

## **This paper should be cited as:**

W. Dąbrowski and A. Stasiak, "Nauczanie inżynierii wymagań z wykorzystaniem technik kształcenia na odległość," in *Postępy e-edukacji*, L. Banachowski, Ed. Warszawa: Wydawnictwo PJWSTK, 2013, pp. 117–126.

## Rozdział 7

---

# **Nauczanie inżynierii wymagań z wykorzystaniem technik kształcenia na odległość**

*Włodzimierz Dąbrowski\**

*Andrzej Stasiak\*\**

\*Polsko Japońska Wyższa Szkoła Technik Komputerowych  
ul. Koszykowa 86, 00-662 Warszawa, wlodek@pjwstk.edu.pl

\*\*Wojskowa Akademia Techniczna, Wydział Cybernetyki,  
ul. Kaliskiego 2, 00-908 Warszawa, astasiak@wat.edu.pl

Inżynieria wymagań jest ważnym działem inżynierii oprogramowania. Znaczenie inżynierii wymagań i prawidłowego przebiegu procesów zarządzania wymaganiami jest często kluczowym czynnikiem sukcesu dla projektów informatycznych. Znane są przypadki, gdy porażka projektu wynikała z błędów podczas analizy wymagań [1, 2, 3]. Świadomość i znajomość procesów pozyskiwania, dokumentowania i zarządzania wymaganiami dla współczesnych inżynierów informatyków nie zawsze jest wystarczająca. W niniejszej pracy zebrano doświadczenia autorów w obszarze kształcenia inżynierii wymagań z wykorzystaniem technik nauczania na odległość. Techniki zdalne autorzy stosowali na różnych poziomach nauczania, na pierwszym i drugim stopniu oraz kursach podyplomowych kształcenia informatyków. Metody te stosowane były też w różnym trybie studiów – dla studentów studiów stacjonarnych jako techniki wspomagające oraz dla studentów studiujących w modelu zdalnym SPRINT [4] jako techniki podstawowe. Główną platformą wykorzystaną do prowadzenia zajęć była platforma IBM Jazz [5]. Autorzy opracowali metodyczne podejście do nauczania inżynierii wymagań z zastosowaniem serwerowego środowiska Jazz i desktopowych środowisk CASE – IBM Rational (RequisitePro i Software Architect). Metoda została zweryfikowana w ciągu ostatnich trzech lat na dziesięciu kursach inżynierii oprogramowania.

## **Wprowadzenie**

Inżynieria wymagań na oprogramowanie jest relatywnie nowym pojęciem w informatyce. Zostało ono stworzone w celu określenia wszystkich czynności związanych z szeroko rozumianym pozyskiwaniem, dokumentowaniem i zarządzaniem wymaganiami dla systemów komputerowych. Słowo „inżynieria” w tym kontekście należy rozumieć jako systematyczną i zdefiniowaną działalność zapewniającą powodzenie tych procesów. Widoczne jest tu zatem odmienne od tradycyjnego podejścia, w którym inżynieria rozumiana jest jako „*używanie właściwości materii, energii oraz obiektów abstrakcyjnych dla tworzenia konstrukcji, maszyn i produktów, przeznaczonych do wykonywania określonych funkcji lub rozwiązania określonego problemu*” [6].

O roli inżynierii wymagań w procesie wytwarzania oprogramowania i znaczenia tej dyscypliny dla biznesowego powodzenia projektu informatycznego pisano wielokrotnie [1, 2, 3].

Wykształcenie inżyniera, który będzie miał odpowiednią wiedzę i umiejętności z inżynierii wymagań, nie jest łatwe. Powodem jest złożoność tej dyscypliny, konieczność łączenia różnych obszarów wiedzy<sup>1</sup> i technik oraz konieczność pracy w zespole. Z drugiej zaś strony odpowiednia wiedza i umiejętności w obszarze inżynierii wymagań są pożądanymi elementami w wykształceniu każdego inżyniera informatyka. Zdają sobie z tego sprawę zarówno uczelnie kształcące na kierunkach informatyka, jak i sami studenci, obserwując wymagania rynku pracy. O istotności tego obszaru świadczy intensywny rozwój organizacji takich jak IREB – *International Requirements Engineering Board* – organizacja powstała w 2006 roku, która nadzoruje międzynarodowy system akredytacji i certyfikacji w dziedzinie inżynierii wymagań, CPRE Certified Professional in Requirements Engineering i liczba wydanych certyfikatów (ponad 11 000).

