

Model use case

- Dr hab. inż. Ilona Bluemke

MODEL USE CASE

Bibliografia:

A. Cockburn -" Jak pisać efektywne przypadki użycia", WNT 2004

<http://www.usecases.org>

Model Use Case

- Przedstawia system z punktu widzenia użytkownika (różnych klas użytkowników systemu).
- Modeluje zachowanie systemu w odpowiedzi na polecenia użytkownika.

Na tym etapie tworzone są diagramy „Use Case”

Posłużą one do następnych etapów projektowania oraz do końcowego testowania systemu pod kątem spełniania wymogów użytkowników.

Określa **CO** system robi

Model logiczny

Przedstawia system w postaci klas, powiązań i interakcji między nimi, zachowań obiektów należących do tych klas oraz sekwencji działań systemu.

Na tym etapie tworzy się następujące diagramy:

- klas, obiektów
- sekwencji (interakcji)
- współpracy
- przejść stanów

Określa **CO jest** w systemie, **JAK** system działa

Model implementacyjny i wdrożeniowy

Model implementacyjny

- Przedstawia system jako moduły, podsystemy, zadania. Na tym etapie powstaje diagram komponentów.

Model wdrożeniowy (Deployment)

- Modeluje fizyczne rozmieszczenie modułów systemu na komputerach. Uwzględnia wymagania sprzętowe, obszary krytyczne.
- Na tym etapie tworzymy diagramy rozmieszczenia (deployment).

Use Case – przykład użycia

Opisuje pewne zachowanie systemu, interakcje systemu ze środowiskiem zewnętrznym (człowiek, inny system – aktor)

Ciąg wykonywanych przez system akcji, które na żądanie aktora realizują jego cele i dostarczają mierzalne wyniki.

Reprezentuje wymagania funkcjonalne

use case – **CO** system robi a **NIE** jak to robi

Elementy diagramów use case

- aktor
- przypadek użycia
- relacje (asocjacje, zależności)

Aktor (osoba, system zewnętrzny):

- korzysta z systemu
- dostarcza/odbiera dane do/z systemu
- administruje systemem

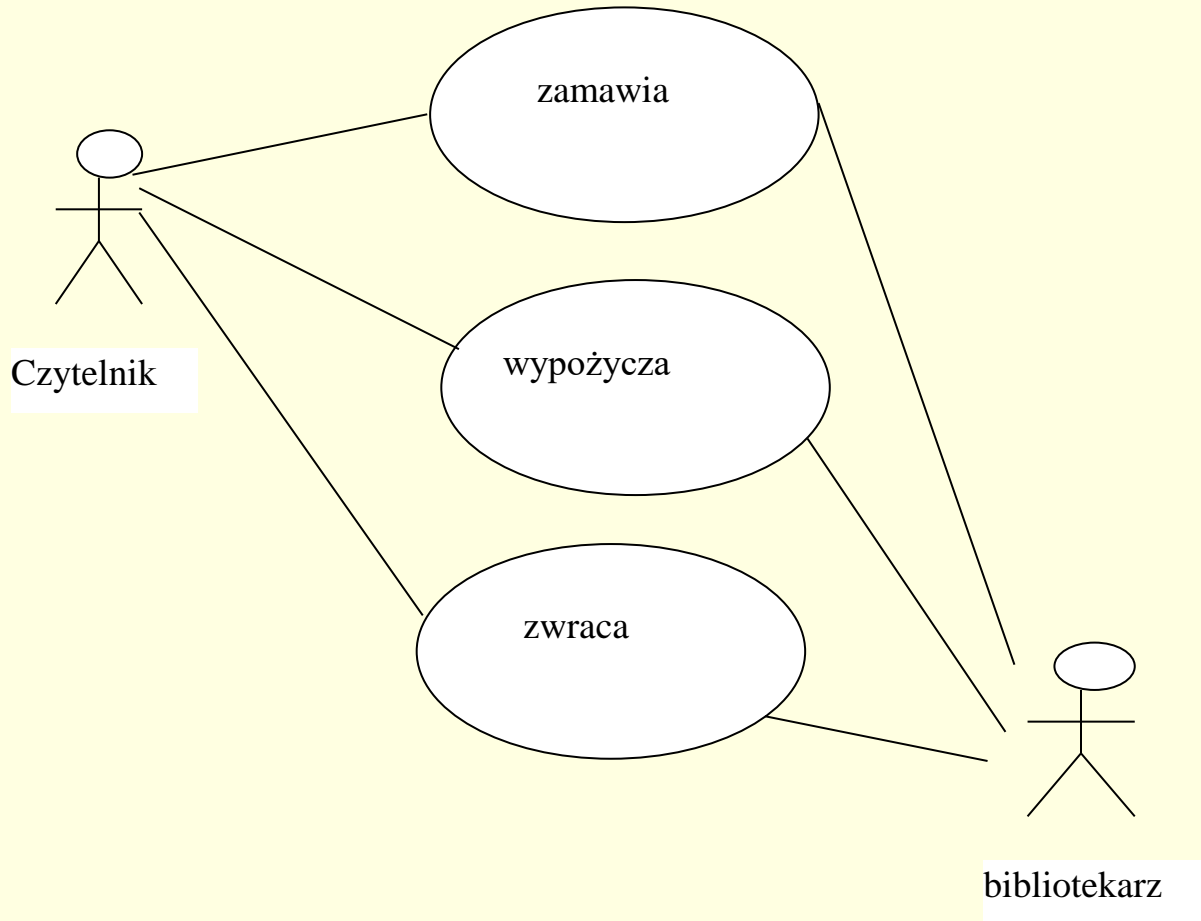
Aktor

Aktor: przyczyna napędzająca przypadki użycia, sprawca zdarzeń powodujących uruchomienie przypadku użycia, odbiorca danych wyprodukowanych przez przypadki użycia.

Aktor – osoba, organizacja, inny system komputerowy.

Grupa osób pełniących pewną rolę, a nie konkretna osoba

Przykład diagramu use case



Działania wstępne

rozpoznanie dziedziny problemu

- wywiady z udziałowcami
- określenie użytkowników i ich punktów widzenia
- modelowanie procesów biznesowych
- źródła danych
- wizja systemu

Strukturalizacja use case'ów

- przykład użycia może być specjalizacją innego (**generalizacja**)
- przykład użycia może być włączany jako część innego (**<<include>>**)
- przykład użycia może rozszerzać zachowanie innego (**<<extend>>**)

Relacje

- **Zależność** (dependency) między elementami, zmiana w jednym może wpływać na drugi, skierowanie pokazuje kierunek zależności . - - - - ->
- **Generalizacja** (dziedziczenie) —————>
- **Asocjacja** (association) powiązanie.
 - Dwukierunkowa : —————
 - Jednokierunkowa: —————>

Scenariusz przypadku użycia

Opisuje się sekwencję zdarzeń:

- jak się rozpoczyna,
- przepływ zdarzeń,
- jak się kończy,
- jakie są interakcje z aktorem.

