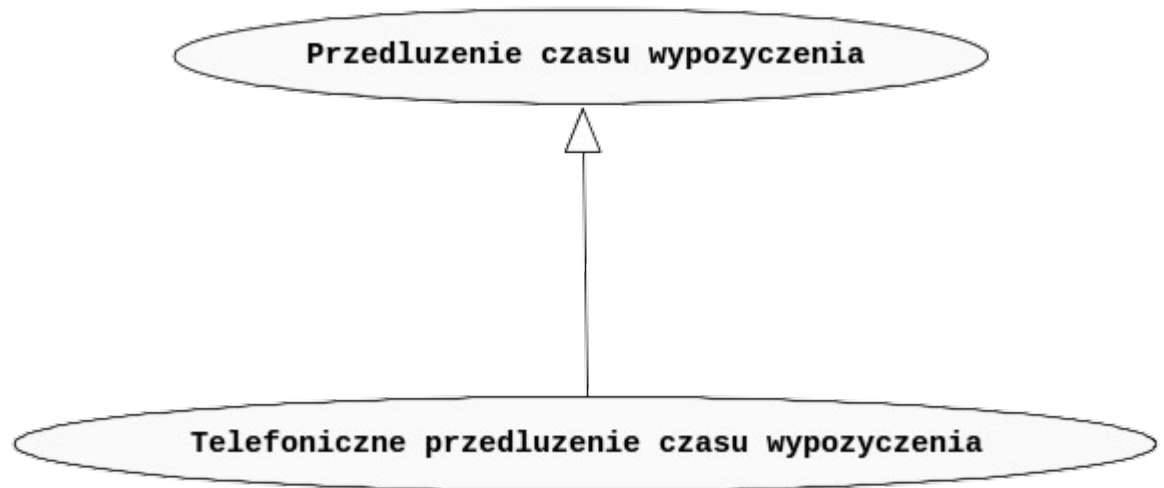
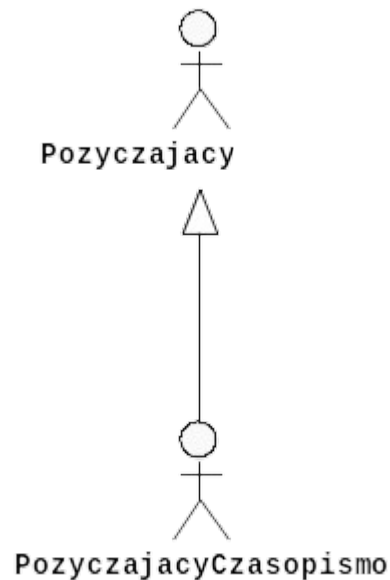
Warianty przypadków użycia («**extend**»):



UML – przegląd wybranych diagramów

Diagram przypadków użycia cd.

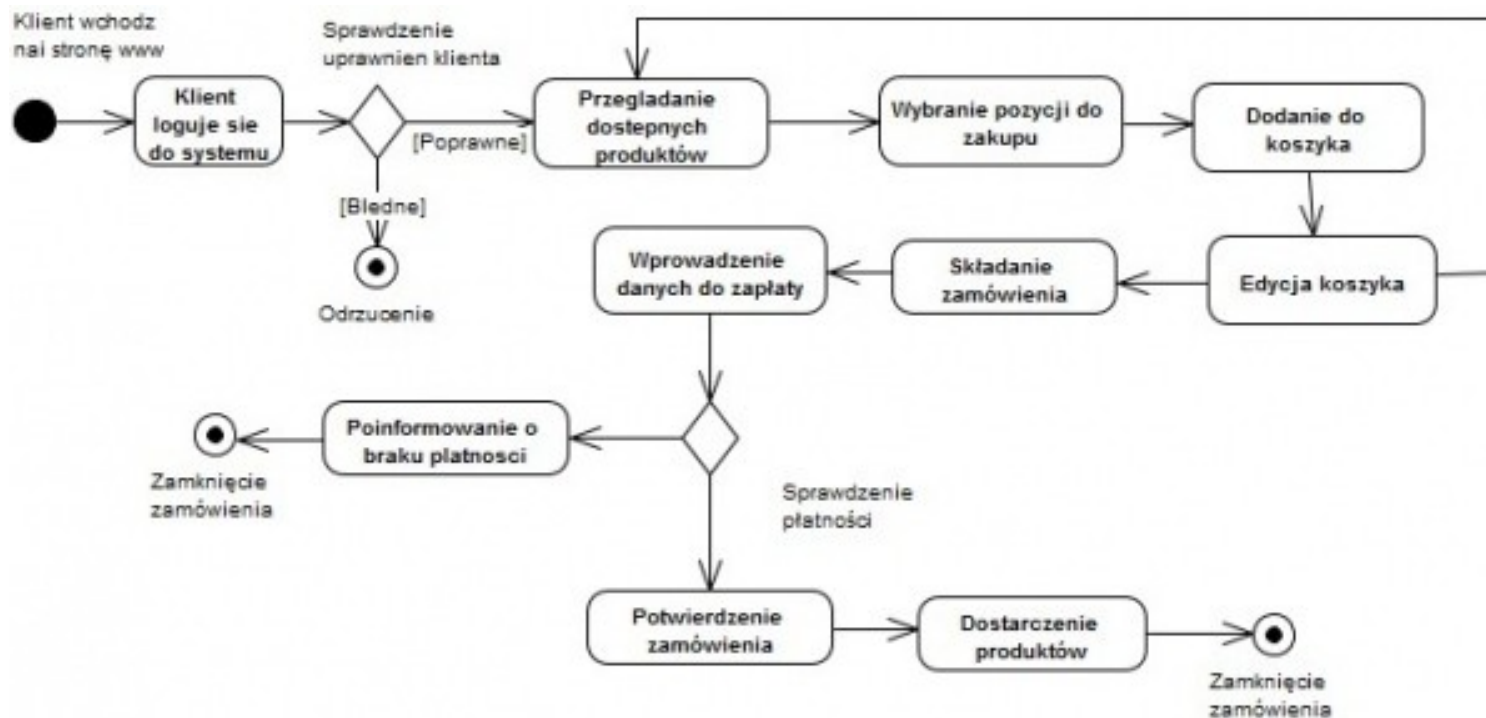
Generalizacja przypadków użycia i aktorów:



UML – przegląd wybranych diagramów

Diagram aktywności (czynności)

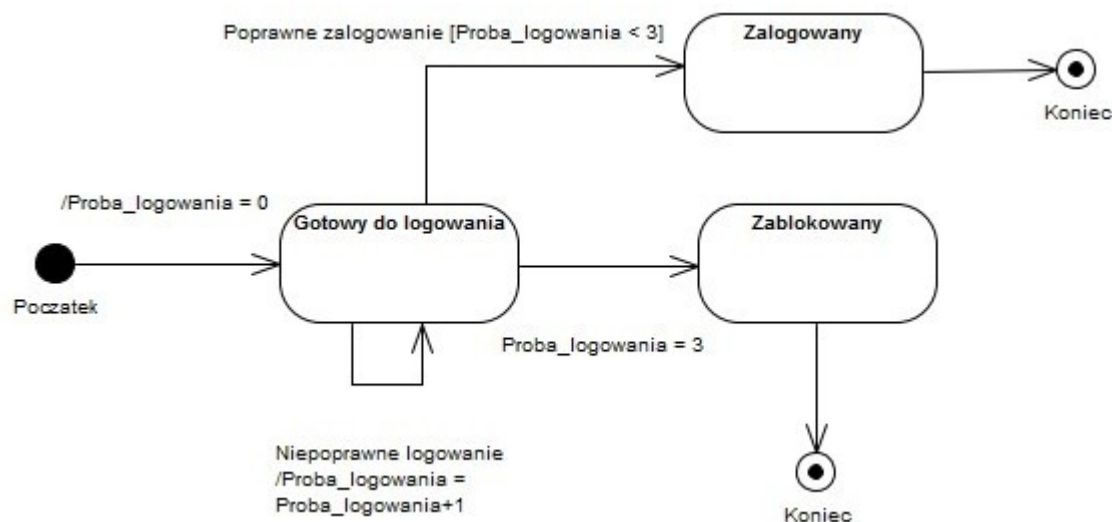
Służy do zobrazowania ciągu czynności wykonywanych w ramach systemu lub jego części.



