

**KATEDRA INFORMATYKI
STOSOWANEJ PŁ**

**METODOLOGIA INŻYNIERII
OPROGRAMOWANIA**

Wymagania i Modele Przypadków Użycia

Przygotował: mgr inż. Radosław Adamus

Wprowadzenie

Podstawą każdego projektu, którego celem jest budowa oprogramowania są wymagania, czyli warunki, jakie na produkt końcowy nakłada klient. Warunki te są punktem wyjścia do zdefiniowania cech systemu, opisujących jego funkcjonalność, wydajność, wiarygodność czy ergonomiczność. Należy zwrócić uwagę na to, że osobą, która weryfikuje system pod kątem spełnienia bądź niespełnienia oczekiwań, jest klient. Dlatego tak ważne jest, aby wymagania były zdefiniowane w sposób możliwie najpełniejszy i jednoznaczny. Dodatkowo sposób zapisu tych wymagań powinien służyć nie tylko projektantom systemu, ale również musi być zrozumiały i jasny dla klienta, dzięki czemu może on dokonać weryfikacji samego procesu pozyskiwania wymagań. Twórcy systemu zyskują wówczas poczucie, że to, co budują jest tym, czego użytkownicy i klienci potrzebują.

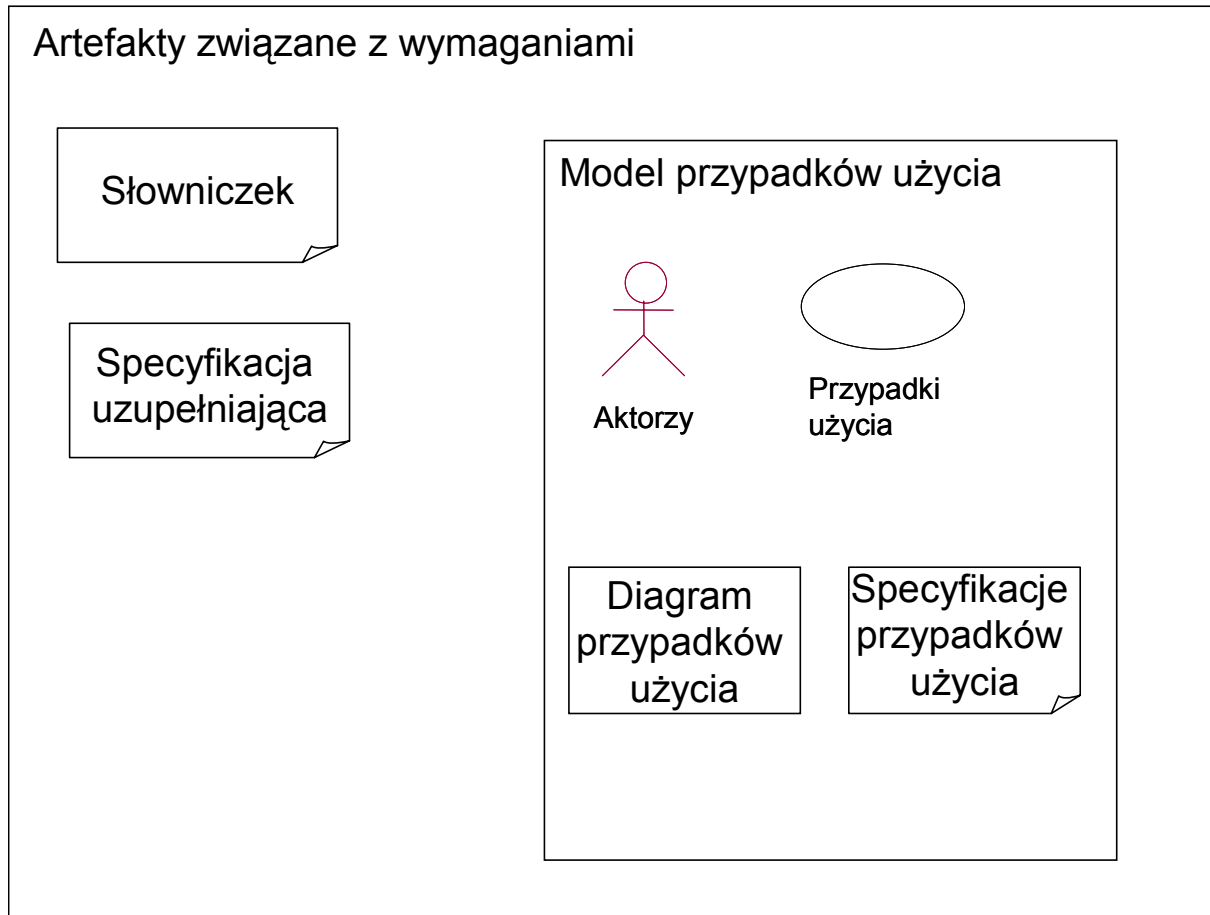
Przykładem sposobu zapisu wymagań jest *Model Przypadków Użycia*. Opisuje on to, co system będzie robił. Jest swego rodzaju kontraktem pomiędzy klientami a twórcami. Umożliwia użytkownikom i klientom weryfikację czy system jest rzeczywiście tym, czym być powinien. Z punktu widzenia twórców, dzięki modelowi przypadków użycia są oni w stanie dokonać weryfikacji swojej pracy (czy budujemy to, czego się od nas oczekuje).



