

Plan nauczania

REQB®

Certyfikowany Specjalista Inżynierii Wymagań

Poziom Podstawowy



**Requirements
Engineering**
Qualifications Board

Wersja 2.1

2014

Uwagi na temat praw autorskich

Wszystkie prawa do wersji angielskiej zastrzeżone dla ©Global Association for Software Quality, GASQ

Wszystkie prawa do wersji polskiej zastrzeżone dla SJSI – Stowarzyszenie Jakości Systemów Informatycznych

Historia zmian

Wersja	Data	Uwagi
0.1	17/04/2006	Pierwsza wersja sylabusu, stworzenie podstawowej struktury sylabusu
0.2	20/07/2006	Rozszerzenie do wersji 0.1
0.3	04/09/2006	Dalsze rozszerzenie i korekta wersji 0.1
0.4	10/10/2006	Korekta wersji 0.3
0.5	15/12/2006	Korekta wersji 0.4
0.6	07/02/2007	Kompletna korekta wersji 0.5
0.7	10/04/2007	Korekta wersji do przeglądu
0.8	15/06/2007	Wersja alfa
0.9	01/09/2007	Wersja beta
1.0	15/01/2008	Wydanie wersji 1.0
1.1	29/05/2008	Aktualizacja do wersji 1.1
1.2	01/07/2008	Aktualizacja do wersji 1.2
1.3	15/05/2011	Aktualizacja do wersji 1.3
1.4	30/06/2013	Zmiany wynikające z przeglądu wersji 1.4
1.5	19/08/2013	Zmiany: zmiana i zredukowanie ilości LO, dodanie mapowania pomiędzy BO a LO, zmiany w strukturze rozdziału 3 Proces Inżynierii Wymagań.
1.6	20/10/2013	Zmniejszenie ilości BO, zmiana klasyfikacji wymagań, zmiana definicji modelu konceptualnego, dodanie większej liczby wymagań ćwiczeniowych, zmniejszenie listy polecanych pozycji literatury, dołączenie uwag po przeglądzie wersji 1.5.
1.7	01/11/2013	Niewielkie zmiany po ostatnim cyklu przeglądów.
2.0	01/12/2013	Wersja ostateczna.
2.1	09/02/2014,	Małe zmiany.
2.1 PL	09/09/2015	Wersja polska

Cel główny

Motywy przewodnim tego planu nauczania było ciągle zwiększanie się złożoności systemów oraz oprogramowania a także naszej od nich zależności. Rezultatem jest wysoki poziom zależności od liczby błędów w oprogramowaniu, sprzęcie komputerowym oraz innych produktach. Z tego też względu REQB (ang. *Requirements Engineering Qualifications Board*) zdecydowało się na opracowanie jednolitych międzynarodowych wytycznych w obszarze Inżynierii Wymagań. Standardy przypominają bowiem języki, tzn. tylko gdy jesteśmy je w stanie zrozumieć, możemy z nimi efektywnie pracować. Grupa ekspertów zebrała się i stworzyła niniejszy plan nauczania, by opracować taki jednolity język w obszarze Inżynierii Wymagań

Podziękowania

Dokument ten został stworzony przez główny zespół Grupy Roboczej REQB dla Poziomu Podstawowego (edycja 2013), w składzie: Karolina Zmitrowicz (przewodnicząca), Sergey Kalinov, Beata Karpińska, Andrey Konushin, Salvatore Reale, Alexander Lindner, Ingvar Nordström, Alain Ribault.

Zespół główny dziękuje wszystkim Krajowym Radom za ich wkład i sugestie.

Tłumaczenie na język polski wykonał zespół w składzie: Bartosz Bielewicz, Adam Ścierański, Lucjan Stapp, Adam Wałęska, przeglądy: Grzegorz Libor, Radosław Smilgin, Lucjan Stapp i Karolina Zmitrowicz

Spis Treści

Wstęp	6
1 Wstęp do wymagań	13
1.1 Pojęcie wymagania.....	15
1.1.1 Koncepcja problemu, rozwiązania oraz produktu	15
1.1.2 Definicja i klasyfikacja wymagań	15
1.1.3 Typowe atrybuty wymagań	17
1.1.4 Jakość wymagań	19
1.1.5 Inżynieria Wymagań, zarządzanie wymaganiami, opracowywanie wymagań	22
1.2 Standardy oraz normy.....	24
2 Kontekst Inżynierii Wymagań	26
2.1 Inżynieria Wymagań w kontekście.....	27
2.2 Procesy powiązane.....	29
3 Proces Inżynierii Wymagań.....	30
3.1 Wprowadzenie do procesu Inżynierii Wymagań	31
3.2 Ogólny proces Inżynierii Wymagań.....	32
3.3 Role i odpowiedzialności.....	36
4 Zarządzanie wymaganiami.....	38
4.1 Wstęp do zarządzania wymaganiami.....	40
4.2 Zarządzanie projektem i ryzykiem	41
4.3 Śledzenie wymagań.....	45
4.4 Zarządzanie konfiguracją i zarządzanie zmianą	47
4.5 Zapewnienie jakości	50
5 Opracowanie wymagań	54
5.1 Wstęp do opracowywania wymagań.....	56
5.2 Identyfikacja wymagań	57
5.3 Analiza wymagań	71
5.3.1 Modelowanie rozwiązania.....	74
5.4 Specyfikacja wymagań	80

5.5	Walidacja i weryfikacja.....	84
6	Modele Inżynierii Wymagań.....	85
6.1	Modele i podejścia do wytwarzania i utrzymania.....	86
6.2	Modele dojrzałości.....	91
7	Wsparcie narzędziowe.....	93
7.1	Korzyści z narzędzi.....	94
7.2	Kategorie narzędzi.....	95
7.3	Wybór narzędzia.....	96
8	Tabele/Rysunki.....	98
9	Książki i publikacje.....	100
10	Indeks.....	102

Wstęp

Cel

Ten plan nauczania definiuje Poziom Podstawowy programu szkoleniowego REQB Certyfikowany Specjalista Inżynierii Wymagań (CPRE). Zespół REQB stworzył ten plan nauczania we współpracy z GASQ Global Association for Software Quality. Zakres programowy REQB obejmuje proces Inżynierii Wymagań dla wszystkich typów produktów związanych z technologiami informatycznymi (IT), w skład których wchodzi: oprogramowanie, sprzęt komputerowy, usługi i procesy biznesowe, a także dokumentacja.

REQB przekazuje niniejszy sylabus różnym grupom w niniejszych celach:

1. Radom Krajowym: w celu przetłumaczenia na język lokalny oraz akredytacji trenerów. Tłumaczenie może oznaczać zaadaptowanie planu nauczania do konkretnych potrzeb językowych oraz może wymagać modyfikacji bibliografii (książek oraz publikacji), by móc skutecznie przenieść treści na język lokalny.
2. Komisjom Egzaminacyjnym: w celu umożliwienia stworzenia pytań egzaminacyjnych w językach lokalnych spełniających jednocześnie Cele Kształcenia zdefiniowane w planie nauczania.
3. Trenerom starającym się o akredytację trenerską REQB, w celu dostarczania szkoleń i materiałów szkoleniowych. Wymagane jest by wszystkie obszary niniejszego sylabusa zostały ujęte w dokumentach szkoleniowych.
4. Kandydatom do certyfikacji, jako materiał przygotowawczy do egzaminu, (jako część szkolenia lub niezależnie).
5. Międzynarodowej społeczności Inżynierii Wymagań, w celach poszerzania wiedzy i kompetencji z zakresu Inżynierii Wymagań.

Egzaminy

Egzamin na Certyfikowanego Specjalistę Inżynierii Wymagań Poziomu Podstawowego oparty jest na niniejszym planie nauczania. Wszystkie sekcje planu nauczania mogą być pokryte w pytaniach egzaminacyjnych. Dane pytania egzaminacyjne nie musi koniecznie dotyczyć jednej sekcji planu nauczania; może odnosić się do kilku sekcji na raz.

Format pytań egzaminacyjnych to test wyboru.

Do egzaminu można podejść po ukończeniu akredytowanego kursu lub podczas otwartych sesji egzaminacyjnych (bez uprzedniego uczestnictwa w kursie). Szczegółowe informacje dotyczące dat poszczególnych egzaminów można znaleźć na stronie REQB (www.reqb.org), na stronie GASQ (www.gasq.org) lub stronie lokalnego egzaminatora w Waszym kraju (www.sjsi.org).

Akredytacja

Dostawcy kursu REQB Certyfikowanego Specjalisty Inżynierii Wymagań Poziomu Podstawowego muszą otrzymać akredytację Global Association for Software Quality. Tamtejsi eksperci przeprowadzają przegląd dokumentacji szkoleniowej dostawcy pod kątem dokładności oraz zgodności z zawartością Celów Kształcenia zawartych w sylabusie. Kurs akredytowany uznawany jest za zgodny z planem nauczania. Na koniec takiego kursu może zostać przeprowadzony oficjalny egzamin na Certyfikowanego Specjalistę Inżynierii Wymagań (ang. CPRE) lub przez niezależny instytut certyfikacyjny (zgodnie z zasadami ISO 17024).



Akredytowanych Dostawców Kursów można rozpoznać poprzez oficjalne logo REQB Akredytowany Dostawca Kursu (ang. REQB Accredited Training Provider).

Międzynarodowość

Niniejszy plan nauczania powstał przy współpracy międzynarodowych ekspertów. Zawartość sylabusu może być zatem uznana za standard międzynarodowy. Dzięki temu możliwe jest szkolenie oraz egzaminowanie w różnych krajach na tym samym poziomie.

Dodatkowe korzyści

Cele, korzyści oraz główny nacisk programu REQB Certyfikowanego Specjalisty Inżynierii Wymagań Poziomu Podstawowego przedstawia poniższa tabela (Tabela 1 Cele programu Poziomu Podstawowego, jego korzyści oraz główny nacisk).

Cele	
Uzyskanie nowych kluczowych kwalifikacji	Każde oprogramowanie, sprzęt komputerowy, lub rozwiązanie usługowe oparte jest na wymaganiach interesariuszy i ma na celu zaspokojenie specyficznych potrzeb biznesowych. By móc dostarczyć rozwiązanie zgodne z potrzebami grupy docelowej, potrzebna jest odpowiednia Inżynieria Wymagań. Zarządzanie wymaganiami oraz przetworzenie ich w konkretny projekt rozwiązania jest bardzo ważnym zadaniem, jednak w celu osiągnięcia pełnego sukcesu oraz dostarczenia możliwie najlepszego rozwiązania niezbędne jest również rozważenie jakości. Wszystkie te aspekty zostały omówione w sylabusie Poziomu Początkującego.
Wzrost satysfakcji klientów	Satysfakcja klienta zostaje osiągnięta poprzez spełnienie jego oczekiwań oraz potrzeb za pomocą zaproponowanego rozwiązania. Udoskonalona Inżynieria Wymagań minimalizuje rozbieżności pomiędzy oczekiwaniami a postrzeganiem zgodności. Dodatkowo wysoka jakość wzmacnia lojalność klienta.
Minimalizacja kosztów wytwarzania oraz późniejszych	Odpowiednia Inżynieria Wymagań minimalizuje ryzyka projektowe i produktowe, a także pomaga uniknąć kosztów związanych z przerabianiem produktu, które wynikają często z rozbieżności pomiędzy oczekiwaniami klienta a efektywnością wytwarzanego rozwiązania. Dobrze zarządzane wymagania dodatkowo redukują koszty związane z przyszłymi usprawnieniami oraz czynnościami naprawczymi.
Przewaga konkurencyjna	Inżynieria Wymagań pomaga w dostarczaniu lepszych produktów oraz usług spełniających wszystkie potrzeby oraz oczekiwania biznesowe. Lepsze produkty oraz usługi pomagają z kolei osiągnąć zaufanie oraz lojalność pożądaných grup docelowych oraz zwiększają przewagę nad konkurencją.

Cele	
Nacisk	
Wymagania	Zrozumienie podstawowych koncepcji związanych z wymaganiami, ich klasyfikacja oraz poziom abstrakcji; objaśnienie znaczenia atrybutów wymagań oraz roli kryteriów jakości.
Proces Inżynierii Wymagań	Objaśnienie procesu Inżynierii Wymagań, jego aktywności, aktorów, rezultatów (produktów), najważniejszych zasad oraz dobrych praktyk dla wytwarzania oraz zarządzania wymaganiami oprogramowania, sprzętu komputerowego oraz usług.
Standardy, normy oraz regulacje prawne	Przegląd najważniejszych standardów, norm oraz regulacji prawnych oraz ich wpływu na Inżynierię Wymagań.
Inżynieria Wymagań – wytwarzanie i modele utrzymania	Objaśnienie zasad usprawniania procesu Inżynierii Wymagań: od procesów ogólnych do modeli szczegółowych.
Narzędzia i techniki	Objaśnienie użycia oraz korzyści płynących z poszczególnych technik identyfikacji wymagań; modelowanie problemów oraz ich rozwiązań; zapewnianie jakości oraz techniki kontroli; wsparcie narzędziami procesu Inżynierii Wymagań.
Ćwiczenia	Pisanie dobrych wymagań; kontrola jakości wymagań; identyfikacja i analiza wymagań.

Tabela 1 Cele programu Poziomu Podstawowego, jego korzyści oraz główny nacisk

Program Poziomu Podstawowego REQB Certyfikowany Specjalista Inżynierii Wymagań jest przeznaczony dla wszystkich osób zaangażowanych w tworzenie rozwiązań produktowych/biznesowych, a także w zarządzanie (włączając analityków biznesowych oraz systemowych), zespoły marketingowe, projektantów sprzętu komputerowego/oprogramowania, projektantów GUI (Graficznego Interfejsu Użytkownika), menadżerów projektu, przedstawicieli klienta, zespołów utrzymania oraz technicznych, audytorów IT (technologii informacyjnych), oraz przedstawicieli działów zarządzania jakością.

Głównym celem programu Poziomu Podstawowego jest dostarczenie wspólnej terminologii oraz powszechnego, spójnego zrozumienia kluczowych koncepcji związanych z procesem Inżynierii Wymagań. Baza wiedzy dostarczona poprzez program Poziomu Podstawowego wspiera powszechne definicje oraz koncepcje związane z Inżynierią Wymagań, a także wyjaśnia sam proces

Inżynierii Wymagań wraz z jego rezultatami (produktami). Program został oparty na powszechnie akceptowanych standardach oraz dobrych praktykach.

Cele Kształcenia

Cele Kształcenia zawarte w tym sylabusie zostały podzielone na różne poziomy poznawcze (kognitywne) wiedzy (tzw. K-poziomy) . Dzięki temu kandydat ma możliwość rozpoznania w każdym momencie danego “poziomu wiedzy”.

Każda z sekcji sylabusu posiada powiązany z nią poziom kognitywny:

- K1 - Biegłość/Wiedza: Znajomość konkretnych szczegółów tj. pojęcia, definicje, fakty, dane, zasady, pryncypia, teorie, charakterystyki, kryteria, procedury. Uczniowie są w stanie przywołać i wyrazić swoją wiedzę.
- K2 - Rozumienie: Uczniowie są w stanie wyjaśnić i podsumować fakty używając własnych słów, podać przykłady, zrozumieć konteksty i zinterpretować zadania.
- K3 - Zastosowanie: Uczniowie są w stanie zastosować swoją wiedzę w nowych specyficznych sytuacjach, np. poprzez zastosowanie konkretnych zasad, metod lub procedur.

Rezultaty Biznesowe

Po ukończeniu programu REQB Certyfikowany Specjalista Inżynierii Wymagań Poziomu Podstawowego dana osoba potrafi:

- BO01 Przekazać podstawowe koncepcje Inżynierii Wymagań, wymagania oraz ich role w cyklu życia produktu.
- BO02 Uświadomić znaczenie procesu inżynierii Wymagań oraz jego produktów.
- BO03 Przekazać główne czynności oraz cele procesu Inżynierii Wymagań oraz jego poszczególnych części tj. zarządzanie wymaganiami i opracowywanie wymagań.
- BO04 Przekazać podstawowe umiejętności oraz kompetencje wymagane od inżyniera wymagań.
- BO05 Przekazać znaczenie Inżynierii Wymagań dla sukcesu tworzenie i/lub zarządzania projektem.
- BO06 Przekazać możliwości wykorzystania różnych technik pozyskiwania wymagań.
- BO07 Przekazać znaczenie śledzenia i priorytetyzacji, a także ich znaczenia dla procesów Inżynierii Wymagań.
- BO08 Przekazać możliwości użycia modelowania w tworzeniu rozwiązania biznesowego dla danego problem biznesowego.
- BO09 Przekazać znaczenie zapewnienia jakości i kontroli jakości dla procesu Inżynierii Wymagań.

- BO10 Przekazać podobieństwa i różnice występujące w procesie Inżynierii Wymagań pomiędzy powszechnymi modelami związanymi z procesami tworzenia/utrzymania.

Poziom szczegółowości

Plan nauczania REQB Certyfikowany Specjalista Inżynierii Wymagań Poziomu Podstawowego ma na celu wsparcie spójnego modelu szkolenia i egzaminowania na szczeblu międzynarodowym. By osiągnąć ten cel sylabus został podzielony na poniższe moduły:

- Ogólne cele instruktażowe opisujące intencje stojące za programem certyfikacji REQB Poziomu Podstawowego.
- Cele kształcenia dla każdego obszaru wiedzy opisujące obiektywne kognitywne rezultaty kursu oraz kwalifikacje osiągnięte przez uczestników.
- Lista informacji przydatnych w nauczaniu, zawierająca opis, odniesienia do innych źródeł tj. zaakceptowana literatura techniczna, normy lub standardy, jeśli takowe są wymagane.
- Lista pojęć, które uczestnik jest w stanie przywołać w pamięci i zrozumieć. Poszczególne pojęcia zostały szczegółowo opisane w „Słowniku terminów stosowanych w inżynierii Wymagań REQB”. Zawartość planu nauczania nie porywa całego obszaru wiedzy z zakresu Inżynierii Wymagań. Pokrywa natomiast zakres i poziom szczegółowości odpowiedni dla certyfikacji Poziomu Podstawowego.

Organizacja planu nauczania

Plan nauczania składa się z siedmiu głównych rozdziałów.

Nagłówek u góry każdego z rozdziałów opisuje temat omawiany w danym rozdziale oraz minimalny czas, który należy spędzić nad jego zawartością.

Cele kształcenia, które należy osiągnąć po przerobieniu danego rozdziału znajdują się na jego początku.

W każdym z rozdziałów znajduje się pewna liczba sekcji. Dla każdej z sekcji został zdefiniowany minimalny czas, który należy spędzić nad jej zawartością. Podsekcje nieposiadające swego własnego miernika czasu zaliczają się do ogółu czasu minimalnego przeznaczonego na daną sekcję.

Na przykład:

1. Wstęp do Wymagań	90 minut
----------------------------	-----------------

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu.

1.1 Dlaczego wymagania są niezbędne

LO-1.1.1 Podaj definicję wymagania (K1)

Główny nagłówek wskazuje na potrzebę zarezerwowania sobie minimum 90 minut czasu, by móc w pełni przyswoić materiał zawarty w tym rozdziale.

W każdym z rozdziałów może znajdować się kilka sekcji, a każda z nich cechować się będzie specyficznym celem kształcenia. Zawartość konkretnej sekcji musi pozwalać na osiągnięcie odpowiadającego jej celu kształcenia.

1 Wstęp do wymagań	110 minut
---------------------------	------------------

Pojęcia

Inżynieria Wymagań, kryteria jakości dla wymagań, krytyczność, ograniczenia biznesowe, ograniczenia techniczne, ograniczenie, opracowywanie wymagań, priorytet, problem biznesowy, produkt, rozwiązanie, system, walidacja, weryfikacja, wymagania funkcjonalne, wymagania niefunkcjonalne, wymagania produktowe, wymaganie, zarządzanie wymaganiami, zobowiązanie

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

1.1 Pojęcie wymagania

1.1.1 Pojęcie problemu, rozwiązania i produktu

LO-1.1.1 Wyjaśnij pojęcia problemu, rozwiązania i produktu (K2)

1.1.2 Definicja i klasyfikacja wymagań

LO-1.1.2 Wyjaśnij pojęcie wymagań oraz ich klasyfikacji (np. wymagania produktowe a ograniczenia) (K2)

LO-1.1.3 Wyjaśnij różnicę pomiędzy różnymi poziomami wymagań (np. wymaganie biznesowe, wymagania danego rozwiązania (systemu) i wymagania danego produktu (komponentu)) oraz ich znaczenie w inżynierii wymagań (K2)

1.1.3 Typowe atrybuty wymagań

LO-1.1.4 Wyjaśnij pojęcie typowych atrybutów wymagania (np. zobowiązanie, priorytet i krytyczność) (K2)

1.1.4 Jakość wymagań

LO-1.1.5 Zrozum typowe problemy dotyczące wymagań oraz wyjaśnij rolę walidacji, weryfikacji i kryteriów jakości w zwiększaniu jakości wymagań (K2)

1.1.5 Inżynieria Wymagań, zarządzanie wymaganiami i opracowywanie wymagań

LO-1.1.6 Zrozum potrzebę spojrzenia procesowego na inżynierię wymagań oraz jej wpływ, w celu uniknięcia błędów w Inżynierii Wymagań (K2)

1.2 Standardy i normy

LO-1.2.1 Wyjaśnij możliwe korzyści płynące z wdrożenia standardów dla Inżynierii Wymagań (K2)

1.1 Pojęcie wymagania

90 minut

By zrozumieć dyscyplinę Inżynierii Wymagań oraz jej główne wejścia i wyjścia, niezwykle istotnym jest zarysowanie jej kontekstu oraz wprowadzenie najważniejszych pojęć oraz definicji.

Pełną listę pojęć oraz definicji można znaleźć w “Słowniku terminów stosowanych w Inżynierii Wymagań”.

Celem niniejszego rozdziału jest podanie podstawowych koncepcji Inżynierii Wymagań oraz sposobu, w jaki odnoszą się one do samych wymagań.

1.1.1 Koncepcja problemu, rozwiązania oraz produktu

Głównymi wejściami do Inżynierii Wymagań są problemy, potrzeby lub cele biznesowe. Problem biznesowy jest opisem tego, co klient chciałby zrobić w celu realizacji lub usprawnienia procesów biznesowych. Zatem problem opisuje lub też pomaga opisać potrzeby biznesowe klienta.