UML – przegląd wybranych diagramów

Diagram stanów (maszyny stanów)

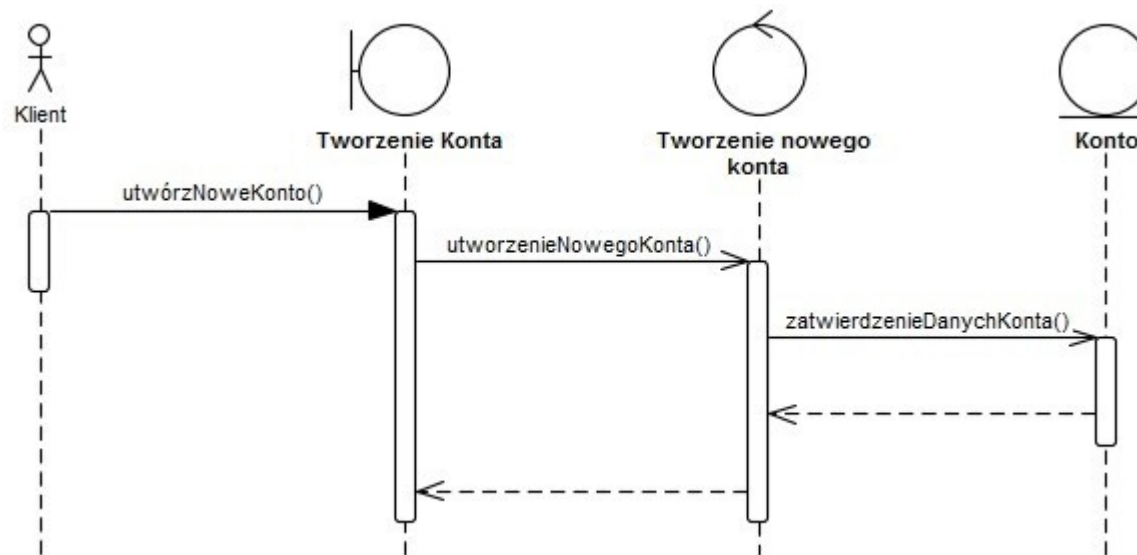
Przedstawia następstwo stanów, które może przyjąć „obiekt” oraz czynniki wpływające na zmianę stanu.



UML – przegląd wybranych diagramów

Diagram sekwencji

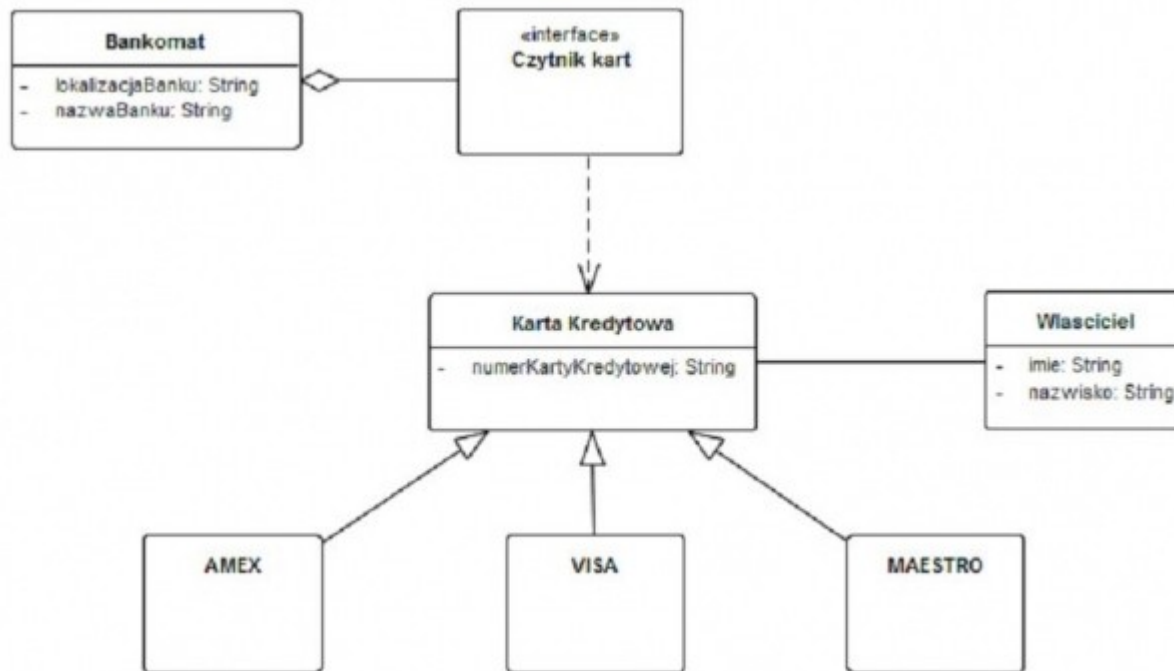
Służy do zobrazowania interakcji między obiektami oraz związane z nimi zależności czasowe.



UML – przegląd wybranych diagramów

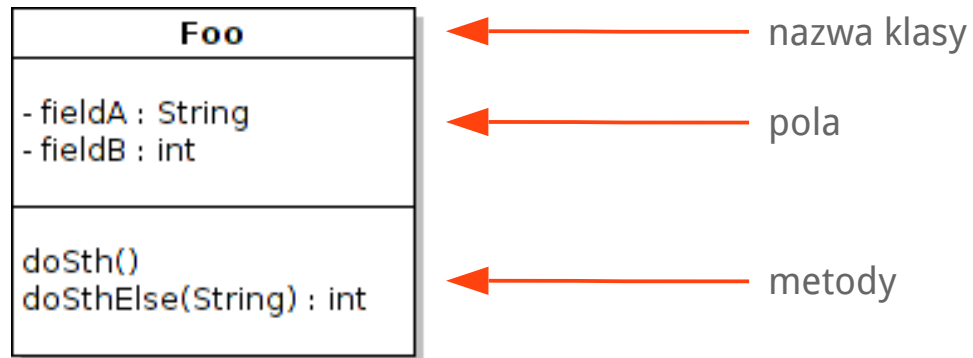
Diagram klas

Przedstawia strukturę systemu na poziomie klas i powiązań między nimi.



