

Stanisław Wrycza,
Bartosz Marcinkowski, Jacek Maślankowski

UML

2.x

ĆWICZENIA ZAAWANSOWANE

Dowiedz się, jak użyteczny jest język UML!

- Poznaj najnowszy standard języka UML
- Naucz się w pełni wykorzystywać jego możliwości
- Poszerz swoją wiedzę o zagadnienia zaawansowane i profile
- Zostań ekspertem dzięki praktycznym przykładom

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiejkolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Recenzent: dr Piotr Soja, Uniwersytet Ekonomiczny w Krakowie

Redaktor prowadzący: Michał Mrowiec

Wydawnictwo HELION

ul. Kościuszki 1c, 44-100 GLIWICE

tel. 32 231 22 19, 32 230 98 63

e-mail: helion@helion.pl

WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

http://helion.pl/user/opinie?cwu212_ebook

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-246-4366-0

Copyright © Helion 2012

Printed in Poland.

- [Poleć książkę na Facebook.com](#)
- [Kup w wersji papierowej](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

Spis treści

Wstęp	5
Rozdział 1. Zmiany, udoskonalenia i profile języka UML	9
1.1. Ewolucja ujednoliconego języka modelowania	9
1.2. Struktura języka UML	10
1.3. Profile języka UML	18
1.4. SysML jako praktyczna implementacja profili UML	21
1.5. Struktura języka SysML	21
1.6. UML 2.x — najistotniejsze modyfikacje standardu	26
Rozdział 2. Diagram wymagań systemowych	37
Rozdział 3. Diagramy przypadków użycia	59
Rozdział 4. Diagramy klas oraz definiowania bloków	73
Rozdział 5. Diagram bloków wewnętrznych oraz diagramy wdrożeniowe	89
Rozdział 6. Diagramy parametryczne	107
Rozdział 7. Diagramy czynności	123
Rozdział 8. Diagramy sekwencji	145
Rozdział 9. Diagramy maszyn stanowych	163
Rozdział 10. Diagramy pakietów	179
Rozdział 11. Generowanie kodu źródłowego	193
Bibliografia	203

Wstęp

Modelowanie systemów informatycznych przy użyciu języka UML znalazło **szeroką gamę aplikacji** w biznesie i jest przedmiotem nauczania na kierunkach informatycznych uczelni na całym świecie. Korzyści z użytkowania języka UML zainspirowały inne środowiska zawodowe do opracowania własnych standardów przyjmujących postać profili tego języka. Z punktu widzenia analityków, projektantów czy inżynierów systemów informatycznych znaczącym rozwiązaniem o charakterze profilu jest zwłaszcza język SysML, który wykorzystując architekturę języka UML, służy aplikacjom o ukierunkowaniu technicznym.

Od niemal dekady w sposób konsekwentny Katedra Informatyki Ekonomicznej Uniwersytetu Gdańskiego zajmuje się zagadnieniami metodologicznymi, aplikacyjnymi i dydaktycznymi związanymi z **językiem UML i jego profilami**, zwłaszcza SysML. Jest to niezbędne ze względu na nieustanny rozwój tych języków, powstawanie kolejnych ich wersji i nowych profili będących rezultatem prac stałych grup roboczych przede wszystkim w ramach OMG — Object Management Group oraz innych zainteresowanych organizacji, np. INCOSE — International Council on Systems Engineering.

W wyniku studiów, prac własnych oraz monitorowania rozwoju języków UML i SysML autorzy zdecydowali się na wydanie kolejnej książki, której **cele** są następujące:

- ❑ udokumentowanie i skomentowanie w rozdziale pierwszym zmian, które wprowadzono w standardach UML i SysML w ostatnich latach, począwszy od wersji UML 2.0 i SysML 1.0, aż do wersji UML 2.4 (2011) oraz SysML 1.2 (2010);
- ❑ opracowanie w rozdziałach 2. – 10. zestawu przykładów i zadań dotyczących praktycznego zastosowania kilkunastu najbardziej znaczących rodzajów diagramów, stosownie do najnowszych wersji języka UML i jego profilu — języka SysML;
- ❑ zaprezentowanie zasad transformacji diagramów języka UML na kod źródłowy aplikacji informatycznych w rozdziale 11.

Każdy z rozdziałów o charakterze ćwiczeniowym (2. – 10.) cechuje się **jednolitą strukturą**. Składa się na nią każdorazowo 7 przykładów, a więc problemów z podaniem ich rozwiązań, oraz 7 zadań do rozwiązania przez odbiorców podręcznika — studentów kierunków informatycznych lub uczestników kursów szkoleniowych. Materiał zawarty w tej książce może bowiem również posłużyć menedżerom, analitykom i projektantom systemów informatycznych, którzy mogą dzięki niej opanować najnowsze rozwiązania, metody i techniki z zakresu modelowania systemów informatycznych. Przykłady i zadania w rozdziałach 2. – 10. dotyczą następujących **rodzajów diagramów**:

- ❑ wymagań systemowych (SysML);
- ❑ przypadków użycia (UML);
- ❑ klas (UML) oraz definiowania bloków (SysML);
- ❑ bloków wewnętrznych (SysML) oraz wdrożeniowych (UML), tj. komponentów i rozlokowania;
- ❑ parametrycznych (SysML);
- ❑ czynności (UML);
- ❑ sekwencji (UML);
- ❑ maszyn stanowych (UML);
- ❑ pakietów (UML).

Powyższy zestaw diagramów został starannie wyselekcjonowany po dogłębnej analizie istotności oraz częstości ich stosowania przez praktyków modelowania systemów informatycznych.

Niniejsza książka **uzupełnia merytoryczny zasób treści** prezentowanych w dotychczas firmowanych przez Katedrę Informatyki Ekonomicznej UG pozycjach o UML i SysML, do których należą:

- ❑ Wrycza S., Marcinkowski B., Wyrzykowski K. (2005); *Język UML 2.0 w modelowaniu systemów informatycznych*, Helion, Gliwice;
- ❑ Wrycza S. (2007, red.); *UML 2.1. Ćwiczenia*, Helion, Gliwice;
- ❑ Wrycza S., Marcinkowski B., Wyrzykowski K. (2007); *Tablice informatyczne. UML*, Helion, Gliwice;
- ❑ Wrycza S., Marcinkowski B. (2010); *Język inżynierii systemów SysML. Architektura i zastosowania*, Helion, Gliwice.

Publikacje te stanowią logicznie i chronologicznie uporządkowany ciąg merytorycznej wiedzy na temat UML i jego profilu SysML. Traktowane i studiowane łącznie stanowią obszerny i spójny zasób wiedzy na temat języków UML w wersjach 1.0 – 2.4 oraz SysML (1.0 – 1.2). Zamierzeniem autorów jest **zachowanie aktualności wiedzy** na temat standardu UML i pochodnych poprzez ujmowanie metod, technik i kategorii modelowania wprowadzanych przez OMG i INCOSE w kolejnych wersjach opisywanych standardów. Zakres tematyczny tego zestawu logicznie

i merytorycznie powiązanych książek w odniesieniu do aktualnie obowiązujących rodzajów diagramów UML oraz jego profilu — języka SysML — zaprezentowano w tabeli W.1.

Tabela W.1. Zakres tematyczny książek na temat UML i SysML autorstwa pracowników KIE UG

	Język UML 2.0 w modelowaniu systemów infor- matycznych (2005)	UML 2.1. Ćwi- czenia (2007)	Tablice in- formatyczne. UML (2007)	Język inżynierii systemów SysML. Archi- tektura i zasto- sowania (2010)	UML. Ćwi- czenia za- awansowa- ne (2011)
Diagramy wymagań systemowych				✓	✓
Diagramy przypadków użycia	✓	✓	✓	✓	✓
Diagramy klas	✓	✓	✓		✓
Diagramy obiektów	✓	✓	✓		
Diagramy definiowania bloków				✓	✓
Diagramy bloków wewnętrznych				✓	✓
Diagramy komponentów	✓	✓	✓		✓
Diagramy rozlokowania	✓	✓	✓		✓
Diagramy parametryczne				✓	✓
Diagramy czynności	✓	✓	✓	✓	✓
Diagramy sekwencji	✓	✓	✓	✓	✓
Diagramy komunikacji	✓	✓	✓		
Diagramy harmonogramowania	✓	✓	✓		
Diagramy sterowania interakcją	✓	✓	✓		
Diagramy maszyny stanowej	✓	✓	✓	✓	✓
Diagramy struktur połączonych	✓	✓	✓		
Diagramy pakietów	✓	✓	✓	✓	✓
Diagramy profili					✓
Metodyka RUP	✓				
Modelowanie biznesowe	✓		✓		
Modelowanie analityczne	✓		✓		
Narzędzia CASE	✓	✓			
Implementacja		✓			✓

Niniejsza książka cechuje się **innowacyjnymi przykładami i zadaniami**, głównie z zakresu aplikacji dla gospodarki elektronicznej. Poruszono w niej takie zagadnienia jak: planowanie akcji marketingowej z wykorzystaniem systemu CRM, monitorowanie i sterowanie ruchem pojazdów w mieście, serwis internetowy obsługi konferencji międzynarodowej, system rezerwacji miejsc hotelowych, portal firmy zajmującej się badaniami sondażowymi, system rozliczania świadczeń medycznych, serwis przechowywania zasobów w środowisku rozproszonym, obsługa sklepu internetowego, obieg dokumentów z podpisem elektronicznym, głosowy system obsługi magazynów, aukcje internetowe, system elektronicznej rezerwacji zakupu biletów, serwis internetowy firmy kurierskiej oraz wiele innych. Wykraczają one znacząco poza standard systemów transakcyjnych w firmach i instytucjach najczęściej ilustrujących język UML w dostępnych na rynku książkach krajowych i zagranicznych.



Zmiany, udoskonalenia i profile języka UML

1.1. Ewolucja ujednoliconego języka modelowania

Język modelowania systemów informatycznych UML stał się nieodłącznym narzędziem tworzenia większości realizowanych współcześnie projektów informatycznych. Jednocześnie architektura tego języka stanowi poważne wyzwanie dla twórców systemów informatycznych. Wykorzystanie języka i jego profili w kolejnych zróżnicowanych obszarach zastosowań biznesowych i innych ma daleko idący wpływ na licznosc kategorii modelowania zawartych w tym języku oraz praktykę ich zastosowania. Zdefiniowane w poszczególnych wersjach standardu UML diagramy i ich elementy składowe podlegają więc **ustawicznym zmianom, udoskonaleniom i rozszerzeniom**.

Rozwój specyfikacji języka stał się na przestrzeni lat jednocześnie przedmiotem kontrowersji i dyskusji — liczni autorzy tematycznych publikacji i praktycy niejednokrotnie zgłaszali krytyczne uwagi o nadmiernym stopniu rozbudowania samej architektury języka. Istotnie, założenia dotyczące uniwersalności języka UML przyczyniały się do **wzrostu zakresu i złożoności standardu**. Tymczasem w większości praktycznych zastosowań pojawiała się potrzeba wykorzystania jedynie pewnego konsekwentnie powtarzającego się podzbioru jego kategorii modelowania. Tym samym problem złożoności standardu próbowano rozwiązać poprzez zgłaszanie propozycji wprowadzenia tzw. lekkich (ang. *light*) wersji języka.

Jednocześnie pojawiała się nieustanna presja na uzupełnianie standardu o pojęcia specyficzne dla specjalistycznych obszarów zastosowań. Doświadczenia w zakresie rozwoju języka UML i rosnąca liczba jego zastosowań gospodarczych czy technicznych stały u podstaw uwzględnienia w standardzie **mechanizmów profilowania**, tj. możliwości opracowania pochodnych języka, które uwzględniają specyfikę i wyzwania zróżnicowanych dziedzin zastosowań.

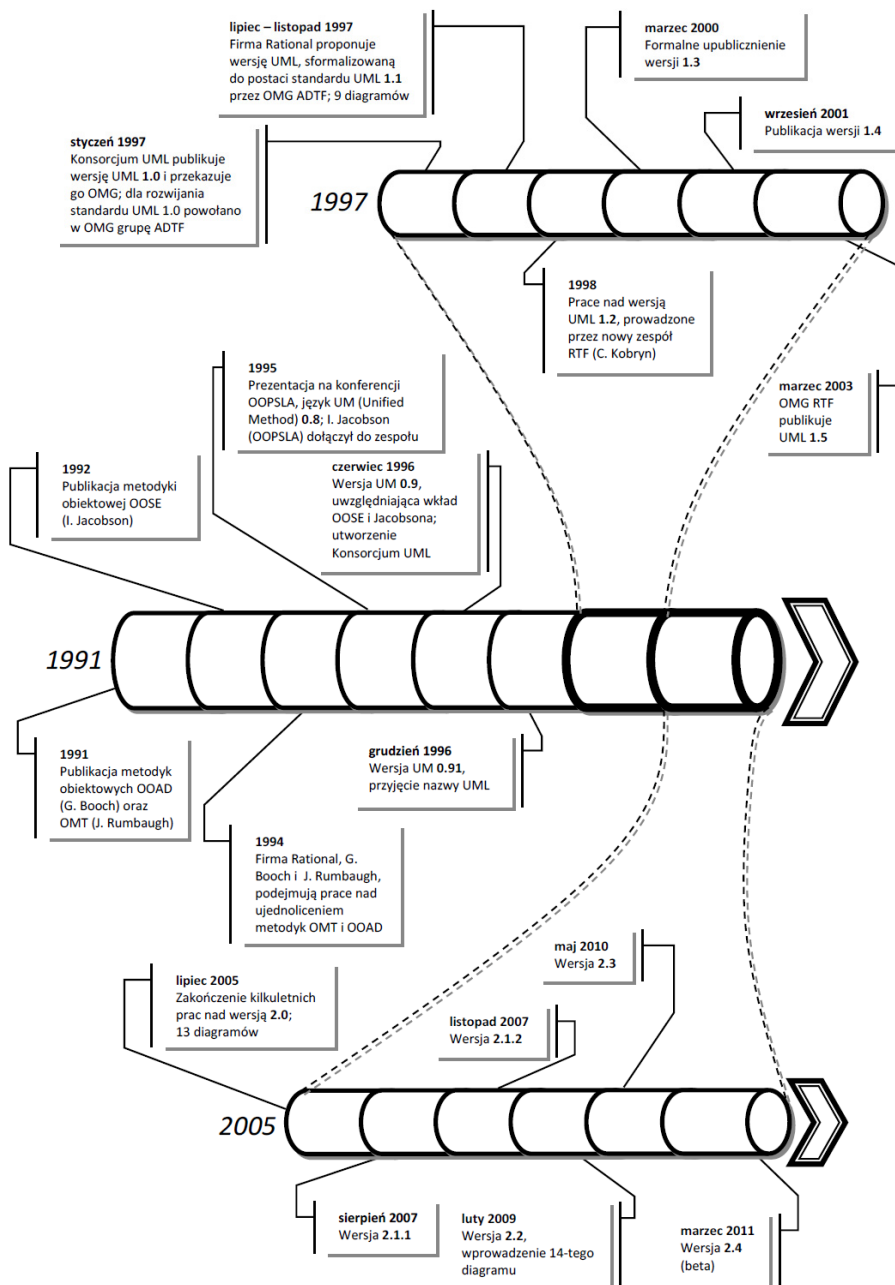
Niezależnie od zgłaszania propozycji profili na przestrzeni ostatnich dwóch dekad język UML podlegał daleko idącym przekształceniom i zmianom o charakterze ewolucyjnym. Najbardziej istotne fakty tej ewolucji, poszczególne fazy i **kamienie milowe** w formułowaniu i udoskonalaniu standardu przedstawiono w ujęciu chronologicznym na rysunku 1.1.

Bardziej istotne rewizje i modyfikacje standardu UML, zasygnalizowane na rysunku 1.1, wymagają syntetycznego komentarza. I tak, charakterystyce poszczególnych zdarzeń, roli instytucji oraz grup roboczych zaangażowanych w proces zmian i udoskonaleń języka UML poświęcono tabelę 1.1. Wskazano w niej również publikacje źródłowe oraz hiperłącza do dokumentacji zawierających opis i specyfikację poszczególnych wersji standardu.

1.2. Struktura języka UML

Język UML w każdej ze swoich wersji składał się z określonej, **sukcesywnie rosnącej liczby rodzajów diagramów** składowych. Najbardziej utrwaliła się struktura trzynastu diagramów języka UML 2.0 z 2005 roku. Od tego czasu wraz z wersją 2.2 wprowadzono jeszcze jeden rodzaj diagramu UML — diagram profili. Tak więc, w momencie wydawania niniejszej książki standard uwzględnia 7 rodzajów diagramów struktury i tyle samo rodzajów diagramów dynamiki. Listę diagramów UML wraz z ich podziałem przedstawiono na rysunku 1.2.

Przedstawione na rysunku 1.2 diagramy języka UML 2.4 **zdefiniowano** w tabeli 1.2 z uwzględnieniem aktualnie używanych w najpopularniejszych narzędziach CASE wyróżników poszczególnych diagramów. Należy zwrócić uwagę, że równolegle funkcjonują formy pełne i skrócone wyróżników. Jednocześnie aktualna dokumentacja języka UML nie różnicuje wyróżników poszczególnych rodzajów diagramów interakcji.



Rysunek 1.1. Ewolucja języka UML

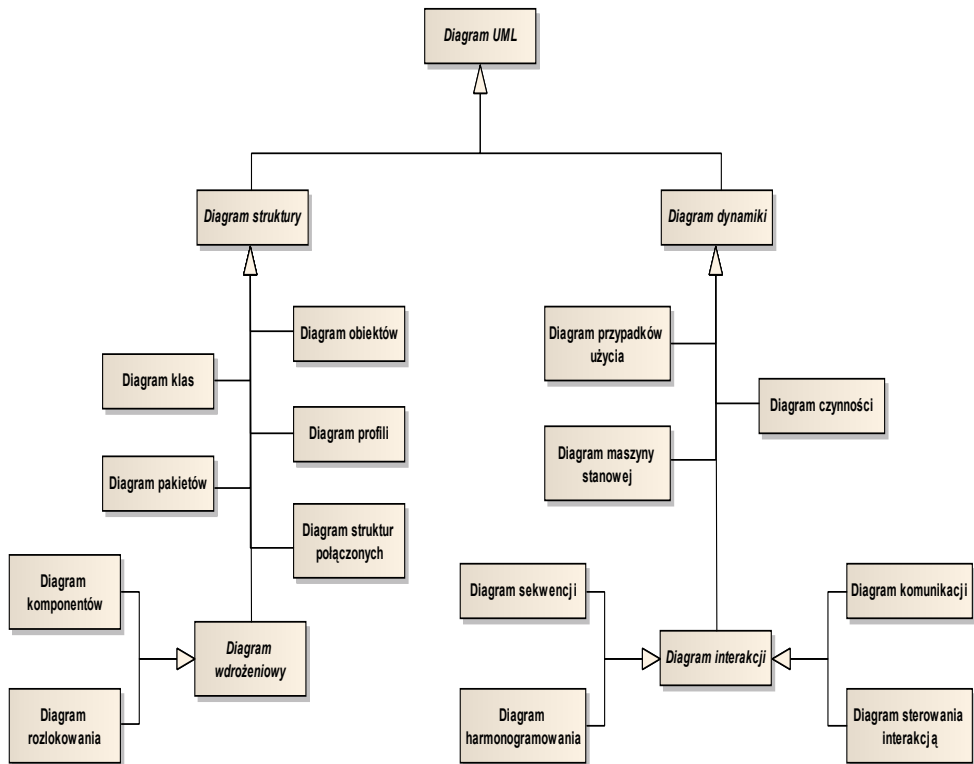
Tabela 1.1. Charakterystyka modyfikacji standardu UML

Wersja	Główne zmiany	Publikacja
1.0 styczeń 1997	Propozycja specyfikacji UML została przekazana przez konsorcjum czołowych korporacji informatycznych zapewniających wsparcie projektowi rozwoju UML grupie roboczej Analysis and Design Task Force (ADTF). Grupa ta powołana została w ramach organizacji OMG na potrzeby rozwijania standardu UML.	Booch G., Rumbaugh J., Jacobson I. (1997); <i>Unified Modeling Language, Version 1.0</i> , Rational Software Corporation
1.1 listopad 1997	Wersja 1.1 jest adaptacją wersji 1.0 UML wykonaną przez OMG.	Publikacje wewnętrzne OMG
1.2 czerwiec 1998	Zmiana standardu UML przez nowy zespół RTF — Revision Task Force, który opracowywał i zatwierdzał zmiany w kolejnych wersjach od 1.3 do 1.5 we współpracy z partnerami OMG; zespół ten kierowany był przez Crisa Kobryna.	
1.3 marzec 2000	Niewielkie zmiany w metamodelu, semantyce i notacji UML.	http://www.omg.org/spec/UML/1.3
1.4 wrzesień 2001	Dodano możliwość organizowania rozszerzeń w ściśle zdefiniowane profile, zaktualizowano symbole widoczności, wprowadzono możliwość asynchronicznego wywoływania komunikatów na diagramach interakcji, udostępniono możliwość dodawania wielu stereotypów do klasyfikatorów, dodano artefakty w celu prezentowania fizycznego odwzorowania komponentów.	http://www.omg.org/spec/UML/1.4
1.5 marzec 2003	Zintegrowano wersję 1.4 z semantyką akcji, zdefiniowano koncepcję przepływu danych w celu przenoszenia danych pomiędzy akcjami. UML 1.5 składa się z 9 oficjalnie zdefiniowanych rodzajów diagramów (4 diagramy struktury i 5 dynamiki) oraz pakietów.	http://www.omg.org/spec/UML/1.5
1.4.2 lipiec 2004	Oficjalnie zaakceptowano jako specyfikację ISO — standard ISO/IEC 19501 (oczekiwano dwa lata na pozytywną decyzję akceptacji standardu).	http://www.omg.org/spec/UML/#ISO19501

Tabela 1.1. Charakterystyka modyfikacji standardu UML — ciąg dalszy

Wersja	Główne zmiany	Publikacja
2.0 lipiec 2005	Formalnie zdefiniowano diagramy pakietów. Poszerzono grupę diagramów interakcji o dwa nowe — diagramy harmonogramowania i diagramy sterowania interakcją. Na bazie diagramów współdziałania wymienianych w wersjach 1.x standardu UML wyodrębniono diagramy komunikacji i diagramy struktur połączonych. W istniejących rodzajach diagramów wprowadzono liczną grupę nowych kategorii modelowania — w szczególności dotyczy to diagramów czynności i diagramów sekwencji.	http://www.omg.org/spec/UML/2.0
2.1.1 sierpień 2007	Wprowadzono drobne usprawniające modyfikacje i poprawki błędów. Najbardziej istotną zmianą było wprowadzenie nowej kategorii modelowania związanej z asocjacją — przynależności.	http://www.omg.org/spec/UML/2.1.1
2.1.2 listopad 2007	Mniejszej rangi zmiany edytorskie prowadzące do poprawy spójności obszernej dokumentacji.	http://www.omg.org/spec/UML/2.1.2
2.2 luty 2009	Wprowadzono istotną zmianę poprzez dodanie nowego, czternastego rodzaju diagramów UML — diagramów profili; diagramy te sklasyfikowano jako diagramy struktury. Skorygowano sygnalizowane wcześniej problemy ze spójnością — zaktualizowano w tym celu m.in. składnię operacji.	http://www.omg.org/spec/UML/2.2
2.3 maj 2010	Doprecyzowano pojęcia i właściwości klas asocjacyjnych. Dodano właściwość <code>final</code> , która pozwala na wyłączenie możliwości dziedziczenia klas, a tym samym blokuje możliwość ich redefinicji. Zaktualizowano diagramy komponentów i struktur połączonych. Zaproponowano konwencję przejść lokalnych w diagram maszyn stanowych.	http://www.omg.org/spec/UML/2.3
2.4 — Beta 2 marzec 2011	Zaktualizowano klasy i pakiety poprzez dodanie atrybutu pakietu URI (<i>Unified Resource Identifier</i>), który pozwala na jednoznaczny identyfikację pakietu poprzez wskazanie jego lokalizacji. Wprowadzono nowe typy akcji. W związku z wprowadzeniem URI usunięto niektóre zdarzenia (m.in. tworzenia, niszczenia czy wykonywania w diagramach sekwencji) lub zmieniono ich nazwy. Zmodyfikowano wybrane stereotypy.	http://www.omg.org/spec/UML/2.4

Źródło: opracowanie własne na podstawie (OMG, 2011a)



Rysunek 1.2. Diagramy UML 2.4

Źródło: opracowanie własne na podstawie (OMG, 2011a)

Tabela 1.2. Rodzaje diagramów języka UML

	Lp.	Rodzaj diagramu	Definicja	Wyróżnik rodzaju diagramu	
				Forma pełna	Forma skrócona
Diagramy struktury	1	Diagram klas (ang. <i>class diagram</i>)	Graficzne przedstawienie statycznych, deklaratywnych elementów dziedziny przedmiotowej oraz związków między nimi	class	–
	2	Diagram obiektów (ang. <i>object diagram</i>)	Wystąpienie diagramu klas odwzorowujące strukturę systemu w wybranym momencie jego działania	object	–
	3	Diagram pakietów (ang. <i>package diagram</i>)	Graficzne przedstawienie logicznej struktury systemu w postaci zestawu pakietów połączonych zależnościami i zagnieżdżeniami	package	pkg
	4	Diagram struktur połączonych (ang. <i>composite structure diagram</i>)	Graficzne przedstawienie wzajemnie współdziałających części dla osiągnięcia pożądanej funkcjonalności współdziałania	composite structure	–
	5	Diagram profili (ang. <i>profile diagram</i>)	Graficzne przedstawienie implementacji mechanizmów rozszerzalności ukierunkowanej na adaptację i kustomizację określonych kategorii modelowania metamodelu języka UML do różnych platform i specjalistycznych dziedzin zastosowania	profile	–

Tabela 1.2. Rodzaje diagramów języka UML — ciąg dalszy

	Lp.	Rodzaj diagramu	Definicja	Wyróżnik rodzaju diagramu	
				Forma pełna	Forma skrócona
Diagramy wdrożeniowe	6	Diagram rozlokowania (ang. <i>deployment diagram</i>)	Rodzaj diagramu wdrożeniowego, który przedstawia sieć połączonych ścieżkami komunikowania węzłów z ulokowanymi na nich artefaktami	deployment	dep
	7	Diagram komponentów (ang. <i>component diagram</i>)	Rodzaj diagramu wdrożeniowego, który wskazuje organizację komponentów i zależności między nimi	component	cmp
Diagramy dynamiki	8	Diagram przypadków użycia (ang. <i>use case diagram</i>)	Graficzne przedstawienie przypadków użycia, aktorów oraz związków między nimi występujących w danej dziedzinie przedmiotowej	use case	uc
	9	Diagram czynności (ang. <i>activity diagram</i>)	Graficzne przedstawienie sekwencyjnych i (lub) współbieżnych przepływów sterowania oraz danych pomiędzy uporządkowanymi ciągami czynności, akcji i obiektów	activity	act
	10	Diagram maszyny stanowej (ang. <i>state machine diagram</i>)	Graficzne odzwierciedlenie dyskretnego skokowego zachowania skończonych systemów stan-przejsie	state machine	stm

Tabela 1.2. Rodzaje diagramów języka UML — ciąg dalszy

Lp.	Rodzaj diagramu	Definicja	Wyróżnik rodzaju diagramu	
			Forma pełna	Forma skrócona
Diagramy interakcji	11	Diagram sekwencji (ang. <i>sequence diagram</i>)	Rodzaj diagramu interakcji opisujący interakcje pomiędzy instancjami klasyfikatorów systemu w postaci sekwencji komunikatów wymienianych między nimi	interaction sd
	12	Diagram komunikacji (ang. <i>communication diagram</i>)	Rodzaj diagramu interakcji specyfikujący strukturalne związki pomiędzy instancjami klasyfikatorów biorącymi udział w interakcji oraz wymianę komunikatów pomiędzy tymi instancjami	interaction sd
	13	Diagram harmonogramowania (ang. <i>timing diagram</i>)	Rodzaj diagramu interakcji reprezentujący na osi czasu zmiany dopuszczalnych stanów, jakie może przyjmować instancja klasyfikatora uczestnicząca w interakcji	interaction sd
	14	Diagram sterowania interakcją (ang. <i>interaction overview diagram</i>)	Rodzaj diagramu interakcji dokumentujący przepływ sterowania pomiędzy logicznie powiązanymi diagramami i fragmentami interakcji z wykorzystaniem kategorii modelowania diagramów czynności	interaction sd

Źródło: opracowanie własne na podstawie (Wrycza, Marcinkowski i Wyrzykowski, 2005)

1.3. Profile języka UML

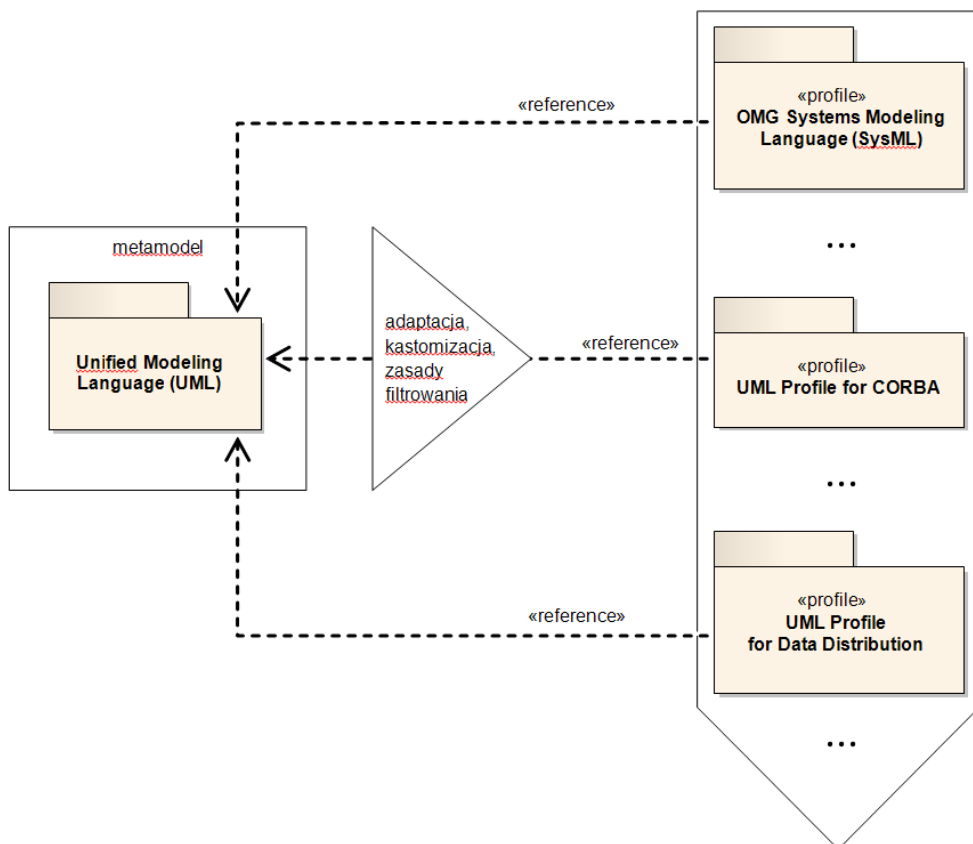
Od momentu swego zaistnienia język UML użytkowany był poza analizą i projektowaniem systemów informatycznych w **silnie zróżnicowanych obszarach zastosowań** z wielu dziedzin gospodarczych, społecznych czy technicznych. Ich specyfika powodowała zainteresowanie twórców i użytkowników systemów wprowadzonymi uprzednio do języka UML mechanizmami rozszerzalności. Mechanizmy te pozwalają na spontaniczne opracowywanie autorskich modyfikacji oryginalnego standardu UML. Jednak z czasem projektanci i inżynierowie systemów dochodzili do wniosku, że na bazie tych specyficznych rozszerzeń można opracować swego rodzaju sformalizowane dialekty języka UML znacznie lepiej dopasowane do specyfiki dziedziny przedmiotowej niż UML w oryginalnym kształcie (por. rysunek 1.3). Wykorzystywały one w znacznym stopniu modele i diagramy źródłowego standardu UML, ale wzbogacały je o szereg całkowicie nowych, oryginalnych kategorii modelowania. Te nowatorskie adaptacje, rozszerzenia języka UML ukierunkowane na specyficzną dziedzinę czy platformę określa się mianem profili. I tak, **profil UML** (OMG, 2011b):

- ❑ stanowi celowy, właściwy dla danego obszaru zastosowań podzbiór oryginalnego języka UML, tj. wykorzystuje wybrane diagramy oraz kategorie modelowania standardu UML według racjonalnych zasad ich filtrowania — adaptacji i kustomizacji do docelowego profilu;
- ❑ wprowadza nowe pojęcia i kategorie specyficzne dla danego profilu, a tym samym będące poza zakresem samego standardu UML;
- ❑ wprowadza ściśle zdefiniowane reguły wykorzystania standardowych bądź dołączanych kategorii modelowania;
- ❑ uzupełnia semantykę kategorii modelowania wyrażoną w języku naturalnym;
- ❑ porządkuje i strukturyzuje architekturę języka stanowiącego dany profil.

Zasady filtrowania kategorii modelowania z aktualnego kanonu języka UML, ich adaptacji i kustomizacji do tematycznego profilu przedstawia rysunek 1.3.

Aktualnie organizacja OMG sprawuje opiekę i nadzór merytoryczny nad następującymi profilami języka UML:

- ❑ **OMG Systems Modeling Language (SysML)**, dedykowanym modelowaniu głównie systemów technicznych, inżynierskich na podstawie inżynierii systemów;
- ❑ **PIM and PSM for Smart Antenna**, przeznaczonym do modelowania inteligentnych systemów antenowych;



Rysunek 1.3. Mechanizm tworzenia tematycznych profili językowych

- ❑ **Service Oriented Architecture Modeling Language (SoaML®)**, służącym projektowaniu i specyfikowaniu usług w ramach architektury usługowej SOA;
- ❑ **UML Profile for CORBA®**, umożliwiającym wyrażanie semantyki interfejsów obiektów CORBA z wykorzystaniem języka UML;
- ❑ **UML Profile for CORBA® Component Model (CCM)**, pozwalającym na wyrażanie w języku UML składowych modelu komponentów użytkowanego w trakcie tworzenia i wykorzystywania aplikacji CORBA;
- ❑ **UML Profile for CORBA® & CORBA® Component Model (CCCMP)**, wspomagającym użytkowanie metodyki Model Driven Architecture w odniesieniu do standardu CORBA;

- ❑ **UML Profile for Data Distribution**, dedykowanym analizie i projektowaniu obiektowych systemów wykorzystujących usługi rozproszenia danych;
- ❑ **UML Profile for Enterprise Application Integration (EAI)**, zapewniającym standard wymiany metadanych w celu dostępu do interfejsów aplikacji;
- ❑ **UML Profile for Enterprise Distributed Object Computing (EDOC)**, upraszczającym tworzenie systemów bazujących na komponentach w zgodności z metodykami Model Driven Architecture oraz Service Oriented Architecture;
- ❑ **UML Profile for Modeling and Analysis of Real-time and Embedded Systems (MARTE)**, rozszerzającym możliwości UML w zakresie tworzenia systemów czasu rzeczywistego i wbudowanych w oparciu o architekturę Model Driven Architecture;
- ❑ **UML Profile for Modeling QoS and Fault Tolerance Characteristics and Mechanisms**, umożliwiającym wizualizację koncepcji związanych z zapewnieniem jakości usług oraz odporności oprogramowania na błędy;
- ❑ **UML Profile for Schedulability, Performance and Time**, koncentrującym się na definiowaniu standardowych paradygmatów obliczania i prognozowania charakterystyk czasowych oraz wydajnościowych w systemach czasu rzeczywistego;
- ❑ **UML Profile for Software Radio**, służącym tworzeniu oprogramowania dla usług radiowych;
- ❑ **UML Profile for System on a Chip (SoCP)**, zapewniającym reprezentację modelu integrującego osadzone procesory, moduły pamięci i inne elementy sprzętowe architektury SoCP;
- ❑ **UML Profile for Voice**, pozwalającym na standaryzację modeli dialogu wykorzystywanych w aplikacjach głosowych;
- ❑ **UML Testing Profile**, definiującym język projektowania, wizualizacji, specyfikacji, analizy, konstruowania oraz dokumentowania artefaktów systemów testowania.

Wraz ze wzrostem liczby proponowanych profili twórcy standardu UML wypracowali mechanizm wizualizacji ich definicji z wykorzystaniem dedykowanych diagramów, tj. diagramów profili. Lista profili podlega nieustannej aktualizacji i uzupełnieniom. W tym kontekście podkreślić należy zwłaszcza szczególne znaczenie **języka SysML** dla informatyków z uwagi na jego rolę w specyfikowaniu wymagań systemów informatycznych oraz szeroko pojętym modelowaniu systemów o zastosowaniach technicznych.

1.4. SysML jako praktyczna implementacja profili UML

Profiłem UML, który znajduje szczególne zastosowanie w modelowaniu systemów informatycznych jest OMG Systems Modeling Language (SysML), szczegółowo omówiony w książce (Wrycza i Marcinkowski, 2010). **Wartość dodaną** tego profilu z punktu widzenia twórcy systemu informatycznego należy upatrywać w kilku wymiarach. Przede wszystkim SysML daje możliwość formalnego zdefiniowania wymagań zarówno funkcjonalnych, jak i pozafunkcjonalnych w postaci diagramu wymagań systemowych. UML mimo szerokiej gamy diagramów nie oferuje takiej możliwości. Tymczasem, jak potwierdzają liczne badania, znaczący udział błędów we wdrażanym oprogramowaniu wynika z niewłaściwie zrealizowanej fazy specyfikacji wymagań. Opisana korzyść nie jest jedyną korzyścią dla analityków i projektantów systemów wynikającą z uwzględnienia języka SysML w projekcie informatycznym. Wprowadza on bowiem również:

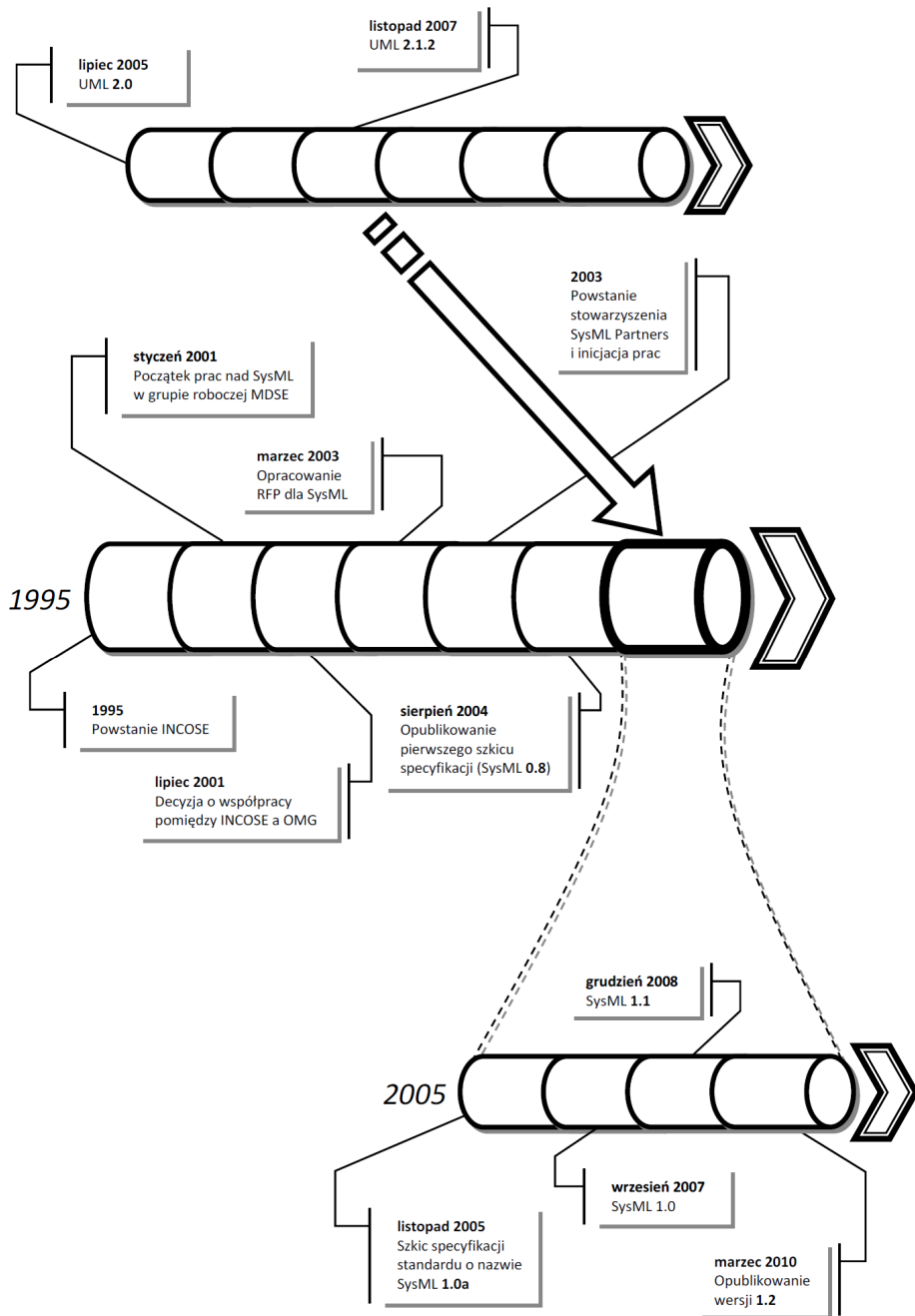
- ❑ **diagramy parametryczne** umożliwiające przeprowadzanie studiów nad różnymi wersjami produktu lub produktów pod kątem analizy i wyboru najlepszych wariantów (ang. *trade-off*), analizy wrażliwości i optymalizacji projektu na zasadzie symulacji;
- ❑ szereg **rozszerzeń do diagramów czynności**, ukierunkowanych na wspieranie specyficznej funkcjonalności systemów ciągłych i strumieniowych;
- ❑ alternatywne w stosunku do standardu UML możliwości opisanie struktury systemu w postaci **diagramów definiowania bloków** oraz **diagramów bloków wewnętrznych**.

Chronologiczny rozwój języka SysML zilustrowano na rysunku 1.4.

Zestawienie oficjalnych wersji dokumentacji SysML wraz z charakterystyką **merytorycznych zmian** w tych specyfikacjach ujmuje tabela 1.3.

1.5. Struktura języka SysML

Aktualna specyfikacja języka SysML ujmuje 9 rodzajów diagramów, w tym 4 diagramy struktury, 4 diagramy dynamiki oraz stanowiący samodzielną kategorię diagram wymagań systemowych. Oprócz skomentowanych w podrozdziale 1.4 istotnych nowości w stosunku do języka UML, tj. diagramów wymagań systemowych, diagramów parametrycznych, rozszerzeń w zakresie diagramów czynności oraz diagramów definiowania bloków i diagramów bloków wewnętrznych, profil ten



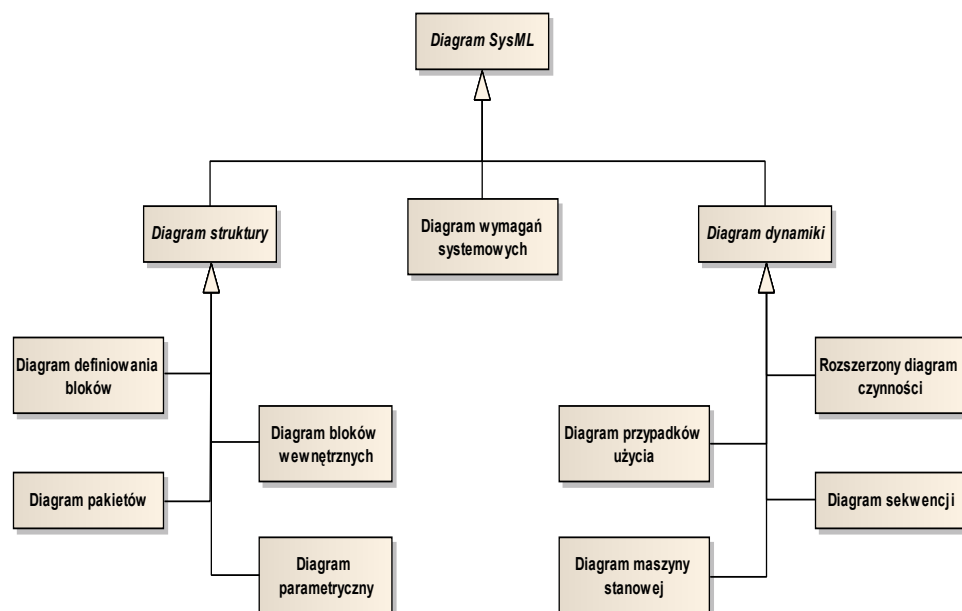
Rysunek 1.4. Ewolucja języka SysML

Tabela 1.3. Główne zmiany w poszczególnych wersjach języka SysML

Wersja	Główne zmiany	Publikacja
1.0 wrzesień 2007	Oficjalne opublikowanie standardu ujmującego uwagi i korekty edycyjne zgłoszone przez zespół SysML Finalization Task Force w marcu 2007 r.	http://www.omg.org/spec/SysML/1.0/
1.1 grudzień 2008	Większy nacisk na egzemplifikację wykorzystania kategorii modelowania. Wzbogacenie listy sekcji części bloku o sekcję wartości początkowych. Doprecyzowanie kwestii blokowej notacji czynności, uwzględnienie definicji miar i jednostek miar.	http://www.omg.org/spec/SysML/1.1/
1.2 czerwiec 2010	Synchronizacja dokumentacji z formalnie udostępnionym standardem UML 2.3. Możliwość specyfikowania instancji, czynności strukturyzowane i inne mniej znaczące modyfikacje.	http://www.omg.org/spec/SysML/1.2/

Źródło: opracowanie własne na podstawie (OMG, 2011c)

wykorzystuje również diagramy i kategorie modelowania wywodzące się bezpośrednio z języka UML. Są to diagramy przypadków użycia, diagramy maszyn stanowych, diagramy sekwencji oraz diagramy pakietów. **Kompletny zestaw diagramów języka SysML** w wersji 1.2 ilustruje rysunek 1.5.

**Rysunek 1.5.** Diagramy SysML 1.2

Definicje, podstawowe dyscypliny zastosowań oraz wyróżniki diagramów standardu SysML zawiera tabela 1.4. W tabeli wskazano ponadto bezpośrednie odpowiedniki poszczególnych diagramów SysML w oryginalnej dokumentacji UML.

Tabela 1.4. Rodzaje diagramów SysML

Rodzaj diagramu języka SysML 1.2	Definicja	Wyróżnik	Dyscyplina zastosowań	Odpowiednik w języku UML 2.4
Diagram wymagań systemowych	Graficzne przedstawienie wymagań systemowych i ich relacji z innymi kategoriami modelowania systemu	req	Inżynieria wymagań	–
Diagram przypadków użycia	Graficzne przedstawienie przypadków użycia, aktorów oraz związków między nimi występujących w danej dziedzinie przedmiotowej	uc	Specyfikowanie i precyzowanie wymagań funkcjonalnych	Diagram przypadków użycia
Rozszerzony diagram czynności	Graficzne przedstawienie sekwencyjnych i (lub) współbieżnych przepływów sterowania oraz danych pomiędzy uporządkowanymi ciągami czynności, akcji oraz obiektów	act	Analiza funkcjonalna	Diagram czynności
Diagram sekwencji	Rodzaj diagramu interakcji opisujący interakcje pomiędzy instancjami klasyfikatorów systemu w postaci sekwencji komunikatów wymienianych między nimi	sd	Analiza i projektowanie	Diagram sekwencji

Tabela 1.4. Rodzaje diagramów SysML — ciąg dalszy

Rodzaj diagramu języka SysML 1.2	Definicja	Wyróżnik	Dyscyplina zastosowań	Odpowiednik w języku UML 2.4
Diagram maszyny stanowej	Graficzne odzwierciedlenie dyskretnych skokowych zachowań skończonych systemów stan-przejsie	stm	Projektowanie systemów, symulacje i generowanie szkieletowego kodu źródłowego	Diagram maszyny stanowej
Diagram definiowania bloków	Graficzne przedstawienie struktury systemu w postaci bloków, ich cech i związków	bdd	Analiza i projektowanie	Diagram klas
Diagram bloków wewnętrznych	Graficzne przedstawienie wewnętrznej struktury bloku wyrażanej poprzez wzajemnie powiązane części bloków	ibd	Analiza i projektowanie	Diagram struktur połączonych
Diagram parametryczny	Graficzne przedstawienie ograniczeń parametrycznych pomiędzy kategoriami modelowania struktury systemu	par	Analiza wydajnościowa i ilościowa	—
Diagram pakietów	Graficzne przedstawienie logicznej struktury systemu w postaci zestawu pakietów połączonych zależnościami i zagnieżdżeniami	pkg	Zarządzanie modelami	Diagram pakietów

Źródło: (Wrycza i Marcinkowski, 2010)

1.6. UML 2.x — najistotniejsze modyfikacje standardu

Zmiany, które wprowadzono w języku UML, począwszy od wersji 2.0, można **liczyć w setkach**. W znakomitej większości są to jednak modyfikacje drobne, głównie o charakterze poprawek edytorskich lub usuwania dość oczywistych przeoczeń czy braku precyzji. Ogromny wysiłek poświęcony na te zmiany sprzyjał zapewnieniu spójności dokumentacji oraz jej zgodności z innymi formalnymi specyfikacjami OMG. W związku z tym merytoryczna zawartość większości pozycji książkowych traktujących o UML 2.0 pozostaje aktualna. Jednak w stosunku do tego poziomu wiedzy autorzy zawarli w niniejszej pracy najważniejsze w ich opinii zmiany z punktu widzenia profesjonalisty uczestniczącego w **procesie modelowania systemu informatycznego**. Są to:

- ❑ diagramy profili wprowadzone w wersji 2.2;
- ❑ kategoria przynależności asocjacji dostępna od wersji 2.1.1;
- ❑ możliwość specyfikowania atrybutów URI (*Unified Resource Identifier*) w odniesieniu do pakietów wprowadzona w wersji 2.4.

1.6.1. Diagramy profili

Propozycja siódmego diagramu struktury języka UML jest **najbardziej zauważalną zmianą** wprowadzoną w zakresie standardu UML od 2005 roku. Przy poniższej prezentacji tego nowego rodzaju diagramów zachowano konwencję opisu każdego z rodzajów diagramów UML stosownie do jednolitego wzorca zastosowanego w podstawowym podręczniku autorów (Wrycza, Marcinkowski i Wyrzykowski, 2005).

1.6.1.1. Znaczenie diagramów profili

Intensywne wykorzystanie możliwości tworzenia profili języka UML przez indywidualnych autorów, organizacje oraz firmy zainspirowało zespół odpowiedzialny za rozwój i spójność specyfikacji tego języka do **formalnego sprecyzowania mechanizmu wizualizacji architektury profili**. Tym samym taksonomia diagramów języka UML została wzbogacona o diagram profili będący w istocie szczególnym przypadkiem diagramów pakietów ukierunkowanym na zagadnienia związane z rozszerzeniami standardu wyjściowego.



Diagram profili to graficzne przedstawienie implementacji mechanizmów rozszerzalności ukierunkowanej na adaptację i kastomizację określonych kategorii modelowania metamodelu języka UML do różnych platform i specjalistycznych dziedzin zastosowań.

Podstawą konstrukcji profili UML jest **metamodel** tego języka. Wiodącym zadaniem metamodelu jest specyfikacja koncepcji, zasad i elementów składowych samego języka UML. Metamodel zawiera metaklasy, które są przenoszone, a następnie dostosowywane do specyfiki dziedziny zastosowania oraz rozszerzane przez dany profil. Selekcja metaklas z metamodelu przez profil odbywa się zgodnie z **zasadami filtrowania**. Zasady te określają elementy składowe metamodelu, które widoczne są w profilu w trakcie jego użytkowania, podczas gdy pozostałe elementy zostają ukryte. W szczególnym przypadku profil może importować kompletny metamodel języka UML. Profil ujmuje kategorie modelowania bazujące na metaklasach:

- ❑ bezpośrednio przenoszonych do profilu poprzez powiązanie metaklasy z docelowym profilem z wykorzystaniem zależności odwołania;
- ❑ ujętych w pakiecie powiązanych z docelowym profilem z wykorzystaniem zależności odwołania, o ile nie dokonano bezpośredniego importowania pakietu (pakietów) zagnieżdżonego źródłowego pakietu lub jego elementów;
- ❑ niebędących bezpośrednim lub pośrednim adresatem zależności odwołania, lecz powiązanych związkiem rozszerzania z co najmniej jednym stereotypem stanowiącym element składowy profilu docelowego.

1.6.1.2. Kategorie pojęciowe oraz notacja graficzna

Jako że diagram profili jest adaptacją diagramu pakietów języka UML, aktualne pozostają wszystkie reguły wykorzystania pakietów w projekcie informatycznym. W szczególności dotyczy to zagnieżdżania pakietów, alternatywnych sposobów notacji pakietów czy też wprowadzania związków pomiędzy pakietami. Naturalnie diagram profili wprowadza szereg **oryginalnych kategorii modelowania** oraz stereotypów. Kategorie te ujęto w tabeli 1.5.

Tabela 1.5. Kategorie pojęciowe diagramu profili UML i ich notacja

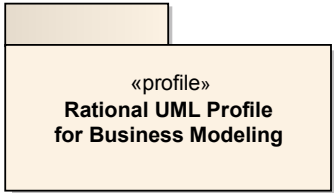
Nazwa	Notacja	Charakterystyka
Profil (ang. <i>profile</i>)		Profil stanowi szczególny przypadek pakietu języka UML pozwalający na rozszerzenie metamodelu podlegającego profilowaniu poprzez adaptację i kustomizację z uwzględnieniem kategorii modelowania specyficznych dla danej dziedziny zastosowań, platformy czy też metody tworzenia oprogramowania.

Tabela 1.5. Kategorie pojęciowe diagramu profili UML i ich notacja — ciąg dalszy

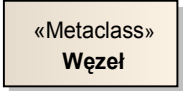
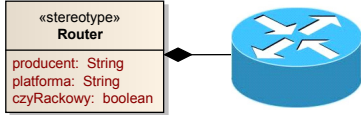
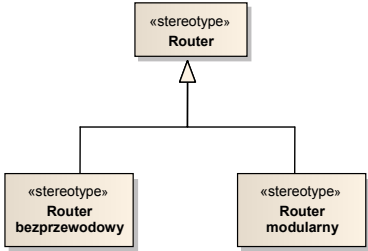
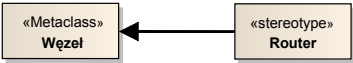
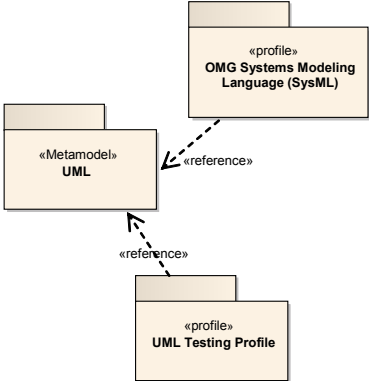
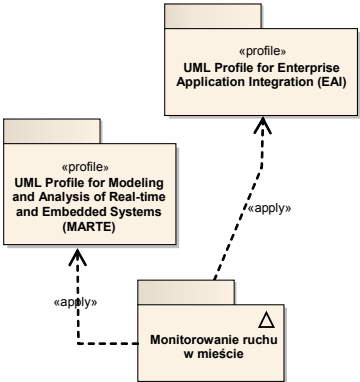
Nazwa	Notacja	Charakterystyka
Metaklasa (ang. <i>metaclass</i>)		Element składowy profilu będący podstawą definicji klas w modelu docelowym wykorzystującym dany profil. Metaklasa może podlegać rozszerzaniu poprzez przyporządkowanie jej autorskiego stereotypu bądź zestawu stereotypów. Tym samym instancje metaklas w modelu docelowym przejmują wszystkie właściwości i rozszerzenia danej metaklas.
Stereotyp (ang. <i>stereotype</i>)	a)  b)  c) 	Mechanizm rozszerzalności języka UML przypisujący dodatkowe cechy opisowe lub graficzne metaklasie adaptowanej przez profil z metamodelu odniesienia. Stereotypy mogą poza nazwą posiadać dodatkowe atrybuty oraz powiązane elementy graficzne.
Uogólnienie (ang. <i>generalization</i>)		Związek o charakterze taksonomicznym umożliwiający dziedziczenie cech pomiędzy stereotypami.

Tabela 1.5. Kategorie pojęciowe diagramu profili UML i ich notacja — ciąg dalszy

Nazwa	Notacja	Charakterystyka
Rozszerzenie (ang. <i>extension</i>)		Rodzaj asocjacji dedykowany przypisywaniu stereotypu bądź stereotypów do wybranej metaklasy w ramach profilu. Rozszerzenie może przyjmować wszystkie cechy asocjacji (takie jak nazwa, role, liczebności, agregacja, kwalifikacja czy przynależność), z wyłączeniem nawigacji. Domyślnie przyjmuje się, że wprowadzane przez profil stereotypy mogą być stosowane uznaniowo. W przeciwnym przypadku rozszerzeniu przypisuje się ograniczenie <i>required</i>.
Zależność odwołania (ang. <i>reference</i>)		Związek zależności identyfikujący metamodelę lub ich metaklasy, do których nawiązuje dany profil.
Zależność zastosowania profilu (ang. <i>profile application</i>)		Związek zależności oznaczający wykorzystanie przez dany pakiet (model) zawartości wskazanych profili.

Źródło: opracowanie własne na podstawie (OMG, 2011)

1.6.1.3. Studium diagramów profili

Profile języka UML mogą w sposób zasadniczy różnić się skalą złożoności. Po wszechnie występują profile, których skala odpowiada zawartości pojedynczego pakietu, koncentrujące się na ściśle specjalistycznym zagadnieniu. Formie takiej odpowiadają chociażby proponowane przez różnych autorów rozszerzenia standardu mające na celu dostosowanie języka UML do potrzeb modelowania biznesowego. Z drugiej strony, opracowuje się rozwiązania, których zawartość merytoryczna wymaga uporządkowania w kilkudziesięciu pakietach tworzących złożoną architekturę profilu. Na rysunku 1.6 zaproponowano syntetyczny profil usprawniający modelowanie infrastruktury sieciowej organizacji gospodarczej — *netUML Light*. Profil ten rozszerza dwie kluczowe metaklasy diagramu rozłokowania UML — *Węzeł* oraz *Ścieżka komunikowania*. Rozszerzenia dokonano poprzez wprowadzenie szeregu stereotypów zorganizowanych w merytorycznie wyodrębnione struktury hierarchiczne. Stereotypy *Router*, *Router bezprzewodowy*, *Przełącznik*, *Chmura*, *Urządzenie końcowe*, *Medium bezprzewodowe* oraz *Kabel szeregowy* mają charakter stereotypów graficznych. Konstrukcja profilu sprawia, że zastosowanie stereotypów ścieżek komunikowania jest obligatoryjne. Jednocześnie profil *netUML Light* uwzględnia podstawowe wyliczeniowe typy wartości, tj. *Standard* oraz *Typ Światłowodów*. Ujmują one listę wartości, które charakteryzują typy odpowiednio: kabli szeregowych oraz światłowodów.

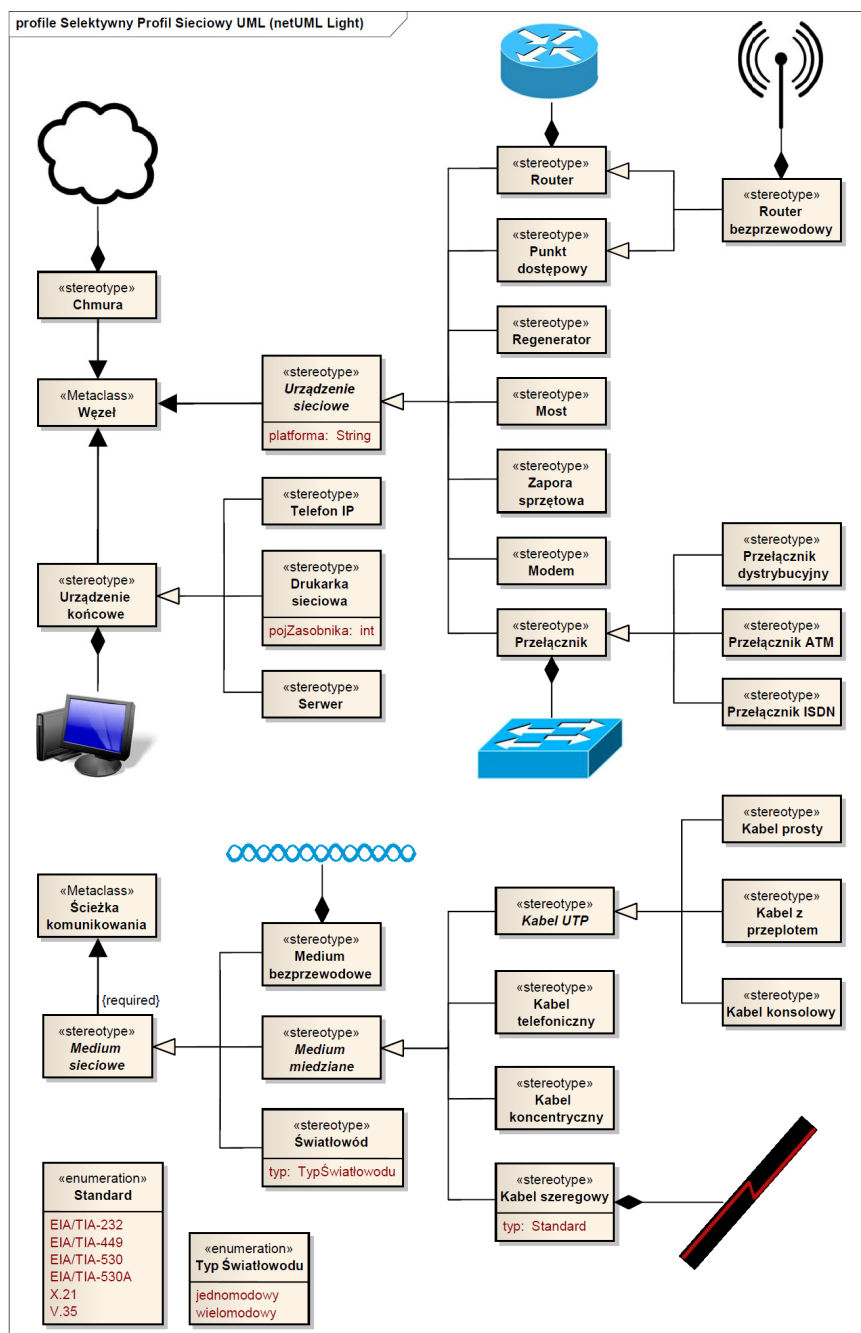
Profil *netUML Light* znalazł zastosowanie w tworzeniu *Modelu Infrastruktury Sieciowej Pomorskiej Akademii Regionalnej Cisco*, jak zaprezentowano na rysunku 1.7. Założenia modelu docelowego ujmowały precyzyjną specyfikację platform oraz modeli zastosowanych urządzeń sieciowych, stąd zdefiniowano *Katalog Produktów*, wyodrębniono jego pakiety składowe i przeprowadzono import zawartości katalogu do opracowywanego modelu.

1.6.2. Przynależność

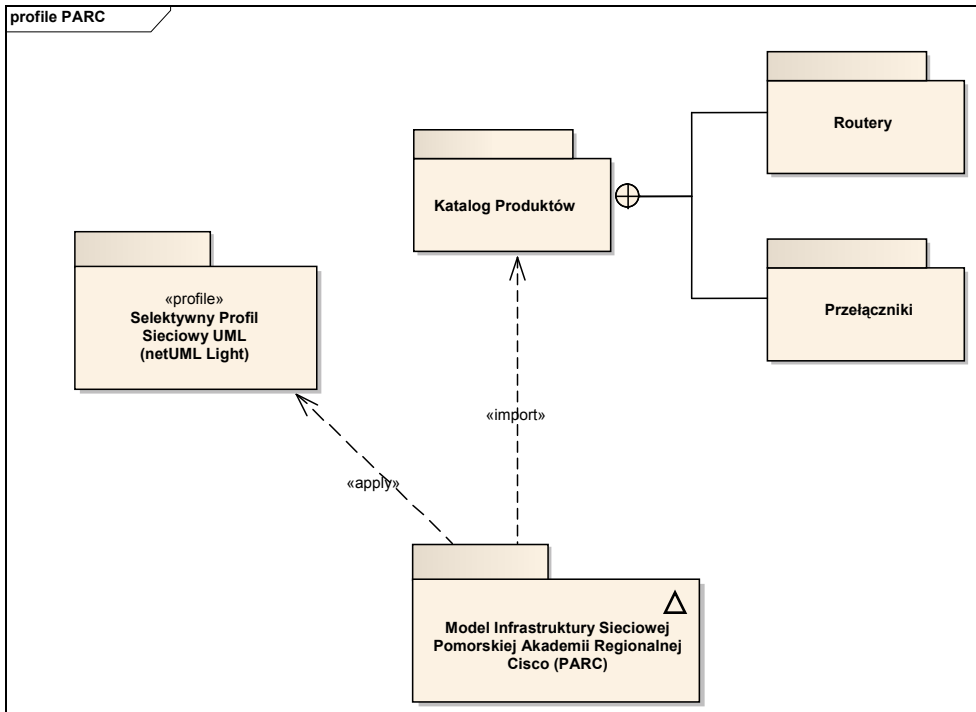
Przynależność (ang. *ownership*) asocjacji rozstrzyga jednoznacznie kwestię **przypisania końcówki asocjacji**, do jednej z dwóch kategorii modelowania:

- ❑ asocjacji łączącej klasyfikatory;
- ❑ klasyfikatora powiązanego daną asocjacją z innym klasyfikatorem.