Tekst nieformalny, tekst strukturalny (z warunkami początkowymi i końcowymi), pseudokod

Format opisu przypadku użycia (1)

Nazwa: <W postaci wyrażenia czasownikowego>

Kontekst użycia: <Cel, normalne warunki wystąpienia>

Zakres i poziom: <Czy przypadek użycia dotyczy całego przedsiębiorstwa, wybranego systemu czy fragmentu oprogramowania? Na jakim poziomie szczegółowości jest opisany?>

Aktor główny: <Nazwa głównego aktora, opis jego roli>

Pozostali aktorzy i udziałowcy: <Nazwy aktorów, ich interesy>

Wyzwalacze / Inicjacja: <Zdarzenie powodujące rozpoczęcie przypadku użycia>

Warunki początkowe: <Co system zapewnia przed zezwoleniem na rozpoczęcie przypadku użycia?>

Format opisu przypadku użycia (2)

Warunki końcowe:

- **Gwarancje powodzenia:** <Warunki spełnione po pomyślnym wykonaniu głównego scenariusza przypadku użycia>
- **Minimalne gwarancje:** <Minimalne wymagania prawdziwe na końcu każdego przebiegu przypadku użycia (również niepoprawnego)>

Główny scenariusz powodzenia / Przepływ podstawowy:

<Numer kroku> <Opis akcji>
<Numer kroku> <Opis akcji>

Przepływy alternatywne:

<Numer zmienionego kroku> <Opis akcji>

Punkty rozszerzenia: <Miejsca i warunki rozszerzeń>

Specjalne wymagania (np. нефunkcjonalne):

Scenariusz główny –przykład *zamawia książkę*

1. System prezentuje ekran wyszukiwania.
2. **Czytelnik** wprowadza dane bibliograficzne.
3. System przeszukuje katalog i wyświetla listę tytułów.
4. **Czytelnik** przegląda listę i wybiera tytuł.
5. System wyświetla listę książek.
6. **Czytelnik** zamawia wolną książkę.
7. System wyświetla okno logowania.
8. **Czytelnik** wprowadza nazwę i hasło.
9. System autoryzuje czytelnika.
10. System potwierdza przyjęcie zamówienia.

Scenariusze alternatywne

Scenariusz **alternatywny 1** (odmowa autoryzacji)

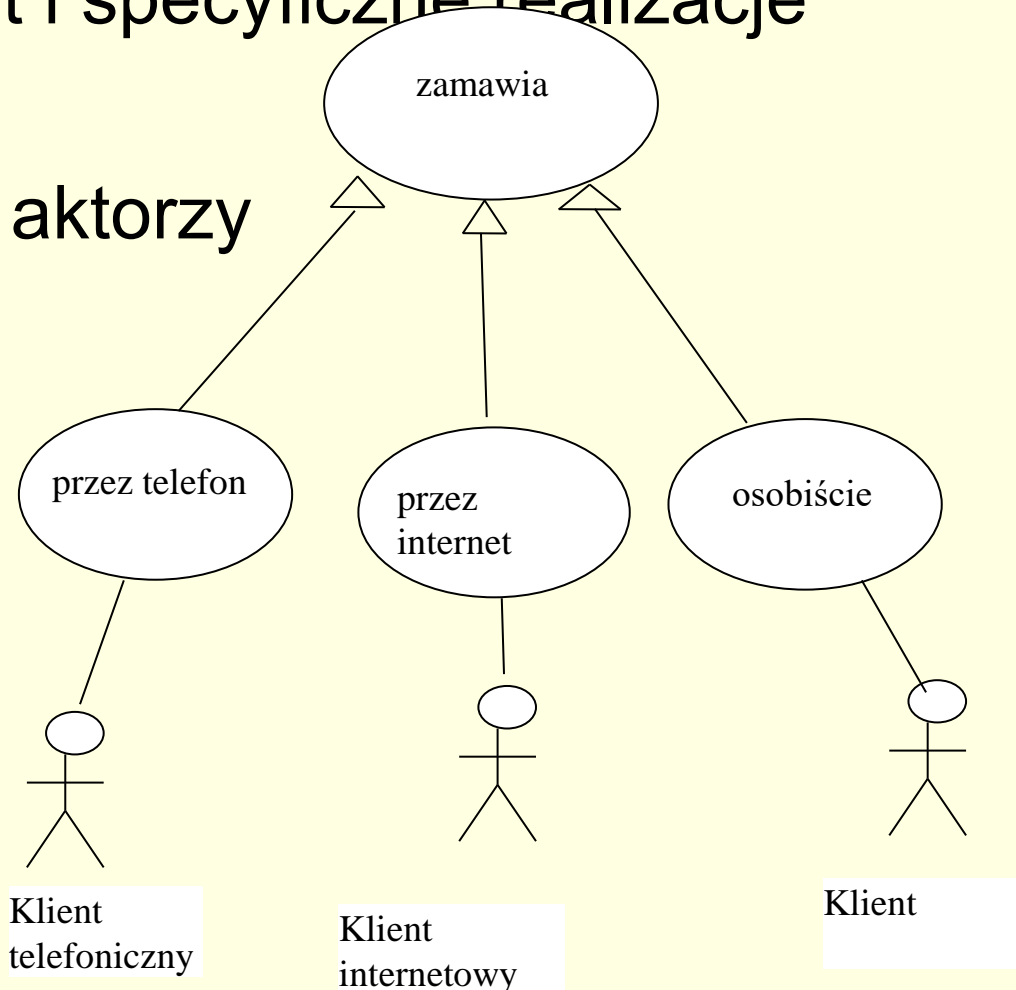
- 1 – 8 Jak w scenariuszu głównym.
- 9 System odmawia autoryzacji: powrót do kroku 7.

Scenariusz **alternatywny 2** (brak wolnej książki)

- 1 – 5 Jak w scenariuszu głównym.
- 6. Brak wolnej książki: czytelnik zapisuje się do kolejki.
- 7. System wyświetla okno logowania.
- 8. *Czytelnik* wprowadza nazwę i hasło.
- 9. System autoryzuje czytelnika.
- 10. System potwierdza zapisanie do kolejki.