Proces wytwarzania oprogramowania, którego podstawą są przypadki użycia nazywa się *procesem sterowanym przypadkami użycia* (ang. *Use-case driven*)

Artefakty związane z wymaganiami

Rysunek 1 pokazuje elementy projektu związane z wymaganiami.



Rys.1 Artefakty związane z wymaganiami.

Słowniczek

Przy tworzeniu systemu dla określonej dziedziny problemowej (biznesowej) napotyka się wiele pojęć, których znaczenie może być nie do końca zrozumiałe dla projektantów systemu. Jednocześnie wiele pojęć z dziedziny informatyki może być obce dla klientów i użytkowników. Dlatego dla zachowania spójności i umożliwienia poprawnej komunikacji pomiędzy członkami zespołu projektowego oraz klientami należy zdefiniować pojęcia, które będą wykorzystywane w całej dokumentacji projektowej. Definicja takich pojęć powinna być umieszczona w tzw. *Słowniku Pojęć*.

Aby *Słownik Pojęć* spełniał swoje zadanie, (referencja dla znaczeń wyrażen wykorzystywanych w dokumentacji) należy w trakcie tworzenia dokumentacji

projektowej (wymagania, projekt, dokumentacja użytkownika) konsekwentnie używać nazw w nim zdefiniowanych.

Specyfikacja uzupełniająca

Nie wszystkie wymagania da się wyrazić w opisanym poniżej *modelu przypadków użycia*. W przypadku tzw. wymagań нефункциональных niezbędna jest dodatkowa dokumentacja zwana *specyfikacją uzupełniającą*. Do wymagań opisanych w tym dokumencie należą m.in.: oczekiwana wydajność systemu; odporność na błędy, czyli niezawodność; zdolność przetwarzania danych przy dużym ich natężeniu, itp.

Mimo tego, że dokument ten jest nazywany specyfikacją uzupełniającą to pełni on bardzo ważną rolę w opisie wymagań dla projektowanego systemu.

Model przypadków użycia

Podstawowym zadaniem *modelu przypadków użycia* jest opisanie działania systemu z punktu widzenia jego użytkowników (aktorów w modelu). Innymi słowy model przypadków użycia mówi, **co** system ma robić natomiast nie odpowiada na pytanie **jak** będzie to robił.

Model przypadków użycia składa się z następujących elementów:

- Aktorzy
- Przypadki użycia
- Specyfikacje przypadków użycia

Aktorem jest dowolny byt zewnętrzny, który uczestniczy w interakcji z systemem. Każdy potencjalny aktor może wchodzić w interakcję z systemem na pewną liczbę jemu właściwych sposobów. Każdy z tych sposobów nosi nazwę *przypadku użycia* i reprezentuje przepływ operacji w systemie związany z obsługą zadania zleconego przez aktora w procesie interakcji. Należy pamiętać o tym, że aktor nie oznacza konkretnej osoby.