## **Nauczanie inżynierii wymagań**

Nauczanie inżynierii wymagań jest obecne w większości programów nauczania na kierunkach informatycznych. Techniki nauczania zdalnego dają obecnie możliwość organizacji pracy w zespołach studenckich przy pracy nad złożonymi projektami. Jest to szczególnie ważne przy kształceniu umiejętności inżyniera wymagań. W przemyśle informatycznym wykorzystanie narzędzi wspomagających w modelu zdalnym staje się coraz powszechniejsze.

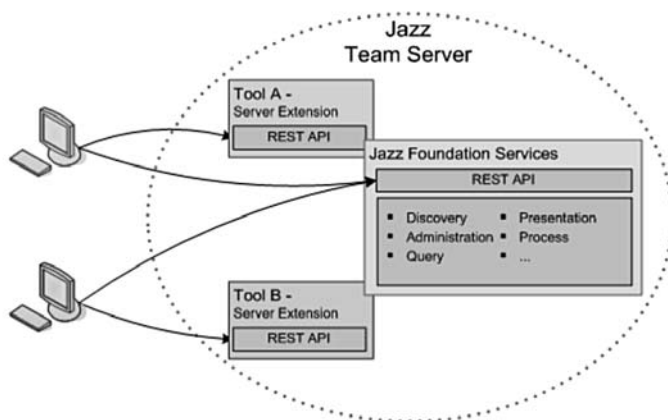
Z tego względu Autorzy zdecydowali się na wprowadzenie do zajęć ze studentami oprogramowania dedykowanego do wspomagania pracy zespołów projektowych w postaci platformy Jazz i produktów IBM Rational, jako reprezentatywnego środowiska pracy inżyniera informatyka [7], z którym z dużym prawdopodobieństwem może się spotkać absolwent uczelni w swoim życiu zawodowym.

Sama platforma Jazz stanowi w proponowanym rozwiązaniu serwerowe środowisko do komunikacji między członkami zespołu oraz między studentem a nauczycielem i może w znacznym stopniu usprawnić tę komunikację. Podstawowe cechy platformy, korzystne z punktu widzenia nauczania zdalnego, to między innymi: możliwość pracy z dowolnego miejsca, możliwość pracy w zespole, współdzielenie wyników pracy studenta z innymi członkami zespołu oraz z nauczycielem, łatwość raportowania postępów prac i planowania z uwzględnieniem iteracyjnego modelu zarządzania na przykład zgodnie z metodyką SCRUM.

---

<sup>1</sup> W procesach inżynierii wymagań zawsze uczestniczą dwie strony: Zamawiający i Wykonawca, które posługują się hermetycznymi językami, których połączenie pozwala na dostarczenie opisu rozwiązania. Te różne języki stanowią zasadnicze utrudnienie w komunikacji między stronami, przyczyniając się w znacznym stopniu do niepowodzeń projektów informatycznych [1].

Platforma Jazz udostępnienia zestaw narzędzi służących do konkretnych zadań i etapów w ramach realizacji projektu<sup>2</sup>, w tym także dla zagadnienia gromadzenia wymagań.



Rysunek 7.1. Architektura platformy Jazz Team Server

Ogólna architektura tego rozwiązania została przedstawiona na rysunku 7.1. W ramach tego produktu instalowany jest *Jazz Team Server*, który służy do zarządzania całą platformą, a także udostępnia poszczególne narzędzia jako usługi na serwerze. Daje to możliwość połączenia się z serwerem jazz i korzystania z usług za pomocą protokołu http (i https) lub desktopowej aplikacji klienckiej RSA.

Autorzy na potrzeby zajęć prowadzonych ze studentami opracowali metodyczne podejście do nauczania inżynierii wymagań z zastosowaniem produktów platformy Jazz. Podejście to bazuje na pakiecie przypadków studyjnych, odpowiednich scenariuszach działania i interakcji student – prowadzący oraz metodach systematycznego oceniań działań studentów. Metoda ta została zweryfikowana w ciągu ostatnich trzech lat na dziesięciu kursach inżynierii oprogramowania.