Generalizacja use case

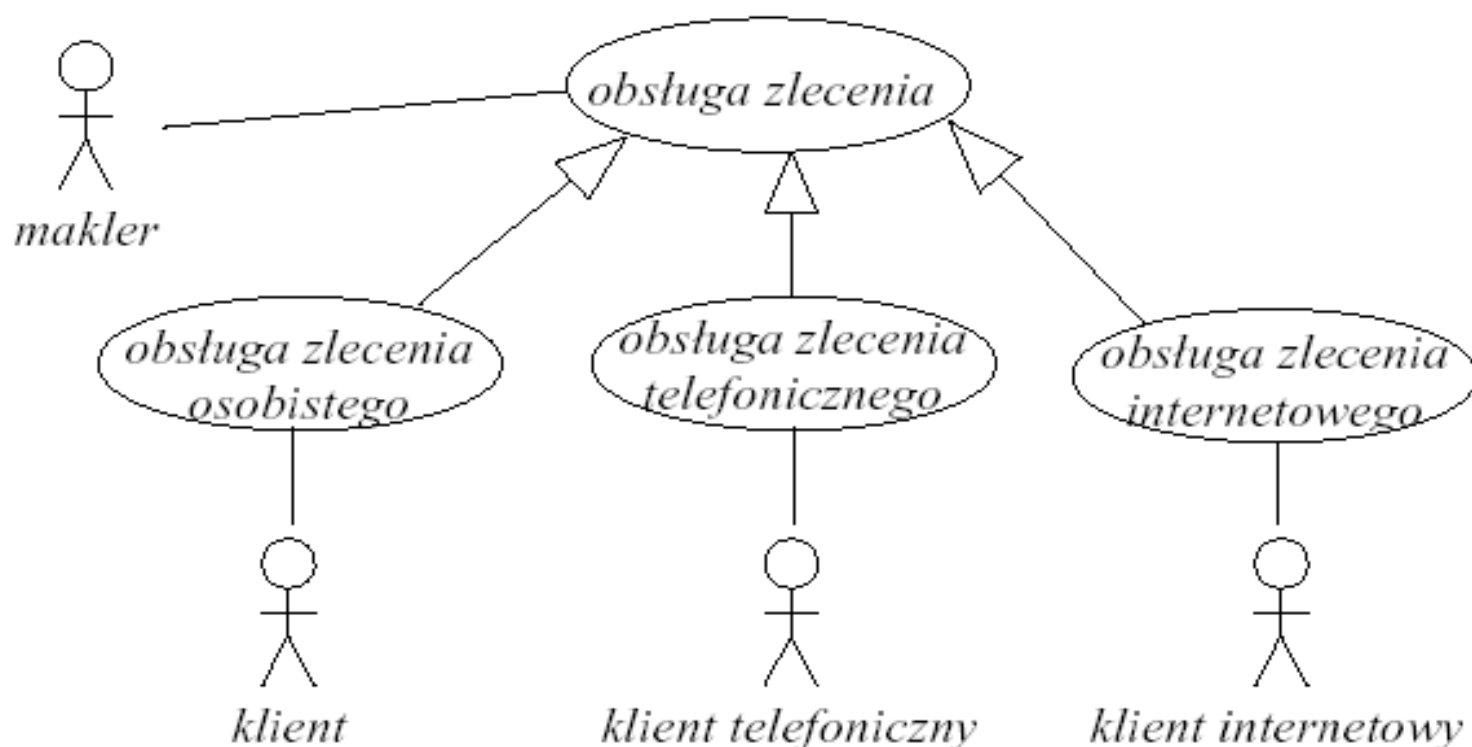
- Ogólny schemat i specyficzne realizacje
- Ten sam cel
- Mogą być różni aktorzy



Generalizacja

- Ogólny schemat i specyficzne realizacje
- Ten sam cel
- Różni aktorzy

Przykład: biuro maklerskie.



obsługa zlecenia

1. Makler przyjmuje zlecenie i wprowadza do systemu BM.
2. System BM blokuje środki na rachunku klienta.
3. System BM przekazuje zlecenie do systemu Warset.

obsługa zlecenia telefonicznego

- 1.1. Makler odbiera i wprowadza do systemu BM dane klient .
- 1.2. Makler odbiera i wprowadza do systemu hasło klienta.
- 1.3. System potwierdza autoryzację klienta.
- 1.4. Makler odbiera i wprowadza zlecenie do systemu BM.
- 2 – 3. Jak w przypadku generalizującym.

obsługa zlecenia internetowego

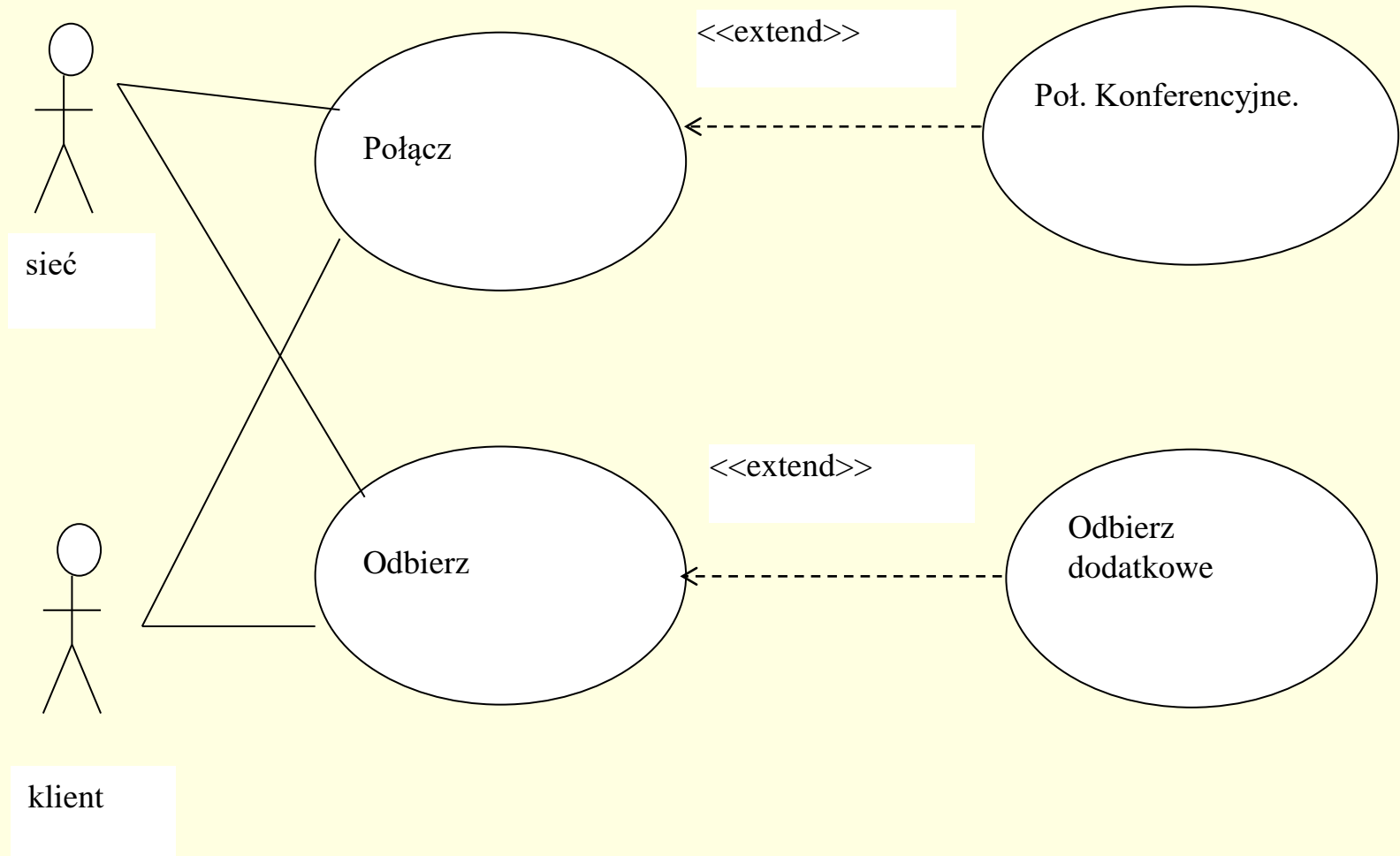
.....