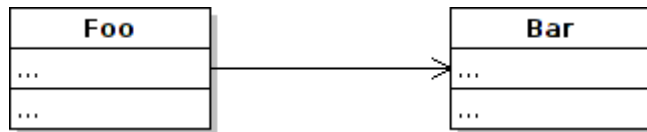
UML – przegląd wybranych diagramów

Budowa pojedynczego bloku diagramu klas

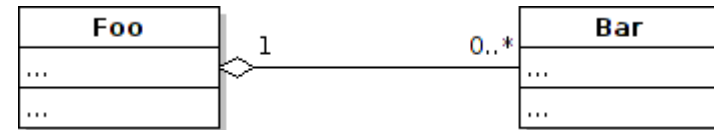


UML – przegląd wybranych diagramów

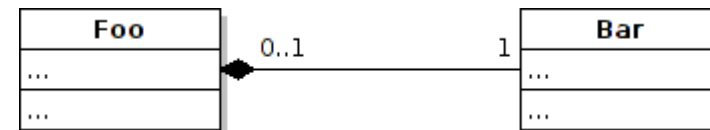
asocjacja



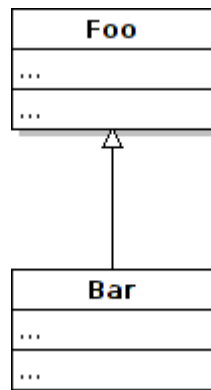
agregacja



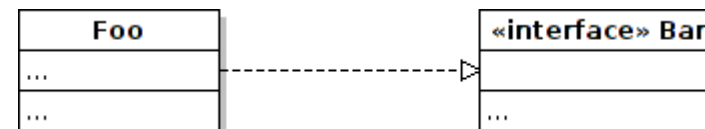
kompozycja



dziedziczenie



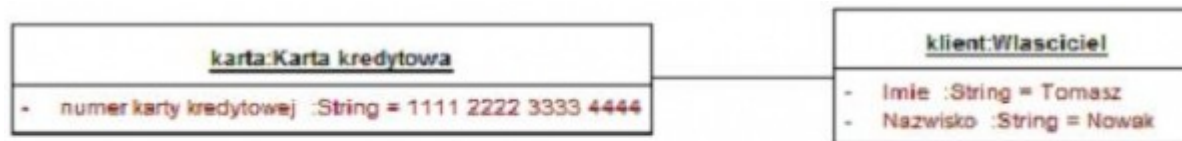
implementowanie interfejsu



UML – przegląd wybranych diagramów

Diagram obiektów

Może być uzupełnieniem diagramu klas.
Przedstawia obiekty w danym kontekście (czasowym).



UML – przegląd wybranych diagramów

Diagram komponentów

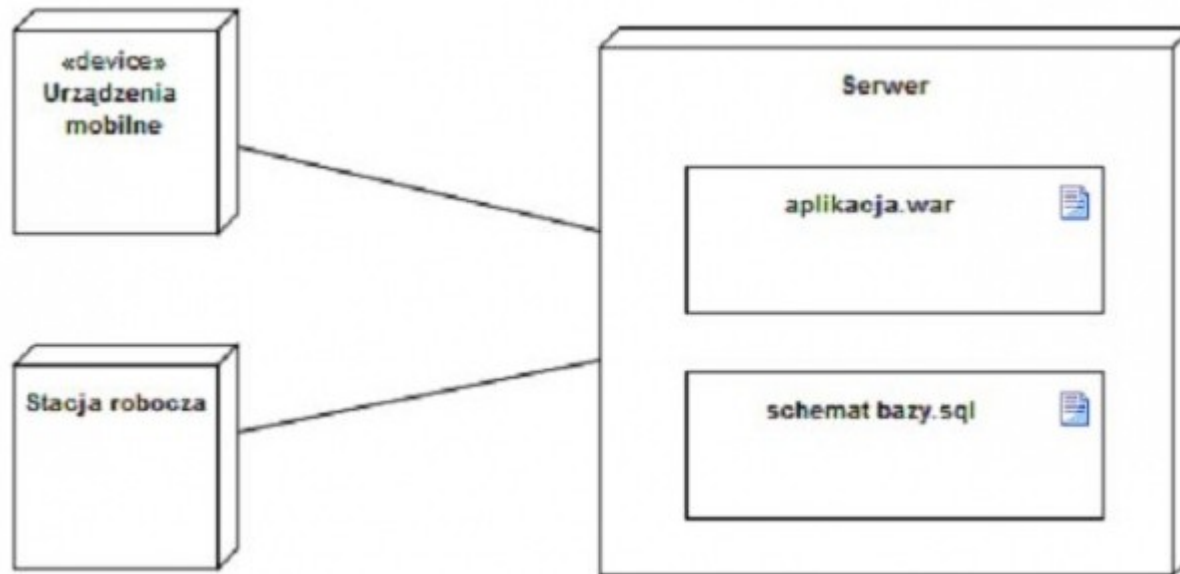
Przedstawia budowę systemu w szerszej perspektywie.
Pozwala na modelowanie architektury logicznej.



UML – przegląd wybranych diagramów

Diagram wdrożenia

Przedstawia fizyczną realizację architektury.



Wybrane diagramy omówimy na bardziej konkretnym przykładzie.

Trafił nam się fajny projekt:



Przykładowy projekt

Plansza składa się z 40 pól.
Każde pole ma przypisaną określoną funkcję

Po 16 kart „Szansa” i „Kasa społeczna”

W danej chwili
gracz znajduje się
na jednym polu

Dwie kości



Każdy gracz ma pieniądze...

...karty własności...

...oraz hotele i domy

Diagram przypadków użycia

W omawianym przypadku podstawowym aktorem jest gracz.