Proponowane podejście bazuje na modelu dyscyplin procesów wytwórczych zgodnych z modelem RUP, a w zakresie zarządzania projektem na metodyce SCRUM.

Model dyscyplin procesów wytwórczych wspierających inżynierię wymagań zgodnych z RUP składa się z następujących elementów:

1. Działania merytoryczne:
  - a) Definiowanie wymagań;
  - b) Specyfikowanie wymagań;
  - c) Weryfikacja wymagań;

<sup>2</sup> Podczas zajęć studenci wykorzystują następujące aplikacje tworzące środowisko Jazz: Rational Team Concert – jako środowisko zarządzania projektem (zwykle wykorzystując metodykę SCRUM lub RUP) i zmianą, Rational Quality Manager – jako środowisko zarządzania testami i jakością.

- d) Specyfikacja wymagań na oprogramowanie.
- 2. Zarządzanie projektem:
  - a) Utworzenie zespołu projektowego;
  - b) Zdefiniowanie procesu wytwórczego;
  - c) Utworzenie planów projektu.
- 3. Definiowanie środowiska projektu:
  - a) Utworzenie przestrzeni projektu;
  - b) Utworzenie repozytoriów wymagań:
    - Modeli werbalnych
    - Modeli wizualnych
- 4. Zarządzanie zmianą:
  - a) Utworzenie strumienia projektu;
  - b) Utworzenie komponentów projektu;
  - c) Zdefiniowanie wersji i wydań projektu.

Działania te muszą być wsparte odpowiednim przygotowaniem środowiska i zasobów zdalnej edukacji. Jest to bardzo ważne, ze względu na ograniczenia czasowe i dużą liczbę celów edukacyjnych do osiągnięcia. W naszej ocenie skutecznym sposobem ich osiągnięcia jest:

1. Precyzyjne określenie zadań i zdefiniowanie listy produktów jako artefaktów projektowych do wykonania przez studentów (zgodnie ze stale aktualizowanym harmonogramem);
2. Dostarczenie przewodników, które prezentują sposób wykonania każdego z zadań (w szczególności autorskich przewodników multimedialnych prezentujących sposoby realizacji zadań z wykorzystaniem zaproponowanego środowiska projektowego);
3. Dostarczenie wzorców w postaci szablonów dokumentacyjnych opracowywanych przez studentów artefaktów.

W procesie nauczania inżynierii wymagań na odległość naturalne jest, że należy z dużą uwagą i starannością podejść do przygotowania narzędziowego wsparcia procesów gromadzenia, zarządzania i śledzenia wymagań przez studenckie zespoły projektowe. W tym zakresie wykorzystujemy wspomniane wcześniej środowiska desktopowe (do pracy lokalnej) oraz serwerowe, dostępne przez platformę WEB. Nauka narzędzi realizowana jest w formie samokształcenia na podstawie przygotowanych materiałów szkoleniowych i praktycznie weryfikowana jest podczas warsztatów i przeglądów projektu. Warto tu również zaznaczyć, że każdy uczestnik projektu nie tylko uczestniczy w jego realizacji, ale również mierzy swój wysiłek, korzystając z metryk wspieranych przez platformy narzędziowe. Dzięki temu wykładowcy mają szansę zobiektywizowanej oceny nakładów pracy poniesionych przez poszczególnych studentów jako członków zespołów projektowych. Aby osiągnąć cele realizowanych przedmiotów, zespoły część wymagań modelują:

1. zwinnie: dla SCRUM'a tworząc opisy sytuacji (user story), artefakty w postaci wymaganych planów, jako zaległości produktowe, wersji i sprintu poznając odpowiednie praktyki: spotkania produktowe, sprintu, retrospektywne;
2. zgodnie z procesem RUP: w oparciu o model FURPS tworzą: wymagania słownikowe – TERM, żądania udziałowca – STRQ, cechy systemu – FEAT, wymagania typu przypadki użycia – UC, wymagania poza funkcjonalne (dodatkowe) – SR.