obsługa zlecenia osobistego

Rozszerzenie <<extend>>

- Typowy schemat i dodatkowe czynności
- Określone punkty rozszerzenia
- Brak odrębnych aktorów
- Przykład użycia podstawowy może wystąpić sam, ale w pewnych warunkach jego zachowanie może być rozszerzone – w pewnych punktach (extension point), przez inny przypadek), pokazane opcjonalne zachowanie systemu
- Wariant, opcja

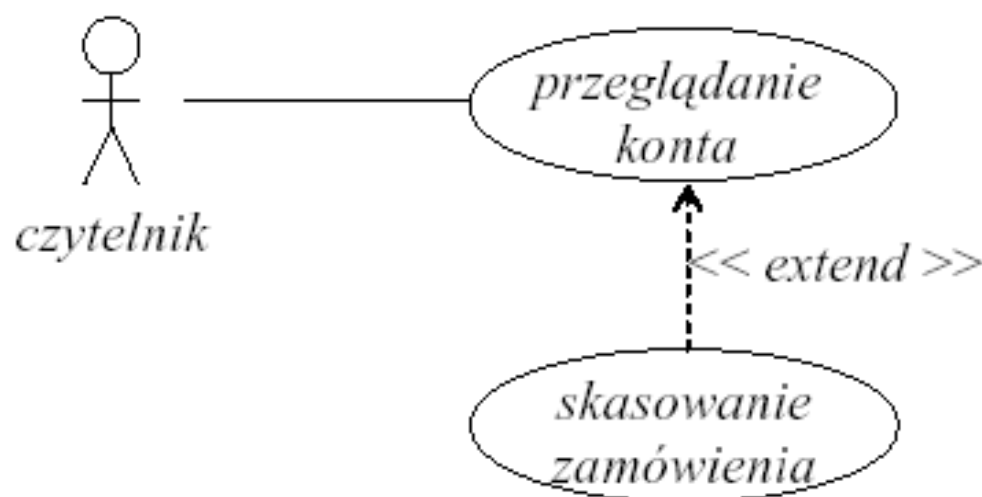
Przykład <<extend>>



Rozszerzenie

- Typowy schemat i dodatkowe czynności
- Określone punkty rozszerzenia
- Brak odrębnych aktorów

Przykład: system biblioteczny.



przeglądanie konta

1. Czytelnik wybiera opcję przeglądania konta.
2. System wyświetla okno logowania.
3. Czytelnik wprowadza nazwę i hasło.
4. System autoryzuje czytelnika.
5. System wyświetla stan konta (***kasowanie***).
6. Czytelnik wybiera następną opcję.

skasowanie zamówienia

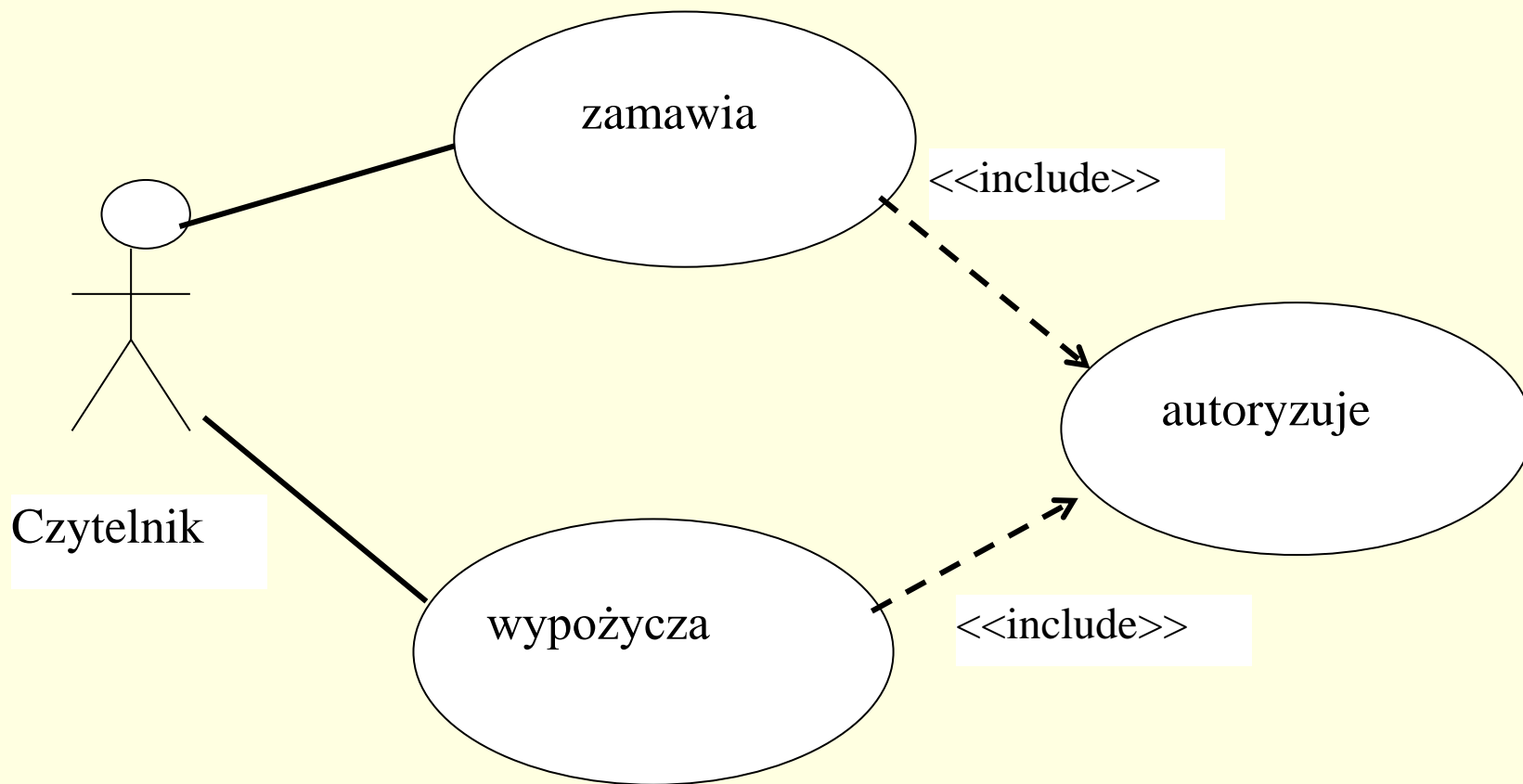
wstawić w punkcie rozszerzenia ***kasowanie***:

1. Czytelnik kasuje zamówienie.
2. System potwierdza skasowanie zamówienia.

Zawieranie <<include>>

- Przykład użycia włącza zachowanie innego, przykład „included” nie może być samodzielny
- Wyodrębnienie fragmentu przypadku użycia
- Możliwe fragmenty wspólne
- Brak odrębnych aktorów

Przykład <<include>>



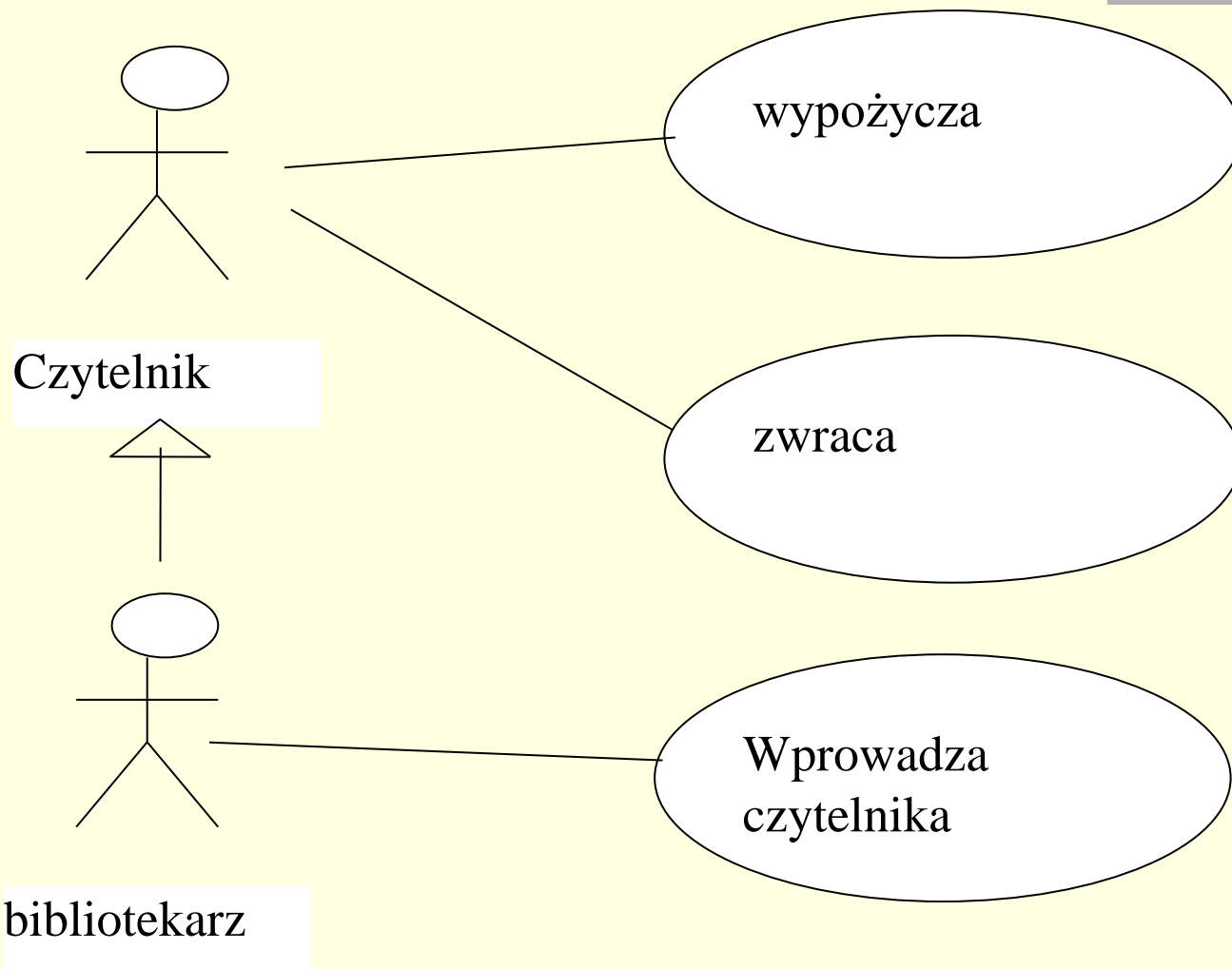
zamówienie książki

1. System prezentuje ekran wyszukiwania.
2. *Czytelnik* wprowadza dane bibliograficzne.
3. System przeszukuje katalog i wyświetla listę pozycji (tytułów).
4. *Czytelnik* przegląda listę i wybiera pozycję.
5. System wyświetla listę książek.
6. *Czytelnik* zamawia wolną książkę.
7. **Wykonaj autoryzację czytelnika.**
8. System potwierdza przyjęcie zamówienia.

