

Inżynieria Oprogramowania

Laboratoria nr 9–10

wersja 1.0

Temat: Wykonanie obiektowego modelu zachowania systemu w postaci diagramów sekwencji.

Zadanie 1. (Diagramy sekwencji dla operacji realizacji przypadków użycia)

Polecenie: Należy uzupełnić wykonany w poprzednim etapie zajęć obiektowy model struktury oprogramowania systemu informatycznego i jego implementację dla wcześniej wybranych dwóch przypadków użycia, przez wykonanie modelu zachowania systemu w postaci diagramów sekwencji i jego implementację dla realizacji tych przypadków użycia.

Należy wykonać diagramy sekwencji modelujące operacje realizacji dwóch przypadków użycia i operacje w nich zawarte (rekurencyjnie):

- diagram dla przebiegu operacji rozpoczynającej realizację 1. przypadku użycia;
- diagram dla przebiegu operacji rozpoczynającej realizację 2. przypadku użycia;
- diagramy dla przebiegów operacji użytych na tych i innych diagramach sekwencji, jeśli to konieczne.

Każda z tych operacji powinna mieć swój własny diagram sekwencji, tylko dla niej przeznaczony.

Operacje zawarte w diagramach sekwencji powinny być w odpowiednich klasach na diagramach klas.

Objaśnienie: Należy ograniczyć się do pokazania przepływu sterowania między klasami i obiektami (instancjami klas), jaki jest konieczny do wykonania tej operacji, dla której przeznaczony jest diagram sekwencji. Inne rzeczy należy pominąć na diagramie, ale zawrzeć w implementacji (w kodzie).

Jeśli diagram sekwencji przeznaczony jest dla operacji A(), to:

- operację A() modeluje synchroniczna wiadomość znaleziona;
- zakończenie operacji A() modeluje synchroniczna wiadomość zgubiona, jeśli to konieczne;
- wszystkie inne operacje pokazane na tym diagramie są częścią przebiegu operacji A(), czyli są wykonywane bezpośrednio lub pośrednio w niej.

Gdy na diagramie sekwencji przebieg operacji A() zawiera wykonanie operacji B(), czyli gdy z prostokąta aktywacji operacji A() wychodzi wiadomość operacji B(), to przebieg operacji B() może być pokazany na osobnym diagramie sekwencji, jeśli operacja B() jest złożona (więcej niż 1 para wiadomości: synchroniczna + zwrotna) i:

- operacja B() wykonywana jest też w innym miejscu tego samego lub innego diagramu sekwencji;
- lub operacja B() nie mieści się na tym diagramie;
- lub operacja B() realizuje inny przypadek użycia;
- lub operacja B() jest rekurencyjnym wykonaniem operacji A().

Na diagramie należy m.in.:

- umieścić linie życia dla klas lub obiektów (instancji klas), które są inicjatorami lub wykonawcami operacji składających się na przebieg operacji, dla której diagram powstaje.
- wywołanie danej operacji pokazać przez odpowiednią wiadomość z nazwą tej operacji i jej parametrami, jeśli są używane;
- zakończenie danej operacji pokazać przez wiadomość zwrotną z nazwą kończącej operacji i wartością zwracaną przez tę operację, jeśli coś zwraca;
- zastosować połączone fragmenty (ramki) z odpowiednimi operatorami w celu zamodelowania m.in.:
 - opcjonalności wykonania operacji (*opt*);
 - alternatywności wykonania operacji (*alt*);
 - współbieżności lub dowolnej kolejności wykonania operacji (*par*, *seq*);
 - wielokrotności wykonania operacji (*loop*).
- utworzyć sekwencje operacji odpowiednie do użytych wzorców projektowych.

Zadanie 2. (Implementacja zachowania oprogramowania)

Polecenie: Na podstawie wykonanych diagramów klas, jeśli uległy zmianie, należy wykonać kod klas: definicje klas, ich atrybutów, operacji itd.

Na podstawie wykonanych diagramów sekwencji należy ręcznie napisać kod operacji na nich pokazanych.

Objaśnienie: Kod klas należy wygenerować automatycznie przy pomocy programu *Visual Paradigm* i umieścić go w projekcie programistycznym utworzonym w wybranym narzędziu IDE.

Następnie należy ręcznie poprawić i uzupełnić ten kod (m.in. o kod modelowanych operacji), aby możliwe było uruchomienie tworzono systemu informatycznego i wywołanie operacji realizacji wybranych przypadków użycia.

Kod operacji powinien być zgodny z jej diagramem sekwencji oraz, jeśli trzeba, uzupełniony o dotkowe rzeczy (w tym operacje), których brak na diagramie, ale są konieczne do prawidłowego działania oprogramowania.

Opracowano na podstawie [instrukcji 6](#) i [instrukcji 7](#) „Laboratorium z przedmiotu: Inżynieria Oprogramowania W04ITE-S10011G”, autorstwa dr inż. Zofii Kruczkiewicz.

Te instrukcje zawierają też przykłady iteracyjnego wykonania podobnych zadań i tutoriale do programu Visual Paradigm.