Istotne w procesie edukacji inżynierii wymagań jest również nauczenie dokumentowania wymagań. Studenci opracowują specyfikacje wymagań w oparciu o standardowe wzorce: formalny, nieformalny, tabelaryczny czy RUP. Zwykle najwięcej trudności przysparza studentom opracowanie poprawnego UML'owego modelu wymagań, a w nim: procesy strukturalizacji wymagań oraz zarządzanie ich złożonością.

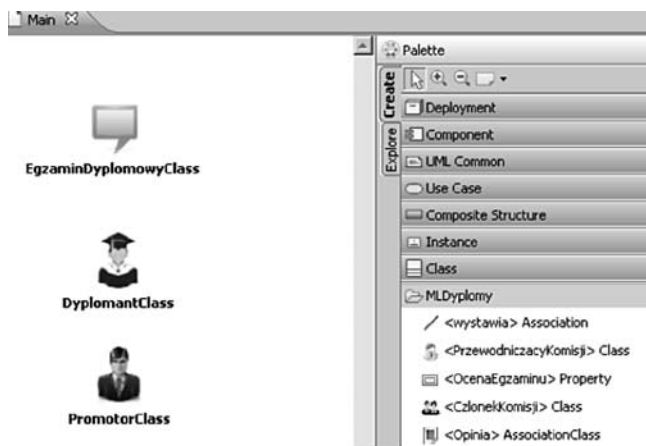
Z tego względu szczególnie istotne jest dokonanie przeglądu wymagań, najlepiej w oparciu o jednolite kryteria (tabela 7.1).

Tabela 7.1. Formularz oceny wymagań opracowanych zgodnie z procesem wytwórczym RUP

Lp	Artefakty oceniane podczas przeglądu	Max.
1	Dokument: przedstawienie problemu – dokument wykonawcy	5
2	Dokument: wizja projektu Dyplom	5
3	Raport: wyniki analizy lingwistycznej – lista kandydatów na usługi (przypadki użycia) i klasy systemu	5
4	Dokument: Słownik systemu Dyplom – z pojęciami w notacji BNF	5
5	Opracowanie planu (zaległości produktowych i sprintu)	5
6	Raport: wymagania typu STRQ (żądania udziałowców)	5
7	Raport: wymagania typu TERM (słownikowe)	5
8	Raport: wymagania typu FEAT (cechy systemu)	5
9	Raport: wymagania typu: UC (wymagania funkcjonalne)	5
10	Raport: widok śledzenia żądań udziałowców na cechy systemu	5
11	Raport: model przypadków użycia	5
12	Raport: modele aktywności (dla wybranych przypadków użycia)	5
13	Raport: Definicje kart CRC + przykładowe scenariusze	5
14	Tablice: jako widoki realizacji planów, przez członków zespołu, na platformie jazz	5
<b>Razem:</b>		70

Całość zajęć w modelu zdalnym opiera się na dwu istotnych filarach: pracy nad jednym spójnym projektem oraz samokształceniu studenta. Pierwszy z nich jest realizowany metodą Project Oriented Learning. Studenci realizują w czasie całych zajęć jeden wybrany projekt – ramach studiów I stopnia studenci zwykle budują złożone rozwiązania portalowe, natomiast w ramach II stopnia rozbudowują (bądź tylko modyfikują) je do modelu usługowego, budując rozwiązanie w architekturze SOA. Na rysun-

ku 7.2 przedstawiono elementy realizowanego projektu studenckiego. Projekt dotyczył walidacji wymagań, wobec systemu wspomagania obsługi egzaminów dyplomowych na uczelni wyższej. Weryfikacja narzędzia polegała na zapisaniu przez wszystkich członków zespołu w utworzonym profilu modelu swojego egzaminu dyplomowego (na studiach I stopnia).

