

Jacek Pruszyński<sup>1</sup>, Krzysztof Wyrzykowski<sup>2</sup>

Premium Technology Sp. z o.o. , Dział Inżynierii Oprogramowania  
Politechnika Gdańska, Katedra Inżynierii Oprogramowania

## USPRAWNIENIE PROCESU ZARZĄDZANIA WYMAGANIAMI W PRZEDSIĘWZIĘCIU PRODUKCYJNYM NA BAZIE METODYKI RATIONAL UNIFIED PROCESS

### Streszczenie

Artykuł opisuje koncepcję usprawnienia procesu zarządzania wymaganiami w informatycznym przedsięwzięciu produkcyjnym<sup>3</sup> z wykorzystaniem metodyki Rational Unified Process (RUP). Jest również próbą przedstawienia doświadczeń z przedsięwzięcia wdrożeniowego<sup>4</sup> obejmującego automatyzację tej koncepcji z wykorzystaniem technologii IBM Rational Software i Oracle. Artykuł podkreśla krytyczną rolę mentoringu oraz efektywnego transferu wiedzy dla sukcesu wdrożenia.

### 1. WSTĘP

Sformalizowanie procesu zarządzania wymaganiami w każdym przedsięwzięciu informatycznym co najmniej średniej wielkości, stanowi kluczowy warunek jego sukcesu z biznesowego i technicznego punktu widzenia. Z biznesowego punktu widzenia przez sukces przedsięwzięcia rozumiemy koniunkcję trzech warunków: (a) realizacja zakresu zakontraktowanego z klientem, (b) zakończenie przedsięwzięcia w zaplanowanym terminie, oraz (c) nieprzekroczenie budżetu przedsięwzięcia. W trakcie przebiegu przedsięwzięcia proces zarządzania wymaganiami odgrywa kluczową rolę w zapewnieniu spójnej, jednoznacznej i efektywnej komunikacji między jego udziałowcami.

Ponieważ oczekiwanie, że wszystkie wymagania zostaną wyspecyfikowane na początku przedsięwzięcia i nie będą się one zmieniać (rygorystyczne podejście modelu kaskadowego) jest nierealistyczne, wobec tego zastosowanie modelu iteracyjno—przyrostowego staje się racjonalną alternatywą. Rational Unified Process (RUP) [1] jest

<sup>1</sup> Jacek Pruszyński jest ekspertem w dziedzinie procesu wytwórczego w Dziale Inżynierii Oprogramowania w firmie Premium Technology Sp. z o.o.

<sup>2</sup> Krzysztof Wyrzykowski jest doktorantem w Katedrze Inżynierii Oprogramowania Politechniki Gdańskiej.

<sup>3</sup> *Przedsięwzięcie produkcyjne* to przedsięwzięcie, w którym dokonano usprawnienia procesu zarządzania wymaganiami.

<sup>4</sup> *Przedsięwzięcie wdrożeniowe* to przedsięwzięcie usprawnienia procesu zarządzania wymaganiami i jego wdrożenie, zostało ono zrealizowane w kontekście informatycznego przedsięwzięcia produkcyjnego.

taką alternatywą, modelem iteracyjno—przyrostowym. Metodyka ta, będąca bazą wiedzy procesowej inżynierii oprogramowania, stanowi wielokrotnie sprawdzony w przedsięwzięciach komercyjnych fundament metodyczny. Natomiast wspomagająca metodykę RUP technologia IBM Rational jest jedną z kilku możliwych opcji usprawnienia i(lub) automatyzacji procesu zarządzania wymaganiami.