autoryzacja czytelnika

1. System wyświetla okno logowania.
2. *Czytelnik* wprowadza nazwę i hasło.
3. System autoryzuje czytelnika.

Generalizacja aktorów



Użycie generalizacji aktorów

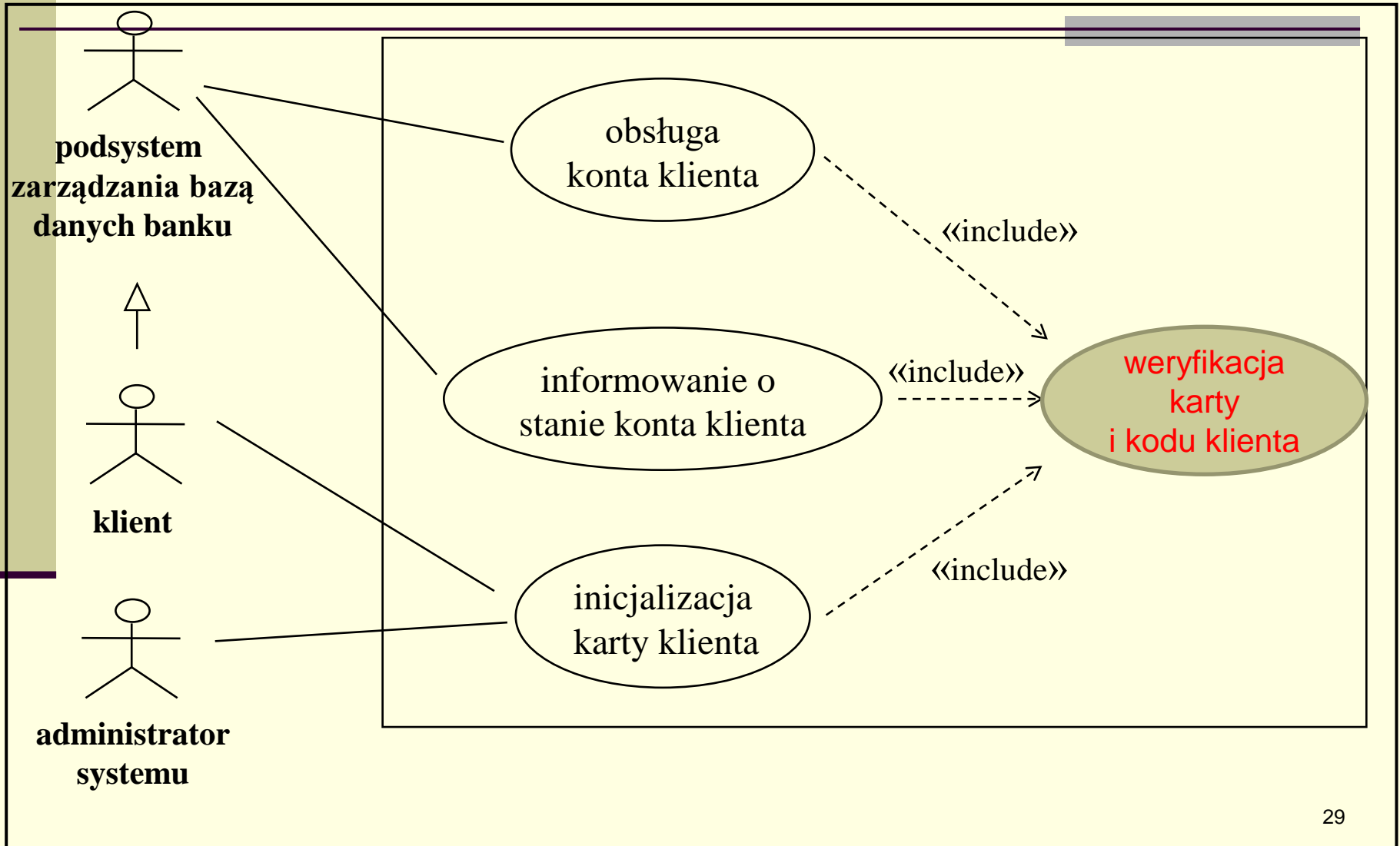
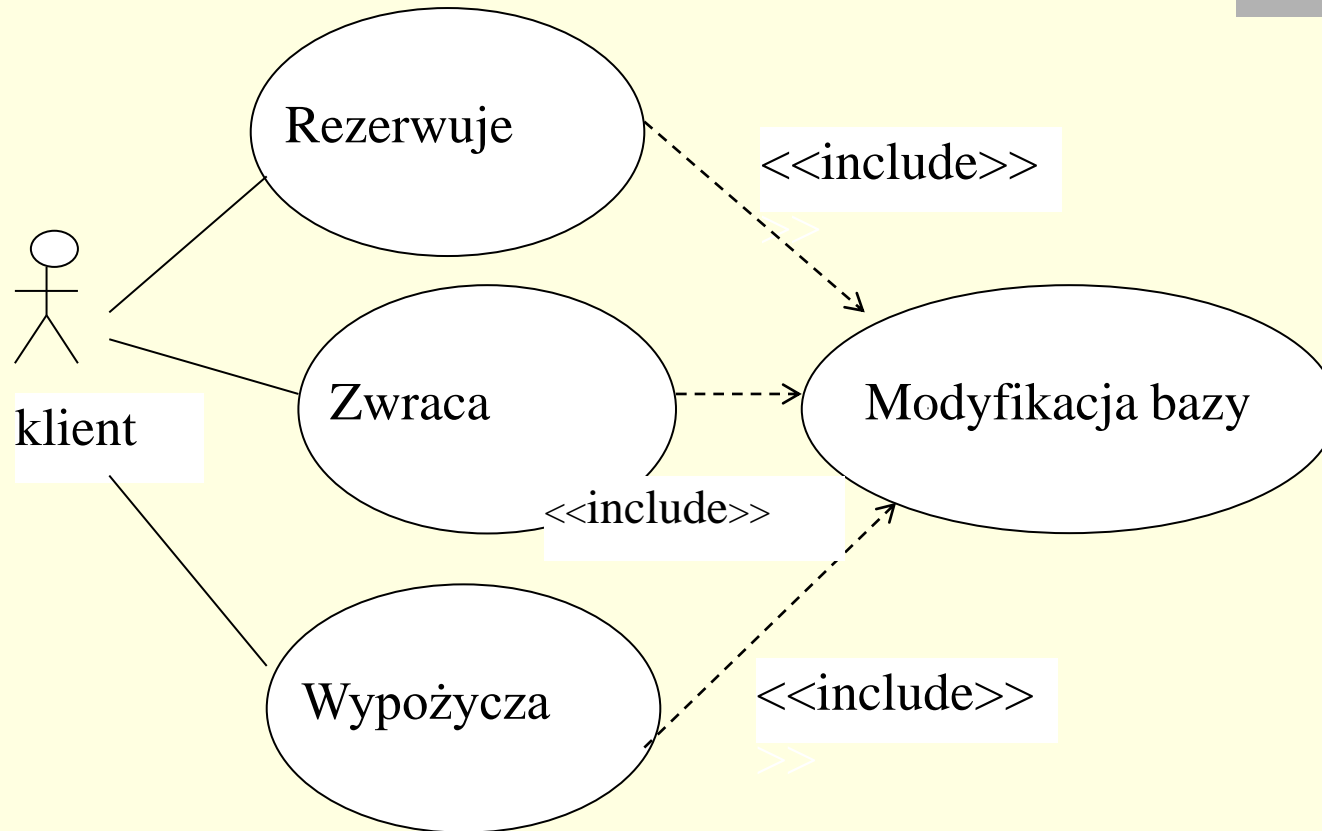