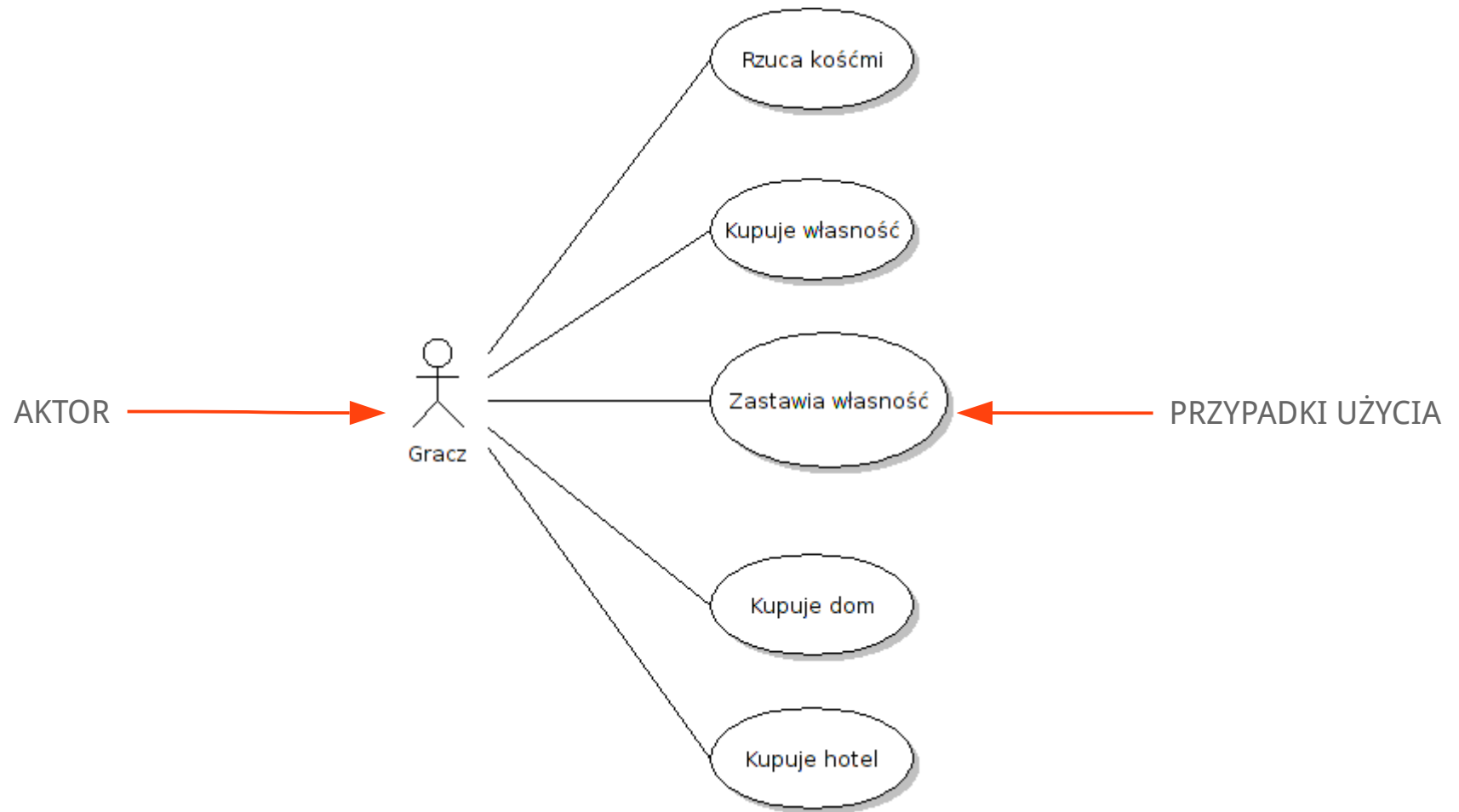
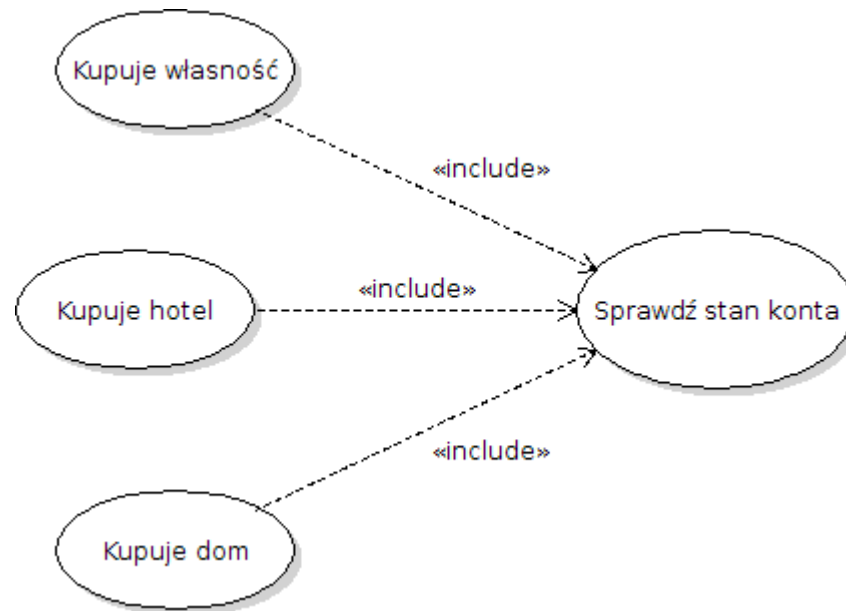


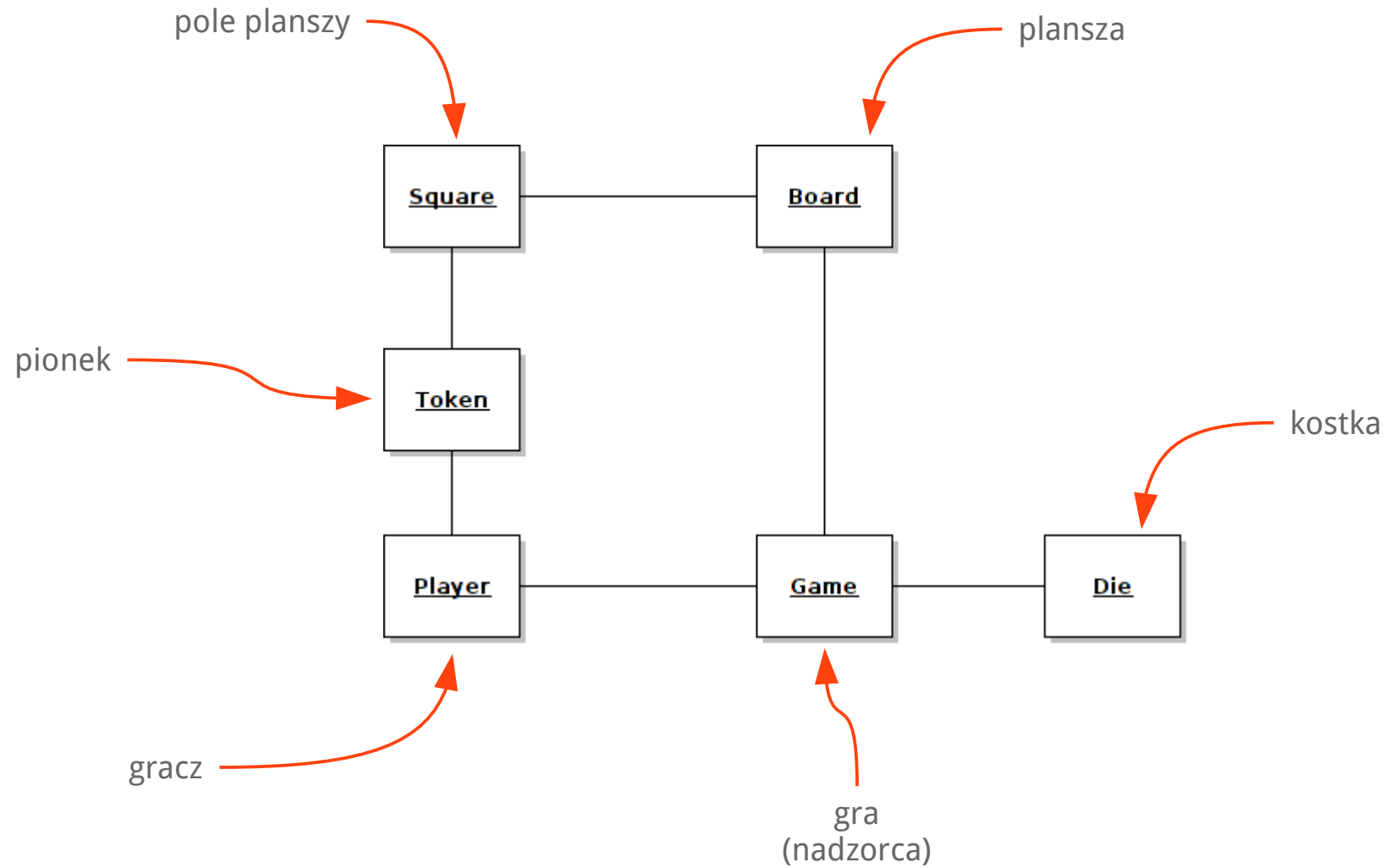
Diagram przypadków użycia

Oczywiście można pójść dalej i przedstawić na diagramie szczegóły, np.:

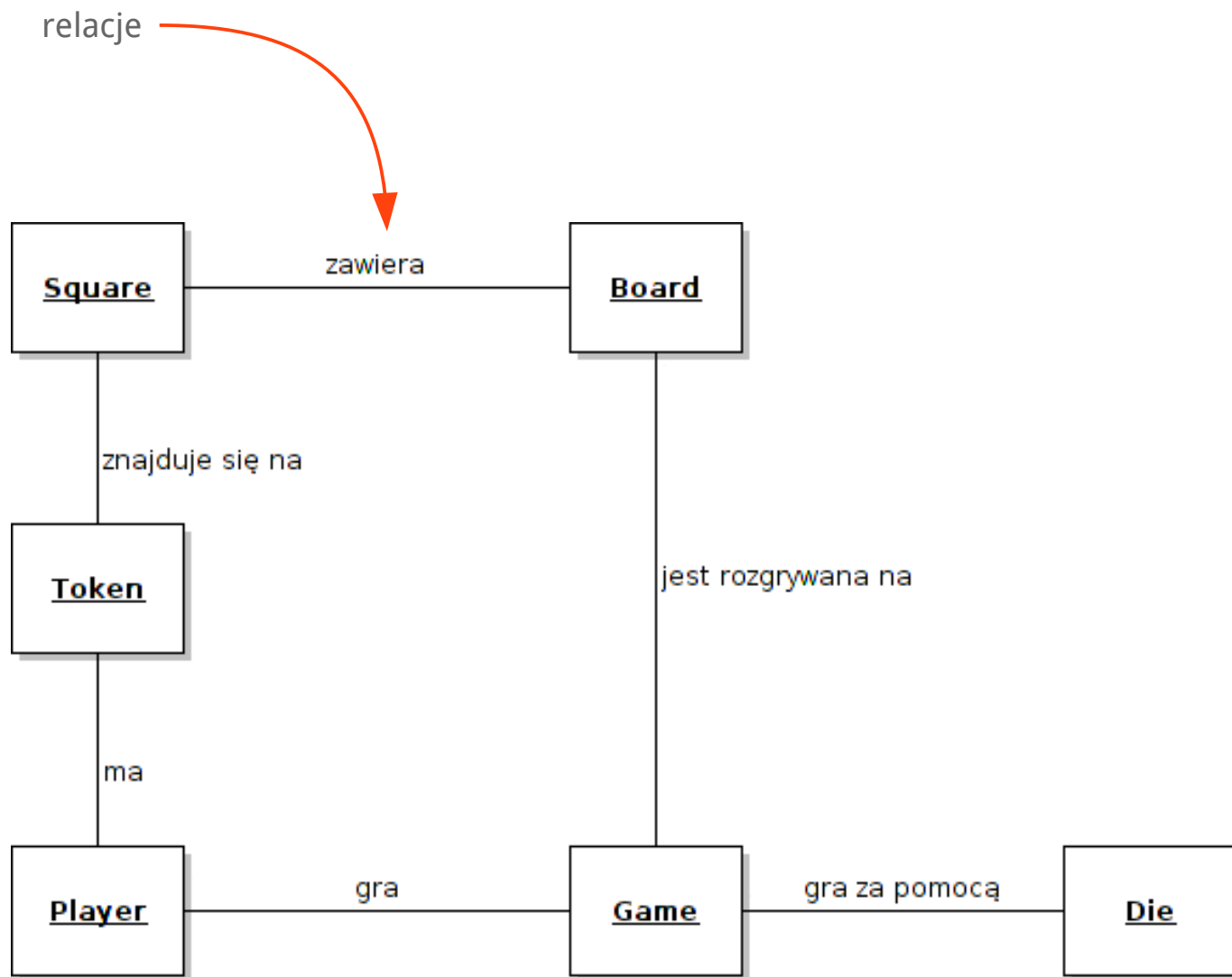


Wszystkie trzy przypadki użycia związane z kupowaniem muszą się wiązać ze sprawdzeniem stanu konta.

Ogólna budowa



Ogólna budowa



Ogólna budowa

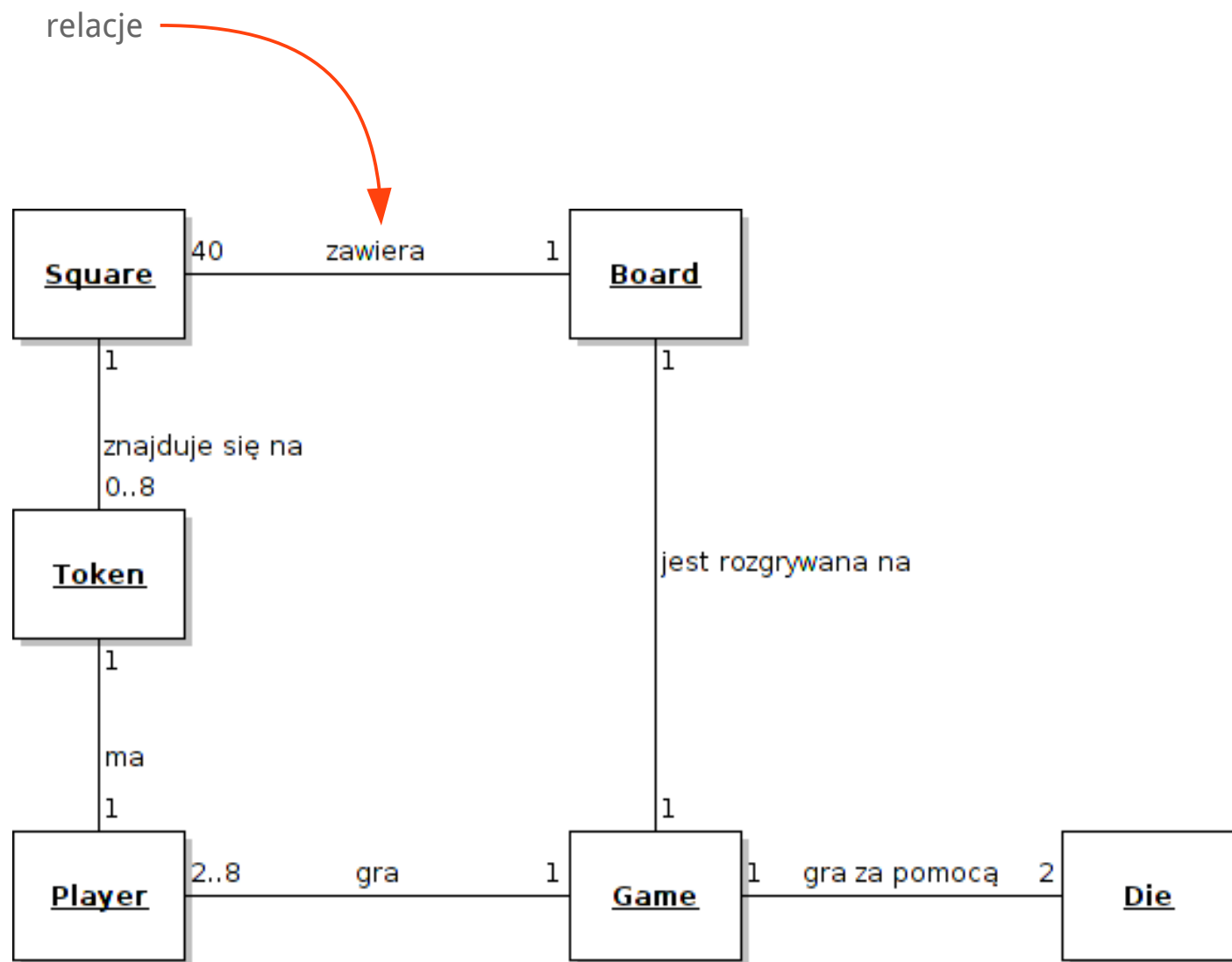


Diagram klas – przykład

Jak będzie wyglądał diagram klas związanych z planszą?