## **2. CHARAKTERYSTYKA PRZEDSIĘWZIĘCIA PRODUKCYJNEGO**

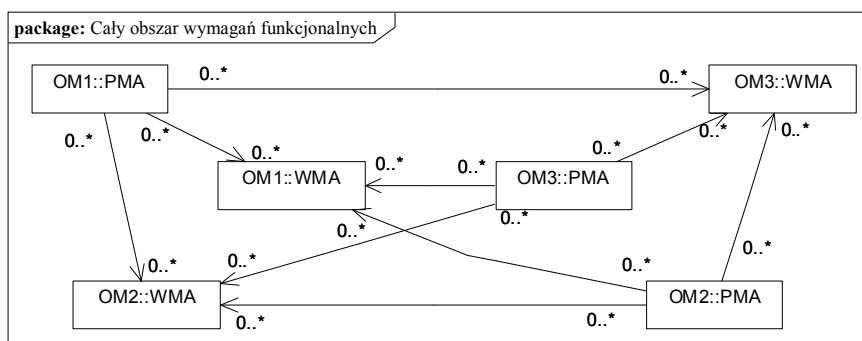
### **2.1 Zastosowanie podejścia iteracyjnego**

W przedsięwzięciu produkcyjnym zastosowano wariant podejścia iteracyjnego. Cykl życia przedsięwzięcia podzielono na fazy. Każda faza składała się z sekwencji kilku iteracji. W odróżnieniu od metodyki RUP, która wyróżnia jedynie cztery fazy tj.: Rozpoczęcia, Rozwinięcia, Konstrukcji i Przekazania, liczba faz w omawianym przedsięwzięciu produkcyjnym była większa. W szczególności odpowiednikiem fazy Konstrukcji w RUP było kilka faz przedsięwzięcia produkcyjnego. Liczba faz wraz z zakresem do realizacji dla każdej fazy została wyznaczona przez zapisy kontraktowe dokonane na początku przedsięwzięcia. Natomiast liczba iteracji w każdej fazie była planowana bezpośrednio przed jej rozpoczęciem i mogła być zwiększana w trakcie realizacji fazy (np. wskutek przesunięcia niezrealizowanego zakresu z poprzednich iteracji na koniec fazy—zjawisko tzw. "spadów" z poprzednich iteracji). W ramach danej iteracji realizowany był przyrost funkcjonalności systemu, który był uzgadniany na etapie jej planowania. Uzgodniony zakres nie mógł być rozszerzany. Z perspektywy procesu zarządzania wymaganiami uzgodnienie zakresu iteracji miało istotne znaczenie dla planowego przechodzenia do kolejnych faz.

### **2.2 Obszar zarządzania wymaganiami funkcjonalnymi**

Cały obszar wymagań funkcjonalnych podzielono na trzy *obszary merytoryczne (OM)*. Dodatkowo każdy z nich podzielono na szereg modułów analitycznych. *Moduł analityczny* reprezentował pewien zakres funkcjonalności systemu (np. zasilanie rejestrów słownikowych systemu). Wyróżniono dwie kategorie modułów analitycznych: (a) *podstawowe moduły analityczne (PMA)*—które ściśle określały wyodrębniony zakres funkcjonalności systemu, (b) *wspólne moduły analityczne (WMA)*—które określały wspólny zakres funkcjonalności systemu, wykorzystywany przez dwa lub więcej podstawowych modułów analitycznych. Rysunek 1 przedstawia koncepcję podziału całego obszaru wymagań funkcjonalnych.

Udziałowcami obszaru zarządzania wymaganiami funkcjonalnymi po stronie Wykonawcy byli: (a) Główny Analityk, który odpowiadał za zarządzanie całym obszarem wymagań oraz (b) Analitycy, każdy odpowiadał za jeden lub więcej modułów analitycznych.