Jednym z głównym celów Inżynierii Wymagań jest zdefiniowanie rozwiązań biznesowych dla danych problemów, potrzeb lub celów biznesowych. Rozwiązanie stanowi odpowiedź na potrzeby klienta. Potrzeby te zaś są najczęściej wyrażone w formie wymagań oraz stanowią wejście dla Inżynierii Wymagań. Rozwiązaniem może być system lub moduł danego system, nowa lub zmieniona funkcjonalność, zmiana w konfiguracji, usprawnienie procesu etc. Ogólnie rzecz ujmując rozwiązanie jest udoskonaleniem wymagania wysokopoziomowego.

Wyjście realizacji procesu wytwarzania zwane jest produktem.

Dla firm szkoleniowych: wyjaśnij koncepcję problemu, rozwiązania oraz produktu na przykładach zaczerpniętych z prawdziwych projektów tworzenia bądź utrzymywania produktu.

1.1.2 Definicja i klasyfikacja wymagań

Zgodnie z IEEE 610, wymaganie można zdefiniować jako:

- 1) Warunek lub zdolność potrzebną interesariuszowi do rozwiązania danego problemu lub osiągnięcia celu.
- 2) Warunek, który musi zostać spełniony lub zdolność, którą musi posiadać dany system lub moduł systemu, by móc wypełnić dany kontrakt, standard, specyfikację lub inne formalnie nałożone dokumenty.

3) Udokumentowaną reprezentacją warunku lub zdolności opisanych powyżej w punktach (1) i (2)

Jednym z głównych celów wymagań jest wyrażenie potrzeb i oczekiwań klienta wobec planowanego rozwiązania. Wymagania służą również jako podstawa do oszacowania, planowania, wykonania oraz monitorowania czynności projektowych, dlatego też często są one częścią umów, zamówień czy kontraktów. Wymagania nie tylko definiują co ma zostać zrobione, ale przede wszystkim ustalają granice, zakres dostawy, plan dostawy oraz inne usługi umowne.

Ogólnie wymagania mogą zostać zaklasyfikowane jako wymagania i ograniczenia produktu.

Wymagania produktowe opisują właściwości, aspekty, cechy oraz usługi wymagane od danego produktu, a także zawierają wymagania funkcjonalne i niefunkcjonalne. Wymagania funkcjonalne opisują funkcjonowanie (zachowanie) produktu, podczas gdy wymagania niefunkcjonalne opisują tzw. atrybuty jakościowe (czasem nazywane celami jakościowymi).

Różnica pomiędzy wymaganiami funkcjonalnymi i niefunkcjonalnymi może być wyrażona za pomocą poniższych sformułowań:

- Wymagania funkcjonalne opisują **co** dane rozwiązanie robi.
- Wymagania niefunkcjonalne opisują **jak** dane rozwiązanie wykonuje swoje funkcje.

Ograniczenia są ograniczeniami nałożonymi na proces inżynierii, operacje systemu lub jego cykl życia. Istnieją dwa typy ograniczeń: biznesowe i techniczne:

- Ograniczenia biznesowe określają elastyczność projektu dla implementacji wymaganego rozwiązania (np. ograniczenia finansowe, czasowe, dostępności zasobów, umiejętności zespołu projektowego oraz ograniczenia prawne występujące w danej dziedzinie, specyficznych standardów oraz regulacji prawnych, a także inne ograniczenia danej organizacji).
- Ograniczenia techniczne to wszystkie ograniczenia związane z architekturą danego rozwiązania (np. platformy sprzętowe lub oprogramowanie, język programowania lub technologia, rozmiar bazy danych, wykorzystanie zasobów, rozmiar wiadomości oraz czas, rozmiar oprogramowania, maksymalna ilość i wielkość plików, archiwów oraz innych elementów danych).

W przypadku organizacji pracujących z danym produktem wymaganie może być zewnętrzne lub wewnętrzne – w zależności od miejsca pochodzenia. Niektóre mogą pochodzić od klientów, inne zaś z samej organizacji zajmującej się produktem. Niemniej jednak wymagania pochodzące z różnych źródeł powinny być analizowane razem.

Wymagania mogą być sklasyfikowane w różne typy, co odzwierciedla fakt, iż różne wymagania dotyczą różnych aspektów produktu bądź procesu jego tworzenia. Wymagania mogą zostać sklasyfikowane w poziomach abstrakcji reprezentujących różne poziomy szczegółowości danego wymagania od wysokopoziomowych potrzeb biznesowych do szczegółowych wymagań związanych

Poziom podstawowy

z oprogramowaniem lub funkcjami systemu, a nawet do elementów projektu rozwiązania (np. prototypy ekranów, klawiatury urządzeń mobilnych etc.).

Niektórymi z typowo używanych poziomów abstrakcji wymagań są:

- Wymagania biznesowe
 - Wysokopoziomowe wymaganie definiujące, co biznes chce osiągnąć, ale bez określenia, w jaki sposób ma to zostać wdrożone. Wymagania tego typu wyrażają potrzeby, pragnienia oraz oczekiwania klienta. Często nazywa się je wysokopoziomowymi wymaganiami biznesowymi bądź po prostu wymaganiami klienta.
- Wymagania rozwiązania/systemu
 - Przetworzenie (udoskonalenie/uszczegółowienie) wymagań klienta opisujących dane rozwiązanie jako jedną z możliwych ścieżek spełnienia jego wymagań. Rozwiązanie może zawierać wymagania niezwiązane z IT (Technologią Informacyjną), jeśli tyczy się ono np. zmian w procesach lub rolach, bądź też w samej organizacji. Wymagania rozwiązania/systemu są wyrażeniem wymagań klienta używając bardziej technicznej terminologii, która może zostać wykorzystana przy podejmowaniu decyzji podczas projektowania.
- Wymagania produktu/modułu
 - Funkcje i charakterystyki rozwiązania. Kompletna specyfikacja modułu danego produktu, zawierająca dopasowanie, formę, funkcje, wydajność etc. Ten poziom abstrakcji jest często pomijany, gdy uważany jest za część projektowania rozwiązania.

Podczas pracy z wymaganiami na różnych poziomach, niezwykle ważne jest zdefiniowanie i utrzymanie śledzenia (sekcja 4.3 *Śledzenie wymagań*). Śledzenie jest powiązaniem pomiędzy artefaktami produktu na różnych poziomach, np. pomiędzy wymaganiami biznesowymi a wymaganiami rozwiązania lub pomiędzy wymaganiami a powiązаныmi z nimi przypadkami testowymi.

Dla organizacji szkoleniowych: podaj przykłady wymagań różnego typu oraz na różnych poziomach abstrakcji.

1.1.3 Typowe atrybuty wymagań

Do najważniejszych atrybutów wymagań wysokopoziomowych należą:

- Zobowiązanie
- Priorytet

- Krytyczność

Zobowiązanie jest stopniem obligatoryjności spełnienia wymagania. Zazwyczaj definiuje się je za pomocą słów kluczowych przypisanych wymaganiom wysokopoziomowym (“powinien”, “może”, “potrafi”) [zgodnie z ISO/IEC/IEEE 29148: 2011, poprzednio IEEE 830]. W niektórych przypadkach używa się słowa „nie może”, by wyrazić pewien aspekt rozwiązania, którego nie wolno mu wykonać. Słowa kluczowe “powinien”, “może”, “potrafi” zazwyczaj odnoszą się do wymagań biznesowych przed sporządzeniem umowy. Po uzgodnieniu i nakreśleniu podstaw wymagań, słowa kluczowe powinny ustalać dany poziom obligatoryjności w bardziej precyzyjnych słowach: „będzie” lub „zrobi”.

Poziom zobowiązania wymagań może być również wyrażony za pomocą notacji MoSCoW (**M**ust have: “Musiał mieć”, **S**hould have: “Powinien mieć”, **C**ould have: “Mógłby mieć”, **W**ill not have this time but will have in the future: “Nie będzie mieć teraz ale będzie mieć w przyszłości”).

Gdy dostawca rozwiązania i klient osiągną porozumienie, zobowiązania wobec wymagań uzyskiwane są od uczestników projektu.

Wymagania mogą ewoluować w trakcie projektu. Podczas ewolucji wymagań uzyskiwanie od uczestników projektu zobowiązań dla zmienionych wymagań zapewnia zobowiązania wobec zmian w planach projektu, czynnościach z nim związanych oraz pracą produktu.

Jedną z najistotniejszych implikacji zobowiązania jest odpowiedzialność za produkt końcowy. Może pojawić się również odpowiedzialność prawna za produkty końcowe. Odpowiedzialność prawna jest zazwyczaj związana ze specyficznym wymaganiem (np. wymagania środowiskowe dla elektrowni nuklearnej lub wymaganie dotyczące bezpieczeństwa dla samolotu), które musi zostać spełnione w dostarczonej produkcie. Odpowiedzialności mogą również odnosić się do defektów znalezionych w produkcie. Zobowiązanie jest jednym ze sposobów wyrażenia spełnienia odpowiedzialności prawnych.

Odpowiedzialność prawna powinna zostać zdefiniowana w kontrakcie pomiędzy dostawcą a klientem. Niektóre firmy mogą być dodatkowo zobowiązane do spełnienia wymagań umownych bądź prawnych lub standardów specyficznych dla danej gałęzi przemysłu.

Kolejny atrybut wymagania – priorytet – określa ważność/pilność danego wymagania. Zgodnie ze SWEBOK, im wyższy priorytet, tym spełnienie danego wymagania jest ważniejsze dla osiągnięcia całościowych celów produktu. Zazwyczaj priorytet klasyfikowany jest na z góry ustalonej skali tj. obowiązkowy, bardzo pożądany, pożądany, lub opcjonalny. Priorytet musi być często odpowiednio zrównoważony w stosunku do kosztów tworzenia i implementacji.

Krytyczność wymagania jest wynikiem oceny ryzyka szkód, jakie wystąpiłyby, jeśli dane wymaganie nie zostałoby spełnione. Krytyczność wyrażona jest w poziomach; im poziom jest wyższy, tym poważniejsze są konsekwencje w przypadku usterki w danej funkcjonalności.

W niektórych przypadkach atrybuty wymagań można przedstawić za pomocą modelu Kano. Model ten skupia się na rozróżnieniu cech produktu w następujący sposób:

Poziom podstawowy

- Atrybuty progowe lub podstawowe – są to cechy, które dany produkt musi posiadać, by móc spełnić potrzeby klienta.
- Atrybuty operacyjne – umiejętność, wiedza, zdolność lub charakterystyka zachowania związana ze sposobem, w jaki produkt dostarczy swe zdolności klientowi. Atrybuty operacyjne zwiększają satysfakcję klienta.
- Atrybuty emocjonalne – atrybuty nieprzewidziane przez klienta (czasem zwane “nieznanymi potrzebami”), które zostają odkryte przez dostawcę. Zwiększają one satysfakcję klienta i mogą dostarczyć znaczącej przewagi nad konkurencją.

Model Kano może zostać użyty w celu określenia zarówno zobowiązania, jak i priorytetu.

W dodatku do atrybutów opisanych powyżej, wymagania mogą zostać skategoryzowane wg następujących cech [ISO/IEC/IEEE 29148:2011, poprzednio IEEE 1233]:

- Identyfikacja
- Wykonalność
- Ryzyko
- Źródło
- Typ

Dla firm szkoleniowych: wyjaśnij typowe atrybuty wymagania i podaj kilka przykładów demonstrujących ich znaczenie.

1.1.4 Jakość wymagań

Inżynieria Wymagań jest jednym z ważnych czynników składających się na sukces każdego projektu. Wymagania są podstawą do dalszej pracy, dlatego też niezmiernie ważne jest zapewnienie możliwe jak najlepszej jakości wymagań, a także pozostałych produktów procesu inżynierii wymagań. Poniżej znajduje się lista najważniejszych problemów, które należy uwzględnić podczas pracy z wymaganiami:

- Niejasne cele biznesowe dla wymagań lub samego projektu
- Problemy komunikacyjne (często mające swoje źródło w barierach językowych lub wynikające z braku wiedzy, w tym z braku wiedzy dziedzinowej dla danego rozwiązania biznesowego), niejasny przekaz (często mający swoje źródło w niewystarczającej lub niemierzalnej definicji wymagania, bądź braku wspólnego słownika terminów)
- Zmienność wymagań (często spowodowana niejasnymi celami biznesowymi dla wymagań lub samego projektu)
- Niska jakość wymagań (patrz kryteria jakości wymagań opisane poniżej)

Poziom podstawowy

- Poziłacanie (dodawanie dodatkowych cech, które w rzeczywistości są zbędne)
- Niewystarczające zaangażowanie interesariuszy
- Brak interesariuszy (często spowodowane słabym procesem Inżynierii Wymagań)
- Niedokładne planowanie (często spowodowane niejasnymi lub brakującymi celami biznesowymi dla wymagań lub samego projektu)
- Niewystarczające wymagania lub specyfikacje rozwiązania (ich wynikiem są luki bądź brakujące lub fragmentaryczne wymagania)

Niektóre z problemów wymienionych powyżej mogą zostać rozwiązane w dość prosty sposób poprzez zastosowanie kryteriów jakości dla wymagań. Zgodnie z [Wiegers], powszechne kryteria jakości dla wymagań stwierdzają, iż każde wymaganie musi być:

- Poprawne – wymaganie musi precyzyjnie opisywać cechy, które mają być dostarczone. Punktem odniesienia dla ewaluacji poprawności jest źródło wymagania (np. klienci lub wysokopoziomowe wymaganie systemowe).
- Wykonalne – wymaganie musi być możliwe do zaimplementowania w obrębie znanych możliwości i/lub ograniczeń systemu lub środowiska.
- Konieczne – wymaganie powinno dokumentować to, co klient (lub interesariusze) rzeczywiście potrzebują oraz co jest niezbędne, by spełnić zewnętrzne wymagania lub interfejs lub określony standard.
- Priorytetyzowalne – wymaganie powinno mieć przypisany priorytet wskazujący, jak niezbędne jest ono dla wydania danego produktu.
- Jednoznaczne – wymaganie powinno dać się zinterpretować tylko w jeden sposób. Różne osoby zapoznające się z nim powinny mieć tę samą interpretację oraz rozumienie wymagania.
- Weryfikowalne – wymaganie powinno być możliwe do zweryfikowania, czy zostało poprawnie zaimplementowane.
- Pojedyncze – wymaganie nie zawiera wielu wymagań; powyższe stwierdzenie implikuje dostateczną szczegółowość, z jaką dane wymaganie powinno zostać wyspecyfikowane.
- Niezależne od projektu (implementacji) (abstrakcyjne) – wymaganie powinno określać “co ma zostać zrobione”, nie “jak coś ma zostać zrobione”. Zawartość wymagania nie powinna sugerować szczegółów implementacji.

Kryteria jakości mogą zostać zaimplementowane nie tylko do pojedynczego wymagania, ale także w odniesieniu do specyfikacji wymagań.

Główne kryteria jakości dla specyfikacji stwierdzają, iż specyfikacja musi być:

Poziom podstawowy

- Kompletna – specyfikacja wymagań nie powinna pominąć żadnego wymagania ani niezbędnej informacji. Kompletność jest również określana jako cecha pożądana poszczególnego wymagania oraz poziomu szczegółowości.
- Spójna – wymagania opisane w specyfikacji nie mogą być w konflikcie z innymi wymaganiami produktu lub z wysokopoziomowymi (systemowymi lub biznesowymi) wymaganiami.
- Modyfikowalna – specyfikacja musi pozwalać na wprowadzanie zmian w wymaganiach. Musi być utrzymywana historia zmian dokonanych w każdym z wymagań.
- Identyfikowalna (możliwa do śledzenia) – każde wymaganie powinno posiadać możliwość dotarcia do jego źródła (np. wysokopoziomowe wymaganie systemowe, przypadek użycia lub stwierdzenie klienta) oraz powiązane z nim artefakty implementacji (np. elementy projektowe, kod źródłowy, przypadki testowe).

Dla firm szkoleniowych: wyjaśnij typowe kryteria jakości oraz podaj kilka przykładów wymagań spełniających i niespełniających te kryteria.

Zapewnianie odpowiedniego poziomu jakości wymagań lub innych produktów procesu Inżynierii Wymagań może być wspierane przez walidację i weryfikację.

Zgodnie z CMMI czynności walidacyjne wykazują, iż dany produkt lub część produktu spełnia swoją rolę, gdy zostaje umieszczony w środowisku swego przeznaczenia. A więc “wiesz, czy stworzyłeś odpowiedni produkt”. Klienci często mają problem z odpowiednim opisaniem produktów lub wymagań, a walidacja pomaga im zrozumieć, co jest potrzebne (przy użyciu narzędzi takich jak scenariusze, przypadki użycia, prototypowanie etc.).

Walidacja zazwyczaj przeprowadzana jest ze wsparciem klienta lub interesariuszy u klienta, a jej celem jest potwierdzenie, że dane wymagania lub specyfikacja wymagań w dokładny sposób opisuje to, czego dany klient potrzebuje.

Zgodnie z CMMI weryfikacja dostarcza punktów kontrolnych, w których wybrane produkty pośrednie lub rezultaty są weryfikowane w celu potwierdzenia spełnienia przez nie wymagań. Czynności weryfikacyjne skupiają się na przyrostowym potwierdzaniu implementacji wymagań; pozwalają na wczesne oraz ciągłe potwierdzanie poprawności budowane produktu.

Weryfikacja jest porównaniem produktu pośredniego (nieskończonego/w trakcie tworzenia) z jego specyfikacją. W ten sposób zostaje ustalone, czy produkt jest tworzony poprawnie oraz czy stworzone w poprzedniej fazie specyfikacje zostały spełnione. Najbardziej powszechnymi technikami weryfikacji są przeglądy, analiza statyczna oraz testowanie dynamiczne. Do walidacji zaś używane są przeglądy oraz testowanie dynamiczne.

Różnica między walidacją a weryfikacją może zostać ujęte w poniższy sposób:

- Weryfikacja – czy poprawnie stworzyliśmy nasz produkt?

- Walidacja – czy stworzyliśmy poprawny produkt?

1.1.5 Inżynieria Wymagań, zarządzanie wymaganiami, opracowywanie wymagań

Inżynierię Wymagań (IW) można zdefiniować jako podkategorię inżynierii oprogramowania skupioną na określaniu, tworzeniu oraz utrzymywaniu wymagań sprzętowych oraz oprogramowania [SWEBOOK, Sommerville]. Zgodnie z CMMI inżynieria wymagań zawiera zarządzanie wymaganiami oraz opracowywanie wymagań.

Celem zarządzania wymaganiami jest zarządzanie uzgodnionymi zestawami wymagań danego rozwiązania, by zapewnić zgodność między nimi, planami projektowymi a tworzonymi produktami.

Zarządzanie wymaganiami składa się z funkcjonalnych ram pracy Inżynierii Wymagań oraz z procesów wspierających opracowywanie wymagań. Kolejną rolą zarządzania wymaganiami jest ustalenie oraz dostarczenie interfejsów do innych procesów tworzenia i zarządzania, zapewnione poprzez powiązania Inżynierii Wymagań tj. zarządzanie projektem, zarządzanie ryzykiem, projektowanie, zarządzanie konfiguracją, zarządzanie zmianą oraz zapewnianie jakości.

Opracowywanie wymagań jest zestawem czynności, zadań, technik oraz narzędzi służących do identyfikacji, analizy, dokumentacji i walidacji wymagań na różnych poziomach abstrakcji. Zawiera ono proces przekształcania potrzeb w wymagania, jak również tworzenie (przetwarzanie wysokopoziomowych wymagań) rozwiązania dla tych wymagań.

Dobry proces opracowywania wymagań jest jednym z głównych czynników gwarantujących sukces danego projektu. Dobrze zebrane, przeanalizowane, udokumentowane oraz zarządzane wymagania obniżają ryzyka projektowe poprzez tworzenie jasnych i zrozumiałych podstaw do projektowania rozwiązania. Jednakże, znaczenie dobrych wymagań i procesu Inżynierii Wymagań jest wciąż nie do końca pojmowane lub po prostu zaniedbywane.

Przyczyny zaniedbywania inżynierii wymagań mogą być następujące:

- Presja czasu i kosztów
- Wyłączne skupienie na dostarczeniu szybkich rezultatów
- Rozważanie jedynie wymagań funkcjonalnych
- Brak zrozumienia ważności Inżynierii Wymagań dla sukcesu projektu

Konsekwencje Inżynierii Wymagań niskiej jakości mogą być następujące:

- Niskiej jakości wymagania
- Częste zmiany wymagań podczas tworzenia produktu

Poziom podstawowy

- Wymagania niespełniające kryteriów akceptacji i/lub niedostarczające wartości dla klienta lub innych interesariuszy
- Brakujące wymagania lub ograniczenia

1.2 Standardy oraz normy	20 minut
---------------------------------	-----------------

Uwaga: zapamiętywanie numerów identyfikacyjnych poszczególnych standardów nie jest wymagane.

Istnieje wiele standardów oraz norm procesowych użytecznych w Inżynierii Wymagań. Niektóre z nich podają modele procesów związanych z tworzeniem rozwiązań, inne dostarczają wytyczne do pisanie różnych rodzajów specyfikacji wymagań lub klasyfikacji wymagań.

Jednym z głównych celów standardów jest zapewnienie krajowej oraz międzynarodowej zgodności w zakresie jakości produktów i procesów. Standardy normalizują metody i rezultaty wytwarzania, dostarczają wspólnej terminologii oraz ułatwiają porozumiewanie się pomiędzy biznesem a technologią.

Poniżej wyszczególnione zostały najistotniejsze wg REQB standardy oraz normy.

Standardy:

- ISO 9000. Rodzina standardów ISO 9000 dostarcza wymagań dla systemów zarządzania jakością oraz definiuje koncepcje i podstawy systemu zarządzania jakością (QMS).
- ISO/IEC 25000 (poprzednio ISO 9126). Definiuje model jakości oparty na sześciu kategoriach (funkcjonalność, niezawodność, użyteczność, wydajność, utrzymywalność, przenaszalność) które mogą stanowić podstawę dla pozyskiwania wymagań, specyfikacji wymagań a także walidacji oraz weryfikacji wymagań.
- IEEE Standard 610.12-1990. IEEE Standard Glossary of Software Engineering Terminology (Norma opisująca definicje oraz terminy używane w inżynierii oprogramowania)
- IEEE Standard 830-1998. IEEE Recommended Practice for Software Requirements Specifications (Charakteryzuje treść i cechy dobrej dokumentacji wymagań).
- IEEE Standard 1233-1998. IEEE Guide for Developing System. Requirements Specifications (Przewodnik dla opracowania i rozwoju specyfikacji wymagań systemowych).
- IEEE Standard 1362-1998. IEEE Guide for Information Technology-System Definition – Concept of Operations (ConOps) Document (Dokument opisujący charakterystykę proponowanego system z punktu widzenia przyszłego użytkownika)

Uwaga: Nazwy standardów podane są po angielsku, bo – w większości wypadków – nie ma ich tłumaczeń na język polski.