Kategorię przynależności (Wrycza i Marcinkowski, 2008) należy rozumieć jako jedną ze zdefiniowanych **cech asocjacji**, obok cech takich jak liczebność, nazwa czy role. Przynależność wiąże się ze wszystkimi diagramami UML, w których wykorzystuje się pojęcie asocjacji. Najbardziej istotne znaczenie ma ona naturalnie w diagramach klas. Kategoria przynależności ma graficzną postać niewielkiego, wypełnionego okręgu umieszczonego w końcówce danej asocjacji bezpośrednio przy krawędzi klasyfikatora. Średnica tego okręgu jest w przybliżeniu równa połowie wysokości rombu oznaczającego agregację. Ta różnica wielkości zapobiega myleniu przynależności i agregacji (zwłaszcza całkowitej).

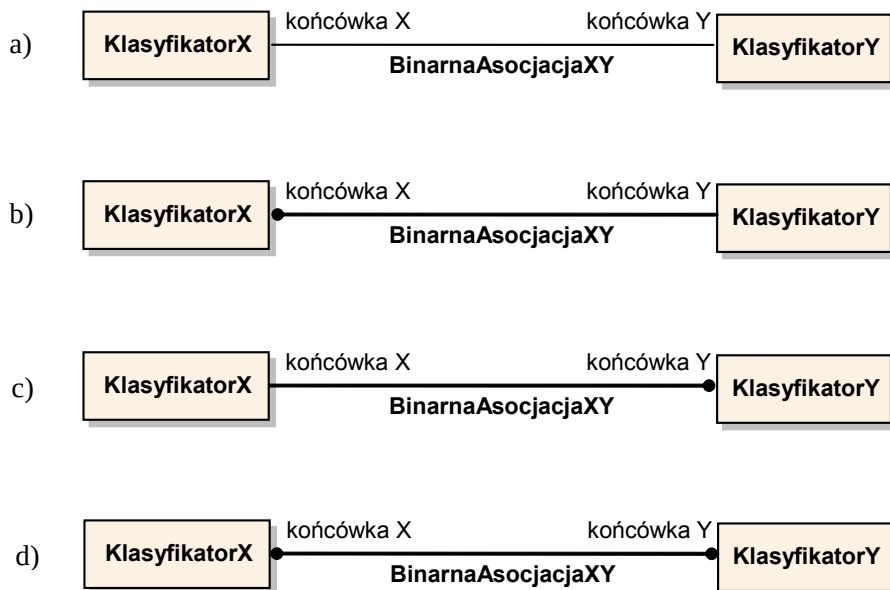


Rysunek 1.6. Definiowanie profilu w języku UML



Rysunek 1.7. Wykorzystanie profilu języka UML

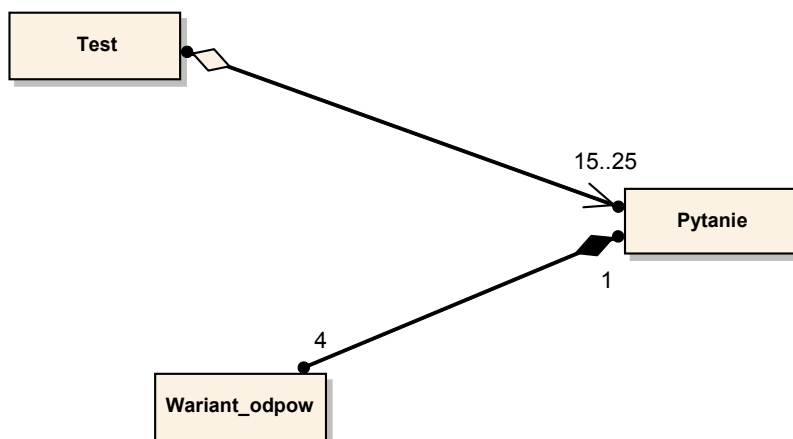
Interpretację kategorii przynależności przedstawiono na rysunku 1.8. I tak, nieobecność tego oznaczenia wskazuje na przynależność końcówki do asocjacji, natomiast jego obecność — na przynależność końcówki do klasyfikatora zamieszczonego po przeciwnej stronie asocjacji (zakładając zastosowanie asocjacji binarnej). Tym samym na rysunku 1.8a obie końcówki asocjacji należą do *BinarnejAsocjacjiXY* łączącej oba klasyfikatory. Na rysunku 1.8b *końcówkaX* zakończona symbolem wypełnionego okręgu przynależy do *klasyfikatoraY*. Z kolei *końcówkaY* pozbawiona tego oznaczenia przynależy do *BinarnejAsocjacjiXY*. Na rysunku 1.8c przedstawiono sytuację odwrotną, podczas gdy na rysunku 1.8d obie końcówki asocjacji przynależą do łączonych klasyfikatorów, odpowiednio *klasyfikatoraY* oraz *klasyfikatoraX*. Na rysunku 1.8 zastosowano uniwersalne oznaczenie klasyfikatora, które powinno być przetransponowane na docelowym diagramie na odpowiednie oznaczenia klasyfikatorów właściwych dla danego rodzaju diagramu.



Rysunek 1.8. Notacja przynależności

Źródło: (Wrycza i Marcinkowski, 2008)

Asocjacja wprowadzająca kategorię przynależności może być opisywana **w połączeniu z innymi cechami asocjacji**, tj. nazwami, rolami, nawigacją, liczebnością, kwalifikacją czy też agregacją, jak zaprezentowano na rysunku 1.9.



Rysunek 1.9. Łączenie kilku cech asocjacji

Źródło: (Wrycza i Marcinkowski, 2008)

Kategoria przynależności ma istotny wpływ na rozmieszczenie związanych z asocjacją kategorii nawigacji, agregacji i kwalifikacji. Szczególnego wyjaśnienia wymaga kwestia nawigacji i agregacji, nie tylko ze względu na swoją zawartość merytoryczną, ale i **poprawność graficzną**. Oznaczenia nawigacji i agregacji (całkowitej i częściowej) umieszczane były bowiem dotychczas samodzielnie w końcówce asocjacji bezpośrednio przy krawędzi klasyfikatora. Obecnie graficznie przy krawędzi umieszcza się w pierwszej kolejności notację przynależności, a dopiero w dalszej oznaczenia innych cech asocjacji.

W przypadku wersji 2.0 standardu UML w odniesieniu do nawigacji zakładano, że końcówka z nieprzypisanym symbolem kierunku nawigacji (strzałką) była przynależna do asocjacji, podczas gdy druga końcówka — do przeciwległego klasyfikatora. W świetle wprowadzenia odrębnej notacji przynależności wraz z wersją 2.1.1 standardu UML nie można dalej czynić takiego założenia. Wcześniejsza konwencja oznaczania zdezaktualizowała się, bowiem oznaczenia przynależności i nawigacji należy stosować *łącznie* (por. rysunek 1.9).

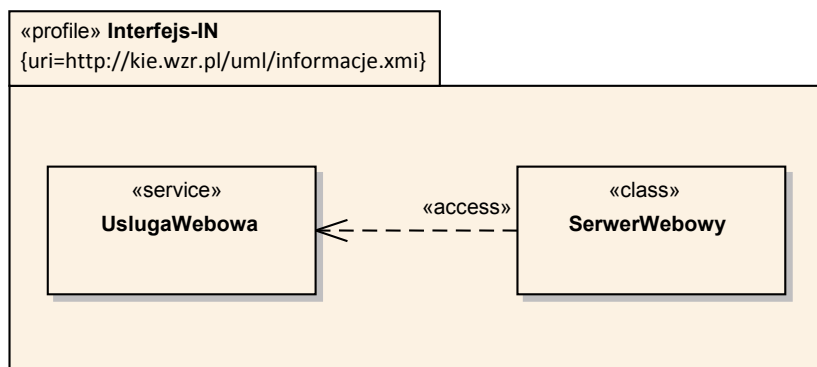
1.6.3. Atrybut URI

Począwszy od wersji 2.4 języka UML, pakiet może posiadać opcjonalny atrybut URI, który pełni rolę **unikalnego identyfikatora pakietu**. Został on wprowadzony przede wszystkim w celu wspomagania wymiany komunikatów profili z zastosowaniem standardu XMI (*XML Metadata Interchange*), czyli ujednoliconego sposobu wymiany informacji o metadanych.

Wprowadzenie atrybutu URI jest podyktowane podążaniem za regułami i składnią specyfikacji **IETF URI** opisanej w dokumencie RFC 2396, a następnie rozwiniętej również w specyfikacji RFC 3986. Obie wspomniane specyfikacje pozwalają na zapisywanie URI jako URL (*Unified Resource Locator*), URN (*Unified Resource Name*) lub w obu tych formach jednocześnie. Przykładami takich nazw mogą być:

- ❑ `http://kie.wzr.pl/uml/informacje.xmi` (URL);
- ❑ `urn:pl:wzr:kie:informacje:1.8.3` (URN).

Atrybut URI może być zapisywany w formie { uri = <uri> } po nazwie profilu lub pakietu, jak zaprezentowano na rysunku 1.10.



Rysunek 1.10. Pakiet wzbogacony o atrybut URI



Diagram wymagań systemowych

2.1. Ćwiczenia

Ć W I C Z E N I E

2.1 Serwis internetowy obsługi konferencji międzynarodowej

Popularną platformą spotkań i wymiany myśli, koncepcji, rozwiązań, problemów badawczych pracowników nauki są międzynarodowe konferencje. Ich wartość mierzy się poziomem naukowym, o który dba komitet programowy konferencji, a także poziomem organizacyjnym, za który odpowiedzialny jest komitet organizacyjny. Silne ośrodki naukowe i kraje o wysoko rozwiniętej technologii informatycznej — jak również miasta wspierające turystykę biznesową — starają się przyciągnąć jak największą liczbę konferencji. Są one jednym z czynników budujących prestiż kraju i miasta, a przede wszystkim narzędziem stymulowania działalności naukowej danego ośrodka naukowego.

Organizacja i przebieg cyklicznej konferencji obejmuje przede wszystkim następujące etapy i czynności:

1. Opracowanie oferty konferencji w odpowiedzi na zaproszenie do zgłaszania propozycji takiego wydarzenia. Propozycja zawiera: zaproszenie do konferencji, jej obszary tematyczne, skład komitetów: sterującego, programowego i organizacyjnego, harmonogram przygotowań i realizacji konferencji, planowany budżet, opis miejsca konferencji, listę sponsorów, patronów honorowych oraz medialnych, bazę hotelową, obsługę przez lokalnych dostawców usług konferencyjnych.

2. Zaprezentowanie oferty przed Komitetem Sterującym odpowiedzialnym za cykl konferencji. W przypadku pozytywnej decyzji przejście do kolejnych etapów.
3. Po zaakceptowaniu oferty podjęcie następujących działań:
 - ☐ ustalenie ostatecznego składu Komitetu Programowego;
 - ☐ ustalenie docelowej listy tematów konferencji;
 - ☐ ustalenie ostatecznego składu Komitetu Organizacyjnego;
 - ☐ uzgodnienie szczegółów i podpisanie umowy z ośrodkiem, w którym konferencja się odbędzie;
 - ☐ ustalenie wysokości opłat konferencyjnych i treści formatki rejestracyjnej uczestników;
 - ☐ opracowanie zawartości portalu konferencyjnego;
 - ☐ zaproszenie głównych prelegentów (ang. *keynote speakers*);
 - ☐ opracowanie zaproszenia na konferencję (wersja elektroniczna i papierowa);
 - ☐ nawiązanie współpracy z siecią i serwisem internetowym hoteli;
 - ☐ wyznaczenie przewodniczących ścieżek tematycznych podejmujących również promocję konferencji;
 - ☐ rozesłanie zaproszenia elektronicznego (CfP, *Call for Papers*) na tematyczne listy dyskusyjne, a wersji papierowej do potencjalnie zainteresowanych ośrodków naukowych;
 - ☐ rozpoczęcie negocjacji i podpisanie umowy z wydawnictwem;
 - ☐ zaproszenie ośrodków naukowych do udziału w konsorcjum doktoranckim;
 - ☐ zaproszenie ośrodków naukowych do udziału w warsztatach i tutorialach;
 - ☐ zaproszenie producentów i usługodawców zainteresowanych konferencją do uczestnictwa w tematycznej wystawie;
 - ☐ ustalenie zasad recenzowania referatów;
 - ☐ stworzenie lub zainstalowanie serwisu formatowania referatów oraz ich recenzowania.
4. Przyjmowanie za pośrednictwem systemu konferencyjnego propozycji:
 - ☐ referatów;
 - ☐ paneli;
 - ☐ warsztatów;
 - ☐ tutoriali;
 - ☐ plakatów;
 - ☐ wystąpień w ramach konsorcjum doktoranckiego;
 - ☐ zgłoszeń uczestników niebędących autorami.

5. Recenzowanie referatów:
 - ☐ zamknięcie listy nadesłanych referatów;
 - ☐ udostępnienie recenzentom kont wraz z hasłami dostępu;
 - ☐ wyznaczenie lub wylosowanie recenzentów dla poszczególnych referatów;
 - ☐ monitorowanie postępów recenzowania i wysyłanie ponagleń;
 - ☐ zamknięcie procesu recenzowania;
 - ☐ ocena referatów na podstawie recenzji i podjęcie jednoznacznej decyzji: akceptacja lub odrzucenie referatu;
 - ☐ wybór najlepszych referatów;
 - ☐ wysyłanie decyzji autorom referatów.
6. Czynności bezpośrednio przedkonferencyjne:
 - ☐ przygotowanie programu konferencji na podstawie efektów procesu recenzowania;
 - ☐ zorganizowanie referatów w sesje tematyczne;
 - ☐ ustalenie pełnego harmonogramu konferencji;
 - ☐ ostateczna rejestracja rezerwacji w serwisach hotelowych;
 - ☐ rejestracja na imprezy towarzyszące;
 - ☐ podpisanie umowy z firmą cateringową.
7. Realizacja konferencji:
 - ☐ finalizacja rejestracji uczestników i wręczenie materiałów konferencyjnych;
 - ☐ monitorowanie przebiegu konferencji;
 - ☐ publikowanie programu poszczególnych sesji konferencji;
 - ☐ realizacja płatności;
 - ☐ pełne rozliczenie finansowe konferencji;
 - ☐ ewaluacja konferencji.

Nadrzędnym wymaganiem jest *Obsługa konferencji z wykorzystaniem technologii sieciowych*. Niniejszemu wymaganiu przydzielono zatem numer porządkowy *K1*. System obsługi konferencji międzynarodowej musi w szczególności automatyzować *Zarządzanie zawartością portalu konferencyjnego (K1.1)* i *Obsługę referatów (K1.2)* oraz zapewniać *Wsparcie realizacji konferencji (K1.3)*. Wymienione funkcjonalności zamieszczono więc w hierarchii wymagań bezpośrednio pod wymaganiem o numerze *K1*. Należy podkreślić, że dobór nazw wymagań jest kwestią elastyczną. Mogą to być zarówno nazwy czynności, jak i specyfikacje parametrów technicznych, cech wydajnościowych, niezbędnych do spełnienia kryteriów w zakresie bezpieczeństwa itd.

W ramach *Zarządzania zawartością portalu konferencyjnego* wyszczególniono następujące wymagania na niższym poziomie hierarchii:

- ❑ *Opracowywanie oferty konferencji (K1.1.1);*
- ❑ *Prezentacja oferty konferencji (K1.1.2);*
- ❑ *Wsparcie dla zróżnicowanych kanałów komunikacji (K1.1.3);*
- ❑ *Obsługa sprofilowanych kont użytkowników (K1.1.4);*
- ❑ *Obsługa zaproszeń (K1.1.5).*

Z kolei wymaganie *Obsługa referatów* jest wymaganiem nadrzędnym dla poniższych wymagań systemowych:

- ❑ *Recenzowanie referatów (K1.2.1);*
- ❑ *Zarządzanie treścią materiałów konferencyjnych (K1.2.2).*

Wsparcie realizacji konferencji warunkuje zawarcie usług wykorzystywanych w systemie już po przeprowadzeniu oceny oraz selekcji zgłoszonych referatów przy wykorzystaniu wsparcia systemu obsługi konferencji międzynarodowej. Na etapie wstępnych prac nad modelem systemu wyszczególniono w tym kontekście następujące funkcjonalne wymagania systemowe:

- ❑ *Obsługa umów z interesariuszami (K1.3.1);*
- ❑ *Obsługa procesu rejestracji uczestników (K1.3.2);*
- ❑ *Harmonogramowanie konferencji (K1.3.3);*
- ❑ *Monitorowanie finansów konferencji (K1.3.4).*

Na podstawie powyższego opisu należy opracować diagram wymagań systemowych serwisu internetowego obsługi konferencji (rysunek 2.1).



Wskazówki dydaktyczne

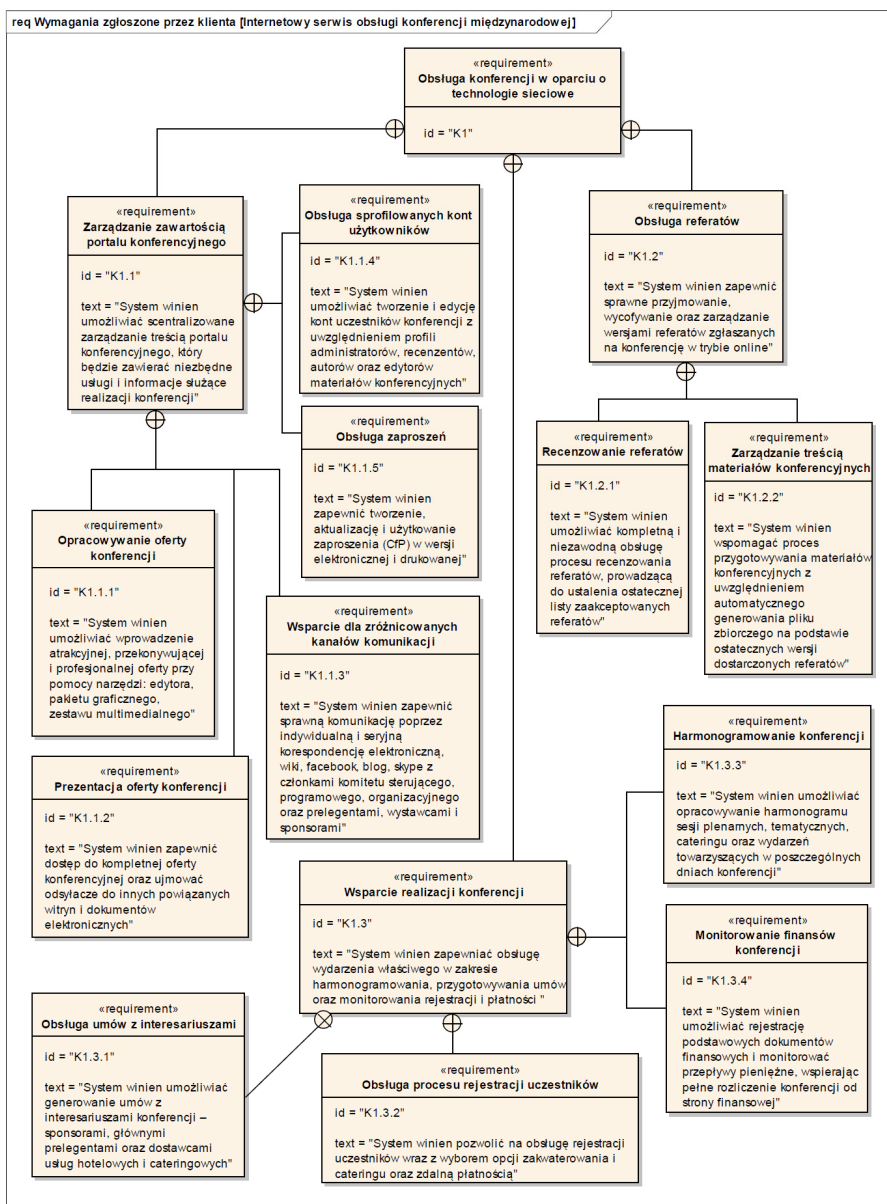
Na podstawie przedstawionej powyżej opisowej charakterystyki procesu organizacji konferencji sformułowano najistotniejsze wymagania w stosunku do powstającego systemu obsługi konferencji. Wymagania te zostały sformalizowane do postaci diagramu na rysunku 2.1. Należy pamiętać, że poszczególne wymagania systemowe można następnie dekomponować do postaci uszczegółowionych diagramów wymagań systemowych, jak zaprezentowano w ćwiczeniu 2.2.

Ć W I C Z E N I E

2.2

Wymaganie recenzowania referatów nadesłanych na konferencję

O prestiżu konferencji świadczy przede wszystkim jakość wygłoszonych na niej referatów. Stąd Komitety konferencyjne — Organizacyjny, a zwłaszcza Programowy — przykładają wielką wagę do procesu recenzowania. Niezwykle istotną i wrażliwą kwestią jest opracowanie bogatej funkcjonalności podsystemu nadsyłanych prac —



Rysunek 2.1. Diagram wymagań serwisu internetowego obsługi konferencji międzynarodowej

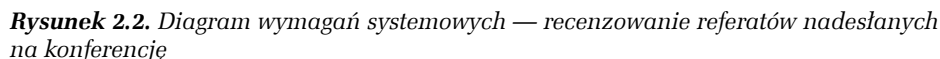
proponowanych referatów, plakatów i autoreferatów prac doktorskich. Funkcjonalność ta powinna umożliwić precyzyjną ocenę wartości merytorycznej referatu, jego oryginalnego wkładu naukowego, a także zagadnień edytorskich. Złożone funkcje

recenzowania realizowane są przez członków Komitetu Programowego i recenzentów za pomocą elektronicznych pulpitów dostępnych za pośrednictwem przeglądarki internetowej. W związku z hierarchicznym procesem recenzowania zawartość pulpitów udostępnianych recenzentom, przewodniczącym ścieżek tematycznych i członkom Komitetu Programowego podejmującym ostateczną decyzję jest zróżnicowana.

Centralnym zagadnieniem w procesie recenzowania referatów jest udostępnienie *Dynamicznej zawartości pulpitu*. *Dynamiczna zawartość pulpitu* jest podstawą funkcjonowania projektowanego portalu konferencyjnego bazującego na technologii LAMP (Linux, Apache, MySQL, PHP). Dynamiczna zawartość jest realizowana z wykorzystaniem języków AJAX oraz JavaScript. Dostęp do pulpitu musi być chroniony, stąd sformułowano wymaganie *Zapewnienie bezpieczeństwa w logowaniu przez system haseł, captcha oraz bezpieczny protokół https*. Aby zapewnić aktualność list referatów ujmowanych w ramach poszczególnych pulpitów, niezbędne jest *Bieżące utrzymywanie bazy referatów*.

Funkcjonalność *Dynamiczna zawartość pulpitu* jest powielana na docelowe *pulpity: recenzenta* oraz *członka Komitetu Programowego*. Obydwa te wymagania powiązane są z wykorzystaniem zależności stereotypowanej copy z *Dynamiczną zawartością pulpitu*, przez co możliwe jest przeniesienie treści tego wymagania źródłowego do obydwu wymagań docelowych. Zależność powielania stwarza użyteczną możliwość łatwego aktualizowania wszystkich wymagań docelowych w przypadku dokonania dowolnej pożądanej zmiany w wymaganiu źródłowym. Umożliwia to praktyczne zrealizowanie koncepcji ponownego wykorzystania oprogramowania na zasadzie software reuse. W związku z zastosowaniem zależności powielania obydwu rodzajom pulpitów udostępniana jest funkcjonalność utrzymywania bazy zgłoszonych prac oraz zapewnienia bezpieczeństwa w systemie. Wymagania zawarte w *Dynamicznej zawartości pulpitu recenzenta* dekomponowane są na pozostałe szczegółowe wymagania przedstawione na rysunku 2.2. Są to *Wprowadzanie i aktualizacja recenzji*, *Rejestrowanie uwag dla członków Komitetu Programowego w strukturze hierarchicznej procesu recenzowania* oraz *Monitorowanie stopnia wykonania uprzednio zgłoszonych poprawek*.

W kontekście *Dynamicznej zawartości pulpitu recenzenta* recenzent ma możliwość *Wprowadzania i aktualizacji recenzji* oraz *Rejestrowania uwag dla członków Komitetu Programowego*. Jeżeli recenzja nie zostanie przesłana przez recenzenta w określonym terminie, następuje *Automatyczne ponaglanie recenzentów* poprzez rozsyłanie listów elektronicznych przypominających o datach. Zamawiający system zgłosił konieczność obsługi funkcji *Monitorowania stopnia wdrożenia uprzednio zgłoszonych poprawek* zarówno przez recenzentów, jak i przez Komitet Programowy. Tym samym wymaganie to zagnieźdżono zarówno w *Dynamicznej zawartości pulpitu recenzenta*, jak i w *Dynamicznej zawartości pulpitu członka Komitetu Programowego*.



Określone wymaganie może być potencjalnie powiązane z kilkoma innymi wymaganiami poprzez związki i zależności. I tak, *Generowanie statystyk* powiązane jest związkiem zagnieżdżenia z *Dynamiczną zawartością pulpitu członka Komitetu Programowego* i zależnością śledzenia ze *Zbiorną wysyłką listów zawierających recenzję*, ponieważ statystyki te można określić dopiero po dokonaniu zbiorczej wysyłki. Również *Edytowanie treści listu akceptującego lub odrzucającego* jest w związku zagnieżdżenia z *Dynamiczną zawartością pulpitu członka Komitetu*

Programowego. Zbiorcza wysyłka listów zawierających recenzję jest możliwa w wyniku wcześniejszego Edytowania treści listu akceptującego lub odrzucającego referat, toteż te dwa wymagania powiązane są zależnością śledzenia (stereotyp trace). Naturalną konsekwencją rozesłania listów akceptujących lub odrzucających poszczególne referaty jest konieczność zawarcia uwag redakcyjnych i wskazówek dotyczących ostatecznego formatu referatu. Stąd sformułowano wymaganie Przesyłania autorom instrukcji lub odnośnika do standardów redagowania zaakceptowanego referatu, które jest wymaganiem pochodnym w stosunku do Zbiorczej wysyłki listów zawierających recenzję.

Na podstawie powyższego opisu należy opracować diagram dla wymagania o nazwie *Recenzowanie referatów nadesłanych na konferencję* (rysunek 2.2).



Wskazówki dydaktyczne

Niniejsze ćwiczenie ilustruje jeden z kroków dekompozycji wymagań systemowych — od wymagań sformułowanych w sposób ogólny, poprzez kolejne poziomy szczegółowości, aż do poziomu wymagań elementarnych. W omawianym ćwiczeniu obsługi konferencji międzynarodowej do dekompozycji wybrano wymaganie recenzowania referatów nadesłanych na konferencję. W odniesieniu do innych wymagań zidentyfikowanych w ćwiczeniu 2.1 opracowywane są odrębne diagramy wymagań systemowych.

Ć W I C Z E N I E

2.3

Tabelaryczna specyfikacja wymagań systemowych

Alternatywną formą prezentacji wymagań jest postać tabelaryczna. W tabeli tej określa się rodzaje związków (zagnieżdżenie lub inne rodzaje zależności) pomiędzy wymaganiami docelowymi a źródłowymi. Tabela wymagań powinna zatem zawierać pięć następujących kolumn:

- ☐ numer porządkowy wymagania docelowego;
- ☐ nazwa wymagania docelowego;
- ☐ rodzaj zależności;
- ☐ numer porządkowy wymagania źródłowego;
- ☐ nazwa wymagania źródłowego.

Na podstawie ćwiczenia 2.2 należy opracować zestawienie wymagań języka SysML w postaci tabelarycznej (tabela 2.1).

Tabela 2.1. Dokumentacja związków pomiędzy wymaganiami w postaci tabelarycznej

Wymaganie docelowe		Rodzaj związku	Wymaganie źródłowe	
Numer porządkowy	Nazwa		Numer porządkowy	Nazwa
K1.2.1.4.1	<i>Przydzielanie ustalonej liczby recenzentów w hierarchicznej strukturze procesu recenzowania</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>
K1.2.1.4.2	<i>Edytowanie treści listu akceptującego lub odrzucającego referat</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>
K1.2.1.4.3	<i>Zbiorcza wysyłka listów zawierających recenzję</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>
K1.2.1.4.4	<i>Wybór najlepszego referatu</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>
K1.2.1.4.5	<i>Generowanie statystyk dotyczących akceptacji/odrzućenia referatów w postaci graficznej</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>
K1.2.1.5.1	<i>Wprowadzanie i aktualizacja recenzji</i>	zagnieżdżenie	K1.2.1.5	<i>Dynamiczna zawartość pulpitu recenzenta</i>
K1.2.1.5.2	<i>Wyrażanie i rejestrowanie uwag dla członków Komitetu Programowego w hierarchicznej strukturze procesu recenzowania</i>	zagnieżdżenie	K1.2.1.5	<i>Dynamiczna zawartość pulpitu recenzenta</i>
K1.2.1.3.1	<i>Monitorowanie stopnia wdrożenia uprzednio zgłoszonych poprawek</i>	zagnieżdżenie	K1.2.1.4	<i>Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego</i>

Tabela 2.1. Dokumentacja związków pomiędzy wymaganiami w postaci tabelarycznej — ciąg dalszy

Wymaganie docelowe		Rodzaj związku	Wymaganie źródłowe	
Numer porządkowy	Nazwa		Numer porządkowy	Nazwa
K1.2.1.3.1	Monitorowanie stopnia wdrożenia uprzednio zgłoszonych poprawek	zagnieżdżenie	K1.2.1.5	Dynamiczna zawartość pulpitu recenzenta
K1.2.1.4	Dynamiczna zawartość pulpitu członka Komitetu Organizacyjnego	zależność powielania copy	K1.2.1.3	Dynamiczna zawartość pulpitu
K1.2.1.5	Dynamiczna zawartość pulpitu recenzenta	zależność powielania copy	K1.2.1.3	Dynamiczna zawartość pulpitu
K1.2.1.5.1	Wprowadzanie i aktualizacja recenzji	zależność wyprowadzania derive ↳Req	K1.2.1.4.1	Przydzielanie ustalonej liczby recenzentów w hierarchicznej strukturze procesu recenzowania
K1.2.1.6	Przesyłanie autorom instrukcji lub odnośnika do standardów redagowania zaakceptowanego referatu	zależność wyprowadzania derive ↳Req	K1.2.1.4.3	Zbiorcza wysyłka listów zawierających recenzję
K1.2.1.3	Dynamiczna zawartość pulpitu	zależność śledzenia trace	K1.2.1.1	Bieżące utrzymywanie bazy referatów
K1.2.1.3	Dynamiczna zawartość pulpitu	zależność śledzenia trace	K1.2.1.2	Zapewnienie bezpieczeństwa w logowaniu przez system haseł, captcha oraz bezpieczny protokół https

Tabela 2.1. Dokumentacja związków pomiędzy wymaganiami w postaci tabelarycznej — ciąg dalszy

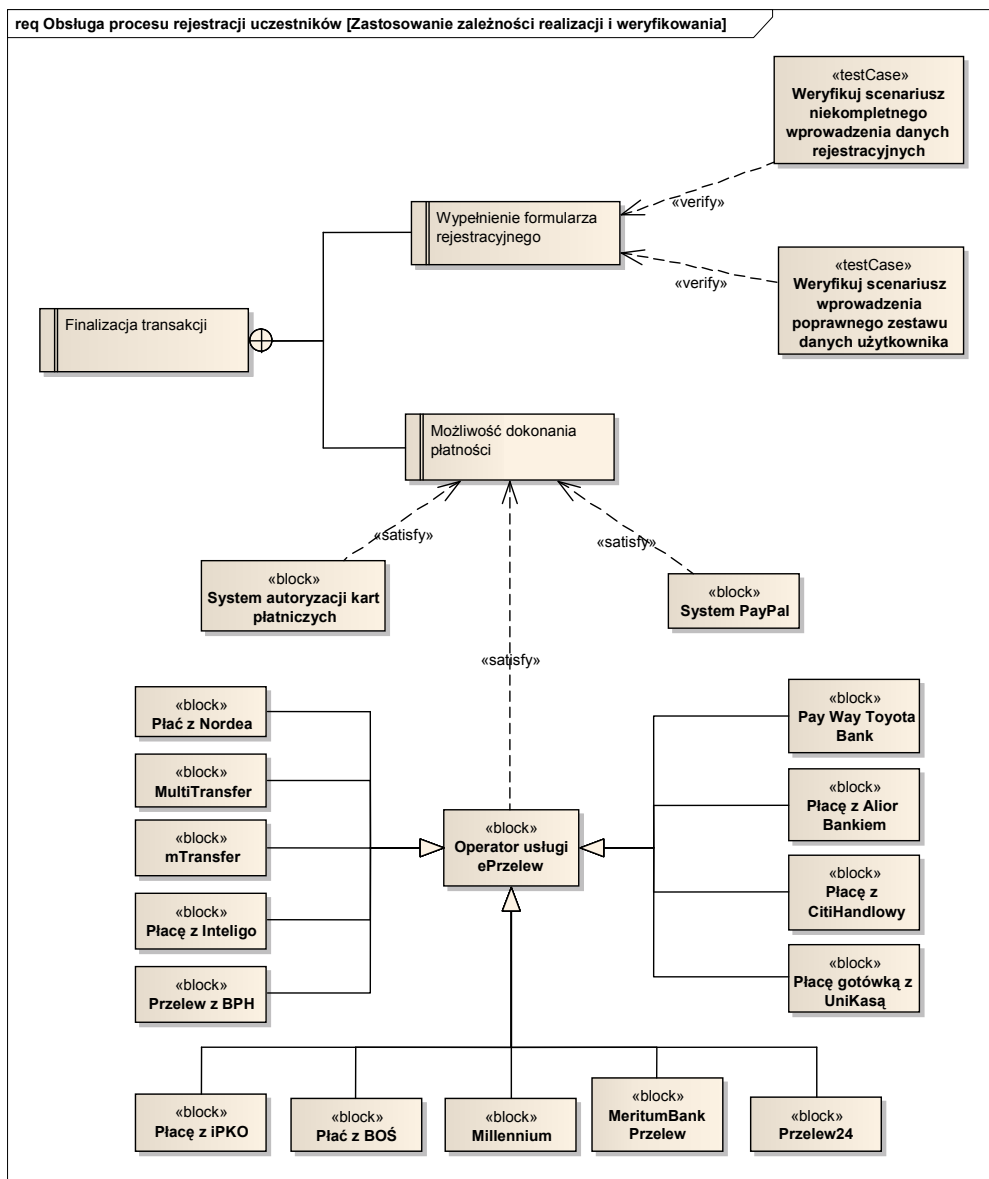
Wymaganie docelowe		Rodzaj związku	Wymaganie źródłowe	
Numer porządkowy	Nazwa		Numer porządkowy	Nazwa
K1.2.1.7	Automatyczne ponaglanie recenzentów w przypadku niedotrzymania terminów przeprowadzenia recenzji	zależność śledzenia trace	K1.2.1.5.1	Wprowadzanie i aktualizacja recenzji
K1.2.1.4.3	Zbiorcza wysyłka listów zawierających recenzję	zależność śledzenia trace	K1.2.1.4.2	Edytowanie treści listu akceptującego lub odrzucającego referat
K1.2.1.4.5	Generowanie statystyk dotyczących akceptacji/odrzućenia referatów w postaci graficznej	zależność śledzenia trace	K1.2.1.4.3	Zbiorcza wysyłka listów zawierających recenzję

Ć W I C Z E N I E

2.4 Rejestracja i płatności w sklepie internetowym

Obecnie coraz częściej zakupy dokonywane są przez Internet. Klient, który chce zakupić dany produkt, nie musi posiadać wcześniej konta w serwisie internetowym sklepu, a jedynie wypełnić formularz rejestracyjny podczas dokonywania zakupów. Warunkiem *Finalizacji transakcji* w sklepie jest *Wypełnienie formularza rejestracyjnego* powiązanego z dokonaniem płatności. *Wypełnienie formularza rejestracyjnego* łączy się z *Weryfikacją kompletności i poprawności* danych rejestracyjnych. Po wprowadzeniu poprawnych danych klienta istnieje *Możliwość dokonania płatności* poprzez wybrany system płatności elektronicznych. Należy zaznaczyć, że użytkownik może nie rejestrować płatności, a przesyłkę otrzymać za pobraniem. Zestaw płatności dostępnych dla klienta obejmuje *System autoryzacji kart płatniczych*, *System PayPal* oraz *Operator usługi ePrzelew*.

Na podstawie powyższego opisu należy opracować diagram wymagań systemowych dla systemu obsługi rejestracji i płatności za zakupione produkty, wykorzystując w tym celu zależności realizacji oraz weryfikowania (rysunek 2.3).



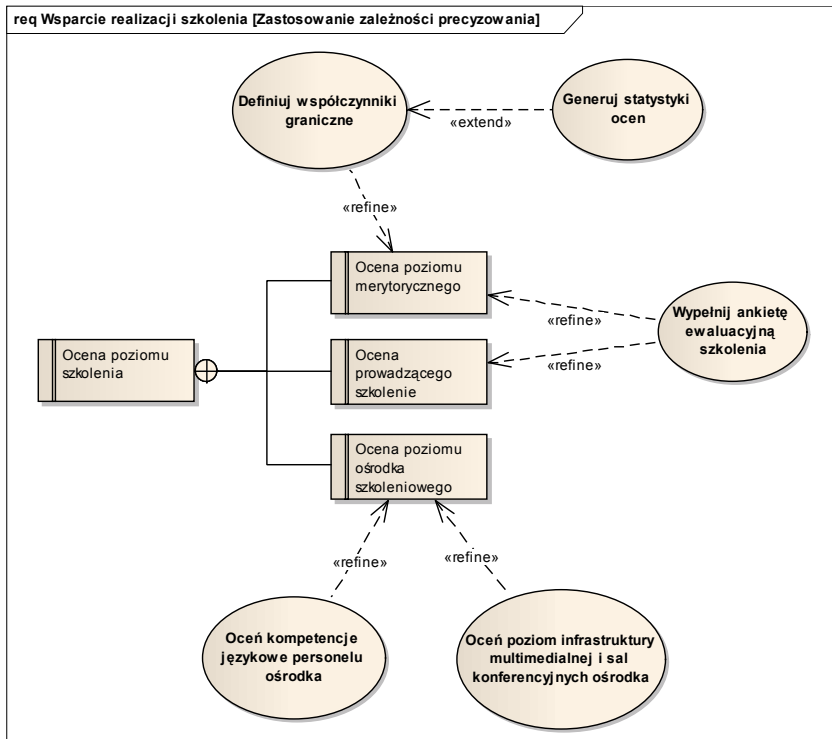
Rysunek 2.3. Diagram wymagań obsługi rejestracji i płatności

ĆWICZENIE

2.5 Ocena poziomu szkolenia

Każde cykliczne szkolenie podlega ocenie, tak aby możliwe było ulepszenie jego kolejnej edycji. Najważniejsza jest *Ocena poziomu merytorycznego* — a poza nią *Ocena poziomu ośrodka szkoleniowego* i *Ocena prowadzącego szkolenie*. Podstawowe dane *Oceny poziomu merytorycznego* otrzymuje się z systemu ewaluacji poprzez *Generowanie statystyk ocen*, na które składają się oceny sali wykładowej, infrastruktury (komputerów), materiałów szkoleniowych czy przydatności szkolenia. W ten sposób oblicza się istotny dla oceny jakości szkolenia wskaźnik poziomu jakości wyrażony procentowo (przyjmuje się, że dla dobrego i użytecznego szkolenia powinien być nie mniejszy niż 70% — wartości te można konfigurować w ramach przypadku użycia *Definiuj współczynniki graniczne*). Każdy z uczestników jest proszony o wypełnienie ankiety ewaluacyjnej. Jej wyniki służą dokonaniu wymienionych wcześniej rodzajów ocen.

Należy opracować diagram wymagań systemowych dla opisywanego systemu oceny poziomu szkolenia (rysunek 2.4).



Rysunek 2.4. Diagram wymagań oceny szkolenia

Ć W I C Z E N I E

2.6 System wykładów do wyboru

W ramach oceny istniejących rozwiązań informatycznych wydział dużej uczelni zdecydował się na całkowite zastąpienie dotychczas wykorzystywanego systemu obsługi zapisów na wykłady do wyboru (WdW). Działanie to podyktowane było możliwością zintegrowania rozwiązań dotychczas funkcjonujących niezależnie od siebie w ramach wspólnego portalu uczelnianego. Jednocześnie skorzystano z okazji do ujednolicenia stosowanych technologii.

Podstawowym problemem eksploatacyjnym systemu zapisów na wykłady do wyboru jest bardzo duże obciążenie systemu w momencie rozpoczęcia zapisów. Tym samym założenia projektu wdrożenia nowej edycji systemu obejmują wykorzystanie wydajnych technologii bazodanowych oraz infrastruktury serwerowo-sieciowej. Dodatkowo planuje się opracowanie zasad zapisów, wedle których studenci odmiennych trybów, poziomów, a w szczególnych przypadkach także specjalności studiów realizują zapisy w innych terminach. System musi także zapewniać bezpieczeństwo transmisji i przechowywania danych, cechować się układem graficznym oraz logotypami spójnymi z innymi aplikacjami webowymi będącymi w użytkowaniu wydziału.

Dostęp do najszerzej funkcjonalności otrzymuje wykładowca. Po zalogowaniu ma on dostęp do *listy wykładów oferowanych* w danym semestrze z podziałem na wykłady otwarte i takie, które z racji zbyt małej liczby chętnych są *nieuruchomione*. Każdy wykładowca ma *możliwość zarządzania listą wykładów* zgłoszonych przez siebie. Ponadto do każdego wykładu można *zamieszczać komentarze*. Istnieje również możliwość *podglądu listy studentów*, którzy zapisali się na wykład. Lista ta może być również *edytowana* poprzez dodawanie lub usuwanie studentów. W ramach tej funkcjonalności wyróżnia się *zarządzanie kontami*, co pozwala między innymi na resetowanie haseł studentom lub blokowanie ich kont. Aby zapewnić bezpieczeństwo transmisji danych, *dane przesyłane są w oparciu o SSL*. Z kolei *szyfrowanie wrażliwych danych* następuje z wykorzystaniem algorytmu RSA. Konta studentów mogą być *eksportowane* lub *importowane* w *formacie XML* pomiędzy różnymi systemami. Do tego wykorzystywany jest *szablon graficzny właściwy dla wydziału*. Istotą tego systemu jest *zgodność z kilkoma przeglądarkami*, czego efektem jest poprawne wyświetlanie w Microsoft Internet Explorer, Mozilla Firefox oraz Google Chrome.

W ramach *podglądu listy studentów* udostępniana jest funkcjonalność *sprawdzania listy obecności*. Na jej podstawie istnieje możliwość *generowania statystyk WdW* oraz *raportów w formacie PDF*. Wszelkie zmiany dokonane w systemie są przechowywane w postaci *historycznych WdW*. Istnieje możliwość *wyróżniania historycznych WdW* na liście wykładów. Wykłady zakończone są *automatycznie usuwane z bieżącej listy wykładów po upływie 12 miesięcy* od ich zakończenia.

Aby nie obciążać znacząco systemu, zdecydowano się na wymaganie funkcjonalne *zróznicowanie terminów wpisów*, na co składają się *zróznicowania dla odmiennych*:

- ❑ *specjalności*;
- ❑ *poziomów studiów*;
- ❑ *trybów studiów*.

Realizacja tych wymagań funkcjonalnych zmusza do uwzględnienia wymagań pozafunkcyjnych związanych z eksploatowaną infrastrukturą informatyczną. Zalicza się do nich przede wszystkim *zapewnienie wysokiej wydajności systemu*, na co składa się *wyбір wydajnej bazy danych MS SQL Server 2008 R2* oraz *wdrożenie wysokowydajnych technologii sieciowych*.

Na podstawie niniejszego opisu systemu zapisu na wykłady do wyboru należy przygotować diagram wymagań systemowych (rysunek 2.5).



Wskazówki dydaktyczne

Należy zwrócić uwagę na to, że nie wszystkie wymagania zilustrowane na diagramie muszą być obligatoryjnie powiązane różnymi odmianami związków. Wprowadzanie związków jest konsekwencją merytorycznych zależności pomiędzy kategoriami modelowania. I tak, zaprezentowane na rysunku 2.5 wymaganie *Zapewnienie wysokiej wydajności systemu* (wraz z jego wymaganiami zagnieżdżonymi) nie jest w sposób bezpośredni powiązane z żadnymi innymi wymaganiami na tym diagramie, mimo że wszystkie one są elementami składowymi tego samego pakietu.

Alternatywnym sposobem prezentowania związków pomiędzy wymaganiami jest zastosowanie konwencji notatkowej. Zostało to zobrazowane na rysunku 2.6.



Wskazówki dydaktyczne

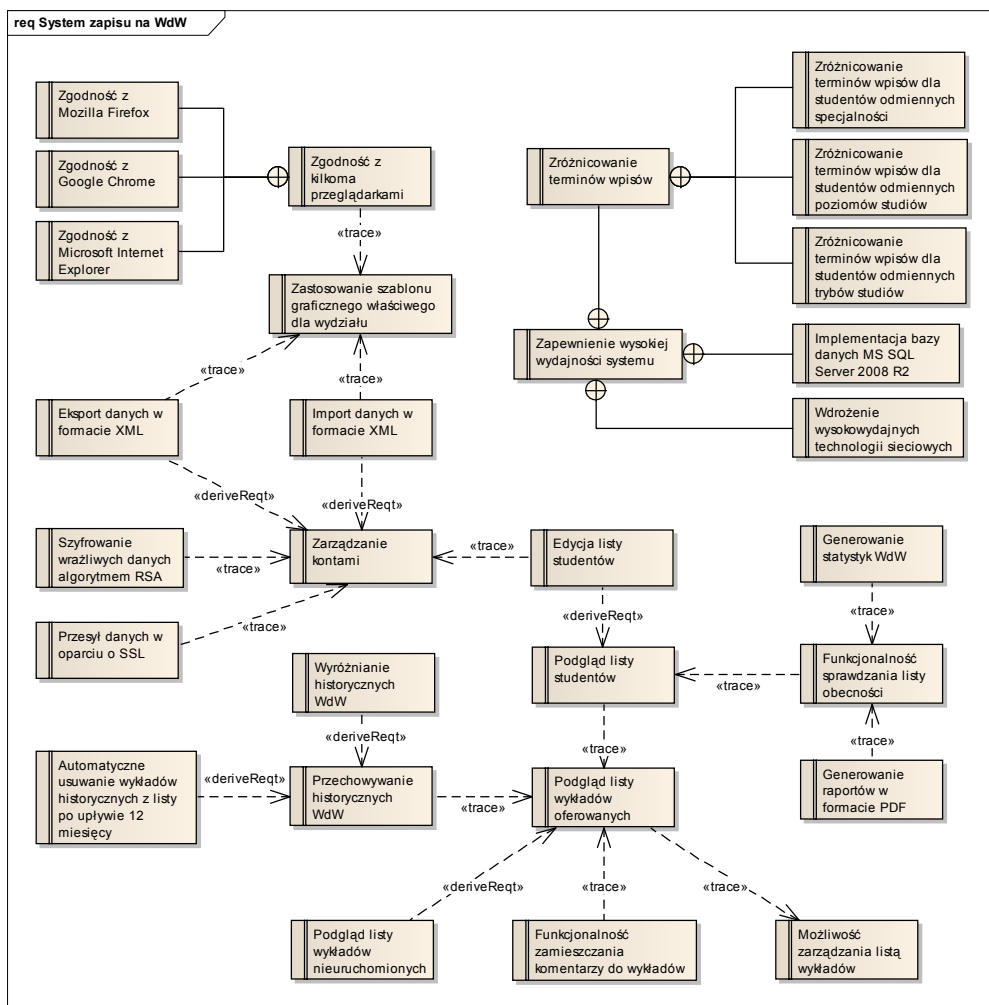
Konwencja notatkowa stanowi jedną z form wizualizacji związków pomiędzy wymaganiami. W konwencji tej każdemu powiązanemu wymaganiu systemowemu przypisuje się notatkę zawierającą słowo/słowa kluczowe wskazujące na rodzaj związku (np. *Derived*, *SatisfiedBy* czy też *Master*) oraz nazwę powiązanej kategorii modelowania. Z uwagi na to, że w praktyce na diagramach wymagań systemowych (poza samymi wymaganiami) wprowadzać można szereg innych kategorii modelowania, każda taka kategoria powinna mieć przypisany stereotyp tekstowy wskazujący na jej rodzaj. Jeśli z danym wymaganiem powiązanych jest kilka różnych kategorii modelowania, można zarówno:

- ♦ zastosować pojedynczą notatkę ujmującą listę wszystkich elementów powiązanych (tzw. konwencja notatkowa zagregowana);

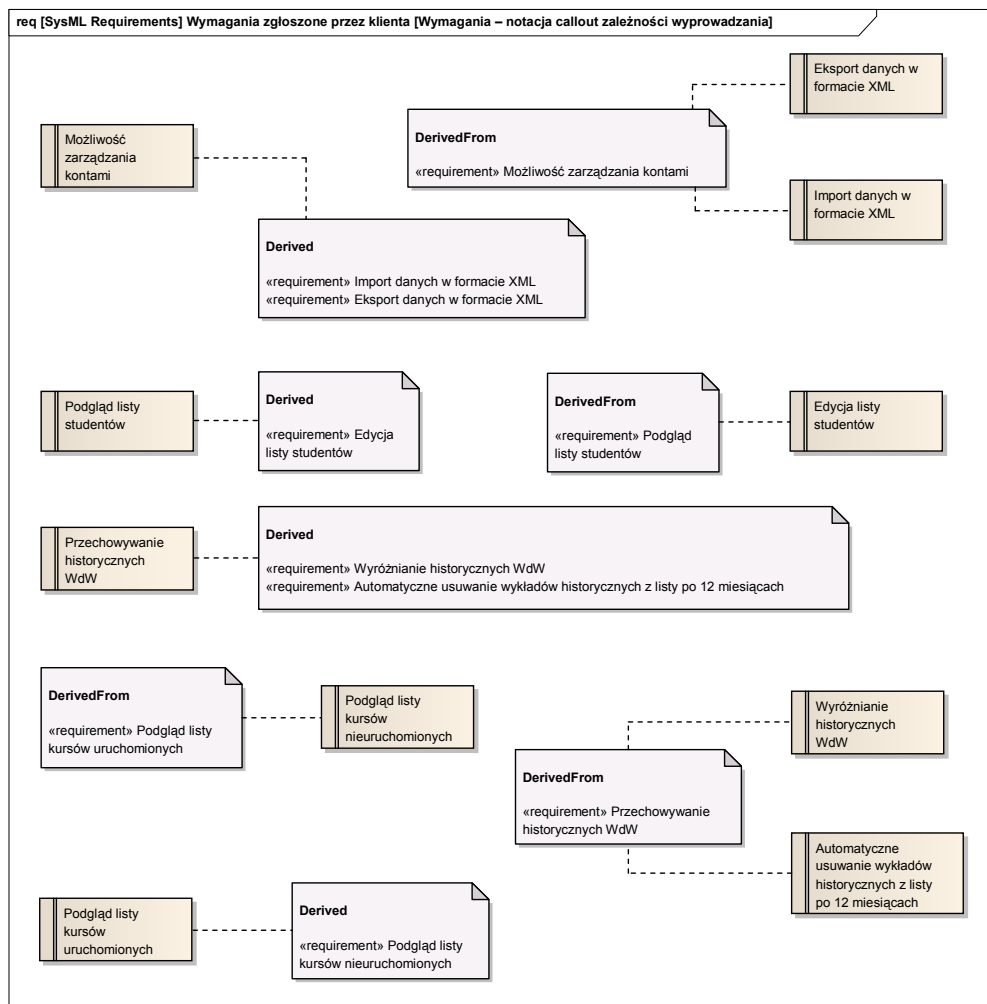
jak również

- ♦ przypisywać osobne notatki, z których każda wskazuje na odrębną powiązaną kategorię modelowania.

Opisana forma dokumentowania związków jest równoważna konwencji bezpośredniej, w której związki są szkicowane na diagramie w standardowy sposób, i sprawdza się zwłaszcza w przypadku relatywnie nieskomplikowanych diagramów oraz używania uproszczonych narzędzi modelowania niedysponujących pełną paletą związków.



Rysunek 2.5. Diagram wymagań systemowych dla systemu zapisów na wykłady do wyboru



Rysunek 2.6. Diagram wymagań systemowych w notacji callout dla systemu zapisów na wykłady do wyboru

Ć W I C Z E N I E

2.7**Dozór elektroniczny**

Centralny urząd państwowy ogłasza przetarg na wykonanie dozoru elektronicznego w kierowanych przez siebie placówkach. Państwowy system dozoru elektronicznego powinien obejmować następujące moduły:

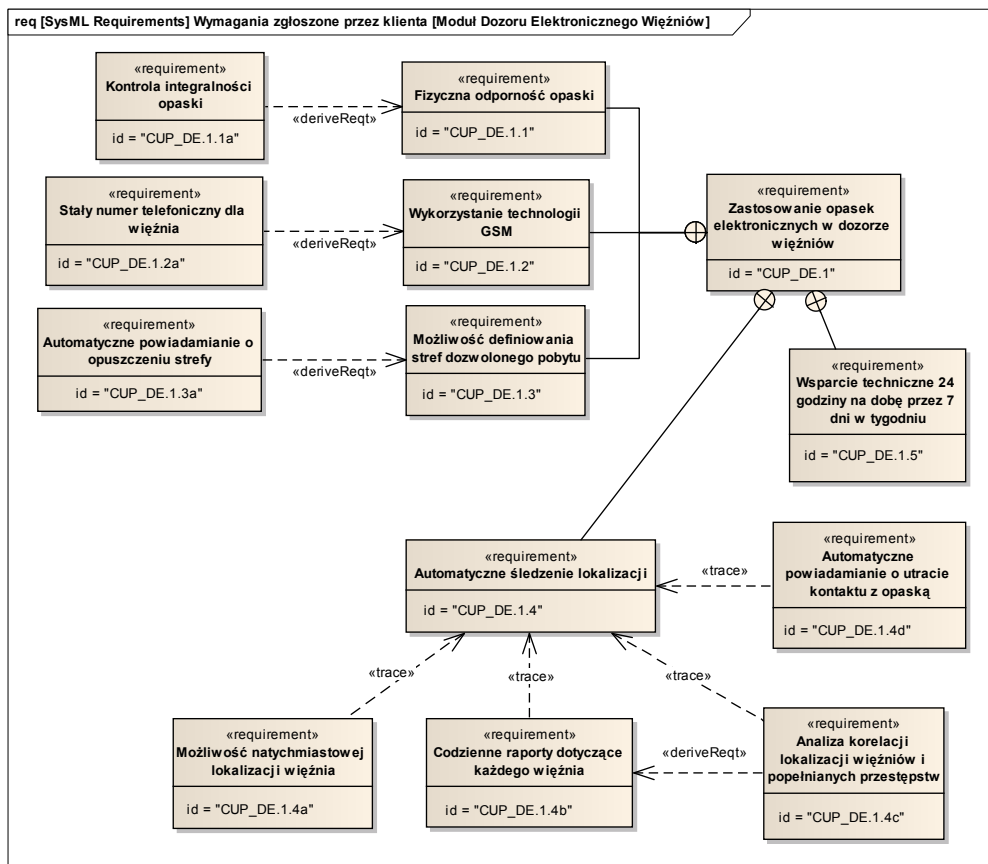
- ☐ opaskę elektroniczną w placówkach więziennych;
- ☐ rozpoznawanie twarzy na stadionach piłkarskich;
- ☐ analizę serwisów społecznościowych oraz ruchu w sieci internetowej dla stworzenia map społecznościowych;
- ☐ monitoring miasta;
- ☐ analizę bilingów rozmów przestępców w sieciach telefonicznych i mobilnych;
- ☐ analizę korespondencji elektronicznej, w tym SMS-ów i e-maili;
- ☐ przechowywanie i identyfikację DNA, siatkówki i tęczówki oka, odcisków palców i dłoni;
- ☐ monitorowanie z powietrza bezzałogowym samolotem wyposażonym w kamerę;
- ☐ wykrywanie wzorców zachowań poprzez eksplorację danych w bazach danych osobowych oraz rejestrach administracyjnych;
- ☐ monitorowanie zachowania osób i grup, działów pracujących w centralnym urzędzie państwowym;
- ☐ wykrywanie zagrożeń terrorystycznych;
- ☐ satelity i specjalistyczne samoloty posiadające sensory wykrywające zagrożenia chemiczne, jądrowe, podziemne instalacje budowy broni jądrowej i chemicznej;
- ☐ wykrywanie podsłuchów;
- ☐ inne wynikające z aktualnego rozwoju technologii.

Dla każdego z wyżej wymienionych modułów systemu przygotowano stosowne specyfikacje wymagań systemowych. W odniesieniu do dozoru elektronicznego więźniów z *Zastosowaniem opasek elektronicznych* zleceniodawca zgłosił następujące wymagania:

- ☐ *Kontrola integralności opaski;*
- ☐ *Fizyczna odporność opaski;*
- ☐ *Stały numer telefoniczny dla więźnia;*
- ☐ *Wykorzystanie technologii GSM;*
- ☐ *Automatyczne śledzenie lokalizacji;*
- ☐ *Analiza korelacji lokalizacji więźniów i popełnianych przestępstw;*
- ☐ *Wsparcie techniczne 24 godziny przez 7 dni w tygodniu;*
- ☐ *Codziennie raporty dotyczące każdego więźnia;*

- ❑ Możliwość natychmiastowej lokalizacji więźnia;
- ❑ Możliwość definiowania stref dozwolonego pobytu;
- ❑ Automatyczne powiadamianie o opuszczeniu strefy;
- ❑ Automatyczne powiadamianie o utracie kontaktu z opaską.

Należy opracować diagram wymagań systemowych dla opisanego modułu (rysunek 2.7).



Rysunek 2.7. Diagram wymagań systemowych dla modułu dozoru elektronicznego więźniów



Wskazówki dydaktyczne

Identyfikatory wymagań systemowych umożliwiają porządkowanie rozbudowanego repozytorium wymagań. Oficjalna dokumentacja nie narzuca składni identyfikatorów wymagań systemowych. Tym samym kwestia ta pozostaje w gestii zespołu opracowującego rozwiązanie informatyczne. W tym konkretnym przypadku zdecydowano się na

wypracowanie rozwiązania, w którym każdemu wymaganiu przypisano przedrostki wskazujące na zleceniodawcę (CUP), kodową nazwę projektu (DE). Numeracja zawarta w poszczególnych wymaganiach wskazuje na poziom szczegółowości. I tak, wymaganie nadrzędne ma przypisany nr 1, podczas gdy jego wymagania składowe — 1.1, 1.2, 1.3 itd. Z kolei wymagania zależne przyjmują numerację wymagania źródłowego oraz końcówkę — a, b, c itd.

2.2. Zadania

Zadanie 2.1.

Na podstawie rysunku 2.2 przedstaw zależność pomiędzy wymaganiem źródłowym *Bieżące utrzymywanie bazy zgłoszonych prac* a wymaganiami docelowymi *Bieżące utrzymywanie bazy referatów*, *Bieżące utrzymywanie bazy plakatów* oraz *Bieżące utrzymywanie bazy autoreferatów doktoranckich*. Jaki rodzaj zależności łączy wymaganie źródłowe z wymaganiami docelowymi?

Zadanie 2.2.

Na podstawie rysunku 2.2 skonkretyzuj zależność pomiędzy *wprowadzaniem aktualizacji recenzji* a *rejestrowaniem uwag do członków Komitetu Programowego*. Treść referatu i treść recenzji wykonanej przez recenzenta widoczne są dla recenzentów — liderów ścieżek i przewodniczącego Komitetu Programowego, którzy podejmują ostateczne decyzje o przyjęciu lub odrzuceniu referatu.

Zadanie 2.3.

Użytkownik zgłosił szereg nowych wymagań systemowych. Podstawową wymaganą funkcjonalnością jest to, aby system nadzorował, czy wymagana i wcześniej sprecyzowana liczba recenzji została nadesłana w określonym terminie. Zakłada się, że w modelowanym systemie niezbędna liczba recenzji dla danego referatu wynosi trzy. Jeżeli liczba recenzji nie jest kompletna, wysyła się ponaglenia. W przypadku gdy recenzent nie reaguje na ponaglenia, wyznaczany jest nowy recenzent. Uzupełnij wskazaną część diagramu zaprezentowanego na rysunku 2.2 o nowe wymagania i określ rodzaje zależności występujących pomiędzy nimi.

Zadanie 2.4.

Przedstaw i zinterpretuj tabelarycznie zależności pomiędzy wymaganiami: rodzaj zależności, stereotyp, interpretacja, wymaganie źródłowe i docelowe oraz docelowa kategoria modelowania.

Zadanie 2.5.

Zapoznaj się z funkcjonalnością Systemu obsługi międzynarodowej konferencji EasyChair (<http://www.easychair.org>) i przedstaw ją na diagramie wymagań. Wymagania funkcjonalne przedstaw jedynie dla modułu rejestrowania autorów referatów.

Zadanie 2.6.

Wysokość opłaty za szkolenie jest uzależniona od terminu dokonania wpłaty. Najniższa opłata jest dokonywana przez uczestników, którzy rejestrują się najwcześniej, tzw. *early birds registration*. Wyższe płatności dotyczą uczestników rejestrujących się w regulaminowym terminie, a najwyższe dotyczą tych, którzy rejestrują się już na samym szkoleniu. Rozwiń zaproponowane na rysunku 2.3 płatności, uwzględniając w nich zależności pomiędzy terminem rejestracji i dokonaniem płatności za szkolenie.

Zadanie 2.7.

Zaprezentowane na rysunku 2.2 wymagania przedstaw w pełnej postaci zawierającej numery porządkowe, nazwy oraz treść poszczególnych wymagań określonych na rysunku nazwami.



3

Diagramy przypadków użycia

Ć W I C Z E N I E

3.1 Sklep internetowy

System sklepu internetowego obsługują dwaj aktorzy: *Klient* i *Administrator*. Funkcjonalność sklepu służy potrzebom i zamówieniom *Klienta*. Podstawową czynnością wykonywaną przez *Klienta* jest *Wybierz produkt*. Usługa ta polega na prezentowaniu podzielonej na grupy i podgrupy listy produktów, które można zakupić w sklepie. Po kliknięciu nazwy produktu z listy *Klient* ma możliwość zapoznania się z jego opisem (nazwa, zdjęcie, specyfikacja techniczna, cena) i kliknięcia opcji dodaj do koszyka. Jest to jedna z opcji składających się na przypadek użycia *Wybierz produkt*. Z kolei jeżeli *Klient* zażyczy sobie zawężenia listy istniejących produktów pod względem podanych kryteriów, wybiera opcję *Wyszukaj produkt*. Wybrane produkty umieszczane są w koszyku. *Klient* ma możliwość przeglądania zawartości koszyka, zmieniania ilości zawartych w nim produktów oraz usuwania poszczególnych pozycji z koszyka. Wykorzystuje w tym celu przypadek użycia *Zarządzaj koszykiem*.

Przejdzie do składania zamówienia uwarunkowane jest wykonaniem przypadku użycia *Autoryzuj użytkownika*. Po autoryzacji klient może przejść do składania zamówienia, którego funkcjonalność zawiera się w przypadku użycia *Złóż zamówienie*. Zamawiający system sprecyzował, że opcja składania zamówienia ma być inicjowana z poziomu koszyka. W ramach składania zamówienia *Klient* zobowiązany jest wybrać formę płatności. Z perspektywy sklepu wyróżnić można dwa podstawowe mechanizmy płatności:

- ❑ płatność online, którą *Klient* może wykonać bezpośrednio po złożeniu zamówienia w konsekwencji zainicjowania przypadku użycia *Realizuj płatność online*;
- ❑ płatność za pobraniem, której zaksięgowanie leży w gestii *Administratora* systemu, poprzez przypadek użycia *Zaksięguj płatność za pobraniem*.