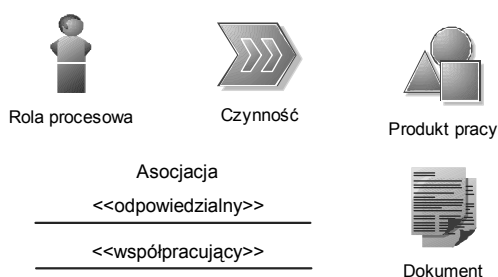


Rys.1. Koncepcja podziału obszaru wymagań funkcjonalnych

### 3. PRZEDSIĘWZIĘCIE WDROŻENIOWE

#### 3.1 Proces zarządzania wymaganiami

Usprawniony proces zarządzania wymaganiami opierał się na metodyce Rational Unified Process, a ściślej na jej dyscyplinie Zarządzanie Wymaganiami (ang. *Requirements*). Opracowany proces zarządzania wymaganiami został zdefiniowany w Planie Zarządzania Wymaganiami (ang. *Requirements Management Plan - RMP*). Do jego opracowania wykorzystano zmodyfikowany szablon RMP metodyki RUP. Natomiast do wizualizacji usprawnionego procesu zarządzania wymaganiami wykorzystano profil SPEM (ang. *Software Process Engineering Metamodel*) [3] języka UML [4][5][7]. Model opracowano w narzędziu IBM Rational Rose z zastosowaniem (a) diagramów przypadków użycia, (b) diagramów czynności, (c) diagramów klas oraz podstawowych kategorii modelowania profilu SPEM (rys.2). Publikacja w formacie html została automatycznie wygenerowana w narzędziu na podstawie modelu.



Rys.2. Podstawowe kategorie modelowania profilu SPEM [6] języka UML

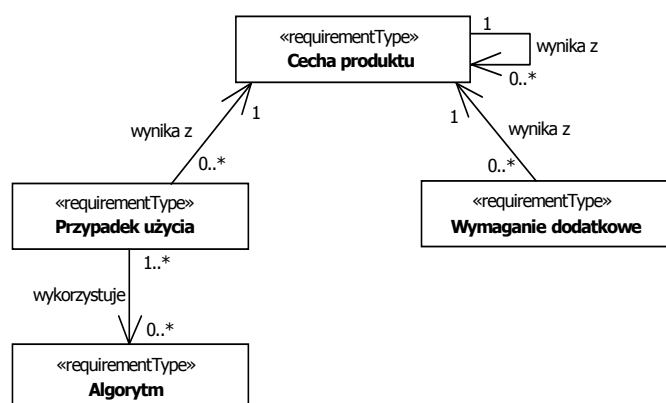
#### 3.2 Struktura wymagań

W celu opisanie struktury wymagań w przedsiębiorstwie produkcyjnym podzielono je na trzy główne kategorie: (a) systemowe, (b) modułowe i (c) wspólne. *Wymagania systemowe* to te, które dotyczyły całego systemu. *Wymagania modułowe* to te, należące do

jednego z podstawowych modułów analitycznych. Natomiast *wymagania wspólne* to te, które należały do jednego ze wspólnych modułów analitycznych.

Zdefiniowano również szereg typów wymagań<sup>5</sup> [2], które w terminologii języka UML są klasyfikatorami [4][5][7]. Każdy typ wymagania określał kategorię wymagań i był opisywany przy pomocy jednego lub więcej atrybutów. Każdy atrybut posiadał nazwę i typ, który wyznaczał dziedzinę jego wartości.

Zdefiniowano następujące typy wymagań: (a) *cecha systemu*—usługa dostarczana przez system (zwana także cechą produktu), (b) *przypadek użycia*—funkcja systemu widoczna z perspektywy użytkownika systemu, (c) *wymaganie dodatkowe*—wymaganie pozafunkcjonalne lub ograniczenie projektowe, (d) *algorytm*—szczególny rodzaj przypadku użycia opisujący procedurę przetwarzania danych, był zawsze włączany do innego przypadku użycia. Dla każdego z typów wymagań zdefiniowano adekwatny zbiór atrybutów. Zdefiniowano także semantykę dozwolonych zależności między poszczególnymi typami wymagań. Diagram klas (rys.3) przedstawia typy wymagań i ich związki.