Uwaga: IEEE 830, IEEE 1233 oraz IEEE 1362 zostały zastąpione przez ISO/IEC/IEEE 29148:2011

- SWEBOK - The Guide to the Software Engineering Body of Knowledge (znane jako ISO Technical Report 19759). SWEBOK opisuje ogólnie zaakceptowaną wiedzę na temat

Poziom podstawowy

inżynierii oprogramowania. Posiada 10 obszarów wiedzy podsumowujących podstawowe koncepcje, a także zawiera listę referencji wskazującą dalsze bardziej szczegółowe pozycje. Jeden z obszarów wiedzy poświęcono Inżynierii Wymagań.

- SEBOK – The Guide to the System Engineering Body of Knowledge.

Normy procesowe dla wytwarzania:

- ISO 12207. Systems and software engineering — Software life cycle processes (Standard dla Procesów w Cyklu Życia Oprogramowania)
- ISO 15288. Systems and software engineering -- System life cycle processes (Procesy Cyklu Życia Oprogramowania)

Obie normy mogą zostać wykorzystane do wspierania organizacji w procesie tworzenia rozwiązania.

Normy procesowe dla oceny i usprawniania procesów:

- ISO 15504 – Information technology (Technologia Informacyjna) — ocena procesu, znane także jako SPICE (Software Process Improvement and Capability Determination).
- CMMI. Capability Maturity Model Integration

Obie normy mogą zostać wykorzystane do ewaluacji i usprawniania procesów. Dodatkowo definiują one kluczowe obszary inżynierii wymagań.

Dla firm szkoleniowych: wyjaśnij, dlaczego standardy są ważne podczas definiowania procesu inżynierii wymagań.

2 Kontekst Inżynierii Wymagań

40 minut

Pojęcia

Analiza Biznesowa

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

2.1 Inżynieria Wymagań w kontekście

- LO-2.1.1 Wyjaśnij kontekst Inżynierii Wymagań oraz sposób, w jaki Inżynieria Wymagań jest częścią cyklu życia tworzenia rozwiązania oraz utrzymania (K2)

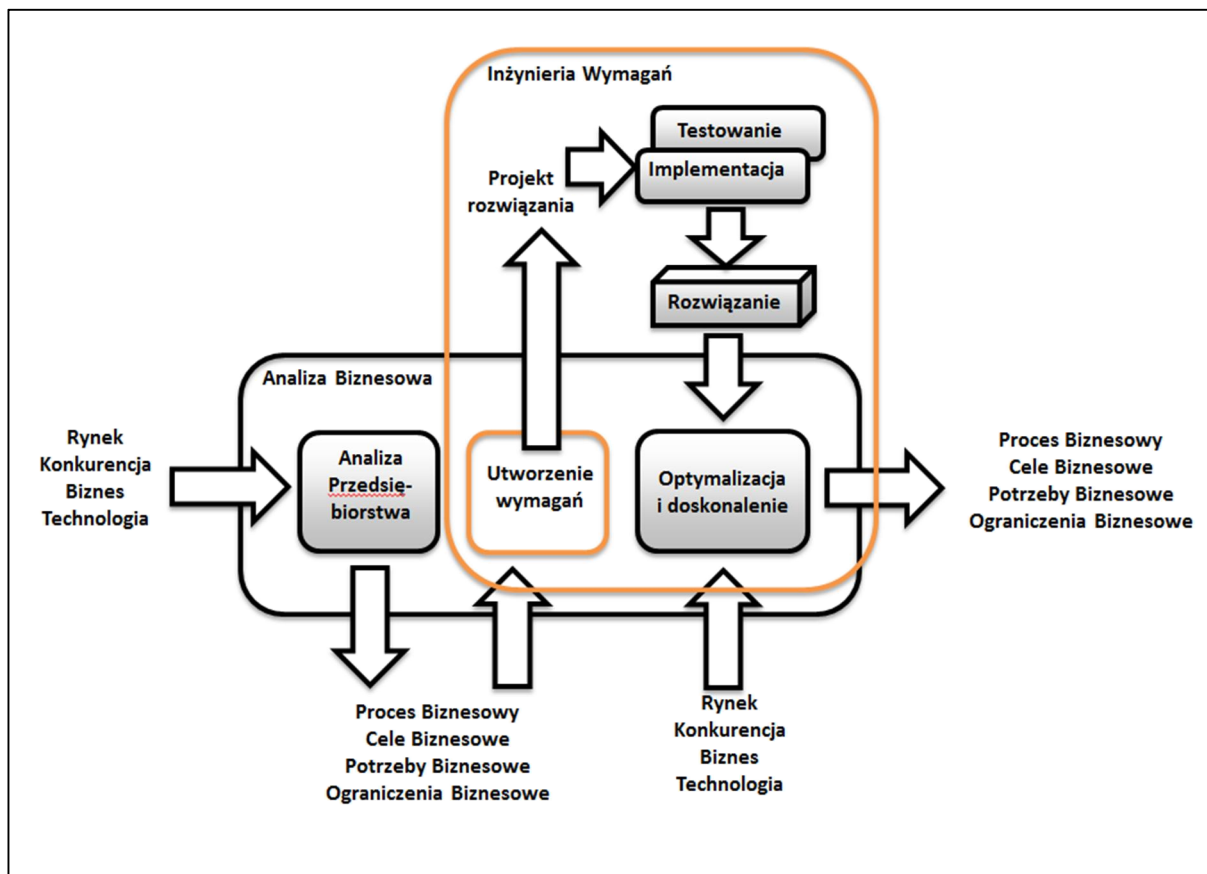
2.2 Powiązane procesy

- LO-2.2.1 Wyjaśnij związki oraz interakcje pomiędzy Inżynierią Wymagań a innymi dyscyplinami podczas tworzenia projektu (K2)

2.1 Inżynieria Wymagań w kontekście

20 minut

Inżynieria wymagań nie istnieje w izolacji. Jest połączona z innymi dyscyplinami (Rysunek 1), ponadto powinna być włączona w cały proces tworzenia rozwiązania.



Rysunek 1 Kontekst Inżynierii Wymagań

Punktem startowym Inżynierii Wymagań jest analiza biznesowa. W celu zaproponowania możliwie najlepszego rozwiązania dla danego problemu biznesowego kluczowym jest jego poprawne zdefiniowanie [IQBBA].

Analiza biznesowa jest dziedziną dostarczającą zestawu czynności, narzędzi oraz metod mających na celu ustalenie potrzeb, problemów oraz celów biznesowych, a także określenie odpowiednich rozwiązań mogących zaspokoić w/w potrzeby lub rozwiązać dane problemy. Te rozwiązania biznesowe mogą zawierać tworzenie oprogramowania lub modułów oprogramowania, rozszerzeń istniejącego oprogramowania, usprawnień procesów biznesowych, zmian organizacyjnych etc. Analiza biznesowa określa rozwiązanie dla problemu biznesowego, co staje się punktem wejścia dla dalszej Inżynierii Wymagań. Cele oraz potrzeby biznesowe określone podczas analizy biznesowej wchodzą w skład wymagań dla danego rozwiązania. Stąd też Inżynieria Wymagań może

Poziom podstawowy

być uznawana za kontynuację lub część procesu analizy biznesowej. Rezultatem Inżynierii Wymagań jest projekt rozwiązania, z czasem zaimplementowany, przetestowany i w końcu dostarczony klientowi.

Zaimplementowane rozwiązanie rzadko pozostaje niezmiennie przez dłuższy okres czasu. Zarówno rynek, jak i organizacje ulegają zmianom, technologia ewoluuje, a organizacje nawiązują nowe kontakty. Niemal wszystkie systemy zostaną zmienione i doczekają się swoich nowych wersji wydania. Jest niezmiernie istotne, by analiza biznesowa i Inżynieria Wymagań zajmowały się również czynnościami związanymi z utrzymaniem. Dlatego też analiza biznesowa i Inżynieria Wymagań trwają przez cały cykl życia danego rozwiązania. Po dostarczeniu rozwiązania analiza biznesowa koncentruje się na usprawnieniach zwiększających rynek zbytu, poprawieniu wydajności lub uzyskaniu przewagi nad konkurencją. W związku z tym z pewnością pojawią się nowe potrzeby biznesowe. Owe potrzeby zostają przetworzone w procesie Inżynierii Wymagań by móc zaproponować nowe, ulepszone rozwiązania lub rozszerzyć istniejące dzięki nowym lepszym cechom.

Dla firm szkoleniowych: wyjaśnij kontekst Inżynierii Wymagań na przykładach z życia wziętych. Podaj przykłady związków pomiędzy Inżynierią Wymagań a analizą biznesową (w zakresie różnych produktów oraz czynności).

2.2 Procesy powiązane**20 minut**

Inżynieria Wymagań, a w szczególności jej część zwana zarządzaniem wymaganiami, funkcjonuje w szerszym kontekście i jest silnie związana z następującymi procesami:

- Analiza biznesowa – jak zostało wyjaśnione w poprzedniej sekcji, analiza biznesowa i Inżynieria Wymagań są ze sobą silnie powiązane.
- Zarządzanie projektem – kierownik produktu powinien zawrzeć w planie zarządzania projektem specyficzne aktywności związane z Inżynierią Wymagań, a także przeznaczyć odpowiednią ilość czasów oraz zasobów, by zapewnić ich prawidłowe przeprowadzenie. Inżynierowie wymagań mogą stworzyć plan wymagań, w którym zawarte zostaną szczegółowo opisane zadania wymagane do ukończenia specyfikacji wymagań.
- Zarządzanie ryzykiem – niektóre wymagania mogą wprowadzić ryzyka projektowe lub produktowe, które powinny być zarządzane zgodnie z procesem zarządzania ryzykiem.
- Analiza i projektowanie – wymagania są obowiązkowym wejściem dla analizy i czynności związanych z projektowaniem. Wymagane jest śledzenie wymagań oraz elementów analizy i projektowania.
- Zarządzanie konfiguracją i zmianą – konfiguracja wymagań powinna być zarządzana zgodnie z procesem zarządzania konfiguracją. Wymaganie może być powiązane z elementem konfiguracji.
- Testowanie – testowanie dostarcza informacji zwrotnych o różnicach pomiędzy podanymi wymaganiami a stworzonymi funkcjonalnymi bądź niefunkcjonalnymi atrybutami produktu. W celu dostarczenia kompletnej przejrzystości pokrycia testami wymagane jest śledzenie wymagań, testów oraz innych artefaktów. W zależności od powiązanych elementów testów, dane wymaganie może posiadać status pokazujący czy zostało ono zaimplementowane, czy też nie.
- Zarządzanie wydaniem – punktem startowym zarządzania wydaniem jest rezultat zarządzania wymaganiami. Dla każdego rozwiązania lub wydanego produktu musi zostać zapewniona identyfikacja zaimplementowanych wymagań dla danego wydania.

W zależności od modeli tworzenia oraz różnych produktów Inżynieria Wymagań może być połączona również z innymi procesami.

Dla firm szkoleniowych: podaj przykłady związków pomiędzy inżynierią wymagań a innymi dyscyplinami (w znaczeniu różnych produktów oraz czynności).

3 Proces Inżynierii Wymagań	80 minut
------------------------------------	-----------------

Pojęcia

Dostawca, interesariusz, inżynier wymagań, klient, menedżer wymagań, ogólny proces Inżynierii Wymagań, opracowywanie wymagań

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

3.2 Ogólny proces Inżynierii Wymagań

LO-3.2.1 Wyjaśnij cele Inżynierii Wymagań oraz typowe czynności procesu Inżynierii Wymagań (K2)

3.3 Role i odpowiedzialności

LO-3.3.1 Wyjaśnij role występujące podczas opracowywania wymagań oraz zarządzania wymaganiami (K2)

LO-3.3.2 Podaj podstawowe role w Inżynierii Wymagań oraz wyjaśnij ich wpływ na proces inżynierii wymagań (K1)

LO-3.3.3 Podaj główne umiejętności oraz kompetencje wymagane od osoby zaangażowanej w Inżynierię Wymagań (K1)

3.1 Wprowadzenie do procesu Inżynierii Wymagań	40 minut
---	-----------------

Struktura procesu Inżynierii Wymagań zależy od różnych czynników, np. dojrzałość i kultura organizacji, używany model tworzenia etc. REQB używa poniższej kategoryzacji:

- Ogólny proces Inżynierii Wymagań
- Proces Inżynierii Wymagań w modelach i podejściach wytwarzania i utrzymania produktu
- Proces Inżynierii Wymagań w modelach dojrzałości

Ogólny proces Inżynierii Wymagań powinien być punktem wyjścia dla każdej organizacji zaangażowanej w tworzenie rozwiązań, gdyż dostarcza on najistotniejszych procesów dotyczących obsługi wymagań. Ogólny proces inżynierii wymagań może zostać bez problemu zaadaptowany do specyficznych potrzeb danej organizacji, a także używanych przez nią modeli tworzenia i utrzymania produktu. Rezultatem tej adaptacji jest proces Inżynierii Wymagań wstępujący w modelach i podejściach wytwarzania i utrzymania produktu. Dodatkowo poza modelami procesów tworzenia istnieją także modele dojrzałości służące głównie ocenie obecnej dojrzałości danego przedsiębiorstwa oraz wprowadzaniu usprawnień służących podniesieniu wydajności tej organizacji. Pozwala to na zdefiniowanie kolejnego wariantu ogólnego procesu Inżynierii Wymagań – procesu Inżynierii Wymagań w modelach dojrzałości. Ów wariant procesu inżynierii wymagań został ujęty w programie REQB poziomu zaawansowanego.

Ogólny proces Inżynierii Wymagań powinien zostać zaadaptowany do potrzeb danej organizacji, biorąc pod uwagę rodzaje tworzonych rozwiązań, dziedzinę biznesową, procesy biznesowe, kulturę organizacji, a także poziom kompetencji oraz umiejętności personelu odpowiedzialnego za proces inżynierii wymagań.

Dla firm szkoleniowych: wyjaśnij podejście REQB do procesu Inżynierii Wymagań oraz zademonstruj potrzebę dopasowania ogólnego procesu Inżynierii Wymagań do specyficznych potrzeb.

3.2 Ogólny proces Inżynierii Wymagań**20 minut**

Inżynieria Wymagań jest dyscypliną obejmującą procesy pozyskiwania, strukturyzacji oraz zarządzania wymaganiami. Do specyficznych działań zawartych w ogólnym procesie Inżynierii Wymagań należą:

- Identyfikacja wymagań
- Analiza wymagań
- Specyfikacja wymagań
- Walidacja i weryfikacja wymagań
- Śledzenie wymagań
- Zarządzanie konfiguracją i zmianą
- Zapewnienie jakości

Proces Inżynierii Wymagań jest ustrukturyzowanym zestawem w/w czynności. Aktywności te zostały zorganizowane w procesach zarządzania wymaganiami lub opracowywania wymagań, w zależności od celu oraz fazy tworzenia danego rozwiązania. Kompletny opis procesu powinien zawierać również powiązania z pozostałymi pokrewnymi dziedzinami i obszarami (np. z analizą biznesową lub testowaniem), wejściami oraz wyjściami poszczególnych czynności, informacjami na temat odpowiedzialności za konkretną czynność oraz jej rezultaty, wymaganymi kompetencjami oraz narzędziami wspierającymi aktywności (więcej szczegółów w rozdziale 7 *Wsparcie narzędziowe*).

Główne czynności, wejścia i wyjścia dla konkretnych procesów Inżynierii Wymagań przedstawia Tabela 2.

Poziom podstawowy

Proces	Wejście	Główne czynności	Wyjście
Opracowywanie wymagań			
Pozyskiwanie wymagań	Potrzeby biznesowe Wstępne określenie zakresu Interesariusze	Zbieranie wymagań od interesariuszy Uszczegóławianie znanych wymagań wysokopoziomowych Wyłączanie niepotrzebnych cech	Wymagania biznesowe Ograniczenia Założenia
Analiza wymagań	Wymagania biznesowe Ograniczenia Założenia Interesariusze Aktywa organizacyjne Techniki modelowania Narzędzia	Opracowywanie (zmiana) wymagań biznesowych w wymagania systemowe/rozwiązania oraz dalej aż do wymagań produktu/modułu. Rozwiązywanie konfliktów pomiędzy wymaganiami Ustalanie granic dla rozwiązania Tworzenie modeli rozwiązania biznesowego	Uzgodnione wymagania Określenie zakresu Model rozwiązania biznesowego
Specyfikacja wymagań	Wymagania Ograniczenia Założenia Interesariusze Aktywa organizacyjne	Dokumentowanie wymagań	Dokument specyfikacji wymagań
Walidacja i weryfikacja wymagań	Wymagania Dokument specyfikacji wymagań	Sprawdzenie jakości wymagań oraz dokumentacji specyfikacji wymagań	Zwalidowane wymagania Zwalidowane dokumenty specyfikacji wymagań Defekty Problemy Zmiany

Poziom podstawowy

Proces	Wejście	Główne czynności	Wyjście
Zarządzanie wymaganiami			
Śledzenie wymagań	Wymagania Interesariusze Aktywa organizacyjne Narzędzia	Ustalenie struktury dla śledzenia Określanie oraz utrzymanie śledzenia wymagań	Macierz śledzenia
Zarządzanie konfiguracją i zmianą	Wymagania Narzędzia Żądania zmiany Plany projektowe Macierz śledzenia	Identyfikacja i zarządzanie elementami konfiguracji Żądania, ustalenie dostępności, planowanie, implementowanie, a także ewaluacja zmian systemu, dokumentacji lub innych produktów projektu Przeprowadzanie analizy ryzyka Przetwarzanie zmian oraz zapewnianie śledzenia zmian	Zatwierdzone wymagania Rezultaty analizy ryzyka Zmiana wymagań oraz dokumentów specyfikacji Plany projektowe
Zapewnienie jakości	Wymagania Dokumenty specyfikacji wymagań Modele rozwiązania Interesariusze Aktywa organizacyjne Standardy Narzędzia Macierz śledzenia Analiza ryzyka	Zapewnianie, że różne procesy i produkty inżynierii wymagań są dobrej jakości Stosowanie odpowiednich standardów Przeprowadzanie przeglądów oraz innych czynności zapewniania jakości	Wymagania Dokumenty specyfikacji wymagań Modele rozwiązania biznesowego Defekty Problemy Zmiany Ulepszenia Procesy i czynności inżynierii wymagań - opisane i udostępnione w formie szkoleń dla pracowników

Tabela 2 Przegląd czynności, wejść oraz wyjść dla procesów Inżynierii Wymagań

Wydajność procesu Inżynierii Wymagań jest różna w zależności od organizacji lub projektu. Istnieją bowiem czynniki, zewnętrzne lub wewnętrzne, posiadające negatywny wpływ na proces Inżynierii Wymagań lub produkty.

Poziom podstawowy

Czynniki wewnętrzne (np. wewnątrz organizacji dostarczającej produkt) zawierają:

- Brak wiedzy na temat dziedziny biznesowej lub użytkownika
- Brak wiedzy na temat technologii wymaganej przez klienta
- Nieefektywne podejścia/metodologie/narzędzia Inżynierii Wymagań
- Niewystarczające doświadczenie, umiejętności i kompetencje personelu

Czynniki zewnętrzne (np. na zewnątrz organizacji dostarczającej produkt) zawierają:

- Brak komunikacji lub nieefektywne podejście do komunikowania wymagań, potrzeb lub oczekiwań
- Niejasne i/lub zmieniające się cele biznesowe skutkujące niestabilnymi wymaganiami
- Niewystarczająca wiedza na temat procesu wytwarzania produktu
- Niewystarczające zaangażowanie użytkowników i/lub interesariuszy

Inżynieria Wymagań zajmuje się różnymi zadaniami, w zależności od swej roli w tworzeniu rozwiązania lub utrzymaniu cyklu życia, a także od ograniczeń oraz charakterystyki planowanego rozwiązania.

Proces Inżynierii Wymagań oraz jej produkty (tj. specyfikacja wymagań lub propozycja rozwiązania biznesowego) powinny dostarczać możliwie najwyższą wartość dla klienta oraz pozostałych interesariuszy. By móc zapewnić, że zaplanowane rozwiązanie w najwyższym stopniu spełni oczekiwania i potrzeby klienta wymagania powinny być tworzone z perspektywy (punktu widzenia) klienta. Poniżej kilka przykładowych metod procesu Inżynierii Wymagań zorientowanych na klienta:

- Analiza i projektowanie zorientowane na klienta
- Podejścia prototypowe
- Używanie demonstracji/pokazów jako środka walidacji poszczególnych przyrostów systemu

Dla firm szkoleniowych: podaj przykładowy przegląd czynności, wejść i wyjść procesu Inżynierii wymagań dla konkretnego obszaru biznesowego lub projektu.

3.3 Role i odpowiedzialności

20 minut

Pomyślny proces Inżynierii Wymagań wymaga jasnego zdefiniowania ról i odpowiedzialności. W ogólności, role można zakwalifikować jako role, które wpływają lub na które wpływa Inżynieria Wymagań i role w Inżynierii Wymagań.

Najważniejsze role, które wpływają i na które wpływa Inżynieria Wymagań to odbiorca (klient) i dostawca. Odbiorca to osoba, grupa lub organizacja zamawiająca rozwiązanie, podczas gdy dostawca (zwany też sprzedającym) dostarcza rozwiązanie.

Klient formułuje swoje potrzeby i dostarcza swoje początkowe potrzeby biznesowe i oczekiwania. Są one dostarczane wraz ofertą/ zamówieniem usługi. Dostawca jest odpowiedzialny za analizę tych potrzeb i na tej podstawie opracowuje wymagania. Głównym celem dostawcy jest dostarczenie rozwiązań spełniających potrzeby klienta.

Role w Inżynierii Wymagań to Zarządzający Wymaganiami i Deweloper Wymagań. Sa oni także nazywani Inżynierami Wymagań.