Płatność online można zrealizować na trzy sposoby. Stąd przewidziano odpowiednio trzy szczegółowe przypadki użycia w tym zakresie:

- ❑ *Realizuj płatność kartą kredytową;*
- ❑ *Realizuj płatność w systemie PayPal;*
- ❑ *Realizuj płatność za pomocą usługi ePrzelew.*

Płatności te trafiają bezpośrednio do jednego z trzech aktorów odpowiedzialnych za fizyczną realizację transakcji. Są to odpowiednio:

- ❑ *System operatora kart;*
- ❑ *System PayPal;*
- ❑ *Operator usługi ePrzelew.*

Z kolei *Administrator* systemu odpowiedzialny jest za funkcjonalności wyrażone przypadkami użycia *Zarządzaj produktami*, *Zarządzaj kontami użytkowników*, *Przeglądaj listę zawartych transakcji*, *Zmień statusy przesyłek* oraz wcześniej wspomnianą *Zaksięguj płatność za pobraniem*. Dwa pierwsze z wymienionych przypadków użycia są typu CRUD (*Create*, *Read*, *Update*, *Delete*). Przykładowo inicjując przypadek użycia *Zarządzaj produktami*, *Administrator* może dodać nowy produkt (*C*), przeglądać listę produktów znajdujących się w sprzedaży (*R*), aktualizować dane poszczególnych produktów (*U*) oraz usuwać wybrane produkty (*D*).

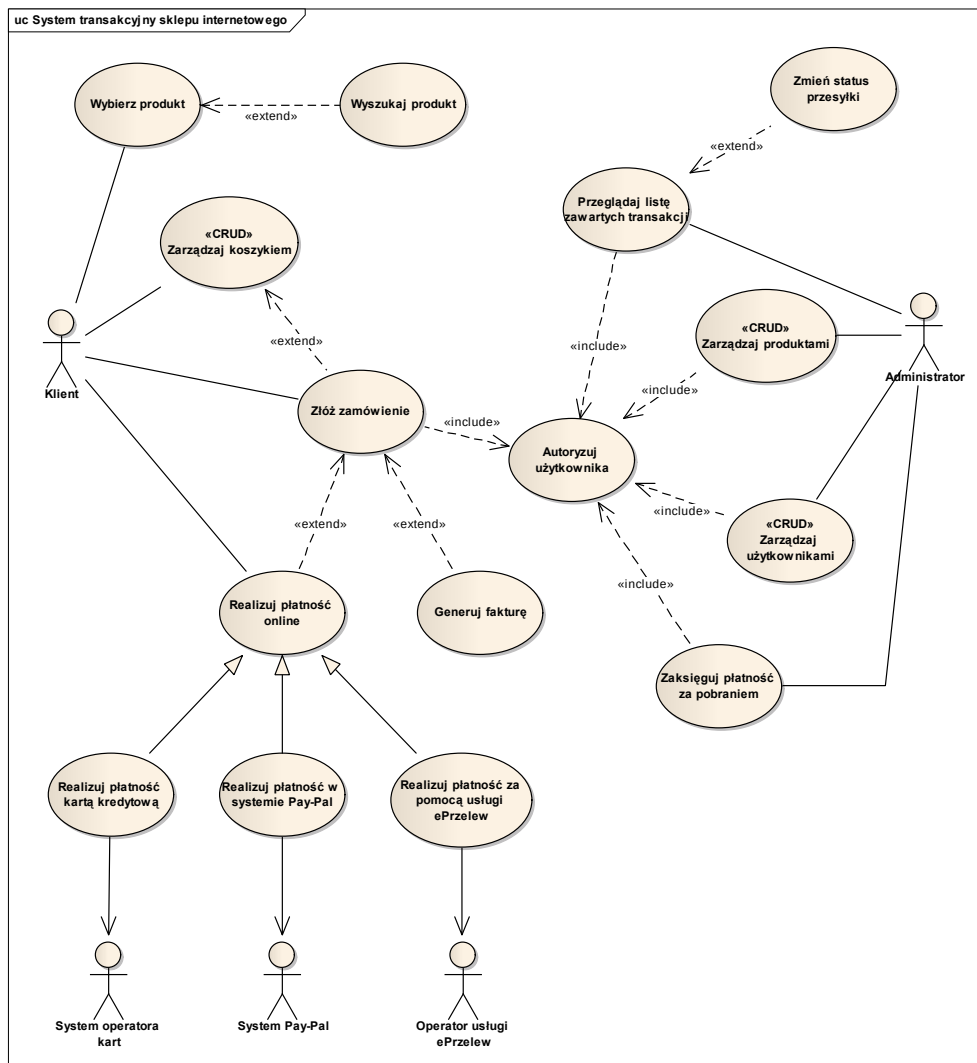
Na podstawie powyższego opisu należy utworzyć diagram przypadków użycia prezentujący funkcjonalność sklepu internetowego (rysunek 3.1).

Ć W I C Z E N I E

3.2 Dokumentacja przypadku użycia *Zarządzaj koszykiem*

W ramach opracowywania modelu przypadków użycia przygotowanie półformalnej, tabelarycznej dokumentacji poszczególnych przypadków użycia zlecono grupie analityków systemowych. Analityk odpowiedzialny za przypadek użycia *Zarządzaj koszykiem* sklasyfikował niniejszy przypadek jako niezbędny ze względu na techniczną realizację systemu i zarazem ważny dla klienta.

Standardowy scenariusz realizacji przypadku prowadzi do złożenia zamówienia i zawiera wyświetlenie zawartości koszyka. Klient ma jednak możliwość modyfikacji zarówno listy produktów w koszyku, jak również liczebności poszczególnych pozycji produktowych. W każdym przypadku system aktualizuje sumaryczną cenę wybranego produktu oraz wartość całego zamówienia. W przypadku opuszczenia strony zawartość koszyka jest przechowywana przez 24 godziny i koszyk jest przywracany przy każdym nowym wejściu na stronę przed upływem tego czasu. Klient zwrócił uwagę na konieczność zapewnienia krótkiego czasu reakcji systemu, co należy tym samym odzwierciedlić w dokumentacji.



Rysunek 3.1. Diagram przypadków użycia dla sklepu internetowego

Na podstawie powyższego opisu należy opracować dokumentację dla przypadku użycia *Zarządza koszykiem* (tabela 3.1).

Tabela 3.1. Dokumentacja przypadku użycia Zarządzaj koszykiem

Nazwa:	Zarządzaj koszykiem
Numer:	S2.1
Twórca:	Analitik systemowy Krzysztof Zaremba
Poziom ważności:	wysoki
Typ przypadku użycia:	ogólny, niezbędny
Aktorzy:	Klient
Krótki opis:	Klient ma możliwość przeglądania listy produktów i kwoty łącznej do zapłaty, usuwania produktów, zmiany liczby zakupionych produktów
Warunki wstępne:	Musi być wybrany co najmniej jeden produkt
Warunki końcowe:	Przejsie do składania zamówienia
Główny przepływ zdarzeń:	Klient wybiera opcję <i>Koszyk zakupów</i> System wyświetla listę uprzednio wybranych produktów Klient wybiera opcję <i>Złóż zamówienie</i>
Alternatywne przepływy zdarzeń:	2a. System wyświetla informację <i>Brak produktów w koszyku zakupów</i> 3a. Klient zmienia liczbę wybranych produktów poprzez kliknięcie na przyciskach zmniejsz oraz zwiększ liczbę produktów 3a.1. System aktualizuje łączną cenę wybranych produktów oraz wartość całego zamówienia 3b. Klient wybiera opcję usunięcia produktu 3b.1. System aktualizuje listę produktów oraz wartość całego zamówienia
Specjalne wymagania:	System ma działać w czasie rzeczywistym, przy czym maksymalny czas odpowiedzi powinien wynosić nie więcej niż 5 sekund
Notatki i kwestie:	W przypadku opuszczenia strony zawartość koszyka jest przechowywana przez 24 godziny i koszyk zostaje przywrócony przy każdym nowym wejściu na stronę przed upływem tego czasu



Ć W I C Z E N I E

3.3 Planowanie akcji marketingowej z wykorzystaniem systemu CRM

Jednym z zadań systemu CRM jest planowanie akcji marketingowych określonego produktu lub usługi. W planowaniu tym uczestniczą następujący aktorzy:

- ☐ *Menedżer akcji marketingowej;*
- ☐ *Kierownik działu finansowego;*
- ☐ *Pracownik działu marketingu;*
- ☐ *Administrator systemu CRM (informatyk).*

Aktorzy ci występują również jako aktor osobowy — *Zespół marketingowy*, który wykonuje następujące funkcje — przypadki użycia:

- ☐ *Zidentyfikuj cele akcji;*
- ☐ *Sprecyzuj rynek docelowy akcji;*
- ☐ *Ustal plan działań;*
- ☐ *Wyznacz pracowników odpowiedzialnych za poszczególne zadania;*
- ☐ *Opracuj harmonogram akcji.*

Przyjęto adekwatne odpowiedzialności poszczególnych członków *Zespołu*. *Menedżer akcji marketingowej* jest głównym aktorem przypadku użycia *Zarządzaj akcją*. Nad budżetem przedsięwzięcia sprawuje nadzór *Kierownik działu finansowego*. Aktor ten użytkuje przypadki użycia *Monitoruj budżet* oraz *Zaksięguj koszty*. Merytorycznym wykonawcą akcji jest *Pracownik działu marketingu* — przypadek użycia *Realizuj akcję*. Wreszcie *Informatyk* — *administrator systemu CRM* wspomaga cały *Zespół* poprzez przypadek użycia *Komputerowo obsługuj akcję marketingową*.

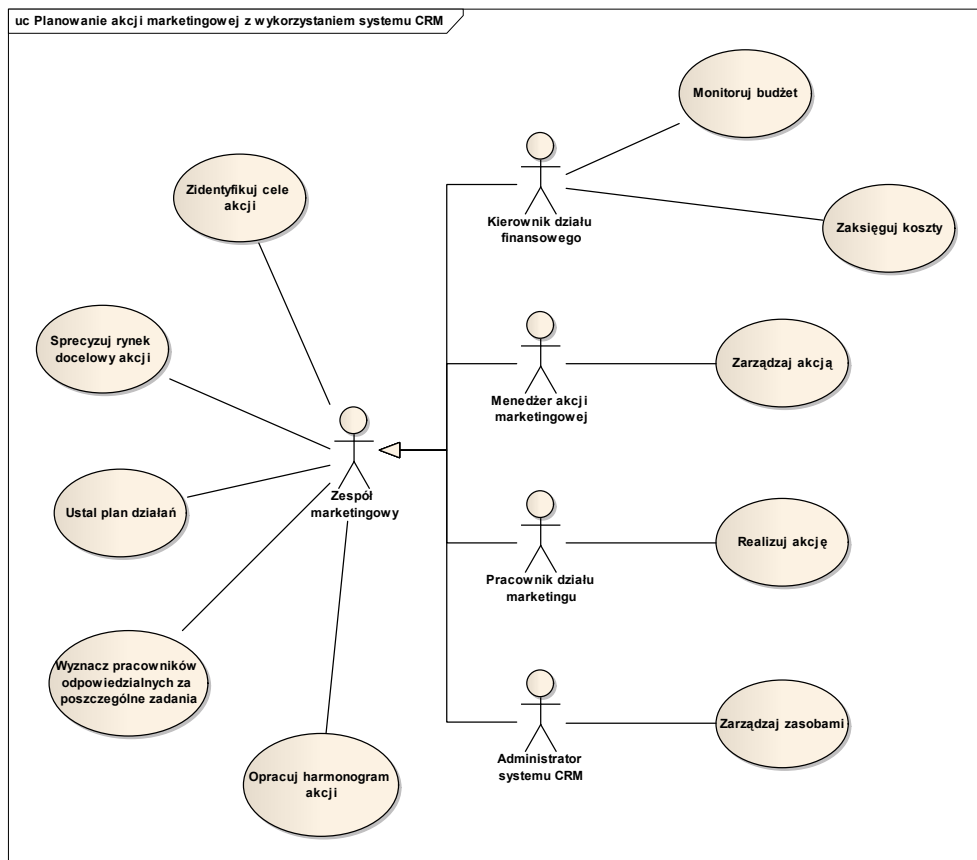
Należy opracować diagram przypadków użycia prezentujący możliwe wykorzystanie systemu CRM w zakresie planowania akcji marketingowej (rysunek 3.2).

Ć W I C Z E N I E

3.4 Funkcjonalność systemu CRM

Funkcjonalność rozwiązania klasy CRM przygotowywana pod kątem potrzeb konkretnego zamawiającego dedykowana została w pierwszym rzędzie następującym grupom użytkowników:

- ☐ *Pracownikom działu sprzedaży;*
- ☐ *Pracownikom działu marketingu;*
- ☐ *Analitykom.*



Rysunek 3.2. Diagram przypadków użycia dla akcji marketingowej z wykorzystaniem systemu CRM

Z uwagi na konieczność zróżnicowania uprawnień dla poszczególnych usług w systemie w każdej z wyżej wymienionych grup wyszczególnić należy podgrupy menedżerskie. Dodatkowo zamawiający zdecydował się na udostępnienie *Klientom* dostępu do informacji o nich gromadzonych.

Do zadań wchodzących w zakres odpowiedzialności pracowników działu sprzedaży zaliczyć należy *Zarządzanie możliwościami śledzenia transakcji*, *Zarządzanie relacjami z partnerami handlowymi*, *Zarządzanie kontaktami z klientem*, *Profilowanie produktów*, *Konfigurowanie sprzedaży* oraz *Przeprowadzanie sprzedaży przez Internet*. Z kolei *Zarządzanie pracą poszczególnych handlowców* stanowi obowiązek *Menedżera działu sprzedaży*. W systemie przewidziano funkcjonalność wspierającą to zadanie.

Pracownicy działu marketingu będą mieli możliwość wykorzystania wdrażanego rozwiązania w zakresie *Zarządzania namiarami (kontaktami)*, *Zarządzania kampaniami promocyjnymi*, *Zarządzania zasobami marketingowymi*, *Zarządzania reklamacjami*, *Profilowania klienta* oraz *Personalizowania profilu klienta*. Dla menedżerów tego działu przewidziano funkcjonalność *Zarządzania Call Center*, *Zarządzania czasem pracy*, jak również *Zarządzania wielokanałowym centrum kontaktów*.

Analitycy mogą wykorzystywać system w zakresie wspomagania *Przeprowadzania badań i analiz marketingowych*, *Analiz koszykowych*, *Analiz wykorzystania kanałów elektronicznych* czy też *Zarządzania telemarketingiem*. *Analitycy o uprawnieniach menedżerskich* mogą *Analizować sprzedaż*.

Dla powyższego opisu należy utworzyć diagram przypadków użycia prezentujący funkcjonalność systemu CRM (rysunek 3.3).

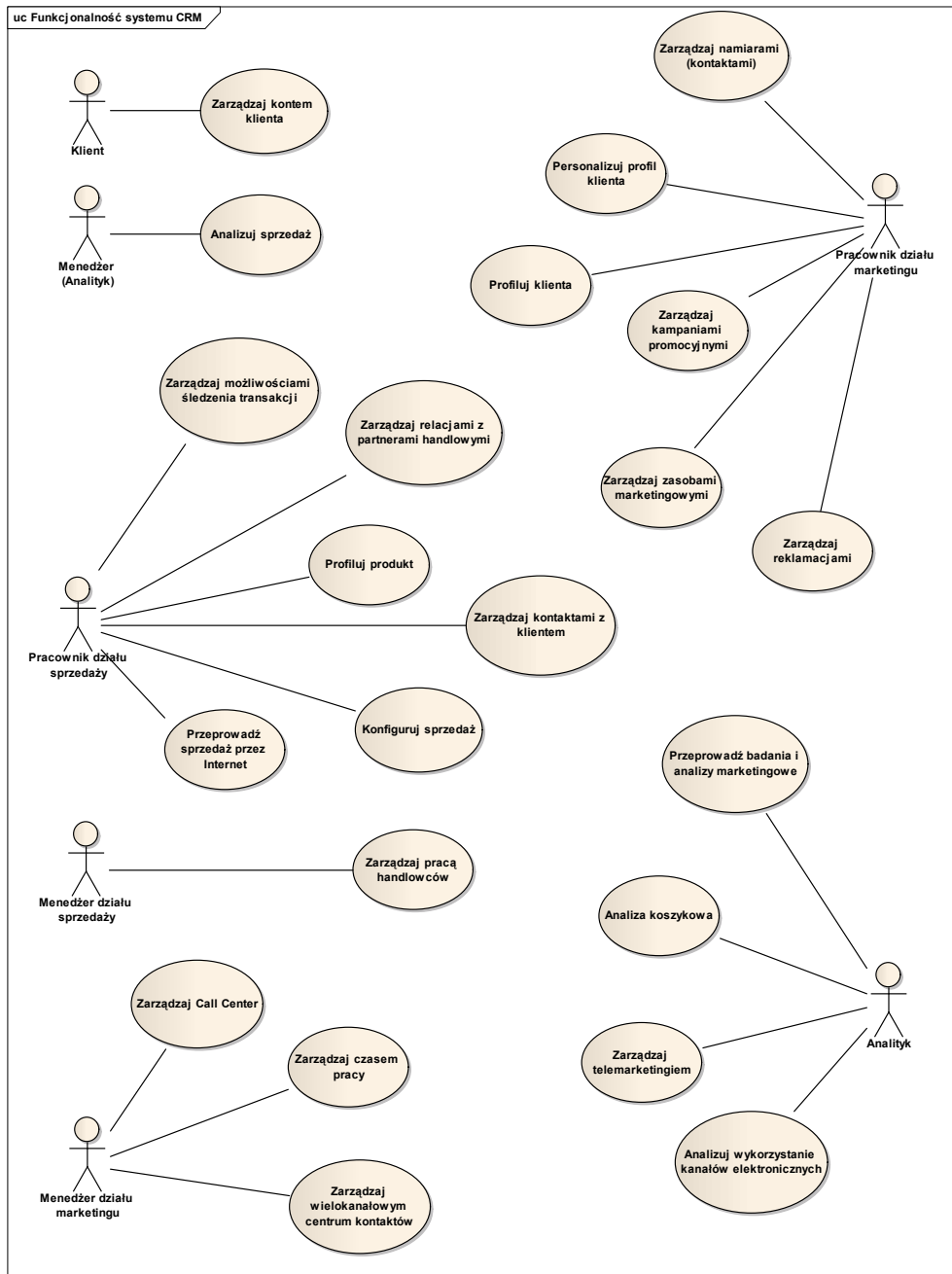
Ć W I C Z E N I E

3.5 System monitorowania ruchu w mieście

Wygodny dla kierowców ruch samochodów we współczesnym mieście stwarza wiele problemów. Chodzi tu przede wszystkim o kwestie ekologiczne i logistyczne. Ruch samochodów w samym mieście jest przyczyną zakorkowywania się miast, zwłaszcza w godzinach szczytu, co skutkuje wzrostem zanieczyszczenia powietrza, hałasu oraz bardzo poważnymi problemami z parkowaniem. Rozbudowa infrastruktury drogowej w wielu miastach osiągnęła poziom krytyczny. Stąd mieszkańcy i władze miasta zainteresowane są zmniejszaniem ruchu samochodowego na rzecz wykorzystania transportu publicznego.

Władze miasta zdecydowały się na wdrożenie systemu informatycznego, który będzie monitorował ruch poszczególnych pojazdów i generował opłaty dla kierowców za wjazd do centralnej części miasta. Wiąże się to z koniecznością zainstalowania dostarczonego przez miasto chipu na przedniej tablicy rejestracyjnej pojazdu. Wymóg ten dotyczy jedynie osób zameldowanych w mieście. Odpowiedzialność za użytkowanie systemu ponosi grupa operatorów będących pracownikami działu obsługi systemu kontroli ruchu. Wśród nich:

- ❑ *Menedżer* — posiadający dostęp do modułów raportowania, *raportuje ruch drogowy* z możliwością *raportowania w okresie rozliczeniowym*;
- ❑ *Administrator systemu* — odpowiedzialny za *zarządzanie bazą pojazdów* oraz *parametryzację*, w tym wprowadzanie stawek godzinowych i zarządzanie użytkownikami;
- ❑ *Kontroler identyfikacji pojazdów* — odpowiadający za *prowadzenie wizualnej identyfikacji tablic pojazdów*;
- ❑ *Referent finansowy* — odpowiedzialny za *generowanie i zatwierdzanie faktur*;
- ❑ *Referent raportów windykacji* — opracowujący *zestawienia właścicieli pojazdów zalegających z opłatami*.



Rysunek 3.3. Diagram przypadków użycia funkcjonalności systemu CRM

Na drogach wjazdowych do centrum miasta ustawiono zestawy składające się z czujnika — *Laserowego detektora ruchu*, kamer, radiowego czytnika impulsów generowanych przez chipy wbudowane w tablice rejestracyjne samochodów. W momencie *wykrycia ruchu pojazdu* przez dowolny detektor następuje próba *przeprowadzenia automatycznej identyfikacji pojazdu* po impulsie wygenerowanym przez wbudowany chip. W przypadku braku takiego impulsu następuje wykonanie pary fotografii — tablic rejestracyjnych z przodu i z tyłu tego samochodu.

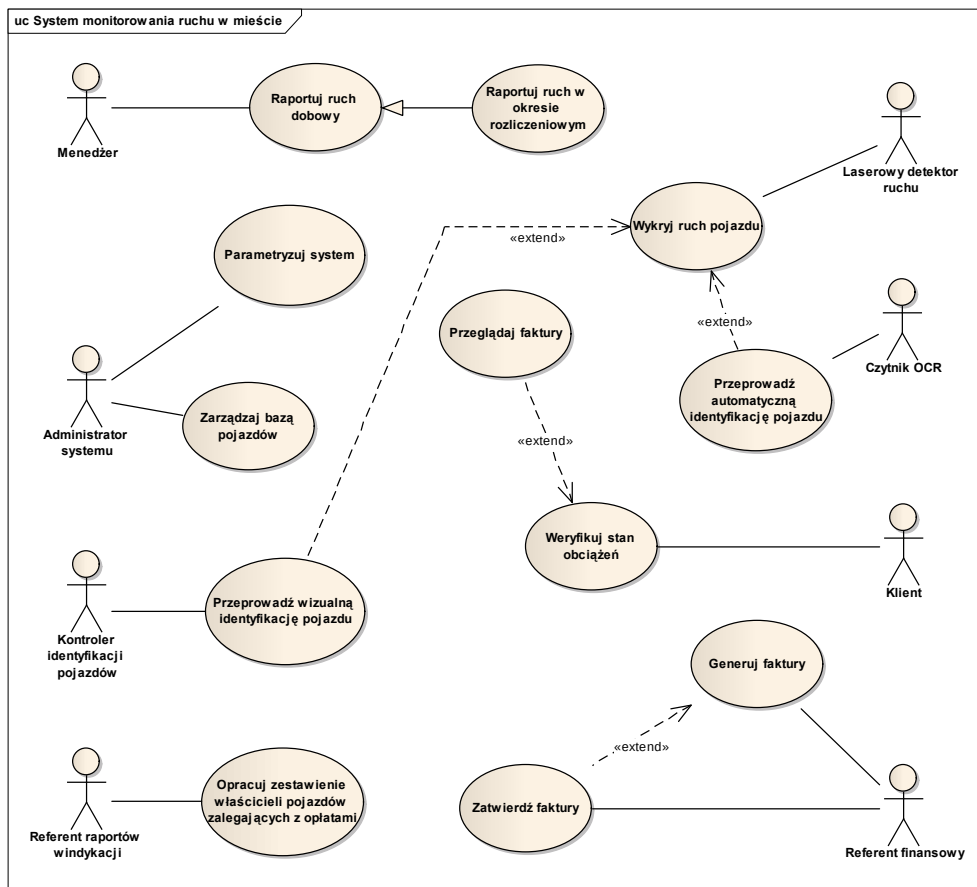
Dane zbierane przez zestawy ustawione na drogach wjazdowych do miasta przekazywane są do miejskiego centrum monitorowania ruchu. W przypadku gdy w pojeździe nie jest zainstalowany chip, wykonane przez kamerę zdjęcie jest przekazywane do centrum, gdzie za pomocą *Czytnika OCR (Optical Character Recognition)* pojazd jest identyfikowany. Jeżeli wynik otrzymany dla tablicy przedniej różni się od wyniku dla tablicy tylnej (co może być spowodowane np. zabrudzeniem czy niską jakością zdjęcia), zdarzenie trafia na pulpit *Kontrolera identyfikacji pojazdów*, który dokonuje *wizualnej identyfikacji pojazdu*. Dotyczy to następujących danych: godziny wjazdu lub wyjazdu oraz danych pojazdu. W przypadku jednoznacznej identyfikacji pojazdu dzięki otrzymanemu impulsowi następuje automatyczne naliczenie opłaty. Należy podkreślić, że opłaty nie są pobierane od samochodów z rejestracjami zagranicznymi. Opłaty są zróżnicowane w zależności od dnia tygodnia, godziny wjazdu oraz czasu przebywania w centrum miasta. System *generuje raz w miesiącu faktury*, do których dostęp jest możliwy poprzez stronę WWW. Zaległości z tytułu tych opłat obciążone są opłatą karną. Użytkownik ma również możliwość *przeglądania online aktualnego stanu obciążeń* z tytułu przejazdu i parkowania w mieście.

System zachęca kierowców samochodów osobowych do korzystania z transportu publicznego, podnosząc jednocześnie jakość życia mieszkańców. Na podstawie powyższego opisu należy przygotować diagram przypadków użycia prezentujący funkcjonalność systemu monitorowania ruchu w mieście (rysunek 3.4).

Ć W I C Z E N I E

3.6 System rezerwacji hotelowych

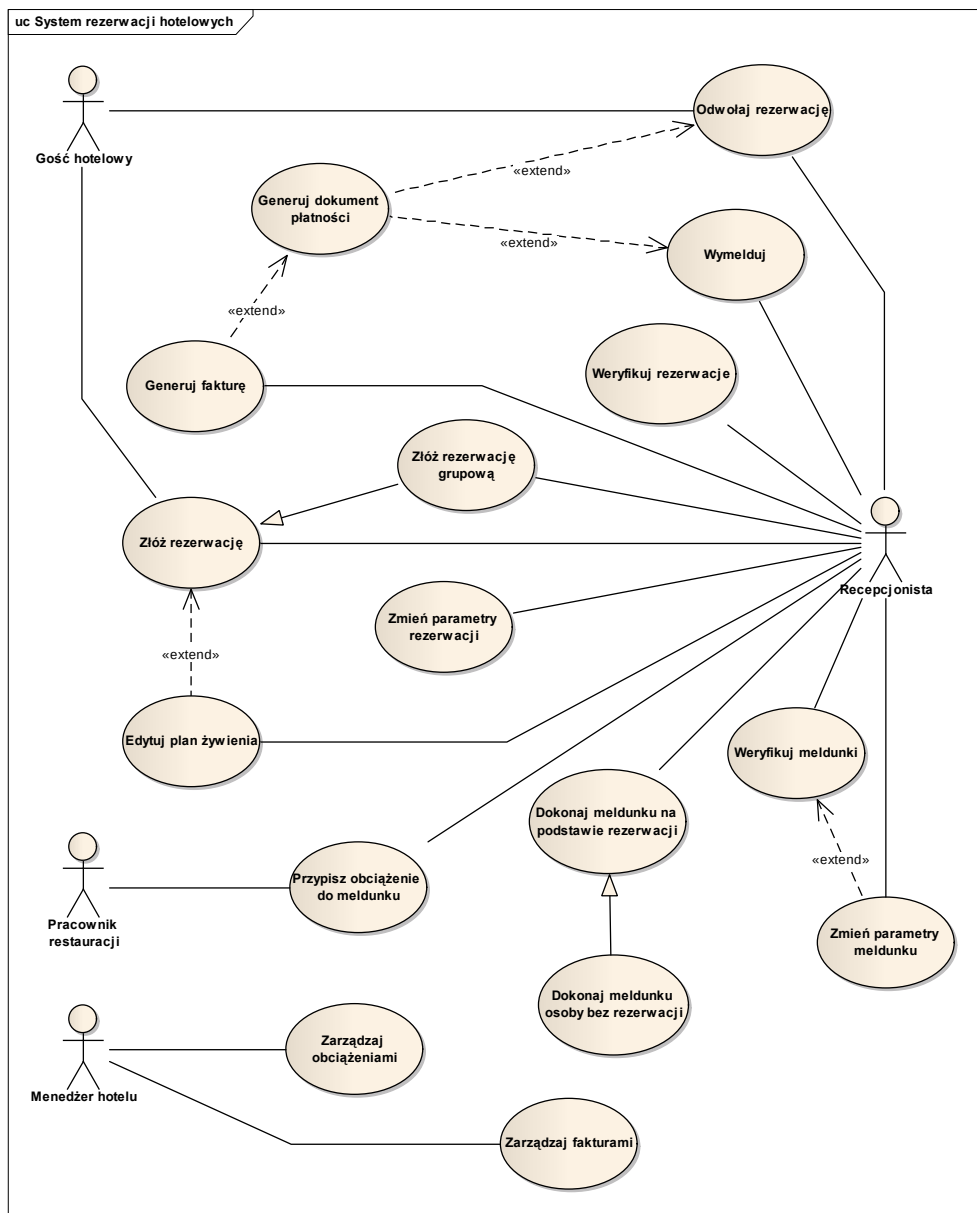
Współczesne hotele wspomagają swoją działalność rozbudowanymi systemami obsługi gości hotelowych, zarządzania oraz finansowo-księgowymi. Użytkownikami systemu rezerwacji hotelowych są *goście hotelowi, recepcjoniści, pracownicy restauracji oraz menedżerowie hotelu*. *Gość hotelowy* oczekuje realizacji następujących funkcjonalności przyjmujących na diagramie postać przypadków użycia: *złóż rezerwację* i *odwołaj rezerwację*. Szczególnym przypadkiem pierwszej z wymienionych funkcjonalności jest *złożenie rezerwacji grupowej*. W razie konieczności gość hotelowy może zaproponować dostosowanie do indywidualnych potrzeb standardowej oferty gastronomicznej (przypadek użycia *edytuj plan żywienia*). Dowolne przypadki użycia *gościa hotelowego* mogą być wykonywane przez podstawową



Rysunek 3.4. Diagram przypadków użycia systemu monitorowania ruchu w mieście

grupę użytkowników tego systemu, którymi są *recepjoniści*. Mają oni dostęp do wszystkich głównych funkcjonalności, a przede wszystkim *przyjmowania rezerwacji*, *weryfikowania rezerwacji*, *dokonywania meldunku* (zarówno na podstawie rezerwacji, jak również bez rezerwacji), *wymeldowywania*, *odwołania rezerwacji*, a także *zmiany parametrów rezerwacji* i *zmiany parametrów meldunku*. Do obowiązków *recepjonistów* należy również *generowanie faktury* i związanego z nią *dokumentu płatności*. Należności wynikające z korzystania z posiłków w restauracji przez gości hotelowego *przypisywane są jako obciążenia do meldunku* przez *pracowników restauracji*. Tym samym są one automatycznie uwzględniane na ostatecznym rachunku dla klienta. *Menedżer hotelu* jest odpowiedzialny za *zarządzanie fakturami*, na co składa się ich *edytowanie*, *zatwierdzanie*, *anulowanie* czy *drukowanie*, oraz *zarządza obciążeniami apartamentów i pokoi*, jeśli zostały one w sposób nieprawidłowy przypisane przez personel hotelu.

Należy utworzyć diagram przypadków użycia dla opisanego systemu rezerwacji hotelowych (rysunek 3.5).



Rysunek 3.5. Diagram przypadków użycia dla systemu rezerwacji hotelowych

Ć W I C Z E N I E

3.7 System obsługi konferencji

Wymagania systemowe serwisu internetowego obsługi konferencji międzynarodowej zaprezentowane graficznie w ćwiczeniu 2.1 stały się podstawą opracowania diagramu przypadków użycia dla tego systemu. Zakres wymagań funkcjonalnych uzasadnia wyszczególnienie następujących aktorów:

- ❑ *Autora*;
- ❑ *Recenzenta*;
- ❑ *Komitetu Programowego*.

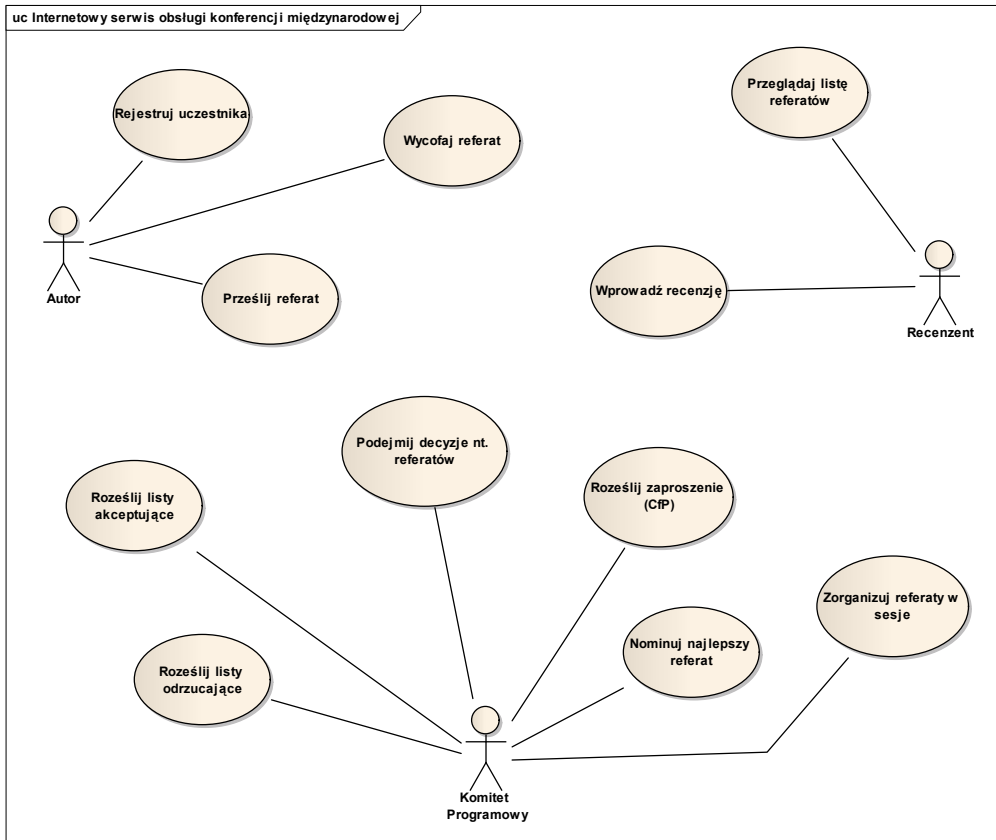
Starania *Komitetu Programowego* koncentrują się wokół zapewnienia odpowiedniej liczby nadesłanych referatów o akceptowalnym poziomie jakości. W związku z tym system powinien zapewnić *rozsyłanie zaproszeń* (*Call for Papers*) do potencjalnych *Autorów*, na tematyczne listy dyskusyjne, serwisy społecznościowe (np. Facebook) oraz do znaczących ośrodków akademickich. W odpowiedzi zainteresowani *Auto-rzy przesyłają referaty* do systemu obsługi konferencji, a po zaakceptowaniu *reje-strują się*, wnosząc odpowiednie opłaty konferencyjne. *Autor* powinien mieć również możliwość *wycofania referatu*. Referaty przechowywane w bazie są oceniane przez *Recenzentów*, którym przyporządkowuje się ich zadeklarowaną liczbę. *Re-cenzenci przeglądają listę referatów*, zapoznają się z nimi, analizują je, a następnie *wprowadzają recenzję*. Z kolei *Komitet Programowy podejmuje decyzje na temat referatów i organizuje zaakceptowane referaty w sesje*. Wynikiem decyzji dotyczących referatów jest *rozesłanie w systemie listów akceptujących i odrzucających* poszczególne referaty. Do decyzji *Komitetu Programowego* należy wybór i *nomina-cja najlepszego referatu*.

Na podstawie powyższego opisu oraz wymagań systemu konferencji opisanych w rozdziale drugim niniejszej książki należy opracować diagram przypadków użycia (rysunek 3.6).

3.2. Zadania

Zadanie 3.1.

Na podstawie ćwiczenia 2.1 opracuj hierarchię występujących w tym ćwiczeniu aktorów, wykorzystując związek uogólnienia. Przyporządkuj również aktorów do czterech rodzajów: osobowych i nieosobowych (systemy, urządzenia i czas), wykorzystując w tym celu stereotypy tekstowe lub graficzne.



Rysunek 3.6. Diagram przypadków użycia dla systemu obsługi konferencji

Zadanie 3.2.

Na podstawie ćwiczeń 3.1 i 3.2 opracuj dokumentację następujących przypadków użycia:

- ☐ Złóż zamówienie;
- ☐ Realizuj płatność online;
- ☐ Autoryzuj użytkownika.

Szczególną uwagę zwróć na dokładne opracowanie scenariuszy przypadków użycia opisanych w przepływach głównych i alternatywnych.

Zadanie 3.3.

Po wykonaniu zadań 3.1 i 3.2 wybierz trzy sklepy internetowe o zasięgu światowym (np. <http://www.amazon.com> czy <http://www.wbshop.com>) i skonfrontuj je z opisanym przykładem. Czym się różnią? Opracuj pełny diagram przypadków użycia dla jednego z przeanalizowanych sklepów.

Zadanie 3.4.

Spójność i dekompozycję diagramu przypadku użycia realizuje się poprzez związki występujące pomiędzy przypadkami użycia. Podstawowe rodzaje związków to: asocjacje, zależności, uogólnienia i realizacje. W nawiązaniu do ćwiczenia 3.1 zilustruj:

- ☐ zależności include i extend pomiędzy przypadkami użycia;
- ☐ uogólnienia wybranych przypadków użycia i wybranych aktorów;
- ☐ realizację określonego przypadku użycia.

Zadanie 3.5.

Na rysunku 3.6 zaprezentowano wybrane przypadki użycia serwisu internetowego obsługującego konferencję międzynarodową (ćwiczenie 2.1). Określ zależności występujące pomiędzy poszczególnymi przypadkami użycia i przypisz im odpowiednio stereotypy include oraz extend.

Zadanie 3.6.

Opracuj tabelaryczną specyfikację wybranego przypadku użycia z ćwiczenia 3.4.

Zadanie 3.7.

Korzystając z wyszukiwarek internetowych, zapoznaj się z typowym zakresem działalności biura rachunkowego. Przeanalizuj możliwości wspomagania działalności biznesowej biura rozwiązaniami informatycznymi i zaproponuj kompletną listę przypadków użycia dla projektowanego systemu. Zidentyfikuj i przydziel stereotypy dla przypadków użycia typu CRUD. Dokonaj pogrupowania przypadków użycia względem kryterium poziomu istotności w systemie. Kolorem czerwonym oznacz przypadki niezbędne do realizacji systemu, pomarańczowym — istotne, żółtym — przeciętnie istotne, a zielonym — mało istotne.



4

Diagramy klas oraz definiowania bloków

4.1. Ćwiczenia

Ć W I C Z E N I E

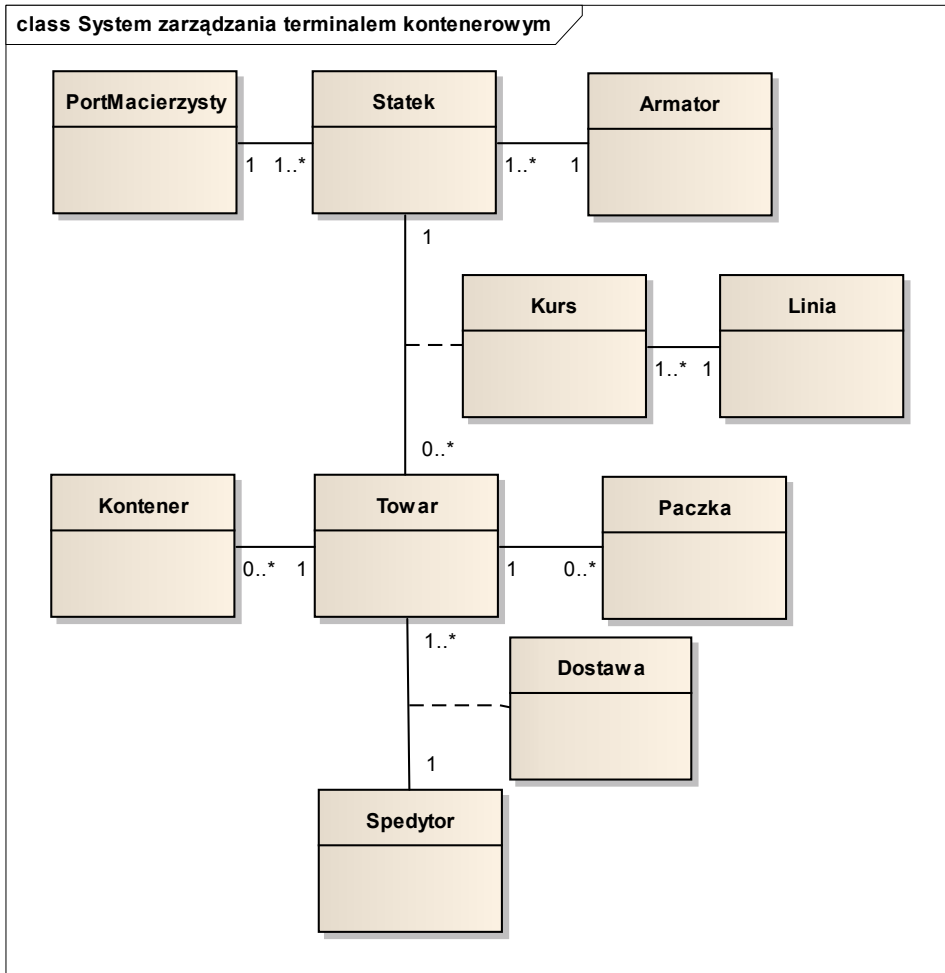
4.1

System zarządzania terminalem kontenerowym

Zarządzanie terminalem kontenerowym można sklasyfikować jako bardzo złożony proces logistyczny. System informatyczny wspierający zarządzanie takim przedsiębiorstwem musi tym samym ewidencjonować nie tylko posiadane zasoby, ale również wszelkiego rodzaju zdarzenia mające związek z wpływieniem i wypłynięciem statków, przyjęciem towaru itp. Stawia to określone wymagania przed zespołem opracowującym założenia w zakresie systemowej bazy danych.

Wpływający do terminalu *Statek* jest przyporządkowany do odpowiadającego mu *Portu macierzystego*, który stanowi zarówno miejsce jego rejestracji, jak i postoju. Każdy *Statek* jest przypisany również do *Armatora*, który najczęściej jest właścicielem statku. Na jednym *Statku* może znaleźć się wiele *Towarów*, które mogą być przewożone w *Kontenerach* lub w zwykłych *Paczках*. *Paczkę* lub *Kontener* przypisuje się do *Rejsu statku*, czyli trasy i związanego z nią okresu, jaki upływa od wypłynięcia z portu wyjściowego do momentu dotarcia do punktu docelowego. Każdy *Statek* pływa po *Liniach* (np. Morza Bałtyckiego czy oceanicznych) wyznaczonych dla danego *Rejsu Statku*. System powinien zatem pozwalać na wyświetlenie kompletnych informacji dotyczących *Rejsów statków*. *Towary* i *Kontenery* są dostarczane do portu wyjściowego i odbierane w porcie docelowym drogą lądową przez wyspecjalizowane firmy spedycyjne, zwane *Spedytorami*. Zdarzenia takie ewidencjonowane są w systemie jako *Dostawy*.

Należy zaproponować listę klas umożliwiających ewidencjonowanie w systemie faktu wpłynięcia lub wypłynięcia *Statku* z *Towarem* (*Kontenerami*) oraz wydania lub przyjęcia *Towaru* przez *Spedytora*. Zidentyfikowane klasy należy powiązać stosownymi związkami, tworząc tym samym diagram klas języka UML na poziomie konceptualnym (rysunek 4.1).



Rysunek 4.1. Diagram klas systemu zarządzania terminalem kontenerowym



Ć W I C Z E N I E

4.2 System obsługi firmy transportowej

Firma transportowa świadczy swoje usługi, wykorzystując do tego celu zróżnicowany tabor pojazdów. Poszczególne rodzaje pojazdów wymieniono poniżej, ujmując podstawowe parametry danego pojazdu:

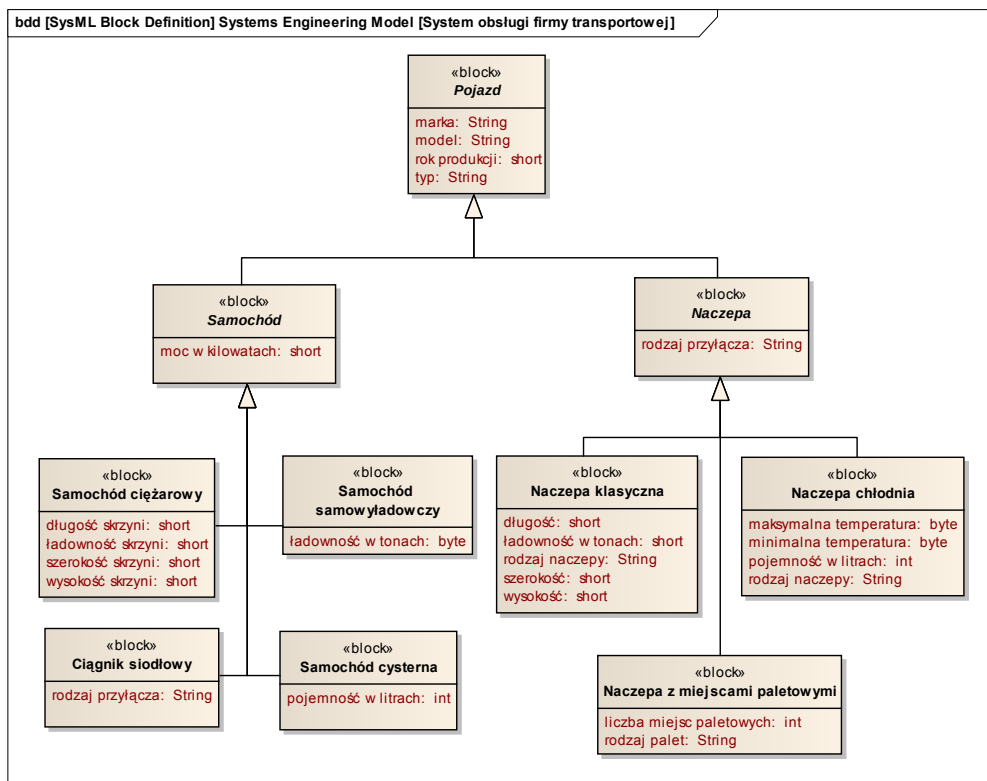
- ❑ *Naczepy klasyczne (rodzaj przyłącza, rodzaj naczepy, długość, szerokość, wysokość, ładowność w tonach, marka, model, typ, rok produkcji);*
- ❑ *Naczepy z miejscami paletowymi (rodzaj przyłącza, rodzaj palet, liczba miejsc paletowych, marka, model, typ, rok produkcji);*
- ❑ *Naczepy-chłodnie (rodzaj przyłącza, rodzaj naczepy, pojemność w litrach, minimalna i maksymalna temperatura, marka, model, typ, rok produkcji);*
- ❑ *Samochody samowyladowcze (moc, ładowność w tonach, marka, model, typ, rok produkcji);*
- ❑ *Samochody ciężarowe (moc, długość, wysokość, szerokość i ładowność skrzyni, marka, model, typ, rok produkcji);*
- ❑ *Ciągniki siodłowe (rodzaj przyłącza, moc, marka, model, typ, rok produkcji);*
- ❑ *Samochody-cysterny (moc, pojemność w litrach, marka, model, typ, rok produkcji).*

Należy utworzyć diagram definiowania bloków dla systemu obsługi firmy transportowej ukierunkowany na prezentację struktury danych na temat dostępnego taboru pojazdów pod kątem wyboru optymalnego samochodu i naczepy (rysunek 4.2).

**Wskazówki dydaktyczne**

Wspólne atrybuty dla bloków konkretnych zostały zdefiniowane w ramach bloków abstrakcyjnych (ang. *abstract*), których nazwa pisana jest pochyłą czcionką. Nazwy atrybutów zostały w powyższym ćwiczeniu napisane zgodnie z notacją poziomu konceptualnego, stąd występowanie polskich liter oraz spacji w nazwach atrybutów. W przypadku poziomu implementacyjnego występowanie polskich znaków oraz spacji w nazwach atrybutów jest niedopuszczalne. Nazwy bloków zwyczajowo pisze się w liczbie pojedynczej, jeżeli nie zmienia to istoty bloku.

Wymiary *Naczepy klasycznej* oraz *Samochodu ciężarowego* są podawane w centymetrach, dlatego użyty został typ `short`, który pozwala na zapisywanie liczb z przedziału od -32 768 do 32 767.



Rysunek 4.2. Diagram definiowania bloków systemu obsługi firmy transportowej

Ć W I C Z E N I E

4.3 Elektroniczny indeks studenta

Jednym z symboli statusu studenta jest indeks, tradycyjnie od wielu lat wręczany na inauguracjach roku akademickiego. Wraz ze wzrostem liczby studentów obsługa i wypełnianie indeksów angażują w coraz większym stopniu pracowników naukowo-dydaktycznych oraz kadrę administracyjną. Toteż aktualnie w wielu uczelniach następuje popularyzacja indeksów elektronicznych minimalizujących wykorzystanie dokumentów drukowanych, liczbę popełnianych pomyłek oraz zwiększających efektywność pracy dziekanatów i znacznie usprawniających proces dydaktyczny zarówno dla *studentów*, jak i *wykładowców*.

Typowy indeks elektroniczny umożliwia:

- ☐ przeglądanie listy przedmiotów wraz z ocenami;
- ☐ rejestrację studentów na zajęcia kursowe;

- ❑ wprowadzanie ocen z tytułu zaliczeń i egzaminów;
- ❑ wprowadzanie ocen poprawkowych i rejestrowanie egzaminów warunkowych;
- ❑ zamykanie sesji egzaminacyjnej;
- ❑ powiązanie ocen w indeksie z ocenami w protokole;
- ❑ przenoszenie ocen z protokołu wykładowcy do indeksu studenta;
- ❑ weryfikację numeru legitymacji studenckiej i jej ważności.

System elektroniczny indeks studenta korzysta z zasobów systemu kadrowego uczelni, w którym przechowywane są podstawowe dane personalne *studentów* i *pracowników*. System zawiera informację o wszystkich *przedmiotach* wykładanych na danym wydziale w danej *sesji egzaminacyjnej*. *Wykładowca* uzyskuje możliwość dokonywania *wpisów* i korekt tych wpisów w *indeksie elektronicznym*. Po zakończeniu *sesji egzaminacyjnej* generowany jest z każdego przedmiotu *protokół*, który jest formalnym zatwierdzeniem zaliczenia *przedmiotu* przez *studenta*. Pracownicy posiadający wymagany stopień naukowy mogą zostać *promotorami* seminariów dyplomowych. Następuje wówczas przypisanie *promotorów* do *studentów* na zasadzie jeden do wielu. *Indeks elektroniczny* wiąże jednocześnie *studenta* z *grupą* wykładową. Podstawą akceptacji przystąpienia *studenta* do *sesji egzaminacyjnej* jest ważna *legitymacja studencka*, która zawiera podstawowe dane personalne *studenta* pobrane bezpośrednio z bazy danych.

W ramach wdrażania autorskiego rozwiązania informatycznego w zakresie zarządzania zajęciami dydaktycznymi studentów jedna z uczelni zdecydowała się przyjąć szereg ograniczeń. Wśród założeń dla systemu wymienić należy:

- ❑ przyjęcie, że numer grupy jest zawsze liczbowy — np. 101;
- ❑ ewidencjonowanie w systemie wyłącznie pracowników dydaktycznych oraz naukowo-dydaktycznych.

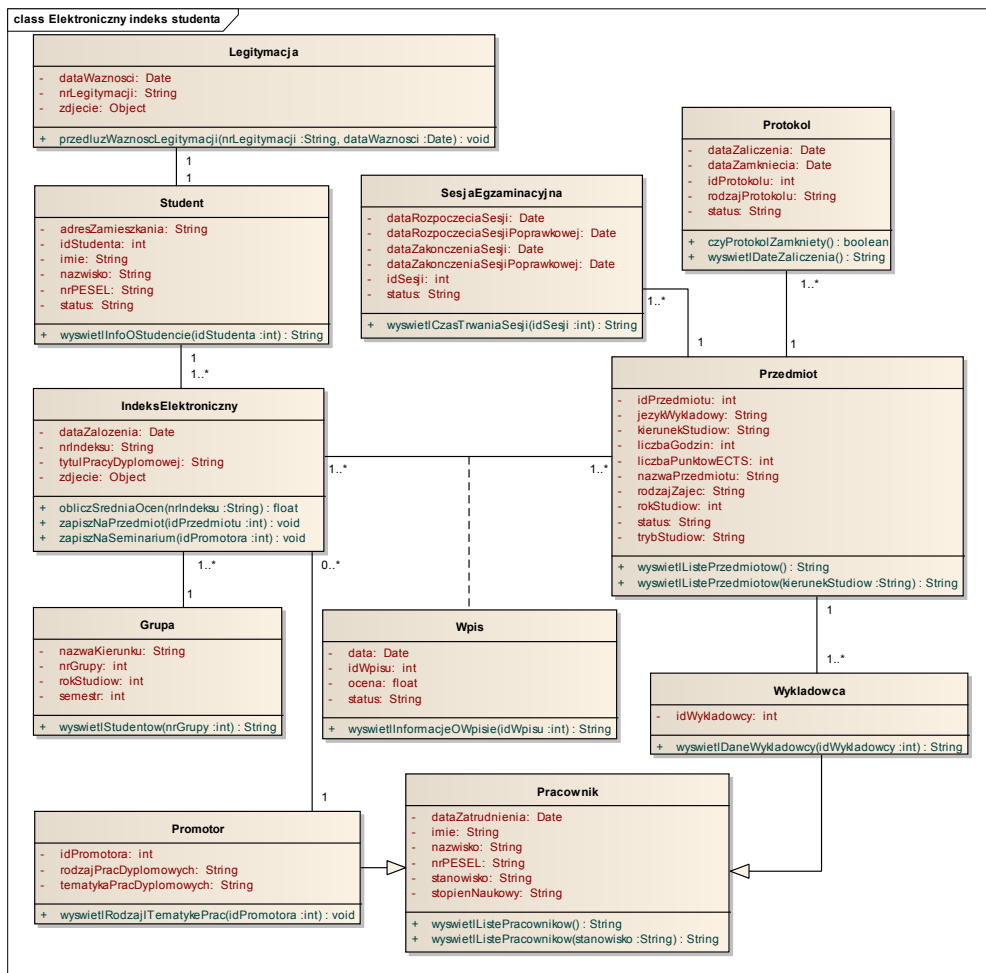
Projektant systemu informatycznego został obciążony zadaniem zaproponowania diagramu klas ilustrującego strukturę danych niezbędnych do zrealizowania wymienionej funkcjonalności (rysunek 4.3). W diagramie ująć należy szczegółową listę atrybutów wraz z typami danych oraz wymienić podstawowe metody, które mają być zaimplementowane w poszczególnych klasach.

Ć W I C Z E N I E

4.4

Formularz dokumentujący historię kontaktów z klientem w systemie CRM

Integralną częścią systemu CRM jest formularz dokumentujący historię kontaktów z klientem. Jest to forma zapisu informacji o interakcjach z nabywcą zebranych przez pracownika odpowiedzialnego za kontakty z daną osobą lub podmiotem.



Rysunek 4.3. Diagram klas elektronicznego indeksu studenta

W odróżnieniu od formularza danych klienta omawiany formularz nie zawiera danych adresowych i innych informacji bezpośrednio związanych z klientem. Koncentruje się on natomiast na zebraniu danych pozwalających na sporządzanie list rankingowych oraz programów lojalnościowych klientów.

Dane zamieszczane w formularzu koncentrują się na dokumentowaniu stanu aktualnego oraz historii kontaktów ujmującej:

- ❑ datę kontaktu;
- ❑ nazwisko oraz imię osoby kontaktowej;
- ❑ formę kontaktu (e-mail, telefon itd.);

- ❑ *namiar kontaktu* (numer telefonu, adres e-mail itd.);
- ❑ *treść kontaktu*;
- ❑ *status kontaktu*;
- ❑ *priorytet kontaktu* (niski, normalny, wysoki, krytyczny);
- ❑ *listę dokumentów wysłanych oraz otrzymanych*;
- ❑ *przeprowadzone oraz zaplanowane spotkania*.

Klient nie ma możliwości wglądu w dane na swój temat. W celu spełnienia wymagań implementacyjnych analitycy systemowi zdecydowali się na dodatkowe ujęcie następujących wartości:

- ❑ *identyfikatora kontaktu*, który jest kluczem głównym;
- ❑ *identyfikatora klienta* będącego kluczem obcym, wskazującym na tabelę z danymi klienta.

Wartości *TypKontaktu*, *StatusKontaktu* oraz *PriorytetKontaktu* zostały wyszczególnione z zastosowaniem wyliczeniowych typów wartości. Zostały one oznaczone poprzez zastosowanie stereotypu enumeration. Wyliczeniowe typy wartości ilustrują kompletną listę wariantów, jakie przyjąć może dana wartość. I tak, *TypKontaktu* może być przeprowadzony przez następujące kanały:

- ❑ *telefon*;
- ❑ *faks*;
- ❑ *formularz webowy*;
- ❑ *e-mail*;
- ❑ *skype*;
- ❑ *gadu-gadu*;
- ❑ *pocztę*;
- ❑ *serwis społecznościowy*;
- ❑ *wideokonferencję*;
- ❑ *wiki*;
- ❑ *blog*;
- ❑ *inny*.

Kolejnym typem wyliczeniowym jest *StatusKontaktu* przyjmujący następujące stany:

- ❑ *wstępny*;
- ❑ *nawiązany*;
- ❑ *historyczny*.

Ostatnim z zastosowanych typów wyliczeniowych jest *PriorytetKontaktu*. Z jego pomocą ustalana jest istotność danego kontaktu z punktu widzenia poziomu współpracy z klientem. *Priorytet* może przyjmować następujące wartości:

- ☐ niski;
- ☐ normalny;
- ☐ wysoki;
- ☐ krytyczny.

Jednym z elementów bloku *Kontakt* są wartości *Przeprowadzone spotkanie* oraz *Zaplanowane spotkanie*. Ze względu na złożoność tego typu danych opisowych zdecydowano się na ustrukturyzowanie wprowadzanych informacji poprzez wyodrębnienie osobnego bloku o nazwie *Spotkanie*. Składa się on z:

- ☐ *identyfikatora spotkania*;
- ☐ *daty spotkania*;
- ☐ *miejsca spotkania*;
- ☐ *dokumentów przekazanych oraz otrzymanych podczas spotkania*;
- ☐ *złożonych i otrzymanych propozycji*;
- ☐ *wynikających zobowiązań*.

Na podstawie szczegółowej charakterystyki formularza dokumentującego historię kontaktów z klientem należy zaproponować diagram definiowania bloków zgodnie ze standardem SysML (rysunek 4.4).



Wskazówki dydaktyczne

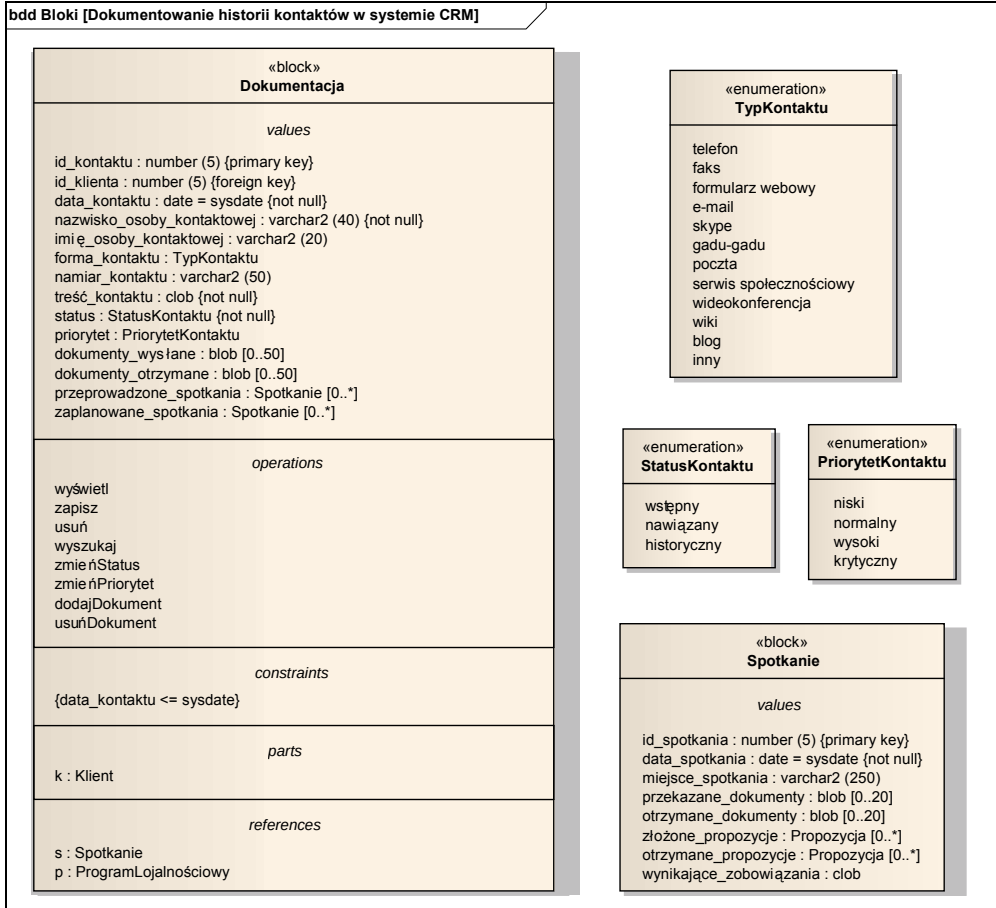
Zaprezentowany w niniejszym ćwiczeniu diagram bloków posiada ograniczenia (*primary key*, *foreign key*, *not null*), które mogą być kluczami głównymi, obcymi lub sprawdzać, czy wartości są niepuste. Na diagramie zaprezentowano również wartość domyślną dla daty — *sysdate*, która odzwierciedla bieżący dzień. Podczas tworzenia diagramu bloków analityk często podejmuje decyzje, jakiego typu pozycje *values* będą przyjmować domyślne wartości.

Ć W I C Z E N I E

4.5 Obsługa transakcji w systemie CRM

Nawiązując do ćwiczenia 4.4, należy utworzyć diagram definiowania bloków nakształający szerszy kontekst organizacji danych w systemie CRM. Trzeba przewidzieć następujące bloki:

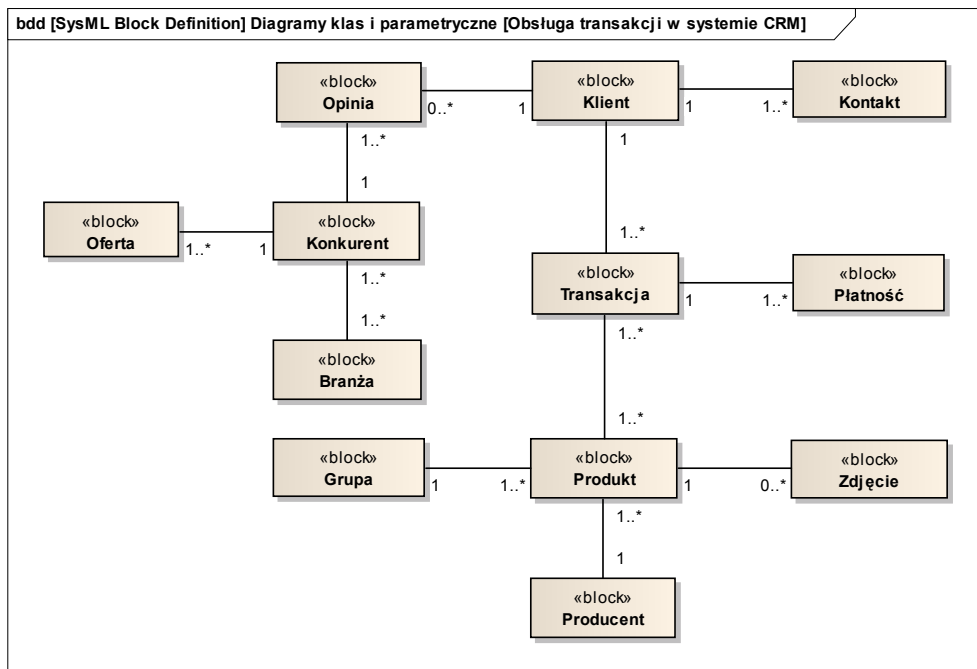
- ☐ *Klient*;
- ☐ *Kontakt* do klienta;
- ☐ *Transakcja*;
- ☐ *Płatność* za transakcję;
- ☐ *Produkt*;



Rysunek 4.4. Diagram definiowania bloków formularza dokumentującego historię kontaktów z klientem w systemie CRM

- ❑ Grupa produktów;
- ❑ Producent produktu;
- ❑ Zdjęcie produktu;
- ❑ Konkurent;
- ❑ Opinia klientów o konkurencie;
- ❑ Oferta konkurenta;
- ❑ Branża konkurenta.