Rys.3. Typy wymagań i związki między nimi

W odróżnieniu od typowego podejścia [2], zdefiniowana struktura wymagań umożliwiała tworzenie wielostopniowej hierarchii zależności między cechami systemu. Wynikało to z faktu, że cecha systemu w przedsięwzięciu produkcyjnym mogła reprezentować zarówno wysokopoziomą cechę funkcjonalną (np. import rejestru), jak i niskopoziomowe wymaganie (np. dotyczące interfejsu użytkownika), które mogło być powiązane z wysokopoziomowymi cechami funkcjonalnymi.

### 3.3 Dokumenty analityczne

W przedsięwzięciach informatycznych oprócz wymagań występują również dokumenty analityczne, w których umieszcza się wymagania i dodatkowe informacje związane z procesem zarządzania wymaganiami. Dokumenty te mogą być zatem traktowane jako kontenery wymagań. Liczba i rodzaje dokumentów analitycznych stosowanych w przedsięwzięciu zależą od jego specyfiki.

<sup>5</sup> Klasyfikacje podstawowych kategorii wymagań i typów wymagań są względem siebie ortogonalne.

Dla omawianego przedsięwzięcia produkcyjnego, w ramach uzgodnień między Zamawiającym a Wykonawcą zdefiniowano trzy typy dokumentów analitycznych: (a) *Specyfikacja Ogólnych Cech Produktu (SOCP)*—zawierała zarys funkcjonalności realizowanej w danej iteracji dla konkretnego modułu analitycznego, (b) *Specyfikacja Szczegółowych Cech Produktu (SSCP)*—zawierała szczegółową definicję cech systemu realizowanych w danej iteracji dla konkretnego modułu analitycznego oraz (c) *Specyfikacja Funkcji Produktu (SFP)*—zawierała specyfikacje wszystkich przypadków użycia realizowanych w danej iteracji dla konkretnego modułu analitycznego.

W przedsięwzięciu produkcyjnym dla każdej iteracji był wytwarzany zbiór dokumentów analitycznych, składający się z jednego egzemplarza każdego typu. SOCP służył do ogólnego ustalenia zakresu realizowanego w ramach iteracji. Na jego podstawie tworzono SSCP, który podlegał formalnemu zatwierdzeniu przez Zamawiającego i Wykonawcę. Następnie Wykonawca wytwarzał SFP i przysyłał ją do zaopiniowania Zamawiającemu.

Istotne znaczenie dla efektywnego, operacyjnego zarządzania przedsięwzięciem produkcyjnym miała możliwość realokacji wymagań między iteracjami, co skutkowało występowaniem tych samych wymagań w różnych dokumentach analitycznych. Wdrożenie opracowanego rozwiązania istotnie usprawniło tę funkcję poprzez dostarczenie funkcjonalności umożliwiającej zautomatyzowane tworzenie lub usuwanie zależności między danym wymaganiem i dokumentem analitycznym, co oznaczało alokację wymagania w dokumencie lub jego usunięcie z dokumentu.

### 3.4 Automatyzacja rozwiązań

Do automatyzacji opracowanego rozwiązania wykorzystano technologię Rational firmy IBM. Zastosowano następujące narzędzia IBM Rational: (a) RequisitePro, (b) SoDA, i (c) Rose, oraz bazę danych Oracle 10g.

RequisitePro wykorzystano do zbudowania centralnego repozytorium wymagań na fundamencie bazy Oracle. W narzędziu tym stworzono jeden projekt dla całego przedsięwzięcia produkcyjnego i odwzorowano w nim strukturę wymagań i ich atrybutów (patrz rozdział 3.2). Wykorzystano następujące rodzaje widoków udostępnianych przez RequisitePro: (a) *atrybutów*—tabelaryczna postać wymagań, (b) *macierz śledzenia zależności*, (c) *drzewo śledzenia zależności*. W celu polepszenia efektywności pracy analityków dla każdego modułu analitycznego zdefiniowano ponad 40 widoków.