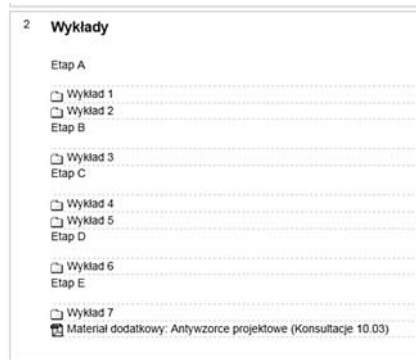


Rysunek 7.2. Elementy projektu realizowane w zespole studenckim

Na rysunku 7.3 przedstawiony jest element modelu statycznego, który został wykonany przez studentów w ramach zajęć. Aby zbudować oba te modele, studenci modelowali szkice dziedzin projektu, w ramach których:

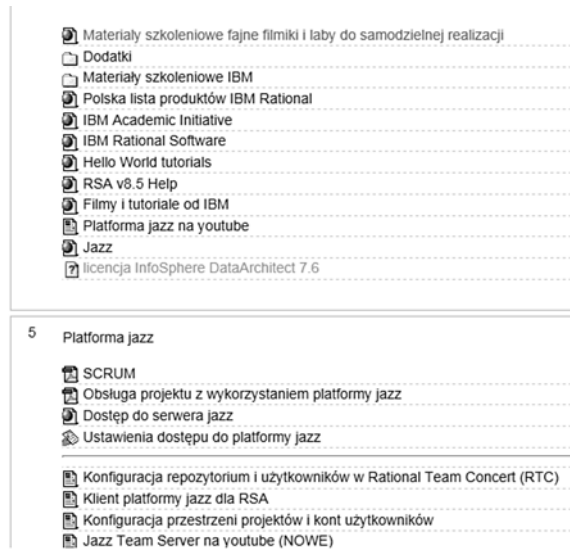
1. opisywali problemy projektowe, tworząc zarówno werbalne, jak i graficzne (w postaci komiksów) opisy dziedzin;
  - a) tworzyli historyjki użytkownika;
  - b) modelowali zarówno karty CRC, jak i scenariusze ich użycia;
2. budowali ontologie (dla dziedzin projektu):
  - a) Określ pojęcia i ich znaczenie;
  - b) Określ dziedziny pojęć;
  - c) Określ w dziedzinie projektu zbiór predykatów;
  - d) Określ relacje między pojęciami;
  - e) Zdefiniuj atrybuty pojęć, a następnie hierarchię pojęć;
3. tworzyli metamodele:
  - a) Zdefiniuj kluczowe abstrakcje w każdej z modelowanych dziedzin;
  - b) Określ atrybuty kluczowych abstrakcji;
  - c) Zbuduj metamodel dla każdej dziedziny;
4. zbuduj profil:
  - a) Utwórz projekt profilu;
  - b) Zdefiniuj stereotypy;





Rysunek 7.4. Repozytorium wykładów kursu zdalnego z podziałem na etapy projektu

dań w narzędziu, którym posługują się studenci w czasie realizacji projektu oraz odnośniki do zweryfikowanych i sprawdzonych materiałów zewnętrznych – na przykład do dyskusji lub szkoleń proponowanych przez firmę IBM. Na rysunku 7.5 przedstawiono fragment witryny przedmiotu prowadzonej na platformie moodle z materiałami uzupełniającymi do kursu.



Rysunek 7.5. Repozytorium kursu na odległość na platformie moodle

Budowa i utrzymywanie repozytorium materiałów (podstawowych i uzupełniających) jest istotnym elementem przyczyniającym się do sukcesu kursu nauczania inżynierii wymagań. Na przestrzeni ostatnich kilku lat repozytorium to było rozwijane i dostosowywane do potrzeb studentów. Zawsze cieszyło się zainteresowaniem i studenci postrzegali je jako istotną pomoc. Spotkaliśmy się też z opiniami uczestników kursu, że repozytorium jest za duże, zbyt rozbudowane, a co z tym idzie trudne do wykorzy-



stania. Opinie te utwierdzają nas w przekonaniu, że udostępnianie studentom takiego repozytorium – w szczególności sprawdzonych materiałów uzupełniających – jest niezbędne, a zdolność części naszych słuchaczy poruszania się w dużej liczbie materiałów źródłowych i ich poprawnej selekcji jest niewystarczająca.

Istotnym elementem kursu inżynierii wymagań jest system oceniania i komentowania wyników prac studentów i postępów w projekcie. Każdy etap projektu jest oceniany i komentowany przez wykładowcę. Do oceny wykorzystywany jest przy tym specjalnie opracowany na potrzeby kursu system oceny jakości przesłanego rozwiązania wraz z metryką punktową.

Ciekawym natomiast doświadczeniem było zbudowanie platformy komunikacyjnej ze studentami o nazwie „Podium”, na której prezentowane były najlepsze rozwiązania oraz dyskusja związana z ich oceną.