Do licznych celów systemu klasy CRM zaliczyć należy m.in. gromadzenie informacji na temat konkurentów (por. rysunek 4.5). Poszczególni klienci podczas dokonywania transakcji są proszeni o wypełnienie krótkiej ankiety, w której oceniają jakość transakcji oraz wyrażają subiektywną ocenę na temat konkurentów firmy, jeżeli kiedykolwiek dokonywali u nich transakcji zakupu. Na tej podstawie opracowywane są programy lojalnościowe dla klientów oraz podnosi się jakość procesu sprzedaży poprzez dostosowanie się do uwag zgłaszanych przez poszczególnych klientów.



Rysunek 4.5. Diagram definiowania bloków obsługi transakcji w systemie CRM



Wskazówki dydaktyczne

Diagram definiowania bloków w powyższym ćwiczeniu opisuje strukturę bazodanową systemu klasy CRM. Istotnym elementem są tutaj liczebności, za pomocą których można zidentyfikować potencjalne klucze, takie jak główne i obce w tabelach bazodanowych. Ponieważ jest to diagram zaliczany do konceptualnych (pominięto na tym etapie chociażby atrybuty), dopuszczalne jest modelowanie związków wiele do wielu.

Klient ma możliwość dodawania *Opinii* o *Konkurentach*. O jednym *Konkurencie* każdy *Klient* może wyrazić wiele *Opinii*. *Opinia* podczas dodawania musi być przypisana do *Konkurenta*. Stąd też liczebność pomiędzy *Opinią* a *Konkurentem* jest 1..*-1.

Ć W I C Z E N I E

4.6

Kancelaria prawnicza

Współczesne kancelarie prawnicze to wysoko wyspecjalizowane organizacje biznesowe, zwykle kilkuosobowe, nastawione na osiąganie zysku. Ich specjalizacja opiera się na klasycznych zawodach prawniczych, wśród których wyróżnić należy zawody będące:

- ❑ efektem odbytych aplikacji radcowskich i adwokackich będące najpopularniejszą grupą profesji;
- ❑ połączeniem wybranych funkcji prawniczych, np. notariusze.

Istotnym czynnikiem wpływającym na atrakcyjność kancelarii prawniczych jest możliwość zatrudniania *Aplikantów*, którzy po ukończeniu studiów, odbyciu aplikacji i pozytywnym zdaniu egzaminów uzyskują prawo do wykonywania zawodu. Prawnicy zatrudnieni jako *Aplikanci* pełnią zwykle funkcje pomocnicze, gdyż samodzielnie nie mogą wykonywać funkcji *Adwokata* czy *Radcy prawnego*. Stąd też każdy *Aplikant* jest przypisany do nadzorującego go *Adwokata*, *Radcy prawnego*, czy też innego przedstawiciela wybranej profesji prawniczej.

Klient przychodzący do kancelarii prawniczej wypełnia formularz *Sprawy*. W zależności od kategorii *Sprawy* (opinia, porada prawna, sporządzenie aktu notarialnego, występowanie w roli obrońcy lub oskarżyciela) jest ona kierowana do właściwego pracownika — a w przypadku większej złożoności *Sprawy* do grupy pracowników. *Aplikant* ze względu na brak samodzielności nie może być przyporządkowany bezpośrednio do *Sprawy*. Należy podkreślić, że jeden *Klient* może wnieść wiele *Spraw*, a wielu klientów może wnieść jedną *Sprawę*.

Należy sporządzić conceptualny diagram klas obsługi systemu kancelarii prawniczej, który pozwoli na ewidencję *Pracowników*, w tym *Aplikantów* oraz *Klientów*, i wnoszonych przez nich *Spraw* (rysunek 4.6). Zakłada się, że zgodnie z wymienionymi powyżej profesjami w kancelarii zatrudnieni będą jedynie *Radcy prawni*, *Adwokaci* oraz *Notariusze*.

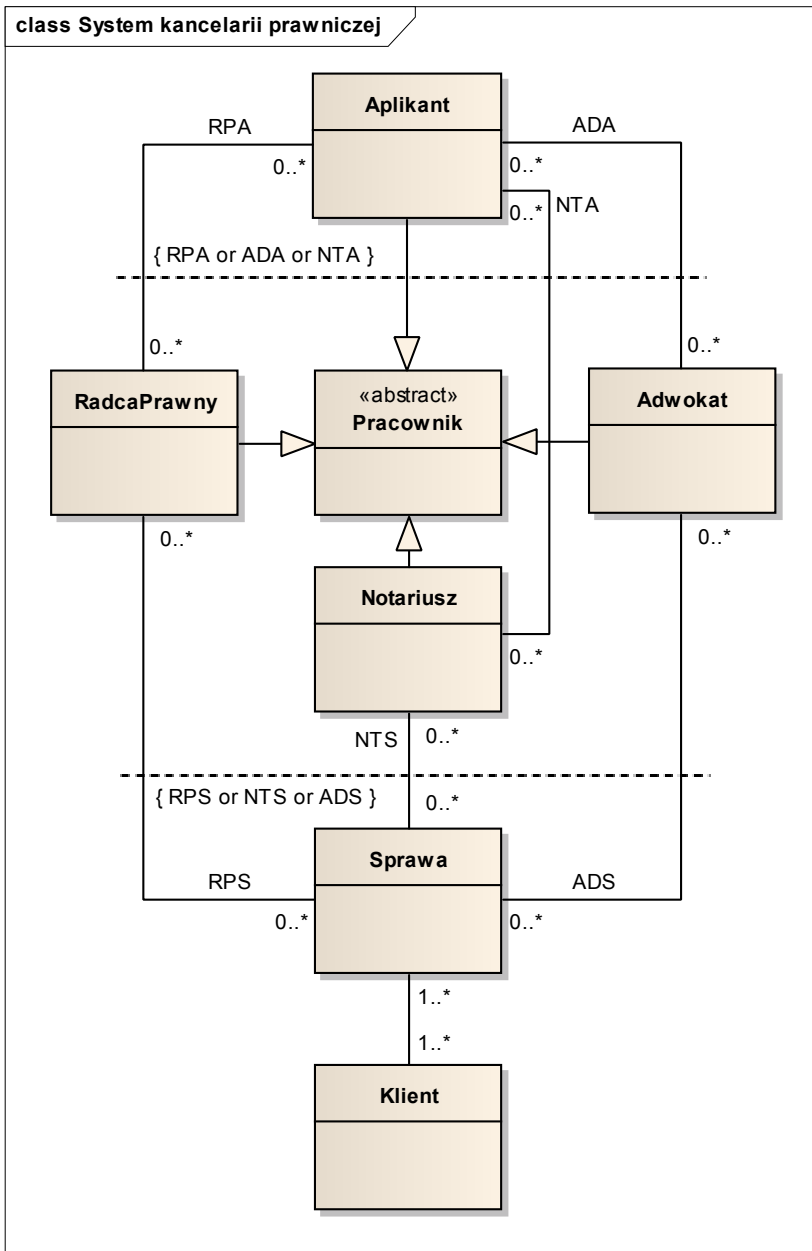
**Wskazówki dydaktyczne**

Na diagramie poszczególne ograniczenia oznaczają:

- ◆ *ADA* — *Adwokat* — *Aplikant*;
- ◆ *NTA* — *Notariusz* — *Aplikant*;
- ◆ *RPA* — *Radca prawny* — *Aplikant*.

Analogicznie do przypisania odpowiednio *Adwokata*, *Notariusza* lub *Radcy prawnego* do *Sprawy* używane są ograniczenia *ADS*, *NTS*, *RPS*.

Pracownik jest bytem abstrakcyjnym, który posiada podstawowy zestaw atrybutów przekazywanych do klas dziedziczących: *Adwokata*, *Notariusza*, *Radcy prawnego* i *Aplikanta*.



Rysunek 4.6. Diagram klas systemu kancelarii prawniczej

Ć W I C Z E N I E

4.7 System rezerwacji hotelowych

Zarządzanie współczesnym hotelem zmierzające do zapewnienia satysfakcji gości hotelowych jest zadaniem złożonym i możliwym do osiągnięcia tylko przez użycowanie sprawnego systemu informatycznego. Jedynie część istotnych funkcji takiego systemu przedstawiono w ćwiczeniu 3.6, na rysunku 3.5. Warunkiem funkcjonowania takiego systemu jest niezawodna baza danych zawierająca aktualne dane na bieżąco modyfikowane. Baza ta powinna zawierać szereg klas ściśle powiązanych ze sobą związkami, głównie asocjacjami. Przede wszystkim niezbędne jest przechowywanie i wyszukiwanie danych o *gościach hotelowych* i związanych z nimi *rezerwacjach* i *meldunkach*. Poza indywidualnymi *gośćmi hotelowymi* rezerwacji mogą dokonywać *firmy*. W ramach rezerwacji *gość hotelowy* może określić dogodną dla siebie propozycję *standardu żywienia*. Ważnym elementem takiej bazy danych jest funkcjonalność związana z niezawodną obsługą *płatności* i *fakturowania* za podstawowe usługi hotelowe, ale także oferowane przez hotel *usługi turystyczne* i *usługi dodatkowe*. Realizacja płatności za usługi może uwzględniać zniżkę, jaka została przypisana danej *firmie* lub *gościowi hotelowemu*. W tym celu stosowane są klasy asocjacyjne *PłatnośćUsługaTurystyczna* oraz *PłatnośćUsługaDodatkowa*.

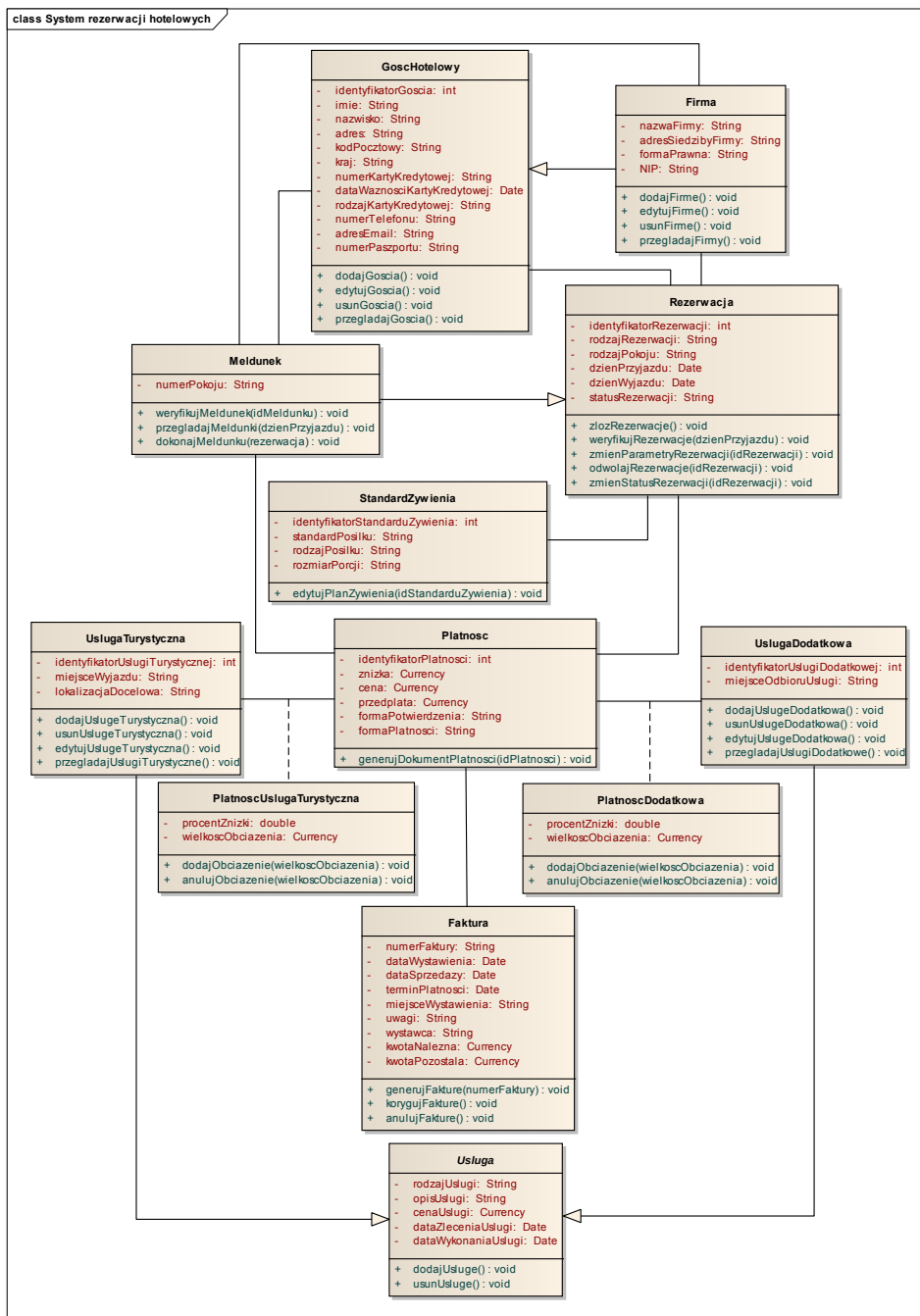
Specyfikacja klas w diagramie klas przyszłej bazy danych systemu rezerwacji hotelowych musi zostać wypełniona identyfikacją wszystkich niezbędnych atrybutów i operacji na tych atrybutach.

Należy opracować diagram klas dla bazy danych systemu rezerwacji hotelowych. Na rysunku 4.7 zidentyfikowano i wymieniono niezbędne dla funkcjonowania systemu atrybuty i operacje dla poszczególnych klas. Rysunek ten zawiera również oznaczenia uogólnień i klas asocjacyjnych.

**Wskazówki dydaktyczne**

Należy zwrócić uwagę na to, że meldunek może być utworzony na podstawie rezerwacji lub bez wcześniejszej rezerwacji. Stąd też na diagramie w klasie *Meldunek* znajduje się metoda o nazwie *dokonajMeldunku*, której atrybutem jest wartość logiczna (tak, nie) o nazwie *rezerwacja*.

Usługa jest klasą abstrakcyjną, która przekazuje swoje atrybuty oraz metody do klas *UsługaTurystyczna* oraz *UsługaDodatkowa*. Dzięki temu system jest bardziej elastyczny i skalowalny.



Rysunek 4.7. Diagram klas dla bazy danych systemu rezerwacji hotelowych

4.2. Zadania

Zadanie 4.1.

System klasy Business Intelligence działający na podstawie danych pochodzących z hurtowni danych będzie pozwalał na generowanie raportów dla kierownictwa w cyklach miesięcznych. Raporty te będą zawierały listy rankingowe dla minionego miesiąca:

- ❑ najlepiej sprzedających się produktów;
- ❑ klientów, którzy kupili najwięcej produktów;
- ❑ ogólne dane dotyczące sprzedaży — zysk oraz przychód.

Sporządź diagram klas dla modułu raportowania, który będzie zawierał następujące dane:

- ❑ *Listę rankingową produktów*: do produktu ma być przyporządkowana ilość sprzedanych produktów (we właściwej jednostce miary), kwota uzyskana za te produkty oraz informacja o tym, którzy klienci kupowali najwięcej produktu, gdzie:
 - ❑ *Dane o kliencie*: nazwa klienta, NIP, REGON, ulica, numer domu, numer lokalu, kod pocztowy, miasto, kraj;
 - ❑ *Dane o produkcie*: nazwa produktu, kod produktu, grupa produktów, podgrupa produktów, jednostka miary;
- ❑ *Ogólne informacje o wyniku za ubiegły miesiąc*: zysk oraz przychód.

Powyższe dane mają uwzględniać aspekt czasu — co oznacza, że dane dotyczące liczby sprzedanych sztuk oraz wyniku finansowego powinny posiadać odpowiedni znacznik czasowy. Nazwy atrybutów, klas oraz metod powinny być zgodne z notacją implementacyjną. Typy danych muszą odzwierciedlać typy zgodne z platformą .NET. Modyfikatory dostępu powinny pozwalać na dostęp do atrybutów tylko poprzez metody.

Zadanie 4.2.

Uzupełnij ćwiczenie 4.7 o liczebności pomiędzy klasami. Na jego podstawie utwórz nowy diagram klas dla systemu hotelowego obsługującego zarządzanie obciążeniami. Diagram ten powinien nawiązywać do diagramu utworzonego w ćwiczeniu 4.7 i łączyć się z nim poprzez klasę `Meldunek`.

Zadanie 4.3.

Na podstawie ćwiczenia 4.1 zaproponuj dodanie klas abstrakcyjnych oraz uzupełnij prezentowany diagram klas o atrybuty i metody.

Zadanie 4.4.

Na podstawie ćwiczenia 4.6 rozwiń diagram klas systemu kancelarii prawniczej, dodając do niego atrybuty oraz metody.

Zadanie 4.5.

Opracuj diagram definiowania bloków dla firmy kurierskiej w celu umożliwienia wyboru różnych opcji potwierdzenia doręczenia przesyłki kurierskiej. Poszczególne opcje powinny obejmować co najmniej potwierdzenia telefoniczne, faksem, SMS-em oraz e-mailem.

Zadanie 4.6.

Przedstaw w postaci związku uogólnienia dziedziczenie atrybutów i operacji następujących bloków związanych z klasyfikacją przesyłek kurierskich:

- ☐ krajowe;
- ☐ importowe;
- ☐ eksportowe;
- ☐ z materiałami niebezpiecznymi;
- ☐ wartościowe;
- ☐ niestandardowe (nietypowe gabaryty, waga i właściwości towaru).

Po opracowaniu hierarchii bloków należy każdy z tych bloków uzupełnić adekwatnymi wartościami i operacjami, wykorzystując następujące informacje: każdą z przesyłek można nadawać w innym standardzie, co wiąże się ze zróżnicowaną wysokością opłat: pilna wysyłka lotnicza, dedykowana w transporcie drogowym, ekonomiczna; nadawca każdej przesyłki może wybrać godzinę jej doręczenia spośród trzech opcji: do godziny 9:00, do godziny 12:00, do godziny 18:00.

Zadanie 4.7.

Opracuj diagram definiowania bloków zawierający sekcję `values` dla następujących obiektów graficznych stosowanych na mapach cyfrowych w systemach GIS:

- ☐ punkt (wielkość, kolor);
- ☐ linia (styl, kolor);
- ☐ powierzchnia (wzór i kolor wypełnienia);
- ☐ tekst (typ, rozmiar oraz kolor czcionki);
- ☐ wielobok (wzór i kolor wypełnienia);
- ☐ symbol (rozmiar, kolor).

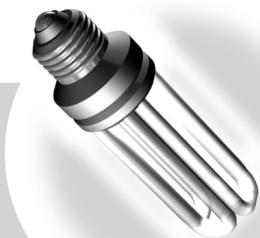


Diagram bloków wewnętrznych oraz diagramy wdrożeniowe

5.1. Ćwiczenia

Ć W I C Z E N I E

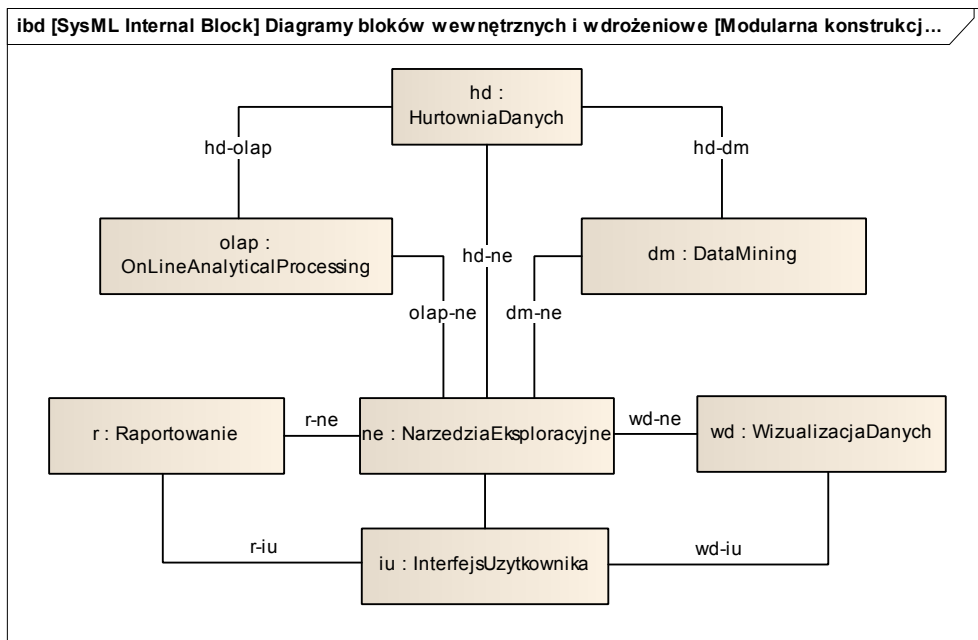
5.1 Modułarna konstrukcja systemów klasy Business Intelligence

W celu obniżenia kosztów zakupu systemu informatycznego klienci coraz częściej mają możliwość selektywnego nabycia wyłącznie interesującej ich funkcjonalności. Możliwość taką stwarza modułarna konstrukcja złożonych systemów informatycznych zarządzania, gdzie każdy moduł oferuje tylko wybrany zasób funkcjonalności. Najlepszym przykładem budowy modułowej są systemy klasy Business Intelligence, w których każdy z podsystemów może być traktowany jako oddzielny moduł, powiązany jednak funkcjonalnie z całością innych podsystemów. W klasycznym systemie Business Intelligence wyróżnić należy:

- ❑ *hurtownię danych*;
- ❑ narzędzia analityczne, w tym:
 - ❑ OLAP (*OnLine Analytical Processing*) do wielowymiarowej analizy danych;
 - ❑ *Data Mining*, które wykorzystują techniki takie jak sieci neuronowe, analizę regresji i korelacji itp.;
- ❑ *narzędzia eksploracyjne* umożliwiające między innymi wielowymiarowy wgląd w dane biznesowe;
- ❑ *narzędzia do wizualizacji danych* pozwalające na sporządzanie wykresów czy map;

- ❑ narzędzia do *raportowania* i do zapytań ad hoc oferujące predefiniowany zestaw raportów;
- ❑ *interfejs użytkownika* dostępny przez stronę WWW, pozwalający na korzystanie z szeregu narzędzi Business Intelligence.

Na podstawie powyższego opisu należy przedstawić modułarną konstrukcję systemu klasy Business Intelligence w postaci diagramu bloków wewnętrznych (rysunek 5.1).



Rysunek 5.1. Diagram bloków wewnętrznych modularnej konstrukcji systemów klasy Business Intelligence



Wskazówki dydaktyczne

Prezentowany diagram jest używany głównie do prezentowania podsystemów. Nie tylko podsystemów informatycznych, ale przede wszystkim inżynierskich, np. modułów systemu przeciwpółślizgowego ABS instalowanego w samochodach. Pomiędzy poszczególnymi blokami (podsystemami) zamieszczono konektory, których opis określa rodzaj przepływu — przykładowo *hd-olap* oznacza przepływ danych z hurtowni danych do podsystemu OLAP. Doprecyzowanie jest możliwe poprzez specyfikację interfejsów/portów.

Ć W I C Z E N I E

5.2

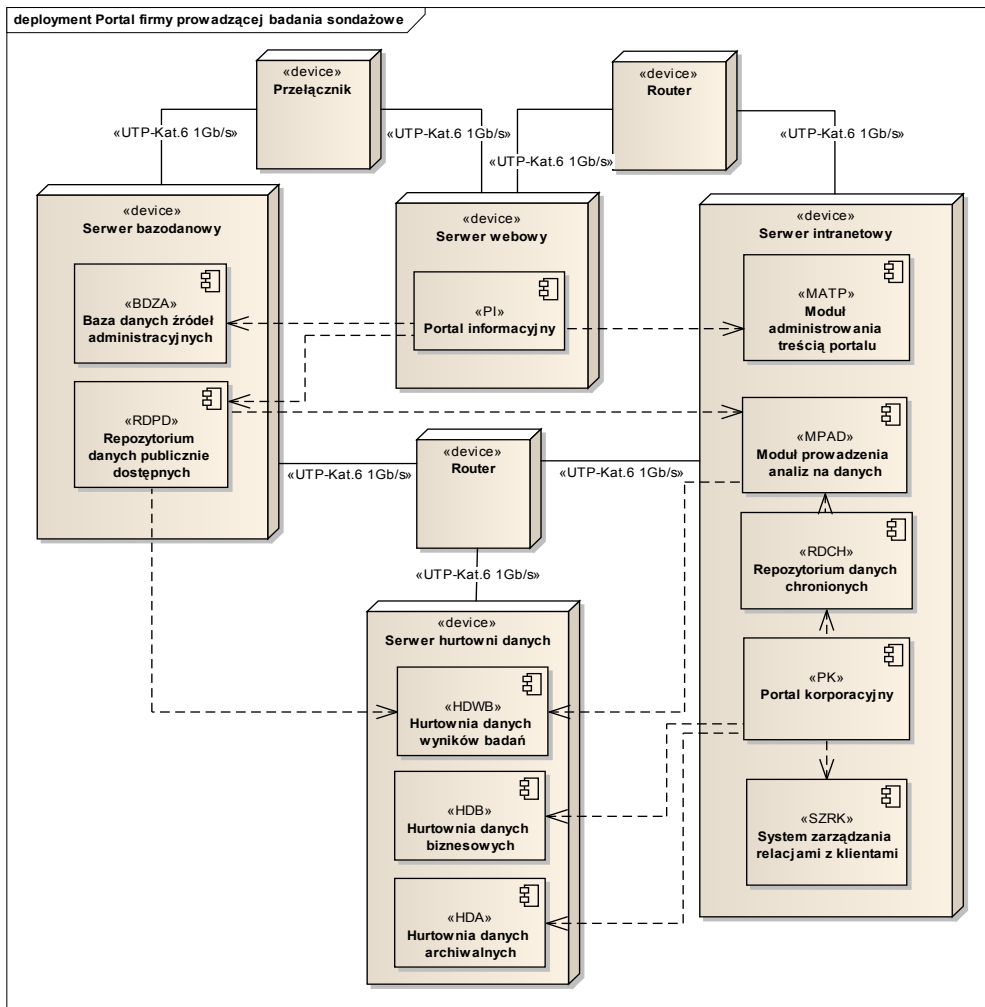
Portal informacyjny firmy prowadzącej badania sondażowe

Firma prowadząca badania społeczne działa na zasadzie przyjmowania zleceń od klientów, którymi mogą być zarówno osoby fizyczne, jak i organizacje posiadające osobowość prawną. Dodatkowo w celu autopromocji firmy prowadzone są przez nią badania na tematy mające duży wydźwięk społeczny w danym momencie. Wyniki tych badań są publikowane na *Portalu informacyjnym (PI)* firmy. Istotnym uzupełnieniem treści tego portalu jest prezentowanie informacji z powszechnie dostępnych źródeł administracyjnych. Są one publikowane z podaniem łącza do witryny internetowej organizacji, od której pochodzi konkretna informacja. Dane te są przechowywane w *Bazie danych źródeł administracyjnych (BDZA)*. Firma posiada dodatkowo *Hurtownię danych wyników badań (HDWB)*, w której przechowywane są wyniki badań panelowych w postaci szeregów czasowych. W ten sposób pod różnymi pojęciami gromadzone są wszystkie wyniki badań panelowych. Dokonywanie analiz na danych zawartych w *HDWB* zapewnia *Moduł prowadzenia analiz na danych (MPAD)*. Analizy i wyniki badań są przechowywane w dwóch wydzielonych repozytoriach danych: *Repozytorium danych publicznie dostępnych (RDPD)* i *Repozytorium danych chronionych (RDCH)*. *RDPD* jest bezpośrednio zintegrowane z *PI*, co pozwala wszystkim użytkownikom Internetu na dostęp do publicznych danych ustrukturyzowanych. *Moduł RDCH* dostępny jest tylko w sieci intranet firmy, a dostęp do niego jest ograniczony tylko do wybranej grupy użytkowników zajmujących się prowadzeniem badań. Dodatkowo firma posiada *Hurtownię dokumentów archiwalnych (HDA)*, gdzie w postaci plików przechowywane są wszystkie dokumenty, jakie kiedykolwiek zostały przysłane do firmy oraz z niej wysłane — w tym również zeskanowane wersje papierowe tych dokumentów. Bezpośrednio z portalem informacyjnym zintegrowany jest *Moduł administrowania treściami portalu (MATP)*, gdzie pracownicy firmy mają możliwość publikowania na portalu informacyjnym nowych treści w postaci artykułów, opracowań wyników badań oraz plików XLS. Istotnym uzupełnieniem systemów dostępnych w firmie jest System ERP zawierający moduł CRM występujący pod nazwą *System zarządzania relacjami z klientami (SZRK)*. Bieżące wyniki finansowe firmy wynikające z prowadzonej działalności są archiwizowane w *Hurtowni danych biznesowych (HDB)*, co pozwala na całościowe spojrzenie na prowadzony biznes wraz z dokonywaniem oceny realizacji strategii firmy.

Wszystkie wymienione systemy współpracują ze sobą, a dostęp do nich jest możliwy poprzez *Portal korporacyjny (PK)*. Pełne uprawnienia do wszystkich systemów posiada kierownictwo organizacji. Pozostali pracownicy mają ograniczony dostęp do wybranych funkcjonalności wymienionych powyżej systemów informatycznych. Ze względu na prywatny charakter danych przechowywanych w systemach, niektóre z nich są przechowywane w strefie intranetowej dostępnej za dwoma wewnętrznymi *Routerami* (*Serwer intranetowy* oraz *Serwer hurtowni danych*). Urządzenia strefy zewnętrznej, obejmujące *Serwer bazodanowy* oraz *Serwer webowy*, są połączone *Przełącznikiem* (ang. *switch*). Poszczególne komponenty zostały zainstalowane na następujących urządzeniach:

- ❑ Serwer webowy: PI;
- ❑ Serwer bazodanowy: BDZA, RDPD;
- ❑ Serwer intranetowy: MATP, MPAD, RDCH, PK, SZRK;
- ❑ Serwer hurtowni danych: HDWB, HDB, HDA.

Należy utworzyć diagram rozlokowania prezentujący architekturę i powiązania pomiędzy systemami informatycznymi, które znajdują się w opisanej powyżej firmie (rysunek 5.2).



Rysunek 5.2. Diagram rozlokowania portalu informacyjnego firmy prowadzącej badania sondażowe



Wskazówki dydaktyczne

Poszczególne systemy, podsystemy i moduły stanowią komponenty. Natomiast węzłami oznaczane są urządzenia (stereotyp device). Przerywane strzałki zwrócone są grotem w kierunku komponentu, z którego pobierają dane. Nie oznacza to jednak, że przepływ jest jednokierunkowy. Kierunek strzałki wskazuje jedynie, że jest tam kierowane zapytanie, na które w odwrotnym kierunku spływa odpowiedź.

Ć W I C Z E N I E

5.3

Hurtownie danych — agregacja danych podczas migracji z heterogenicznych baz źródłowych

Hurtownie danych należą do grupy systemów wspierających podejmowanie decyzji w organizacjach. Ich istotą jest integrowanie danych pochodzących z różnych źródeł, takich jak systemy CRM, ERP czy SCM — a następnie generowanie na ich podstawie raportów, zwykle z wykorzystaniem systemów klasy Business Intelligence. Badania rynku hurtowni danych wskazują, że wysyłane do hurtowni zapytania bardzo często dotyczą danych zagregowanych. Oznacza to, że dane mogą zostać wcześniej zagregowane w wybranych przekrojach w celu przyspieszenia dokonywania ich analiz.

Omawiany system informatyczny działa w firmie dystrybucyjnej zajmującej się sprzedażą sprzętu komputerowego na terenie całego kraju. W opisywanym przypadku hurtownia danych pozyskuje i integruje dane pochodzące z następujących systemów informatycznych:

- ❑ *ERP (Enterprise Resource Planning);*
- ❑ *CRM (Customer Relationship Management);*
- ❑ *WFM (Workflow Management);*
- ❑ *SCM (Supply Chain Management).*

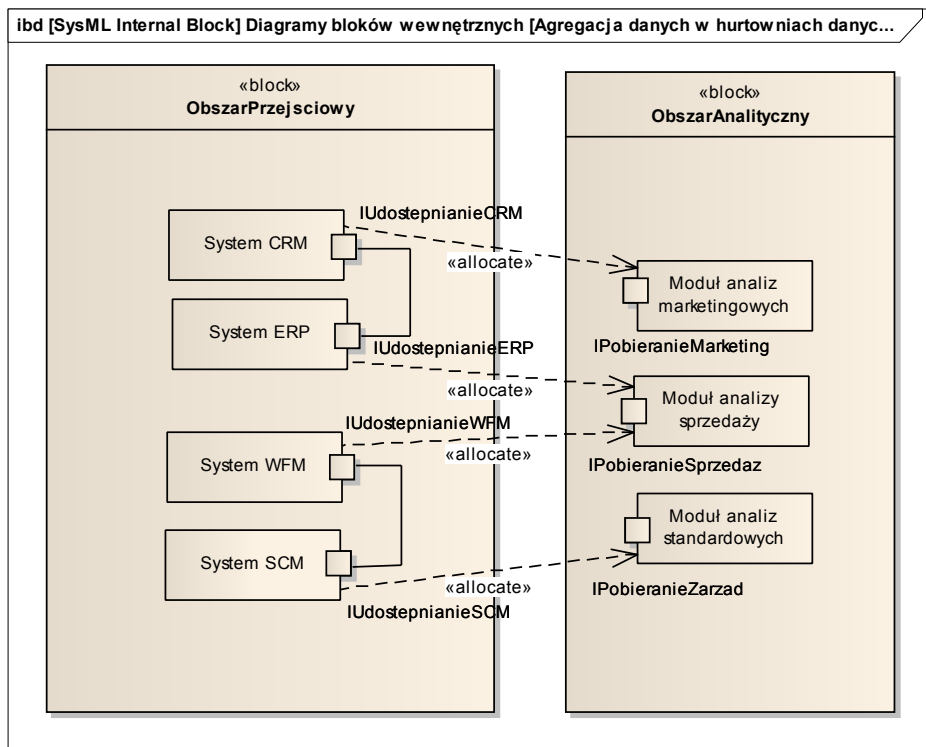
Gromadzone w wyżej wymienionych systemach dane są umiejscowione w *obszarze przejściowym*. Znajdują się tam wydzielone bazy danych będące źródłem dla hurtowni danych, z której systemy analityczne pobierają dane. Korzystanie z hurtowni danych planowane jest w trzech jednostkach organizacyjnych:

- ❑ dziale marketingu, dla *analiz marketingowych*, gdzie będzie badana sprzedaż produktów pod kątem nakładów na reklamę, w tym rozsyłanie ulotek reklamowych do potencjalnych klientów;
- ❑ dziale sprzedaży, w celu dokonywania *analiz sprzedaży*, w ramach których będą tworzone programy lojalnościowe dla najlepszych klientów na podstawie wielkości dokonywanych przez nich zakupów oraz częstości ich dokonywania;

- ❑ zarządzie organizacji korzystającym z *modułu analiz standardowych* pozwalającego na obserwację trendów zachodzących w sprzedaży, kosztów dystrybucji oraz nakładu na usługi marketingowe, co przełoży się również na obserwację kondycji finansowej firmy.

Ze względu na wrażliwy charakter danych (dział marketingu i sprzedaży nie mogą mieć dostępu do danych o kondycji finansowej firmy) przyjęto, że zostaną wydzielone trzy hurtownie tematyczne (tzw. *data mart*) gromadzące jedynie dane przeznaczone osobno dla każdego z trzech działów. Odpowiednio przyjęto następujące nazwy tych tematycznych wystąpień: Marketing, Sprzedaż oraz Zarząd.

Bazując na zamieszczonym opisie, projektant systemowy zidentyfikował kluczowe bloki i zilustrował ich elementy składowe z wykorzystaniem diagramu bloków wewnętrznych (rysunek 5.3).



Rysunek 5.3. Diagram bloków wewnętrznych agregacji danych podczas migracji z heterogenicznych baz źródłowych



Ć W I C Z E N I E

5.4 Aplikacja typu czat działająca w architekturze klient — serwer

Obecnie komputery osobiste oraz urządzenia mobilne coraz częściej wykorzystywane są do komunikacji w czasie rzeczywistym, gdzie bezpośrednio po wpisaniu komunikatu na jednym komputerze następuje jego odebranie na innym, zwykle odległym geograficznie komputerze.

Współczesne języki programowania, do których zaliczyć należy język Java, pozwalają na implementację w prosty sposób aplikacji w architekturze klient — serwer. W niniejszym ćwiczeniu program składa się z wymienionych poniżej pakietów (podsystemów) i zawartych w nich klas.

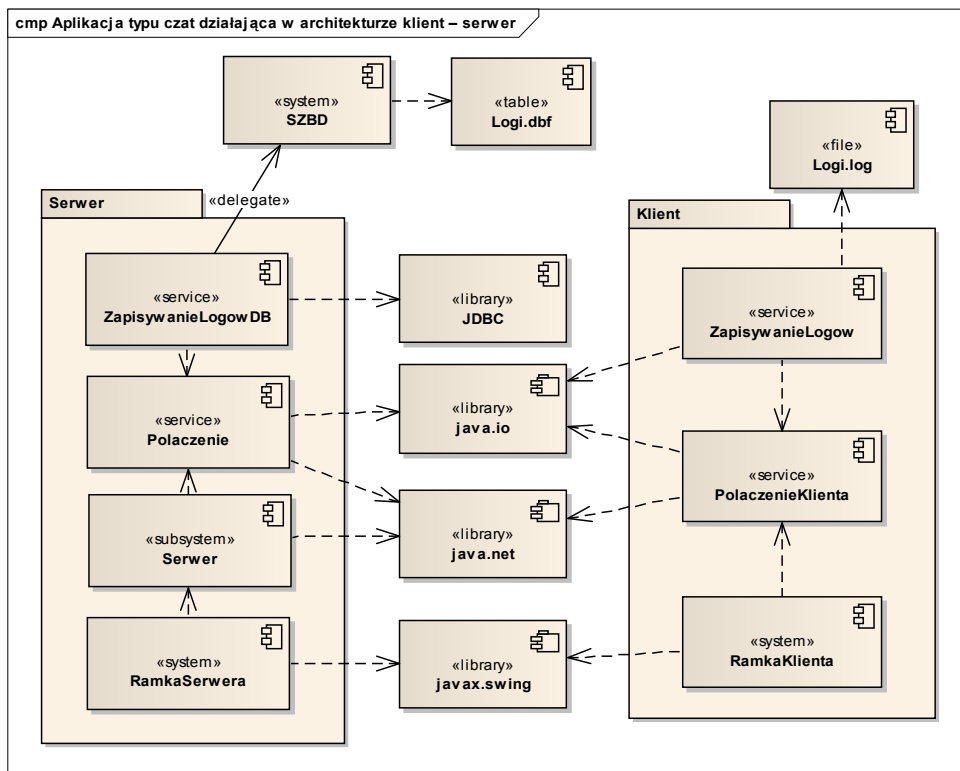
Serwer.jar — umożliwia nawiązywanie wielu połączeń pomiędzy klientami:

- ❑ *ZapisywanieLogowDB.java* — klasa pozwalająca na połączenie z bazą danych i zapisywanie w niej informacji o zdarzeniach mających miejsce na serwerze; niezbędna jest biblioteka *JDBC (Java DataBase Connectivity)*;
- ❑ *Polaczenie.java* — definiująca połączenie, korzystająca z bibliotek *java.io* oraz *java.net*;
- ❑ *Serwer.java* — umożliwiająca łączenie z wieloma klientami z wykorzystaniem klasy *Polaczenie.java* oraz biblioteki *java.net*;
- ❑ *RamkaSerwera.java* — główna aplikacja definiująca graficzny interfejs użytkownika, korzystająca z biblioteki *javax.swing*.

Klient.jar — umożliwia połączenie pomiędzy klientami, wykorzystując w tym celu podsystem *Serwera*:

- ❑ *ZapisywanieLogow.java* — klasa zapisująca informację o nawiązywanych połączeniach oraz treść prowadzonych rozmów w plikach tekstowych; niezbędna jest biblioteka *java.io*;
- ❑ *PolaczenieKlienta.java* — pozwalająca na nawiązywanie połączeń pomiędzy klientami z wykorzystaniem serwera; niezbędne są biblioteki *java.io* oraz *java.net*;
- ❑ *RamkaKlienta.java* — główna aplikacja kliencka wykorzystująca klasę *PolaczenieKlienta.java* w celu otwierania połączeń oraz biblioteki *javax.swing* dla obsługi interfejsu graficznego.

Na podstawie zamieszczonego opisu należy opracować diagram komponentów prezentujący aplikację klient — serwer (rysunek 5.4).



Rysunek 5.4. Diagram komponentów aplikacji typu czat działającej w architekturze klient – serwer



Wskazówki dydaktyczne

Zbiór wielu klas w języku Java kompilowany jest do pojedynczego pakietu, stąd obecność dwóch podstawowych pakietów — *Serwer.jar* oraz *Klient.jar*. Ze względu na rozbudowany charakter poszczególnych plików źródłowych klasy w ramach pakietów opisane są przez komponenty. W niniejszym ćwiczeniu zrezygnowano z używania rozszerzeń plików w nazwach komponentów (.java). Na zewnątrz systemu funkcjonuje system zarządzania bazą danych (SZBD), do którego dostęp jest realizowany poprzez usługę *ZapisywanieLogowDB*. Za pomocą tego pakietu odbywa się zapis i odczyt danych wykonywany na tabeli o nazwie *Logi.dbf*, zatem SZBD jest zależny od istnienia tej tabeli. Podobna sytuacja ma miejsce z plikiem generowanym przez aplikację kliencką (*Logi.log*), który nie jest składnikiem pakietu Klienta, a usługa *ZapisywanieLogow* zależy od istnienia tego pliku. W powyższym opisie celowo pominięto szczegóły implementacyjne, takie jak obsługa zdarzeń za pomocą interfejsu *ActionListener*, czy też obsługa wielowątkowości za pomocą interfejsu *Runnable* lub klasy *Thread*. Istotą tego diagramu jest przedstawienie podstawowych komponentów, z jakich złożony jest system, oraz wzajemnych powiązań pomiędzy tymi komponentami. Biblioteki w języku Java są reprezentowane jako pakiety.

Ć W I C Z E N I E

5.5

Sieć hurtowni odzieżowych z oddziałami terenowymi

Sprawne zarządzanie współczesną firmą wymaga szybkiego reagowania na zmiany pojawiające się w jej otoczeniu. Do tych zmian należy zaliczyć między innymi zwiększenie popytu na oferowane produkty, czy też odchodzenie klientów z przedsiębiorstwa. Warto zatem wyprzedzać pojawiające się na rynku trendy oraz przewidywać możliwe posunięcia klientów. W tym celu wykorzystuje się hurtownię danych wraz z innymi rozwiązaniami wchodzącymi w skład szeroko pojętych systemów klasy Business Intelligence.

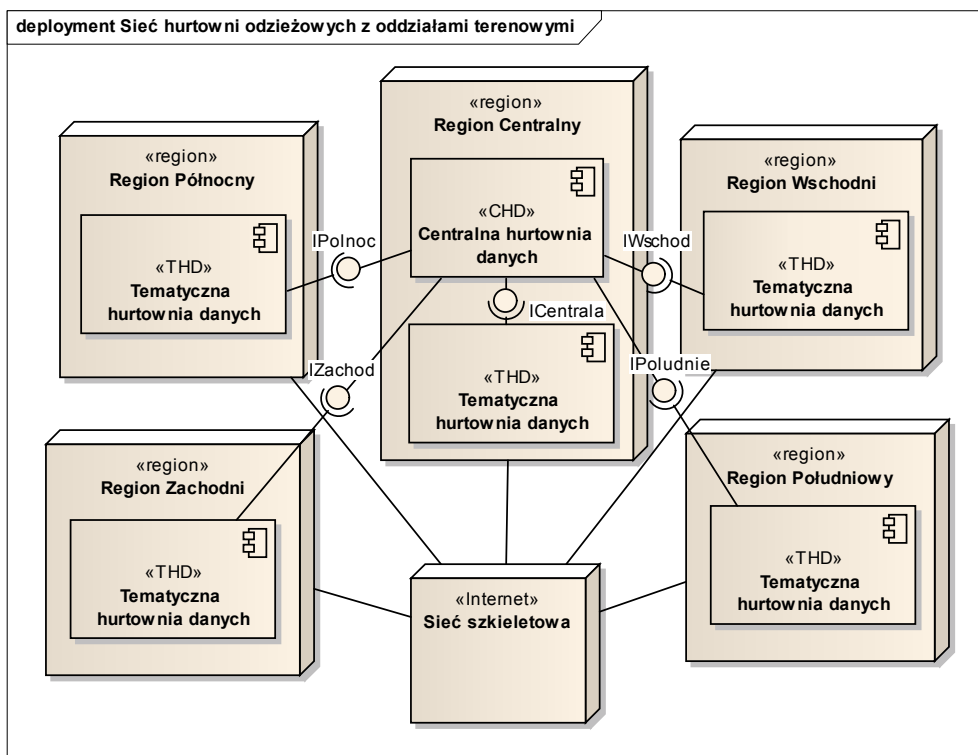
Firma będąca siecią hurtowni odzieżowych próbuje zwiększyć przewagę konkurencyjną poprzez wdrożenie hurtowni danych. Jednak jej struktura jest inna, niż ma to miejsce w przypadku typowych firm z tej branży charakteryzujących się daleko idącą centralizacją. Mianowicie została ona podzielona na pięć mniejszych obszarów regionalnych, które ściśle ze sobą współpracują, jednak niezależnie zbierają zamówienia na produkty i prowadzą rozliczenia finansowe. Dopiero po zatwierdzeniu zleceń informacje te przepływają do centrali firmy. Nazwy poszczególnych oddziałów terenowych oraz miejscowości, w których znajdują się ich siedziby, są następujące:

- ☐ *Region Północny* — Gdańsk;
- ☐ *Region Zachodni* — Szczecin;
- ☐ *Region Centralny* — Warszawa;
- ☐ *Region Wschodni* — Lublin;
- ☐ *Region Południowy* — Kraków.

Poszczególne regiony mają posiadać tzw. zależne *Tematyczne hurtownie danych*. Spływające do nich dane powinny być przekazywane raz w tygodniu do *Centralnej hurtowni danych* zlokalizowanej w Warszawie. Połączenia pomiędzy poszczególnymi oddziałami terenowymi a centralą mają wykorzystywać technologię VPN w istniejącej infrastrukturze szkieletowej sieci. Dodatkowo oddziały terenowe będą mogły się ze sobą wzajemnie komunikować za pomocą rozwiązań ekstranetowych (VPN) z koniecznością ręcznej autoryzacji.

Zespół wdrożeniowy został obarczony zadaniem utworzenia diagramu rozlokowania dla opisanego systemu (rysunek 5.5).





Rysunek 5.5. Diagram rozlokowania sieci hurtowni odzieżowych z oddziałami terenowymi

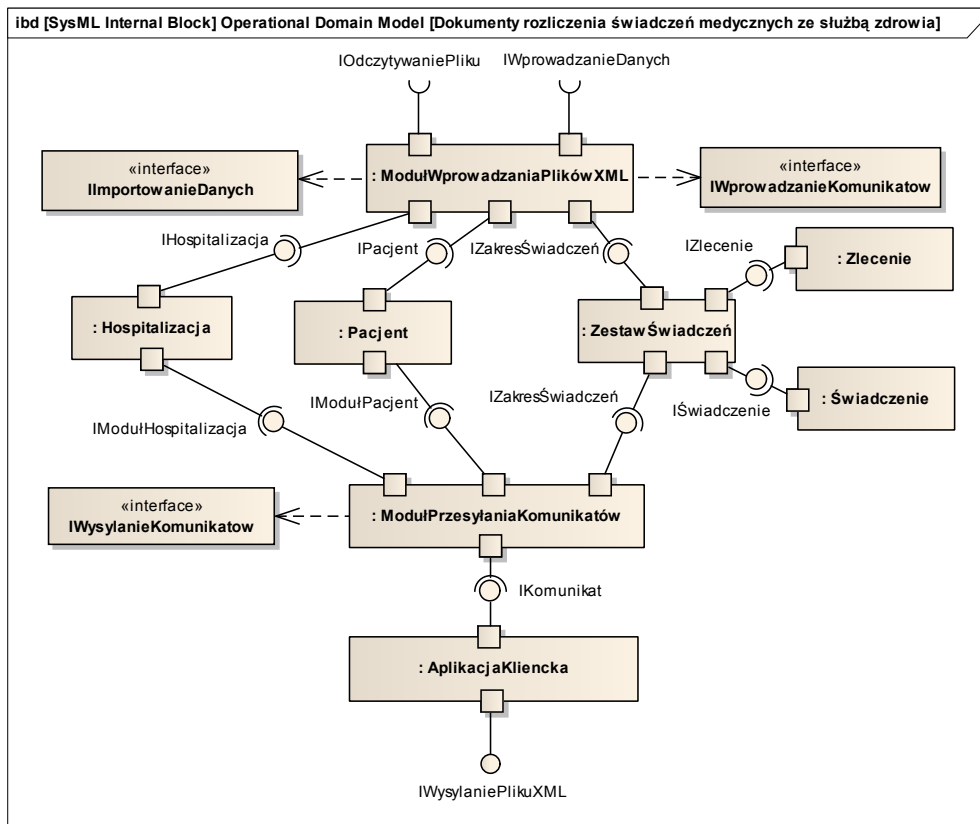
Ć W I C Z E N I E

5.6 Dokumentacja rozliczenia świadczeń medycznych ze służbą zdrowia

W celu rozliczenia świadczeń medycznych zakłady opieki zdrowotnej są zobligowane do przysyłania danych w formacie XML do centralnego organu zajmującego się przydzielaniem środków na usługi medyczne. Wzorce komunikatów są publikowane na stronie internetowej. Organem centralnym jest Narodowy Fundusz Zdrowia. Komunikaty można podzielić następująco:

- ☐ Zestaw świadczeń (dane zestawu);
- ☐ Zlecenie;
- ☐ Hospitalizacja;
- ☐ Pacjent;
- ☐ Świadczenie.

Należy zaproponować diagram bloków wewnętrznych ilustrujący części oraz interfejsy bloku dedykowanego rozliczaniu usług medycznych przez centralny organ administracji rządowej (rysunek 5.6).



Rysunek 5.6. Diagram bloków wewnętrznych rozliczania usług medycznych



Wskazówki dydaktyczne

W niniejszym ćwiczeniu przyjęto założenie, że do obsługi konkretnych komunikatów przeznaczony jest WebService o nazwie *Obsługa świadczeń*. W ramach tej usługi webowej istnieją interfejsy udostępniające, które są implementowane w oprogramowaniu klienckim przeznaczonym do przesyłania tych komunikatów.

Ć W I C Z E N I E

5.7**Architektura bankomatu**

Współczesne bankomaty pozwalają nie tylko na wypłatę gotówki, ale również na wykonywanie szeregu innych czynności. Wśród elementów składowych bankomatu można wyróżnić:

- ☐ *Komputer sterujący;*
- ☐ *Moduł wypłacający;*
- ☐ *Moduł depozytowy do wpłacania;*
- ☐ *Czytnik kart płatniczych;*
- ☐ *Czytnik banknotów wpłacanych;*
- ☐ *Monitor;*
- ☐ *Klawiatura bankomatowa;*
- ☐ *Sejf do przechowywania banknotów;*
- ☐ *Drukarka pokwitowań transakcji;*
- ☐ *Kamera;*
- ☐ *Moduł szyfrujący wysyłaną informację o transakcji;*
- ☐ *Urządzenie telekomunikacyjne do wysyłania informacji o transakcjach.*

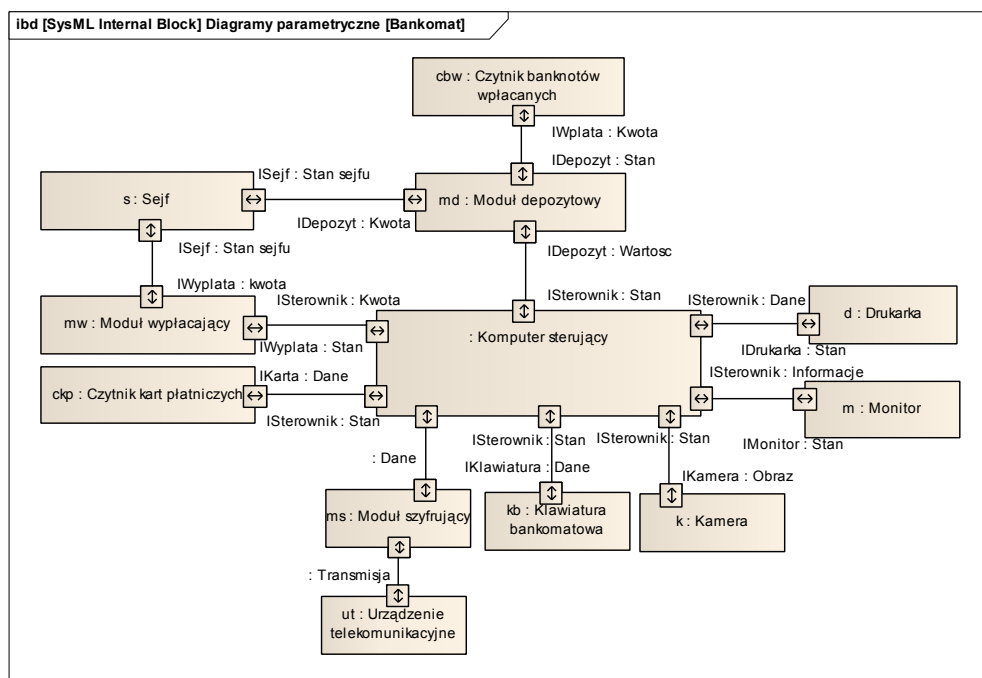
W powyższym opisie celowo zostały pominięte integralne części komputera sterującego, takie jak magistrala, pamięć czy kontrolery systemowe.

Na podstawie wymienionej listy należy opracować diagram bloków wewnętrznych przedstawiający budowę bankomatu oraz zasoby przesyłane pomiędzy poszczególnymi jego częściami (rysunek 5.7).

**Wskazówki dydaktyczne**

Centralną częścią bankomatu jest komputer sterujący, który nie posiada identyfikatora. Zgodnie z założeniami języka SysML identyfikatory są przyporządkowywane tylko modułom, które nie są integralną częścią bankomatu i mogą zostać wymienione.





Rysunek 5.7. Diagram bloków wewnętrznych architektury bankomatu

5.2. Zadania

Zadanie 5.1.

Na podstawie przypadku dowolnie wybranego sklepu internetowego utwórz diagram rozłokowania oraz osadź w nim artefakty. Należy wziąć pod uwagę fakt, że sklep internetowy może jednocześnie korzystać z serwera WWW, serwera aplikacji oraz serwera bazodanowego.

Zadanie 5.2.

Opracuj diagram komponentów dla portalu firmy, który zawiera:

- ☐ dane kontaktowe;
- ☐ charakterystykę działalności;
- ☐ reklamę firmy;
- ☐ wyszukiwarkę;

- ☐ forum;
- ☐ mapę witryny;
- ☐ blog;
- ☐ asystę techniczną;
- ☐ aktualności;
- ☐ formularz rejestracyjny;
- ☐ szkolenia;
- ☐ formatkę logowania;
- ☐ nagrody i wyróżnienia przyznane firmie;
- ☐ osiągnięcia firmy;
- ☐ prezentacje profilu firmy;
- ☐ informacje dla inwestorów.

Zadanie 5.3.

Wraz z upowszechnieniem się Internetu coraz więcej przedsiębiorstw publikuje swoją ofertę poprzez serwisy internetowe. W tym celu wykorzystuje się serwery dedykowane lub wirtualne, wykupywane w profesjonalnych firmach zajmujących się hostingiem stron WWW. W ramach hostingu użytkownik może wykupić wraz z serwerem WWW następujące usługi:

- ☐ bazę danych MySQL lub PostgreSQL;
- ☐ dostęp do plików poprzez FTP;
- ☐ konta poczty elektronicznej;
- ☐ adres(y) WWW;
- ☐ serwer PHP, Python lub PERL.

Poszczególne serwery wybranej firmy hostingowej połączone są wysokoprępową siecią światłowodową wykorzystującą przełączniki klasy Cisco 6500. Poszczególne serwery PHP, Python oraz PERL są uruchomione na trzech wirtualnych serwerach ulokowanych na pojedynczym fizycznym serwerze o nazwie *Serwer języków programowania*. Konta poczty elektronicznej obsługiwane są przez *Serwer pocztowy* będący fizycznym węzłem w ramach infrastruktury firmy hostingowej. Adresy WWW (nazwy domenowe) są delegowane na dwóch serwerach DNS (*Dynamic Name System*), odpowiednio *Serwer DNS1* oraz *Serwer DNS2*. *Serwer plików* umożliwia dostęp przez usługę FTP z opcją włączenia bezpieczeństwa za pomocą protokołów SFTP lub SCP. Najistotniejszy z punktu widzenia firmy *Serwer WWW* działający w oparciu o technologię Apache pozwala na bezpośredni dostęp z Internetu do przechowywanych serwisów internetowych. Każda strona jest uruchamiana na wirtualnym serwerze konfigurowanym bezpośrednio w usłudze Apache. Pomiedzy Internetem a serwerem WWW znajduje się router będący jednocześnie

Równoważnikiem obciążenia (ang. *load balancer*). Jego rolą jest optymalny rozkład ruchu przychodzącego i kierowanie go do poszczególnych węzłów sieciowych wymienionych powyżej.

Opracuj diagram rozlokowania z osadzonymi komponentami dla firmy hostingowej, uwzględniając wszystkie interfejsy udostępniające i pozyskujące, jakie mogą zostać umiejscowione pomiędzy poszczególnymi aplikacjami uruchamianymi na serwerach firmy.

Zadanie 5.4.

Jak podkreślają (Kang, Yoo i Kim, 2006), współczesne zamożne kraje i społeczeństwa dotyka problem starzenia się społeczeństwa. Zdrowie starszych osób często szwankuje i musi być pod stałą kontrolą. Zadanie to może być znakomicie wspomagane dzięki wykorzystaniu odpowiednich urządzeń elektronicznych. Powstała nowa dziedzina wiedzy medycznej — telemedycyna, która jest ukierunkowana na pomoc w takich przypadkach. Urządzeniem służącym monitorowaniu zdrowia osób starszych jest WIHMD (*Wrist-Worn Integrated Health Monitoring Instrument with a Tele-Reporting Device*).

Urządzenie WIHMD składa się z sześciu modułów pomiaru funkcji życiowych, wśród których znajdują się:

- ☐ detektor upadku;
- ☐ jednokanałowy elektrokardiogram;
- ☐ ciśnieniomierz do nieinwazyjnego pomiaru ciśnienia krwi;
- ☐ pulsoksymetr do mierzenia zawartości tlenu w krwi;
- ☐ moduł mierzenia częstości oddechu w określonej jednostce czasu;
- ☐ termometr do mierzenia temperatury ciała.

Celem użytkowania tego urządzenia jest wysyłanie sygnałów o aktualnym stanie zdrowia i miejscu pobytu monitorowanej osoby za pośrednictwem telefonii komórkowej do lekarza prowadzącego. Połączony z WIHMD system w sytuacjach zagrożenia życia wysyła od lekarza do pacjenta informacje wskazujące niezbędny zabieg lub zażycie lekarstwa stosownie do otrzymywanych komunikatów.

Na podstawie powyższego opisu utwórz diagram bloków wewnętrznych, specyfikując interfejsy w postaci bloku.

Zadanie 5.5.

Utwórz diagram rozlokowania, który zawiera komponenty prywatnej chmury (ang. *cloud computing*). Powinien on zawierać między innymi współdzieloną moc obliczeniową, urządzenia komputerowe i sieciowe, zasoby dyskowe, pamięć, oprogramowanie systemowe, aplikacje. W celu uzyskania poszerzonych informacji na temat elementów składowych prywatnej chmury skorzystaj z opracowań dostępnych na stronie: <http://www.microsoft.com/virtualization/>.

Zadanie 5.6.

Zarówno poszczególne koncerny samochodowe, jak i Unia Europejska wspierają rozwiązania w zakresie konstrukcji i użytkowania inteligentnych samochodów. Poza standardowymi funkcjami poprzez interfejs obsługi inteligentnego samochodu i jego system sterowania głosem wykonywane będą między innymi następujące funkcje:

- ☐ wykrywanie pieszych;
- ☐ rozpoznawanie znaków drogowych;
- ☐ sprawne prowadzenie rozmów telefonicznych (zestawy głośnomówiące);
- ☐ prowadzenie korespondencji i wyszukiwanie informacji w Internecie poprzez wykorzystanie technologii rozpoznawania mowy;
- ☐ dyktowanie treści wiadomości i edytowanie jej przez zestaw komend głosowych;
- ☐ zróżnicowanie dostępu do funkcji podczas rutynowej jazdy, jazdy na autostradzie, w korku i podczas postoju;
- ☐ odczytywanie przez syntezytor mowy wiadomości elektronicznych ze skrzynki pocztowej;
- ☐ odsłuchiwanie treści otrzymywanych wiadomości;
- ☐ korzystanie z wyszukiwarki sklepów i restauracji znajdujących się w pobliżu;
- ☐ wyszukiwanie informacji adresowych i danych na temat cen paliw w pobliskich stacjach benzynowych;
- ☐ komunikacja ze znajdującymi się w pobliżu pojazdami i infrastrukturą drogową.

Opracuj diagram bloków wewnętrznych przedstawiający modułową konstrukcję inteligentnego samochodu, uwzględniając wymienioną powyżej funkcjonalność. W diagramie należy zastosować specyfikację portów transmisyjnych.

Zadanie 5.7.

Wśród systemów informatycznych wspomagających zarządzanie firmą kluczowe znaczenie posiadają systemy ERP (*Enterprise Resource Planning*) i CRM (*Customer Relationship Management*). System CRM wspiera proces zarządzania relacjami z klientami poprzez oferowanie następujących podstawowych funkcjonalności:

- ☐ Marketing;
- ☐ Sprzedaż;
- ☐ Serwis;
- ☐ Call Center.

Z systemem CRM może być powiązany system klasy Business Intelligence (BI), którego celem jest przekazywanie kierownictwu wyższego szczebla raportów w postaci analiz i podsumowań na temat aktualnej sytuacji w zakresie sprzedaży produktów.

W opisywanym przypadku przyjęto założenie, że system BI korzysta bezpośrednio z bazy danych systemu CRM, dokonując miesięcznych podsumowań sprzedaży i przekazując je w sposób półautomatyczny kierownictwu. Brak pełnej automatyzacji wynika z konieczności przeprowadzenia weryfikacji przez głównego księgowego raportu przekazywanego kierownictwu.

Raport wygenerowany za pośrednictwem systemu BI zawiera następujące informacje prezentowane w formie tabelarycznej oraz graficznej:

- ❑ łączną kwotę zapłaconą za produkty sprzedane w ubiegłym miesiącu;
- ❑ liczbę sprzedanych produktów wyrażoną w odpowiedniej dla danego produktu jednostce miary;
- ❑ listę produktów, które sprzedawały się najlepiej w wybranym okresie, wraz z wielkością nakładów finansowych poniesionych na ich reklamę oraz średnim dziennym przychodem wynikającym ze sprzedaży danego produktu i łącznym zyskiem netto;
- ❑ listę rankingową produktów sporządzoną na podstawie relacji nakładów poniesionych na produkcję i marketing do przychodów uzyskanych ze sprzedaży produktu;
- ❑ zysk lub stratę netto oraz brutto w danym okresie;
- ❑ szacunki dotyczące sprzedaży w najbliższym miesiącu wraz z listą produktów, które mogą się sprzedawać w przyszłości.

Jak wspomniano powyżej, raport nie jest przekazywany bezpośrednio kierownictwu. Za sporządzenie raportu odpowiedzialny jest główny księgowy, który korzysta z pulpitu systemu BI, klikając odpowiednie łącze generowania raportu i określając jego parametry czasowe. Następnie na podstawie wcześniejszych zestawień wyników sprzedaży weryfikuje poprawność danych, po czym przekazuje je kierownictwu organizacji.

Opracuj diagram bloków wewnętrznych prezentujący wymienione powyżej moduły systemu.