W celu umożliwienia definiowania zależności między wymaganiami a dokumentami analitycznymi zdefiniowano specjalny typ wymagania ADOC reprezentujący te dokumenty. Wartości atrybutów wymagania ADOC reprezentowały moduł analityczny i iterację, której ten dokument dotyczył. Dla każdego modułu analitycznego i poszczególnych typów wymagań stworzono *macierze śledzenia zależności* (ang. traceability matrix), które umożliwiały alokację wymagań poszczególnych typów do dokumentów analitycznych. Przykładowo macierz śledzenia zależności UC na ADOC (przypadki użycia na dokument analityczny) umożliwiała alokację lub realokację wymagań w wytwarzany dokument analityczny dla danego modułu analitycznego.

Dla całego przedsięwzięcia produkcyjnego zdefiniowano spójny słownik pojęć projektowych i umożliwiono analitykom wstawianie dowolnego podzbioru tych pojęć do danego dokumentu analitycznego. W tym celu zdefiniowano specjalny typ wymagania PT wraz z odpowiednimi atrybutami oraz utworzono macierz śledzenia zależności PT na

ADOC (pojęcia projektowe na dokumenty analityczne). Alokacja pojęć projektowych do dokumentów była zautomatyzowana i tworzona analogicznie do innych typów wymagań.

W automatyzacji rozwiązania szczególną rolę odegrało narzędzie IBM Rational SoDA, które posłużyło do generowania dokumentów analitycznych. Każdy dokument analityczny istniał najpierw w postaci źródłowej (szablonu), która składała się z dwóch sekcji: (a) *redagowanej* i (b) *generowanej*. Ta pierwsza to fragment dokumentu redagowany manualnie przez analityka. Natomiast sekcja generowana to fragment dokumentu, który był generowany automatycznie, na podstawie bieżącej zawartości centralnego repozytorium wymagań z wykorzystaniem narzędzia SoDA. Dokumenty analityczne redagowano przy pomocy edytora MS Word. Automatyzacja rozwiązania obejmowała zaprojektowanie kompletnego i ujednoliconego zbioru stylów, wykorzystywanych w trakcie redagowania dokumentu. Dodatkowo w dokumentach analitycznych dokonano wizualnego rozróżnienia sekcji redagowanych i generowanych. Wykorzystując technologię VBA, zaimplementowano dodatkową funkcjonalność, umożliwiającą analitykom wygodną i efektywną pracę. Polecenia zostały pogrupowane i udostępnione w postaci specjalnego paska narzędziowego. Nowa funkcjonalność obejmowała: (1) przetwarzanie dokumentu analitycznego: (1a) skojarzenie dokumentu analitycznego z określoną wersją modułu analitycznego w określonej iteracji, (1b) generowanie dokumentu analitycznego, (2) wersjonowanie dokumentu analitycznego: (2a) stworzenie kolejnej wersji dokumentu analitycznego, (2b) aktualizacja jego metryki, (3) formatowanie dokumentu analitycznego: (3a) wstawianie tabel, (3b) formatowanie akapitów jako listy numerowanej lub buletowanej, (3c) formatowanie akapitów jako nagłówków (do czterech poziomów), (3d) podświetlanie sekcji generowanych, (4) polecenia współpracy z RequisitePro: (4a) wyświetlanie struktury wymagań w centralnym repozytorium wymagań oraz (4b) przełączenie się do aplikacji RequisitePro.

Wykorzystując narzędzie IBM Rational Rose zdefiniowano model przypadków użycia dla całego przedsięwzięcia produkcyjnego. Przechowywane były w nim te elementy modelu, których nie przechowuje się w repozytorium RequisitePro tj.: (a) aktorów, (b) związków pomiędzy aktorami, (c) związków pomiędzy aktorami i przypadkami użycia oraz (d) związków pomiędzy przypadkami użycia.