Duży zakres zadań związanych z Inżynierią Wymagań skutkuje różnorodnością tytułatury dla osób wykonujących zadania z Inżynierii Wymagań. Różne organizacje używają własnych nazw, takich jak Inżynier Wymagań, Zarządzający Wymaganiami, Deweloper Wymagań, Analityk Biznesowy, Analityk Systemowy, Architekt Rozwiązań, Architekt Systemowy, Projektant i inne. Regionalne różnice pomiędzy tymi rolami istnieją i zależą od kultury, przyzwyczajeń i tradycji. Problemem jest to, że dokładna specyfikacja i kompetencje potrzebne do tych ról rzadko są zdefiniowane, jasne i zrozumiałe.

By zapewnić powszechne zrozumienie odpowiedzialności i ról, REQB definiuje role Zarządzającego Wymaganiami i Dewelopera Wymagań. Ta kwalifikacja skutkuje rozróżnieniem zadań wykonywanych w Zarządzaniu Wymaganiami i Rozwijaniu Wymagań (ang. Requirements Development). Zarządzający Wymaganiami to osoba z ogólną odpowiedzialnością za dokumentowanie, analizowanie, śledzenie, priorytetyzację i koordynowanie zgodności wymagań, a następnie sterowanie zmianami i prezentowaniem wymagań odpowiednim interesariuszom. Deweloper Wymagań to osoba głównie zaangażowana w opracowywanie wymagań, ich analizę, dokumentowanie i priorytetyzację.

Zarządzający Wymaganiami i Deweloper Wymagań powinni charakteryzować się następującymi umiejętnościami:

- Kompetencją metodologiczną (tzn. praktyczną wiedzą o Inżynierii Wymagań, procesie, metodach, technikach i narzędziach)
- Wiedzą o źródłach najlepszych praktyk i najważniejszych standardach związanych z Inżynierią Wymagań
- Precyzyjnym, analitycznym i jasnym myśleniem,

Poziom podstawowy

- Umiejętnością działania w sposób strukturalny
- Umiejętnościami moderowania i negocjacji
- Pewnością siebie
- Zdolnością argumentowania i przekonywania
- Umiejętnościami językowymi i komunikacyjnymi
- Odpornością na stres

Kolejna rola związaną z Inżynierią Wymagań to Interesariusz. Interesariusz to grupa lub osoba, na którą wpływa przedsięwzięcie lub jest w jakiś sposób odpowiedzialna za wyniki. Interesariusze to osoby lub organizacje aktywnie zaangażowane w projekt, lub na których interesy mogą wpływać wyniki wykonania lub zakończenia projektu. Mogą to być osoby fizyczne, prawne lub abstrakcyjne.

Często występują konflikty pomiędzy interesariuszami, co skutkuje sprzecznymi wymaganiami. Problem sprecznych wymagań powinien być rozwiązany podczas Analizy Wymagań.

Jest wiele kategorii interesariuszy, w tym:

- Klienci
- Użytkownicy końcowi
- Menadżerowie projektów
- Inżynierowie odpowiedzialni za tworzenie i pielęgnację systemu
- Klienci z organizacji, którzy będą używać systemu
- Organy zewnętrzne (np. regulatorzy prawni)
- Eksperti dziedzinowi
- Dostawcy podobnych rozwiązań
- Organizacje odpowiedzialne za utrzymanie

Zidentyfikowanie wszystkich interesariuszy jest konieczne dla zrozumienia wszystkich punktów widzenia związanych z planowanym rozwiązaniem. Jeżeli jacyś interesariusze zostaną pominięci, może to skutkować pominięciem wymagań lub istotnych ograniczeń wpływających na zakres lub kształt rozwiązania.

Dla firm szkoleniowych: opisz typowego interesariusza, jego punkt widzenia na projekt i wszystko co ma być dostarczone, główne cele oraz wzory komunikacji (np. Właściciel Produktu / Sponsor, Menadżer Projektu, Klient, Inżynier Wymagań, Zespół Zwinny, Scrum Master}.

4 Zarządzanie wymaganiami	160 minut
----------------------------------	------------------

Pojęcia

Analiza ryzyka, analiza wpływu, element konfiguracji, identyfikacja ryzyka, Komitet Kontroli Zmian, kontrola jakości, łagodzenie ryzyka, metryka, plan zarządzania ryzykiem, przegląd, punkt odniesienia, ryzyko, ryzyko produktowe, ryzyko projektowe, śledzenie, śledzenie pionowe, śledzenie poziome, żądanie zmiany, zapewnienie jakości, zarządzanie konfiguracją, zarządzanie ryzykiem, zarządzanie zmianą, zmiana

Cele kształcenia dla Poziomu Podstawowego inżynierii wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

4.2 Zarządzanie projektem i ryzykiem

- LO-4.2.1 Wyjaśnij, dlaczego Inżynieria Wymagań jest ważna w projektach i przedstaw odpowiednie przykłady (K2)
- LO-4.2.2 Podaj błędy, które mogą wystąpić w Inżynierii Wymagań (K1)
- LO-4.2.3 Rozróżnij pomiędzy typowymi ryzykami projektowymi a produktowymi związanymi z wymaganiami (K2)
- LO-4.2.4 Opisz, używając przykładów, jak analiza ryzyka i zarządzanie ryzykiem mogą być wykorzystane w zarządzaniu wymaganiami (włączając w to planowanie wymagań) (K2)

4.3 Śledzenie wymagań

- LO-4.3.1 Zrozum potrzebę śledzenia (K2)
- LO-4.3.2 Rozpoznaj różne rodzaje śledzenia (K1)

4.4 Zarządzanie konfiguracją i zmianą

- LO-4.4.1 Wyjaśnij cechy zarządzania konfiguracją i zarządzania zmianą oraz rolę Komitetu Kontroli Zmian (K2)
- LO-4.4.2 Wyjaśnij pojęcia: element konfiguracji, defekt, żądanie zmiany (K2)

4.5 Zarządzanie jakością

- LO-4.5.1 Przedstaw czynniki, które mają wpływ na jakość procesu i produktu Inżynierii Wymagań (K1)
- LO-4.5.2 Wyjaśnij, w jaki sposób produkty Inżynierii Wymagań wspierają testowanie (K2)
- LO-4.5.3 Wyjaśnij, w jaki sposób metryki mogą być użyte przy ocenie i poprawie jakości procesu i produktów Inżynierii Wymagań (K2)

4.1 Wstęp do zarządzania wymaganiami**10 minut**

Zarządzanie wymaganiami jest zbiorem działań zarządzających i wspierających, które mają na celu zapewnienie poprawności wykonania procesu opracowywania wymagań podczas cyklu życia produktu. Innymi działaniami w ramach zarządzania wymaganiami są zarządzanie konfiguracją i zmianą oraz śledzenie powiązań.

Zarządzanie wymaganiami obejmuje następujące czynności:

- Śledzenie wymagań
- Zarządzanie konfiguracją i zmianą
- Zapewnienie jakości

4.2 Zarządzanie projektem i ryzykiem**40 minut**

Wiele z głównych powodów niepowodzeń projektów jest związanych z wymaganiami. Zaniedbanie Inżynierii Wymagań może spowodować, że wymagania są nieprecyzyjne lub sprzeczne, albo nie spełniają kryteriów i potrzeb interesariuszy. W związku z tym staranny i uporządkowany proces inżynierii wymagań jest niezbędnym elementem każdego projektu.

Aby zminimalizować ryzyko, inżynieria wymagań powinna dostarczać wkładu do następujących obszarów:

- Koncepcja projektu (ang. *Project conception*) – Inżynieria Wymagań wspiera identyfikację klientów/interesariuszy i ich celów, wizji oraz potrzeb, a także oczekiwań odnoszących się do rozwiązania danego problemu biznesowego, a także wspiera ustalanie wymagań wysokopoziomowych.
- Negocjacje kontraktu (ang. *Contract negotiation*) – Inżynieria Wymagań wspiera ocenę wymagań klientów, określenie wstępnego zakresu oraz zasobów wymaganych dla projektu, a także określenie kosztu wytwarzania (np. implementacji wymagań).
- Definicja projektu (ang. *Project definition*) – Inżynieria Wymagań zawiera definicję ról, zadań, czynności oraz dodatkowych procesów (np. zarządzanie zmianą). Ma ona także wkład we wstępny projekt architektury oraz procesu testowego.
- Realizacja projektu (ang. *Project execution*) – na tym etapie Inżynieria Wymagań dostarcza podstaw do implementacji wymagań oraz weryfikacji i walidacji wymagań (np. testowanie). Powinna także monitorować zmiany i wymuszać przegląd planów i dostosowanie ich do obecnego zakresu rozwiązania w przypadku zmian wymagań.

Nieefektywna lub słaba Inżynieria Wymagań zwiększa ryzyko projektowe poprzez przekazywanie usterek do kolejnych etapów rozwoju rozwiązania. Dlatego też podczas planowania procesu należy uwzględnić zarówno zbiór najlepszych praktyk, jak i częstych ryzyk oraz błędów w Inżynierii Wymagań. Najczęstsze błędy związane z Inżynierią Wymagań to:

- Niejasne wymagania biznesowe lub brak celów biznesowych, które mają być osiągnięte poprzez rozwiązanie
- Zmieniające się wymagania, często będące wynikiem niejasnych celów projektu lub braku jasno zdefiniowanej dziedziny biznesowej klienta. Zmiany wymagań nie są postrzegane jako błąd w podejściach zwinnych oraz iteracyjnych
- Niejasne odpowiedzialności (zarówno po stronie klienta, jak i dostawcy)
- Rozbieżności pomiędzy oczekiwaniami interesariuszy a rezultatami projektu
- Niewystarczające zaangażowanie interesariuszy

Poziom podstawowy

- Brak śledzenia, często wynikający z niedokładnego oszacowania wpływu zmiany wymagań na inne obszary produktu będącego w fazie rozwoju
- Nieprecyzyjne szacowanie kosztu lub zakresu oraz\lub definicja projektu z niemożliwymi do osiągnięcia kamieniami milowymi (co często wynika z niejasnych wymagań)

Potencjalne problemy przedstawione powyżej, można postrzegać jako ryzyko. Aby poradzić sobie z tymi - a także innymi - ryzykami, konieczny jest proces zarządzania ryzykiem.

Efektywne zarządzanie ryzykiem jest kluczowe do zmniejszenia ryzyk projektowych i produktowych. Identyfikacja, odpowiednia analiza oraz planowanie adekwatnych reakcji na ryzyka minimalizują szanse wystąpienia ryzyka oraz jego konsekwencje w przypadku, gdy to ryzyko wystąpi.

Ryzyka nie dotyczą tylko wymagań. Uogólniając, ryzyko może być zdefiniowane jako szansa zdarzenia lub sytuacji występujących i powodujących niepożądane skutki lub potencjalny problem. Ryzyko wyrażane jest w poziomach, gdzie poziom ryzyka jest określany jako prawdopodobieństwo wystąpienia niepomyślnego zdarzenia oraz jego wpływu (szkody wynikłej z tego wydarzenia) [ISTQB].

Istnieją dwa główne typy ryzyka: ryzyka produktowe oraz ryzyka projektowe.

Ryzyka projektowe to te ryzyka, które odnoszą się do zdolności projektu do spełnienia jego celów, takie jak:

- Czynniki organizacyjne (np. umiejętności, braki kadrowe oraz szkoleniowe, kwestie polityczne, problemy z interesariuszami komunikującymi swoje potrzeby i oczekiwania względem planowanego rozwiązania lub niewłaściwe oczekiwania i podejście do inżynierii wymagań)
- Kwestie techniczne (np. problemy w zdefiniowaniu właściwych wymagań lub technologii/architektury rozwiązania, zakres, w jakim wymagania nie mogą być spełnione przy obecnych ograniczeniach, niska jakość projektu, kodu, danych konfiguracyjnych, testów, wymagań lub dokumentacji rozwiązania)
- Problemy dostawcy (np. komponenty przygotowywane przez stronę trzecią, nie są dostarczone na czas lub kwestie kontraktowe)
- Ryzyko biznesowe wynikające z niskiej jakości (np. ryzyko utraty klientów w sytuacji, gdy dostarczone rozwiązanie jest niewiarygodne lub niewystarczające)

Ryzyka produktowe to potencjalne obszary awarii (niepożądanych przyszłych zdarzeń lub zagrożeń) w oprogramowaniu lub systemie – są to ryzyka w odniesieniu do jakości produktu. Obejmuje to:

- Wyższe ryzyko awarii dostarczonego oprogramowania (oprogramowanie lub system nie realizuje pożądaných funkcji w określonych limitach)
- Niska jakość dokumentacji oprogramowania (niekompletna, niespójna, trudna do utrzymania)

Poziom podstawowy

- Ryzyko, że oprogramowanie/sprzęt wyrządzi szkodę jednostce lub organizacji
- Słabe charakterystyki oprogramowania (np. funkcjonalność, wiarygodność, użyteczność lub wydajność)
- Niska integralność i jakość danych (np. kwestie migracji danych, problemy z konwersją danych, problemy z transportem danych, naruszenie standardów danych)
- Oprogramowanie nie wykonuje pożądanych funkcji i nie spełnia potrzeb interesariuszy

Proces zarządzania ryzykiem wspiera identyfikację potencjalnych czynników, które mogą mieć negatywny wpływ na realizację projektu i przygotowanie odpowiedniej reakcji na ryzyko, jeśli wystąpi.

Zarządzanie ryzykiem składa się z następujących czynności [ISTQB]:

- Identyfikacja ryzyka (ang. *Risk Identification*)
- Analiza ryzyka (ang. *Risk Analysis*)
- Łagodzenie ryzyka (ang. *Risk Mitigation*)

Podczas identyfikacji ryzyk istotne jest, by zbadać pełną sieć interesariuszy, jako że różne ich grupy lub indywidualni interesariusze mogą wskazywać różne ryzyka. Dla przykładu, sponsor biznesowy może obawiać się o wartość rozwiązania, podczas gdy najważniejszym ryzykiem dla inżyniera wymagań może być brak komunikacji z przedstawicielem biznesowym klienta.

Analiza ryzyka zawiera ustalenie poziomów ryzyka – prawdopodobieństwo i potencjalny wpływ. Oba te atrybuty mogą być wyrażone w poziomach lub – bardziej dokładnie – w formie numerycznego oszacowania (np. prawdopodobieństwo niekorzystnego zdarzenia może być wyrażone jako wartość procentowa, a wpływ może być wyrażony jako straty finansowe)

Łagodzenie ryzyka polega na planowaniu odpowiednich reakcji dla ryzyk zidentyfikowanych jako najbardziej istotne w oparciu o wyniki analizy ryzyka. Podstawowe techniki łagodzenia ryzyka można podzielić na cztery główne kategorie:

- Unikanie ryzyka
- Redukcja ryzyka
- Dzielenie ryzyka
- Ignorowanie/akceptacja ryzyka

Podczas prac nad procesem zarządzania ryzykiem należy utworzyć i utrzymywać plan zarządzania ryzykiem. Plan ten powinien być stworzony przed planem projektu a następnie okresowo aktualizowany (np. po każdej iteracji/osiągnięciu kamienia milowego). Plan zarządzania ryzykiem powinien umożliwiać efektywną kontrolę bezpieczeństwa dla zarządzania ryzykami oraz zawierać harmonogram implementacji punktów kontrolnych i osoby odpowiedzialne za te czynności.

Plan zarządzania ryzykiem obejmuje:

- Listę ryzyk (sklasyfikowanych w typy)
- Właścicieli
- Prawdopodobieństwa wystąpienia każdego z ryzyk

Poziom podstawowy

- Dotkliwość skutku wystąpienia danego ryzyka (wraz z kosztem, jeśli to możliwe)
- Strategie łagodzenia dla każdego ryzyka (wraz z osobą/grupą odpowiedzialną za podejmowanie czynności łagodzących ryzyko)
- Powiązanie ryzyk z wymaganiami i innych artefaktami projektowymi
- Macierz oceny ryzyka

Dla firm szkoleniowych: przedstawić przykład ryzyk związanych z wymaganiami i przykładowe akcje łagodzenia ryzyka

4.3 Śledzenie wymagań

40 minut

Wymagania nie są stabilne – rozwijają się w trakcie cyklu życia projektu. Przyczyny ciągłego rozwoju i proponowanych zmian mogą być następujące:

- Nowe pomysły lub potrzeby klienta (np. wynikające z nowych przepisów, zmian w biznesie, nowych produktów)
- Kontynuowana praca (na przykład: uszczegółowienie zdefiniowanych już wymagań wysokopoziomowych, następną fazą projektu, ulepszanie i optymalizacja już zaimplementowanych funkcjonalności)
- Nowe powiązania w projekcie (na przykład: integracja z nowymi systemami, nowe kanały dostępu takie jak kanały mobilne lub internetowe)

W zależności od poziomu analizy wymagań i/lub implementacji, do wymagań będą przypisane różne statusy. Cykl życia wymagań jest często wyrażany poprzez różne statusy, dla przykładu:

- Nowe (proponowane)
- Zatwierdzone
- Sprzeczne
- Zaimplementowane
- Zmodyfikowane
- Usunięte
- Wdrożone

Różne organizacje oraz podejścia mogą używać różnych cykli życia wymagań (i różnych statusów). W wielu przypadkach cykl życia wymagań, żądanie zmiany, defekt są bardzo podobne i są zarządzane poprzez te same narzędzie.

Możliwość śledzenia dostarcza metod do zarządzania rozwijającymi się wymaganiami i innymi, związanymi z nimi, artefaktami.

Możliwość śledzenia zapewnia kontrolę, że zostały przeprowadzone wszystkie ważne etapy procesu wytwarzania. Powinna być wykonana dwukierunkowo dla wszystkich artefaktów (np. od wymagań do artefaktów projektowych oraz od artefaktów projektowych do wymagań). Możliwość śledzenia jest również ważna dla testowania, weryfikacji i walidacji.

Typowymi celami śledzenia są:

- Analiza wpływu
- Analiza pokrycia
- Dowód implementacji
- Użycie wymagań (śledzenie wymagań jako dowód, iż wymagania zostały użyte oraz wskazanie, jak zostały użyte)
- Ponowne użycie wymagań

W celu zapewnienia dobrej możliwości śledzenia, należy nadawać wymaganiom unikalne nazwy.

Istnieją dwa podstawowe typy śledzenia:

- Śledzenie horyzontalne (poziome)
 - Przedstawia zależności pomiędzy wymaganiami na tym samym poziomie
- Śledzenie wertykalne (pionowe)
 - Przedstawia zależności pomiędzy artefaktami na różnych poziomach (np. powiązania pomiędzy wymaganiami a specyfikacją rozwiązania lub specyfikacją wymagań, przypadkami testowymi, kodem, modułami, planami itp.)

W kontekście wymagań śledzenie horyzontalne jest zazwyczaj używane, aby przedstawić powiązania pomiędzy różnymi typami wymagań (np. śledzenie horyzontalne pomiędzy wymaganiami funkcjonalnymi oprogramowania a wymaganiami dotyczącymi interfejsu użytkownika (GUI) dla danej funkcji). Śledzenie wertykalne jest często używane do zarządzania pokryciem wymaganiami na różnych poziomach (np. śledzenie wertykalne pomiędzy wymaganiami biznesowymi a wymaganiami rozwiązania)

Oba typy śledzenia mogą być użyte do wsparcia analizy wpływu.

Zgodnie z ISO/IEC/IEEE 29148:2011 (poprzednio IEEE 830), statyczne szablony śledzenia mogą zostać zrealizowane poprzez:

- Odnośniki tekstowe w wymaganiach do identyfikatorów innych artefaktów
- Odnośniki tekstowe w atrybutach
- Hyperlinki
- Macierze

Dla firm szkoleniowych: wyjaśnić pojęcie śledzenia i przedstawić przykład użycia

4.4 Zarządzanie konfiguracją i zarządzanie zmianą	30 minut
--	-----------------

Aby zapewnić właściwe zarządzanie wymaganiami, należy wdrożyć zarządzanie konfiguracją. W wielu przypadkach, wymagania zmieniają się i kolejne zmiany mogą mieć wpływ na inne artefakty projektu.

Celem zarządzania konfiguracją jest ustanowienie i utrzymanie integralności produktu (komponenty, dane, dokumentacja) i artefaktów oprogramowania, przez cały cykl życia projektu i produktu.

Zarządzanie konfiguracją jest dyscypliną, która wykorzystuje techniczne oraz administracyjne narzędzia i techniki do identyfikacji i udokumentowania cech funkcjonalnych i fizycznych elementu konfiguracji, kontroli zmian tych cech oraz rejestracji i raportowania postępu zmiany i statusu implementacyjnego, a ostatecznie – weryfikacji zgodności z określonymi wymaganiami [IEEE 610]

Element konfiguracji jest artefaktem, dokumentem, produktem (sprzętowym i/lub programowym), który ma wartość dla użytkownika końcowego i jest traktowany jako pojedynczy byt w procesie zarządzania konfiguracją [IEEE 610]

Typowe elementy konfiguracji używane w inżynierii wymagań to:

- Poszczególne wymagania
- Specyfikacje wymagań
- Modele

Zarządzanie konfiguracją zapewnia, iż wszystkie produkty pracy (wyniki) inżynierii wymagań są zidentyfikowane, pod kontrolą wersji, są śledzone pod względem zmian, są wzajemnie powiązane, a także powiązane z innymi elementami projektowymi (np. artefaktami programistycznymi i testowymi), tak aby można było zachować ciągłość śledzenia przez cały proces produkcyjny.

Podczas planowania projektu, powinny być wybrane, udokumentowane i zaimplementowane procedury i infrastruktura (narzędzia) zarządzania konfiguracją. Ma to związek z tym, iż elementy konfiguracji powinny być zdefiniowane i wprowadzone pod kontrolę zmian tak szybko, jak to możliwe.

Dla firm szkoleniowych: przedstawić przykłady działań zarządzania konfiguracją dla różnych produktów procesu inżynierii wymagań.

Ważnym elementem związanym z konfiguracją jest zmiana. Zmiany wymagań mogą być zgłoszone w dowolnym momencie w trakcie realizacji projektu oraz po wdrożeniu gotowego oprogramowania w środowisku produkcyjnym.