Przypadki użycia reprezentuje sekwencję operacji, niezbędnych do wykonania zadania zleconego przez aktora, np. potwierdzenie pisma, złożenie zamówienia, itp. Każde takie wyróżnialne zadanie jest opisywane przez osobny *przypadek użycia*.

Zależności pomiędzy wszystkimi przypadkami użycia i wszystkimi aktorami wyraża się w ramach *modelu przypadków użycia* na *diagramach przypadków użycia*.

Specyfikacja przypadku użycia to szczegółowy opis działania systemu dla określonego przypadku użycia.

Aktorzy

Aktorem jest wszystko to, co wymienia dane z systemem i, z punktu widzenia systemu, należy do środowiska zewnętrznego. Zazwyczaj aktorem jest osoba, ale może nim być także pewna organizacja (np. biuro prawne) lub inny system komputerowy. **Aktor** modeluje **grupę osób pełniących pewną rolę**, a nie **konkretną osobę**. Jedna osoba może wchodzić w interakcję z systemem z pozycji wielu aktorów; np. być zarówno sprzedawcą, jak i klientem. I odwrotnie, jeden aktor może odpowiadać wielu konkretnym osobom, np. aktor "strażnik budynku".

Wyszukiwanie aktorów

Zanim rozpoczniemy wyszukiwanie przypadków użycia należy wyszukać aktorów naszego systemu. Dopiero, gdy mamy zdefiniowanych użytkowników naszego systemu możemy na tej podstawie określić, jakiej funkcjonalności potrzebują.

Aby poprawnie określić aktorów należy odpowiedzieć na następujące pytania:

- Jaka grupa użytkowników potrzebuje wspomaganie ze strony systemu (np. osoba wysyłająca korespondencję)?
- Jacy użytkownicy są konieczni do tego, aby system działał i wykonywał swoje funkcje (np. administrator systemu)?
- Z jakich elementów zewnętrznych (innych systemów, komputerów, czujników, sieci, itp.) musi korzystać system, aby realizować swoje funkcje.

Po wyszukaniu aktorów, należy ustalić:

- nazwę dla każdego aktora/roli,
- zakresy znaczeniowe dla poszczególnych nazw aktorów oraz relacje pomiędzy zakresami (np. sekretarka → pracownik administracji → pracownik → dowolna osoba). Niekiedy warto ustalić hierarchię dziedziczenia dostępu do funkcji systemu dla aktorów.

Przypadki użycia

Przypadek użycia opisuje zadanie, które wykonuje system dla konkretnego aktora (aktorów). Istotnym jest fakt, że opis ten jest dokonywany z punktu widzenia aktora (użytkownika systemu). Czyli przypadek użycia mówi nam, jakie działania wykonuje system, aby aktor mógł otrzymać określony wynik. Sekwencja zdarzeń przypadku użycia jest inicjowana przez aktora, który zgłasza prośbę wykonania przez system pewnego zadania.

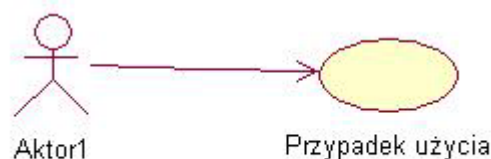
Wskazówki do wyszukiwania przypadków użycia.

- Dla każdego aktora, znajdź funkcje (zadania), które powinien wykonywać w związku z jego działalnością w zakresie zarówno dziedziny przedmiotowej, jak i wspomaganie działalności systemu informacyjnego.
 - Czy aktor powinien być poinformowany o pewnych zdarzeniach, które zachodzą w systemie?
 - Czy aktor musi informować system o zmianach, jakie zachodzą w środowisku zewnętrznym?
 - Jakie informacje muszą być modyfikowane lub tworzone w systemie?
- Staraj się powiązać w jeden przypadek użycia zespół funkcji realizujących podobne cele. Unikaj rozbicia jednego przypadku użycia na zbyt wiele pod-przypadków.
- Nazwy dla przypadków użycia: powinny być krótkie, ale jednoznacznie określające charakter zadania lub funkcji. Nazwy powinny odzwierciedlać czynności z punktu widzenia aktorów, a nie systemu, np. *“wpłacanie pieniędzy”*, a nie *“przyjęcie pieniędzy od klienta”*.