Diagramy parametryczne

6.1. Ćwiczenia

Ć W I C Z E N I E

6.1 Racionalizacja wykorzystania energii elektrycznej

Firma produkcyjna zużywa rocznie kilkaset megawatów (MW) energii elektrycznej. Opłata za energię jest zależna od wielu czynników, wśród których wyróżnić należy między innymi długość użytkowania produktu. Dla ograniczenia kosztów z tym związanych budowany jest system wspomagania optymalnego wykorzystania energii. Celem tego systemu jest korygowanie zużycia energii poprzez właściwe określenie zasobów zużywających energię i pory ich użytkowania.

Jak wskazano w opracowaniu (Energa-Operator, 2011), typowy wzór na korzystanie z energii elektrycznej przedstawia się następująco:

$$O_{\text{poi}} = S_{\text{SVn}} * P_i + \sum_{k=1}^r S_{\text{ZVnk}} * E_{\text{pik}} + S_{\text{oSj}} * E_{\text{ok}} + S_{\text{op}} * P_i + O_a$$

gdzie:

- ❑ O_{poi} — opłata za dystrybucję energii elektrycznej dla danego odbiorcy w zł;
- ❑ S_{SVn} — składnik stały stawki sieciowej w zł/kW/miesiąc lub w zł/miesiąc dla odbiorców energii elektrycznej z grup taryfowych G;
- ❑ P_i — moc umowna z uwzględnieniem współczynników korygujących określonych w umowie danego odbiorcy (w kW) lub liczba miesięcy dla odbiorców energii elektrycznej z grup taryfowych G;

- ❑ S_{ZVnk} — składnik zmienny stawki sieciowej dla strefy czasowej „k” w zł/MWh lub zł/kWh;
- ❑ E_{pik} — ilość energii elektrycznej pobranej z sieci przez odbiorcę w strefie czasowej k w MWh lub kWh;
- ❑ S_{oSj} — stawka jakościowa w zł/MWh lub w zł/kWh;
- ❑ E_{ok} — ilość energii elektrycznej zużytej przez odbiorcę oraz innych odbiorców przyłączonych do jego sieci, korzystających z krajowego systemu elektroenergetycznego — w MWh lub kWh;
- ❑ S_{op} — stawka opłaty przejściowej w zł/kW/miesiąc lub w zł/miesiąc dla odbiorców energii elektrycznej z grup taryfowych G;
- ❑ O_a — opłata abonamentowa w zł;
- ❑ r — liczba rozliczeniowych stref czasowych (1, 2 lub 3).

W przypadku przekroczenia energii biernej pobranej w okresie rozliczeniowym nadwyżka rozliczana jest zgodnie ze wzorem:

$$O_b = k * C_{rk} * \left(\sqrt{\frac{1 + \text{tg}^2 FI}{1 + \text{tg}^2 FI_0}} - 1 \right) * A$$

gdzie:

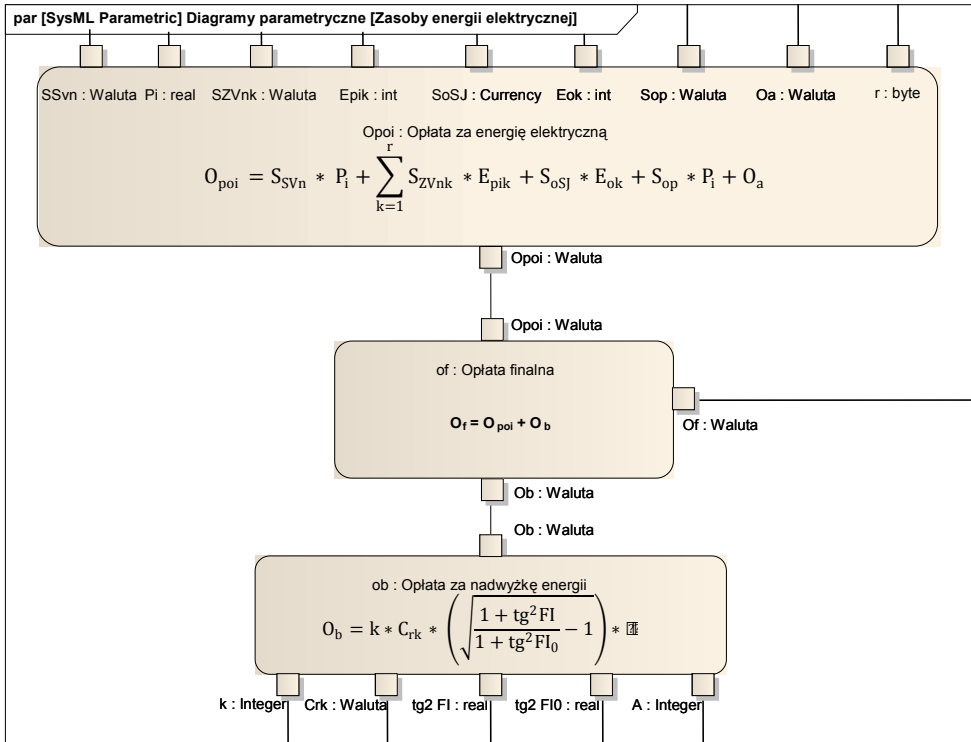
- ❑ O_b — opłata za nadwyżkę energii biernej wyrażona w zł;
- ❑ k — krotność ceny C_{rk} ;
- ❑ C_{rk} — cena energii elektrycznej obowiązująca w dniu zatwierdzenia taryfy, wyrażona w zł/MWh lub zł/kWh;
- ❑ $\text{tg}^2 FI$ — współczynnik mocy wynikający z pobranej energii biernej;
- ❑ $\text{tg}^2 FI_0$ — umowny współczynnik mocy określony w warunkach przyłączenia lub w umowie;
- ❑ A — energia czynna pobrana całodobowo lub dla strefy czasowej, w której prowadzona jest kontrola poboru energii biernej, wyrażona w MWh lub kWh.

Należy utworzyć diagram parametryczny umożliwiający firmie optymalne planowanie wykorzystania energii (rysunek 6.1).

Ć W I C Z E N I E

6.2 Selekcja systemu bazodanowego

Duża firma z północy Polski do tej pory nie przykładła większej wagi do specyfiki użytkowanych systemów bazodanowych. Zwykle wraz z wdrożeniem nowego systemu, który miał usprawnić pracę w firmie, kupowano licencję na nową bazę danych.



Rysunek 6.1. Diagram parametryczny planowania zużycia energii elektrycznej

Zarząd firmy podjął decyzję, że sumaryczny koszt utrzymywania tych różnorodnych systemów musi zostać zmniejszony. Zadaniem działu informatyki jest utworzenie modelu decyzyjnego, który pozwoli na podstawie różnych parametrów wybrać system bazodanowy odpowiadający wszystkim potrzebom firmy.

Jednym z kryteriów wyboru systemu jest cena, chociaż nie przypisano jej zbyt dużej wagi w całości oceny. Należy podkreślić, że na *cenę zakupu* składa się zarówno *faktyczna cena netto produktu*, jak i *cena licencji* oraz *cena asysty technicznej* przez najbliższe 5 lat.

Kryteria wyboru systemu podzielono na obligatoryjne oraz opcjonalne, które z kolei można różnicować dalej na ilościowe i jakościowe. Do kryteriów obligatoryjnych jakościowych należy zaliczyć:

- ❑ obsługę rozszerzeń Business Intelligence za pomocą implementacji wielowymiarowych struktur w technologii ROLAP;
- ❑ 64-bitową architekturę systemu;
- ❑ wsparcie dla wirtualizacji;
- ❑ automatyczną replikację danych.

Do kryteriów obligatoryjnych ilościowych, w przypadku których określono minimalną wartość, jaka musi zostać spełniona, zaliczono:

- ☐ obsługę minimum ośmiu procesorów — 10 punktów, z czego następne 2 punkty za obsługę każdego kolejnego procesora;
- ☐ wielkość bazy danych — minimum 5 TB — 20 punktów, z czego 1 punkt za każde kolejne 500 GB.

Wśród kryteriów opcjonalnych jakościowych wymieniono następujące:

- ☐ pracę w systemie Linux — 5 punktów;
- ☐ obsługę przez interfejs WWW — 2 punkty.

Ostatnią grupę kryteriów — opcjonalnych ilościowych — stanowią:

- ☐ licencjonowanie na wiele serwerów — 10 punktów za każdy serwer, w przypadku licencji na jeden serwer 0 punktów;
- ☐ wsparcie dla innych narzędzi analitycznych niż ROLAP — za każdy kolejny typ (MOLAP, HOLAP) postanowiono przyznać 2 punkty;
- ☐ cenę — darmowe produkty: 20 punktów; płatne: za każde kolejne 2000 zł odjęcie 1 punktu.

Należy opracować diagram definiowania bloków, który stanowi podstawę diagramu parametrycznego, w celu czteroetapowej ewaluacji systemów baz danych (rysunek 6.2).

Ć W I C Z E N I E

6.3

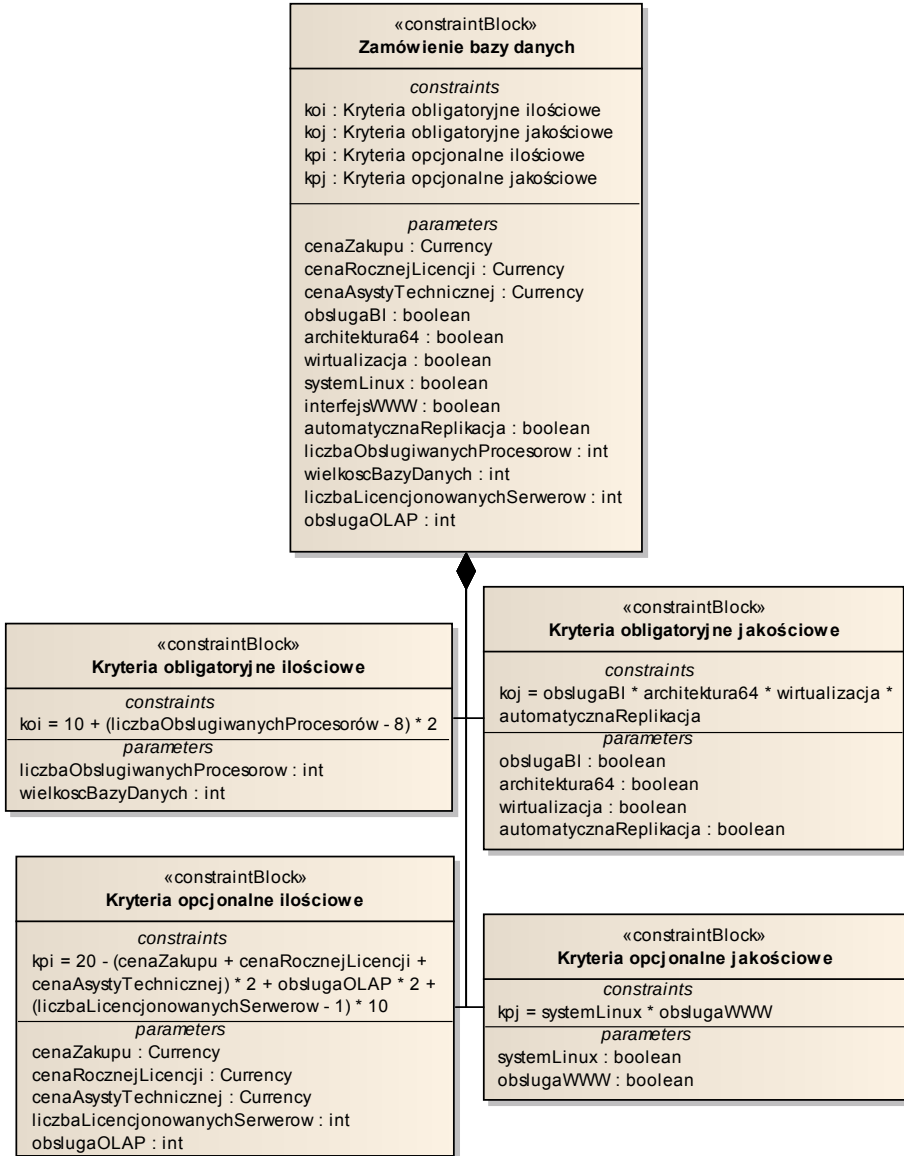
Warianty wyboru systemu bazodanowego

Dla kryteriów wyboru systemu bazodanowego opisanego w ćwiczeniu 6.2 przyjęto następujące wagi:

- ☐ obligatoryjne jakościowe — muszą zostać spełnione, w innym przypadku przyznawanych jest 0 punktów z tytułu całego kryterium;
- ☐ obligatoryjne ilościowe — brak spełnienia warunku minimalnego skutkuje przyznaniem 0 punktów za całość, w innym przypadku 30% całości;
- ☐ opcjonalne ilościowe — 50% całości;
- ☐ opcjonalne jakościowe — 20% całości.

Wynikiem powinna być ocena ilościowa wyrażona w punktach, z dokładnością do 2 miejsc po przecinku. Wartość minimalna, jaka może być uzyskana przez produkt, wynosi 0 punktów. Wartość maksymalna nie jest określona ze względu na występowanie kryteriów ilościowych.

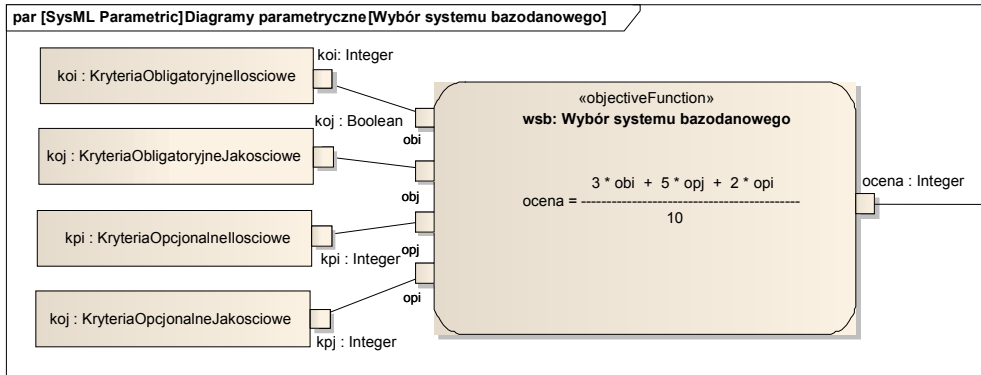
bdd [SysML Block Definition] Diagramy parametryczne [Wybór systemu bazodanowego]



Rysunek 6.2. Diagram definiowania bloków jako podstawa diagramu parametrycznego wyboru systemu bazodanowego

Należy opracować diagram parametryczny opisujący funkcję celową wyboru systemu bazodanowego (rysunek 6.3), zakładając przyjęcie następujących oznaczeń:

- ❑ *koi* — Kryteria obligatoryjne ilościowe;
- ❑ *k oj* — Kryteria obligatoryjne jakościowe;
- ❑ *kpi* — Kryteria opcjonalne ilościowe;
- ❑ *k pj* — Kryteria opcjonalne jakościowe.



Rysunek 6.3. Diagram parametryczny wariantów wyboru systemu bazodanowego

Ć W I C Z E N I E

6.4

Algorytm drzewa opinającego w sieciach komputerowych

Współczesne sieci komputerowe wspierane są przez liczne mechanizmy usprawniające ich funkcjonowanie. Należy do nich zaliczyć algorytm drzewa opinającego (ang. *spanning tree*), którego celem jest zapobieganie powstawaniu pętli w sieciach opartych na przełącznikach.

Algorytm ten używany jest w celu dokonania wyboru przełącznika będącego nadrzędnym w stosunku do innych w drzewie opinającym. Przełącznik ten, nazywany *root-bridgem*, ma za zadanie zarządzanie połączeniami w sieci poprzez ich filtrowanie (inaczej blokowanie) lub udostępnianie. Jest on wybierany na podstawie numeru BID (*Bridge Identifier*), który składa się z:

- ❑ ustawionego priorytetu przełącznika, domyślnie jego wartość wynosi 32768;
- ❑ numeru wirtualnej sieci (ang. *extended VLAN ID*);
- ❑ adresu MAC.

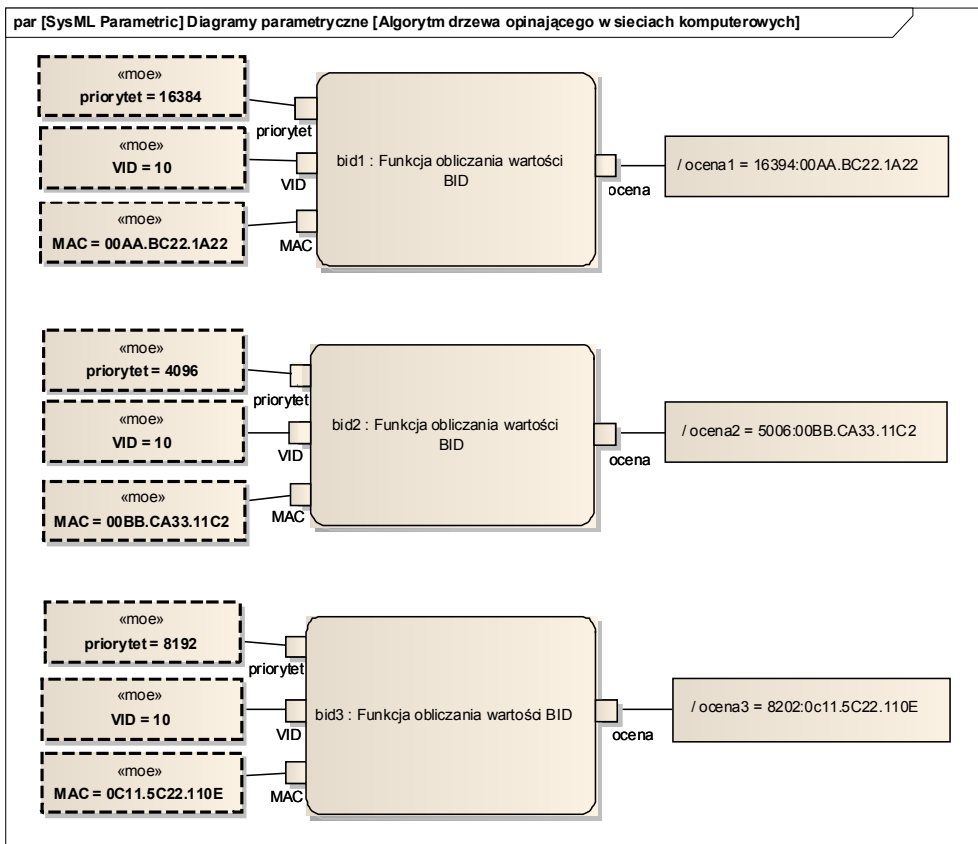
Numer BID może wyglądać następująco: 32769:0A2E.3C11.2233, gdzie 32769 to suma priorytetu (32768) oraz numeru VLAN (1), a 0A2E.3C11.2233 to adres MAC.

Rolę *root-bridge'a* będzie pełnił przełącznik, który ma najniższy BID. Jeżeli zatem priorytety na wszystkich przełącznikach są identyczne, to jako *root-bridge* zostanie wybrany ten z najniższym adresem MAC.

Należy opracować diagram parametryczny zawierający analizę wariantową dla trzech przełączników z następującymi parametrami:

- ❑ przełącznik A: VLAN ID: 10, priorytet: 16384, adres MAC: 00AA.BC22.1A22;
- ❑ przełącznik B: VLAN ID: 10, priorytet: 4096, adres MAC: 00BB.CA33.11C2;
- ❑ przełącznik C: VLAN ID: 10, priorytet: 8192, adres MAC: 0C11.5C22.110E.

Po opracowaniu diagramu (rysunek 6.4) należy ustalić, który z przełączników zostanie wybrany jako *root-bridge*.



Rysunek 6.4. Diagram parametryczny wariantów drzewa opinającego w sieciach komputerowych



Wskazówki dydaktyczne

Analiza wariantowa ma na celu wybranie najlepszego z możliwych wariantów. W tym ćwiczeniu analiza umożliwiła zweryfikowanie, który z przełączników został wybrany jako *root-bridge*. Istotą tego diagramu jest umożliwienie wglądu w wartości parametrów wejściowych osobom oceniającym poszczególne warianty.

Stereotyp moe zastosowany na rysunku 6.4 oznacza miarę efektywności (ang. *measure of effectiveness*).

Ć W I C Z E N I E

6.5

Wybór tras dla pakietów przesyłanych na poziomie dostawcy usług w sieci Internet

Jednymi z podstawowych urządzeń sieciowych działających w sieci Internet są routery. Urządzenia te wykorzystują algorytmy routingu do wyboru najlepszych tras dla pakietów zawierających dane przesyłane w sieci Internet. Istnieją różne kryteria wyboru najlepszych tras, wśród których najpopularniejsze to:

- ❑ dla protokołu routingu RIP — metryką jest najmniejsza liczba routerów na drodze do celu;
- ❑ w protokole routingu OSPF — domyślnie na routerach firmy Cisco metryka jest liczona na podstawie skumulowanej wartości pasma na łączach do celu;
- ❑ w przypadku protokołu routingu EIGRP — domyślnie pasmo oraz opóźnienia, mogą jednak zostać włączone również parametry dotyczące niezawodności oraz obciążenia.

Najbardziej złożony algorytm jest wykorzystywany przez protokół EIGRP. Algorytm ten można przedstawić w postaci następującego wzoru:

$$[A] \text{ Metryka} = \left[\left(K1 * \text{pasma} + \frac{K2 * \text{pasma}}{256 - \text{obciążenie}} + K3 * \text{opóźnienie} \right) * \frac{K5}{K4 + \text{niezawodność}} \right] * 256$$

gdzie:

- ❑ K1, K2, K3, K4 oraz K5 — wagi ustalane przez administratora, przyjmują wartości od 0 do 255;
- ❑ pasmo — stała wartość liczona na podstawie najniższego pasma przypisanego do jednego z łączy na trasie pomiędzy źródłem a celem;
- ❑ opóźnienie — stała wartość liczona jako skumulowana na drodze do celu;
- ❑ obciążenie — wartość liczona dynamicznie na podstawie bieżącego obciążenia trasy do celu;
- ❑ niezawodność — wartość liczona dynamicznie na podstawie bieżącej liczby utraconych pakietów na trasie do celu.

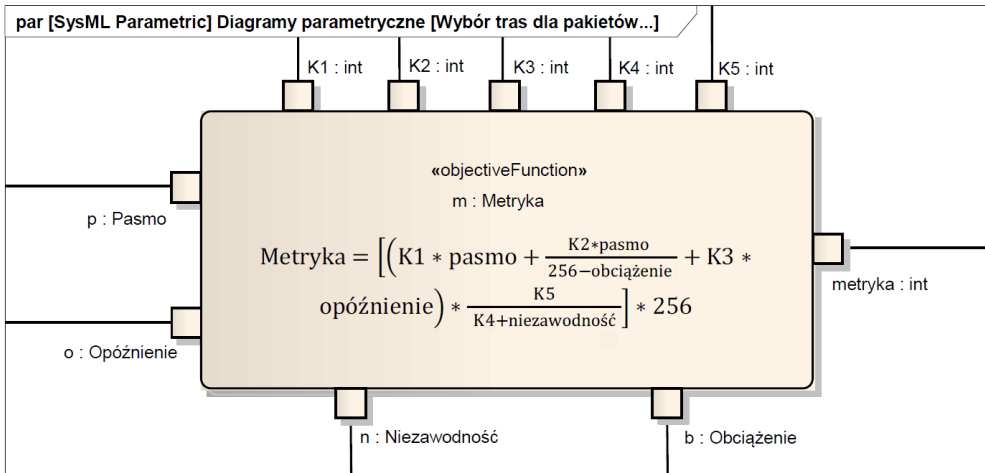
Domyślnie ustawione są następujące wagi: $K1 = 1$, $K2 = 0$, $K3 = 1$, $K4 = 0$ oraz $K5 = 0$. W takim przypadku wzór będący podstawą obliczenia metryki przyjmie następującą uproszczoną postać:

$$[B] \text{ Metryka} = \text{pasma} + \text{opóźnienie}$$

Ponadto jeżeli parametr $K5$ wynosi 0, wówczas część wzoru: $\frac{K5}{K4 + \text{niezawodność}}$ nie jest brana pod uwagę:

$$[C] \text{ Metryka} = \left[K1 * \text{pasma} + \frac{K2 * \text{pasma}}{256 - \text{obciążenie}} + K3 * \text{opóźnienie} \right] * 256$$

Przyjmując, że wagi zostaną ustalone na poziomach: $K1 = 1$, $K2 = 3$, $K3 = 1$, $K4 = 2$ oraz $K5 = 2$, należy opracować diagram parametryczny dla protokołu EIGRP (rysunek 6.5).



Rysunek 6.5. Diagram parametryczny wyboru tras dla pakietów przesyłanych w sieci Internet



Wskazówki dydaktyczne

Ze względu na ustawienie wag poszczególnych parametrów $K1$ do $K5$ na wyższe od zera zastosowano wzór A. Ponieważ z treści polecenia wynika, że nie trzeba wyliczać wartości metryki, utworzono standardowy diagram parametryczny. W przypadku gdy metryka miałaby zostać wyliczona, należałoby zastosować diagram parametryczny dla analizy wariantowej.

Ć W I C Z E N I E

6.6

Pozycjonowanie stron WWW w wyszukiwarce internetowej

O powodzeniu danego przedsięwzięcia biznesowego w dużej mierze decyduje reklama. Obecnie coraz częściej firmy wykorzystują reklamę w Internecie. Wiąże się ona z nakładami finansowymi, często wyliczonymi na podstawie liczby kliknięć danego banera reklamowego. Drugą, tańszą metodą promocji jest pozycjonowanie stron internetowych przedsiębiorstwa w najpopularniejszej wyszukiwarce internetowej, jaką jest Google. Z analiz prowadzonych przez firmę Google wynika, że użytkownicy Internetu, wyszukując dane zagadnienie, bardzo często przeglądają tylko pierwszą stronę wyników. Ważne jest zatem, aby znaleźć się w pierwszych 10 wynikach zapytania mieszczących się na pierwszej stronie wyświetlanej przez tę wyszukiwarkę.

W tym celu wykorzystywany jest mechanizm zwany pozycjonowaniem. Strona internetowa w zależności od struktury, zawartości oraz liczby łączy do niej z innych witryn może uzyskać mniejszą lub większą liczbę punktów. Im więcej punktów zostanie przyznanych stronie, tym większa szansa, że zostanie ona wyświetlona na jednym z pierwszych wyników wyszukiwania dla danego zapytania.

Jest wiele kryteriów punktowania strony, wśród których wyróżnić należy liczbę powtarzających się słów kluczowych zamieszczonych na stronie, poprawność znaczników META, nazwę strony itd. Jednak najważniejszym kryterium pozwalającym na podniesienie wartości PageRank jest liczba łączy w innych witrynach internetowych odwołujących się do ocenianej strony. Oznacza to, że jeżeli do strony odwołuje się 50 łączy z różnych witryn, to jest ona wyżej punktowana niż strona, do której odnośniki zostały zamieszczone jedynie na 20 witrynach. Na liczbę przyznanych punktów ma wpływ również opis łączy — hiperłącze wskazujące na stronę powinno zostać opisane najczęściej występującymi słowami kluczowymi.

Ogólny wzór na obliczenie PageRank jest następujący:

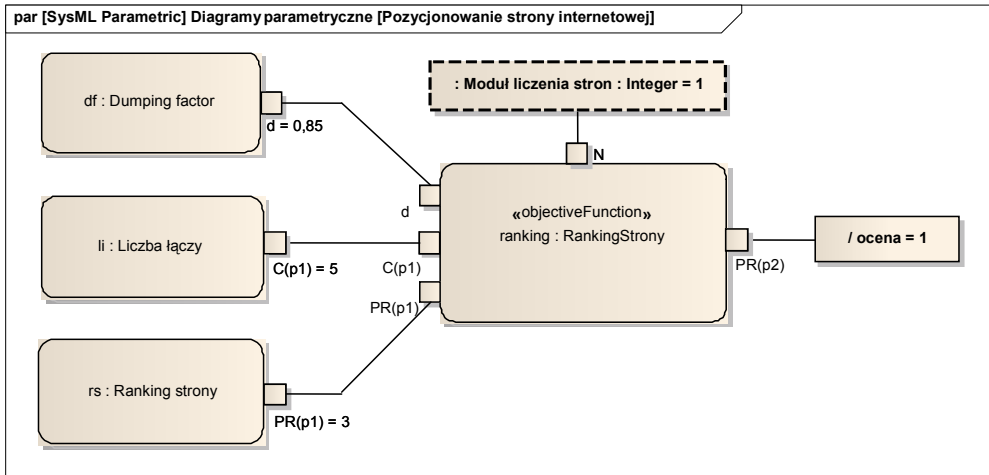
$$PR(p_i) = (1 - d) + d(PR(p_1)/C(p_1) + \dots + PR(p_n)/C(p_n)),$$

gdzie:

- ❑ d — tzw. *dumping factor*, ustawiony na stałe, zwykle przyjmuje wartość 0,85;
- ❑ $PR(p_n)$ — wartość PageRank witryny wskazującej na ocenianą stronę;
- ❑ $C(p_n)$ — ogólna liczba łączy zawartych na stronie internetowej.

Wartość PageRank jest zawsze wyrażana w wartościach całkowitych.

Należy opracować diagram parametryczny, który będzie umożliwiał wyliczenie wartości PageRank (rysunek 6.6).



Rysunek 6.6. Diagram parametryczny pozycjonowania strony internetowej



Wskazówki dydaktyczne

Moduł liczenia stron internetowych wskazujących na ocenianą stronę jest osobnym podsystemem, który pozwala na iteracyjne zbieranie ocen z poszczególnych witryn zawierających odsyłacze do ocenianej strony. Na diagramie zaprezentowano przykład działania algorytmu dla pojedynczej strony internetowej, której ocena przechowywana jest w zmiennej $PR(p2)$. Przekazywane wartości są obliczone zgodnie z zaprezentowanym algorytmem, wynikiem jest wartość 0,66, co w liczbach całkowitych zostało wyrażone jako 1 (ocena dla $PR(p2)$).

Ć W I C Z E N I E

6.7

Usługa webowa pobierająca dane makroekonomiczne

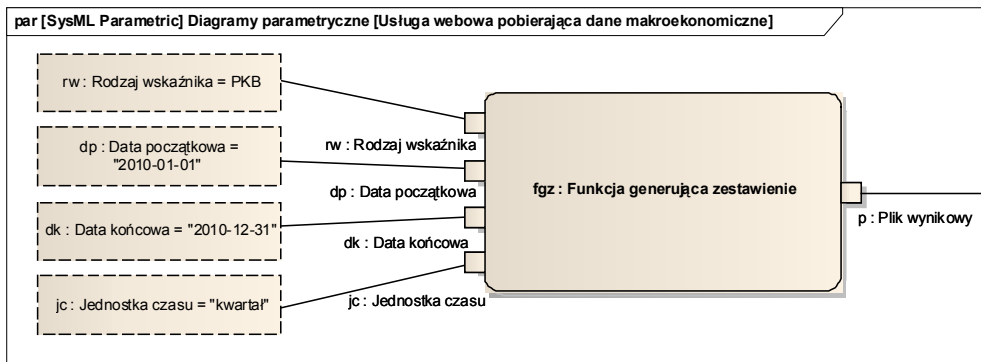
Programiści piszący programy komputerowe mogą obecnie coraz częściej korzystać z całego zestawu przygotowanych uprzednio usług webowych (ang. *web services*). Jedną z takich usług może być usługa webowa pobierająca dane z serwera na podstawie uprzednio zdefiniowanych parametrów przekazanych w postaci pliku XML.

Opisywana w niniejszym ćwiczeniu usługa webowa pobiera dane w postaci podstawowych wskaźników makroekonomicznych, takich jak inflacja, stopa bezrobocia czy wartość PKB. Wskaźniki te pobierane są dla wcześniej zdefiniowanego szeregu czasowego, np. kilku lat w przekroju miesięcznym. Osoba, która chce pobrać dane, powinna dostarczyć do usługi webowej w formie pliku XML następujące parametry:

- ❑ Rodzaj wskaźnika (wartość PKB, stopa bezrobocia itp.);
- ❑ Data początkowa (dzień, miesiąc, rok);
- ❑ Data końcowa (dzień, miesiąc, rok);
- ❑ Jednostka czasu (dzień, miesiąc, kwartał, rok);
- ❑ rodzaj Pliku wynikowego (PDF, XLS, DOC).

Rezultatem działania niniejszej usługi webowej jest wygenerowanie zestawienia w postaci pliku w formacie wybranym przez użytkownika.

Należy zaprojektować diagram parametryczny dla funkcji, która na podstawie parametrów wyjściowych wygeneruje obiekt, ten zaś następnie poprzez mechanizm serializacji zostanie przekształcony na plik w wybranym przez użytkownika formacie (rysunek 6.7).



Rysunek 6.7. Diagram parametryczny usługi webowej pobierającej dane makroekonomiczne



Wskazówki dydaktyczne

Elementem wyjściowym diagramu parametrycznego może być obiekt. Serializacja jest mechanizmem umożliwiającym zapisanie obiektu w postaci pliku. W związku z tym, gdy elementami wejściowymi lub wyjściowymi diagramu parametrycznego są pliki, należy je przedstawiać jako obiekty.



6.2. Zadania

Zadanie 6.1.

Jednym z kierunków rozwiązań technicznych z zakresu informatyki ekologicznej (ang. *green IT* lub *green computing*) jest odzyskiwanie ciepła emitowanego przez serwery w centrach danych. Dzięki zaawansowanym technologiom i instalacjom, takim jak chłodzenie free cooling, systemy odzysku ciepła i pompy ciepła, można odzyskać ponad 20% emitowanego ciepła. Jest ono wykorzystywane do ogrzewania pomieszczeń biurowych, domów mieszkalnych, a nawet sprzedawane zakładom energetyki ciepłej. Średniej wielkości centrum danych umożliwia odzyskiwanie energii z ciepła przekładające się na wiele gigawatogodzin rocznie. Rozwiązania te umożliwiają również obniżanie stawek za zużycie energii ciepłej. Odzysk ciepła z centrów danych i innych ośrodków informatycznych zmniejsza emisję dwutlenku węgla o kilka do kilkunastu ton rocznie.

The Green Grid zaproponował współczynnik efektywności energetycznej o akronimie PUE (*Power Usage Effectiveness*). Oblicza się go jako iloraz wolumenu energii dostarczanej do energii faktycznie wykorzystanej przez centrum danych. Wartość tego współczynnika uzależniana jest od klimatu, w którym centrum danych jest zlokalizowane. W Europie Środkowej i Północnej parametr ten kształtuje się na poziomie od 2,0 do 1,5, a na południu Europy wskaźnik ten przyjmuje wartość powyżej 2,0. Jeżeli wykorzystuje się chłodzenie wiatrem, jak np. w niektórych ośrodkach amerykańskich, PUE może kształtować się na poziomie 1,2 — a nawet niżej w przypadku zainstalowania większej liczby serwerów. Na podstawie powyższego opisu uzupełnionego o wiedzę o instalacjach odzysku ciepła opracuj diagram parametryczny.

Zadanie 6.2.

Jak wskazuje (Massel, 2001), w inteligentnych systemach zarządzania ruchem drogowym istotnym elementem jest zwiększanie przepustowości dróg, co można zrealizować poprzez zapewnianie minimalnego odstępu pomiędzy pojazdami. Bezpieczny odstęp przy założeniu określonej prędkości pojazdów oblicza się na podstawie wzoru:

$$l_b = \frac{v^2}{2} \left(\frac{1}{a_2} - \frac{1}{a_1} \right) + v \cdot t_r + l_1$$

gdzie:

- ❑ l_b — bezpieczny odstęp w metrach;
- ❑ v — prędkość pojazdów w km/h;
- ❑ a_1 — opóźnienie pierwszego pojazdu przy hamowaniu awaryjnym w m/s²;

- ❑ a_2 — opóźnienie drugiego pojazdu przy hamowaniu normalnym w m/s^2 ;
- ❑ vt_r — czas reakcji kierowcy drugiego pojazdu w sekundach;
- ❑ l_1 — długość pierwszego pojazdu w metrach.

Na podstawie niniejszego opisu opracuj diagram parametryczny obliczania bezpiecznego odstępu pojazdów.

Zadanie 6.3.

Pokonywanie różnicy poziomów w transporcie wodnym po żeglownych rzekach i kanałach możliwe jest dzięki śluzom stanowiącym przegrodzone komory wodne. Jak podają (Nitsch-Cisak i Jurczak, 1983) oraz (Szling i Winter, 1998), w celu obliczenia przepustowości śluzy stosuje się następujący wzór:

$$P = \frac{Wt_n}{T}$$

gdzie:

- ❑ P — przepustowość teoretyczna w ciągu 1 roku [t];
- ❑ W — największa ładowność zestawu pływającego [t];
- ❑ t_n — okres nawigacji (sezon) na danej drodze wodnej [s];
- ❑ T — czas śluzowania [s].

Na podstawie powyższego opisu przepustowości śluzowania opracuj diagram parametryczny obliczający przepustowość śluzy.

Zadanie 6.4.

Modernizacja linii kolejowych wymaga uwzględnienia określonych zasad i reguł dotyczących fizycznych podstaw ruchu pociągu i parametrów infrastruktury kolejowej. Do wyliczenia sumarycznej wartości oporu ruchu pociągu stosuje się następujący wzór (Massel, 2010):

$$W = W_0 + W_R + W_i$$

gdzie:

- ❑ W_0 — opór podstawowy;
- ❑ W_i — opór wynikający z ruchu na wzniesieniu;
- ❑ W_R — opór dodatkowy;
- ❑ W — opór ruchu [N].

Z kolei równanie ruchu pociągu uwzględniające opór ruchu przyjmuje następującą postać:

$$F(v) - W(v) = m(1 + \delta) \frac{dv}{dt}$$

gdzie:

- ❑ F — siła pociągowa [N];
- ❑ m — masa pociągu [kg];
- ❑ v — prędkość pociągu [m/s];
- ❑ δ — współczynnik mas wirujących (0,1 – 0,2);
- ❑ W — opór ruchu [N];
- ❑ t — czas [s].

Na podstawie powyższych reguł opracuj diagram parametryczny dla obliczenia siły pociągowej $F(v)$.

Zadanie 6.5.

Atrybutem każdej współczesnej firmy jest profesjonalny serwis internetowy. Projekt wdrożenia serwisu internetowego nie ogranicza się wyłącznie do jego stworzenia i opublikowania, lecz obejmuje swoim zakresem także nieustanne monitorowanie ruchu w ramach serwisu i usprawnianie samego serwisu. Pomiar ruchu ma szczególne znaczenie w przypadku firm zajmujących się szeroko pojętym biznesem elektronicznym i marketingiem internetowym.

Potencjalne mierniki ruchu w ramach witryny internetowej obejmują liczbę odwiedzin:

- ❑ ogółem;
- ❑ przez internautów, którzy celowo trafili na daną witrynę;
- ❑ przypadkowych.

Wspomniane miary wykorzystywane są między innymi przez narzędzie Google Analytics do formułowania tzw. współczynnika konwersji. Współczynnik ten umożliwia ocenę powodzenia serwisu zarówno w ujęciu bezwzględnym, jak i względnym.

$$L_z = K \cdot L_o$$

$$L_i = (1 - K) \cdot L_o$$

gdzie:

- ❑ L_o — liczba odwiedzin;
- ❑ K przyjmuje wartość od 0 do 1.

$$L_z = k \cdot L_o / 100$$

gdzie:

- ❑ k wyrażane jest procentowo i przyjmuje wartość od 0 do 100%.

Opracuj diagram parametryczny pozwalający na obliczanie współczynnika konwersji.

Zadanie 6.6.

Na podstawie ćwiczenia 6.5 opracuj diagram parametryczny dla analizy wariantowej, obliczając wartość metryki dla następujących wartości parametrów:

- ☐ K1 = 100;
- ☐ K2 = 100;
- ☐ K3 = 50;
- ☐ K4 = 100;
- ☐ K5 = 0;
- ☐ pasmo = 200 000;
- ☐ opóźnienie = 6100;
- ☐ niezawodność = 100;
- ☐ obciążenie = 50.

Zadanie 6.7.

Jedną z powszechnie występujących postaci systemów jest CRM analityczny. Jego charakterystyka została przedstawiona we wcześniejszych rozdziałach niniejszej książki. Istotą są dwa rodzaje analizy:

- ☐ analiza sprzedaży, na którą składa się obliczenie podstawowych wskaźników sprzedaży według różnych przekrojów i związana z tym prezentacja tych wskaźników w formie graficznej;
- ☐ analiza koszykowa, czyli określenie, które produkty lub grupy produktów sprzedają się najlepiej lub w określonej sekwencji.

Na podstawie przykładów dotyczących systemu CRM utwórz dwa diagramy parametryczne — odpowiednio dla analizy sprzedaży oraz analizy koszykowej.



Diagramy czynności

7.1. Ćwiczenia

Ć W I C Z E N I E

7.1

Serwis przechowywania, synchronizacji i udostępniania zasobów Dropbox

Dropbox (<http://www.dropbox.com>) jest serwisem przechowywania, synchronizacji i udostępniania zasobów w chmurze, najczęściej całych folderów oraz pojedynczych plików. Serwis Dropbox może współpracować z wieloma systemami operacyjnymi jednocześnie, umożliwiając dostęp do plików z poziomu przeglądarki. Pliki przesyłane są z wykorzystaniem standardu SSL, co wpływa na poprawę bezpieczeństwa danych. Po zainstalowaniu Dropboka na komputerze użytkownika tworzona jest automatycznie nowa skrzynka umożliwiająca rejestrowanie, przechowywanie oraz udostępnianie folderów i plików. Plik wprowadzony do tej skrzynki wysyłany jest na serwer w chmurze i tam przechowywany jako dogodna kopia zapasowa. Synchronizacja plików w Dropboksie powoduje, że pliki na poszczególnych komputerach są synchronizowane z plikami w chmurze. Wprowadzenie jakiegokolwiek pliku we własnej skrzynce Dropboka odwzorowywane jest w chmurze i odwrotnie. Do jednego konta w serwisie Dropbox użytkownik może podłączyć kilka swoich komputerów. Zwiększa to mobilność usługi, umożliwiając pracę zarówno z laptopem, jak również z komputerem stacjonarnym. Serwisy tego typu mogą spełniać rolę bardzo dogodnego narzędzia w pracy grupowej, kiedy np. międzynarodowy zespół opracowuje projekt. Wszystkie dokumenty merytoryczne, finansowe czy harmonogram są dostępne i modyfikowane na bieżąco w skrzynce projektu w serwisie Dropbox.

Kluczową funkcjonalnością z punktu widzenia serwisu jako całości jest udostępnianie plików w serwisie Dropbox. Przypadek użycia odzwierciedlający omawianą funkcjonalność może być zrealizowany w dwóch głównych wątkach:

- ❑ zamieszczenie pliku za pośrednictwem serwisu internetowego, co wymaga *wybrania zakładki plików, kliknięcia przycisku zamieszczania plików* oraz co najmniej jednokrotnego zrealizowania procedury *kliknięcia przycisku dodawania plików* wraz z wyspecyfikowaniem pliku/plików po uprzednim wyświetleniu *formatki zamieszczania plików na serwerze*;
- ❑ *upuszczenie pliku/plików w folderze Dropbox.*

Zamieszczanie plików za pośrednictwem serwisu internetowego może zostać *anulowane* w dowolnym momencie. Z uwagi na wykorzystanie funkcjonalności dostępnej w sieci w tym przypadku nie zachodzi konieczność dodatkowej weryfikacji dostępności połączenia internetowego. Odmienne sytuacja ma się, gdy plik jest udostępniany poprzez jego umieszczenie w folderze Dropbox. W takiej sytuacji może zachodzić konieczność oczekiwania na *wykrycie aktywnego połączenia internetowego*. Dopiero wtedy następuje *zmiana statusu ikony powiadomienia* na „synchronizowany” i *nawiązanie połączenia SSL*. Fizyczna transmisja z uwagi na minimalizację wykorzystania dostępnego pasma opiera się na przesyłaniu wyłącznie danych różnicowych. Po zweryfikowaniu *listy plików online* serwis *oblicza sumę różnicową, archiwizuje poprzednią wersję* każdego pliku, szyfruje i kompresuje wyłącznie nowe dane oraz przeprowadza *transmisję danych*. Po odebraniu danych na serwerze następuje potwierdzenie pomyślnego przeprowadzenia operacji. Może to być *zaktualizowanie listy plików w zakładce plików*, jeżeli przypadek użycia został zrealizowany za pośrednictwem formatki webowej, jak również *przywrócenie statusu ikony powiadomienia* („zsynchronizowany”) z jednoczesnym wyświetleniem *potwierdzenia przeprowadzenia synchronizacji w formie dymka*.

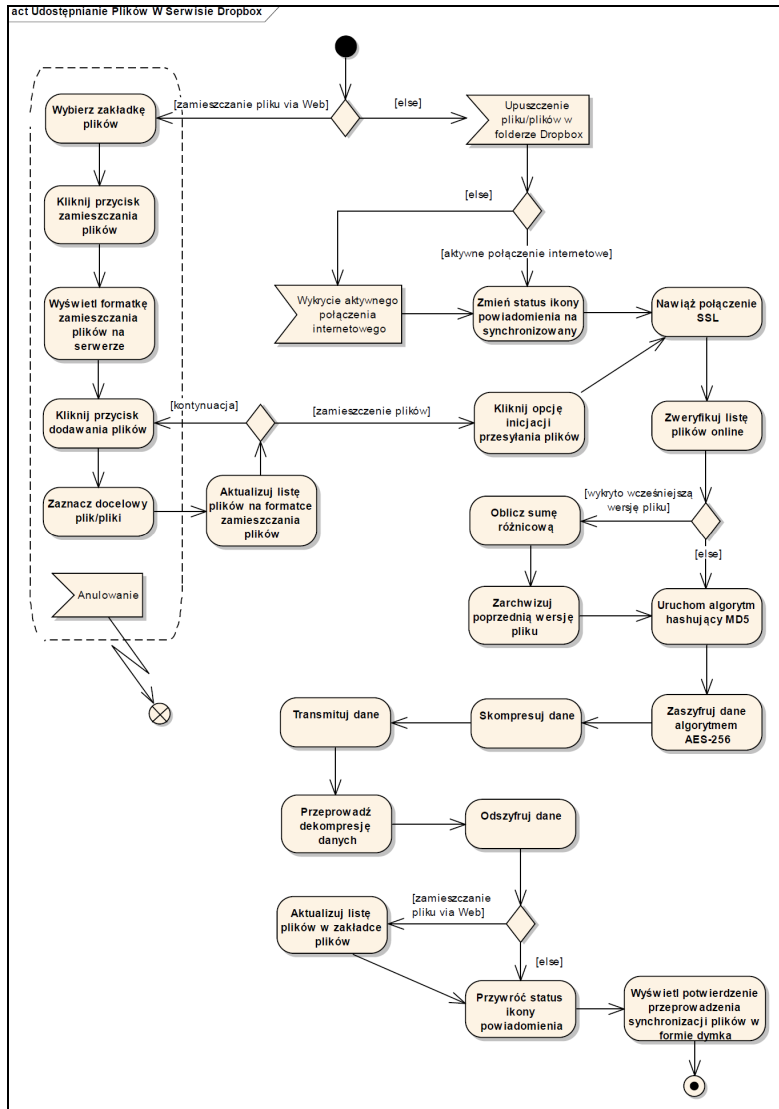
Na podstawie powyższego opisu i źródła opracuj diagram czynności udostępniania plików w serwisie Dropbox (rysunek 7.1).

Ć W I C Z E N I E

7.2 Elektroniczna obsługa reklamacji

Firma prowadząca sprzedaż smartfonów wyodrębniła w swojej strukturze organizacyjnej dział odpowiedzialny za obsługę posprzedażową. Ewentualne reklamacje podlegają kompetencjom *Działu reklamacji* oraz *Działu kontroli jakości*. Firma ściśle współpracuje z siecią serwisową producentów urządzeń mobilnych przejmując obsługę zgłoszenia reklamacyjnego, jeśli reklamacja jest zasadna.

Procedura obsługi reklamacji przedstawia się następująco. *Klient* przynosi do działu reklamacji wadliwie działający smartfon. *Dział reklamacji* weryfikuje formalnie zgłoszenie, po czym przyjmuje sprzęt i przekazuje go do *Działu kontroli jakości*,



Rysunek 7.1. Diagram czynności udostępniania plików w serwisie Dropbox

który merytorycznie ocenia, czy reklamacja jest uzasadniona. Fakt przyjęcia zgłoszenia odnotowywany jest w *Systemie monitorowania reklamacji*. Klient otrzymuje wydruk potwierdzenia przyjęcia reklamacji generowanego automatycznie przez system.

W przypadku uznania zasadności reklamacji pracownik firmy uzyskuje dostęp do pierwotnego zgłoszenia reklamacyjnego i wprowadza szczegółowe dane dotyczące smartfona, tj. jego numer seryjny IMEI, dane producenta, model, kody PIN oraz PUK.

Po wprowadzeniu następuje automatyczne sprawdzenie, czy smartfon nie znajduje się na liście ukradzionych urządzeń mobilnych. W przypadku pozytywnej weryfikacji generowane jest potwierdzenie przyjęcia urządzenia do naprawy. Akceptacja operacji prowadzi do elektronicznego powiadomienia wszystkich zainteresowanych stron o skierowaniu urządzenia do naprawy.

Kolejne czynności odbywają się w serwisie, gdzie smartfon jest przekazywany wraz ze specyfikacją reklamacji. W wyniku oceny dokonanej przez specjalistów określone zostają moduły niezbędne do naprawienia urządzenia. Może to być wyświetlacz, moduł zasilający, klawiatura, obudowa i inne. Serwisant zamawia brakujące moduły w zintegrowanym systemie informatycznym firmy (ZSI). Jeżeli moduły są dostępne w magazynie lokalnym, są kompletowane bez dodatkowej zwłoki. Z kolei złożenie zamówienia na brakujące części u producenta lub dostawcy nieodmiennie wiąże się z okresem oczekiwania na jego realizację. Wówczas następuje zmiana statusu reklamacji na „oczekującą na zamówione części”. Dopiero dostarczenie wymaganych części pozwala na podjęcie dalszych czynności związanych z naprawą urządzenia. Po skompletowaniu wszystkich elementów niezbędnych do naprawy serwis *Przeprowadza naprawę* urządzenia. Serwisanci dokonują rejestracji zużytych modułów, czasu naprawy oraz opisu przebiegu naprawy, tworząc tym samym protokół z naprawy. Po wysłaniu automatycznie generowanego e-maila o zakończeniu naprawy do klienta urządzenie jest do odbioru w *Dziale reklamacji* firmy. Właściciel urządzenia może przyjąć naprawione urządzenie bądź je odrzucić. Ten drugi przypadek traktowany jest jako ponowne zainicjowanie procedury reklamacji. W obu przypadkach następuje zmiana statusu zamówienia w *Systemie monitorowania reklamacji*.

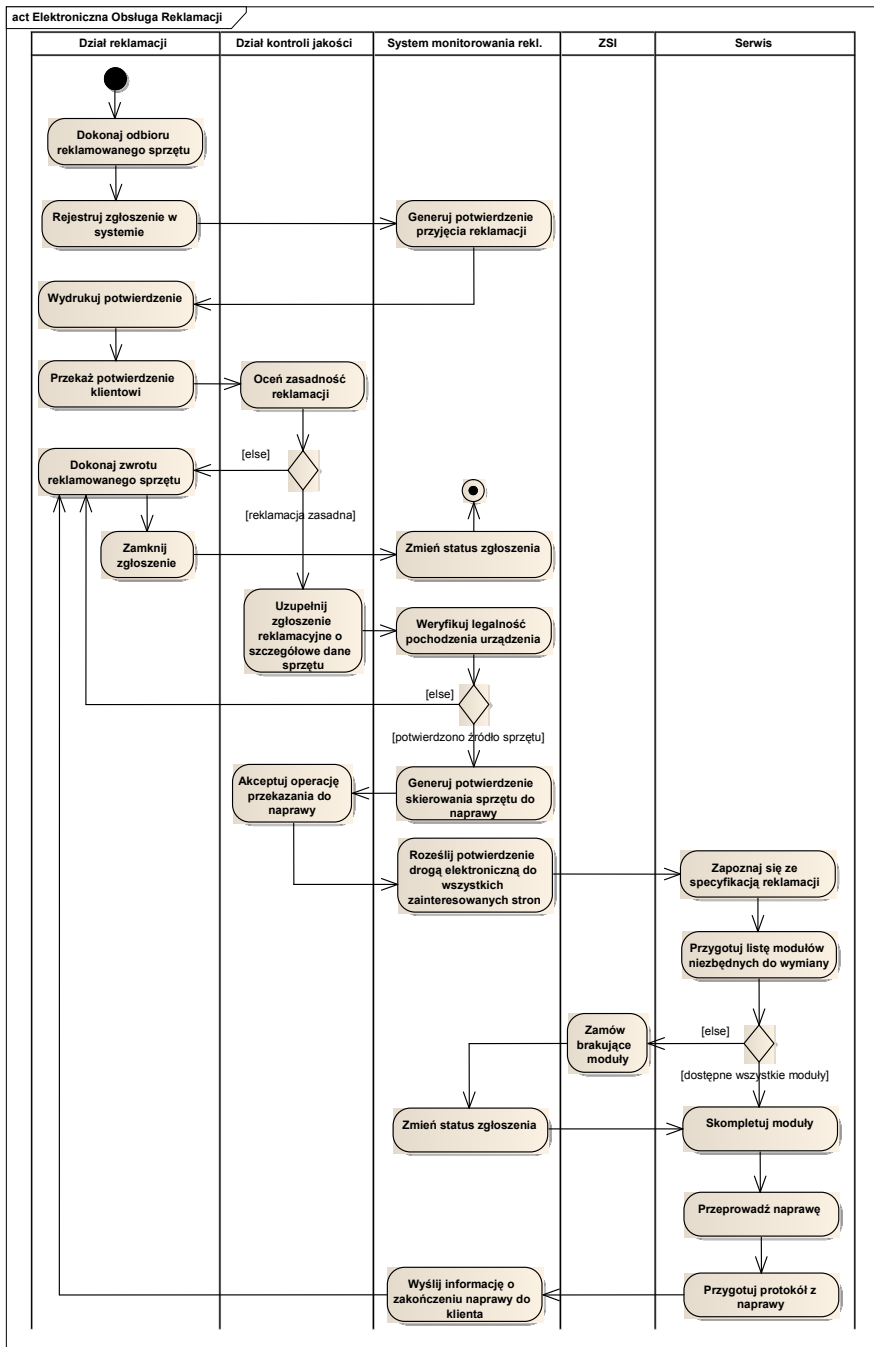
Bazując na powyższym opisie, należy rozwinąć poszczególne etapy elektronicznej obsługi reklamacji i zaprezentować je w postaci diagramu czynności (rysunek 7.2).



Wskazówki dydaktyczne

Pod względem procedury obsługi reklamacji niezwykle istotne jest wskazanie, które jednostki organizacyjne/podmioty obarczone są odpowiedzialnością za poszczególne czynności. O ile diagram sekwencji UML w sposób naturalny uwzględnia tego typu informacje, ujęcie ich na diagramach czynności wymaga wprowadzania dodatkowych formalizmów notacyjnych. Tym samym zdecydowano się na wprowadzenie do diagramu mechanizmu partycji, w którym każda wskazuje na fizyczne miejsce wykonywania czynności bądź system odpowiedzialny za jej realizację.





Rysunek 7.2. Diagram czynności elektronicznej obsługi reklamacji

ĆWICZENIE

7.3 Zapisywanie pliku XML z wykorzystaniem mechanizmu serializacji

Oprogramowanie systemu informatycznego wspierającego biuro nieruchomości musi umożliwiać wymianę danych między aplikacją stacjonarną a mobilną systemu. Wymianę tę realizuje się za pomocą plików XML. Aby utworzyć plik XML bądź pokrewny (HTML, SDMX/ML, SGML), używa się mechanizmu serializacji dostępnego przykładowo w takich współczesnych językach programowania jak Java, C#, C++, Python, PHP czy Ruby.

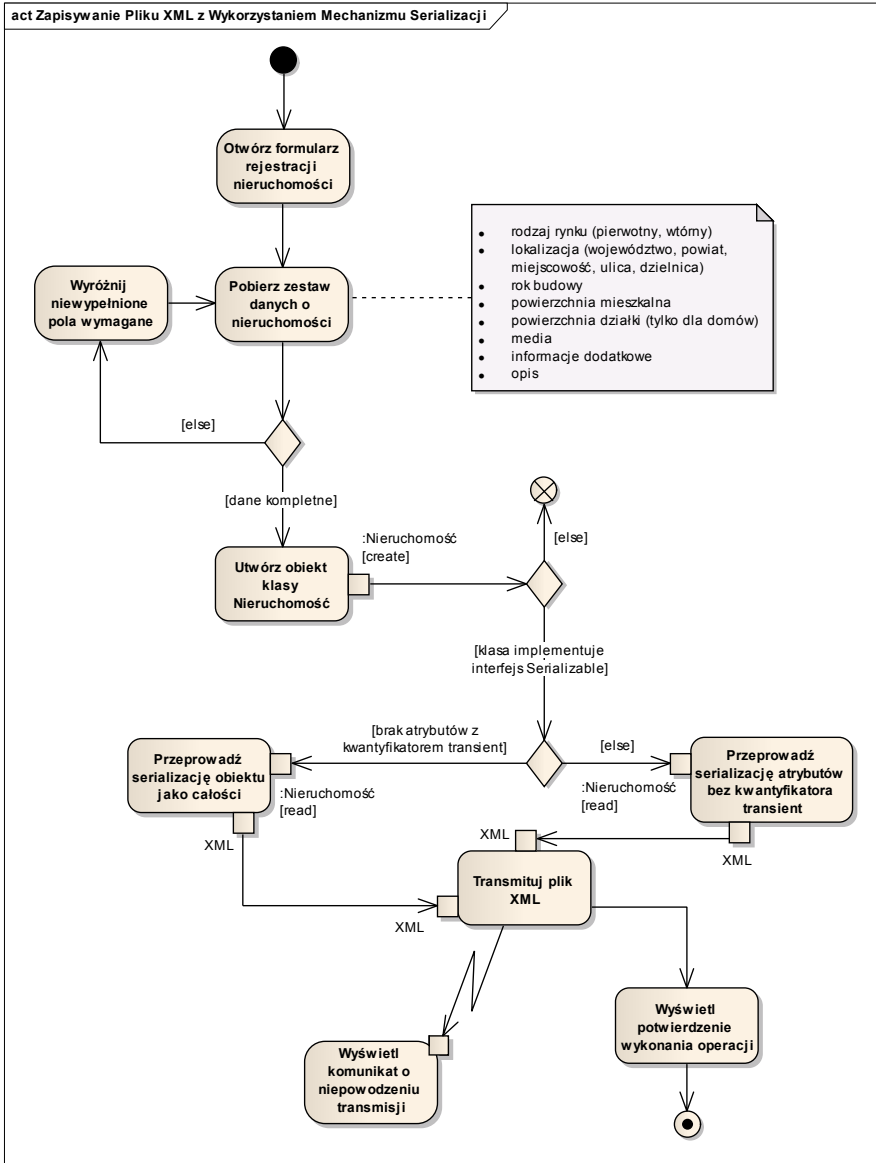
Mechanizm serializacji pozwala na przekształcanie obiektów na pliki, które w postaci strumienia danych mogą zostać zapisane na dysku, przesłane w sieci komputerowej lub przekazane do innego procesu systemu operacyjnego. Obiekt, który uległ serializacji, może zostać poddany procesowi odwrotnemu, czyli przykładowo przekształceniu z pliku na postać obiektu. Proces ten zwany jest deserializacją. W języku Java serializacji i deserializacji ulegają wszystkie atrybuty obiektu, które nie są oznaczone modyfikatorem *transient*. Cała hierarchia klas może ulec serializacji pod warunkiem implementacji interfejsu *Serializable*.

Pracownik firmy korzystający z aplikacji mobilnej przekazuje dane o nowo oferowanej nieruchomości poprzez *otwarcie formularza rejestrowania nieruchomości* i wypełnienie go. Wśród tych danych znajdują się informacje takie jak:

- ☐ *rodzaj rynku* (pierwotny, wtórny);
- ☐ *lokalizacja* (województwo, powiat, miejscowość, ulica, dzielnica);
- ☐ *rok budowy*;
- ☐ *powierzchnia mieszkalna*;
- ☐ *powierzchnia działki* (tylko dla domów);
- ☐ *media*;
- ☐ *informacje dodatkowe*;
- ☐ *opis*.

Podczas *pobierania zestawu danych o nieruchomości* może się okazać, że dane te są niekompletne. Wówczas następuje *wyróżnienie niewypełnionych pól wymaganych*. Wtedy dopiero istnieje możliwość *utworzenia obiektu klasy Nieruchomość*. Jeżeli klasa *implementuje interfejs Serializable* i żadne z pól *nie jest oznaczone modyfikatorem transient*, wówczas następuje *serializacja obiektu jako całości*. W innym przypadku *przeprowadzana jest serializacja tylko atrybutów nieposiadających kwantyfikatora transient*. Po przeprowadzeniu serializacji następuje *transmisja obiektu w pliku XML*. W przypadku poprawnego zakończenia transmisji następuje *wyświetlenie potwierdzenia wykonanej operacji*. W innym przypadku *wyświetlany jest komunikat o niepowodzeniu transmisji*.

Na podstawie powyższych danych należy utworzyć diagram czynności, w ramach którego obiekt z klasy *Nieruchomość* poddany zostanie procesowi serializacji w języku Java (rysunek 7.3). Wynikiem ma być plik XML, który w postaci znaczników opíše nowo dodaną nieruchomość. Plik ten w postaci strumienia danych zostanie przesłany na aplikację stacjonarną biura nieruchomości.



Rysunek 7.3. Diagram czynności zapisywania pliku XML z wykorzystaniem mechanizmu serializacji

**Wskazówki dydaktyczne**

W powyższym ćwiczeniu przyjęto założenie, że klasa typu `Serializable` musi posiadać przynajmniej jeden atrybut, który nie jest typu `transient`. Zakłada się, że klasa, która nie implementuje interfejsu `Serializable`, nie zostanie poddana procesowi transmisji.

Ć W I C Z E N I E**7.4****Planowanie spotkania z wykorzystaniem systemu CRM**

Do palety funkcjonalności systemów CRM zaliczyć należy monitorowanie wolumenów zakupów poszczególnych klientów powiązane ze wspomaganiem organizowania spotkań z klientami w przypadku znaczącego spadku liczby kupowanych przez nich towarów czy usług. Przypadek użycia planowania spotkania z wykorzystaniem systemu CRM opiera się na parametrach granicznych średniomiesięcznych zakupów klienta na koniec danego miesiąca. Parametryzacja jest tematem osobnego przypadku użycia. Sygnalizacja potrzeby podjęcia akcji, w szczególności przeprowadzenia spotkania z klientem, zwyczajowo następuje, jeżeli średniomiesięczne zakupy klienta spadają o 50%.

System udostępnia funkcjonalność umożliwiającą przeglądanie syntetycznej listy klientów wraz z podstawowymi statystykami ilościowymi. Stosownie do skali spadku wyświetlane są odpowiednie znaczniki alarmowe. Przy założeniu pozostawienia granicznej wartości sygnalizacji potrzeby podjęcia akcji (50%) przyjmuje się następującą procedurę powiadamiania:

- ❑ jeśli średniomiesięczny wolumen zakupów spadnie poniżej wartości granicznej, lecz będzie utrzymywał się na poziomie co najmniej 20% wartości odniesienia, system zainicjuje pomarańczowy znacznik alarmowy;
- ❑ jeśli z kolei średniomiesięczny wolumen zakupów spadnie poniżej 20% wartości odniesienia, wyświetlany będzie czerwony znacznik alarmowy.

W odniesieniu do wybranej przez operatora systemu pozycji listy klientów można podjąć szereg akcji. Obejmują one:

- ❑ przesłanie zaproszenia do wykonania zakupów w postaci standardowego listu zapraszającego definiowanego w odrębnym przypadku użycia;
- ❑ złożenie oferty promocyjnej dostosowanej do indywidualnych potrzeb danego klienta;
- ❑ umówienie spotkania, w tym spotkania wielostronnego;
- ❑ odroczenie podjęcia decyzji o 15 dni, w konsekwencji którego system winien śledzić zmiany i powiadomić o nich po upływie wyspecyfikowanego terminu.

Operator systemu ma naturalnie możliwość zaniechania wyboru akcji. W przypadku podjęcia decyzji o spotkaniu z klientem należy wybrać proponowany termin z wykorzystaniem apletu kalendarza oraz wprowadzić lokalizację do stosownego pola tekstowego. Poszczególnych uczestników spotkania dodaje się do formularza

za pomocą uproszczonej wyszukiwarki. Operator może ograniczyć się do dodania wyłącznie klienta w obszarze zainteresowania, jak również uwzględnić jedną lub więcej osób towarzyszących. Potwierdzenie ustalenia terminu i listy uczestników spotkania wyświetlane jest na ekranie. Niezależnie od rodzaju podjętej akcji wszystkie zainteresowane strony otrzymują powiadomienie i ewentualne dokumenty drogą elektroniczną.