Źródłem zmian mogą być:

Poziom podstawowy

- Rozszerzenia istniejącej funkcjonalności
- Defekty znalezione w oprogramowaniu/dokumentacji
- Zgłoszenie nowej funkcjonalności lub zmiana istniejącej funkcjonalności
- Zmiany wynikające z czynników zewnętrznych (zmiany organizacyjne, prawne)

Zmiany będą zdarzać się zawsze i ważne jest, aby je planować w kontekście procesu i czasu.

Należy odróżniać zmianę od defektu. Defekt jest to odstępstwo od wymaganego stanu systemu. Zmiana jest modyfikacją istniejących lub żądaniem nowych cech, wymagań lub funkcji.

Zmiany są przetwarzane przez proces zarządzania zmianą. Proces zarządzania zmianą to proces wnioskowania, określania możliwości, planowania, implementowania i oceny poprawności wprowadzonych zmian w systemie oprogramowania, dokumentach lub innych produktach projektu. Celem zarządzania zmianą jest umożliwienie i wsparcie przetwarzania zmian i zapewnienie możliwości śledzenia zmian.

Proces zarządzania zmianą obejmuje następujące czynności:

1. Identyfikację potencjalnej zmiany
2. Wnioskowanie o nową funkcjonalność\zmiannę funkcjonalności
3. Analiza żądania zmiany (zazwyczaj wykonywana przez Komitet Kontroli Zmian i zawierająca analizę wpływu zmiany)
4. Ocena zmiany (zawierającą ocenę ryzyka, kosztu i pracochłonności zmiany, aby móc podjąć decyzję czy wprowadzać zmianę, czy też nie)
5. Planowanie zmiany (jeśli była zgoda na wprowadzenie zmiany)
6. Implementację zmiany
7. Przegląd oraz zamknięcie zmiany
8. Wdrożenie zmiany do środowiska testowego/produkcyjnego

Zmiana powinna być zgłoszona jako formalny dokument Żądania Zmiany (ang. *Change Request* CR lub *Request for Change* (RFC)). Taki dokument zawiera zwykle pożądane rozwiązanie wraz z dodatkowymi informacjami, takimi jak:

- Dane osoby/działu lub innej jednostki zgłaszającej zmianę
- Data zgłoszenia
- Powód zmiany
- Opis pożądanej zmiany
- Planowaną (pożądaną) datę implementacji zmiany (jeśli dotyczy)
- Planowany (lub dopuszczalny) koszt zmiany (jeśli dotyczy)

Komitet Kontroli Zmian (ang. *Change Control Board* – CCB) lub Komitet Kontroli Konfiguracji sprawdza i podejmuje decyzje odnośnie zmian. Komitet Kontroli Zmian jest grupą, która w oparciu o dostarczone informacje (takie jak ryzyko związane ze zmianą, jej wpływ, nakład pracy wymagany do implementacji) podejmuje decyzje odnośnie tego, czy należy wdrożyć proponowane zmiany.

Poziom podstawowy

KKZ składa się z interesariuszy projektu lub ich reprezentantów. W projektach zwinnych, rola KKZ zwykle jest wykonywana przez zespół deweloperów wraz z Właścicielem Produktu (ang. *Product Owner*) i reprezentantami klienta.

Wprowadzenie KKZ wspiera kontrolowany sposób przeglądania, oceniania i wdrażania zmian produktu lub usługi.

Komitet Kontroli Zmian może składać się z następujących ról:

- Kierownictwo projektu
- Inżynierowie wymagań
- Kierownictwo wytwarzania
- Zapewnienie jakości (zarządzanie jakością, zarządzanie testami)
- Kierownictwo biznesowe
- Klient i/lub przedstawiciel użytkownika końcowego

W zależności od stopnia złożoności i wpływu, zmiana może mieć różny wpływ na system. Małe zmiany mogą wymagać niewielkich modyfikacji, podczas gdy złożone zmiany mogą mieć znaczący wpływ na logikę systemu. Każda zmiana powinna być starannie przeanalizowana w celu identyfikacji ryzyk i oszacowania wartości modyfikacji względem spodziewanych ryzyk.

Aby podjąć decyzję dotyczącą zmiany, istotne jest przeanalizowanie jej wpływu na projekt. Odbywa się to zwykle za pomocą analizy wpływu. Analiza wpływu wykorzystuje możliwości śledzenia, aby sprawdzić, które z artefaktów będą potrzebowały zmiany, lub które powinny być sprawdzone, kiedy dana zmiana jest wprowadzana.

Zmiany w wymaganiach mogą mieć różne konsekwencje dla projektu. Najbardziej typowe skutki to:

- Prace wynikające ze zmiany (w zależności od fazy projektu):
 - Aktualizacja artefaktów analitycznych oraz projektowych (na przykład: specyfikacji)
 - Aktualizacja dokumentacji technicznej i użytkownika
 - Zmiany w strategii testów, planie testów oraz przypadkach testowych
 - Aktualizacja potrzeb/planu szkoleniowego
 - Rozszerzenie/skrócenie zakresu prac programistycznych
 - Zmiany zakresu przygotowania oraz wykonania testów
- Zmiana harmonogramu, budżetu, zasobów

Implementacja zmiany powinna być zweryfikowana przed wdrożeniem w środowiska testowe/produkcyjne.

4.5 Zapewnienie jakości Błąd! Nie zdefiniowano zakładek.	40 minut
---	-----------------

Podczas definiowania procesu zarządzania wymaganiami, koniecznym jest również zdefiniowanie koniecznych działań z dziedziny zapewnienia jakości, które będą stosowane dla zapewnienia, że różne procesy Inżynierii Wymagań i ich produktu będą dobrej jakości.

Zapewnienie jakości (ang. *Quality Assurance*, QA) jest definiowane jako “wszystkie planowane i systematyczne działania zaimplementowane w ramach systemu jakości, wykazane jako konieczne, mające na celu zapewnienie odpowiedniego poziomu zaufania odnośnie tego, że jednostka spełni wymagania dotyczące jakości” [ISO 9000]. Ta definicja implikuje, że działania są podejmowane „planowo i systematycznie” i „zapewniają odpowiedni poziom zaufania”, tzn. będzie osiągnięty pożądaný poziom jakości. Działania te zawierają techniki mające na celu wypełnienie wymagań odnośnie jakości.

Aby osiągnąć wymagany poziom jakości, wymagana jest kontrola jakości. Głównym jej celem jest sterowanie jakością i kontrola jakości produktów lub usług poprzez zastosowanie metod operacyjnych, tak aby zrealizować określone standardy jakości. Metody operacyjne wykorzystywane w inżynierii wymagań obejmują zarządzanie projektem, zarządzanie ryzykiem, zarządzanie zmianą, weryfikację i walidację, przeglądy, zarządzanie konfiguracją czy śledzenie wymagań.

W kontekście Inżynierii Wymagań kontrola jakości może także obejmować weryfikację, czy wytworzona dokumentacja wymagań spełnia odpowiednie kryteria jakościowe.

Podczas planowania działań zapewnienia jakości projektu należy mieć na uwadze różne czynniki, które mogą mieć wpływ na pożądaný poziom jakości produktu. Czynniki te zawierają:

- Kryteria jakościowe lub oczekiwania związane z procesem QA ze strony klienta
- Typ wytwarzanego produktu (np. złożoność lub odbiorcy produktu – produkt przeznaczony dla wąskiego grona odbiorców może mieć niższy poziom niektórych z atrybutów jakościowych od produktów przeznaczonych na rynek masowy)
- Środowisko, w którym produkt jest wytwarzany (np. środowisko techniczne może ograniczać możliwość osiągnięcia pożądanego poziomu atrybutów jakości)
- Dziedzina (złożoność dziedziny biznesowej, poziom innowacji, częstotliwość zmian w biznesie)
- Czynniki prawne, bezpieczeństwa oraz środowiskowe (np. regulacje dotyczące osiągania wyższych poziomów jakości)
- Presja czasowa i finansowa mogą zmniejszać możliwości uruchomienia procesów inżynierii wymagań

Poziom podstawowy

W celu zapewnienia odpowiedniego poziomu jakości, od samego początku projektu powinny być zaplanowane i wykonywane weryfikacja i walidacja wymagań (więcej w rozdziale 5.5 *Walidacja i weryfikacja wymagań*).

Dla zapewnienia jakości i kontroli jakości wymagań mogą być użyte następujące narzędzia i techniki:

- Standardy i szablony
- Przeglądy i inspekcje
- Możliwość śledzenia
- Prototypowanie
- Obserwacja kryteriów jakościowych

Zapewnienie jakości może być także wspierane poprzez zapewnienie możliwości śledzenia wymagań. Testowalne (możliwe do weryfikacji) wymagania mogą być sprawdzone pod kątem ich prawidłowej realizacji oraz zgodności z potrzebami interesariuszy. Śledzenie wymagań wspierane jest przez kryteria akceptacji (więcej w rozdziale 5.5 *Walidacja i weryfikacja wymagań*).

Jakość wymagań/specyfikacji rozwiązania może być udoskonalona poprzez uwzględnienie następujących elementów:

- Opis celu dokumentu, zakresu, definicji i słownika
- Opis celów na różnych poziomach (na przykład: specyfikacja wymagań wysokopoziomowych ma inne cele, niż szczegółowa specyfikacja wymagań funkcjonalnych)
- Zdefiniowanie ograniczeń dotyczących projektu i implementacji
- Określenie stopni/priorytetów wymagań
- Jasne stwierdzenie, co system powinien robić, zamiast jak powinien to robić
- Udokumentowanie kwestii prawnych, założeń, reguł biznesowych i zależności
- Unikanie dodatkowych opisów diagramów, które są jasne i wystarczające (zastąpienie trudnych, abstrakcyjnych tekstów diagramami, jeśli to możliwe)
- Jasno określony katalog użytkowników oraz schemat uprawnień (prawa i uprawnienia użytkownika)
- Stosowanie ustrukturyzowanej prezentacji
- Stosowanie prostego, jasnego, precyzyjnego i jednoznacznego języka

Jak wyjaśniono w rozdziale 3.2 *Ogólny proces inżynierii wymagań*, wymagania są podstawową informacją wejściową dla procesu wytwarzania i testowania systemu. Dobrze zdefiniowane wymagania redukują ryzyko projektowe oraz w szczególności ryzyko awarii produktu, ponieważ wspierają dokładne testowanie. Stabilne wymagania zwiększają szanse dotrzymania terminów.

Inżynieria wymagań jest ściśle powiązana z testowaniem. Dobre przypadki testowe wymagają dobrych wymagań, które mogą być testowane. Z tego powodu bardzo ważne jest zaangażowanie

Poziom podstawowy

testerów podczas czynności specyfikacji wymagań. Produkty inżynierii wymagań wspierają testowanie poprzez dostarczanie tak zwanych podstaw testowych. Wymagania i ich specyfikacje mogą być podstawą testową.

Należy pamiętać, że wymagania powinny być sprawdzane przez testy statyczne (przeglądy), w które można angażować testerów. Może być konieczna akceptacja wymagań ze strony kierowników testów, ponieważ w niektórych przypadkach wymagania mogą okazać się nietestowalne. Testerzy pomagają poprawić jakość wymagań, wskazując słabe punkty i ewentualne wady. Testerzy powinni także uczestniczyć w weryfikacji wymagań w celu zapewnienia testowalności.

Zapewnienie i kontrola jakości wymagają metryk. Metryka jest skalą pomiaru i metodą używaną do pomiaru [ISO 14598]. Metryki umożliwiają dokonanie wymiernego stwierdzenia odnośnie statusu projektu i jakości. Należy pamiętać, iż rezultaty pomiaru (dane zebrane podczas pomiaru) muszą być zawsze porównane z danymi referencyjnymi (odpowiednimi metrykami).

Dla wymagań można zastosować następujące metryki:

- Koszty projektu
 - Liczba wymagań i ich złożoność
- Stabilność projektu
 - Liczba zmienionych wymagań
- Liczba defektów znaleziona w specyfikacji wymagań rozwiązania
 - Typ defektu
 - Liczba defektów w wymaganiach o różnych typach (defekty logiczne, spójności, danych itp.)

Pomiar jakości wymagań jest trudny, ponieważ nie ma jednego sposobu wyrażenia jakości. Dlatego też jakość zazwyczaj mierzona jest poprzez listy kontrolne (ang. checklist) pokrywające domyślne atrybuty jakościowe. Ocenę jakości wymagań umożliwiają następujące pytania:

- Czy wymagania są prawidłowe?
- Czy wymagania są jednoznaczne?
- Czy wymagania są wykonalne?
- Czy wymagania można śledzić?
- Czy wymagania są identyfikowalne?
- Czy wymagania są testowalne?
- Czy wymagania są niezależne od projektu (abstrakcyjne)?

Miarą jakości wymagań może być także wskaźnik zmian (nie dotyczy projektów zwinnych). Zmiany są dokonywane w celu wyjaśnienia niejasnych wymagań i korekty wymagań niekompletnych lub niespójnych. Im wyższa wartość wskaźnika zmian całego zbioru wymagań (co może być spowodowane przez prace zmierzające do wyjaśnienia niezrozumiałych wymagań i/lub niepełny lub niespójny opis wymagań), tym bardziej zagrożony jest projekt. Dlatego wskaźnik zmian powinien być mierzony w celu zarządzania ryzykiem w projekcie.

Dla firm szkoleniowych: przedstawić przykład metryk jakości wymagań i procesu inżynierii wymagań.

5 Opracowanie wymagań	290 minut
------------------------------	------------------

Pojęcia

Analiza Punktów Funkcyjnych, Analiza Punktów Przypadków Użycia, BPMN, burza mózgów, cel, historyjka użytkownika, identyfikacja na podstawie istniejących dokumentów, identyfikacja wymagań, kryteria akceptacji, kwestionariusz, metoda delficka, model, model koncepcyjny, model rozwiązania, model wymagań, modelowanie GUI, obserwacja w terenie, Poker Planistyczny, ponowne użycie, priorytetyzacja, przypadki użycia, rejestracja zadań, reprezentant klienta, S.M.A.R.T., specyfikacja, specyfikacja rozwiązania, specyfikacja wymagań, stopień formalizacji, SysML, terminowanie (praktykowanie), UML, umowa, warsztaty, wizja, wywiad

Cele kształcenia dla Poziomego Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

5.2 Identyfikacja wymagań

- LO-5.2.1 Określ cele identyfikacji wymagań (K2)
- LO-5.2.2 Opisz typowe źródła wymagań i podaj kilka przykładów (K2)
- LO-5.2.3 Zrozum sens typowej wizji projektu dla Inżynierii Wymagań (K2)
- LO-5.2.4 Zrozum role interesariuszy dla identyfikacji wymagań i opisz, jak można zidentyfikować interesariuszy (K2)
- LO-5.2.5 Opisz różne techniki identyfikacji wymagań i zastosuj je dla danych scenariuszy (K3)

5.3 Analiza wymagań

- LO-5.3.1 Opisz cel i główne czynności analizy wymagań (K2)
- LO-5.3.2 Wyjaśnij różnicę pomiędzy wymaganiami a rozwiązaniem (K2)
- LO-5.3.3 Wskaż powody dla szacowania kosztów i podaj przykłady różnych podejść do szacowania (K2)
- LO-5.3.4 Zastosuj procedurę priorytetyzacji w danym scenariuszu (K3)
- LO-5.3.5 Opisz cele i konsekwencje akceptacji wymagań (K2)
- LO-5.3.6 Podaj różne modele analizy wymagań (K1)

Poziom podstawowy

- | | |
|----------|--|
| LO-5.3.7 | Zrozum zastosowanie i opracuj podstawowe modele analizy wymagań (analiza kontekstu, dekompozycja funkcjonalna, modele procesów biznesowych) (K3) |
| LO-5.3.8 | Opisz właściwości podstawowych diagramów UML (diagramy przejść stanów, diagramy przypadków użycia, diagramy aktywności i diagramy klas) i możliwości ich użycia (K2) |

5.4 Specyfikacja wymagań

- | | |
|----------|--|
| LO-5.4.1 | Zrozum i opisz zastosowanie przypadków użycia i historyjek użytkownika podczas wytwarzania specyfikacji wymagań oraz specyfikacji rozwiązania (K2) |
| LO-5.4.2 | Zrozum i opisz cele, charakterystyki, zawartość i typową procedurę tworzenia specyfikacji rozwiązania (K2) |
| LO-5.4.3 | Wyjaśnij, jakie są charakterystyki różnych poziomów formalizacji i wyjaśnij, kiedy może być użyty dany poziom formalizacji (K2) |

5.5 Walidacja i weryfikacja wymagań

- | | |
|----------|---|
| LO-5.5.1 | Podsumuj techniki weryfikacji i walidacji stosowane w celu unikania błędów w wymaganiach (K2) |
| LO-5.5.2 | Zrozum zastosowanie kryteriów akceptacji (K2) |

5.1 Wstęp do opracowywania wymagań**10 minut**

Jak zdefiniowano wcześniej, proces opracowywania wymagań zawiera następujące działania:

- Identyfikację wymagań
- Analizę wymagań
- Specyfikację wymagań
- Walidację i weryfikację wymagań

Działania te nie powinny być traktowane jako pojedyncze fazy bez żadnych powiązań. Bardzo często pojawia się wysoka presja rynkowa na skrócenie cyklu wytwarzania oraz ryzyko częstych zmian czy uaktualnień rozwiązań będących w trakcie rozwoju. Powoduje to, że jest prawie niemożliwe wdrożenie opracowywania wymagań w postaci procesu liniowego, w którym wymagania są zbierane od interesariuszy, analizowane, uszczegóławiane i przekazywane grupom deweloperskim, które później tworzą model rozwiązania i walidują wymagania. W rzeczywistości, wymagania opracowywane są iteracyjne w kierunku określonego poziomu jakości i szczegółowości wystarczającej do wykonania projektu i podjęcia decyzji odnośnie wydania. Prawie zawsze wymagania się zmieniają, co oznacza, że należy podjąć odpowiednie kroki aby złagodzić efekt tych zmian.

5.2 Identyfikacja wymagań**100 minut**

Pierwszym krokiem w procesie opracowywania wymagań jest zazwyczaj identyfikacja wymagań. Głównym zadaniem identyfikacji wymagań jest zebranie wymagań od możliwie wszystkich interesariuszy, nie tylko użytkowników czy sponsorów, ale także od grupy projektowej, rynku i innych zewnętrznych źródeł. Głównymi celami identyfikacji wymagań są:

- Identyfikacja wszystkich pożądaných funkcji, charakterystyk, ograniczeń oraz oczekiwań
- Zorientowanie wymagań względem wizji projektu
- Uszczegółowienie wymagań wysokopoziomowych i jasne opisanie funkcji oraz usług
- Wyłączenie funkcji i cech, których klient nie potrzebuje

Źródła wymagań zawierają:

- Dokumenty
- Systemy będące w użyciu
- Interesariuszy

Źródła te mogą wpłynąć na wybraną technikę identyfikacji wymagań.

Identyfikacja wymagań to nie tylko zbieranie potrzeb interesariuszy poprzez zadawanie pytań – bardzo często przed utworzeniem pełnego zbioru wymagań dla rozwiązania, zebrane informacje muszą być odpowiednio zinterpretowane, przeanalizowane, zamodelowane i zwalidowane. Techniki identyfikacji i użyte narzędzia są czasem wynikiem wyboru diagramów stosowanych do modelowania lub podejścia analitycznego. Wiele technik modelowania implikuje użycie konkretnego rodzaju technik identyfikacji.

Pierwszym krokiem identyfikacji wymagań jest ustalenie problemu, jaki należy rozwiązać. Obejmuje to identyfikację interesariuszy oraz ustalenie (lub zrozumienie wcześniej zidentyfikowanych) wysokopoziomowych celów biznesowych. Interesariusze dostarczają wymagań i ograniczeń, dlatego też bardzo ważne jest zidentyfikowanie wszystkich interesariuszy. Cele biznesowe wspierają utrzymanie wizji tego, co powinno być zrobione – wymagania zebrane podczas identyfikacji wymagań powinny wspierać osiągnięcie celów biznesowych. Dostosowanie wymagań do celów biznesowych pomaga również kontrolować zakres rozwiązania.

Klient dostarcza wstępnych wymagań. Klient jest jednym z kluczowych interesariuszy projektu i jego potrzeby powinny być zaspokojone.

Klient musi być zawsze zaangażowany w prace projektowe. Celem tego jest umożliwienie zrozumienia klienta i wypracowanie wspólnego porozumienia pomiędzy klientem a dostawcą. Dostawca powinien zatem zawsze stawiać się w pozycji klienta.

Jednym z najważniejszych źródeł wymagań jest kontrakt pomiędzy dostawcą a klientem. Umowa (kontrakt) powinna formalnie określać oraz opisywać to, czego żąda klient. Należy zapewnić, iż

Poziom podstawowy

interes klienta jest najważniejszy (np. dostawca nie wymusza preferowanego przez siebie rozwiązania, ale analizuje potrzeby klienta i rekomenduje rozwiązanie, które umożliwi spełnienie tych potrzeb w możliwie najlepszy sposób).

Kontrakt musi być zgodny z zasobami dostępnymi do implementacji rozwiązania oraz być oparty na oszacowaniach, terminach, stawkach i planach projektu. Inżynieria Wymagań dostarcza informacji wejściowej dla tego typu oszacowań.

Kontrakt powinien zawierać:

- Krótki opis planowanego rozwiązania
- Listę spriorytetyzowanych wymagań
- Kryteria akceptacji dla każdego wymagania
- Listę produktów (dokumentacja, kod, działające oprogramowanie, usługi, procesy)
- Terminy dla wytwarzania oraz dostarczenia produktu
- Inne potrzeby i oczekiwania, takie jak preferowana technologia, wymagania odnośnie zasobów etc.

Innym ważnym źródłem reprezentującym podstawowe potrzeby klienta jest wizja projektu. Dla każdego projektu wizja powinna być zdefiniowana od nowa. Wizja powinna definiować, w sposób wysokopoziomowy, cele do osiągnięcia. Bardzo ważne jest posiadanie jasnej definicji wizji projektu, ponieważ daje ona możliwość weryfikacji tego, czy projekt dostarczył wymaganych wartości czy też nie.