Dzięki niej systematyczne przeglądy projektu budują wiedzę u studentów nie tylko na podstawie popełnianych przez „siebie” błędów (każdy artefakt jest oceniany), ale pozwalają spojrzeć na problemy budowy modelu wymagań z perspektywy wielu inżynierów wymagań i dokonać samooceny własnych produktów.

## Podsumowanie

Nauczanie inżynierii wymagań ze względu na złożoność i interdyscyplinarność zagadnienia jest sporym wyzwaniem. Wymaga połączenia technik typowo inżynierskich, jak precyzja wypowiedzi, znajomość języka modelu itp. z technikami miękkimi, takimi jak umiejętność pozyskiwania wymagań, komunikacji w zespole itp. Wyzwanie jest szczególnie trudne, gdy nie mamy możliwości bezpośredniego kontaktu ze studentem, a tak dzieje się w modelu nauczania na odległość. Z pomocą przychodzi nam odpowiednie narzędzia. W naszej metodzie nauczania inżynierii wymagań wykorzystaliśmy zestaw narzędzi pracy grupowej i budowy modeli wymagań dostarczanych przez firmę IBM wraz z platformą jazz.net. Doświadczenia kilku edycji kursów na odległość z wykorzystaniem tej platformy są pozytywne. W znacznym stopniu usprawnia ona pracę i komunikację pomiędzy studentem a wykładowcą oraz między studentami wzajemnie. Wymaga jednak od studenta odpowiedniego obycia i zdobycia doświadczenia w jej obsłudze. Od prowadzącego zajęcia wymaga natomiast utrzymania odpowiedniej infrastruktury serwerowej i w początkowym okresie znacznych nakładów czasu na jej konfigurację. Wysiłek włożony w jej wdrożenie zwraca się jednak dość szybko. Zajęcia z wykorzystaniem platformy jazz.net w efekcie przynoszą, przy większej liczbie studentów, oszczędności czasowe (przede wszystkim przy przeglądaniu i komentowaniu modeli studenckich) oraz podnoszą jakość prowadzonych zajęć i dają więcej satysfakcji uczestnikom kursu.

Należy pamiętać, że platforma jazz.net powinna być uzupełniona metodycznym podejściem do procesu dydaktycznego, z jasnym określeniem zadań dla studentów i podziałem na etapy całego projektu oraz o odpowiednio przygotowane repozytorium wiedzy udostępniane w czasie kursu. Ostatnim czynnikiem sukcesu w nauczaniu in-

żynierii wymagań w modelu zdalnym jest systematyczny kontakt ze studentami prowadzony na platformie i systematyczne przekazywanie studentom informacji zwrotnej o postępach w wykonywanych przez nich projektach.

Kurs zdalny prowadzony według wymienionych zasad jest pracochłonny dla obu stron – i wykładowcy, i studenta. Pozwala jednak wykształcić u studentów niezbędne umiejętności i kompetencje w obszarze inżynierii wymagań, które są weryfikowane w czasie egzaminu na uczelni.

## Literatura

1. “New Standish Group report shows more projects failing and less successful projects” (2009). The Standish Group, [http://www.standishgroup.com/newsroom/chaos\\_2009.php](http://www.standishgroup.com/newsroom/chaos_2009.php).
2. “KPMG New Zealand Project Management Survey 2010” (2010). KPMG, <http://www.kpmg.com/NZ/en/IssuesAndInsights/ArticlesPublications/Documents/Project-Management-Survey-report.pdf>.
3. “Flawed Requirements Trigger 70% of Project Failures” (2006). Info-Tech Research Group, <http://www.infotech.com/research/flawed-requirements-trigger-70-of-project-failures>.
4. Galwas Bogdan, „SPrint – Studia Przez Internet” – model Studiów na Odległość Politechniki Warszawskiej, Wirtualna Edukacja nr 9.
5. Schroter, A., Predicting build outcome with developer interaction in Jazz, Software Engineering, 2010 ACM/IEEE 32nd International Conference on Volume: 2, 2010, Page (s): 511–512.
6. Dean Leffingwell, Zarządzanie Wymaganiami, WNT, Warszawa 2003.
7. jazz.net.