Diagram przypadków użycia z <<include>>



Heurystyka tworzenia przypadków użycia

4 kroki wg Cockburna

1. Zidentyfikuj aktorów i ich cele.
2. Napisz główny scenariusz powodzenia
3. Zidentyfikuj i wylistuj możliwe rozszerzenia (zwłaszcza możliwe błędy)
4. Opisz jak system obsługuje każde rozszerzenie (w tym każdy błąd)

Zalety i wady przypadków użycia

+

- przedstawiają wymagania funkcjonalne w prostym do czytania formacie tekstowym

-

- pokazują tylko wymagania funkcjonalne – bez niefunkcjonalnych
- projekt nie jest tylko tworzony w kategoriach przypadków użycia

Diagram przypadków użycia – podsumowanie

- Przedstawia wymagane zachowanie systemu
- Modeluje kontekst systemu

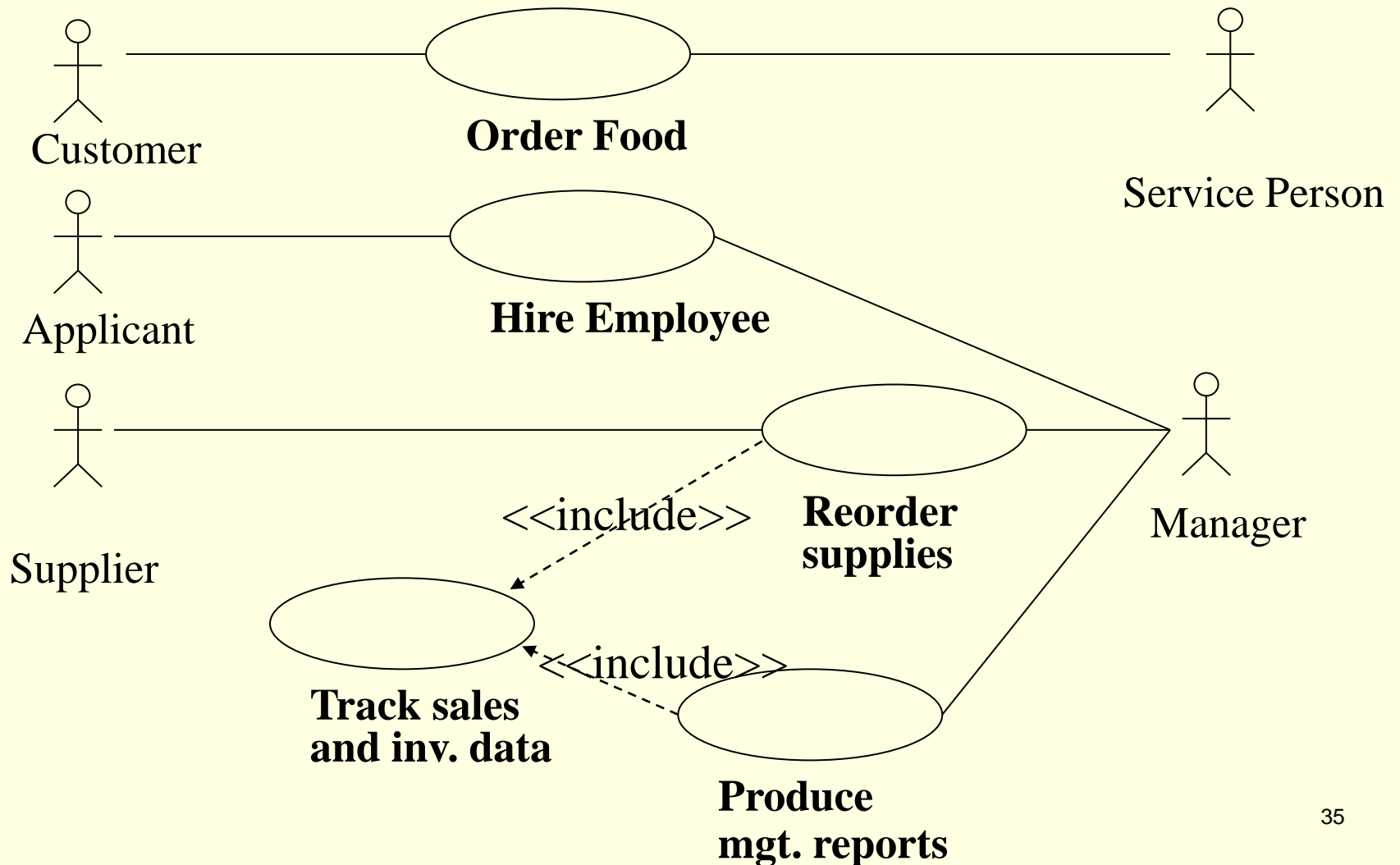
Czynności:

1. Identyfikacja aktorów
2. Określenie przypadków użycia
 - zadania aktora (np. wprowadzanie danych, prezentacja danych, utrzymanie systemu)
3. Uporządkowanie modelu (generalizacja, rozszerzenie, zawieranie)
4. Dokumentacja modelu