Wizja realizowana jest poprzez cele, które projekt powinien osiągnąć. Cele powinny określać wyniki\założenia biznesowe i powinny być zgodne z podejściem S.M.A.R.T. umożliwiającym zapewnienie mierzalności i możliwości sprawdzenia stopnia osiągnięcia celów po zakończeniu projektu. S.M.A.R.T. jest metodą używaną przy ustalaniu i definicji celów jakościowych. Podejście S.M.A.R.T. wymaga aby wszystkie cele posiadały następujące właściwości [G. T. Doran]:

- Konkretny (ang. **S**pecific)
- Mierzalny (ang. **M**easurable)
- Osiągalny (ang. **A**ttainable)
- Istotny/odpowiedni (ang. **R**elevant)
- Określony w czasie (ang. **T**imely)

Podczas identyfikacji wymagań należy zapewnić, iż wszystkie wymagania będą spełniać zarówno wizję projektu, jak i zdefiniowane zgodnie z podejściem S.M.A.R.T. cele.

Przy ustalaniu wizji i celów może pomóc poniższy proces:

1. Określenie źródeł wymagań
2. Analiza stanu obecnego (ocena obiektywna)
3. Ocena stanu obecnego (dodanie elementów subiektywnych)
4. Dedukcja/wypracowanie wizji/celów (bazując na kalkulacjach koszt-zysk)

Poziom podstawowy

Aby zidentyfikować wszystkie wymagania i potrzeby, należy zidentyfikować wszystkich interesariuszy po stronie klienta i dostawcy. Każdy interesariusz, lub każda grupa interesariuszy może dostarczyć nowych wymagań i wpływać na projekt planowanego rozwiązania. Jeśli nie zidentyfikowano wszystkich interesariuszy, istnieje ryzyko, że pewne ważne wymagania lub ograniczenia pozostaną nieznanymi i nie będą uwzględnione w projekcie. Pominięci interesariusze mogą powodować konieczność wprowadzania złożonych zmian w oprogramowaniu w późnej fazie projektu lub po wdrożeniu systemu w środowisku produkcyjnym.

Interesariusze powinni być opisani następującymi pojęciami: nazwa, funkcja, dostępność, dziedzina, cele i interesy.

Niektórzy z interesariuszy mogą tworzyć grupy interesów (np. wszyscy interesariusze biznesowi). Grupy interesów powinny być zebrane razem w celu bardziej efektywnego zarządzania ich wymaganiami.

Procedura identyfikacji i oceny interesariuszy obejmuje następujące czynności:

- Identyfikacja interesariuszy (analiza procesów biznesowych, określenie właścicieli procesów oraz produktów, analiza struktury organizacyjnej i rynku)
- Grupowanie interesariuszy i wyłanianie ich przedstawicieli (jeśli możliwe)
- Określenie relacji
- Identyfikacja potencjalnych konfliktów
- Analiza konfliktów, ich źródeł i określenie możliwości realizacji strategii wygrana-wygrana
- Identyfikacja interesariuszy minimalizujących ryzyko w celu większego zaangażowania ich w czynności projektowe
- Identyfikacja perspektyw interesariuszy

Kiedy wszyscy interesariusze, cele biznesowe oraz wizja są znane, można rozpocząć szczegółową identyfikację wymagań.

Najpopularniejszymi technikami identyfikacji wymagań są:

- Kwestionariusze (ang. *questionnaire*)
- Wywiady (ang. *interview*)
- Samo-rejestracja (nagrywanie) (ang. *self-recording*)
- Pozyskiwanie informacji od przedstawicieli klienta na miejscu/w zespole (ang. *customer representative*)
- Identyfikacja na podstawie istniejących dokumentów
- Ponowne użycie (ponowne użycie specyfikacji z określonego projektu) (ang. *reuse*)
- Burza mózgów (ang. *brainstorming*)
- Obserwacja polowa (na miejscu) (ang. *field observation*)
- Terminowanie (ang. *apprenticing*)
- Warsztaty (ang. *workshop*)
- Przypadki użycia (ang. *use cases*)

Technika	Opis	Zastosowanie	Wady
Kwestionariusze	Kwestionariusz może zawierać pytania otwarte lub zamknięte. Pytanie otwarte wymaga od respondenta sformułowania jego własnej odpowiedzi. W przypadku pytań zamkniętych, respondent jest proszony o wybór odpowiedzi z pewnej liczby możliwych opcji. Opcje te powinny się wykluczać.	Potwierdzanie lub uszczegóławianie wcześniej znanych wymagań; organizacja zawartości wymagań; wybór opcji dla wymagań/rozwiązania	<p>Nieodpowiednie do pozyskiwania niejawnej informacji</p> <p>Niski wskaźnik zwrotu przy niskiej motywacji respondentów</p> <p>Kwestionariusze mogą być ukierunkowane, co utrudnia identyfikację prawdziwych potrzeb użytkowników</p>

Technika	Opis	Zastosowanie	Wady
Wywiad	<p>Wywiad jest techniką konwersacyjną, w której przeprowadzający wywiad zadaje respondentowi pytania w celu uzyskania informacji na określony temat. Technika ta jest bardzo interaktywna i umożliwia modyfikację kolejności uprzednio przygotowanych pytań w zależności od odpowiedzi respondent oraz sytuacji.</p> <p>Przeprowadzenie dobrego wywiadu jest trudniejsze, niż mogłoby się to wydawać. Powodem tego jest typowe zachowanie podczas rozmowy (np. kończenie zdań za partnera), co może prowadzić do nadinterpretacji do danych. Przeprowadzający wywiad powinien zadawać pytania otwarte w celu uzyskania informacji, a pytania zamknięte w celu potwierdzenia (bądź zanegowania) danej możliwości.</p>	Uzyskanie informacji na dany temat; wyjaśnienie wymagań	<p>Czasochłonność</p> <p>Niewystarczająca reprodukcja wyników (trudność uzyskania tych samych odpowiedzi podczas powtarzania wywiadu)</p>

Technika	Opis	Zastosowanie	Wady
Samo-rejestracja	<p>W tej technice interesariusz (np. użytkownik końcowy) dokumentuje swoje czynności wykonywane w celu ukończenia określonego zadania.</p> <p>Oprócz dokumentowania czynności w obecnej postaci („AS-IS”), użytkownik opisuje również zmiany, żądania i potrzeby.</p> <p>Technikami powiązаныmi z tym podejściem są: demonstracje oraz przeglądy dokumentacji.</p>	Zrozumienie skomplikowanych procedur lub procesów, zwłaszcza gdy inżynier wymagań nie jest w stanie samodzielnie przyglądać się użytkownikowi wykonującemu zadanie.	<p>Zaniedbywanie czynności “automatycznych”</p> <p>Wysoka zależność od motywacji i doświadczenia użytkownika</p>
Przedstawiciel klienta na miejscu /w zespole	<p>Podejście to jest jednym z najbardziej efektywnych metod identyfikacji (i walidacji) wymagań, ponieważ umożliwia reprezentantowi systematyczne monitorowanie postępu, weryfikację poprawności projektu oraz dostarczanie informacji zwrotnej wtedy, gdy jest to potrzebne.</p> <p>Obecność reprezentanta klienta w zespole jest jedną z podstawowych zasad metod zwinnych.</p>	Zebranie i zarządzanie wymaganiami w podejściach zwinnych; uzyskanie wymagań zorientowanych na użytkownika, które łatwo mogą być zaakceptowane przez klienta.	<p>Wysoki koszt dla klienta</p> <p>Koszty adaptacji</p>

Technika	Opis	Zastosowanie	Wady
Identyfikacja wymagań na podstawie istniejących dokumentów	<p>Technika ta może być użyta w przypadku, gdy w organizacji istnieje dokumentacja, która może pomóc w identyfikacji wymagań.</p> <p>Dokumentacją taką mogą być modele i mapy procesów, opisy procesów, diagramy organizacyjne, specyfikacje produktów, procedury (np. procedury pracy), standardy i instrukcje, szablony dokumentacji.</p> <p>Zidentyfikowane wymagania są podstawą do dalszej analizy wymagań i muszą być uszczegółowione oraz rozszerzone poprzez inne, powiązane, wymagania.</p>	Zebranie wymagań dla rozwiązania, które pokrywa istniejące procesy biznesowe.	<p>Nie można zastosować, jeśli nie istnieje żadna lub istnieje tylko podstawowa dokumentacja w organizacji</p> <p>Nie można zastosować, jeśli dokumentacja nie jest prawidłowo utrzymywana (nieaktualna)</p>

Technika	Opis	Zastosowanie	Wady
Ponowne użycie (ponowne użycie specyfikacji z określonego projektu)	<p>Ponowne użycie specyfikacji z określonego projektu może być wykonane, jeśli organizacja ukończyła już jeden lub więcej projektów podobnych do aktualnego projektu. Specyfikacja wymagań przygotowana dla poprzedniego projektu(ów) może być użyta w innym projekcie w celu skrócenia czasu trwania analizy i dokumentacji wymagań – i tym samym umożliwienia wcześniejszego rozpoczęcia implementacji.</p> <p>W większości przypadków w nowym projekcie mogą być użyte jedynie niektóre części istniejącej specyfikacji. Dokumentacja, która będzie ponownie używana, powinna być zawsze sprawdzona pod kątem zgodności z aktualnymi potrzebami oraz wymaganiami i odpowiednio dostosowana.</p>	Dla projektów mających dostarczyć produkty dostosowane do konkretnych klientów; aby skrócić czas inżynierii wymagań w projektach dla podobnych rozwiązań.	<p>Wysokie koszty pierwszego projektu.</p> <p>Wymagania, które są bardzo uszczegółowione, mogą wymagać obszernego i kosztownego zarządzania zmianami.</p>

Technika	Opis	Zastosowanie	Wady
<p style="text-align: center;">Observacja polowa</p>	<p>Observacja polowa umożliwia obserwację czynności i procesów wykonywanych przez użytkowników i na tej podstawie identyfikację wymagań systemowych. Wykonywanie obserwacji polega na obserwowaniu użytkowników przy pracy i dokumentowaniu procesu, zadań oraz wyników. W niektórych przypadkach obserwacja jest rozszerzona o wywiad z użytkownikami na temat ich pracy oraz sposobów, w jaki realizują swoje zadania.</p>	<p>Pozyskanie wiarygodnych wymagań poprzez obserwację użytkownika przy pracy; uniknięcie problemów w sytuacjach, gdy interesariusze mają kłopot z wyrażeniem swoich potrzeb; usprawnienie istniejących procesów i systemów biznesowych.</p>	<p>Ryzyko pominięcia przypadków wyjątkowych.</p> <p>Niemożliwe do zastosowania w niektórych sytuacjach (np. z powodów bezpieczeństwa lub prawnych)</p>

Technika	Opis	Zastosowanie	Wady
Terminowanie	<p>Celem terminowania jest pozyskanie wymagań od klienta, szczególnie, gdy procesy i czynności wykonywane przez personel klienta nie są łatwe do opisanego przy użyciu innych technik, takich jak wywiady, lub klient ma trudności z wyrażeniem wymagań dotyczących planowanego oprogramowania.</p> <p>Terminowanie jest procesem uczenia się od klienta jego pracy. Klient, który najlepiej wie, jak wykonać określoną pracę, uczy inżyniera wymagań – jak mistrz i uczeń.</p>	<p>Zrozumienie skomplikowanych procesów biznesowych celem zaproponowania najlepszego rozwiązania; pomoc w przezwyciężeniu trudności pracowników klienta z wyrażaniem myśli i opisywaniem słownym swoich zadań.</p>	<p>Kosztowne i czasochłonne.</p> <p>Niemożliwe do zastosowania w niebezpiecznych środowiskach.</p>
Przypadki użycia	<p>Przypadki użycia pomagają inżynierowi wymagań spojrzeć na funkcjonalność rozwiązania z perspektywy „aktorów”. Aktorem może być użytkownik końcowy lub zewnętrzny system.</p> <p>Diagramy przypadków użycia definiują granice rozwiązania i pokazują powiązania z zewnętrznymi systemami bądź aktorami.</p>	<p>Zebranie i wyjaśnienie wymagań z punktu widzenia użytkownika; wyjaśnienie i organizacji funkcjonalności pożądanymi dla rozwiązania.</p>	<p>Niemożliwe do zastosowania w przypadku, gdy rozwiązanie zawiera głównie wymagania niefunkcjonalne</p>

Technika	Opis	Zastosowanie	Wady
arsztaty	<p>Warsztat jest rodzajem spotkania koncentrującym się na określonym (uprzednio zdefiniowanym i zakomunikowanym uczestnikom) temacie i angażującym interesariuszy reprezentujących zwykle różne obszary i/lub dziedziny na krótki, intensywny okres czasu.</p> <p>Warsztaty angażują osoby mające różne punkty widzenia na dany projekt i pomagają określić i opisać wymagania pochodzące z różnych perspektyw</p> <p>Warsztaty są jedną z kluczowych praktyk w metodach zwinnych, jako że angażują głównych interesariuszy w kształtowanie rejestru produktu (ang. Product Backlog). Właściwie przygotowane i przeprowadzone warsztaty pomagają złagodzić potencjalne trudności takie jak zdalne połączenie przy rozproszeniu geograficznym, sesje w podgrupach, aby złagodzić brak porozumienia itp.</p>	<p>Identyfikacja wymagań w celu określenia zakresu rozwiązania; odkrycie wymagań ukrytych (tj. wymagań, które nie są zdefiniowane wprost, lub których interesariusze nie są świadomi, a muszą być zrealizowane aby zaspokoić niektóre potrzeby lub wymagania wysokopoziomowe); opracowanie lub uszczegółowienie wymagań w nowo zidentyfikowanych obszarach; ustalenie priorytetów wymagań lub osiągnięcie porozumienia odnośnie do wymagań (podpis/akceptacja wymagań); odkrycie i rozwiązanie potencjalnych konfliktów pomiędzy wymaganiami interesariuszy; wykonanie przeglądów wyników procesów lub czynności i rozwiązanie mogących się pojawić problemów.</p>	<p>Trudne do zastosowania w przypadku zespołów rozproszonych geograficznie</p> <p>Dostępność wszystkich osób wymaganych do udziału w warsztacie</p> <p>Podczas warsztatu nie zawsze łatwo osiągnąć porozumienie, dyskusja może utknąć na (pomniejszych) problematycznych kwestiach, tym samym przedłużając proces i demotywując uczestników</p>

Technika	Opis	Zastosowanie	Wady
Burza mózgów	Burza mózgów jest powszechnie stosowaną techniką pozyskiwania wymagań związanych z mało znanymi lub nowymi obszarami działalności organizacji lub funkcjonalnościami planowanego systemu. Umożliwia zebranie wielu pomysłów od różnych interesariuszy w krótkim czasie i przy niskim koszcie. Podczas sesji burzy mózgów, uczestnicy zgłaszają pomysły i koncepcje dotyczące danego problemu.	Rozwiązywanie konfliktów w wymaganiach; zdefiniowanie różnych możliwości rozwiązania; rozwiązanie problemów biznesowych (np. mała wydajność procesu)	Trudne z niezmotywowanymi uczestnikami Trudne do zastosowania w zespołach rozproszonych

Tabela 3 Techniki identyfikacji wymagań

Dla firm szkoleniowych: przedstawić przykłady zastosowania co najmniej trzech różnych technik identyfikacji wymagań.

Aby osiągnąć dobre rezultaty i uniknąć luk w wymaganiach, zazwyczaj stosuje się różne kombinacje opisanych powyżej technik identyfikacji.

Jak opisano wcześniej, wymagania funkcjonalne specyfikują funkcje systemu postrzegane przez użytkownika końcowego. Wymagania funkcjonalne opisują również wyzwalacze procesu takie jak czynności użytkownika, wejścia/wyjścia danych powodujące start procesu biznesowego.

Podczas identyfikacji wymagań istotne jest, by zadawać pytania nie tylko o funkcje ale także o atrybuty jakościowe. Wymagania niefunkcjonalne (ang. *non-functional requirements - NFR*) opisują atrybuty jakościowe całego systemu lub jego określonych komponentów lub funkcji. Ograniczają rozwiązanie, na przykład poprzez wymaganie określonych parametrów wydajnościowych. Wymagania niefunkcjonalne są trudne do opisanie i z tego powodu często wyrażane są niejasno lub nie są w ogóle udokumentowane. To powoduje, iż są one trudne do testowania. Dlatego też należy zwracać szczególną uwagę na wymagania niefunkcjonalne na wszystkich etapach procesu inżynierii wymagań, aby zapewnić, że są one wyrażane jasno i są mierzalne

Wymagania niefunkcjonalne mogą opisać różne aspekty wydajności rozwiązania. Zgodnie z ISO/IEC 25000 (poprzednio ISO 9126), wyróżnia się następujące typy wymagań niefunkcjonalnych:

- Niezawodność (ang. Reliability)
- Użyteczność (ang. Usability)
- Wydajność (Efektywność) (ang. Efficiency)
- Utrzymywalność (ang. Maintainability)
- Przenaszalność (ang. Portability)

Wymagania нефункционалне określają kryteria, które mogą być użyte do oceny działania systemu. Z tego powodu mają one znaczący wpływ na satysfakcję klienta z używania oprogramowania. Wymagania funkcjonalne mają dostarczać funkcji; wymagania нефункционалне określają, jak łatwo i efektywnie te funkcje mogą być używane.

Dla firm szkoleniowych: przedstawić przykłady wymagań funkcjonalnych i нефункционалnych zgodnie z ISO/IEC 25000

Wymagania po identyfikacji powinny być odpowiednio udokumentowane, aby umożliwić ich dalsze śledzenie i analizę wymagań.

Wymagania muszą być określone jasno i precyzyjnie. Aby zapewnić, że są testowalne a ich implementacja może być właściwie kontrolowana, powinny one być mierzalne. Należy pamiętać, że język potoczny ma pewne ograniczenia i wady. Może to spowodować, iż opis wymagań jest niejasny i niejednoznaczny. Dlatego wszędzie tam, gdzie to możliwe powinny być zastosowane odpowiednie standardy i szablony. Standardy zapewniają powszechne zrozumienie i najlepsze praktyki tworzenia specyfikacji, podczas gdy szablony ograniczają język, który może być użyty.

Oprócz standardów i szablonów, ważnym narzędziem dla ułatwienia komunikacji pomiędzy różnymi interesariuszami oraz wprowadzenia pewnej kontroli nad niejednoznacznością języka naturalnego są słowniki.

Opis wymagań musi spełniać różne kryteria jakościowe (patrz p. 1.1.4 *Jakość wymagań*)

W zależności od poziomu abstrakcji, wymagania mogą być opisane mniej lub bardziej szczegółowo. W niektórych modelach, wymagania biznesowe mogą być spisane w formie wysokopoziomowych przypadków użycia (np. *Rational Unified Process*), lub historyjek użytkownika (ang. user stories) w podejściach zwinnych. Uogólniając, typowa struktura wymagania biznesowego powinna obejmować następujące aspekty:

- Użytkownik – dla kogo przeznaczone jest to wymaganie?
- Wynik – jakiego wyniku oczekują interesariusze?
- Przedmiot – co jest przedmiotem wymagania?
- Kwalifikator – jaki jest mierzalny kwalifikator?

Dla przykładu, rozważmy następujące wymaganie: „Agent ubezpieczeniowy musi mieć informację o nowych produktach jeden dzień przed ich wprowadzeniem.” Stwierdzenie to pokrywa wszystkie cztery opisane powyżej elementy w następujący sposób:

- Agent ubezpieczeniowy ← kto
- Musi mieć informację ← jaki wynik
- O nowych produktach ← jaki przedmiot
- Jeden dzień przed ich wprowadzeniem ← kwalifikator

Bardziej szczegółowe wymagania (wymagania systemowe/rozwiązania) mogą być opisane w formie systemowych przypadków użycia, często razem ze scenariuszami lub jako historyjki

Poziom podstawowy

użytkownika (ang. user stories) w podejściach zwinnych. Uogólniając, wymagania systemowe/rozwiązania powinny obejmować następujące aspekty:

- Określenie procesu – do jakiego procesu biznesowego odnosi się wymaganie? Ten element skupia się na funkcjonalności i określeniu wejść oraz wyjść.
- Czynność rozwiązania – jak wywoływana (uruchamiana) jest określona czynność rozwiązania? Czy czynność jest niezależna, czy też zależy od interakcji z użytkownikiem, interfejsem lub innym systemem/procesem?
- Zobowiązania – jaki jest poziom zobowiązania odnośnie spełnienia wymagania? Wyjaśnienie zobowiązania za pomocą słów kluczowych (powinien, musi itp.)
- Ograniczenia logiczne i czasowe – czy istnieją warunki brzegowe dla danego wymagania?

Dla firm szkoleniowych: omówić przykłady opisów wymagań biznesowych i wymagań systemowych/rozwiązania

Zidentyfikowane i przeanalizowane wymagania są dokumentowane w formie specyfikacji (patrz rozdział 5.4 *Specyfikacja wymagań*)

5.3 Analiza wymagań

80 minut

Następnym etapem po pozyskiwaniu/zebraniu wymagań od interesariuszy jest analiza wymagań. Głównym celem analizy jest stworzenie rozwiązania biznesowego, które implementowałoby wymagania. Tworzenie rozwiązania biznesowego wymaga rozpisania potrzeb klienta od poziomu wymagań wobec systemu aż do poziomu wymagań względem komponentów. Towarzyszy temu wykrywanie i rozwiązywanie konfliktów między wymaganiami, zarówno na tych samych jak i różnych poziomach oraz odkrywanie granic rozwiązania włącznie z sposobem interakcji ze środowiskiem.

Procedura analizy wymagań zawiera następujące kroki:

1. Analiza potrzeb
2. Opis rozwiązania
3. Szacowanie nakładu pracy i priorytetyzacja
4. Akceptacja wymagań

Pierwszy etap polega na analizie potrzeb interesariuszy w celu zaproponowania rozwiązania spełniającego ich wymagania. Na tym etapie analizy wymagań często występują aktywności weryfikacji i walidacji w celu sprawdzenia, czy wymagania zostały prawidłowo zrozumiane i zaakceptowane przez interesariuszy. Ważną częścią analizy wymagań jest sprawdzenie jasności i spójności wymagań oraz rozwiązywanie wszelkich konfliktów.

Opis rozwiązania jest przeważnie oparty o modele. Istnieją trzy podstawowe typy modeli analitycznych: modele wymagań, modele rozwiązania, modele koncepcyjne. Modele są tworzone przy pomocy odpowiednich metod analitycznych.

Kolejnym krokiem analizy wymagań jest szacowanie nakładu pracy oraz priorytetyzacja wymagań.

Szacowanie nakładu pracy łączy Inżynierię Wymagań z zarządzaniem projektem. Wczesne prognozy nakładu są oparte na wysokopoziomowych wymaganiach użytych także do oszacowania kosztu projektu. Podczas rozwoju rozwiązania, gdy wymagania ewoluują, początkowo oszacowany koszt służy do kontroli projektu, by w przypadku zmian, móc oszacować ich nakład pracy i koszt.