## 4. WDROŻENIE ROZWIĄZANIA

### 4.1 Transfer wiedzy

Jednym z krytycznych czynników decydujących o sukcesie wdrożenia rozwiązania jest efektywny transfer wiedzy w środowisko Odbiorcy. Podczas wdrożenia wykorzystano następujące rodzaje transferu wiedzy: (a) *szkolenia metodyczne*—w ramach których udziałowcy przedsięwzięcia wdrożeniowego poznali metodykę zarządzania wymaganiami zdefiniowaną w RUP (w tym celu wykorzystano certyfikowane szkolenie firmy IBM: *Requirements Management with Use Cases*), (b) *szkolenie narzędziowe*—w ramach których udziałowcy poznali narzędzie RequisitePro (w tym przypadku wykorzystano certyfikowane szkolenie firmy IBM: *Essentials of Rational RequisitePro*), oraz (c) *warsztaty wdrożeniowe*—dedykowane warsztaty podczas których udziałowcy kompleksowo poznali zasady działania zautomatyzowanego procesu zarządzania wymaganiami na przykładzie rzeczywistych wymagań projektowych. Szczególną rolę w efektywnym transferze wiedzy

w zakresie samego procesu zarządzania wymaganiami odegrała jego wizualizacja z wykorzystaniem profilu SPEM [3] języka UML [4][5][7] (patrz rozdział 3.1).

## 4.2 Mentoring

Grupowy transfer wiedzy jest warunkiem koniecznym, lecz często niewystarczającym dla udanego wdrożenia. Fundamentalną rolę odgrywa tu również mentoring, który może przybierać różne formy. W ramach przedsięwzięcia wdrożeniowego mentoring polegał na indywidualnym coachingu analityków Wykonawcy. Dla każdego uczestnika zorganizowano indywidualną sesję z mentorem, w trakcie której: (a) sprawdzano efektywność opanowania potrzebnej wiedzy metodycznej i narzędziowej do korzystania z opracowanego rozwiązania, (b) uzupełniono zidentyfikowane braki w tym zakresie, (c) demonstrowano wykonywanie poszczególnych czynności procesu zarządzania wymaganiami na przykładzie realnych danych projektowych. Mentoring był krytycznym czynnikiem powodzenia przedsięwzięcia wdrożeniowego.

## 5. PODSUMOWANIE

### 5.1 Ryzyka w projekcie wdrożeniowym

Dla przedsięwzięcia wdrożeniowego zidentyfikowano następujące główne ryzyka: (a) opór przed zmianami organizacyjnymi, (b) ograniczona dostępność udziałowców po stronie Klienta, (c) niejednoznaczność wymagań przedsięwzięcia wdrożeniowego, (d) przekroczenie czasu realizacji przedsięwzięcia wdrożeniowego. W trakcie realizacji przedsięwzięcia zmaterializowało się ryzyko pierwsze, trzecie oraz czwarte.

### 5.2 Model korzyści

Dla przedsięwzięcia wdrożeniowego określono trzy podstawowe kryteria sukcesu: (a) poszerzenie zakresu informacyjnego dla bieżącego procesu wytwórczego, (b) usprawnienie przepływu informacji między członkami zespołu przedsięwzięcia produkcyjnego oraz (c) transfer wiedzy i kompetencji do firmy Wykonawcy. Cele te zostały w znaczącym stopniu osiągnięte.

W rezultacie usprawnienia procesu zarządzania wymaganiami osiągnięto następujące korzyści: (a) powstało centralne repozytorium wymagań—poprzez wykorzystanie technologii firm IBM Rational Software i Oracle, (b) umożliwiono opisywanie własności wymagań—przez zdefiniowanie typów i atrybutów wymagań, (c) umożliwiono opisywanie wielostopniowych zależności między wymaganiami, (d) umożliwiono dokonywanie analizy wpływu (ang. impact analysis) jednych wymagań na inne, (e) umożliwiono lepsze śledzenie postępu prac projektowych—możliwość monitorowania stanu wymagań, (f) umożliwiono łatwą realokację wymagań między iteracjami przedsięwzięcia produkcyjnego, (g) dokonano unifikacji formatu dokumentów analitycznych—poprzez zastosowanie szablonów ze zdefiniowanym zestawem stylów, (h) umożliwiono wykrywanie wymagań wadliwych i niekompletnych (i) zautomatyzowano powstawanie dokumentów analitycznych—poprzez implementację koncepcji generowania tych dokumentów na podstawie danych z centralnego repozytorium wymagań. Dla poszczególnych typów dokumentów