Diagramy przypadków użycia

Diagram przypadków użycia obrazuje:

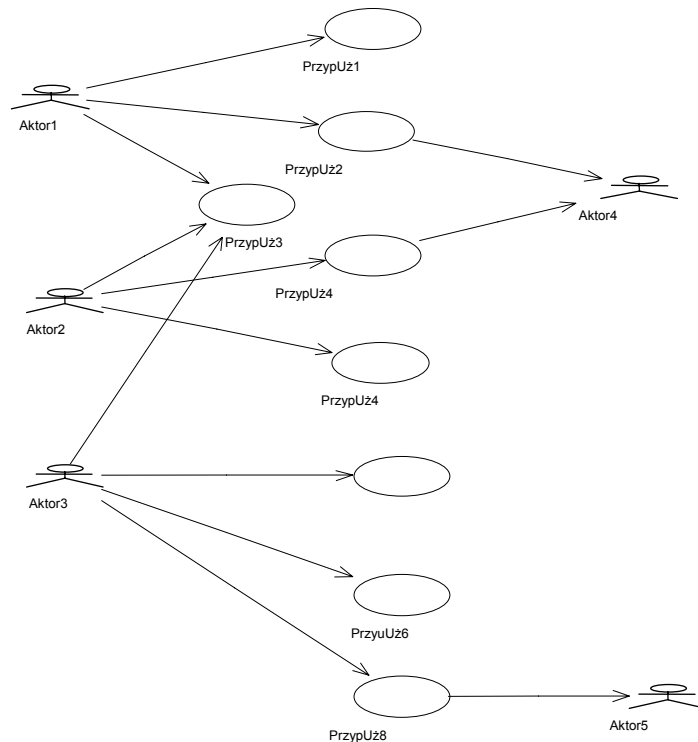
- Związki pomiędzy aktorami a przypadkami użycia – dany przypadek użycia jest powiązany z jednym bądź kilkoma aktorami.



Rys. 2 Powiązanie pomiędzy aktorem i przypadkiem użycia (jednokierunkowe)

- Związki pomiędzy aktorami i aktorami (związek uogólnienia)
- Związki pomiędzy przypadkami użycia a przypadkami użycia – (include, extend, związek uogólnienia)

Diagramy przypadków użycia nie pokazują kolejności, w jakiej ewentualnie mogłyby się wykonywać przypadki użycia.



Rys. 3 Diagram przypadków użycia

Specyfikacje przypadków użycia

Oczywiście samo znalezienie i nazwanie przypadku użycia nie wystarczy, aby w pełni opisać zadania systemu. Dlatego każdy przypadek użycia jest opisany za pomocą *specyfikacji przypadku użycia*. Na specyfikację składają się następujące elementy:

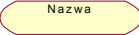

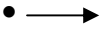



- Nazwa
- Krótki opis
- Przepływ zdarzeń (sterowania)
- Powiązania
- Diagramy aktywności

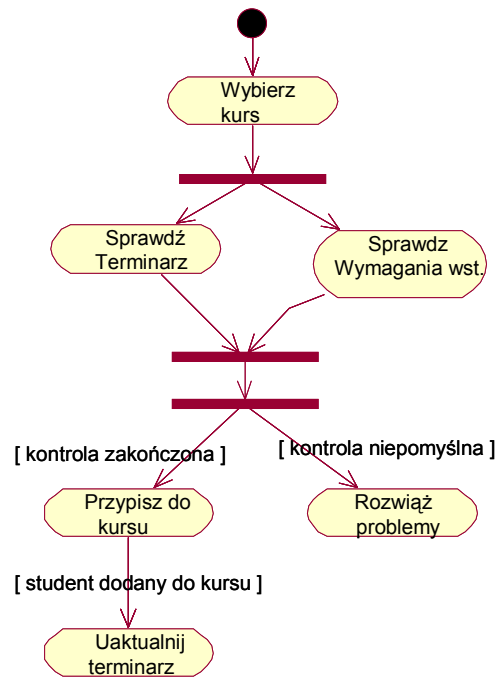
- Diagramy przypadków użycia
- Wymagania specjalne
- Warunki wejściowe (ang. pre-conditions)
- Warunki wyjściowe (ang. post-conditions)
- Inne diagramy