Najbardziej typowymi aspektami będącymi przedmiotem szacowania są:

- Koszty
- Czas
- Wymagania

Koszt projektu zależy od wielu czynników:

- Typu projektu
- Dojrzałości procesu
- Podejścia projektowego oraz testowego, metod i narzędzi

Poziom podstawowy

- Technologii
- Złożoności planowanego rozwiązania
- Celów jakościowych (na przykład, pożądanego poziomu jakości oprogramowania)
- Kwalifikacji zespołu
- Lokalizacji zespołu
- Doświadczenia

Dokładność szacowania nakładu pracy zależy od stopnia postępu projektu oraz jego dojrzałości.

Szacowanie nakładu pracy może być zrealizowane przy użyciu konkluzji przez analogię i procedury algorytmicznej. Oba podejścia wymagają, by procedury szacowania były oparte na danych historycznych i warunkach środowiskowych.

Konkluzja przez analogię przy szacowaniu wysiłku opiera się na porównywaniu kosztów podobnego projektu. Opiera się na doświadczeniu, nie na wzorach matematycznych. W tym podejściu, obecny projekt jest porównywany z przeszłymi projektami. Porównanie może obejmować:

- Liczbę i złożoność wymagań
- Stabilność obszaru biznesowego
- Zakres rozwiązania
- Zastosowaną technologię
- Charakterystyki personelu (np. umiejętności, doświadczenie)

Przykładowe procedury analogii:

- Gra planistyczna (ang. Planning Poker)
- Metoda delficka

Przykład procedury opartej na analogii można spotkać w projektach zwinnych. W takich projektach, szczególnie Scrum-owych występuje metoda szacowania nazywana Grą Planistyczną lub Pokerem Planistycznym (ang. Planning Poker). Metoda ta służy osiągnięciu w zespole konsensusu. Wydajność zespołu jest mierzona tak zwanym wskaźnikiem spalania. Sesje retrospektywne są przeprowadzane po każdym sprincie i mają na celu porównanie oszacowanych punktów do rzeczywiście wymaganego czasu. To z kolei zwiększa zdolność zespołu do szacowania.

Metoda Delficka jest innym przykładem techniki/procedury analitycznej. Jest to ustrukturyzowana technika komunikacji używana do przeprowadzania interaktywnego przewidywania. Angażuje panel ekspertów [Linstone75]. Eksperci proszeni są o udzielenie odpowiedzi na pytania zawarte w kwestionariuszu w kilku rundach. Po każdej rundzie wykonywane jest anonimowe podsumowanie prognoz ekspertów wraz z uzasadnieniem ich osądu. Następnie eksperci rewidują swoje wcześniejsze odpowiedzi uwzględniając odpowiedzi innych członków panelu.

Poziom podstawowy

Istnieje przekonanie, że podczas tego procesu rozrzut odpowiedzi będzie się zmniejszać i grupa będzie dążyć do „właściwej” odpowiedzi. Proces jest zatrzymywany po określonym kryterium zakończenia. Rezultat określają średnie wyniki ostatniej rundy.

Drugim podejściem do szacowania nakładu pracy jest procedura algorytmiczna. W tym podejściu wysiłek wyliczany jest na podstawie parametrów, które mogą opisywać produkt (wielkość, czas trwania), warunki graniczne (wydajność) itp. Można zastosować następujące metody:

- Analiza Punktów Funkcyjnych
- Analiza Punktów Przypadków Użycia

Analiza Punktów Funkcyjnych wymaga zliczenia tak zwanych punktów funkcyjnych planowanego rozwiązania. Punkt funkcyjny jest jednostką pomiaru wyrażającą ilość funkcjonalności biznesowej dostarczanej użytkownikowi przez system informatyczny. Koszt pojedynczego Punktu Funkcyjnego jest wyliczany na podstawie przeszłych projektów. Zidentyfikowane funkcje są ważone według trzech poziomów złożoności; liczba punktów przypisanych do każdego poziomu różni się w zależności od typu punktu funkcyjnego.

Analiza Punktów Przypadków Użycia jest podobna do Analizy Punktów Funkcyjnych, ale wymaga analizy innych aspektów: liczby i złożoności przypadków użycia i aktorów w planowanym rozwiązaniu.

Inną ważną czynnością analizy wymagań jest priorytetyzacja. Priorytetyzacja umożliwia ustalenie względnej ważności wymagań i zaimplementowanie tych najbardziej krytycznych w pierwszej kolejności. Priorytetyzacja wspiera wytwarzanie przyrostowe, jako że wymagania mogą być grupowane po priorytetach i te z najwyższym mogą być planowane do implementacji we wczesnych iteracjach.

Procedura ustalania priorytetów wymagań obejmuje następujące czynności:

1. Grupowanie wymagań – na które składa się identyfikacja wymagań wpływających bądź zależnych od siebie (np. wymagania tworzące jedną złożoną funkcjonalność).
2. Analiza wymagań – analiza zgrupowanych wymagań przez wszystkich interesariuszy w celu ustalenia poziomu ważności i ustalenia priorytetu wymagań. W analizie wymagań jest często używana analiza wpływu.
3. Stworzenie planu projektu wymagań – ustalenie planu, w którym wymagania o wysokich priorytetach będą implementowane jako pierwsze, przypisanie odpowiedzialności za implementację.
4. Planowanie testów przyrostów systemu – zaprojektowanie przypadków testowych dla testów każdego przyrostu systemu na podstawie priorytetów dla konkretnego przyrostu.

Popularnym podejściem do priorytetów jest grupowanie wymagań w kategorie. Zazwyczaj używana jest skala trzystopniowa (np. priorytet wysoki, średni i niski). Ze względu na to, że takie skale są subiektywne i nieprecyzyjne, znaczenie każdego poziomu w używanej skali musi być

Poziom podstawowy

zaakceptowane przez wszystkich interesariuszy. Definicja priorytetu powinna być określona jasno i powinna być kluczowym atrybutem każdego wymagania.

Dla firm szkoleniowych: Wytłumaczyć procedurę priorytetyzacji wymagań na przykładach. Wyjaśnić znaczenie priorytetyzacji wymagań dla planowania wydania/iteracji.

Akceptacja wymagań (określana także mianem uzgodnienia, bądź też podpisu), jest następnym elementem analizy wymagań. Jest to formalna umowa potwierdzająca, że treść i zakres wymagań jest dokładny i kompletny. To bardzo ważne, by upewnić się, że wymagania zostały formalnie zaakceptowane dla każdej fazy projektu, ponieważ akceptacja jest podstawą dla późniejszych czynności. Wymagania wysokopoziomowe (biznesowe) powinny zostać uzgodnione przed rozpoczęciem projektu. Wymagania szczegółowe (na poziomie projektu/systemu oraz modułów/funkcji) powinny być uzgodnione i podpisane przed przejściem do etapu realizacji. Podpisanie wymagań jest zazwyczaj ostatnim zadaniem w analizie wymagań oraz analizie specyfikacji.

Akceptacja wymagań zazwyczaj wymaga zaangażowania interesariuszy projektu, włączając:

- Kierownika Projektu
- Reprezentanta klienta
- Analityków biznesowych i systemowych
- Inżynierów wymagań
- Reprezentantów zespołu ds. zapewnienia jakości, testowania i programistów.

Jednym z celów akceptacji wymagań jest zapewnienie, że wymagania są stabilne/stałe, a wszelkie zmiany będą zarządzane przez formalne żądania zmian. Formalna akceptacja redukuje ryzyko wprowadzania nowych wymagań w trakcie implementacji.

Formalne zaakceptowanie wymagań może także zmniejszyć ryzyko nieporozumień pomiędzy klientem a dostawcą odnośnie zakresu, ponieważ wszystkie wymagania oraz specyfikacje wymagań i specyfikacje produktu powinny zostać dokładnie przejrane przed podpisaniem/akceptacją.

Uzgodnienie wymagań uznaje się za kompletne, gdy wszyscy istotni interesariusze podpisali dokumenty wymagań.

Zakończenie akceptacji wymagań powinno zostać zakomunikowane zespołowi projektowemu, zazwyczaj też uznaje się je za kamień milowy.

5.3.1 Modelowanie rozwiązania

Celem modelowania rozwiązania jest stworzenie modelu rzeczywistego problemu przy użyciu specyficznego rozwiązania biznesowego, dlatego też często uważa się je za część analizy wymagań. Modelowanie rozwiązania może być także przeprowadzone na etapie pozyskiwania wymagań,

Poziom podstawowy

ponieważ niektóre techniki pozyskiwania używają różnych podejść lub notacji do modelowania (np. pozyskiwanie wymagań przy użyciu przypadków użycia w UML). Największym wyzwaniem podczas modelowania rozwiązania jest wybranie i stosowanie odpowiednich diagramów, które opisywałyby rozwiązanie z różnych punktów widzenia. Model taki oprócz tego, że powinien poprawnie reprezentować rozwiązanie biznesowe, to jeszcze powinien być zrozumiały zarówno dla klienta, jak i dla dostarczającego rozwiązanie.

Istnieją trzy podstawowe poziomy modeli:

- Model wymagań – opisuje obszar problemu (biznesowego), często powstaje we wczesnych etapach projektu. Model ten służy do analizy wymagań i szacowania nakładu pracy i jest podstawą dla modelu rozwiązania.
- Model rozwiązania – opisuje obszar rozwiązania z różnych perspektyw, określa zarys implementacji wymagań zarówno funkcjonalnych, jak i нефункциональных. Model rozwiązania biznesowego stanowi podstawę do projektowania rozwiązań.
- Model koncepcyjny – nazywany modelem dziedziny. Zawiera koncepcje (podmioty) oraz relacje pomiędzy nimi. Głównym celem tego modelu jest zapewnienie jasności terminów i koncepcji użytych przez ekspertów domenowych użytych podczas opisywania problemu biznesowego, oraz ustalenie poprawnych relacji pomiędzy różnymi koncepcjami.

Zależnie od punktu widzenia, możliwe jest użycie różnych modeli. Niektóre z punktów widzenia możliwych do użycia podczas modelowania problemu bądź rozwiązania to:

- Widok użytkownika (np. modelowany za pomocą przypadków użycia)
- Widok logiczny (np. modelowany za pomocą wymagań funkcjonalnych)
- Widok procesu (np. modelowany za pomocą diagramów komunikacji, modeli interakcji lub wymagań нефункциональных opisujących np. wydajność procesu biznesowego)
- Widok implementacji (np. modelowany za pomocą modułów rozwiązania)
- Widok instalacji (np. modelowany za pomocą modeli integracji i architektury systemu)

Różne poziomy modelowania oraz różne punkty widzenia mogą zostać opisane innymi diagramami. Aby mieć pełny obraz rozwiązania, przeważnie stosuje się różne kombinacje widoków. Takie podejście skutkuje opisaniem modelu rozwiązania poprzez diagramy przedstawiające różne perspektywy. Niektóre przykładowe modele analizy zostały zaprezentowane poniżej (Tabela 4 Przykładowe metody i modele analizy)

Metoda analizy	Model analityczny
Analiza kontekstowa	Model kontekstowy Model wdrożenia (UML)
Analiza logiki/architektury	Dekompozycja funkcjonalna Diagram wymagań (SysML)
Analiza decyzji	Model maszyny stanowej (UML)
Analiza danych	Model związków encji (ERM) Model klas (UML)
Analiza przypadków użycia	Model przypadków użycia (UML)
Analiza przepływu procesów	Model procesu biznesowego (BPMN) Model aktywności (UML) Model komunikacji (UML)

Tabela 4 Przykładowe metody i modele analizy

Dla firm szkoleniowych: wyjaśnić zalety i sposoby aplikacji podstawowych modeli analizy (model kontekstowy, dekompozycja funkcjonalna, model procesu biznesowego), podać przykłady.

Unified Modeling Language jest jedną z popularnych metod modelowania system. UML zawiera różne rodzaje diagramów dla różnych widoków na system. UML jest ujednoczoną notacją dla analizy i projektowania systemów. UML udostępnia różne diagramy podzielone na diagramy behawioralne i diagramy strukturalne. Diagramy behawioralne przedstawiają cechy zachowania systemu lub procesu biznesowego. Diagramy strukturalne przedstawiają strukturalne elementy składowe systemu lub funkcji.

Typ diagramu (UML 2.4.1)	Zastosowanie
Diagram behawioralny	
Diagram aktywności	Modeluje zachowanie system oraz sposób, w jaki to zachowanie jest powiązane z ogólnym przepływem systemu. Używany zazwyczaj do modelowania procedur, przepływów procesów i pracy
Diagram przypadków użycia	Wyrażają przypadki użycia i relacje pomiędzy aktorami a systemem. Opisuje wymagania funkcjonalne systemu, sposób, w jaki operatorzy zewnętrzni oddziałują na granicy systemu oraz odpowiedzi systemu.
Diagram maszyny stanów	Opisuje sposób, w jaki element może przechodzić pomiędzy stanami, klasyfikuje jego zachowanie zgodnie z wyzwalaczami przejść oraz ograniczeniami.
Diagram czasowy	Definiuje zachowanie różnych obiektów w skali czasowej Dostarcza wizualnej reprezentacji zmian stanów obiektów oraz ich interakcji w czasie
Diagram sekwencji	Ustrukturyzowana reprezentacja zachowania, jako sekwencji kroków w czasie. Używany do prezentacji przepływu zadań, przepływu komunikatów oraz tego, jak w przeciągu czasu element ogólnie współdziałają ze sobą, by osiągnąć dany rezultat
Diagram komunikacji	Prezentuje interakcje pomiędzy elementami w czasie rzeczywistym, wizualizując relacje między-obiektowe.
Diagram przeglądu interakcji	Wizualizuje współpracę pomiędzy innymi diagramami współpracy w celu przedstawienia przepływu kontroli służącemu danemu celowi.

Diagramy strukturalne	
Diagram klas	Przedstawia logiczną strukturę systemu, klasy oraz obiekty tworzące model, opisujące, co istnieje i jakie posiada atrybuty oraz zachowanie. Może być użyty do modelowania danych.
Diagram obiektów	Prezentuje obiektowe wystąpienia klas oraz ich związki w danym punkcie czasu.
Diagram pakietów	Przedstawia organizację elementów modeli w pakiety oraz zależności pomiędzy nimi. Może być wykorzystany do organizacji wymagań odpowiednio do ich typu i poziomu abstrakcji.
Diagram komponentów	Prezentuje części oprogramowania tworzące system oraz ich organizację i zależności.
Diagram wdrożenia	Opisuje architekturę wykonawczą systemu.
Diagram struktury złożonej	Odzwiera wewnętrzną współpracę klas, interfejsów, komponentów (i ich właściwości) celem opisu funkcjonalności.
Diagram profili	Pozwala zaadoptować aplikację na różnych platformach (J2EE lub .NET) lub domeny (real-tilt).

Tabela 5 Diagramy UML i ich zastosowanie

Aby zamodelować bardziej złożone rozwiązanie, szczególnie dla inżynierii systemów, można użyć notacji SysML (System Modeling Language). SysML umożliwia modelowanie szerokiego typu rozwiązań obejmujących sprzęt komputerowy, oprogramowanie, informacje, procesy, personel i udogodnienia.

SysML wykorzystuje siedem diagramów UML i dostarcza dwa nowe diagramy: diagram wymagań, który obejmuje wymagania funkcjonalne, wydajnościowe wymagania oraz wymagania interfejsów, a także diagram parametryczny, który pozwala określić ograniczenia ilościowe i wydajnościowe.

W celu zamodelowania przepływu procesów i komunikacji pomiędzy aktorami, należy użyć innej notacji np. BPMN (Business Process Modeling Notation). BPMN jest zazwyczaj używany we wczesnych etapach analizy wymagań oraz podczas analizy procesów biznesowych wykonywanej razem z klientem.

W celu zamodelowania różnych aspektów interfejsu użytkownika planowanego rozwiązania (szczególnie oprogramowania), można skorzystać z modelowania Graficznego Interfejsu Użytkownika (ang. *GUI Modeling*). Prototypowanie interfejsu graficznego wykorzystuje się w celu zaprojektowania i przetestowania poszczególnych elementów. Metoda ta jest szczególnie wartościowa w przypadku mało uszczegółowionych/niejasnych wymagań oraz wymagań dających

Poziom podstawowy

się zaimplementować na wiele różnych sposobów, pozwala bowiem na dokonanie wyboru przez interesariuszy.

Poniżej wymieniono przykładowe typy prototypów [IBUQ]:

- Prototypy wertykalne [pionowe]: redukcja liczby funkcji do kilku, ale szczegółowych
- Prototypy horyzontalne [poziome]: zintegrowanie tylu funkcji ile tylko można, nie biorąc pod uwagę jako bazy funkcjonalności (używanych głównie dla testów interfejsu użytkownika)
- Prototypy scenariuszowe: funkcje symulowane są dla specyficznych przypadków przy użyciu kombinacji prototypów wertykalnych i horyzontalnych.

Prototypy mogą przybierać różne formy zależnie od przeznaczenia [Inżynieria Użyteczności IBUQ]:

- Prototypy o niskim stopniu odwzorowania (lo-fi) – małe podobieństwo do produktu końcowego; do przeglądu korzyści z danej koncepcji
- Prototypy o wysokim stopniu odwzorowania (hi-fi) – wysokie podobieństwo; przegląd szczegółów konkretnych funkcji
- Forma hybrydowa – prototypy niskiej/wysokiej wierności (lo-hi-fi), na przykład: interaktywna symulacja przy użyciu HTML lub PowerPoint®

Dla firm szkoleniowych: wyjaśnić zalety i sposoby aplikacji podstawowych diagramów UML (diagram aktywności, diagram przypadków użycia, diagram maszyny stanowej i diagram klas), podać przykłady zastosowania diagramów.

5.4 Specyfikacja wymagań

60 minut

Opracowywanie wymagań jest procesem, który oprócz odkrywania i analizowania wymagań, polega także na ułatwieniu skutecznym przekazywaniu tych wymagań różnym interesariuszom. Sposób dokumentacji wymagań odgrywa istotną rolę w zapewnieniu, że zostaną one przeczytane, zanalizowane, ew. zmienione i zwalidowane. Dokumentowanie wymagań określa się mianem specyfikacji wymagań.

Specyfikację definiuje się jako jawny zbiór wymagań wobec materiału, produktu lub usługi, dlatego też można ją uznać za pewnego rodzaju umowę opisującą zachowanie i cechy rozwiązania. Specyfikacja służy śledzeniu i zarządzaniu wymaganiami. W specyfikacji, wymagania opisane są w uporządkowany sposób oraz oddzielnie zamodelowane (wymagania są modelowane niezależnie, wymagania wysokopoziomowe są podzielone na mniejsze do poziomu, na którym poszczególne wymagania stanowią niezależne encje, które mogą być później implementowane i śledzone). Specyfikacja jest dokumentem, który posłuży jako baza do formalnego uzgodnienia wymagań implementowanych w planowanym systemie (albo innej formie). Specyfikacja wymagań może być traktowana jako część albo rezultat analizy wymagań.

Pojęcie specyfikacji może być użyte w kontekście wymagań lub rozwiązania.

Specyfikacja wymagań opisuje obszar danego problemu biznesowego (propozycję rozwiązania biznesowego dla konkretnego problemu, potrzebę lub cel, nową cechę itp.) oraz zawiera co najmniej poniższe informacje:

- Wymagania biznesowe razem z ich kryteriami akceptacji
- Ograniczenia i założenia

Stworzenie specyfikacji wymagań klienta powinno być zadaniem klienta. W niektórych przypadkach, dostawca może pomóc w przygotowaniach specyfikacji wymagań.

Specyfikacja wymagań nie musi być formalnym dokumentem. Specyfikacja wymagań może być rejestrem przebiegu albo zbiorem wymagań utrzymywanych w narzędziu do zarządzania wymaganiami.

Stworzenie specyfikacji dla innych typów specyfikacji wymagań np. wymagań systemowych, wymagań wobec oprogramowania, wymagań bezpieczeństwa, wymagań środowiskowych, prawnych wymagań itp. to zadania dla innych ról.

Specyfikacja rozwiązania opisuje rozwiązanie z różnych punktów widzenia.

Wśród innych typów specyfikacji rozwiązania można znaleźć: funkcjonalną specyfikację wymagań, systemową specyfikację wymagań, specyfikację wymagań oprogramowania.

Specyfikacja funkcjonalna jasno opisuje techniczne wymagania wobec rozwiązania. Specyfikacja funkcjonalna jest podstawą dla rozwoju produktu oraz testowania i musi zawierać precyzyjne informacje na temat wszystkich funkcjonalnych aspektów implementowanego oprogramowania.

Poziom podstawowy

Specyfikacja funkcjonalna nie opisuje sposobu implementacji konkretnej funkcji ani nie precyzuje technologii. Zamiast tego, skupia się na funkcjonalności i interakcji pomiędzy użytkownikiem a produktem.

Zarówno specyfikacja wymagań, jak i rozwiązania może być spisana w formie przypadków użycia (biznesowych bądź systemowych przypadków użycia, zależnie od intencji i odbiorców specyfikacji)

Innym typem specyfikacji jest historyjka użytkownika. Historyjki użytkownika (ang. *User Story*) są stosowane głównie w zwinnych metodykach wytwarzania oprogramowania. Historyjki użytkownika są szybką metodą operowania na wymaganiach klienta/użytkownika. Celem historyjki użytkownika jest umożliwienie szybszej reakcji na gwałtownie zmieniające się wymagania rzeczywistego świata.

Historyjki użytkownika opisują funkcjonalność, która będzie wartościowa dla użytkownika. Historyjki są zbudowane z trzech aspektów [Cohn]:

- Opisowy użyty do celów planowania i jako przypomnienie (często w formie stwierdzenia „Jako [rola użytkownika], chcę [życzenie] aby [uzasadnienie życzenia]”)
- Konwersacja o historyjce służąca pozyskaniu szczegółów
- Testy, które przenoszą i dokumentują szczegóły i które są użyte w celu stwierdzenia kompletności historyjki.

Historyjki występują często razem z osobami (np. archetypy użytkowników) reprezentującymi specyficzne typy użytkowników końcowych.