Na bazie omówionego zestawu procedur powiadamiania i akcji należy zaproponować kompletną listę czynności i zsintetyzować je do postaci diagramu czynności przedstawiającego proces planowania spotkania z wykorzystaniem systemu CRM (rysunek 7.4).

Ć W I C Z E N I E

7.5 Listy kontroli dostępu na urządzeniach sieciowych

Współczesne urządzenia pośredniczące w sieciach komputerowych, do których zalicza się przede wszystkim routery i przełączniki, powinny pozwalać nie tylko na sprawny przesył danych, lecz również umożliwiać filtrowanie ruchu. Zwiększa to zarówno bezpieczeństwo przesyłanych danych, jak również ogranicza wielkość ruchu w globalnej sieci komputerowej, jaką jest Internet. Realizacja tego celu jest możliwa dzięki zastosowaniu list kontrolnych dostępu (ang. *Access Control List* — ACL), które funkcjonują na zasadzie podobnej do działania ścian ogniowych (ang. *firewall*). W tym ćwiczeniu zostały przedstawione zasady działania list kontroli dostępu na routerach firmy Cisco Systems.

Dzięki listom kontroli dostępu pakiet może zostać odrzucony lub przesłany przez dany interfejs (port routera). Istnieją dwa sposoby filtrowania ruchu:

- ☐ za pomocą list standardowych, których numery znajdują się w przedziale od 1 do 99 oraz 1300 do 1999;
- ☐ przy wykorzystaniu list rozszerzonych, gdzie numery znajdują się w zakresie od 100 do 199 oraz 2000 do 2699.

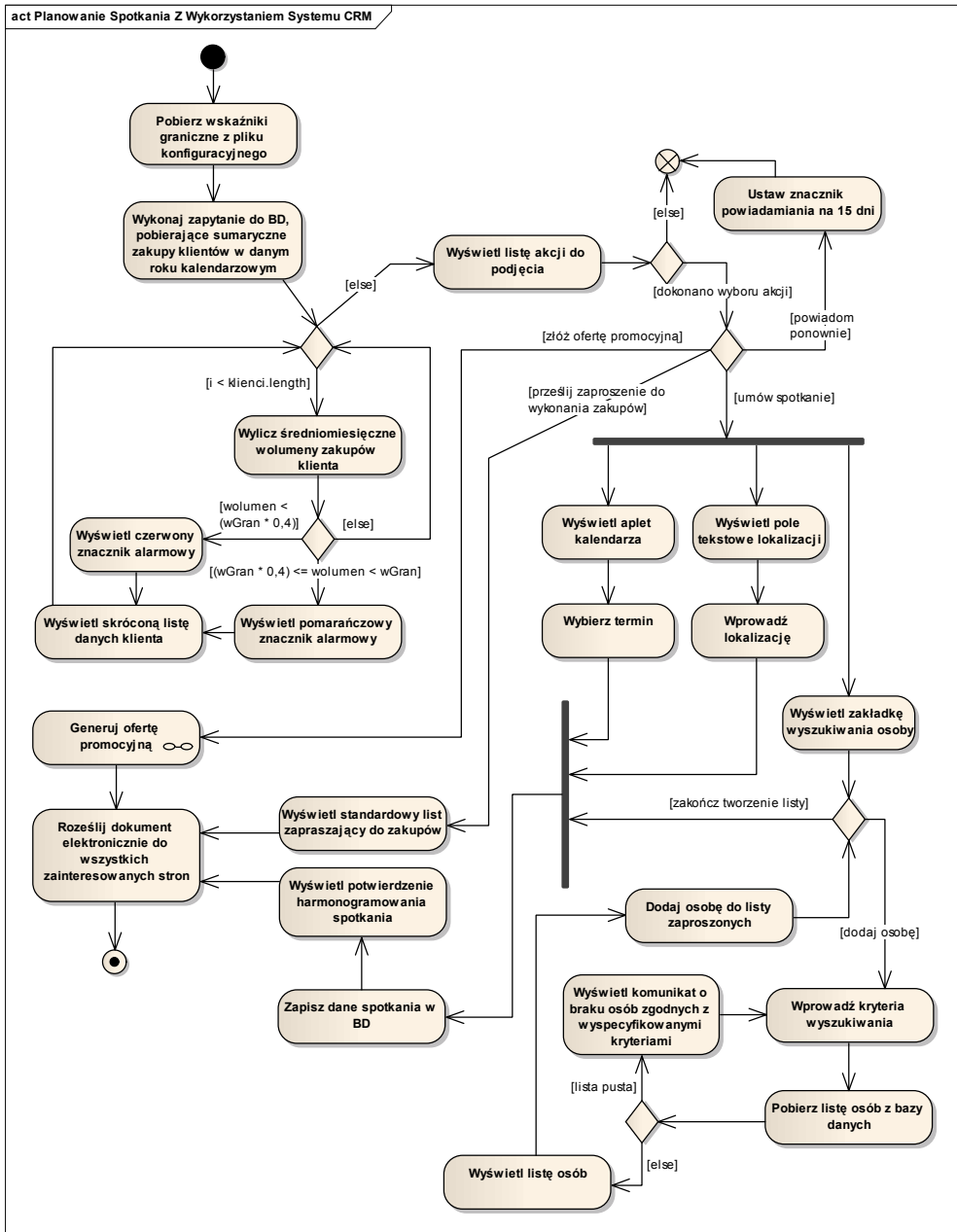
Listy standardowe filtrują ruch jedynie na podstawie adresu źródłowego pakietu. Zatem jeżeli odrzucone mają być pakiety z komputera 192.168.1.10, wystarczy utworzyć następujący wpis:

```
access-list 1 deny host 192.168.1.10
```

i przypisać go do odpowiedniego interfejsu.

Z kolei listy rozszerzone filtrują ruch między innymi na podstawie następujących parametrów:

- ☐ typu protokołu, np. TCP, UDP, ICMP;
- ☐ adresu źródłowego pakietu;
- ☐ adresu docelowego pakietu;
- ☐ numeru portu, np. 80 dla usługi WWW, 22 dla serwera SSH.



Rysunek 7.4. Diagram czynności prezentujący planowanie spotkania z wykorzystaniem systemu CRM

Aby filtrować ruch za pomocą list rozszerzonych, wystarczy podać typ protokołu oraz adresy źródłowy i docelowy. Rodzaj decyzji jest uzależniony od występowania słowa kluczowego `deny` (zablokowanie pakietu) lub `permit` (zezwolenie na przesłanie pakietu). Instrukcja definiująca listę rozszerzoną może być następująca:

```
access-list 100 deny tcp host 192.168.1.10 host 10.0.0.1 eq 80
```

co oznacza, że będą blokowane pakiety z komputera 192.168.1.10 (źródło) do serwera 10.0.0.1 (cel) przesyłane poprzez protokół TCP na port 80 (usługa WWW). W powyższej liście numer portu jest opcjonalnym parametrem. Adresy IP zarówno źródłowe, jak i docelowe mogą wskazywać na całe sieci (grupy komputerów), a nie tylko pojedyncze komputery.

W liście kontrolnej dostępu może występować wiele takich wpisów (instrukcji), które przetwarzane są w sposób sekwencyjny. Jeżeli pierwsza instrukcja nie może zostać dopasowana do pakietu (przez niezgodność adresu źródłowego lub docelowego IP, typu protokołu czy też numeru portu), sprawdzana jest kolejna instrukcja wyspecyfikowana w liście. Jeżeli instrukcja występująca w liście ACL zostanie dopasowana, następuje wykonanie akcji w niej zawartej (przesłanie lub zablokowanie pakietu) i przerwanie sprawdzania kolejnych instrukcji na liście.

Na podstawie powyższego opisu należy opracować diagram czynności wizualizujący decyzyjny charakter działania list kontroli dostępu (rysunek 7.5).



Wskazówki dydaktyczne

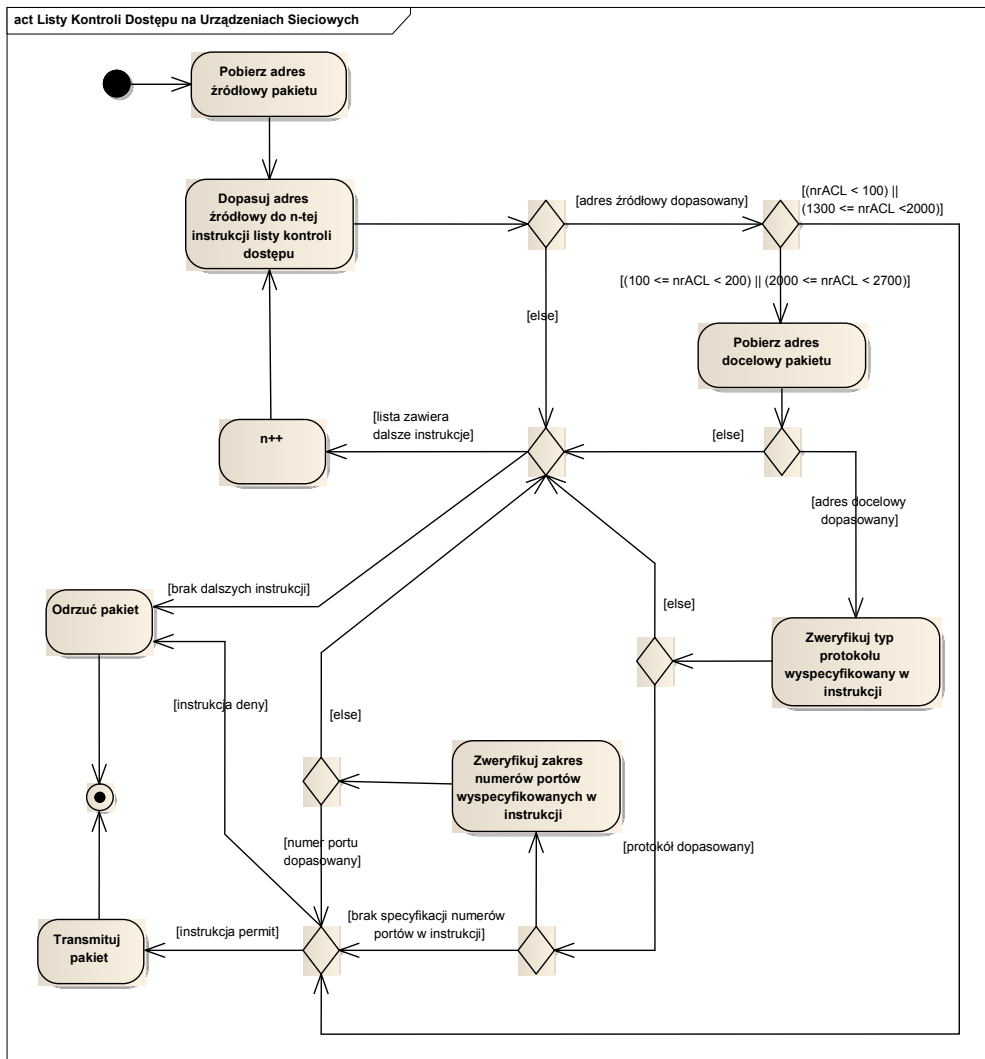
W niniejszym ćwiczeniu ograniczono się do podstawowych zasad działania list kontroli dostępu. W rzeczywistości listy te oferują dużo bardziej złożone reguły dotyczące filtrowania danych.

Ć W I C Z E N I E

7.6 Rozliczanie usługi medycznej

Zakłady Opieki Zdrowotnej finansują swoją działalność na podstawie kontraktów zawartych z funduszem centralnym odpowiedzialnym za finansowanie służby zdrowia. Rozliczenie wykonanych świadczeń następuje drogą elektroniczną poprzez przekazywanie stosownych danych w plikach zapisanych w formacie XML.

Pacjent nieposiadający podstaw do finansowania ze środków funduszu centralnego jest zobowiązany do samodzielnego wniesienia opłaty za wykonane świadczenie medyczne. W przypadku gdy są podstawy do sfinansowania leczenia przez fundusz centralny, zakład przedkłada dokumentację medyczną zawierającą dokumenty poświadczające wykonane usługi zdrowotne, koszty zaaplikowanych leków i inne. Wówczas następuje czynność *Weryfikuj dokument będący podstawą uzyskania świadczenia*. Jeżeli weryfikacja potwierdza możliwość uzyskania świadczenia z funduszu, to przypadek ten kierowany jest do dalszego etapu procesu rozliczenia.

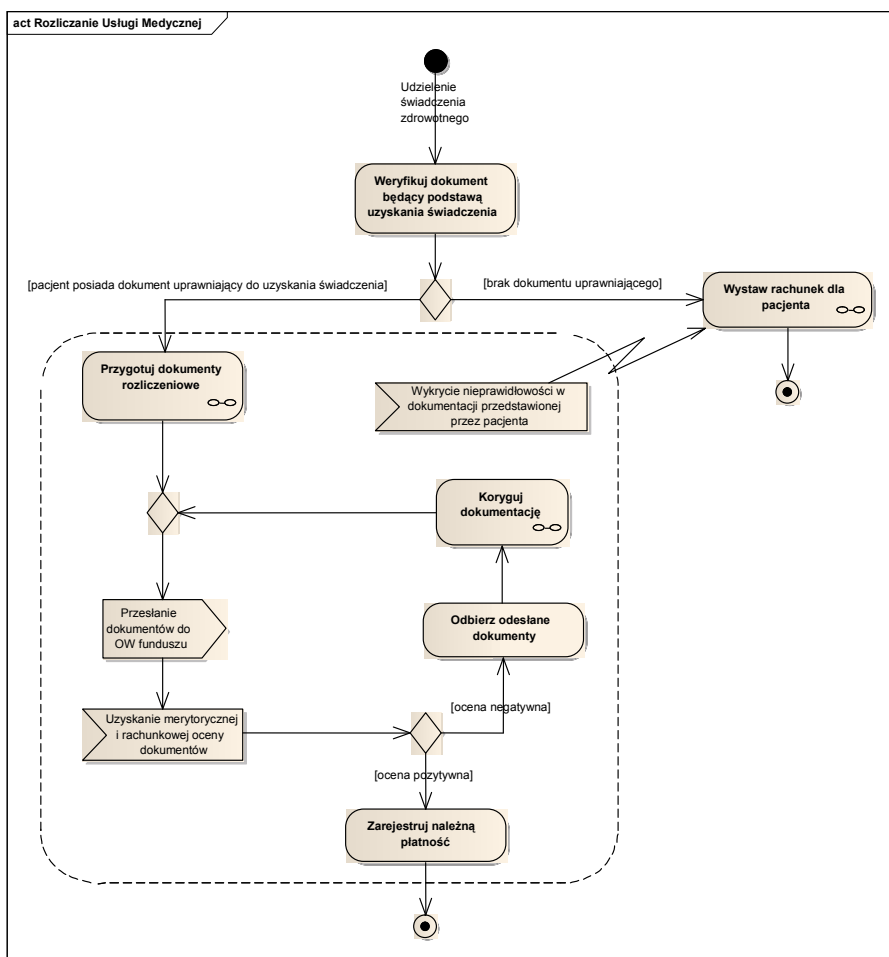


Rysunek 7.5. Diagram czynności list kontroli dostępu na urządzeniach sieciowych

W tym celu podejmowana jest czynność *Przygotuj dokumenty rozliczeniowe*. W dalszym etapie następuje *Przesłanie dokumentów do oddziału wojewódzkiego funduszu*. W oddziale tym możliwe jest *Uzyskanie merytorycznej i rachunkowej oceny dokumentów*. W przypadku oceny pozytywnej wykonywana jest czynność *Zarejestruj należną płatność*. Jeżeli zostanie wystawiona ocena negatywna, to dokumentacja jest odsyłana i po stronie zakładu realizowana jest czynność *Odbierz odesłane dokumenty*. W zakładzie opieki zdrowotnej następuje jej korygowanie lub uzupełnianie poprzez czynność *Koryguj dokumentację*. W przypadku wnoszenia kolejnych po-

prawek i uzupełnień cykl dotyczący przesłania i uzyskania oceny merytorycznej jest powtarzany iteracyjnie. Uzyskanie ostatecznej oceny pozytywnej przesłanej dokumentacji skutkuje czynnością *Zarejestruj należną płatność*. Po wykonaniu tych czynności następuje ponowne przesłanie dokumentacji do oddziału wojewódzkiego funduszu, gdzie po raz kolejny następuje jej ocena. Jeżeli spełnienie standardów formalnych odnośnie do dokumentacji medycznej nie jest możliwe, podejmuje się czynność *Wystaw rachunek dla pacjenta*.

Opisany powyżej proces rozliczania usługi medycznej świadczonej przez Zakład Opieki Zdrowotnej, a finansowanej przez fundusz centralny należy zaprezentować w postaci diagramu czynności (rysunek 7.6).



Rysunek 7.6. Diagram czynności obrazujący rozliczanie usługi medycznej

Ć W I C Z E N I E

7.7 Migracja na bezpłatne oprogramowanie

Wraz z dynamicznym rozwojem oprogramowania typu open source oraz pokrewnego coraz więcej firm rozważa możliwość zrezygnowania z płatnych kopii na rzecz ich bezpłatnych odpowiedników. Wśród najpopularniejszych programów znajdują się programy biurowe oraz systemy serwerowe.

Pierwszą czynnością wykonywaną podczas migracji na wolne oprogramowanie jest *określenie zakresu wymaganej funkcjonalności* niezbędnej do wykonywania zaplanowanych zadań, a następnie *skompletowanie listy wolnych odpowiedników stosowanego oprogramowania własnościowego*. Dalsza uproszczona procedura migracji na wolne oprogramowanie typowego podmiotu biznesowego przewiduje następujące etapy, które mogą być wykonywane jednocześnie:

- ❑ *sprawdzenie, czy do oprogramowania można zamówić wsparcie techniczne, a następnie obliczenie kosztów wsparcia technicznego i sprawdzenie, czy koszt wsparcia technicznego nie przekracza kosztów związanych z użytkowaniem aktualnych rozwiązań;*
- ❑ *sprawdzenie zgodności z popularnymi formatami plików;*
- ❑ *sprawdzenie częstości pojawiania się aktualizacji bezpieczeństwa;*
- ❑ *sprawdzenie, czy dalszy rozwój produktu został zagwarantowany;*
- ❑ *określenie budżetu, jaki ma zostać ustalony na szkolenia użytkowników, i w konsekwencji zweryfikowanie, czy koszt szkolenia użytkowników nie przewyższa kosztów użytkowania obecnie wykorzystywanych produktów przez najbliższe 5 lat;*
- ❑ *porównanie bezpłatnego odpowiednika z płatnym produktem pod względem wymogów sprzętowych w zakresie niezbędnych zasobów systemu komputerowego.*

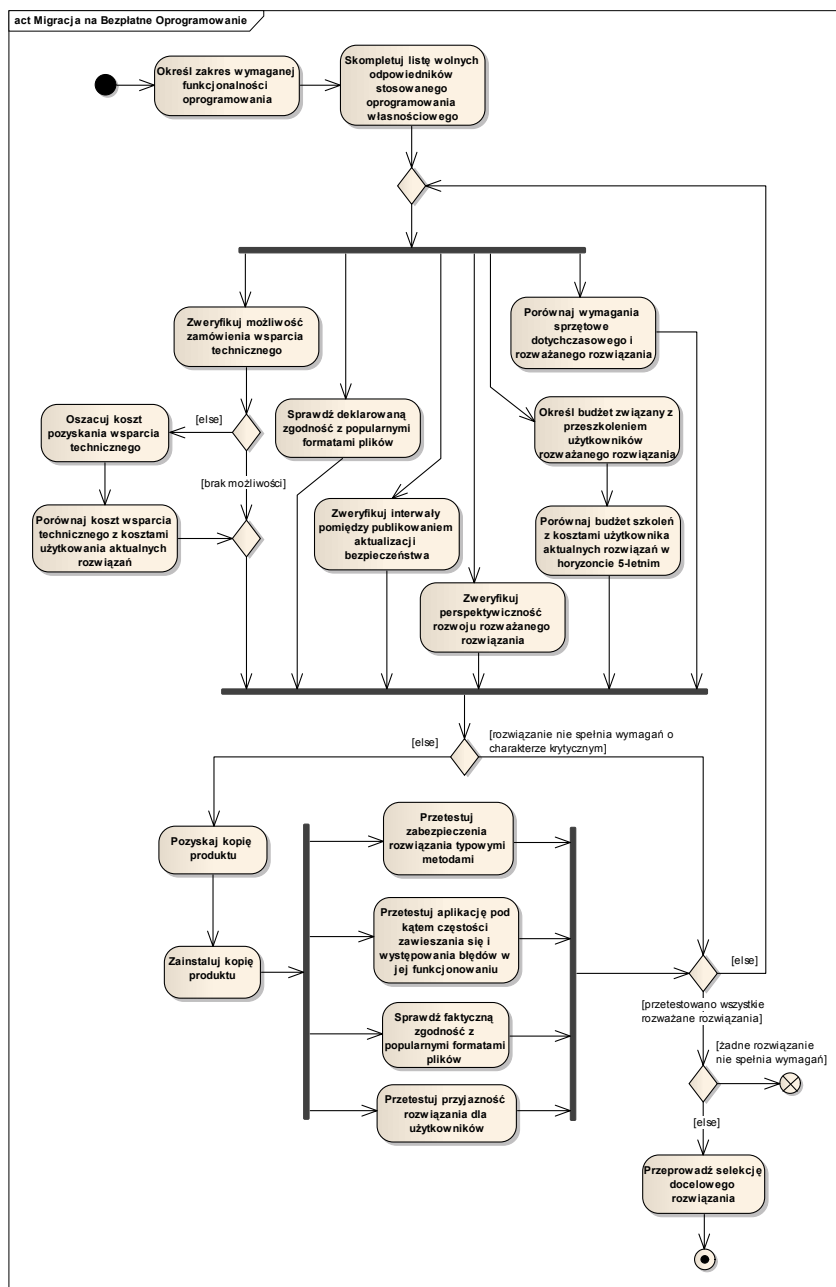
Jeżeli powyższe kryteria porównawcze przemawiają na niekorzyść wdrożenia wolnego oprogramowania i jeżeli przetestowano wszystkie możliwe rozwiązania, podejmowana jest decyzja o zaniechaniu prac nad migracją na wolne oprogramowanie.

W przypadku gdy porównywany darmowy produkt spełnia przyjęte powyżej kryteria, następuje *pozyskanie kopii produktu*, a następnie jej *zainstalowanie*. Testowanie produktu odbywa się w czterech równoległych etapach:

- ❑ *przetestowanie zabezpieczeń systemu typowymi metodami;*
- ❑ *testowanie aplikacji pod kątem częstości zawieszania się i występowania błędów w jej funkcjonowaniu;*
- ❑ *sprawdzenie faktycznej zgodności z popularnymi rodzajami plików;*
- ❑ *testowanie przyjazności dla użytkowników.*

Jeżeli po testach okaże się, że wiele z testowanych programów spełnia postawione wymagania, wówczas następuje *przeprowadzenie selekcji docelowego rozwiązania*.

Należy opracować diagram czynności prezentujący procedurę migracji na bezpłatne oprogramowanie (rysunek 7.7).



Rysunek 7.7. Diagram czynności opisujący zasady migracji na otwarte oprogramowanie

7.2. Zadania

Zadanie 7.1.

Współczesne systemy informatyczne są tworzone i wdrażane z wykorzystaniem różnych założeń metodologicznych. Najpopularniejsze aktualnie są metodyki strukturalne, społeczne, obiektowe i adaptacyjne (Wrycza, 2010). Zauważalna jest tendencja do skracania czasu trwania cyklu życia systemu. Temu właśnie kryterium wychodzą naprzeciw metodyki adaptacyjne zwane również „zwinnymi”, „żwawymi”, „lekkimi” bądź „agilnymi” (ang. *agile*). Jedną z bardziej popularnych i często używanych w tej grupie metodyk jest podejście Scrum, nazwane tak ze względu na swoją istotę — pojęcie młyna w grze w rugby.

Rozwój tej metody jest stymulowany przez Scrum Alliance oraz *Scrum.org*. Metodyka ta pozwala na transformację wymagań w docelowy system czy docelowe oprogramowanie. Jak wyszczególniono w opracowaniu (Brandt, 2010), w metodzie Scrum wypełniane są trzy role:

- ❑ Mistrz Młyna — główny projektant (ang. *Scrum Master*);
- ❑ właściciel produktu (ang. *Product Owner*);
- ❑ zespół projektowy (ang. *The Team*).

Tworzenie systemu realizowane jest w iteracjach określanych mianem sprintów, które trwają od dwóch do sześciu tygodni, a wielkość zespołu wynosi od 5 do 9 osób. W fazie planowania sprintu definiowane są wymagania systemowe (por. rozdział 2) w postaci diagramów wymagań (Wrycza i Marcinkowski, 2010). Zastosować można również diagramy przypadków użycia języka UML. Wymagania znajdują się na liście wymagań do wykonania (ang. *backlog*), które są uporządkowane według swego znaczenia od najważniejszych do najmniej ważnych. Zespół pobiera wymagania z listy wymagań do wykonania i kolejno je realizuje w następujących po sobie sprintach. W każdym sprincie zespół odbywa codzienne spotkania (ang. *daily scrum*), porządkując prace i rezultaty zespołu. Zidentyfikowane problemy rejestruje się na liście trudności (ang. *impediments*), które powinny zostać jak najszybciej rozwiązane przez zespół. Kolejne sprinty wzbogacają funkcjonalność tworzonego systemu zgodnie z przyjętymi wymaganiami oraz kryteriami ukończenia. Brak akceptacji sprintu powoduje powrót wymagania na listę wymagań do wykonania i powtórzenie iteracji. Odrzucenie lub akceptacja sprintu następuje na końcowym spotkaniu przeglądowym (ang. *Sprint Review*). Właściciel produktu, główny projektant i zespół projektowy spotykają się jeszcze raz (ang. *Sprint Retrospective*) dla dokonania analizy retrospekcji sprintu w celu usprawnienia efektywności zespołu w kolejnych sprintach. Sprinty powtarzane są iteracyjnie dla kolejnych wymagań. Sprinty powtarzają się aż do osiągnięcia wystarczającej funkcjonalności systemu (Brandt, 2010).

Na podstawie powyższego opisu oraz publikacji źródłowej (Brandt, 2010) opracuj diagram czynności w metodyce zwinnej Scrum.

Zadanie 7.2.

Współczesne firmy funkcjonujące na rynku globalnym są narażone na ciągłe i szybko postępujące zmiany rynkowe. Z jednej strony pojawiają się zagrożenia dla firm, z drugiej zaś nowe szanse rynkowe. Firmy te są w wysokim stopniu uzależnione od systemu informatycznego wspomagającego ich działalność biznesową. Zmiany, zagrożenia i nowe szanse powodują, że opierając się na wdrożonym w firmie statycznym systemie informatycznym, organizacja biznesowa nie nadąża za zmianami, co może skutkować obniżeniem jej zysków bądź wręcz wypadnięciem z rynku. Wprowadzenie zmian w użytkowanych systemach jest powolne i może być opóźnione ze względu na przyjęte czasochłonne procedury modyfikacji i adaptacji eksploatowanego systemu, a także w związku z obawami kierownictwa IT firmy dotyczącymi zagrożenia obniżenia funkcjonalności systemu użytkowanego i niepewnością co do sprawności i niezawodności nowo tworzonego systemu.

Jak wskazuje (Koława, 2010), coraz częściej remedium na cechy statycznego systemu informatycznego firmy są systemy bądź oprogramowanie jednorazowe, które można porównać do wielu dostępnych w supermarketach artykułów konsumpcyjnych jednorazowego użytku. Systemy takie można tworzyć szybko poprzez zastosowanie metody ADP (*Automated Defect Prevention*). Nie jest ona ukierunkowana na identyfikację i usuwanie pojedynczych błędów oraz defektów, lecz na zapobieganie im oraz eliminowanie ich poprzez wczesne testowanie oprogramowania z wykorzystaniem narzędzi wspomagających wytwarzanie oprogramowania. Uwaga programistów oprogramowania jednorazowego jest więc skierowana na sam proces jego wytwarzania, a nie na przestrzeganie zasad poszczególnych metodyk tworzenia systemów. W trakcie procesu wytwarzania oprogramowania wykonywane są następujące czynności:

- ☐ wczesne testowanie oprogramowania poprzez analizę statyczną i automatyczne testy jednostkowe;
- ☐ testowanie w postaci automatycznych testów funkcjonalnych;
- ☐ zarządzanie procesem wytwarzania oprogramowania.

Tak utworzony jednorazowy system informatyczny, w którym przestrzega się zasad wysokiej jakości procesu wytwarzania oprogramowania, może niemal na bieżąco wspierać niezbędne zmiany dzięki reakcji na aktualne szanse i zagrożenia, co wzmacnia bądź utrzymuje przewagę konkurencyjną firmy na rynku. Takie jednorazowe systemy firma może bez ryzyka wdrażać, rezygnując z dotychczasowych statycznych systemów, zastępując je bardziej efektywnymi, dynamicznymi wersjami lub wręcz nowymi systemami.

Na podstawie opisu metody ADP (albo jej odpowiedników oferowanych przez inne firmy lub freeware) przedstaw proces tworzenia oprogramowania jednorazowego w postaci diagramu czynności.

Zadanie 7.3.

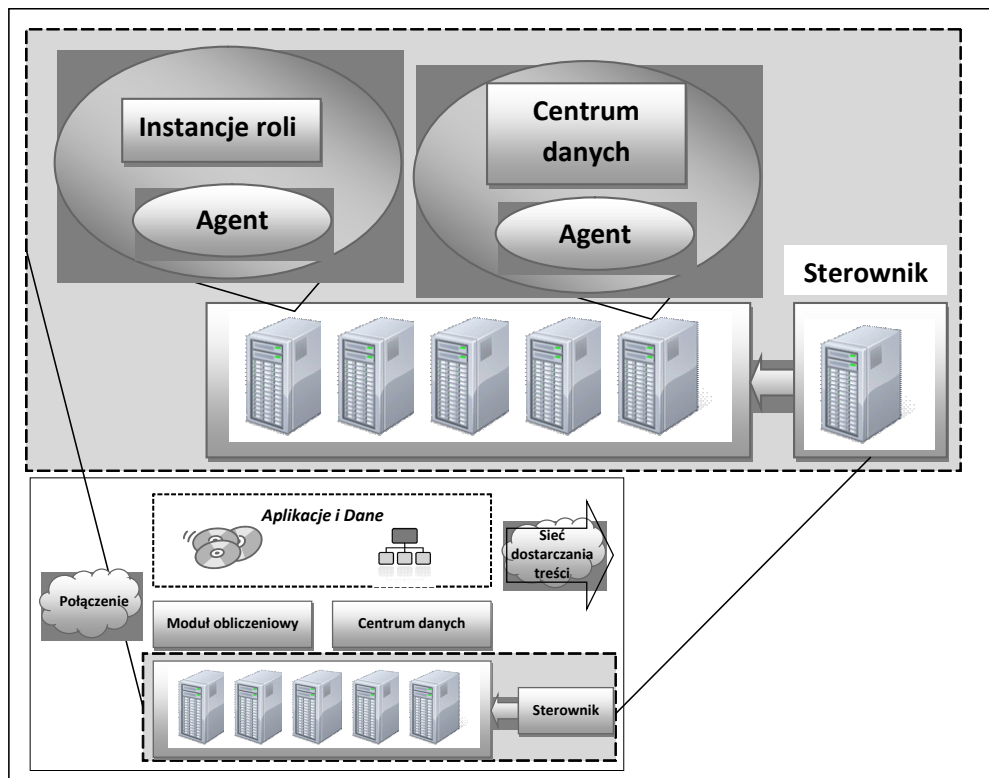
Poważnym wyzwaniem dla menedżerów projektów informatycznych są nieuniknione w praktyce zmiany wprowadzane w trakcie cyklu tworzenia systemu informatycznego. Sam dobór modelu cyklu życia systemu jest w tym przypadku bez znaczenia. Kwestia dotyczy systemów tworzonych zarówno w cyklu liniowym, spiralnym, jak i przyrostowo — iteracyjnym. Często dopiero w fazie wdrażania bądź eksploatacji pilotażowej zgłaszana jest potrzeba zmiany, uzupełnienia wymagania zdefiniowanego początkowo już w fazie modelowania biznesowego, planowania systemu informatycznego bądź specyfikacji wymagań. Problem ten był dostrzegany wcześniej w dziedzinie inżynierii oprogramowania, toteż zaproponowano zróżnicowane metody jego rozwiązywania ujęte w formie zarządzania konfiguracją i zmianami, np. w metodyce RUP. Każda taka zmiana inicjowała swoisty minicykl życia systemu generujący wysokie koszty przekraczające budżet początkowy i opóźnienia w harmonogramie tworzenia systemu. W celu zapobiegania tym zagrożeniom duże firmy, takie jak Hewlett Packard czy Microsoft, zaproponowały wprowadzenie koncepcji i metody — śledzalności oprogramowania (ang. *software traceability*). Śledzalność jest powiązaniem elementów oprogramowania źródłowego z wymaganiem pierwotnym. Jest to zasada charakterystyczna dla organizacji uczących się (ang. *learning organizations*). Powiązanie takich artefaktów oprogramowania jak kod źródłowy, wynikowy i scenariusze testowe znacząco ułatwia dokonywanie zmian bądź wręcz automatyzację ich wdrażania.

Jak zaznacza (Milgin, 2010), zgłaszane zmiany wynikają z nieprecyzyjnego zdefiniowania wymagań przez użytkowników już w fazie definiowania wymagań. Jednak zmiany te są również rezultatem ich niestabilności w związku z szybką transformacją biznesu i stałym udoskonalaniem technologii informatycznej, w tym pojawiania się oprogramowania open source. Zmiany wymagań powodują korekty, a właściwie wzrost budżetu i harmonogramu, a w skrajnych przypadkach upadek systemu. Propozycja zmiany wymagania jest sygnalizowana poprzez powiadomienie pracowników odpowiedzialnych za powiązane z tym wymaganiem oprogramowanie i moduły dokumentacji systemu. Następuje ocena kosztów wprowadzenia nowego wymagania. Kierownictwo decyduje o warunkach akceptacji (lub odrzucenia) oprogramowania i związanych z tym negocjacjach budżetu i harmonogramu. W koncepcji śledzalności PLUS (ang. *traceability PLUS*) przedstawione są powiązania wszystkich modułów oprogramowania, ich testów itp. z wymaganiami. W ten sposób z jednego miejsca można określić wymagania, a następnie zainicjować i uruchomić wszelkie zmiany, jakie są niezbędne w produktach powiązanych z nowym lub zmodyfikowanym wymaganiem.

Utwórz diagram czynności przedstawiający procedurę zastosowania śledzalności w projekcie informatycznym, stosując bloki decyzyjne diagramów czynności.

Zadanie 7.4.

Platforma Windows Azure pozwala na implementację technologii przetwarzania w chmurze (ang. *cloud computing*). Jednym z podstawowych składników platformy Windows Azure jest sterownik (ang. *fabric controller*). Jego miejsce i role w systemie Windows Azure zostały przedstawione na rysunku 7.8.



Rysunek 7.8. Sterownik w systemie Windows Azure

Źródło: Opracowanie własne na podstawie (Chappell, 2010)

Jak precyzuje (Chappell, 2010), wszystkie aplikacje i dane Windows Azure przechowywane są w centrum danych. Sterownik odpowiada za zarządzanie zestawem komputerów oraz oprogramowaniem znajdującym się w tym centrum. Jest on oprogramowaniem działającym w środowisku rozproszonym, replikowanym na grupie komputerów, na których zainstalowano oprogramowanie Windows Azure. W ten sposób zarządza on wszystkimi powiązanymi zasobami: komputerami, przełącznikami sieciowymi, dystrybutorami obciążenia ruchu sieciowego (ang. *load balancer*) i innym sprzętem komputerowym oraz sieciowym. Sterownik posiada oprogramowanie agentkie (ang. *fabric agent*) na każdym z komputerów.

Sterownik może wykonywać następujące czynności:

- ❑ monitoruje wszystkie bieżące aplikacje w chmurze, prowadząc dziennik zdarzeń;
- ❑ desygnuje fizyczne serwery dla obliczeń, w taki sposób, aby optymalizować ich wykorzystanie;
- ❑ zarządza konfiguracją w odniesieniu do każdej aplikacji na Windows Azure w postaci pliku XML, który określa liczbę uruchamianych instancji aplikacji serwerowych;
- ❑ przy każdej instalacji nowej aplikacji konfiguracja jest wykorzystywana do oceny liczby niezbędnych maszyn wirtualnych (ang. *Virtual Machines* — VM);
- ❑ po utworzeniu maszyny wirtualnej każda z nich jest kontrolowana przez sterownik;
- ❑ po wykonaniu określonej aplikacji kontroler automatycznie instaluje i przełącza na wykonanie nowej aplikacji;
- ❑ w przypadku konieczności utworzenia nowej wirtualnej maszyny kontroler resetuje dystrybutor obciążenia ruchu sieciowego;
- ❑ w ramach dostępnej w chmurze infrastruktury sprzętowej według stanu na luty 2011 r. istnieje możliwość utworzenia wirtualnych maszyn o następujących parametrach:
 - ❑ bardzo małe (ang. *extra-small*), jeden rdzeń 1,0 GHz CPU, 768 MB pamięci RAM, 20 GB pamięci masowej;
 - ❑ małe (ang. *small*), jeden rdzeń 1,6 GHz CPU, 1,75 GB pamięci RAM, 225 GB pamięci masowej;
 - ❑ średnie (ang. *medium*), dwa rdzenie 1,6 GHz CPU, 3,5 GB pamięci RAM, 490 GB pamięci masowej;
 - ❑ wielkie (ang. *large*), cztery rdzenie 1,6 GHz CPU, 7 GB pamięci RAM, 1 TB pamięci masowej;
 - ❑ bardzo wielkie (ang. *extra large*), osiem rdzeni 1,6 GHz CPU, 14 GB pamięci masowej, 2,04 TB pamięci masowej.

W instancji dla każdej aplikacji można wykorzystać wiele rdzeni. Oznacza to, że wydajność sprzętu obsługującego aplikacje może się zmieniać w zależności od aktualnego zapotrzebowania na moc obliczeniową.

Powyżej opisane czynności i związane z nimi parametry w Windows Azure przedstaw w postaci diagramu czynności.

Zadanie 7.5.

Jednym z wiodących podejść w zarządzaniu usługami informatycznymi zarówno w sektorze publicznym, jak i w przedsiębiorstwach jest ITIL (*Information Technology Infrastructure Library*). Trzecia rewizja tego zestawu dobrych praktyk skupia

się na pięciu obszarach zarządzania usługami: strategii, projektowaniu, wdrażaniu, eksploatacji oraz ciągłym doskonaleniu. Zgodnie z (ITIL Training, 2010) autorzy standardu ITIL proponują następujący zestaw wskazówek oraz koncepcji w obszarze ciągłego doskonalenia usług:

1. *Określ, co mierzyć.* Jakie dokładnie są obszary i przedmioty pomiaru?
2. *Zdefiniuj obszary, w których można dokonać pomiaru.* Jakie są wymagania względem poziomów usług, ograniczenia budżetowe i wymagana wydajność infrastruktury IT?
3. *Zbierz dane.* Jakie są fakty i nieprzetworzone dane ilościowe w odniesieniu do celów operacji usługowych? Dane winny być zebrane różnymi metodami poprzez zróżnicowane media.
4. *Przetwórz dane.* Dane winny być przetwarzane w nawiązaniu do istotnych czynników powodzenia (ang. *Critical Success Factors*). Dane segmentuje się z myślą o łatwiejszym porównywaniu i analizie.
5. *Analizuj dane.* Jakie braki i niedoskonałości udało się zidentyfikować? Jaki jest wpływ zebranych informacji na organizację?
6. *Opracuj wzorce użytecznych danych i utwórz raport w wybranej formie.* W jakim formacie zaprezentować dane menedżerom firmy? Dane są przekazywane i objaśniane w taki sposób, by odpowiadały oczekiwaniom, potrzebom i wymaganiom firmy, jednocześnie oferując rozwiązania i potencjalne działania na rzecz udoskonalania infrastruktury i zasobów informatycznych.
7. *Implementuj zmiany.* Po zidentyfikowaniu obszarów zmian zebrane dane mogą być katalizatorem inicjacji zmiany. Obszar ciągłego doskonalenia usług obejmuje liczne opcje i alternatywne ścieżki działań, które mogą być podjęte w celu udoskonalenia zarządzania zasobami informatycznymi w firmie.

Mając na uwadze, że doskonalenie usług to działanie o charakterze iteracyjnym, wykorzystaj diagram czynności do przeprowadzenia wizualizacji procesu doskonalenia usług.

Zadanie 7.6.

Budowa chmury prywatnej działającej zgodnie z technologią Cloud Computing realizowana jest w kilku etapach, do których należą między innymi (Microsoft Corporation, 2009a):

- ☐ instalacja systemu operacyjnego;
- ☐ instalacja rozwiązania monitoringu;
- ☐ instalacja rozwiązania do zarządzania środowiskiem wirtualnym;
- ☐ instalacja portalu samoobsługowego chmury;
- ☐ wdrożenie systemu automatyzacji czynności administracyjnych.

Sprecyzuj istotę powyższych etapów i opracuj diagram czynności, wykorzystując oznaczenia przepływów decyzyjnych i współbieżnych.

Zadanie 7.7.

Zastosowanie procedury bezpieczeństwa na urządzeniach sieciowych obejmuje między innymi ograniczanie dostępu do przełącznika przez poszczególne urządzenia końcowe (komputery, drukarki sieciowe itp.). Procedura autoryzacyjna oparta jest na adresach fizycznych MAC przypisanych przez producenta urządzenia do każdej karty sieciowej. Adresy te są niezbędne do komunikowania się komputerów w ramach sieci lokalnych. Przykładem takiego adresu zapisywanego w notacji szesnastkowej jest 00:AA:BB:22:33:44. Każda karta sieciowa na świecie ma inny adres fizyczny, co pozwala na jednoznaczną identyfikację komputera w ramach sieci lokalnych.

Podstawowa zasada działania opiera się na ręcznym przypisaniu wybranej liczby adresów fizycznych do pojedynczego portu przełącznika i tym samym uniemożliwieniu dostępu do sieci przez urządzenia z innymi adresami fizycznymi. Administrator sieci najpierw włącza procedurę bezpieczeństwa na porcie lub grupie portów. Następnie określa maksymalną liczbę adresów fizycznych, jaka może zostać przypisana do portu. W większości przypadków może to być tylko jeden adres. Następnie administrator jest odpowiedzialny za przypisanie adresów fizycznych do poszczególnych portów. Kolejną czynnością jest wyspecyfikowanie sposobu reagowania przełącznika na próbę uzyskania nieautoryzowanego dostępu. Najbardziej radykalną reakcją jest fizyczne wyłączenie portu. Wymaga to późniejszego ręcznego włączenia portu. Alternatywą jest zapobieganie ruchowi z opcjonalnym logowaniem tej informacji w dzienniku zdarzeń.

Opracuj diagram czynności dla procedury bezpieczeństwa dla przełączników sieciowych realizowanej zgodnie z powyższym opisem.



Diagramy sekwencji

8.1. Ćwiczenia

Ć W I C Z E N I E

8.1 System monitorowania dostępności miejsc parkingowych w mieście

Obecnie bardzo dokuczliwym problemem każdego większego miasta jest brak miejsc parkingowych. Każde miasto posiada pewną liczbę parkingów płatnych w ścisłym centrum miasta. Są to między innymi place parkingowe oraz podziemne lub wielopoziomowe parkingi, na które wjeżdża się przez otwartą bramkę. Istnieją cztery strefy parkowania, w których można parkować różną liczbę godzin. Odpowiednio strefa:

- ☐ pomarańczowa — do 2 godzin;
- ☐ żółta — do 5 godzin;
- ☐ zielona — do 10 godzin;
- ☐ niebieska — dowolną liczbę godzin.

Wysokość opłaty zależy również od pory dnia oraz dnia tygodnia. Najwyższe opłaty pobierane są w godzinach 8:00 – 18:00 od poniedziałku do piątku. W weekendy opłaty nie są pobierane.

W każdej strefie wydzielono po kilka tysięcy miejsc parkingowych. Informacja o dostępności wolnych miejsc parkingowych wyświetlana jest na *ekranach typu LED* zawieszonych nad ulicami, rozmieszczonych w różnych częściach miasta. Dzięki temu kierowcy mogą optymalnie zaplanować miejsce postoju swojego samochodu.

Wśród miejsc parkingowych rozróżnia się trzy rodzaje, które pozwalają na zróżnicowanie opłat dla parkowania odpowiednio:

- ☐ samochodów osobowych;

- ☐ samochodów dostawczych;
- ☐ motocykli.

Z opłat za parkowanie zwolnione są osoby niepełnosprawne. Każdy z parkingów ma specjalnie wydzieloną strefę parkowania dla osób niepełnosprawnych znajdującą się jak najbliżej wyjścia.

System monitorowania dostępności miejsc parkingowych łączy wiele elementów składających się na wiele zintegrowanych komponentów. Są to:

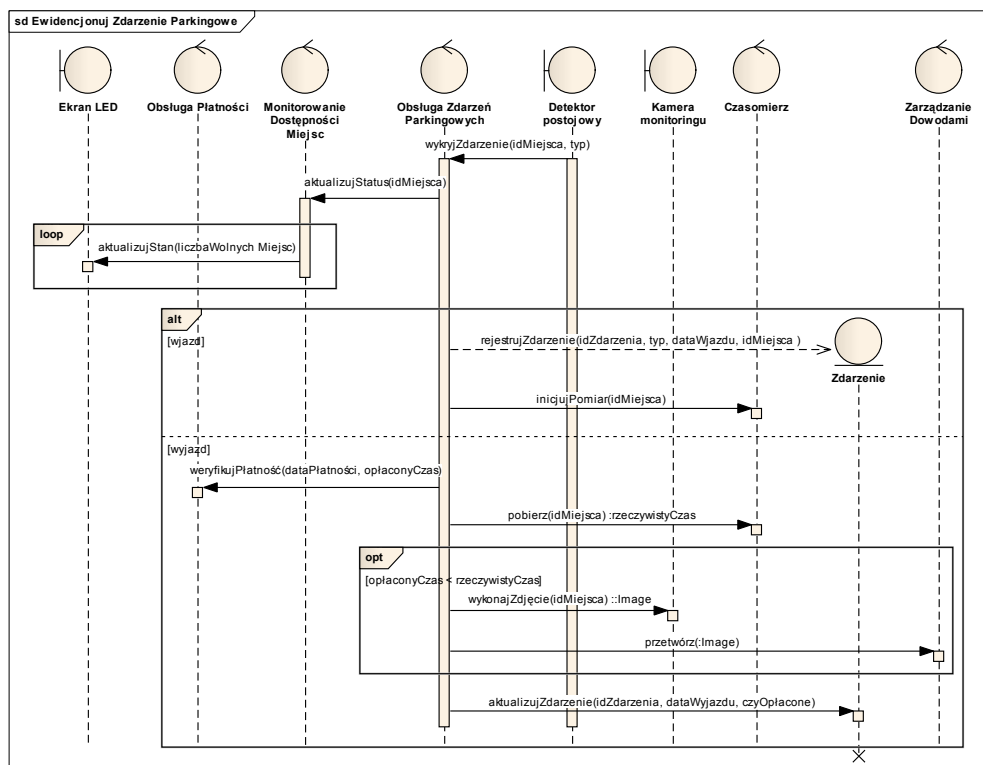
- ☐ bramki wyposażone w elektroniczne czujniki na wjeździe i wyjeździe;
- ☐ *detektory postojowe*;
- ☐ *kamery monitoringu*;
- ☐ *parkometry*;
- ☐ *ekrany LED*.

Detektory postojowe pozwalają na monitorowanie, czy kierowca w wyznaczonym czasie opuścił miejsce parkingowe. *Kamery monitoringu* mają dodatkowo uwierzytelić popełnione wykroczenie. *Obsługa płatności* w systemie umożliwia automatyczne dokonywanie płatności dla kierowców i firm, które zarejestrują swoje dane w systemie. W tym przypadku czas postoju pojazdu na parkingu jest generowany automatycznie na podstawie wjazdu i wyjazdu z danego parkingu z wykorzystaniem w tym celu *Czasomierzy*. W okresie jednego miesiąca rachunek kierowcy bądź firmy jest obciążany kwotą wynikającą z czasu parkowania. Płatność można wносить również w parkometrach, używając w tym celu monet, banknotów, wykupionych żetonów, bezstykowych kart abonamentowych lub kart płatniczych. Kolejną możliwością wniesienia opłat jest przysyłanie przez kierowcę na wskazany numer telefonu wiadomości SMS z informacją o wjeździe na płatne miejsce parkingowe i wyjeździe. W przypadku gdy czas postoju samochodu przekracza czas opłacony, następuje wykonanie zdjęcia i zarejestrowanie tego zdarzenia w module *Zarządzanie dowodami*.

Informacja o obciążeniu miejsc parkingowych na poszczególnych parkingach jest na bieżąco przesyłana do serwera zarządzającego miejscami parkingowymi, który z kolei przekazuje te informacje do sieci *ekranów LED* rozmieszczonych w różnych częściach miasta. Całością systemu zarządza moduł *Obsługa zdarzeń parkingowych*, który w celu aktualizacji liczby miejsc parkingowych komunikuje się z modulem *Monitorowanie dostępności miejsc*.

Należy opracować diagram sekwencji dla przypadku użycia *Ewidencjonuj zdarzenie parkingowe* (rysunek 8.1) obejmującego wykrycie zdarzenia wjazdu bądź wyjazdu i stosowną reakcję systemu na wykryte zdarzenie. Niniejszy przypadek użycia ujmuje ponadto automatyczny komponent reakcji na nieautoryzowane bądź niedostatecznie opłacone parkowanie na danym miejscu parkingowym.





Rysunek 8.1. Diagram sekwencji ewidencjonowania zdarzenia parkingowego

Ć W I C Z E N I E

8.2 Obieg dokumentów z podpisem elektronicznym

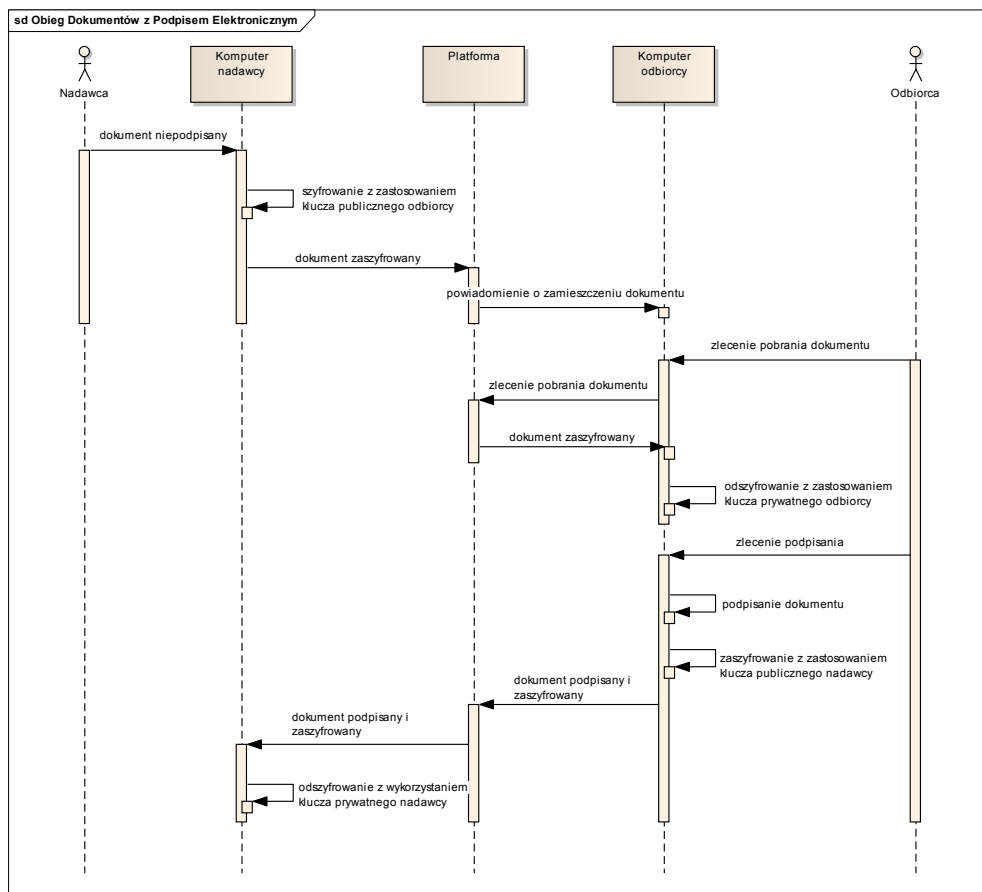
Elektroniczny obieg dokumentów zawierający w sobie możliwość składania podpisu elektronicznego znacząco przyspiesza realizację transakcji gospodarczych i procesów biznesowych. Jest to istotne szczególnie dla współczesnych instytucji finansowych. Platforma ta składa się z dwu narzędzi:

- ❑ serwisu umożliwiającego wymianę dokumentów między firmą a kontrahentem;
- ❑ mobilnego podpisu elektronicznego zatwierdzanego przez centrum certyfikacji uprawnione przez Ministerstwo Gospodarki do autoryzowania podpisów elektronicznych.

Takie rozwiązanie umożliwia zdalną wymianę i akceptację dokumentów pomiędzy partnerami biznesowymi. W ramach takiej platformy partnerzy wyposażeni są w komputer bądź urządzenie mobilne dysponujące dostępem do Internetu, w zależności od kierunku transmisji zwane odpowiednio *komputerem nadawcy* lub *komputerem*

odbiorcy. Połączenie nawiązywane jest przez przeglądarkę internetową, nie wymaga tym samym specjalistycznego, dedykowanego oprogramowania. Połączenie pomiędzy partnerami jest szyfrowane i funkcjonuje na zasadach infrastruktury kłucza publicznego.

W odniesieniu do powyższego opisu należy zaproponować konceptualny diagram sekwencji (rysunek 8.2) ilustrujący ogólny schemat interakcji pomiędzy nadawcą a odbiorcą. Scenariusz zakłada, że klucze publiczne zostały wcześniej wymienione pomiędzy stronami transakcji.



Rysunek 8.2. Diagram sekwencji obiegu dokumentów z podpisem elektronicznym

**Wskazówki dydaktyczne**

Zaproponowany diagram ma charakter konceptualny, dopuszczalny w diagramach sekwencji języka UML. Istotą jest tutaj czytelność diagramu dla użytkownika niezwiązanego z techniką. Tym samym diagram nie przedstawia sekwencji wywoływania metod pomiędzy różnymi klasyfikatorami, lecz w sposób nieformalny ilustruje sekwencje dokumentów i zleceń wymienianych pomiędzy różnymi obiektami uczestniczącymi w interakcji.

Ć W I C Z E N I E**8.3 System ogrzewania budynków**

Połączenie sprzętu informatycznego z systemami ogrzewania budynków może być źródłem znaczących oszczędności. Jak podaje (Ramsey, 2011), pierwsze systemy były stosunkowo proste. Ich działanie polegało na włączeniu systemu w określonych godzinach pracy i znaczącym zredukowaniu użycia energii przez odpowiedni system w czasie nieobecności pracowników. Powszechnie system ten nazywa się HVAC (*Heating, Ventilation, Air-Conditioning*) i zawiera w sobie ogrzewanie, wentylację i klimatyzację. Ten tradycyjny system obsługuje wszystkie pomieszczenia. Tymczasem ich zajętość jest w trakcie godzin pracy zróżnicowana.

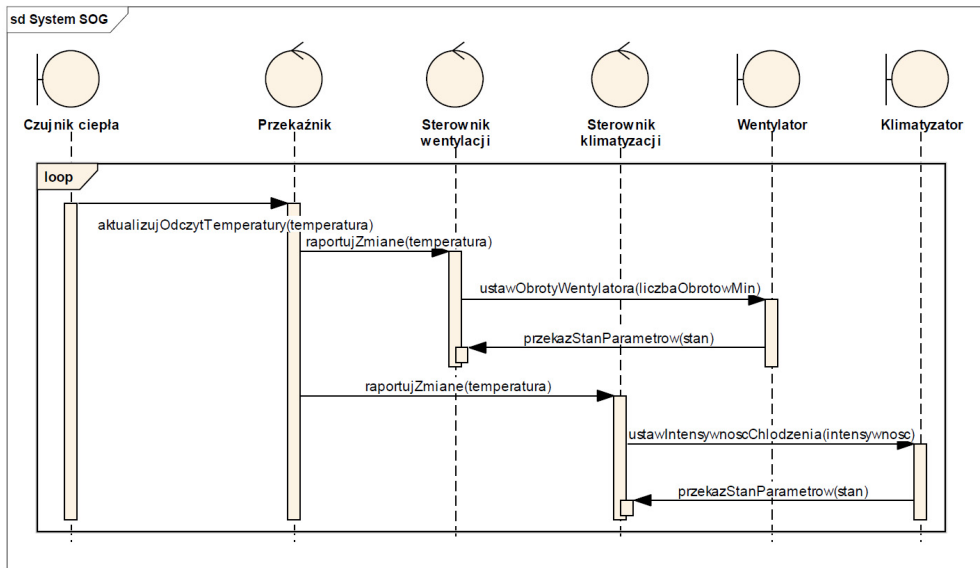
Zespół naukowy Uniwersytetu Kalifornijskiego w San Diego stworzył system inteligentnej obsługi użytkownika energii poprzez sieć czujników. Poniższy przypadek pokazuje, w jaki sposób system może reagować na nadmierne ciepło w pomieszczeniu. *Czujnik ciepła* oraz *Sterowniki wentylacji i klimatyzacji* w czasie rzeczywistym reagują na zajętość poszczególnych pomieszczeń, dostosowując dopływ powietrza. Sygnał z *Czujnika ciepła* jest przekazywany do *Sterowników* poprzez *Przełącznik*. Na podstawie sygnałów zajętości pomieszczenia *Sterowniki* dostosowują dopływ energii i zmieniają moc pracy odpowiednich urządzeń: *Wentylatora* i *Klimatyzatora*. Tak więc serwis HVAC odbiera komunikaty i wykonuje odpowiednie akcje. W ten sposób następuje oszczędność 10 – 15% energii w budynku czy na danym piętrze.

Należy opracować diagram sekwencji dla opisywanego systemu ogrzewania budynków (rysunek 8.3), uwzględniając jedynie czujnik ciepła oraz sterowniki związane z wentylacją i klimatyzacją.

**Wskazówki dydaktyczne**

Zarówno czujnik ciepła, jak również wentylator oraz klimatyzator działają na styku pomiędzy systemem a jego otoczeniem. Stąd też zastosowano stereotyp boundary. Z kolei przełącznik oraz sterowniki wentylacji i klimatyzacji na bieżąco przetwarzają sygnały pochodzące odpowiednio od czujnika temperatury lub otoczenia, zastosowano zatem stereotyp do control.





Rysunek 8.3. Diagram sekwencji systemu ogrzewania budynków

Ć W I C Z E N I E

8.4

System elektronicznej rezerwacji i zakupu biletów

Dynamiczny rozwój infrastruktury dedykowanej rozrywce w Polsce, w szczególności budowa nowych obiektów widowiskowo-sportowych, stwarza stymulujące warunki rozwoju systemów elektronicznej rezerwacji i zakupu biletów. Wdrażanie tego typu systemów umożliwia w szczególności zapoznanie się z katalogiem planowanych imprez w komfortowych warunkach, minimalizację niedogodności towarzyszących zakupowi biletu na imprezę masową czy też obniżenie łącznych kosztów administracyjnych. Nabywca biletu otrzymuje kompletną informację w zakresie dostępności i klasy poszczególnych miejsc wolnych i może poświęcić dowolną ilość czasu na wybór optymalnego miejsca lub miejsc, nie ponosząc przy tym dodatkowych kosztów.

Podstawowe przypadki użycia systemu rezerwacji i zakupu biletów obejmują:

- ☐ Definiuj listę imprez w obszarze zainteresowania;
- ☐ Dodaj miejsce do rezerwacji;
- ☐ Złóż rezerwację grupową;
- ☐ Zatwierdź rezerwację;
- ☐ Realizuj płatność online.

Wykonanie wymienionych przypadków użycia uwarunkowane jest zalogowaniem się do systemu po uprzedniej rejestracji w serwisie. Z punktu widzenia administracji systemu wyróżnić można następujące przypadki użycia:

- ❑ *Zarządzaj katalogiem imprez;*
- ❑ *Zarządzaj miejscami;*
- ❑ *Zarządzaj rezerwacjami dokonanymi;*
- ❑ *Definiuj schemat cenowy;*
- ❑ *Definiuj schematów odbioru biletu.*

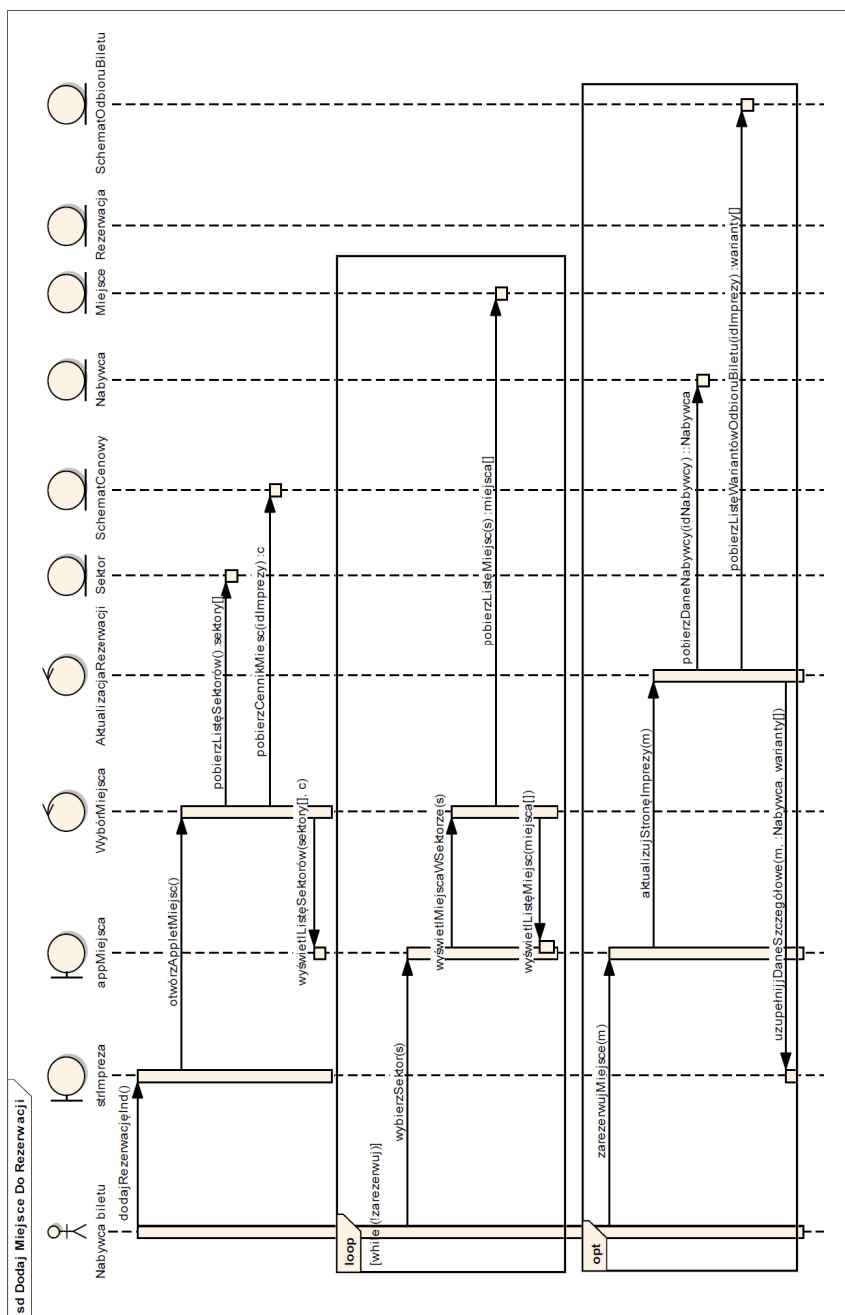
Kluczowe znaczenie w opisywanym systemie posiada przypadek użycia *Dodaj miejsce do rezerwacji*. Przypadek ten inicjowany jest z poziomu strony szczegółów imprezy zawartej na liście imprez w obszarze zainteresowania. W efekcie wyświetlany jest schemat obiektu w podziale na sektory. Poszczególne sektory charakteryzują się zróżnicowaną polityką cenową definiowaną przez administratora systemu w ramach przypadku użycia *Definiuj schemat cenowy*. Ceny w poszczególnych sektorach ujmowane są na schemacie obiektu.