analitycznych osiągnięto następujący stopień automatyzacji ich powstawania: SOPC = 60%, SSCP = 100%, SFP = 90%. Procenty określają stosunek długości wszystkich sekcji generowanych do długości całego dokumentu.

### 5.3 Wnioski

Opracowanie koncepcji usprawnienia procesu zarządzania wymaganiami oraz jej wdrożenie jest złożonym przedsięwzięciem, angażującym wielu udziałowców. O jego powodzeniu w znacznym stopniu decydują trzy czynniki: (a) trafna diagnoza stanu bieżącego, (b) wypracowanie i automatyzacja dobrej koncepcji, oraz czynniki być może najważniejsze (c) aspekty natury psychologicznej i organizacyjnej, takie jak: (c1) przekonanie udziałowców, że wdrażane rozwiązanie ułatwi im pracę, (c2) dostępność udziałowców w trakcie opracowywania koncepcji usprawnienia i podczas jej wdrażania.

Istotną rolę w usprawnionym procesie zarządzania wymaganiami odegrał jego formalny opis (jest to również prawda w odniesieniu do innych procesów). Interesującą propozycją, której zastosowanie w praktyce okazało się efektywne jest wykorzystywanie profilu SPEM [3] języka UML [4][5][7]. W tym kontekście kluczowe znaczenie miało zdefiniowanie możliwych sekwencji czynności procesowych (ang. *activities*), w tym sekwencji równoległych, artefaktów wejściowych (ang. *input artefacts*) i wyjściowych (ang. *output artefacts*). Formalny opis usprawnionego procesu zarządzania wymaganiami zapewnił jednoznaczność jego interpretacji przez jego twórców i użytkowników. Bardzo ważną rolę w efektywnym stosowaniu formalnego opisu procesu wytwórczego odegrał sposób jego publikacji. Najodpowiedniejszą formą okazała się witryna intranetowa udostępniona wszystkim udziałowcom przedsięwzięcia produkcyjnego.

### BIBLIOGRAFIA

- [1] IBM Corp.: *Rational Unified Process v2003.06.15*, 2005.
- [2] Lefingwell D., Widrig D.: *Managing Software Requirements: A Use Case Approach, Second Edition*, Addison-Wesley, 2003.
- [3] Object Management Group, *Software Process Engineering Metamodel Specification, Version 1.1*, 2005.
- [4] Object Management Group, *Unified Modeling Language Specification, Version 1.5*, 2003.
- [5] Rumbaugh J., Jacobson I., Booch G.: *The Unified Modeling Language Reference Manual, Second Edition*, Addison-Wesley, 2005.
- [6] SparxSystems - SPEM Profile: [http://www.sparxsystems.com/profile/SPEM\\_Profile.zip](http://www.sparxsystems.com/profile/SPEM_Profile.zip)
- [7] Wrycza S., Marcinkowski B., Wyrzykowski K.: *Język UML 2.0 w modelowaniu systemów informatycznych*, Helion, 2005.

### IMPROVING REQUIREMENTS MANAGEMENT PROCESS USING RATIONAL UNIFIED PROCESS IN AN INDUSTRIAL PROJECT

#### Summary

The article describes the concept of improving requirements management, on the basis of Rational Unified Process (RUP), in an industrial IT project. The article also tries to presents lessons learned in the course of implementing this concept using the technology of IBM Rational Software and Oracle. The article emphasize the role of mentoring and effective knowledge transfer to secure the success of the implementation.