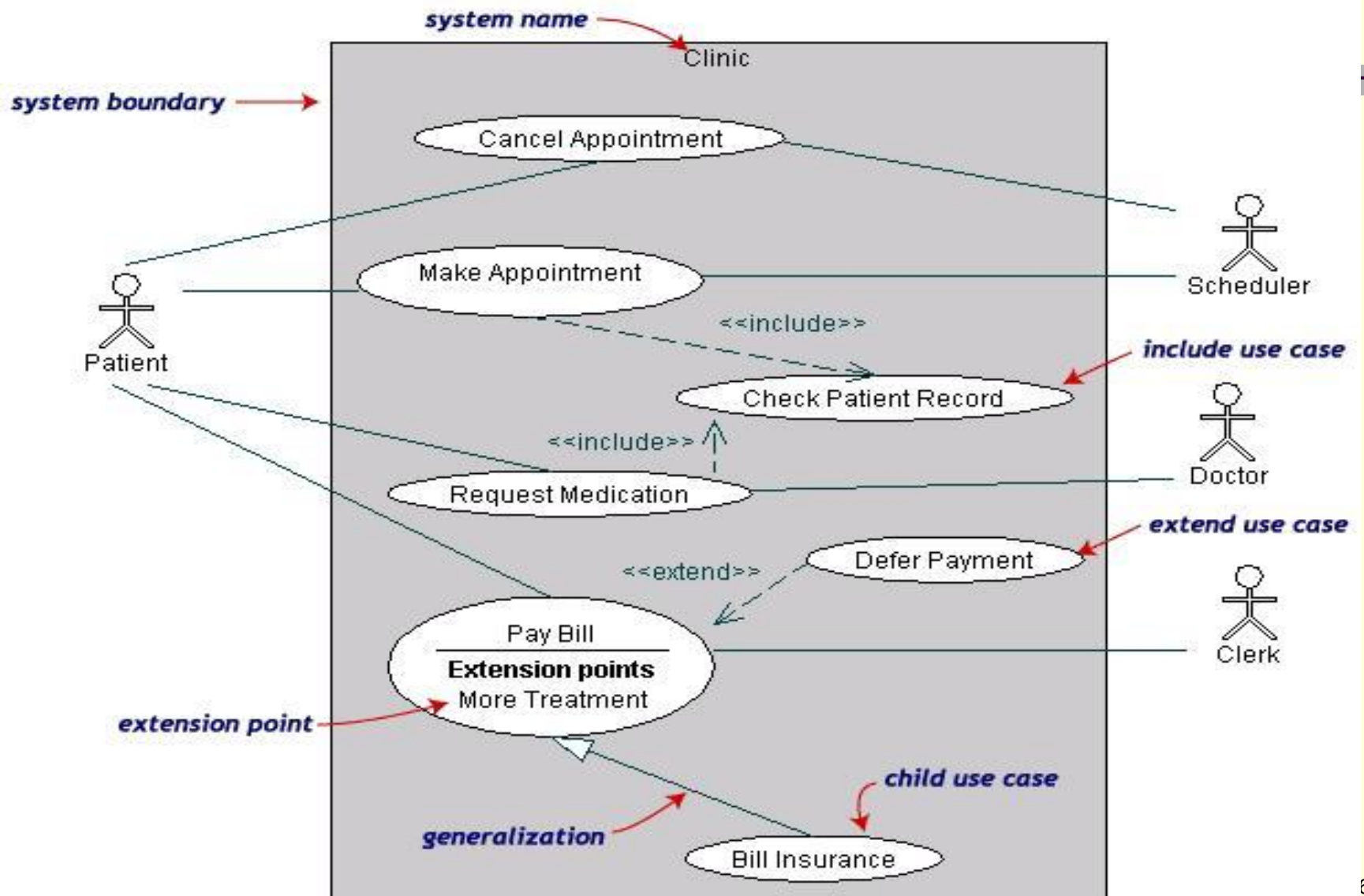
Diagram przypadków użycia – podsumowanie -2

- Przypadki użycia **nie dotyczą problemów projektowych**:
 - struktury danych
 - współbieżności operacji
 - struktury programu
- Diagram przypadków użycia **nie pokazuje kolejności**, w jakiej przypadki mogą być wykonywane. Właściwą kolejność określają diagramy realizowane w modelu logicznym.

Przykłady przypadków użycia

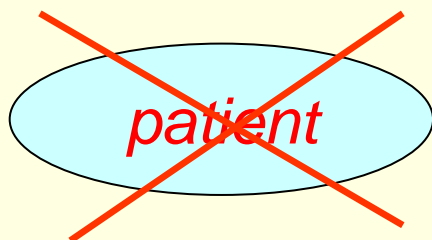


Przykłady

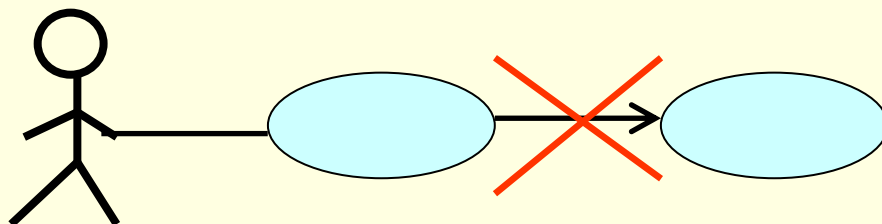


Use Case model – typowe błędy

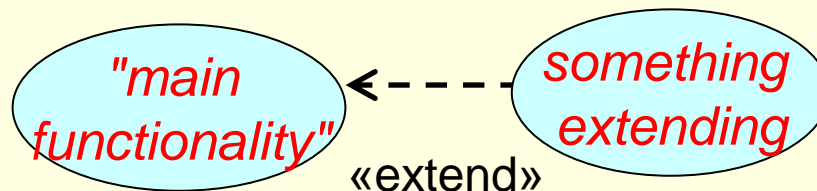
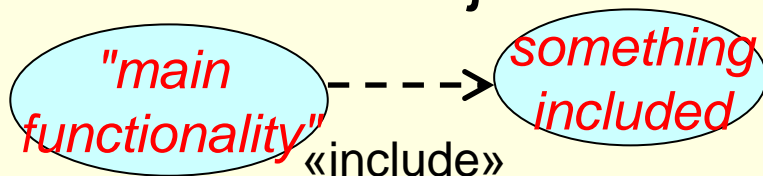
Złe nazwy (rzeczowniki zamiast czasowników):



Niewłaściwe użycie relacji



Kierunki relacji



Use cases vs. internal features

- consider software to run a cell phone:



Use Cases

- call someone
- receive a call
- send a message
- memorize a number

Point of view: user

Internal Functions

- transmit / receive data
- energy (battery)
- user I/O (display, keys, ...)
- phone-book mgmt.

*Point of view: developer /
designer*