Po wybraniu sektora wyświetlana jest lista miejsc w danym sektorze. Kolorystyka miejsc wskazuje na status poszczególnych miejsc (wyłączone ze sprzedaży/zarezerwowane/wolne). Projektanci systemu zdecydowali się na umożliwienie przeglądania wszystkich sektorów, bez względu na liczbę wolnych miejsc w danym sektorze. Rezerwujący może w dowolnym momencie zmienić przeglądany sektor.

W efekcie wybrania konkretnego miejsca strona imprezy zostaje uzupełniona o informację o zarezerwowanym miejscu wraz z danymi nabywcy biletu oraz listą wariantów odbioru biletu. Poszczególne warianty (np. dostawa przez kuriera, odbiór samodzielny w punkcie informacyjnym, wydruk samodzielny) charakteryzują się odmiennymi cenami. Do danej imprezy można przypisać kilka miejsc poprzez wielokrotną realizację przypadku użycia. Ostateczne potwierdzenie chęci uczestnictwa w imprezie, co wiąże się z przydzieleniem numeru rezerwacji, wysłaniem potwierdzenia elektronicznego oraz naliczeniem kompletu opłat, jest już przedmiotem przypadku użycia *Zatwierdź rezerwację*.

Należy opracować diagram sekwencji dla przypadku użycia *Dodaj miejsce do rezerwacji* (rysunek 8.4).





Rysunek 8.4. Diagram sekwencji dodawania miejsca do rezerwacji dla systemu elektronicznej rezerwacji i zakupu biletów

Ć W I C Z E N I E

8.5 **Głosowy system obsługi magazynów**

We współczesnych magazynach żywności, leków, książek itp. wykorzystuje się zautomatyzowane głosowe systemy koordynacji operacji magazynowych, zwane również systemem sterowania głosem. Obsługują one sieci sklepów, supermarketów czy chłodni. Systemy te wspomagają wykonywanie zadań przez pracowników magazynów przewożących towary na wózkach widłowych.

Każdy pracownik wyposażony jest w terminal głosowy oparty na bezprzewodowym zestawie słuchawkowym. Terminal koordynuje poruszanie się wózka pomiędzy alejkami i ładowanie na wózek konkretnych towarów. Podstawowy przypadek użycia systemu, tj. *Koordinuj wykonanie operacji magazynowej*, obejmuje ciąg interakcji prowadzących do dostarczenia partii określonego towaru do punktu odbioru. Komunikacja pomiędzy systemem zarządzania magazynem a terminalem głosowym jest dwukierunkowa — system wydaje głosowe polecenia oraz weryfikuje prawidłowość ich wykonania przez operatora. Tym samym po zatrzymaniu się wózka widłowego we wskazanej alejce, czy też pobraniu wyspecyfikowanej przez system liczby pozycji towarów z określonej półki wymagane jest słowne potwierdzenie parametrów wykonanej operacji. W przypadku wykrycia rozbieżności system wydaje polecenia korygujące.

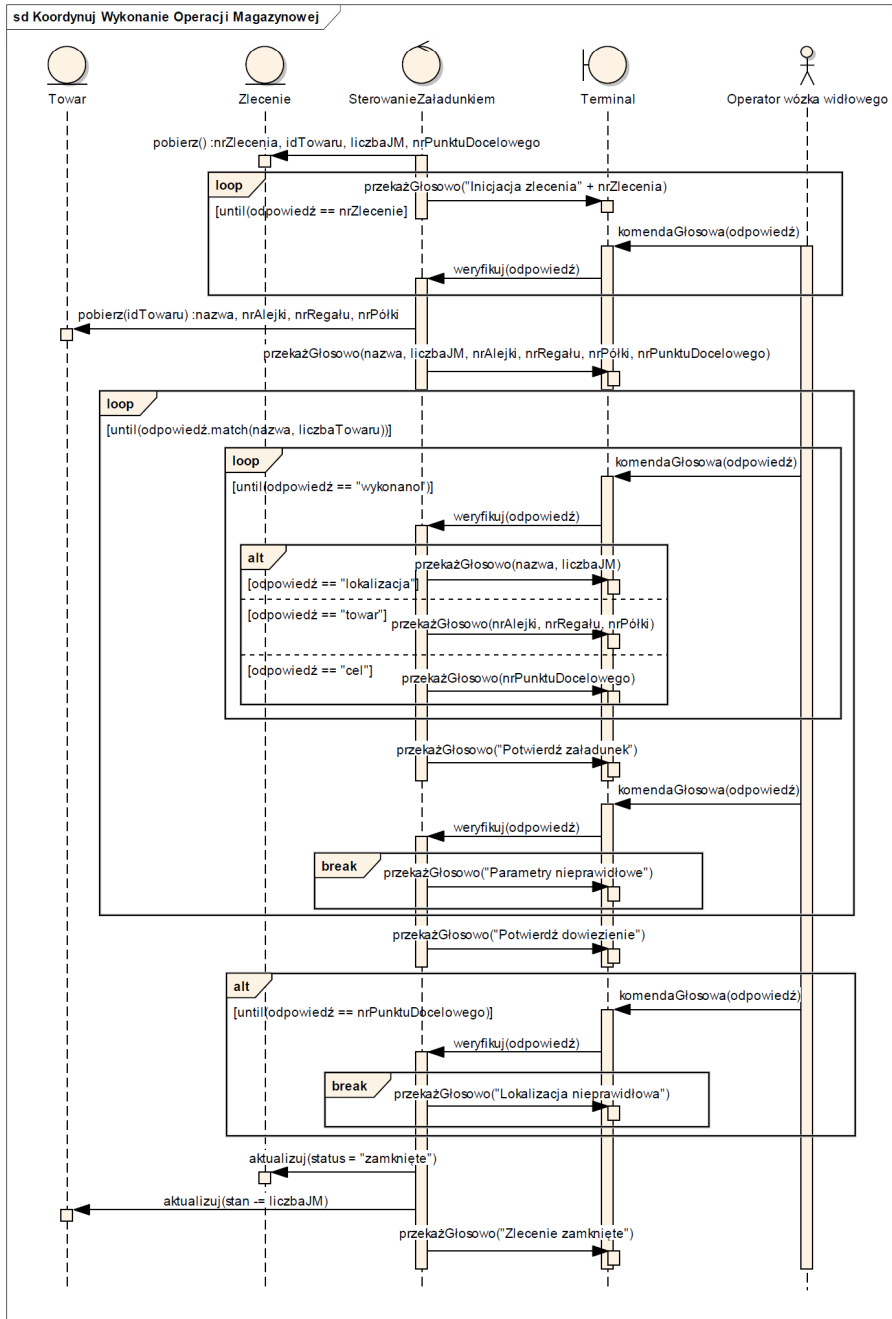
Przyjmując, że w interakcji uczestniczą klasy przechowujące *Towar* i *Zlecenie*, klasa sterująca *SterowanieZaładunkiem*, klasa graniczna *Terminal* oraz aktor *Operator wózka widłowego*, należy zaproponować chronologiczną listę metod, a następnie zaprezentować przypadek użycia *Koordinuj wykonanie operacji magazynowej* w postaci diagramu sekwencji języka UML (rysunek 8.5).

Ć W I C Z E N I E

8.6 **Aukcje internetowe Allegro**

W systemie aukcji internetowych Allegro (<http://www.allegro.pl>) zidentyfikować można szeroki wachlarz przypadków użycia, np. *Zarejestruj się*, *Zaloguj się*, *Dokonaj płatności*, *Wystaw towar na aukcji*, *Kupuj na aukcji*, *Uczestnicz w licytacji*. Do grupy tych przypadków użycia dotyczących aukcji internetowej należy przypadek użycia *Wystaw komentarz*.

Po zainicjowaniu przypadku użycia system pobiera z bazy danych wszystkie obiekty klasy *Aukcja*, których atrybut *status* przyjmuje wartość *zakończona*. Za inicjację tej operacji odpowiedzialna jest klasa sterująca *WystawianieKomentarza*. Następnie pobierane są wszystkie obiekty klasy *Uczestnik*, których *idUczestnika* odpowiada identyfikatorom uczestników we wszystkich uprzednio pobranych aukcjach.



Rysunek 8.5. Diagram sekwencji koordynowania wykonania operacji magazynowej

Na ekranie (formularz `IAukcjeNieskomentowane`) wyświetlana jest lista aukcji z uwzględnieniem nazwy aukcji oraz nazwy i rangi uczestnika. Wybranie przez aktora `Uczestnik` dowolnej aukcji z listy powoduje wyświetlenie formularza `IKomentowanie`. Wybrana aukcja identyfikowana jest po nazwie, co uwzględniają parametry operacji.

Na ocenę transakcji wystawianej przez `Uczestnika` składają się:

- ☐ zgodność przedmiotu z opisem;
- ☐ kontakt ze sprzedającym;
- ☐ czas realizacji zamówienia;
- ☐ koszt wysyłki.

Treść i ocena wprowadzane są w uprzednio wyświetlonym formularzu. W istocie funkcjonalność ta ma charakter opcjonalny, gdyż `Uczestnik` może zrezygnować z wprowadzania komentarza w tym konkretnym momencie. Po zweryfikowaniu kompletności danych przez klasę sterującą `WystawianieKomentarza` następuje zapisanie `idAukcji`, `idUczestnika`, treści oraz oceny w bazie danych. Komplet wymienionych danych reprezentowany jest przez klasę przechowującą `Komentarz`. Jednocześnie zamykany jest formularz `IKomentowanie`, a na formularzu `IAukcjeNieskomentowane` wyświetlane jest potwierdzenie poprawnego ukończenia całej procedury. Jeśli jednak okaże się, że nie wszystkie dane zostały wprowadzone, formularz `IKomentowanie` pozostaje otwarty celem umożliwienia usunięcia braków, a komunikat błędu wyświetlany jest właśnie na tym formularzu. Próba wystawienia komentarza jest powielana aż do momentu zakończenia jej sukcesem bądź całkowitej rezygnacji z wystawiania komentarza.

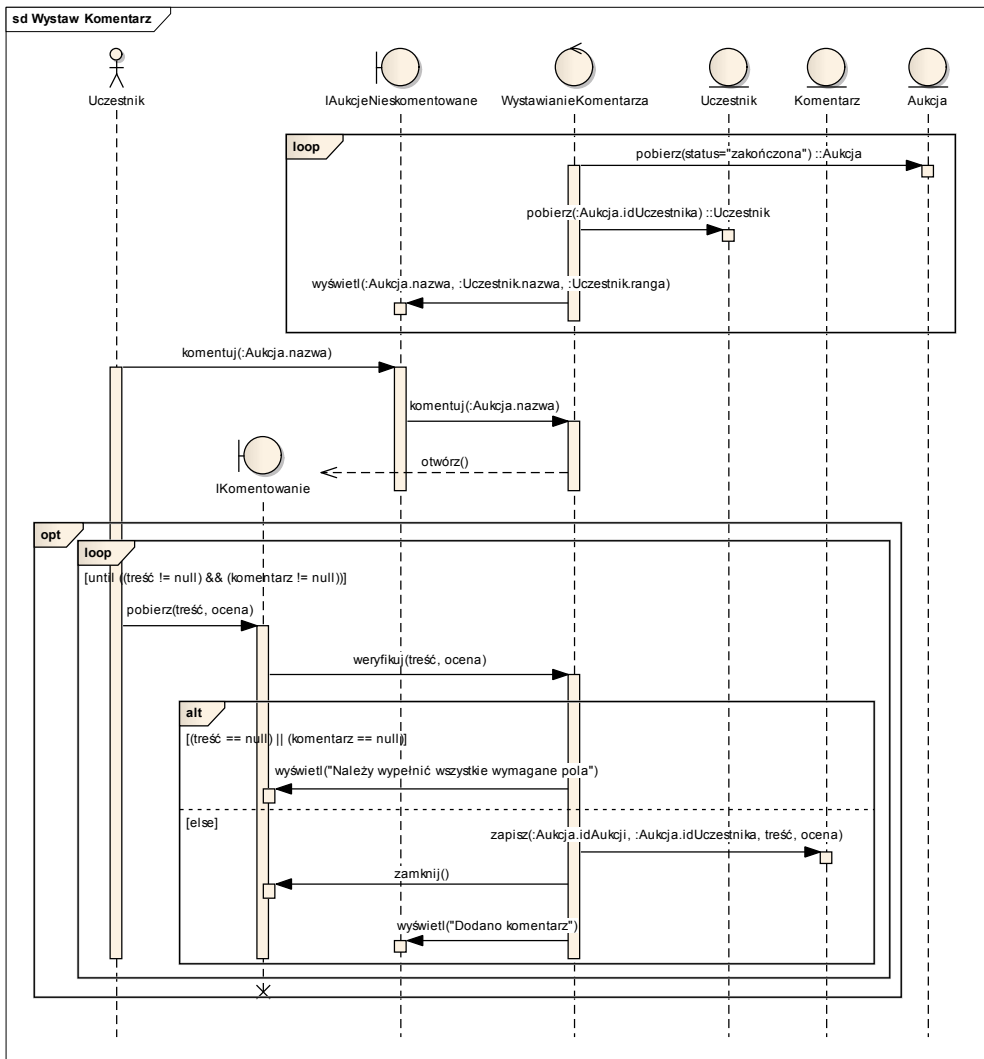
Na podstawie zamieszczonego opisu należy opracować diagram sekwencji dla przypadku użycia *Wystaw komentarz* (rysunek 8.6).

Ć W I C Z E N I E

8.7 System wideokonferencji

Obecnie coraz popularniejszym środkiem komunikowania się, szczególnie w dużych korporacjach czy w środowisku naukowym, są wideokonferencje. Oszczędzają one wiele czasu i wysiłku, uczestnicy nie płacą za bilety lotnicze czy rezerwacje hotelowe. Ważne, strategiczne sprawy można omówić i rozstrzygnąć w międzynarodowym, rozległym geograficznie zespole według aktualnych potrzeb. Do najpopularniejszych pakietów oprogramowania wideokonferencyjnego zalicza się systemy Cisco WebEx, Adobe Acrobat Connect, Marratech, Microsoft NetMeeting czy Skype.

Prowadzący wideokonferencję inicjuje nową sesję poprzez nadanie jej nazwy i dwukrotne wpisanie hasła, wyznaczenie daty sesji oraz czasu trwania spotkania. W określonym przez moderatora czasie uczestnicy uprawnieni do udziału w spotkaniu mogą dołączać się do niego poprzez wpisywanie haseł pozwalających na udział w sesji.



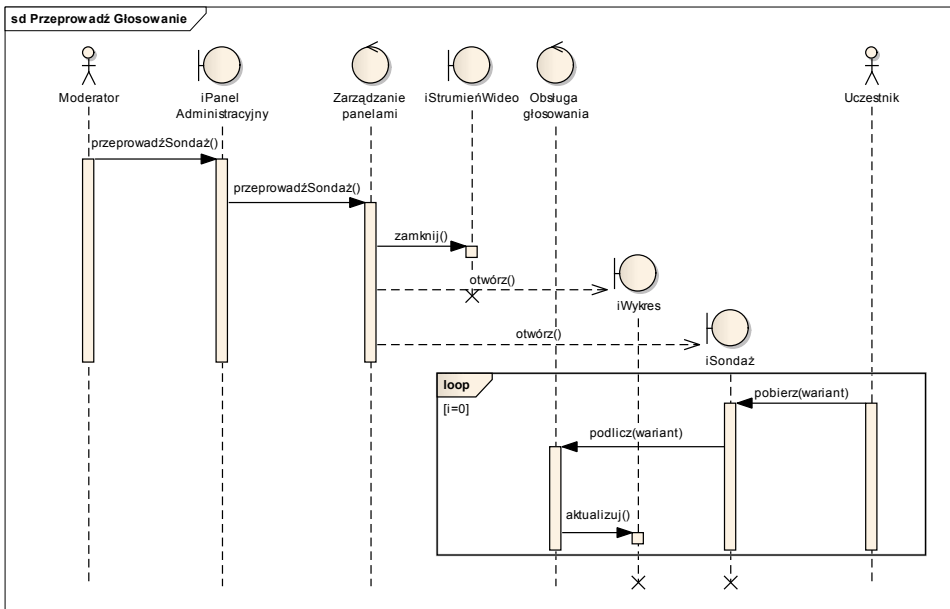
Rysunek 8.6. Diagram sekwencji procesu wystawiania komentarza na aukcji internetowej

Uczestnicy spotkania mają dostęp do usług oprogramowania videokonferencyjnego, tj. listy uczestników, czatu, białej tablicy i kratownicy (ujęć z kamer uczestników). Każdy z uczestników spotkania powinien być wyposażony w mikrofon i kamerę internetową. Uczestnicy, którzy nie posiadają kamery, mogą się wdzwonić i uczestniczyć w spotkaniu na zasadzie telekonferencji. Każdy uczestnik spotkania na tablicy uczestników może zaznaczać wybierane przez siebie opcje wyrażone ikonami i emotikonami, jak zgłoszenie się do wypowiedzi, aplauz, akceptacja, negacja, przerwa. Przebiegiem całej konferencji zarządza moderator. Decyduje on o udzieleniu głosu

zgłaszającemu się do wypowiedzi uczestnikowi i przekazuje mu rolę prezentera. Aktywuje on i wycisza wypowiedzi poszczególnych uczestników. Może on również zarządzać głosowania, przeprowadzać sondaże z wyliczeniem liczby punktów lub wyników wyrażanych procentowo. Moderator nadaje wysoki priorytet danemu problemowi poruszanemu w dyskusji poprzez wprowadzenie wykrzyknika.

W trakcie przebiegu wideokonferencji poszczególni uczestnicy mogą przysyłać wypowiedzi tekstowo w okienku czatu. Bardzo przydatną rolę pełni biała tablica, na której można wyrażać różne opinie, kreślić grafikę itp. Tablica zawiera funkcjonalności w postaci: zapisywania tekstów w różnych krojach i wielkościach czcionek, linii, mazaków kolorowych, wykresów, kształtów, kolorów, gumki i wskaźnika. Istotne jest to, że zarówno moderator, jak i prezenter mogą umieszczać tam dokumenty, prezentację, grafikę, filmy, udostępniać aplikacje, a także pulpit swojego komputera.

Należy opracować diagram sekwencji prezentujący wycinek funkcjonalności systemu wideokonferencji — przypadek użycia o nazwie *Przeprowadź głosowanie* (rysunek 8.7). Po wywołaniu przez *Moderatora* operacji przypisanej do stosownego przycisku *iPaneluAdministracyjnego* następuje otwarcie okien *iWykres* oraz *iSondaż* przy jednoczesnym zamknięciu okna *iStrumieńWideo*. Każdy *Uczestnik* wybiera jeden z *wariantów* odpowiedzi (pętla *for* oparta na zmiennej *i* — przy czym *i* na wejściu przyjmuje wartość 0). Wraz z *pobranem* każdego kolejnego wariantu następuje *podliczenie* statystyk i aktualizacja *iWykresu* w czasie rzeczywistym.



Rysunek 8.7. Diagram sekwencji przeprowadzania głosowania w systemie wideokonferencji

8.2. Zadania

Zadanie 8.1.

Na podstawie funkcjonalności serwisu Allegro (<http://www.allegro.pl>) opracuj diagramy sekwencji dla poszczególnych przypadków użycia wymienionych w ćwiczeniu 8.6, tj. *Zarejestruj się*, *Zaloguj się*, *Dokonaj płatności*, *Wystaw towar na aukcji*, *Kupuj na aukcji*, *Uczestnicz w licytacji*. Postaraj się samodzielnie rozpoznać procedury działania poszczególnych przypadków użycia.

Zadanie 8.2.

Posiłkując się treścią ćwiczeń 7.3 i 10.6, przeprowadź krytyczną selekcję wymienionych w dalszej części zadania klasyfikatorów i komunikatów pod kątem realizacji wybranego przypadku użycia aplikacji biura obsługi nieruchomości. Aplikację tę należy zaliczyć do dziedziny geomatyki (geoinformatyki), która pozwala na pozyskiwanie, przetwarzanie, gromadzenie, analizowanie i udostępnianie informacji geograficznej z systemów informacji przestrzennej GIS (*Geographic Information Systems*) poprzez środki i mechanizmy techniczne, prawne, ekonomiczne i administracyjne. Następnie uzupełnij listę klasyfikatorów oraz komunikatów i przygotuj kompletny diagram sekwencji specyfikujący wybrany przypadek użycia.

Zgodnie z (Bluebell, 2011) klasyfikatory biorące udział w interakcji w tym systemie obejmują m.in.:

- ☐ różne formularze ekranowe aplikacji desktopowej;
- ☐ różne formularze ekranowe aplikacji mobilnej;
- ☐ formatki serwisu internetowego;
- ☐ administratora infrastruktury;
- ☐ pracownika biura nieruchomości;
- ☐ klienta biura nieruchomości;
- ☐ tabele bazy danych nieruchomości;
- ☐ klienta biura nieruchomości;
- ☐ niezbędne klasy sterujące.

Przesyłane w systemie komunikaty symbolizują wywołania operacji, które pozwalają na:

- ☐ pozyskiwanie danych o lokalizacji nieruchomości;
- ☐ wykorzystanie danych do prezentacji na mapie cyfrowej;
- ☐ komunikację z klientami biura nieruchomości za pośrednictwem wypełnionych formularzy;

- ☐ mapowanie obiektowo-relacyjne;
- ☐ dostarczanie żądanej funkcjonalności przez Web Service;
- ☐ rejestrację klienta;
- ☐ uwierzytelnianie użytkownika (zarówno pracownika, jak i klienta);
- ☐ autoryzację użytkownika (zarówno pracownika, jak i klienta);
- ☐ realizację zapytań do relacyjnej bazy danych poprzez mechanizm ADO.NET będący częścią .NET Framework;
- ☐ aktualizację danych przechowywanych w systemie nieruchomości;
- ☐ zamieszczanie zdjęć nieruchomości;
- ☐ edycję danych nieruchomości;
- ☐ edycję zdjęć;
- ☐ usuwanie danych;
- ☐ identyfikację współrzędnych geograficznych poprzez system GPS i zewnętrzny moduł komunikacji w technologii Bluetooth;
- ☐ wysyłanie wiadomości do klienta, opiekuna oferty lub menedżera.

Zadanie 8.3.

W nawiązaniu do zadania 8.2 opracuj diagram sekwencji, który pozwala na wyświetlenie obiektów na mapie cyfrowej poprzez funkcję wyśrodkowania mapy, ustawienia opcji mapy:

- ☐ wyśrodkowanie mapy, ustawienie jej typu i powiększenia;
- ☐ ustawienie opcji mapy;
- ☐ dodanie markera wraz z opisem.

W celu uszczegółowienia powyższych trzech funkcji skorzystaj z dowolnego darmowego systemu GIS.

Zadanie 8.4.

Na podstawie ćwiczenia 8.1 opracuj diagram sekwencji do naliczania opłat parkingowych poprzez wiadomość SMS dla kierowców, którzy:

- ☐ korzystają z systemu wysyłania wiadomości SMS podczas parkowania (wiadomość potwierdzająca wjazd na parking i opuszczenie go);
- ☐ zostali zarejestrowani w systemie i wnoszą co miesiąc opłatę za korzystanie z miejsc parkingowych.

W diagramie uwzględnij następujące klasyfikatory:

- ☐ obsługę płatności;
- ☐ parkometr;
- ☐ operatora telefonii komórkowej;

- ☐ system operatora kart płatniczych;
- ☐ kierowcę;
- ☐ inne.

Zadanie 8.5.

Przyroda we współczesnym uprzemysłowionym świecie ulega stałej degradacji. Jest to poważne wyzwanie ekologiczne, którego rozwiązaniu mogą sprzyjać technologie teleinformatyczne ICT. Ochrona wielkich kompleksów leśnych czy puszczy może być realizowana i pogłębiona dzięki systemom informatycznym — aplikacjom ochrony przyrody. Dokonać tego można na zasadzie użytkowania sensorowych sieci bezprzewodowych, które mogą monitorować m.in. pożary lasów i puszczy, przemieszczanie się zwierząt, wylewy rzek oraz inne ekstremalne zjawiska klimatyczne i przyrodnicze. Systemy te sterują również lotami samolotów bezzałogowych monitorujących obszary ochrony przyrody.

Przeanalizuj dostępne w Internecie opisy tego typu systemów, w szczególności w aspekcie inicjatywy CSIDC (*Computer Society International Design Competition*) opisaną na stronie instytutu IEEE (*Institute of Electrical and Electronics Engineers*) znajdującej się pod adresem <http://www.ieee.org>. Na podstawie zebranych informacji opracuj diagram sekwencji.

Zadanie 8.6.

Po dokładnym zapoznaniu się z ćwiczeniem 8.2 dotyczącym platformy wideokonferencji z podpisem elektronicznym przedstaw diagram sekwencji zawierania transakcji bankowej — dokonania przelewu dla kontrahenta, wykorzystując formularz przelewu zawierający:

- ☐ kwotę przelewu;
- ☐ numer konta bankowego;
- ☐ dane odbiorcy przelewu;
- ☐ tytuł przelewu.

Zaprojektuj wykorzystanie klas przechowujących informacje o koncie i kliencie. Należy uwzględnić możliwość próby dokonania przelewu przy niewystarczających środkach na koncie klienta dokonującego przelewu. Poszczególne pola wypełniane podczas definiowania przelewu powinny być walidowane, w szczególności sytuacja niezgodności rachunku odbiorcy z nazwą.

Zadanie 8.7.

Hurtownie danych są dobrze znaną i powszechnie wykorzystywaną technologią gromadzenia i przechowywania danych w biznesie. Istotą tych systemów było przechowywanie informacji typowo biznesowej. W ostatnich latach jednak powstają

znacznie większe zasoby danych, codziennie wzbogacane o informacje generowane przez ludzi oraz infrastrukturę sieciową i komputerową. Wielkość tych zasobów liczy się w zetabajtach. Firmy w coraz większym zakresie chcą je wykorzystywać dla potrzeb kampanii marketingowych. Używa się w nich odpowiednio skorelowanych danych, np. dotyczących klientów, które mogą pochodzić z różnych źródeł. Dane te przechowywane są w masowo zrównoleglonych bazach danych.

Przedstaw diagram sekwencji, w którym dane do przechowania w zrównoleglonej bazie danych przechowywane są po otrzymaniu komunikatów z następujących źródeł (klasyfikatorów):

- ❑ serwisy bankowe — transakcje bankowe;
- ❑ komputery — rejestry aktywności klientów;
- ❑ sklepy — zapisy transakcji handlowych;
- ❑ hotele — rejestracje gości hotelowych;
- ❑ ośrodki SPA i wypoczynkowe — zapisy pobytu gości;
- ❑ wyszukiwarki i przeglądarki internetowe — zapisy haseł wyszukiwanych przez użytkowników;
- ❑ serwisy poczty elektronicznej — metadane;
- ❑ serwisy społecznościowe — zapisy aktywności w sieciach.



Diagramy maszyn stanowych

9.1. Ćwiczenia

Ć W I C Z E N I E

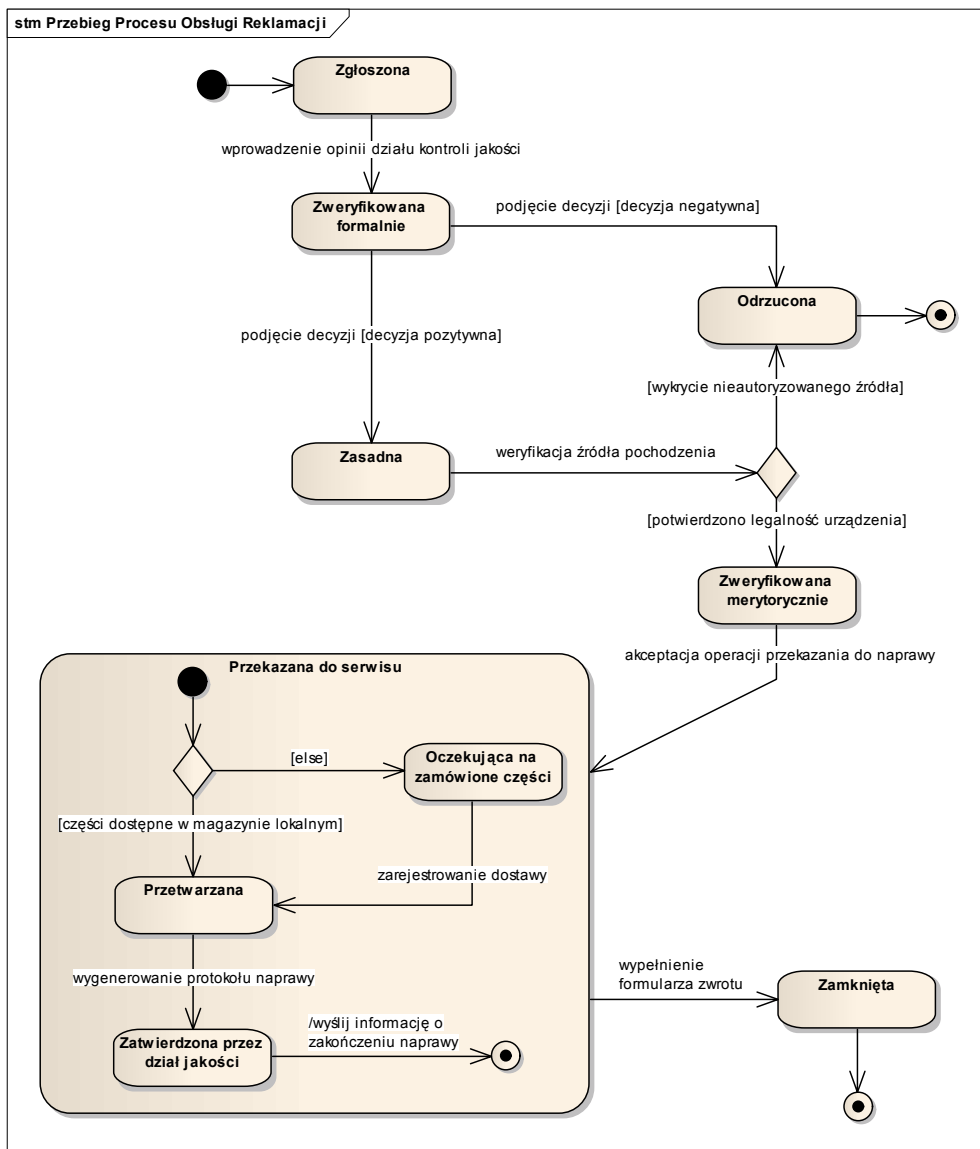
9.1 Przebieg procesu obsługi reklamacji w systemie zarządzania jakością

System zarządzania jakością w firmie dystrybucyjnej związany jest z wieloma aspektami jej funkcjonowania. Istotną częścią tego systemu jest przyjmowanie reklamacji. Przy przyjęciu reklamacji uruchamiany jest elektroniczny proces przyjmowania, ewidencjonowania i monitorowania jej przebiegu. W ramach tego procesu reklamacja może znajdować się w następującym stanie:

- ☐ *Zgłoszona;*
- ☐ *Zweryfikowana formalnie;*
- ☐ *Zasadna bądź Odrzucona;*
- ☐ *Zweryfikowana merytorycznie;*
- ☐ *Przekazana do serwisu;*
 - ☐ *Przetwarzana (może być poprzedzony stanem Oczekująca na zamówione części);*
 - ☐ *Zatwierdzona przez dział jakości;*
- ☐ *Zamknięta.*

Reklamacja na etapie obsługi serwisowej przyjmuje szereg podstawów będących konsekwencją przedstawionej klientowi wysokopoziomowej procedury obsługi reklamacji (por. rysunek 7.2). Kluczowe znaczenie w tym przypadku ma podstan *Zatwierdzona przez dział jakości* finalizujący dany etap obsługi reklamacji.

Należy opracować diagram maszyn stanowych opisujący przebieg procesu obsługi reklamacji (rysunek 9.1).



Rysunek 9.1. Diagram maszyn stanowych przebiegu procesu obsługi reklamacji

Ć W I C Z E N I E

9.2

Stany portu przełącznika w algorytmie drzewa opinającego

W celu zapobiegania powstawaniu pętli w sieciach opartych na przełącznikach wykorzystuje się algorytm drzewa opinającego. Podstawy tego algorytmu zostały zaprezentowane w ćwiczeniu 6.4 niniejszej książki. Pętle mogą prowadzić do zjawisk takich jak burze rozgłoszeniowe czy duplikaty ramek, które mogą uniemożliwić jakąkolwiek pracę w sieci komputerowej.

Istotnym elementem algorytmu STP jest kontrola portów w przełączniku, które w standardowej postaci protokołu drzewa opinającego przyjmują następujące stany:

- ❑ *Nieaktywność* (port wyłączony administracyjnie);
- ❑ *Blokowanie* (nie są wysyłane dane, odbierane są ramki BPDU (*Bridge Protocol Data Unit*); stan ten trwa 20 sekund po podłączeniu kabla do portu;
- ❑ *Nasłuchiwanie* (odbieranie ramek BPDU i budowanie topologii wolnej od pętli); stan trwający 15 sekund;
- ❑ *Uczenie się* (budowanie tablicy adresów MAC na przełączniku sieciowym); stan zajmujący przełącznikowi 15 sekund;
- ❑ *Przesyłanie* (podczas normalnej pracy — wysyłanie i odbieranie ramek z danymi).

Port może też zostać *Wyłączony administracyjnie* odpowiednim poleceniem. Port zaczyna zmieniać stany dopiero w momencie jego włączenia i podłączenia do niego kabla. Domyślnie na większości obecnie produkowanych przełączników wszystkie porty są włączone.

Należy opracować diagram maszyn stanowych prezentujący przejścia pomiędzy powyższymi stanami (rysunek 9.2).

**Wskazówki dydaktyczne**

Poszczególne stany zmieniają się po upływie określonego czasu, co zostało oznaczone na diagramie opisem *After* z podaną liczbą sekund. Stany można zapisywać z wykorzystaniem notacji klasycznej lub zakładkowej. W odróżnieniu od ćwiczenia 9.1 w niniejszym ćwiczeniu zastosowano notację zakładkową, gdzie nazwa stanu umieszczana jest w dodatkowym elemencie w górnej części kategorii modelowania.

Z uwagi na to, że elektroniczne protokoły ocen studentów oparte są na bazach danych zawierających wszystkie oceny studentów z poszczególnych przedmiotów, protokołom tym towarzyszy zazwyczaj rozbudowana funkcjonalność o charakterze statystycznym.

Procedura obsługi protokołów przez poszczególnych wykładowców warunkuje wyszczególnienie licznych stanów protokołu. Po udostępnieniu protokołu wykładowcom może on być edytowany w dowolnej liczbie podejść. Poszczególne modyfikacje zapisywane są w bazie danych. Jeśli wykładowca udostępni możliwość podglądu ocen tymczasowych, studenci mają możliwość podglądu bieżącego statusu w swoich profilach. Każdy wykładowca indywidualnie decyduje o zatwierdzeniu pierwszego podejścia. W tym momencie aktywowana jest funkcjonalność wprowadzania ocen drugiego podejścia, a edycja ocen pierwszego podejścia jest blokowana. Jednocześnie studenci uzyskują podgląd ostatecznych ocen. Drugie podejście również podlega zatwierdzeniu — odbywa się to na analogicznych zasadach.

Zamknięcie protokołu z punktu widzenia wykładowcy skutkuje automatycznym jego uwzględnieniem w portalu dziekanatu, gdzie protokół podlega weryfikacji formalnej. Zweryfikowane protokoły są automatycznie archiwizowane przed zakończeniem kolejnej sesji. Tym samym nie są one ujmowane na liście protokołów wykładowcy oraz są przenoszone do zakładki protokołów archiwalnych w dziekanacie.

Należy opracować diagram maszyn stanowych dla elektronicznego protokołu oceny studentów (rysunek 9.3).

Ć W I C Z E N I E

9.4 Cykl życia relacji z klientem

Jak wskazuje (Karaś, 2003), relacje klienta z firmą mogą trwać bardzo krótko i ograniczać się tylko do jednorazowej transakcji, ale mogą też trwać przez całe pokolenia, np. smakoszy pączków firmy Blikle czy koneserów biżuterii firmy Kruk.

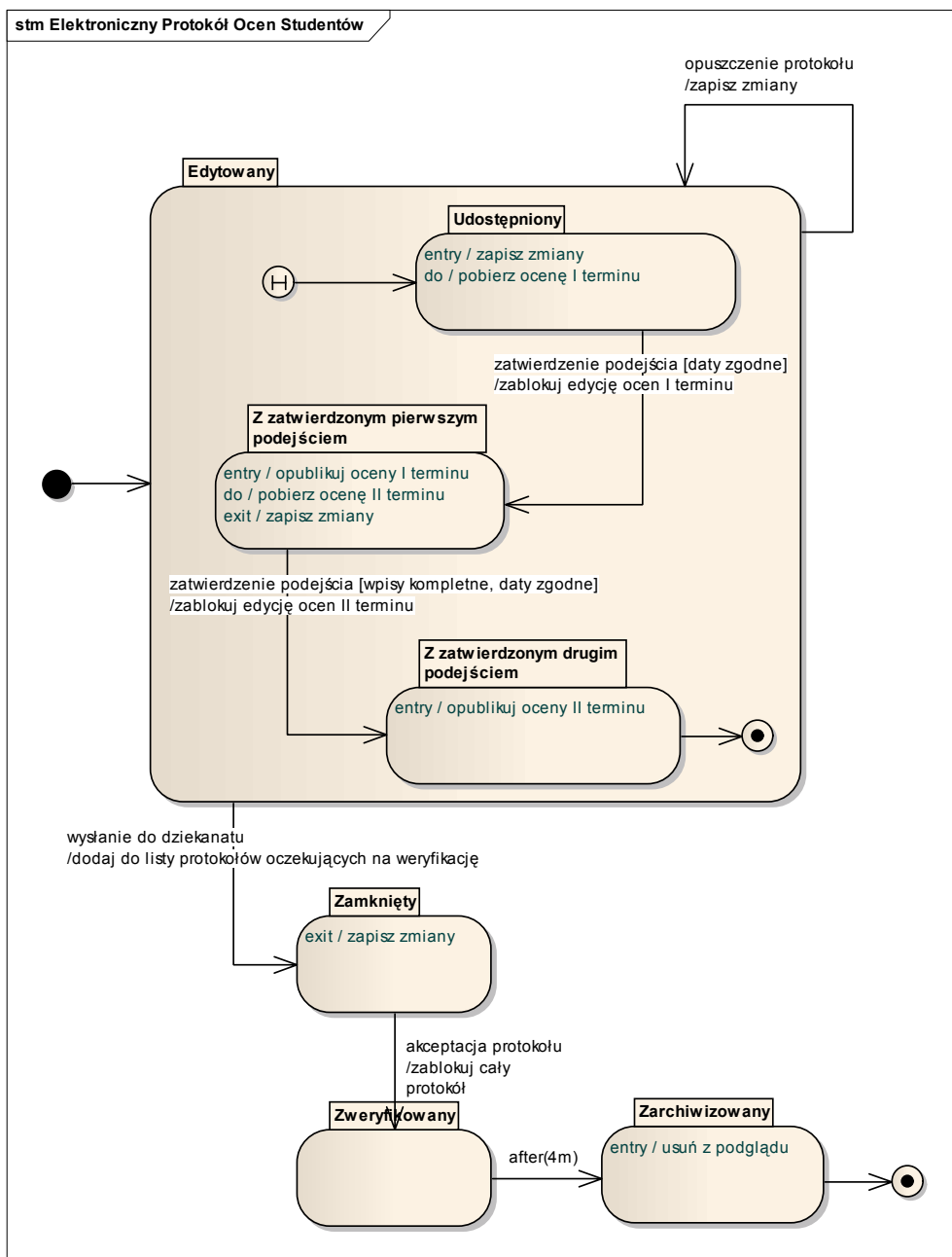
W cyklu życia klienta (ang. *Customer Life Cycle*) wyróżnić należy następujące fazy:

- ❑ promocja ukierunkowana na grupę docelową klientów:

Faza zdobywania potencjalnego klienta;

- ❑ segmentacja na kilka grup klientów z uwzględnieniem kryterium prognozowanych zysków, opracowanie zindywidualizowanej oferty dla poszczególnych grup klientów:

Faza próby przejęcia;



Rysunek 9.3. Diagram maszyn stanowych elektronicznego protokołu oceny studentów

- ❑ skierowanie do potencjalnych klientów zindywidualizowanej oferty obejmującej pakiety programów lojalnościowych:

Faza lojalności;

- ❑ zapewnienie satysfakcji klientów przez przyznanie im programów lojalnościowych, lepszą obsługę, promocje:

Faza realizacji programów lojalnościowych;

- ❑ odchodzenie klientów, spadek popytu, badania rynku i badanie przyczyn odejść:

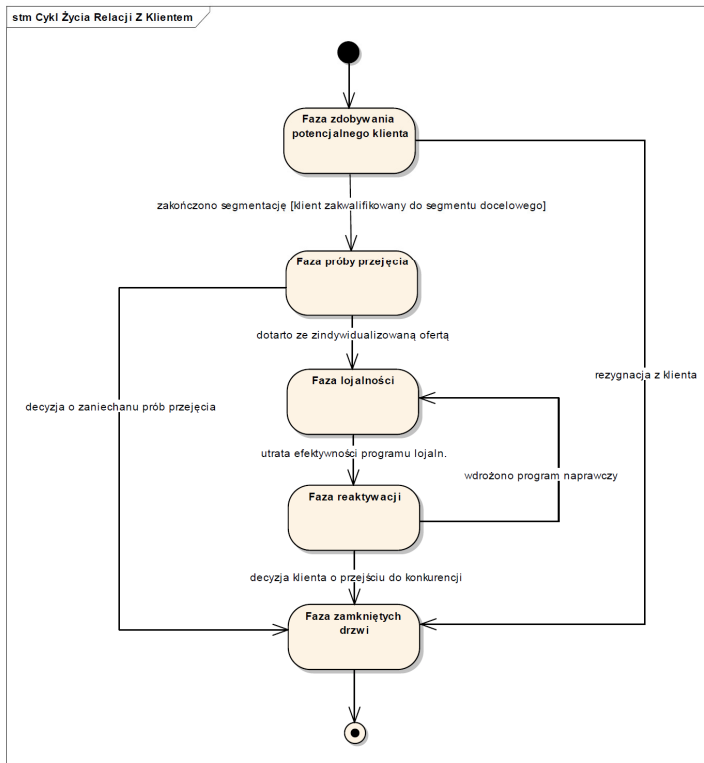
Faza reaktywacji;

- ❑ zastosowanie strategii odzyskiwania (ang. *win-back*):

Faza zamkniętych drzwi;

- ❑ część pozostaje, a znacząca część odchodzi.

Na podstawie powyższego opisu należy przygotować diagram maszyn stanowych (rysunek 9.4).



Rysunek 9.4. Diagram maszyn stanowych przedstawiający cykl życia relacji z klientem

Ć W I C Z E N I E

9.5

Szyfrowanie i deszyfrowanie w sieci VPN

Wraz z upowszechnieniem się Internetu w biznesie coraz większego znaczenia nabrało bezpieczeństwo systemów informatycznych działających w środowisku rozproszonym. Jedną z technologii poprawiających bezpieczeństwo danych przesyłanych pomiędzy odległymi systemami jest VPN (*Virtual Private Network*). Umożliwia ona przesyłanie zaszyfrowanych danych poprzez sieć komputerową, w tym również Internet.

W firmie produkcyjnej technologia VPN jest wykorzystywana do przekazania z oddziału terenowego do centrali firmy informacji o sprzedanych produktach. Proces przekazywania takiego komunikatu ze źródła do celu związany jest z następującymi stanami:

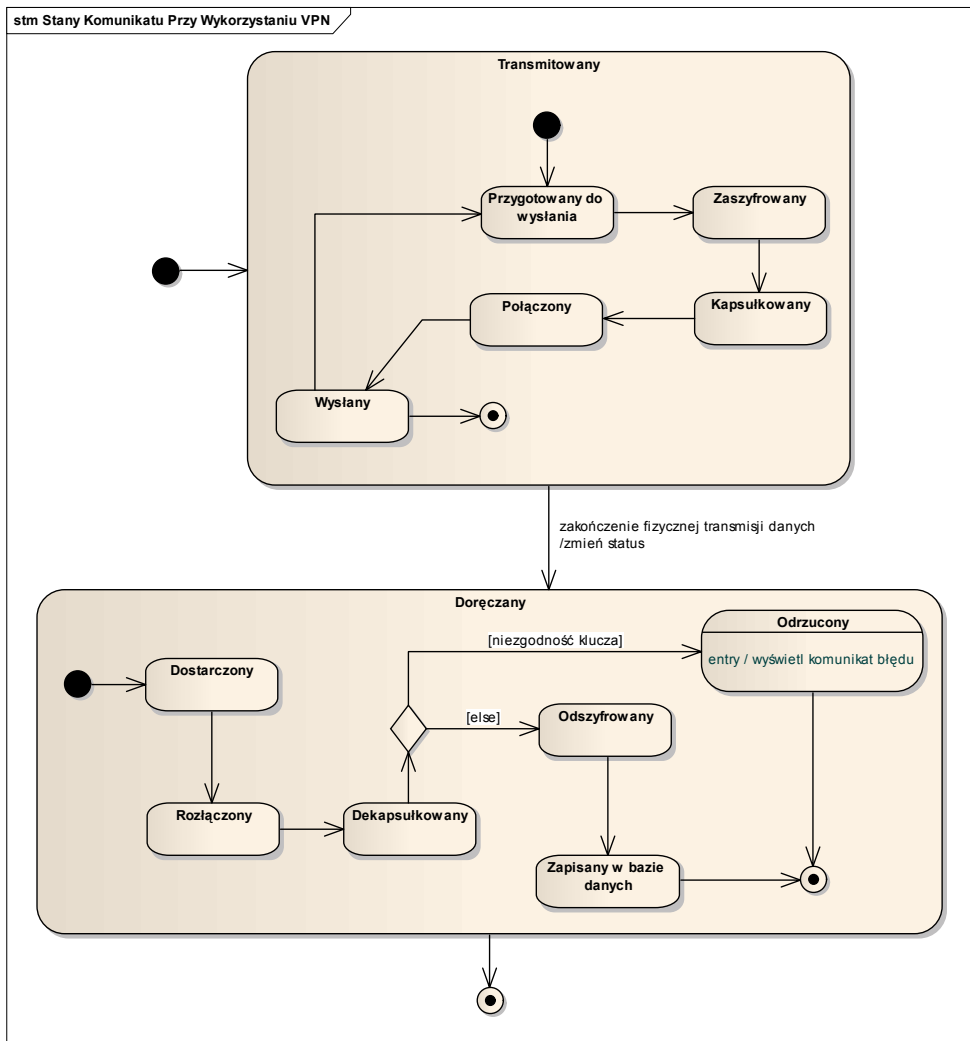
- ☐ *Przygotowany do wysłania;*
- ☐ *Zaszyfrowany;*
- ☐ *Kapsułkowany;*
- ☐ *Połączony;*
- ☐ *Wysłany;*
- ☐ *Dostarczony;*
- ☐ *Rozłączony;*
- ☐ *Dekapsułkowany;*
- ☐ *Odrzucony* (gdy klucz użyty do odszyfrowania nie jest zgodny z kluczem szyfrującym);
- ☐ *Odszyfrowany;*
- ☐ *Zapisany w bazie danych.*

Należy opracować diagram maszyny stanowej dla opisanego komunikatu przekazywanego w sieci VPN (rysunek 9.5).

**Wskazówki dydaktyczne**

W odróżnieniu od typowej maszyny stanowej większość przejść pomiędzy zidentyfikowanymi stanami ma charakter przejść automatycznych.





Rysunek 9.5. Diagram maszyn stanowych prezentujący stany komunikatu w sieciach opartych na technologii VPN

Ć W I C Z E N I E

9.6 Tryby pracy kamery internetowej

Wraz ze wzrostem przepustowości sieci komputerowych dużą popularność zyskały wideorozmowy prowadzone zarówno prywatnie z użyciem takich programów jak Skype czy NetMeeting, jak również w ramach wideokonferencji za pomocą systemów

Cisco Webex czy Adobe Connect. Te dwa wymienione systemy znakomicie wspomagają prowadzenie zajęć dydaktycznych, porad, konferencji dla użytkowników oddległych geograficznie.

W celu zapewnienia ciągłości strumienia wideo używane są kamery internetowe, których sterowniki pozwalają na automatyczny dobór jakości obrazu do parametrów łącza. Cały proces związany z transmisją wideo może prowadzić do przyjmowania przez kamerę następujących stanów:

- ☐ *Włączona;*
- ☐ *Ustawione parametry otoczenia:*
 - ☐ *Ustawiona jasność;*
 - ☐ *Ustawiony kontrast;*
- ☐ *Twarz znaleziona;*
- ☐ *Ustawiona ostrość;*
- ☐ *Biel zrównoważona;*
- ☐ *Jakość łącza rozpoznana:*
 - ☐ *Prędkość łącza rozpoznana;*
 - ☐ *Opóźnienia pakietów obliczone;*
- ☐ *Rozdzielczość kamery ustawiona;*
- ☐ *Transmisja danych rozpoczęta;*
- ☐ *Transmisja danych zakończona;*
- ☐ *Wyłączona.*

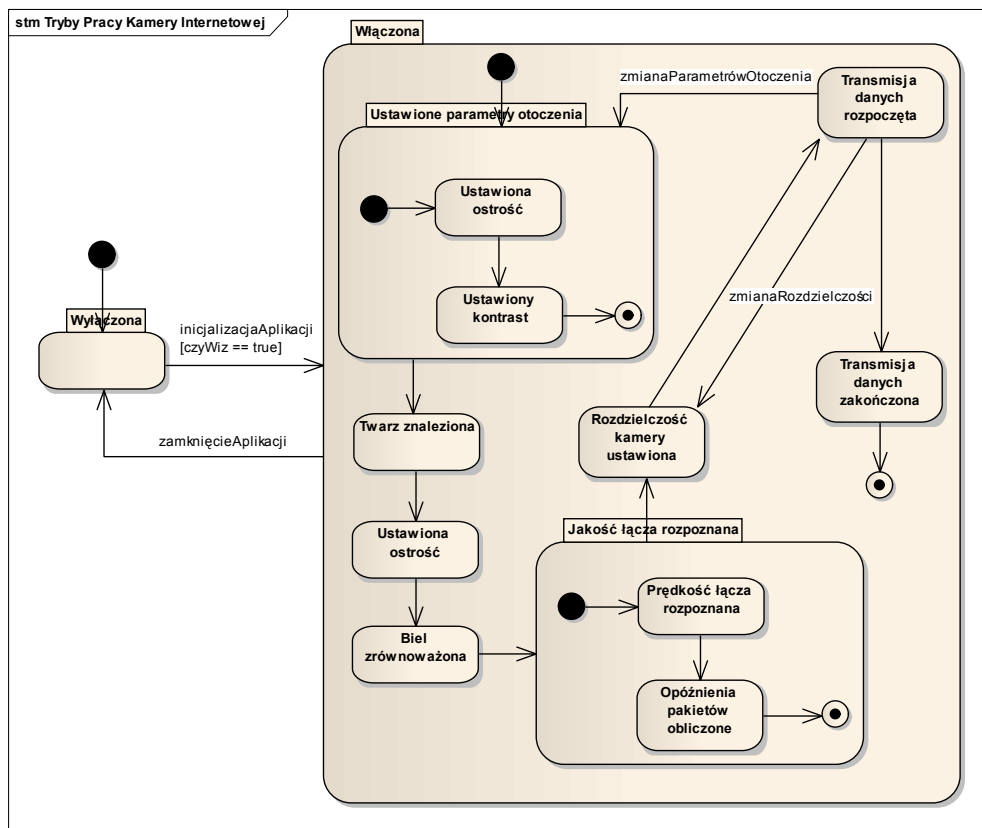
Należy opracować diagram maszyny stanowej przedstawiający wypisane powyżej stany kamery internetowej (rysunek 9.6).

Ć W I C Z E N I E

9.7

Serwis zarządzania zasobami w chmurze Dropbox

Serwis przechowywania, synchronizacji i udostępniania zasobów Dropbox pozwala na przechowywanie folderów oraz plików w chmurze. Po zainstalowaniu aplikacji Dropbox na danym komputerze w strukturze folderów umieszczany jest dedykowany folder. Dla użytkownika tworzony jest stosowny udział w chmurze (serwis internetowy Dropbox). Właściciel konta może współużytkować zasoby zdalne na wielu komputerach. Zamieszczenie zasobów w dedykowanym folderze na danym komputerze powoduje w takiej sytuacji synchronizację z zasobami w chmurze oraz na wszystkich innych komputerach z zainstalowaną aplikacją i przypisanym tym samym kontem użytkownika.



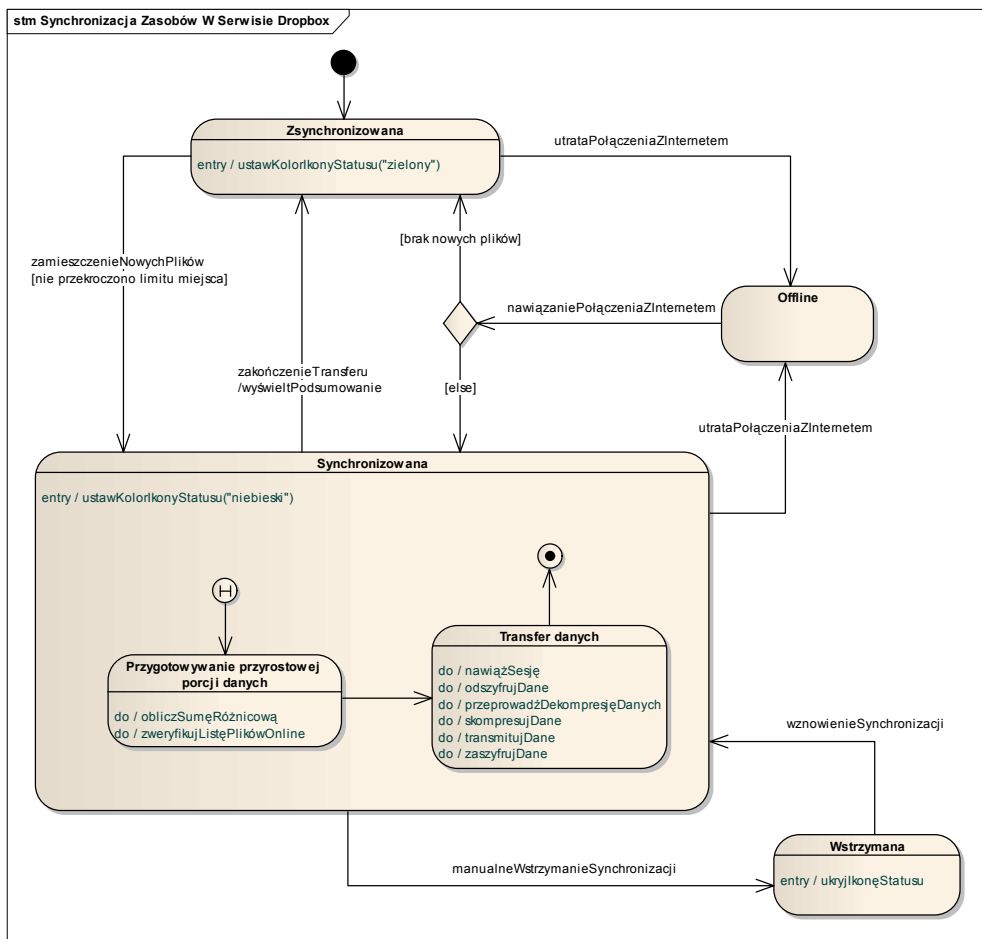
Rysunek 9.6. Diagram maszyn stanowych trybów pracy kamery internetowej

Z punktu widzenia synchronizacji zasobów skrzynka Dropbox może się znajdować w następujących stanach:

- ☐ *Offline*;
- ☐ *Synchronizowana* (zawierający dwa podstany: *Przygotowywanie przyrostowej porcji danych* oraz *Transfer danych*);
- ☐ *Zsynchronizowana*;
- ☐ *Wstrzymana*.

Proces synchronizowania wiąże się z *Przygotowywaniem przyrostowej porcji danych* oraz *Transferem danych*. W tym stanie wyświetlana jest ikona statusu w kolorze niebieskim. Po zakończeniu synchronizacji ikona statusu przyjmuje kolor zielony i na wszystkich synchronizowanych komputerach wyświetlane jest stosowne powiadomienie. Użytkownik ma możliwość wstrzymania procesu synchronizacji wedle własnego uznania.

Należy opracować diagram maszyny stanowej obrazujący proces synchronizacji zasobów w serwisie Dropbox (rysunek 9.7).



Rysunek 9.7. Diagram maszyn stanowych procesu synchronizacji zasobów w serwisie Dropbox

9.2. Zadania

Zadanie 9.1.

Przedstaw za pomocą diagramu maszyny stanowej stany urządzenia PDA. Urządzenia te wyposażone są w zarządzający minisystem, np. Windows CE, PalmOS, Google Android czy Symbian OS. Wśród podstawowych stanów, jakie mogą się pojawić podczas działania systemu, wymienić należy:

- ☐ otwarto książkę adresową;
- ☐ kalendarz;
- ☐ harmonogram;
- ☐ kalkulator;
- ☐ przelicznik walut;
- ☐ listę zadań;
- ☐ dostęp do Internetu;
- ☐ otwarto pocztę elektroniczną;
- ☐ odtwarzający MP3;
- ☐ odtwarzający filmy;
- ☐ odbierający radio;
- ☐ uruchamiający budzik;
- ☐ funkcjonujący jako stoper;
- ☐ robiący zdjęcia aparatem;
- ☐ zapisujący rozmowę na dyktafon.

Opracuj diagram maszyny stanowej uwzględniający wymienione powyżej stany.

Zadanie 9.2.

Serwis <http://www.blip.pl> umożliwia zakładanie i prowadzenie bloga. W ten sposób można umieszczać na bieżąco relacje, zdjęcia oraz filmy z aktualnych wydarzeń. W tym celu bloger wyposażony w tablet i kamerę przekazuje obraz i głos do serwisu. Użytkownicy serwisu mogą na bieżąco otrzymać informację o blogującym.

W ramach prowadzonego bloga zachodzą następujące stany dotyczące zamieszczanych treści:

- ☐ dodana;
- ☐ z załącznikiem (obraz, film);
- ☐ przesłana;
- ☐ opublikowana (dostępna dla pozostałych użytkowników);

- ☐ zarchiwizowana (przeniesiona do archiwum wiadomości);
- ☐ zablokowana (brak możliwości dodawania komentarzy);
- ☐ ukryta (niewidoczna dla pozostałych użytkowników);
- ☐ usunięta.

Opracuj diagram maszyny stanowej dla wymienionych powyżej stanów.

Zadanie 9.3.

Praca lekarzy może być dwa do trzech razy szybsza przy wykorzystaniu systemu rozpoznawania mowy i zamiany jej w tekst. Narzędzie to rozpoznaje mowę zarówno pacjentów, jak i lekarzy, między innymi w trakcie przeprowadzania wywiadu lekarskiego, i zamienia ją w tekst. System z powodzeniem może pełnić tę rolę tak w placówkach medycznych i diagnostycznych, jak i w przychodniach oraz szpitalach. Do użytkowania systemu niezbędny jest komputer wyposażony w słuchawki i mikrofon typu speech mike. W wyniku wykorzystania tego systemu po zakończeniu wywiadu lub wizyty otrzymuje się dokumentację medyczną.

Opracuj diagram maszyny stanowej użytkownika systemu rozpoznawania mowy i zamiany jej w tekst dla wykonania pełnej diagnostyki pacjenta: tomografii komputerowej, USG, radiologii klasycznej, rezonansu magnetycznego czy spirometrii.

Zadanie 9.4.

Coraz bardziej popularną i perspektywiczną formą świadczenia usług teleinformatycznych przez operatorów telekomunikacyjnych, w tym mobilnych, staje się odmiana outsourcingu, zwana *shared networks* — współdzielenie infrastruktury teleinformatycznej. Ma to również zastosowanie w przypadku operatorów wirtualnych, czyli nieposiadających własnej infrastruktury telekomunikacyjnej, a oferujących usługi teleinformatyczne.

Współdzielenie to obejmuje stacje bazowe, maszty, komponenty aktywne sieci. Dzięki temu operatorzy umowy *shared networks* mogą ograniczyć liczbę zakupywanych i eksploatowanych urządzeń własnych, a także zwiększyć efektywność urządzeń już użytkowanych. Mogą oni w ten sposób ustalić swoje priorytety w przedsięwzięciach operacyjnych oraz infrastrukturalnych. Podstawową rolę w tym rozwiązaniu outsourcingu teleinformatycznego odgrywają stacje bazowe, które wykonują następujące działania odpowiadające poszczególnym stanom:

- ☐ projektowanie i wdrażanie sieci;
- ☐ zakup urządzeń;
- ☐ zarządzanie pojemnością sieci;
- ☐ wymiana sieci;
- ☐ modernizacja sieci.

Dla powyższych stanów opracuj diagram maszyny stanowej.

Zadanie 9.5.

Fińscy informatycy utworzyli Osobisty Serwer Zdrowia jako usługę umożliwiającą monitorowanie zdrowia, pozwalającą na interpretację symptomów chorobowych oraz wskazanie diagnozy i terapii w danym przypadku. Oczywiście jest to informacja wstępna, która wymaga potwierdzenia poprzez wizyty i badanie lekarskie. Idea osobistego serwera zdrowia polega na wykorzystaniu słownika pojęć, ich definicji oraz wzajemnych zależności, czyli ontologii. Serwer integruje tego typu ontologię z innymi narzędziami i serwisami e-zdrowia, blogów czy wiki zawierających i aktualizujących informację o różnych chorobach i sposobach leczenia oraz związanych z nimi lekami. System oparty jest na technologii zarządzania wiedzą.

Na podstawie powyższego opisu przedstaw serwer jako maszynę stanową, w trakcie użytkowania znajdującą się w różnych stanach wykorzystania informacji i serwisów.

Zadanie 9.6.

Stały rozwój technologii teleinformatycznych sprawia, że coraz większą popularność zdobywają nowe wersje istniejących protokołów sieciowych. Wśród nich można wyróżnić protokół drzewa opinającego STP (*Spanning Tree Protocol*) opisywany w ćwiczeniach 6.4 i 9.2. Jednym ze standardów oficjalnie zatwierdzonych jest Rapid Spanning Tree, który został ujęty w standardzie IEEE 802.1w. Główną zaletą tego protokołu jest bardzo szybka zbieżność w stosunku do podstawowej wersji STP. Rozszerzenie to wymusiło również zmiany nazewnictwa stanów portów.

Bazując na dokumentacji do standardu Rapid Spanning Tree dostępnej na stronach instytutu IEEE (www.ieee.org), utwórz diagram maszyny stanowej opisujący przejścia pomiędzy nowymi stanami portów.

Zadanie 9.7.

Z uwagi na uciążliwość stosowania rozwiązań zabezpieczeń dostępu do sprzętu komputerowego opartych na kartach kodowych firma zdecydowała się wdrożyć nowe rozwiązanie uwierzytelniające użytkowników komputerów. W tym celu rozpoznano rynek w zakresie nowych rozwiązań w tym obszarze. W wyniku dokonanego rozpoznania ustalono, że coraz częściej uwierzytelnianie jest oparte na technikach takich jak skanowanie siatkówki oka, rozpoznawanie wzorców głosu, czy też wzorców linii papilarnych. Ze względu na koszty zdecydowano się wybrać to ostatnie rozwiązanie.

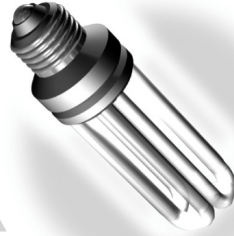
Podjęto decyzję o zastosowaniu skanera linii papilarnych podłączonego do portu USB komputera, który może przyjmować szereg stanów. Najczęściej występującymi metodami skanowania są: optyczna, pojemnościowa, radiowa, ciśnieniowa oraz termiczna. Ze względu na relatywnie wysoką odporność na próby przedstawienia spreparowanego wzorca zdecydowano się na zastosowanie metody radiowej. Dodatkowo oferenci przedstawiali rozwiązania uwzględniające czytnik, który wymaga przesunięcia palcem po skanerze, a nie tylko jego przytknięcia do urządzenia.