Klasa **Board** zawiera czterdzieści obiektów klasy **Square**



Przyjrzyjmy się bliżej klasie **Square**.
Jak wiadomo, na planszy znajdują się różne pola.

Czy jedna klasa wystarczy?

Diagram klas – przykład

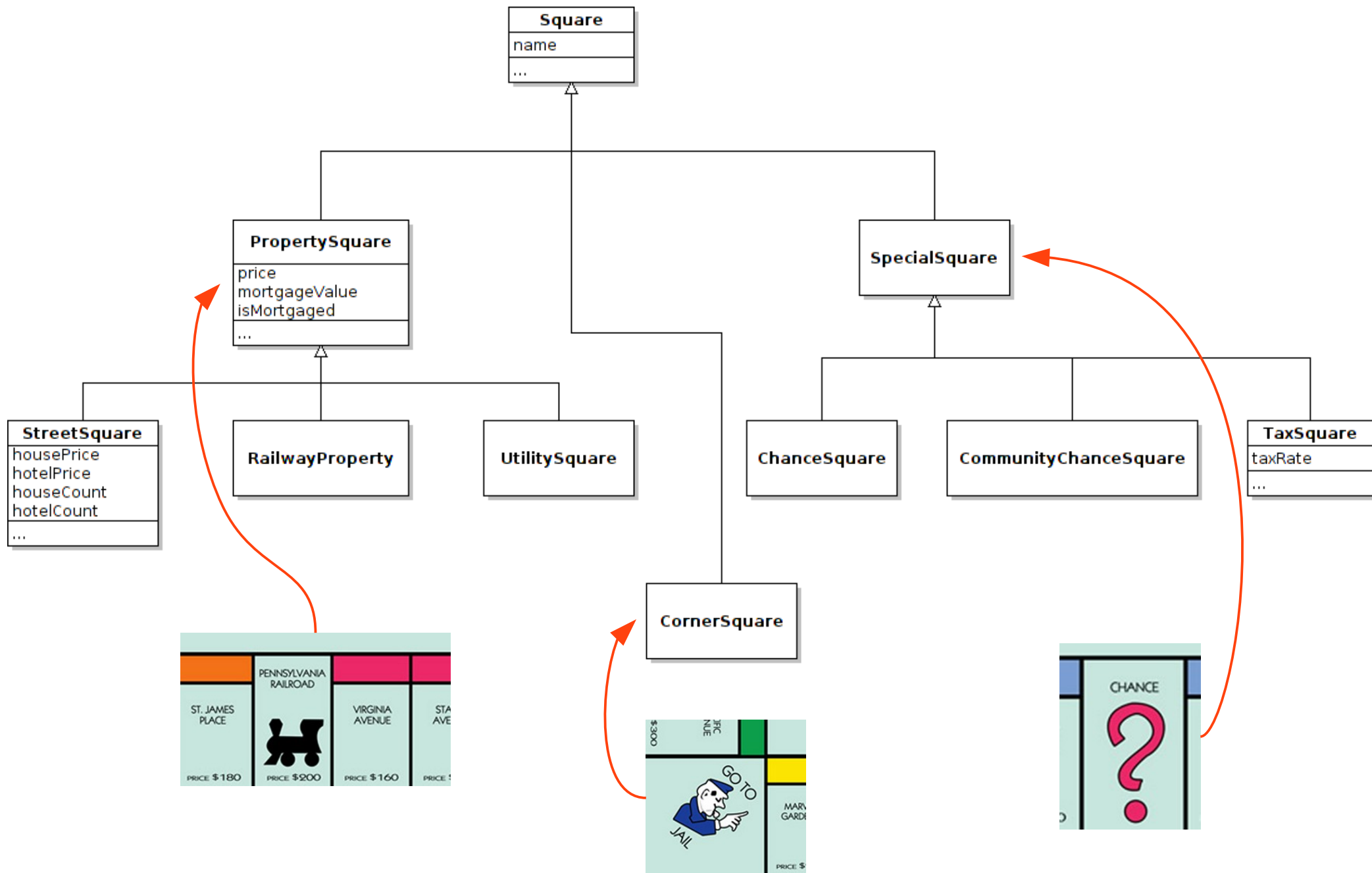


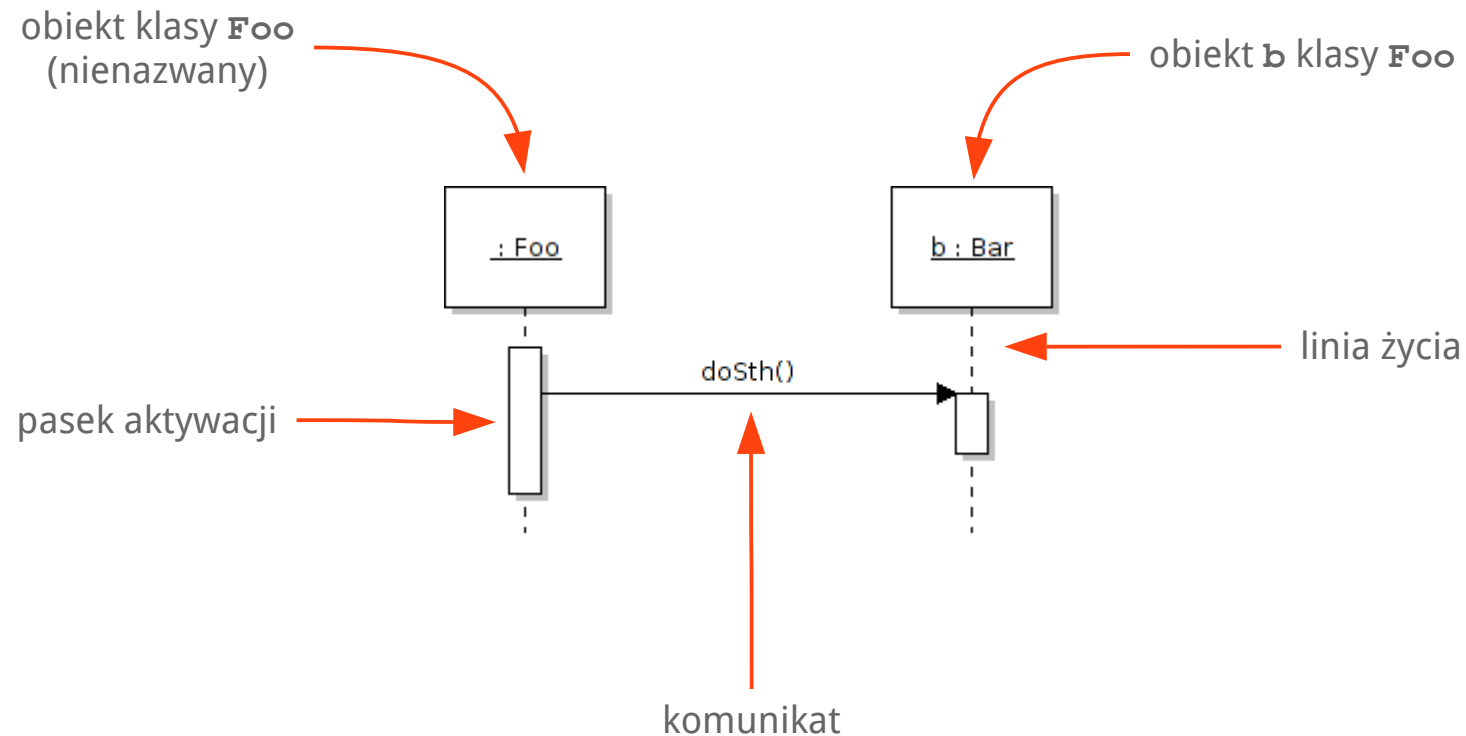
Diagram klas jest **statyczny**.