Diagramy Aktywności

Zadaniem diagramu aktywności jest graficzny opis przepływu sterowania, składającego się na działanie wykonywane w przypadku użycia. Diagramy aktywności są opcjonalnym elementem specyfikacji przypadku użycia (nie jest to ich jedyna funkcja w UML).

Elementy diagramu aktywności:

-  **Aktywność** – reprezentuje wykonywanie pewnego działania, lub kroku w przepływie pracy
-  **Blok decyzyjny** – sprawdzenie warunków (ang. guard condition). W wyniku wybierana jest jedno z alternatywnych przejść. Bloku można użyć również w przypadku, gdy zbiegają się alternatywne wątki.
-  **Przejście**, z zasady nie opisywane, ponieważ z reguły oznacza zakończenie aktywności; może być opatrzone warunkiem, może też być oznaczone symbolem iteracji; akcje opisujące przejścia powinny być raczej dołączone do którejś z aktywności.
-  **Sztabka synchronizująca** (ang. synchronization bar); może być typu **“fork”** (rozdzielenie jednej operacji na kilka przebiegających równolegle) lub typu **“join”** (złączenie kilku operacji równoległych w jedną).
-  **Stan początkowy**
-  **Stan końcowy**



Rys. 4 Przykładowy diagram aktywności

ZADANIA:

Zadanie 1

Zadanie 1: Burza mózgów – grupa (czas 45 min.)

Na podstawie specyfikacji znajdującej dostarczonej przez prowadzącego, w 4 – 5 osobowych grupach należy określić potencjalnych aktorów oraz przypadki użycia dla systemu.

Utwórz słowniczek projektu wykorzystując szablon **słowniczek.dot**.

Utwórz dokument specyfikacji uzupełniający wykorzystując szablon **specyfikacja_uzupelniajaca.dot**.

Zadanie 2: Tworzenie modelu Przypadków Użycia w programie Rational Rose – indywidualnie (czas 45 min.)

Zapoznaj się z własnościami aplikacji Rational Rose.

Utwórz nowy model w programie Rational Rose. Nazwij go zgodnie z wybraną nazwą projektu.

Dodaj do Widoku Przypadków Użycia (ang. *Use Case View*) przypadki użycia zdefiniowane w ćwiczeniu 1.

Dodaj do Widoku Przypadków Użycia (ang. *Use Case View*) aktorów zdefiniowanych w ćwiczeniu 1.

Zadanie 3. Tworzenie specyfikacji przypadków użycia – indywidualnie (czas 60 min.)

Dla każdego przypadku użycia utwórz dokument specyfikacji przypadku użycia wykorzystując szablon **specyfikacja_przyp_uzyc.dot**. We właściwościach dokumentu wpisz swoje nazwisko jako autora specyfikacji. W historii wersji wpisz swoje nazwisko jako autora wersji 1.0.

Podłącz specyfikację do odpowiedniego przypadku użycia w modelu zbudowanym za pomocą Rational Rose.

***Zadanie 4: Tworzenie diagramu przypadków użycia – indywidualnie
(czas 15 min.)***

Narysuj diagram przypadków użycia. Umieść na nim przypadki użycia oraz aktorów. Odpowiednio zdefiniuj powiązania pomiędzy aktorami a przypadkami użycia.

***Zadanie 5: Graficzne modelowanie przepływu zdarzeń dla
przypadku użycia – indywidualnie (czas 60 min.)***

Dla każdego przypadku użycia zdefiniuj diagram aktywności, który będzie graficznie opisywał przepływ sterowania dla przypadku użycia.