Zasada działania takiego urządzenia jest następująca:

- ❑ aby urządzenie działało, musi być przyłączone do komputera i wymieniać z nim informacje;
- ❑ informacja o odciskach palca jest przechowywana w bazie danych serwera, gdzie możliwe jest przechowanie do 750 różnych wzorców linii papilarnych;
- ❑ identyfikacja palca następuje po stronie komputera, a czytnik jest odpowiedzialny jedynie za pobranie odcisku i przesłanie tej informacji do komputera;
- ❑ aby uniknąć próby oszustwa, po przyłączeniu urządzenia do komputera ustanawiany jest klucz szyfrujący i uwierzytelniający transmisję pomiędzy skanerem linii papilarnych a komputerem przechowującym bazę danych o odciskach palców;
- ❑ po wymianie kluczy urządzenie jest cały czas aktywne i oczekuje na przytknięcie palca;
- ❑ pełen proces skanowania palca trwa 3 sekundy, w czasie których palec musi być przytknięty do skanera;
- ❑ po zeskanowaniu palca następuje przekazywanie zaszyfrowanej informacji o odcisku palca do bazy danych o liniach papilarnych znajdujących się w komputerze;
- ❑ w trakcie weryfikacji odcisku w komputerze urządzenie skanujące nie jest aktywne, co trwa 2 sekundy;
- ❑ po pozytywnej lub negatywnej weryfikacji następuje odblokowanie urządzenia skanującego i przejście w stan oczekiwania;
- ❑ urządzenie przestaje oczekiwać w momencie odłączenia od komputera lub wyłączenia go.

Zastosowanie tego rozwiązania powoduje, że system komputerowy może pracować w trzech trybach: uczenia się, identyfikacji oraz wyłączony (zwykle system jest wyłączany poza godzinami pracy firmy).

Opracuj dwa diagramy maszyny stanowej — jeden diagram ma prezentować stany czytnika linii papilarnych, drugi — stany systemu komputerowego. W przypadku diagramu pierwszego szczególną uwagę należy poświęcić specyfikacji przejść między stanami, a w drugim specyfikacji czynności wewnętrznych dla każdego stanu.



Diagramy pakietów

10.1. Ćwiczenia

Ć W I C Z E N I E

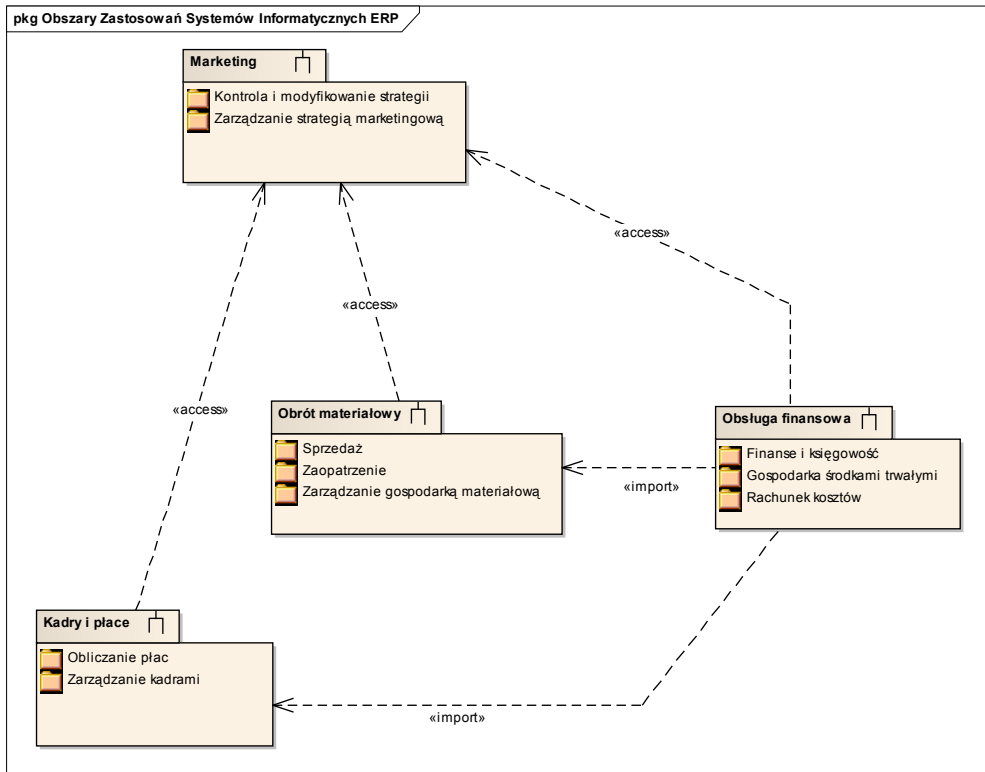
10.1 Obszary zastosowań systemów informatycznych ERP

Systemy ERP powszechnie wspomagają zarządzanie firmami i instytucjami. W zależności od komercyjnego systemu ERP obejmują one różny zakres funkcjonalny i procesowy. Przykładowo spektrum zastosowań systemu ERP dla firmy nieprodukcyjnej może być następujące (Wrycza, 2010):

- ☐ podsystem *Marketing*:
 - ☐ *Kontrola i modyfikowanie strategii (CRM)*;
 - ☐ *Zarządzanie strategią marketingową*;
- ☐ podsystem *Kadry i płace*:
 - ☐ *Obliczanie płac*;
 - ☐ *Zarządzanie kadrami*;
- ☐ podsystem *Obrót materiałowy*:
 - ☐ *Sprzedaż*;
 - ☐ *Zaopatrzenie*;
 - ☐ *Zarządzanie gospodarką materiałową*;
- ☐ podsystem *Obsługa finansowa*:
 - ☐ *Finanse i księgowość*;
 - ☐ *Gospodarka środkami trwałymi*;
 - ☐ *Rachunek kosztów*.

Każdy z podsystemów ma dostęp do modułów podsystemu *Marketing*. Moduły podsystemu *Obrotu materiałowego* oraz *Gospodarka środkami trwałymi* są importowane do modułu *Finansów i księgowości*. Dodatkowo podsystem *Kadrowo-płacowy* jest importowany do modułu *Finansów i księgowości*.

Należy opracować diagram pakietów prezentujący poszczególne obszary zastosowań systemów informatycznych ERP w postaci podsystemów (rysunek 10.1).

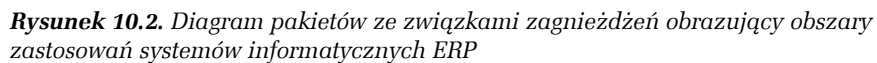


Rysunek 10.1. Diagram pakietów obszarów zastosowań systemów informatycznych ERP

Ć W I C Z E N I E

10.2 Zastosowanie systemów informatycznych ERP — zagnieźdżenia

Istnieje wiele sposobów prezentowania związków pomiędzy pakietami. Jednym z nich jest przedstawione w poprzednim ćwiczeniu osadzanie pakietów. Na podstawie ćwiczenia 10.1 należy opracować diagram pakietów, stosując w celu ich zobrazowania składowe związki zagnieźdżenia (rysunek 10.2).



**Wskazówki dydaktyczne**

Związki zagnieżdżeń pozwalają na zobrazowanie w sposób intuicyjny zależności pomiędzy poszczególnymi pakietami. Inaczej niż w ćwiczeniu 10.1 czynią one diagram bardziej rozbudowanym, przez co powinny być stosowane dla niewielkich systemów.

Ć W I C Z E N I E**10.3 Metody analizy rynku finansowego**

Początkujący inwestor został poproszony o zidentyfikowanie, pogrupowanie oraz prezentację graficzną podstawowych metod analizy rynku finansowego z wykorzystaniem języka UML. Z uwagi na użyteczność pakietów UML jako mechanizmu grupowania zdecydował się na zastosowanie tej techniki. Dogłębna analiza sekcji Edukacja na portalu bossa.pl pozwoliła mu na zapoznanie się z szeregiem technik szczegółowych z zakresu *Analizy technicznej*. Techniki te można uszeregować następująco:

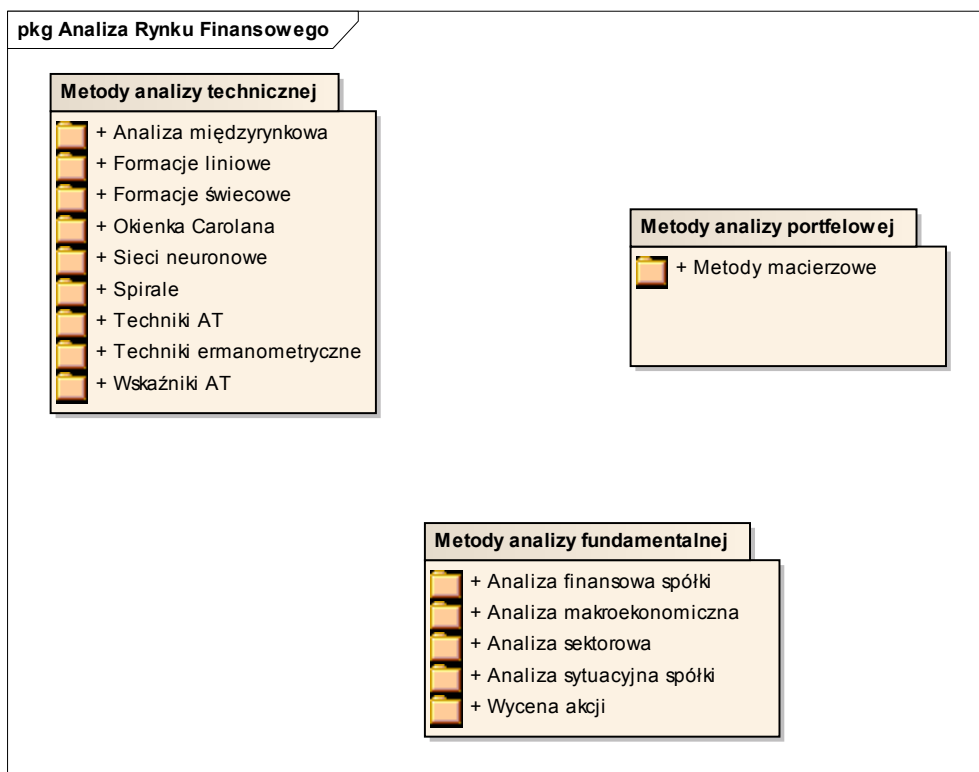
- ☐ *Analiza międzyrynkowa;*
- ☐ *Formacje techniczne, w tym Liniowe oraz Świecowe;*
- ☐ *Okienka Carolana;*
- ☐ *Sieci neuronowe;*
- ☐ *Spirale;*
- ☐ *Techniki AT;*
- ☐ *Formacje cenowe z zakresu teorii Ermanometrii;*
- ☐ *Wskaźniki AT.*

Z kolei lektura książki (Jajuga i Jajuga, 2011) zapoznała inwestora z metodami *Analizy fundamentalnej*; w praktyce zastosowanie analizy fundamentalnej przebiega w kilku etapach, którymi są:

- ☐ *Analiza finansowa spółki;*
- ☐ *Analiza makroekonomiczna;*
- ☐ *Analiza sektorowa;*
- ☐ *Analiza sytuacyjna spółki;*
- ☐ *Wycena akcji spółki.*

Na każdym z wymienionych etapów stosowane są zróżnicowane wskaźniki. Inwestor miał ponadto możliwość zapoznania się z wybranymi *Metodami macierzowymi* znajdującymi zastosowanie w *Analizie portfelowej*.

Na podstawie powyższego opisu należy opracować diagram pakietów dla analizy rynku finansowego (rysunek 10.3).



Rysunek 10.3. Diagram pakietów analizy rynku finansowego

Ć W I C Z E N I E

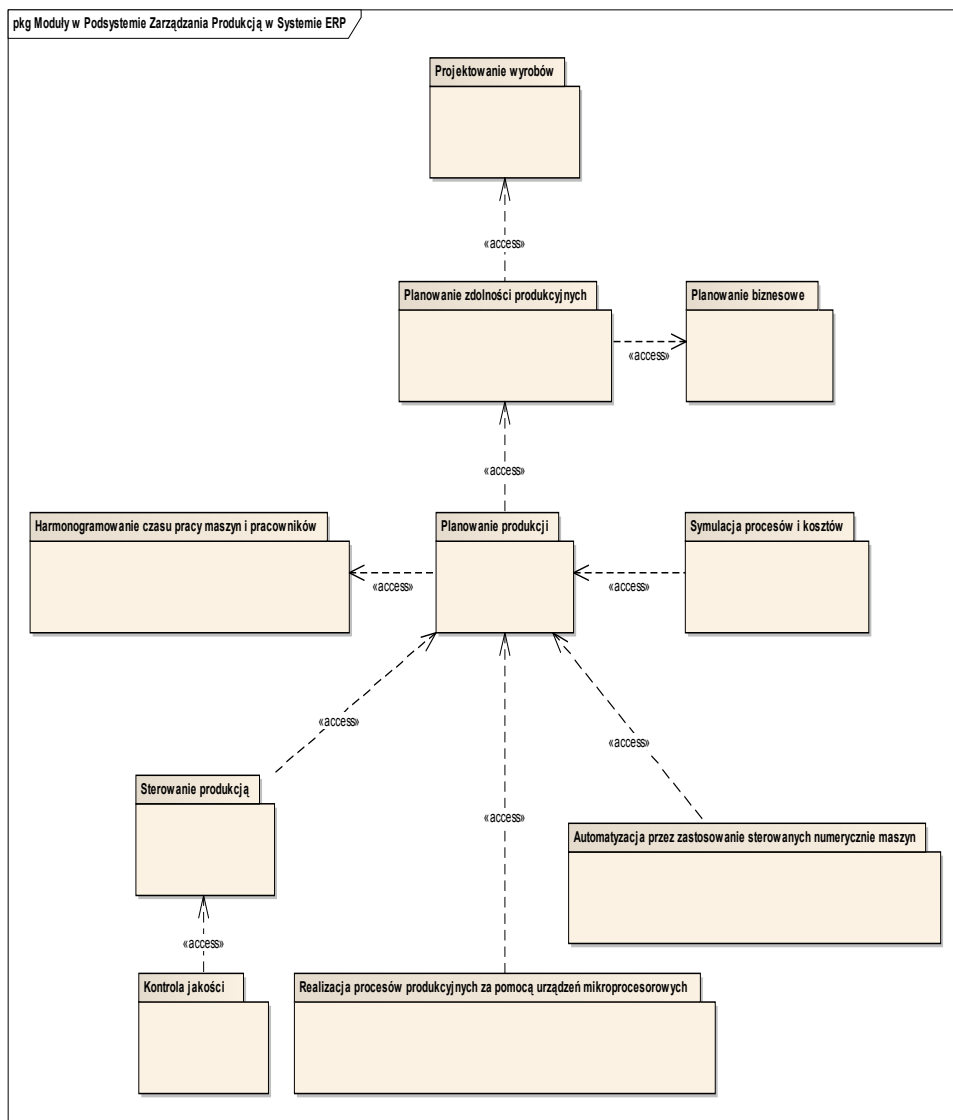
10.4 Moduły w podsystemie zarządzania produkcją w systemie ERP

Klasyczny system ERP zawiera następujące moduły (Knosala, 2003):

- ☐ Planowanie produkcji;
- ☐ Harmonogramowanie czasu pracy maszyn i pracowników;
- ☐ Planowanie zdolności produkcyjnych;
- ☐ Planowanie biznesowe (biznesplan);
- ☐ Projektowanie wyrobów;
- ☐ Sterowanie produkcją;
- ☐ Automatyzacja przez zastosowanie sterowanych numerycznie maszyn;
- ☐ Realizacja procesów produkcyjnych za pomocą urządzeń mikroprocesorowych (robotów);

- ❑ *Kontrola jakości;*
- ❑ *Symulacja procesów i kosztów.*

Na podstawie powyższej specyfikacji należy opracować diagram pakietów prezentujący moduły w podsystemie zarządzania produkcją w systemie ERP (rysunek 10.4).



Rysunek 10.4. Diagram pakietów dla modułów w podsystemie zarządzania produkcją w systemie ERP

Ć W I C Z E N I E

10.5 Serwis internetowy firmy kurierskiej

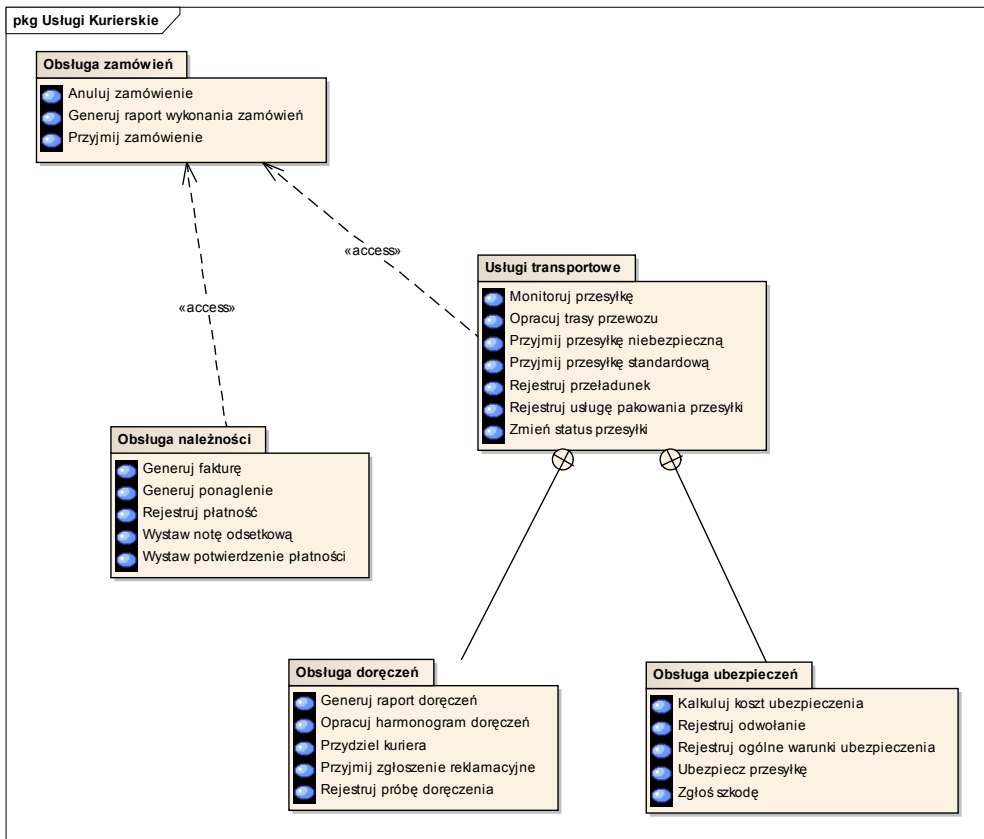
Firma oferująca usługi kurierskie realizuje każdą z tych usług na podstawie określonego modelu biznesowego udokumentowanego odpowiednimi diagramami. Zespół analityków systemowych delegowanych do zainteresowanej firmy opracował wstępną listę przypadków użycia projektowanego systemu. Alfabetyczna lista przypadków użycia przedstawionych do akceptacji na bieżącym etapie gromadzenia wymagań systemowych wygląda następująco:

- ☐ *Anuluj zamówienie;*
- ☐ *Generuj fakturę;*
- ☐ *Generuj ponaglenie;*
- ☐ *Generuj raport doręczeń;*
- ☐ *Generuj raport wykonania zamówień;*
- ☐ *Kalkuluj koszt ubezpieczenia;*
- ☐ *Monitoruj przesyłkę;*
- ☐ *Opracuj harmonogram doręczeń;*
- ☐ *Opracuj trasy przewozu;*
- ☐ *Przydziel kuriera;*
- ☐ *Przyjmij przesyłkę niebezpieczną;*
- ☐ *Przyjmij przesyłkę standardową;*
- ☐ *Przyjmij zamówienie;*
- ☐ *Przyjmij zgłoszenie reklamacyjne;*
- ☐ *Rejestruj odwołanie;*
- ☐ *Rejestruj ogólne warunki ubezpieczenia;*
- ☐ *Rejestruj płatność;*
- ☐ *Rejestruj próbę doręczenia;*
- ☐ *Rejestruj przeładunek;*
- ☐ *Rejestruj usługę pakowania przesyłki;*
- ☐ *Ubezpiecz przesyłkę;*
- ☐ *Wystaw notę odsetkową;*
- ☐ *Wystaw potwierdzenie płatności;*
- ☐ *Zgłoś szkodę;*
- ☐ *Zmień status przesyłki.*

Z uwagi na obszerność listy usług oferowanych przez system analitycy zdecydowali się na logiczne pogrupowanie ich w pakiety będące podstawą wyodrębnienia podsystemów. Tymi grupami są:

- ❑ *Obsługa zamówień;*
- ❑ *Usługi transportowe;*
- ❑ *Obsługa należności;*
- ❑ *Obsługa doręczeń;*
- ❑ *Obsługa ubezpieczeń.*

Należy opracować diagram pakietów (rysunek 10.5), uwzględniając wyszczególnione usługi kurierskie i grupując je według wymienionych powyżej podsystemów.



Rysunek 10.5. Diagram pakietów serwisu internetowego firmy kurierskiej

Ć W I C Z E N I E

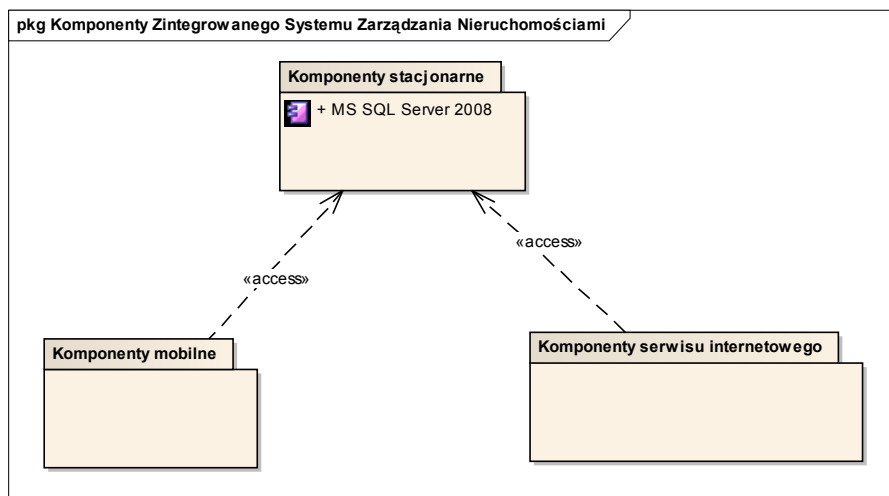
10.6 Biuro nieruchomości

Współczesne biura nieruchomości w wysokim stopniu wykorzystują technologie ICT dla wspomagania swej działalności biznesowej (Dega, 2011). Jednym z bardziej popularnych rozwiązań z zakresu oprogramowania tych biur są mobilne systemy zarządzania nieruchomościami. Standardowo takie mobilne oprogramowanie składa się z trzech zintegrowanych aplikacji (por. Bluebell, 2011):

- ❑ *Stacjonarnej*, umożliwiającej dostęp i zarządzanie danymi firmy;
- ❑ *Mobilnej*, poprzez którą pracownicy prowadzący swą działalność poza siedzibą firmy przesyłają informacje i mają dostęp do systemu mobilnego;
- ❑ *Serwisu internetowego*, poprzez który zasoby i usługi systemu informatycznego obsługi biura nieruchomości są w całości udostępniane pracownikom (intranet), a w niezbędnym zakresie (ekstranet) klientom firmy.

Każda z wymienionych aplikacji wykorzystuje szereg komponentów uszeregowanych w pakiety. Na danym etapie tworzenia dokumentacji systemu nie przewidziano wyszczególniania wszystkich komponentów. Z uwagi na to, że cały system oparto na jednej instancji bazy danych, a jej zawartość jest współużytkowana przez aplikację mobilną oraz serwis internetowy, zdecydowano się na jawne zaprezentowanie tego komponentu.

Należy opracować diagram pakietów przedstawiający komponenty opisanego zintegrowanego systemu zarządzania nieruchomościami (rysunek 10.6).



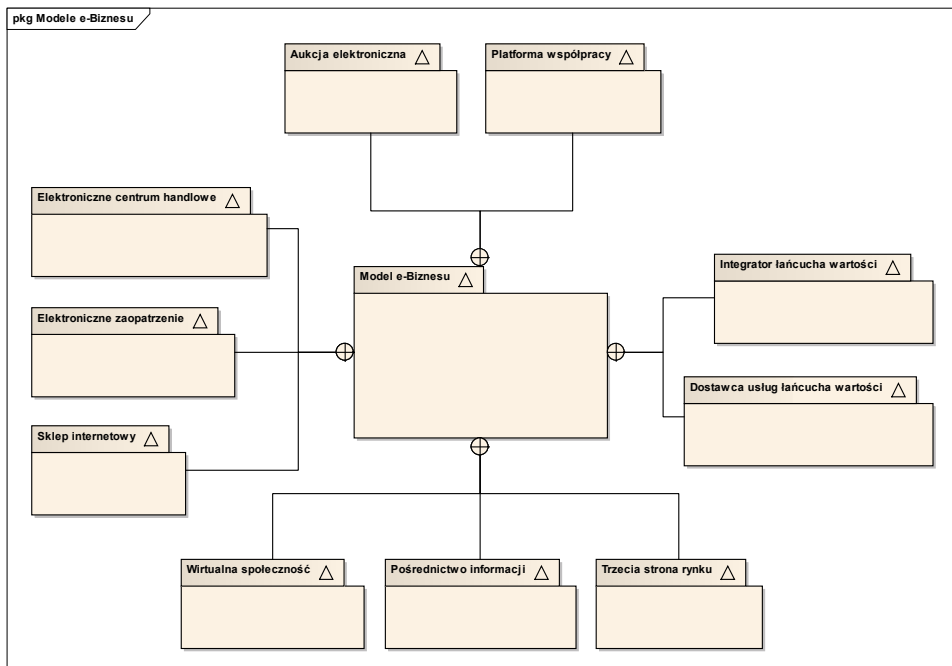
Rysunek 10.6. Diagram pakietów prezentujący komponenty zintegrowanego systemu zarządzania nieruchomościami

Ć W I C Z E N I E

10.7 Modele e-biznesu

Należy opracować diagram pakietów serwisu dydaktycznego (rysunek 10.7), który zawiera przykłady aplikacji e-biznesowych z zakresu następujących modeli e-biznesu według klasyfikacji Paula Timmersa (Timmers, 1998):

- ☐ Sklep internetowy (ang. *e-shop*);
- ☐ Elektroniczne zaopatrzenie (ang. *e-procurement*);
- ☐ Aukcja elektroniczna (ang. *e-auction*);
- ☐ Elektroniczne centrum handlowe (ang. *e-mall*);
- ☐ Trzecia strona rynku (ang. *3rd party marketplace*);
- ☐ Wirtualne społeczności (ang. *virtual communities*);
- ☐ Dostawcy usług łańcucha wartości (ang. *value-chain service providers*);
- ☐ Integratory łańcucha wartości (ang. *value-chain integrators*);
- ☐ Platformy współpracy (ang. *collaboration platforms*);
- ☐ Pośrednictwo informacji (ang. *information brokers*).



Rysunek 10.7. Diagram pakietów dla modeli e-biznesu

10.2. Zadania

Zadanie 10.1.

Na podstawie diagramu pakietów z ćwiczenia 10.1 przedstaw inne zależności pomiędzy wybranymi pakietami.

Zadanie 10.2.

W grudniu 2008 roku uruchomiono portal biblioteczny Europeana.eu (www.europeana.eu) umożliwiający dostęp do zbiorów zdigitalizowanych. Portal ten umożliwia dostęp do około 5 milionów dokumentów elektronicznych w postaci książek, czasopism, fotografii, filmów czy grafik. Największy wkład w zasoby tej biblioteki ma Francja — 47% książek, a Polska — 0,3% publikacji pochodzących z zasobów Federacji Bibliotek Cyfrowych i Biblioteki Narodowej.

Na podstawie danych zawartych w Portalu sporządź diagram pakietów różnych kategorii cyfrowych zasobów przechowywanych w serwisie bibliotecznym.

Przygotuj również diagram pakietów dla poniższych czterech rodzajów społeczności uzupełniających własne zbiory o tematyczne dokumenty elektroniczne:

- ☐ sztuka nowoczesna (fr. *Art Nouveau*);
- ☐ różnorodność ekosystemów (ang. *Biodiversity*);
- ☐ mapy i kartografia (ang. *Maps and Cartography*);
- ☐ średniowiecze i renesans (ang. *Medieval/Renaissance*).

Zadanie 10.3.

Systemy klasy CRM dzielą się na następujące rodzaje:

- ☐ operacyjny:
 - ☐ marketing;
 - ☐ sprzedaż;
 - ☐ serwis;
- ☐ analityczny;
- ☐ interakcyjny.

Na podstawie niniejszej klasyfikacji opracuj diagram pakietów, stosując związki zagnieżdżenia.

Zadanie 10.4.

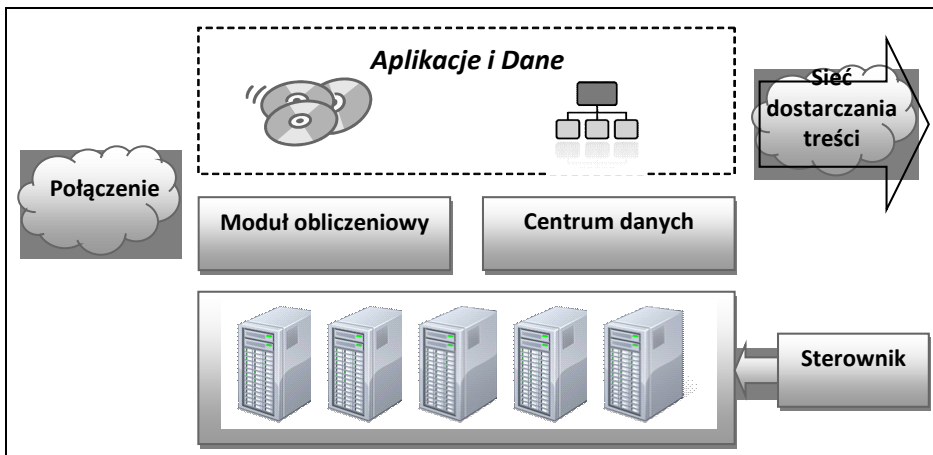
Aktualnie użytkowany jest cały szereg systemów informatycznych GIS, wśród których wyróżnić można między innymi ArcGIS czy darmowe, np. Chameleon oraz TerraView. Opracuj dla wybranego systemu GIS diagram pakietów w odniesieniu do przykładowych obszarów zastosowań, takich jak:

- ☐ geologia;
- ☐ gospodarka leśna;
- ☐ geografia;
- ☐ zarządzanie drogami;
- ☐ meteorologia;
- ☐ turystyka;
- ☐ zarządzanie nieruchomościami;
- ☐ żegluga rzeczna;
- ☐ żegluga morska;
- ☐ akweny;
- ☐ stacje benzynowe.

W opracowaniu diagramu pakietów wykorzystaj zależności merge, import.

Zadanie 10.5.

Korzystając z przedstawionej na rysunku 10.8 architektury systemu Microsoft Windows Azure przeznaczonego dla aplikacji Cloud Computing, opracuj diagram pakietów.



Rysunek 10.8. Modułowa architektura Microsoft Windows Azure

Źródło: Opracowanie własne na podstawie (Chappel, 2010)

Poszczególne komponenty systemu Windows Azure to:

- ❑ moduł obliczeniowy (ang. *Compute*): uruchamia w chmurze aplikacje, które funkcjonują pod systemem Windows Server;
- ❑ magazyn (ang. *Storage*): przechowuje dane binarne i ustrukturyzowane w chmurze;
- ❑ sterownik (ang. *Fabric Controller*): instaluje, zarządza i monitoruje aplikacje; sterownik jest odpowiedzialny również za aktualizację oprogramowania systemu przez platformę Azure;
- ❑ sieć udostępniania zasobów (ang. *Content Delivery Network*, CDN): przyspiesza dostęp do danych binarnych w zasobach Windows Azure poprzez utrzymywanie buforowanych kopii danych w różnych lokalizacjach na całym świecie;
- ❑ łączenie (ang. *Connect*): umożliwia tworzenie połączeń IP między komputerami a aplikacjami Windows Azure.

Zadanie 10.6.

Ochrona zdrowia wymaga nowoczesnego systemu informatycznego, który będzie wspomagał wszystkie jej podstawowe funkcje, służąc pacjentom, placówkom służby zdrowia, personelowi medycznemu i administracji państwowej. W tym celu ogłoszono konkurs na budowę Elektronicznej Platformy Gromadzenia, Analizy i Udostępniania Zasobów Cyfrowych o Zdarzeniach Medycznych (CSIOZ, 2009). System będzie zawierał następujące podsystemy:

- ❑ E-Recepty;
- ❑ Internetowe Konto Pacjenta;
- ❑ Portal;
- ❑ System Gromadzenia Danych Medycznych;
- ❑ System Obsługi Rejestrów;
- ❑ Hurtownię Danych;
- ❑ System Wykrywania Nadużyć;
- ❑ System Wspomagania Rozliczeń;
- ❑ System Weryfikacji;
- ❑ Szybnę Usług;
- ❑ System Administracji;
- ❑ Pomocniczą Bazę Rozszerzonych Danych Medycznych.

Opracuj diagram pakietów dla Elektronicznej Platformy Gromadzenia, Analizy i Udostępniania Zasobów Cyfrowych o Zdarzeniach Medycznych, który integruje te pakiety.

Zadanie 10.7.

Każda firma może korzystać z usług różnego rodzaju chmur:

- ☐ publicznej — infrastruktura, zasoby i usługi udostępnione są dla wszystkich;
- ☐ prywatnej, dzielącej się na:
 - ☐ dedykowaną — tylko dla danej firmy;
 - ☐ wspólnotową — zasoby użytkowane tylko przez zamknięty krąg firm;
- ☐ hybrydowej — zasoby mogą być udostępniane z różnych rodzajów chmur.

Przedstaw w postaci diagramu pakietów możliwości korzystania z zasobów i usług różnych chmur przez daną firmę.



Generowanie kodu źródłowego

11.1. Ćwiczenia

Ć W I C Z E N I E

11.1 Modelowanie w języku UML i jego profilach a wdrożenie systemu

W zakresie języka UML oferowane są narzędzia umożliwiające szybkie przejście z etapu projektowania do implementacji. W odróżnieniu od wielu innych języków do analizy i projektowania systemów informatycznych, takich jak projektowanie strukturalne czy ERD, UML pozwala na precyzyjny opis implementacji systemu. Oznacza to, że projektant narzuca w pewien sposób programistom nazewnictwo atrybutów czy metod (operacji). Wymaga to jednak od projektantów i analityków znajomości języka programowania, w którym nastąpi implementacja. Przykładowo w UML można modelować zarówno na poziomie konceptualnym (ćwiczenie 4.1), jak również implementacyjnym (ćwiczenie 4.3). Przy wykorzystaniu technik modelowania konceptualnego nie zawsze istnieje możliwość bezpośredniego przejścia z fazy projektowania do kodu źródłowego, gdyż większość istniejących języków programowania, takich jak Java, C++ czy Python, nie pozwala na używanie polskich znaków diakrytycznych czy spacji w nazwach atrybutów.

Istnieje wiele narzędzi umożliwiających bezpośrednie przejście z diagramu w języku UML do kodu źródłowego w wybranym języku programowania. Są to chociażby Enterprise Architect, MagicDraw UML, AgileJ Structure Views czy oferowany na licencji open source ArgoUML. W niniejszym rozdziale wykorzystano język programowania Java w celu pokazania sposobu bezpośredniego przejścia z projektowania do implementacji. Zastosowanym narzędziem jest Enterprise Architect firmy Sparx Systems.

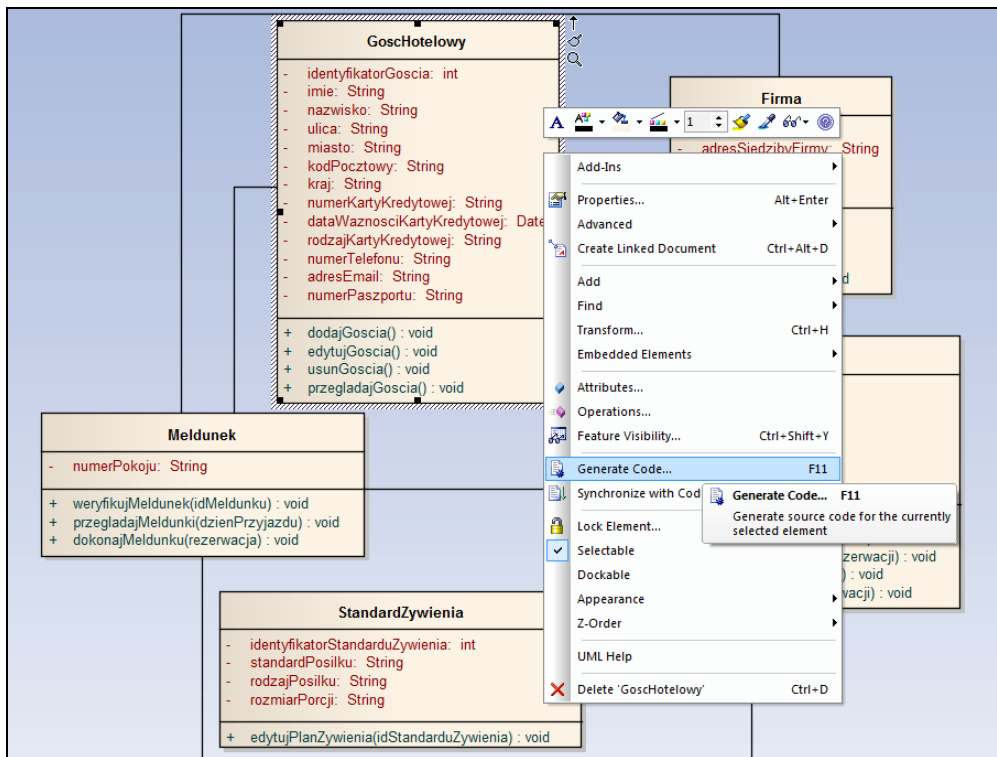


ĆWICZENIE

11.2 Diagramy klas

W celu zaprezentowania istoty przejścia z diagramu klas do kodu źródłowego posłużono się przykładem z punktu 4.7, czyli opisem systemu informacji hotelowych. Należy zauważyć, że przykład ten został wykonany zgodnie z podejściem implementacyjnym. Zarówno nazwy atrybutów, jak również metod nie posiadają polskich znaków oraz spacji. Ponadto typy danych są zgodne z językiem Java.

Aby wygenerować kod źródłowy w narzędziu Enterprise Architect, należy wybrać diagram, a następnie zaznaczyć jego kategorie modelowania, które mają zostać przeniesione do kodu źródłowego. Po wybraniu opcji generowania kodu źródłowego automatycznie powinna zostać utworzona lista klas. Opcja generowania kodu źródłowego została zaprezentowana na rysunku 11.1.



Rysunek 11.1. Diagram klas — generowanie kodu źródłowego w narzędziu Enterprise Architect 8

Utworzona klasa o nazwie `GoscHotelowy` posiada zestaw atrybutów zgodny z przedstawionymi w diagramie (listing 11.1).

Listing 11.1. Implementacja klasy *GoscHotelowy* w języku Java

```
1.  /**
   2.     @author Helion
   3.     @version 1.0
   4.     @created 24-lip-2011 10:58:52
   5.  */
   6.  public class GoscHotelowy {
   7.      private int identyfikatorGoscia;
   8.      private String imie;
   9.      private String nazwisko;
  10.     private String ulica;
  11.     private String miasto;
  12.     private String kodPocztowy;
  13.     private String kraj;
  14.     private String numerKartyKredytowej;
  15.     private Date dataWaznosciKartyKredytowej;
  16.     private String rodzajKartyKredytowej;
  17.     private String numerTelefonu;
  18.     private String adresEmail;
  19.     private String numerPaszportu;
  20.     public Meldunek m_Meldunek;
  21.     public Rezerwacja m_Rezerwacja;
  22.     public GoscHotelowy(){
  23.     }
  24.     public void finalize() throws Throwable {
  25.     }
  26.     public void dodajGoscia(){
  27.     }
  28.     public void edytujGoscia(){
  29.     }
  30.     public void usunGoscia(){
  31.     }
  32.     public void przegladajGoscia(){
  33.     }
  34. }
```

Należy zwrócić uwagę na to, że powyższe metody nie posiadają implementacji, tj. nie wiadomo, w jaki sposób gość hotelowy będzie dodawany czy usuwany. Istnieją dwie opcje rozwiązania tego problemu. Pierwsza to utworzenie diagramów czynności, które szczegółowo opiszą sposób obsługi tych zdarzeń (struktura algorytmów). Druga możliwość to pozostawienie programistom wyboru sposobu implementacji tych metod. W praktyce częściej stosuje się drugą opcję, gdyż projektanci odpowiedzialni są jedynie za pokazanie ogólnej koncepcji systemu. Na powyższym listingu można również zauważyć, że występuje tam odwołanie do dwóch innych klas z diagramu, jakimi są *Meldunek* oraz *Rezerwacja* (linia 17 i 18).

Przyjęto rozwiązanie, że dodawanie gości hotelowych będzie się odbywało poprzez pobranie danych z nowo otwartej formatki tekstowej i zapisanie tych danych w bazie danych MySQL. W związku z powyższym metoda *dodajGoscia* została zaimplementowana w sposób przedstawiony na listingu 11.2.

Listing 11.2. Implementacja w języku Java metody *dodajGoscia()* z klasy *GoscHotelowy*

```
1. public void dodajGoscia(){
2.     FormatkaGoscia fg=new FormatkaGoscia();
3.     fg.setVisible(true);
4.     this.imie=fg.imie.getText();
```

```
5. this.nazwisko=fg.nazwisko.getText();
6. ... (fragment kodu pominięty) ...
7. this.numerPaszportu=fg.numerPaszportu.getText();
8. Class.forName("com.mysql.jdbc.Driver").newInstance();
9. try {
10. Connection polaczenie =
    DriverManager.getConnection("jdbc:mysql://serwermysql/nazwaBazy",
    "uzytkownik", "haslo");
11. Statement skladnia = polaczenie.createStatement();
12. skladnia.addBatch("INSERT INTO GoschHotelowy (VALUES
    (\")+imie+\", \"+nazwisko+
13. ... (fragment kodu pominięty) ...
14. +\" \", \"+numerPaszportu+\" \")");
15. skladnia.executeBatch();
16. polaczenie.close();
17. }
18. catch(IOException e) {
19. JOptionPane.showMessageDialog(null, "Wystąpił błąd podczas dostępu do bazy
    danych.", "System rezerwacji hotelowych", JOptionPane.ERROR_MESSAGE);
20. }
21. catch(Exception e) {
22. JOptionPane.showMessageDialog(null, "Wystąpił nieznany błąd:
    "+e.getMessage(), "System rezerwacji
    hotelowych", JOptionPane.ERROR_MESSAGE);
23. }
24. }
```

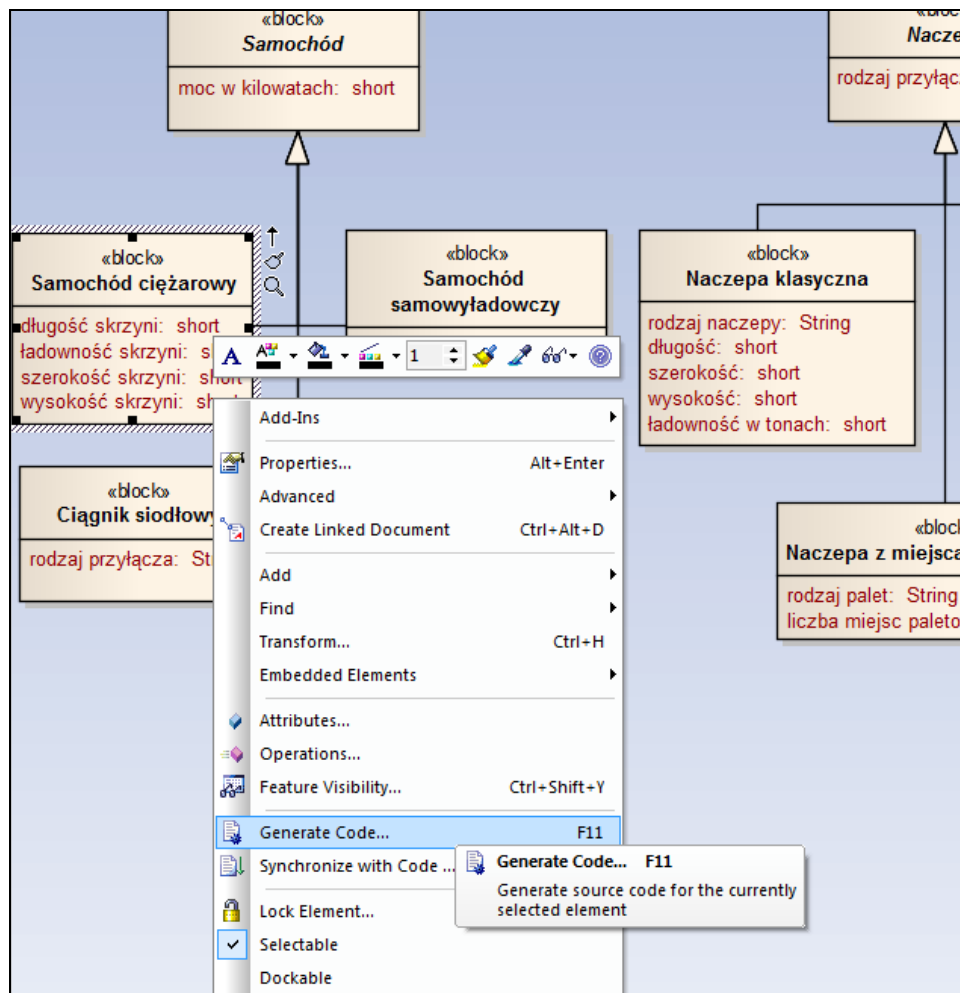
W powyższym wydruku listingu w liniach 6 oraz 13 pominięto analogiczny fragment kodu, którego celem jest zebranie z pól tekstowych znajdujących się na formacie ekranowej pozostałych informacji o nowo dodawanym gościu.

Ć W I C Z E N I E

11.3 Diagramy definiowania bloków

Podobnie jak diagramy klas języka UML, tak i diagramy definiowania bloków mogą zostać bezpośrednio odzwierciedlone w kodzie źródłowym wybranego języka programowania. W niniejszym podrozdziale wykorzystano diagram definiowania bloków z ćwiczenia 4.2, czyli systemu obsługi firmy transportowej. Należy zwrócić uwagę, że diagram ten został utworzony zgodnie z podejściem konceptualnym, w wyniku czego nazwy wartości zawierają polskie znaki diakrytyczne oraz spacje. W rezultacie programista będzie zmuszony do korekty wygenerowanego kodu źródłowego. Sposób wygenerowania kodu źródłowego w narzędziu Enterprise Architect 8 jest zbliżony do opisywanego w rozdziale 11.1 i ogranicza się do wybrania bloku oraz kliknięcia opcji generowania kodu źródłowego (rysunek 11.2).

Uzyskany kod nie skompiluje się w żadnym z powszechnie używanych języków programowania, gdyż z formalnego punktu widzenia nie jest poprawny (listing 11.3). Wygenerowany kod źródłowy wymaga zatem licznych poprawek dostosowujących go do poziomu implementacyjnego zgodnego z językiem Java.



Rysunek 11.2. Diagram definiowania bloków — generowanie kodu źródłowego w narzędziu Enterprise Architect 8

Listing 11.3. Implementacja w języku Java bloku *Samochód ciężarowy* — podejście konceptualne

```

1.  /**
    2.     @author Helion
    3.     @version 1.0
    4.     @created 24-lip-2011 12:02:17
    5.  */
    6.  public class Samochód ciężarowy extends Samochód {
    7.      private short długość skrzyni;
    8.      private short ładowność skrzyni;
    9.      private short szerokość skrzyni;
    10.     private short wysokość skrzyni;
    11.     public Samochód ciężarowy(){

```

```
9.  }
10. public void finalize() throws Throwable {
11.     super.finalize();
12. }
13. }//end Samochód ciężarowy
```

Korekty wymagają zarówno nazwy atrybutów, jak również nazwa klasy. Należy zwrócić uwagę, że klasa ta dziedziczy klasę *Samochód* (linia 3 kodu). Po dokonaniu korekty i rozszerzeniu kodu o metody klasa może się prezentować tak jak na listingu 11.4.

Listing 11.4. Implementacja w języku Java bloku *Samochód ciężarowy* — podejście wdrożeniowe

```
1.  /**
   * @author Helion
   * @version 1.1
   * @created 24-lip-2011 12:05:43
   */
2.
3.  public class SamochodCiezarowy extends Samochod {
4.      private short dlugoscSkrzyni;
5.      private short ladownoscSkrzyni;
6.      private short szerokoscSkrzyni;
7.      private short wysokoscSkrzyni;
8.      public SamochodCiezarowy(){
9.      }
10.     public short getDlugoscSkrzyni() {
11.         return this.dlugoscSkrzyni;
12.     }
13.     public void setDlugoscSkrzyni() {
14.         this.dlugoscSkrzyni=dlugoscSkrzyni;
15.     }
16.     public void finalize() throws Throwable {
17.         super.finalize();
18.     }
19. }//end Samochód ciężarowy
```

Zmiany dokonane w kodzie źródłowym dotyczą linii o numerach od 3 do 8. Główne zmiany dotyczą usunięcia polskich znaków oraz spacji. Dodatkowo klasa powinna zostać uzupełniona o metody pozwalające na dostęp do prywatnych atrybutów klasy. Przykłady tych metod zaprezentowano w liniach 10 – 15 (metody *getDlugoscSkrzyni* oraz *setDlugoscSkrzyni*).

Ć W I C Z E N I E

11.4 Diagramy wymagań

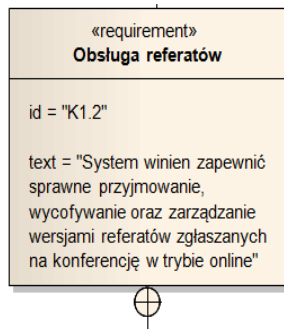
Pozostałe diagramy języka UML oraz jego profilu SysML pozwalają na doprecyzowanie potrzeb informacyjnych użytkowników. W ich przypadku nie ma możliwości bezpośredniego odzwierciedlenia diagramu w kodzie źródłowym, zatem programista ma za zadanie samodzielne wybranie sposobu implementacji poszczególnych wymagań, procesów czy przypadków użycia.

Dla zilustrowania sposobu przejścia z diagramu wymagań do kodu źródłowego posłużono się ćwiczeniem 2.1, czyli przypadkiem systemu obsługi międzynarodowej konferencji. W zależności od wyboru programistów dane wymaganie można zaim-

plementować jako klasę, metodę czy nawet bibliotekę klas. W niniejszym przypadku postanowiono zaimplementować wymaganie *Obsługa referatów* jako klasę w języku Java (rysunek 11.3).

Rysunek 11.3.

Wymaganie
Obsługa
referatów



Poszczególne składowe wymagania *Obsługa referatów* mogą zostać zaprezentowane jako metody. Wówczas wzorzec takiej klasy będzie zgodny z zaprezentowanym na listingu 11.5. Trzeba jednak pamiętać, że w kodzie źródłowym jesteśmy w stanie zrealizować wymagania funkcjonalne. Pozostałe wymagania, wśród których najczęściej występują wymagania pozafunkcjonalne, mogą zostać spełnione w inny sposób.

Listing 11.5. Przykład implementacji wymagania *Obsługa referatów* w języku Java

```

1.  /**
    *   Implementacja wymagania ObsługaReferatow.
    *
    *   @author Helion
    *   @version 0.1
    */
2.  */
3.  public class ObsługaReferatow
4.  {
5.      private Referat referat;
6.      public ObsługaReferatow() {
7.      }
8.      public void przyjmijReferat(Referat referat) {
9.      }
10.     public void wycofajReferat(Referat referat) {
11.     }
12.     public void zarządzajWersjaReferatu(Referat referat) {
13.     }
14. }
  
```

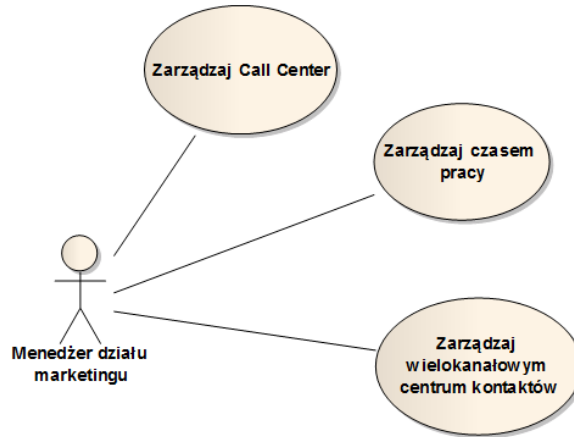
Ć W I C Z E N I E

11.5 Diagramy przypadków użycia

Podobnie jak opisywane w poprzednim punkcie diagramy wymagań, tak i diagramy przypadków użycia nie mogą zostać bezpośrednio przeniesione do kodu źródłowego. Poszczególne kategorie modelowania diagramu mogą zostać przeniesione jako klasy

(np. aktorzy systemu) lub metody (np. przypadki użycia). Przykład takiej implementacji w języku Java na bazie ćwiczenia 3.4 (funkcjonalność systemu CRM) zaprezentowano na rysunku 11.4 oraz listingu 11.6.

Rysunek 11.4.
Przypadki użycia
systemu CRM
przez aktora
Menedżer działu
marketingu



Listing 11.6 prezentujący kod źródłowy klasy napisanej w języku Java jest tylko jedną z wielu możliwości realizacji koncepcji zaprezentowanej na rysunku 11.4. Istotą diagramu przypadków użycia, podobnie jak diagramu wymagań systemowych, jest brak narzucania programiście sposobu rozwiązania postawionego problemu.

Listing 11.6. Przykład implementacji przypadków użycia dla Menedżera działu marketingu w języku Java

```

1.  /**
2.   * Implementacja przypadków użycia dla aktora MenedzerDzialuMarketingu.
3.   *
4.   * @author Helion
5.   * @version 0.1
6.   */
7.  public class MenedzerDzialuMarketingu
8.  {
9.      CallCenter cc;
10.     CzasPracy cp;
11.     CentrumKontaktow ck;
12.     public MenedzerDzialuMarketingu(CallCenter cc, CzasPracy cp,
13.                                     CentrumKontaktow ck)
14.     {
15.         this.cc=cc;
16.         this.cp=cp;
17.         this.ck=ck;
18.     }
19.     public void zarzadzajCallCenter(CallCenter cc) { ... }
20.     public void zarzadzajCzasemPracy(CzasPracy cp) { ... }
21.     public void zarzadzajCentrumKontaktow(CentrumKontaktow ck) { ... }
22. }
  
```

Na listingu 11.6 linie kodu 18 – 20 wymagają uzupełnienia pozwalającego na realizację potrzeb informacyjnych Menedżera działu marketingu.



11.2. Zadania

Zadanie 11.1.

Na podstawie ćwiczenia 4.3 zaimplementuj w postaci kodu źródłowego klasę o nazwie `Student` w języku Java lub C#. Postaraj się napisać implementację operacji o nazwie `infoOStudencie`, wyświetlając wszystkie atrybuty `Studenta`, np. za pomocą metody `System.out.println`.

Zadanie 11.2.

Zaproponuj wzorzec klasy w języku Java lub C# dla diagramu wymagań systemowych zaprezentowanego w ćwiczeniu 2.5 oceny poziomu szkolenia.

Zadanie 11.3.

Opracuj do diagramu przypadków użycia systemu monitorowania ruchu w mieście przykładowy zestaw klas i metod w języku Java lub C#.

Zadanie 11.4.

Na podstawie diagramu definiowania bloków opracowanego dla formularza dokumentującego historię kontaktów z klientem w systemie CRM (ćwiczenie 4.4) przygotuj dokładną specyfikację klasy w postaci kodu źródłowego zapisanego w języku Java lub C#.

Zadanie 11.5.

Ćwiczenie 4.1 opisujący system zarządzania terminalem kontenerowym przenieś w postaci zestawu klas do kodu źródłowego, a następnie zaproponuj atrybuty i metody.

Zadanie 11.6.

Dla ćwiczenia 4.6 opisującego kancelarię prawniczą utwórz zestaw klas w języku Java lub C#, uwzględniając dziedziczenie i abstrakcyjność klas.

Zadanie 11.7.

Wykorzystując diagram z ćwiczenia 4.2, czyli systemu obsługi firmy transportowej, przedstaw w postaci hierarchii klas zapisanej w kodzie źródłowym języka Java lub C# następujące klasy: `Pojazd`, `Naczepa`, `Naczepa chłodnia`. W poszczególnych klasach powinny się znaleźć wymagane atrybuty oraz metody.



Bibliografia

1. Bluebell (2011); *NetLokum — aplikacja i strona www dla agencji nieruchomości*, http://www.blue-bell.pl/aplikacje_biur_nieruchomosci.html, stan na dzień 27.07.2011.
2. Booch G., Rumbaugh J., Jacobson I. (1997); *Unified Modeling Language, Version 1.0*, Rational Software Corporation.
3. Brandt A. K. (2010); *Scrum — prostszy sposób na projekty*, „Computerworld”, nr 18 – 19.
4. Centrum Systemów Informatycznych Ochrony Zdrowia (2009); *Specyfikacja istotnych warunków zamówienia*, <http://www.csioz.gov.pl/file.php?s=ej80NA==>, stan na dzień 27.07.2011.
5. Chappell D. (2010); *Introducing Windows Azure*, <http://www.microsoft.com/windowsazure/Whitepapers/IntroducingWindowsAzure/>, stan na dzień 27.07.2011.
6. Dega (2011); *Oferta dla biur nieruchomości*, <http://www.dega.com.pl/nieruchomosci.asp>, stan na dzień 27.07.2011.
7. Energa-Operator SA (2011); *Taryfa Energa-Operator*, <http://www.energa.pl/object.php/act/sho/oid/fdb671ba25a1065b0d1e5496d41e2550>, stan na dzień 26.07.2011.
8. <http://www.allegro.pl>, stan na dzień 27.07.2011.
9. <http://www.amazon.com>, stan na dzień 27.07.2011.
10. <http://www.blip.pl>, stan na dzień 27.07.2011.
11. <http://www.dropbox.com>, stan na dzień 27.07.2011.
12. <http://www.easychair.org>, stan na dzień 26.07.2011.
13. <http://www.europeana.eu>, stan na dzień 27.07.2011.
14. <http://www.ieee.org>, stan na dzień 27.07.2011.
15. <http://www.wbshop.com>, stan na dzień 27.07.2011.

16. International Organization for Standardization (2004); *Unified Modeling Language Specification. Version 1.4.2*, <http://www.omg.org/spec/UML/#ISO19501>, stan na dzień 26.07.2011.
17. ITIL Training (2010); *ITIL CSI — The 7 Step Improvement Process*, <http://itilcontinualserviceimprovement.blogspot.com/2008/09/itil-csi-7-step-improvement-process.html>, stan na dzień 27.07.2011.
18. Jajuga K., Jajuga T. (2011); *Inwestycje. Instrumenty finansowe, aktywa niefinansowe, ryzyko finansowe, inżynieria finansowa*, PWN, Warszawa.
19. Kang J. M., Yoo T., Kim H. C. (2006); *A Wrist-Worn Integrated Health Monitoring Instrument with a Tele-Reporting Device for Telemedicine and Telecare*, „IEEE Transactions on Instrumentation and Measurement”, no. 5.
20. Karaś B. (2003); *Cykl życia klienta*, <http://www.marketing-news.pl/theme.php?art=63>, stan na dzień 27.07.2011.
21. Knosala R. (2003, red.); *Komputerowo wspomagane zarządzanie*, WNT, Warszawa.
22. Kolawa A. (2010); *Skokowy wzrost wydajności*, Wydawnictwo CDW, Cieszyn.
23. Massel A. (2001); *Elementy inżynierii ruchu*, [w:] Bałuch H. (red.); *Budownictwo komunikacyjne*, WAT, Warszawa.
24. Massel A. (2010); *Projektowanie linii i stacji kolejowych*, PKP Polskie Linie Kolejowe, Warszawa.
25. Microsoft Corporation (2009); *How Far Will You Take Virtual*, <http://www.microsoft.com/virtualization/>, stan na dzień 26.07.2011.
26. Microsoft Corporation (2009a); *Private Cloud Solutions. Bringing You the Full Power of the Cloud*, <http://www.microsoft.com/privatecloud/>, stan na dzień 27.07.2011.
27. Milgin S. (2010); *Na kłopoty ze zmianą wymagań*, „Computerworld”, nr 31 – 32.
28. Nitsch-Cisak H., Jurczak H. (1983); *Analiza funkcjonowania służby komorowej w ujęciu symulacyjnym*, Wydawnictwo Politechniki Wrocławskiej, Wrocław.
29. Object Management Group (2000); *OMG Unified Modeling Language Specification. Version 1.3*, <http://www.omg.org/spec/UML/1.3>, stan na dzień 26.07.2011.
30. Object Management Group (2001); *OMG Unified Modeling Language Specification. Version 1.4*, <http://www.omg.org/spec/UML/1.4>, stan na dzień 26.07.2011.
31. Object Management Group (2003); *OMG Unified Modeling Language Specification. Version 1.5*, <http://www.omg.org/spec/UML/1.5>, stan na dzień 26.07.2011.
32. Object Management Group (2005); *Unified Modeling Language: Superstructure. Version 2.0*, <http://www.omg.org/spec/UML/2.0>, stan na dzień 26.07.2011.

33. Object Management Group (2007); *OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2*, <http://www.omg.org/spec/UML/2.1.2>, stan na dzień 26.07.2011.
34. Object Management Group (2007a); *Unified Modeling Language: Superstructure. Version 2.1.1*, <http://www.omg.org/spec/UML/2.1.1>, stan na dzień 26.07.2011.
35. Object Management Group (2007b); *OMG Systems Modeling Language (OMG SysML™), V1.0*, <http://www.omg.org/spec/SysML/1.0/>, stan na dzień 26.07.2011.
36. Object Management Group (2008); *OMG Systems Modeling Language (OMG SysML™). Version 1.1*, <http://www.omg.org/spec/SysML/1.1/>, stan na dzień 26.07.2011.
37. Object Management Group (2009); *OMG Unified Modeling Language™ (OMG UML), Superstructure. Version 2.2*, <http://www.omg.org/spec/UML/2.2>, stan na dzień 26.07.2011.
38. Object Management Group (2010); *OMG Unified Modeling Language™ (OMG UML), Superstructure. Version 2.3*, <http://www.omg.org/spec/UML/2.3>, stan na dzień 26.07.2011.
39. Object Management Group (2010a); *OMG Systems Modeling Language (OMG SysML™). Version 1.2*, <http://www.omg.org/spec/SysML/1.2/>, stan na dzień 26.07.2011.
40. Object Management Group (2011); *OMG Unified Modeling Language™ (OMG UML), Superstructure. Version 2.4*, <http://www.omg.org/spec/UML/2.4>, stan na dzień 26.07.2011.
41. Object Management Group (2011a); *Unified Modeling Language™ (UML®)*, <http://www.omg.org/spec/UML/>, stan na dzień 26.07.2011.
42. Object Management Group (2011b); *Catalog of UML Profile Specifications*, http://www.omg.org/technology/documents/profile_catalog.htm, stan na dzień 26.07.2011.
43. Object Management Group (2011c); *OMG System Modeling Language (SysML™)*, <http://www.omg.org/spec/SysML/>, stan na dzień 26.07.2011.
44. Ramsey D. (2011); *Using Computers and Sensors to Curb Electricity Use in Buildings*, <http://ucsdnews.ucsd.edu/newsrel/science/04-13CurbElectricity.asp>, stan na dzień 27.07.2011.
45. Szling Z., Winter J. (1998); *Drogi wodne śródlądowe*, Wydawnictwo Politechniki Wrocławskiej, Wrocław.
46. Timmers P. (1998); *Business Models for Electronic Markets*, European Commission, Directorate-General III.
47. Wrycza S. (2007, red.); *UML 2.1. Ćwiczenia*, Helion, Gliwice.
48. Wrycza S. (2010, red.); *Informatyka Ekonomiczna*, PWE, Warszawa.
49. Wrycza S., Marcinkowski B. (2008); *Rozwój języka UML — zmiany w wersji 2.1*; „Software Developer’s Journal”, nr 2.

50. Wrycza S., Marcinkowski B. (2010); *Język inżynierii systemów SysML. Architektura i zastosowania*, Helion, Gliwice.
51. Wrycza S., Marcinkowski B., Wyrzykowski K. (2005); *Język UML 2.0 w modelowaniu systemów informatycznych*, Helion, Gliwice.
52. Wrycza S., Marcinkowski B., Wyrzykowski K. (2007); *Tablice informatyczne. UML*, Helion, Gliwice.

Notatki

PROGRAM PARTNERSKI

GRUPY WYDAWNICZEJ HELION



- 1. ZAREJESTRUJ SIĘ**
- 2. PREZENTUJ KSIĄŻKI**
- 3. ZBIERAJ PROWIZJĘ**

Zmień swoją stronę WWW
w działający bankomat!

Dowiedz się więcej i dołącz już dzisiaj!

<http://program-partnerski.helion.pl>