Przedstawia **budowę** aplikacji
i **zależności** między jej elementami (klasami).

A jak przedstawić **interakcje** zachodzące w aplikacji?

Jak przedstawić interakcje?

Przykładem diagramu przedstawiającego interakcje jest **diagram sekwencji**.



Rzut kośćmi – diagram sekwencji

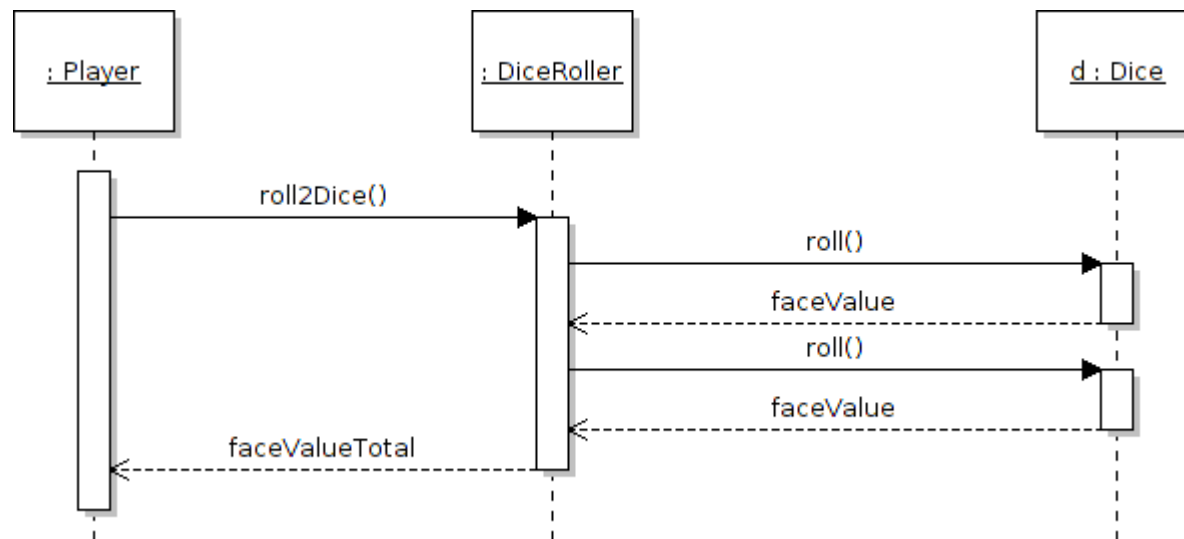
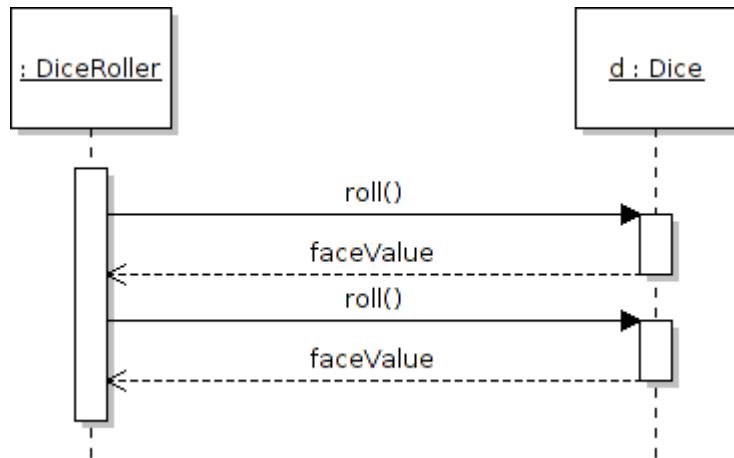
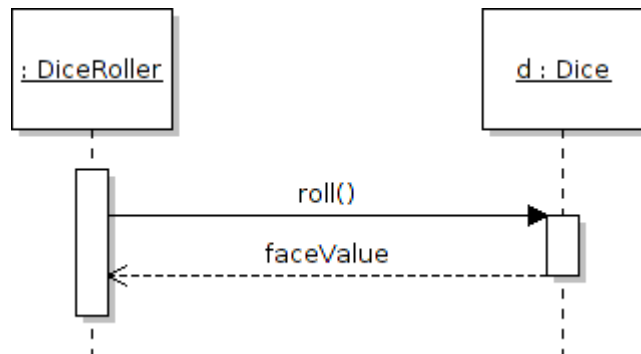
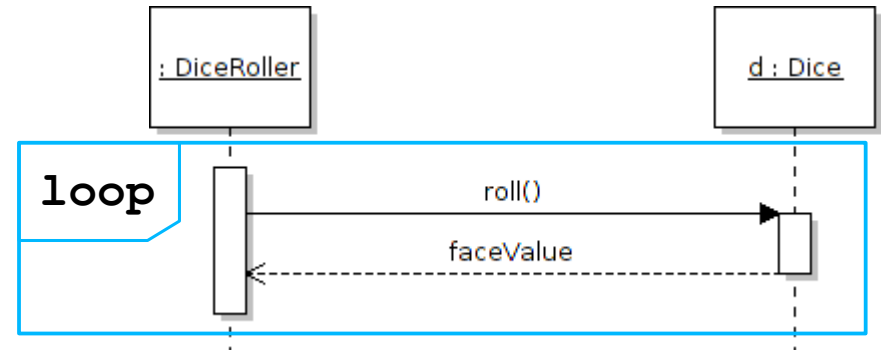


Diagram sekwencji – notacje



=



=

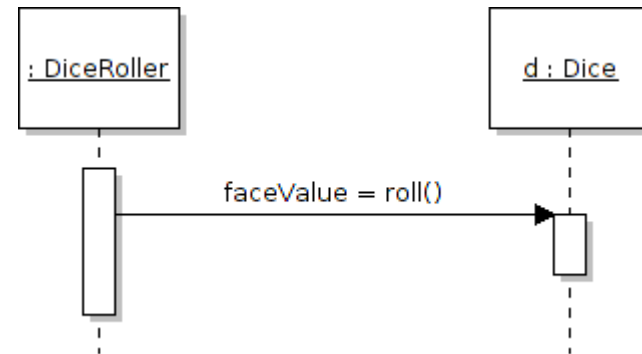


Diagram aktywności

Sprawy związane z przebiegiem gry najlepiej byłoby przedstawić na diagramie aktywności. Oto przykład:

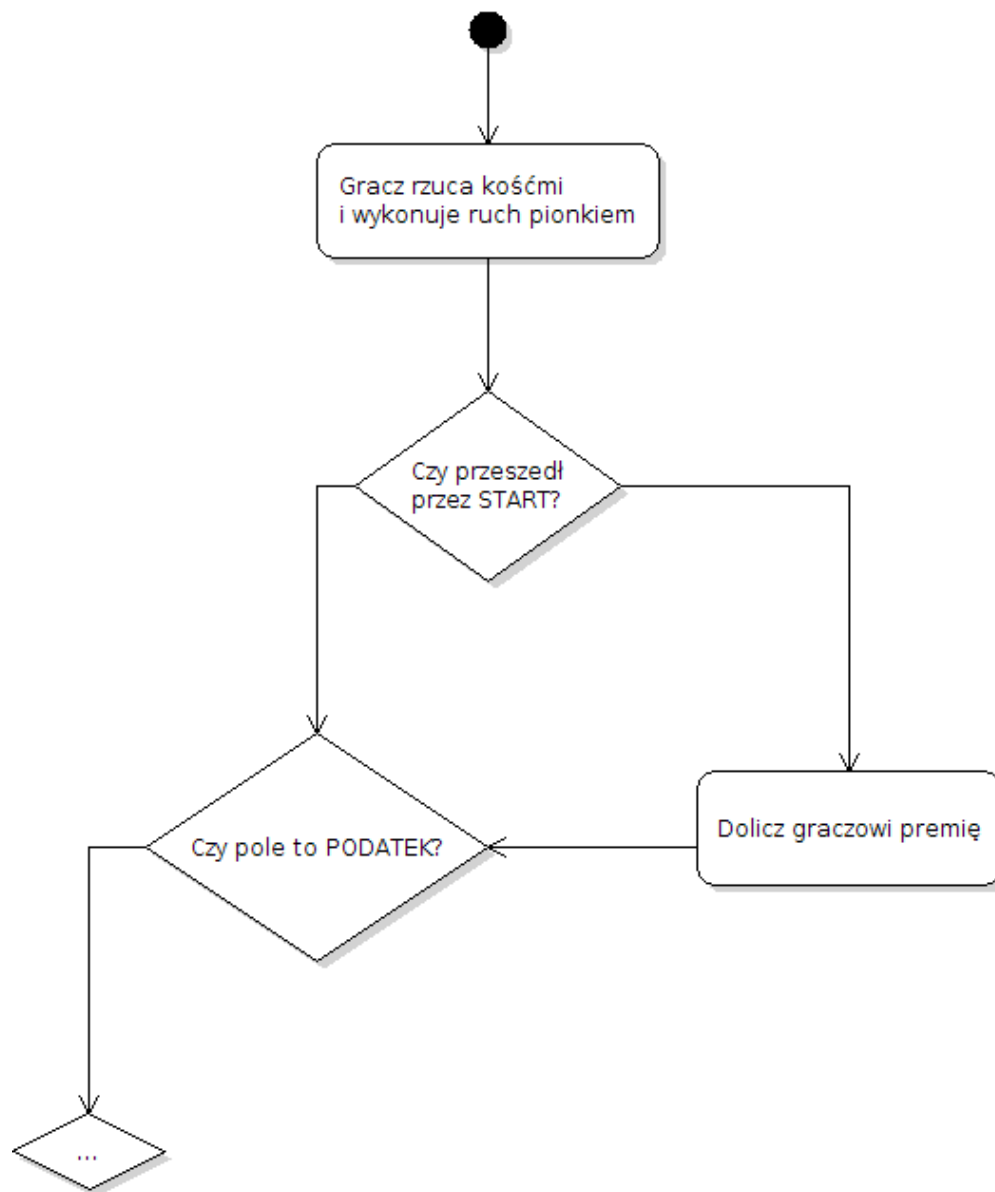
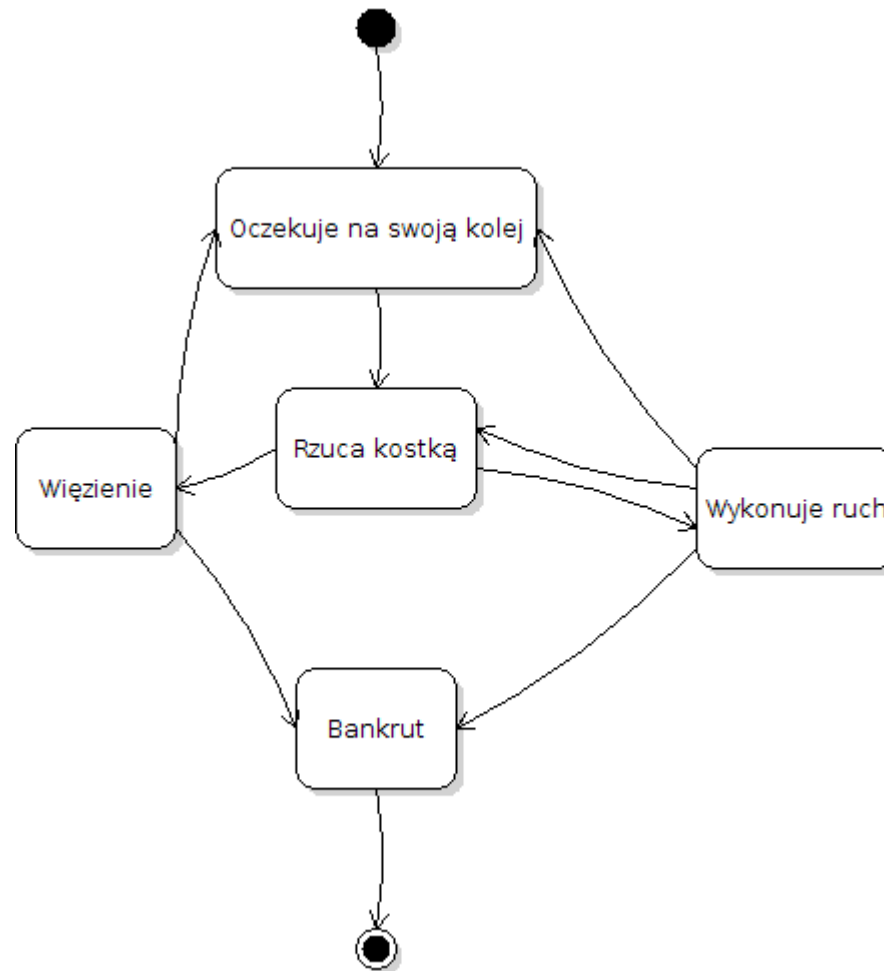


Diagram stanów

Z kolei diagram stanów doskonale się sprawdzi do opisu stanu, w jakim znajduje się gracz:



Podsumowanie

Diagramy UML przydają się do obrazowego przedstawienia budowy i działania systemu.

Zwykle wykorzystuje się tylko kilka typów diagramów. Najczęściej są to:

- diagram **przypadków użycia**,
- diagram **klas**,
- diagram **sekwencji**,
- diagram **aktywności** lub **stanów**.

Diagramy stają się częścią dokumentacji.

Należy przedstawiać na nich najważniejsze aspekty budowy i działania systemu.