Przy tworzeniu specyfikacji można posłużyć się następującymi standardami:

- IEEE 830 Software Requirements Specification (Charakteryzuje treść i cechy dobrej dokumentacji wymaga)
- IEEE 1233 System Functional Specification (Przewodnik dla opracowania i rozwoju specyfikacji wymagań systemowych)
- IEEE 1362 System Performance Specification (Specyfikacja operacji biznesowych systemu)

Specyfikacja jest czynnością służącą sformalizowaniu wyników analizy wymagań. Identyfikacja, analiza i specyfikacja prowadzą do akceptacji wymagań.

Procedura specyfikacji rozwiązania obejmuje następujące czynności:

1. Definicja wizji i celu
 - Określenie celu rozwiązania lub wybranego obszaru rozwiązania opisanego w specyfikacji rozwiązania.
2. Identyfikacja interesariuszy
 - Kto będzie używał produktu?
 - Kto jest odpowiedzialny za wybrane obszary rozwiązania i może wspomóc lub mieć wpływ na tworzenie specyfikacji rozwiązania?
3. Określenie wymagań

- Które wymagania biznesowe wysokiego poziomu powinny zostać pokryte w specyfikacji?
 - Jakie są priorytety wymagań?
4. Ustrukturyzowana specyfikacja wymagań
 - Jakie są zależności pomiędzy wymaganiami?
 - Jak można zorganizować wymagania?
 5. Opis środowiska systemu
 - Kontekst działania rozwiązania
 6. Określenie rozwiązania
 - Definicja systemu oraz zakresu, wraz z odpowiednimi aspektami poza zakresem, ale mającymi wpływ na rozwiązanie/zakres, jak interfejsy do systemów zewnętrznych
 7. Analiza wymagań
 - Szczegółowa analiza wymagań (dekompozycja wymagań biznesowych na wymagania niższego poziomu – wymagania wobec rozwiązania/systemu lub funkcji/modułów systemu)
 - Odnajdywanie i rozwiązywanie konfliktów pomiędzy wymaganiami
 - Mapowanie wymagań biznesowych na wymagania wobec rozwiązania/systemowe oraz tworzenie struktury do śledzenia
 8. Modelowanie problemu
 - Modelowanie rozwiązania problemu biznesowego (może zawierać modelowanie bądź poprawianie istniejących procesów biznesowych). Przeważnie używa się notacji (np. BPMN) do modelowania.
 - Jak rozwiązanie zostanie wbudowane w istniejącą infrastrukturę biznesową?
 9. Modelowanie rozwiązania
 - Zależnie od typu specyfikacji rozwiązania może zawierać charakterystykę oprogramowania, sprzętową, serwisów, procesów itp.
 - Notacja modelowania (np. UML) użyta w celu pokazania rozwiązania z różnych perspektyw

Powyższa procedura angażuje wielu interesariuszy, którzy wspierają pracę nad specyfikacją w różnych dziedzinach i formalizują rezultat procesu analizy wymagań. Zależnie od informacji zgromadzonych i od założeń wybranego podejścia do inżynierii wymagań, poszczególne kroki procedury mogą być wykonywane równocześnie, iteracyjnie bądź też mogą być pominięte (np. w metodykach zwinnych procedura specyfikacji rozwiązania nie jest tak bardzo sformalizowana, ponieważ wymagania oraz samo rozwiązanie są opisane bardzo luźno). Wynik procedury, specyfikacja rozwiązania, jest punktem wejściowym do projektowania oprogramowania, sprzętu i baz danych. Opisuje funkcje (specyfikacje funkcjonalne i нефункционалне) systemu, wydajność systemu oraz ograniczenia operacyjne i te związane z interfejsem użytkownika.

Dokumentacja produktu, wliczając w to specyfikację wymagań oraz rozwiązania, może być tworzona na różnych poziomach formalizacji:

- Nieformalny
- Średnio-formalny

- Formalny

Nieformalne podejście do pisania specyfikacji oznacza, iż dokument jest napisany w języku naturalnym, bez żadnej formalnej notacji. To podejście może być zastosowane, kiedy odbiorcy nie mają doświadczenia z bardziej formalnymi i technicznymi językami specyfikacji i mogliby mieć trudności ze zrozumieniem zawartości dokumentu. Główną słabością tego podejścia jest to, iż jest ono niejednoznaczne i może prowadzić do nieporozumień i nadinterpretacji. Dodatkowo, specyfikacja nieformalna nie jest dobrą podstawą do implementacji oraz testowania, ponieważ nie jest dość jasna i precyzyjna.

Specyfikacja średnio-formalna powinna zawierać formalną notację i być dobrze ustrukturyzowana. Takie specyfikacje opierają się zwykle na określonych szablonach (często pozyskanych z odpowiednich standardów). Specyfikacje średnio-formalne mogą wyrażać wymagania w postaci modeli i używać sformalizowanego naturalnego języka. Jedną z najpopularniejszych notacji używanych do dokumentacji jest UML.

Formalna specyfikacja jest matematycznym opisem rozwiązania, które może być użyty do implementacji. Specyfikacja formalna opisuje, co system powinien robić, zwykle nie opisuje tego, jak powinien to robić. Ponieważ taka specyfikacja jest oparta na matematycznych formułach albo specyficznym języku (np. VDM Specification Language (VDM-SL)/Vienna Development Method Specification Language) i jest trudniejsza do nauczenia, specyfikacja formalna jest używana stosunkowo rzadko i wymaga specyficznej wiedzy.

5.5 Walidacja i weryfikacja**40 minut**

Ponieważ wymagania są podstawą do rozwijania systemu, każdy błąd lub pominięte wymaganie wpływa na inne procesy wytwórcze w projekcie. Należy pamiętać, że defekty wynikające z niskiej jakości wymagań są droższe do naprawienia w późniejszych fazach projektu, niż inne ich rodzaje. Ponadto, im później zostaną wykryte defekty, tym wyższe są koszty ich naprawy.

Dlatego konieczne jest stosowanie weryfikacji („czy tworzymy produkt prawidłowo” i walidacji („czy tworzymy właściwy produkt”) wymagań.

Walidacja i weryfikacja wymagań powinna być procesem ciągłym podczas cyklu wytwórczego, bowiem umożliwiają upewnienie się, że produkt spełnia kryteria jakościowe i potrzeby interesariuszy. Dobrą praktyką jest planowanie i wykonywanie walidacji i weryfikacji wymagań już we wczesnych fazach projektu np. pozyskiwania wymagań. Walidacja i weryfikacja może być przeprowadzona różnymi technikami np. przeglądami, prototypowaniem lub prezentacjami proponowanego rozwiązania interesariuszom w celu zgromadzenia informacji zwrotnej.

Walidacja i weryfikacja powinna także sprawdzać, czy wymagania, specyfikacja rozwiązania/wymagań odpowiadają standardom (szablonom) oraz czy są udokumentowane i przetestowane względem kryteriów jakości (patrz rozdział 1.1.4 *Jakość wymagań*). Ponadto walidacji i weryfikacji powinno się poddać modele powstałe podczas analizy i specyfikacji wymagań.

Ponieważ wymagania są podstawą dla tworzenia rozwiązania i jego testów, jakość wymagań jest krytyczna dla sukcesu projektu. Jasne, kompletne, spójne i testowalne wymagania redukują ryzyko produktowe (a nawet projektowe), jako że pozwalają na szczegółowe testy. Zaleca się włączenie testerów w proces przeglądu wymagań, jako że ich uwagi mogą istotnie wpłynąć na jakość wymagań, jakość specyfikacji wymagań i produktu poprzez identyfikację słabych punktów i potencjalnych błędów.

Testowalność wymagań jest wspierana przez kryteria akceptacji. Kryteria akceptacji to warunki, które muszą zostać spełnione, aby rozwiązanie mogło zostać zaakceptowane przez obie strony dostawcę i klienta. Kryteria te ustala się przed rozpoczęciem projektu – najlepszą praktyką jest definiowanie kryteriów akceptacji jako części dokumentacji kontraktu. Każde wymaganie wysokiego poziomu musi posiadać przynajmniej jedno kryterium akceptacji. Ponadto każde kryterium musi być mierzalne za pomocą realistycznej i uzgodnionej między stronami miary. Takie kryteria są często bazą dla planu jakości projektu i testów akceptacyjnych.

Do firm szkoleniowych: pokazać przykłady kryteriów akceptacji dla przykładowych wymagań.

6 Modele Inżynierii Wymagań**100 minut****Pojęcia**

CMMI, cykl życia produktu, model dojrzałości, model procesu, podejście iteracyjne, podejście sekwencyjne, podejście zwinne, Rational Unified Process, rejestr produktu, SPICE, Właściciel Produktu

Cele kształcenia dla Poziomu Podstawowego inżynierii wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

6.1 Modele i podejścia do wytwarzania i utrzymania

- LO-6.1.1 Podsumuj charakterystyki modelu cyklu życia produktu (K2)
- LO-6.1.2 Wymień podobieństwa i różnice w procesach inżynierii wymagań w modelach wytwarzania i utrzymania produktu (K2)

6.2 Modele dojrzałości

- LO-6.2.1 Zrozum, dlaczego Inżynieria Wymagań jest częścią modeli dojrzałości (K2)

6.1 Modele i podejścia do wytwarzania i utrzymania

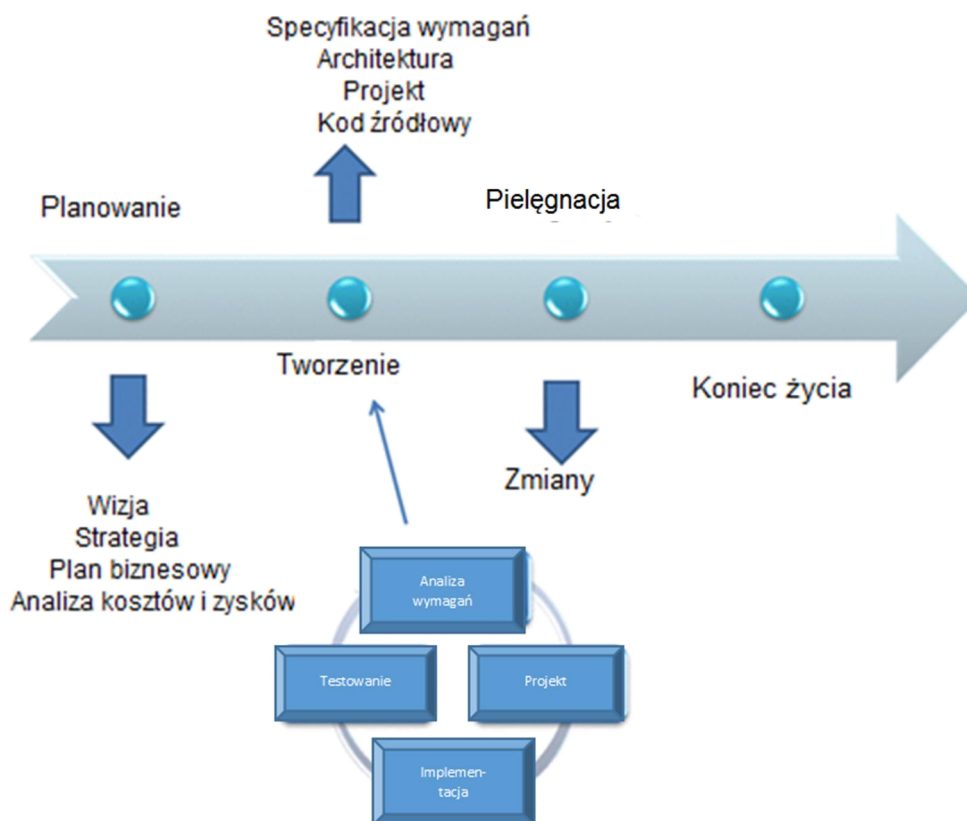
60 minut

Struktura procesu Inżynierii Wymagań często zależy od cyklu wytwórczego. Proces Inżynierii Wymagań powinien zostać dostosowany do specyficznych potrzeb środowiska wytwórczego.

Generalnie niniejszy rozdział ma pokazać, w jaki sposób różne czynności w ramach Inżynierii Wymagań przejawiają się w modelach wytwórczych.

Modele procesu to niezależne od metody opisy procesów wytwórczych. Procesy opisują przeważnie role, aktywności, fazy, dokumentacje mają standardowy format dla planowania, organizacji.

Cykl życia produktu (PLC) definiuje różne fazy wytwarzania produktu, podstawowe z nich to: planowanie, wytwarzanie, utrzymanie i koniec życia. Etap wytwarzania jest zazwyczaj podzielony na kilka faz, często wykonywane są one iteracyjnie.



Rysunek 2 Podstawowy cykl życia produktu (PLC)

Powyższy PLC jest podstawą dla różnych modeli wytwarzania.

Modele wytwórcze mogą być zaklasyfikowane albo jako tradycyjne, albo zwinne.

Tradycyjne podejście może być podzielone na modele sekwencyjne, iteracyjne i przyrostowe (dwa ostatnie przeważnie są łączone razem).

Najprostszy model wytwarzania – model kaskadowy – może być potraktowany jako liniowa reprezentacja ogólnego procesu inżynierii wymagań. Model kaskadowy opiera się na założeniu że można zebrać, zdefiniować i przeanalizować wszystkie wymagania we wczesnej fazie procesu wytwórczego oraz że te wymagania nie ulegną zmianie. Ta słabość eliminuje ten model z wielu współczesnych projektów, w których wymagania są zmieniane i dopasowywane do biznesu, technologii i zmian na rynku.

Model V jest rozwinięciem modelu kaskadowego. Głównym ulepszeniem w modelu V jest podział pomiędzy proces wytwórczy i testy, gdzie każdy etap wytwarzania ma odpowiadający mu etap testów. Model może być postrzegany jako dekompozycja wymagań, w której każdy poziom w procesie wytwarzania ma powiązany z nim poziom testów.

Model V jest dosyć efektywny w przypadku inżynierii wymagań podczas fazy wytwórczej i daje dobrą podstawę dla zarządzania wymaganiami włącznie ze śledzeniem i zarządzaniem zmianami. Opracowywanie wymagań może być utrudnione w modelu V, jako że nie da się zebrać wszystkich potrzebnych wymagań we wczesnych etapach rozwoju. Bardzo często początkowy zakres produktu rozrasta się z powodu pojawiania się nowych wymagań ze strony interesariuszy w późniejszych fazach i otoczenia lub środowiska systemu. To powoduje potrzebę intensywnej pracy i zarządzania zmianami.

Model V można uznać za najlepszy model dla opracowywania wymagań i zarządzania wymaganiami w dobrze zdefiniowanych i stabilnych obszarach.

Do firm szkoleniowych: graficzna reprezentacja modelu V, opis procesu inżynierii wymagań w modelu V

W przypadku bardziej skomplikowanych i mniej stabilnych rozwiązań, poleca się użycie modeli iteracyjnych lub przyrostowych. Podejście iteracyjne/przyrostowe jest oparte na założeniu, że zazwyczaj ciężko jest zebrać wymagania, zwłaszcza w komercyjnych systemach. Techniki iteracyjnego/przyrostowego procesu wytwórczego zakładają współpracę pomiędzy programistami i użytkownikami. Wymagania są często zdefiniowane wyłącznie jako cele biznesowe, które zostaną rozpisane w późniejszych iteracjach.

W wytwarzaniu przyrostowym zakłada się, że lista wymagań jest podzielona na mniejsze części, które są następnie implementowane w kolejnych krokach lub przyrostach. Chodzi o to, by każda część została zaimplementowana, i w najlepszym przypadku, została wypuszczona tak by skrócić czas do wydania systemu (TTM ang Time To Market). Ponadto umożliwia to otrzymanie

Poziom podstawowy

natychmiastowej informacji zwrotnej na temat przyrostu, umożliwiającej wykonanie potrzebnych zmian jeszcze na nieukończonym przyroście. W ten sposób obniża się ryzyko (szczególnie gdy porównamy to do strategii „jednokierunkowej”). Nawet jeśli projekt zostanie zakończony przed wykonaniem wszystkich zamierzonych celów, interesariusze i tak skorzystają na już wygenerowanych przyrostach.

Podejście iteracyjne jest szczególnie przydatne w przypadku dużych projektów z dużą liczbą wymagań. Głównym założeniem podejścia iteracyjnego jest wytworzenie rozwiązania w powtarzających się cyklach (można je nazwać mini-PLC) i w krótszych odstępach czasu. Taka organizacja czyni projekt bardziej kontrolowalnym oraz umożliwia systematyczne monitorowanie i adresowanie ryzyka. Przykładem iteracyjnego modelu jest RUP *Rational Unified Process*, model procesu opracowany przez IBM.

RUP jest modelem wytwarzania oprogramowania zorientowanym obiektowo. Organizuje projekt w 4 główne fazy: początkowa (inencji), opracowania (elaboracji), konstrukcji i przejścia. Każdy z tych etapów ma własne cele i produkty, określone w ramach RUP. RUP jest alternatywnym modelem, w którym zawartość jest wybierana na podstawie ryzyka, jakie niesie za sobą jej implementacji. Liczba iteracji w poszczególnych fazach może być różna i zależy od specyficznych potrzeb projektu lub produktu. Czynności inżynierii wymagań są przedmiotem dyscyplin analizy wymagań i projektowania w modelu RUP.

Do firm szkoleniowych: uszczegółowienie modelu RUP wraz z reprezentacją graficzną; szczegółowe studium dyscypliny wymagań.

Wszystkie powyższe modele przynależą do podejścia tradycyjnego. Dla tradycyjnych podejść największym wyzwaniem jest wsparcie częstych zmian. Większość podejść tradycyjnych nie wspiera wprowadzania zmian w efektywny sposób. Jako odpowiedź na te problemy powstało podejście zwinne. Podejście zwinne oparto o Manifest Zwinnego Wytwarzania Oprogramowania (Manifest Agile), który przedstawia zbiór proponowanych wartości i pryncypiów użytych w środowisku zwinnym. Najważniejszymi zasadami zwinności są [Manifest Agile]:

- Zadowolenie klienta dzięki szybkiemu dostarczaniu wartościowego oprogramowania
- Otwartość na zmiany wymagań, nawet późno w projekcie
- Działające oprogramowanie jest dostarczane często (tygodnie, a nie miesiące)
- Prostota – sztuka minimalizowania ilości koniecznej pracy – jest kluczowa.
- Regularna adaptacja do zmieniających się okoliczności

Implikacje tych zasad są bardzo ważne. W środowiskach zwinnych wymagania są komunikowane i śledzone poprzez mechanizmy takie jak: rejestry produktowe, historyjki i prototypy ekranów. Zobowiązania co do wymagań są wykonywane albo przez zespół, albo uprawnionego kierownika zespołu. Możliwość śledzenia oraz spójność pomiędzy wymaganiami a produktami prac jest zarządzana poprzez wspomniane powyżej mechanizmy oraz poprzez czynności początku i końca iteracji, takie jak „retrospektywy” oraz „demo”. Proces zarządzania wymaganiami nie jest tak formalny, jak w tradycyjnych modelach.

Poziom podstawowy

Proces opracowywania wymagań jest blisko związany z powiązaną definicją specyficznego opisu i analizy historyjek użytkownika oraz scenariuszy. Wiele z aktywności związanych z opracowywaniem wymagań nie jest opisane w modelu zwinnym, ale oczekuje się, że zostaną wykonane jako część pracy Właściciela Produktu.

W środowiskach zwinnych potrzeby i pomysły klienta są iteracyjnie pozyskiwane, rozwijane, analizowane i walidowane. Wymagania są dokumentowane w takich formach, jak historyjki użytkownika, scenariusze, przypadki użycia, rejestry produktowe oraz wyniki iteracji (działający kod w przypadku oprogramowania). To, które wymagania będą adresowane w danej iteracji, zależy od oceny ryzyka oraz priorytetów przypisanych do tego, co pozostało w rejestrze produktowym. To, jaki poziom szczegółowości wymagań (i innych artefaktów) należy zastosować, zależy od potrzeby koordynacji (pomiędzy członkami zespołu, zespołami i następnymi iteracjami) oraz ryzyka utraty wiedzy. Nawet, jeśli klient jest w zespole, nadal może być konieczne utrzymanie osobnej dokumentacji klienta oraz produktu, aby umożliwić eksplorację wielu rozwiązań. W miarę rozwijania rozwiązania, odpowiedzialność za wydzielone wymagania jest przypisana do odpowiednich zespołów.

Przykładami zwinnego podejścia są Programowanie Ekstremalne i SCRUM.

Programowanie Ekstremalne jest metodologią wytwarzania oprogramowania opracowaną przez Kenta Becka w celu odpowiedzi na zmieniające się wymagania klienta. Metodologia zaleca częste dostarczanie oprogramowania do klienta w krótkich cyklach operacji (ramy czasowe), co ma na celu polepszenie produktywności i dostarczenie punktów kontrolnych, w których można zaadaptować nowe wymagania.

Niektóre z charakterystyk Programowania Ekstremalnego obejmują:

- Unikanie programowania funkcjonalności aż do momentu, gdy są rzeczywiście potrzebne
- Oczekiwanie zmian w wymaganiach klienta wraz z upływem czasu i lepszym zrozumieniem problemu
- Częsta komunikacja z klientem oraz pomiędzy programistami
- Kompletny brak określania wymagań (nie ma osobnej „fazy” wymagań przed startem wytwarzania; rozwijanie, doskonalenie i odkrywanie wymagań jest częścią właściwego wytwarzania oprogramowania (programowania))

SCRUM jest zwinnym środowiskiem pracy zawierającym zbiór praktyk i predefiniowanych ról. Jedną z głównych charakterystyk SCRUM jest podział wytwarzania na „przebiegi” (ang. *Sprint*) zwykle o czasie trwania ok. 2 do 4 tygodni. Podczas każdego przebiegu, zespół tworzy tak zwany produkt potencjalnie możliwy do wydania.

SCRUM umożliwia zarządzanie wymaganiami poprzez rejestry. Istnieją 2 typy rejestrów:

- rejestr produktu – wysokopoziomowa lista utrzymywana poprzez cały projekt. Lista gromadzi wymagania w formie opisów wszystkich potencjalnych cech, uszeregowanych według wartości biznesowej. Rejestr produktu jest własnością Właściciela Produktu.

Poziom podstawowy

- Rejestr Przebiegu – lista zadań do wykonania przez zespół w ciągu następnego przebiegu. Funkcjonalności są dzielone na zadania, które – zgodnie z dobrymi praktykami – powinny trwać od czterech do szesnastu godzin pracy. Rejestr Przebiegu jest własnością Zespołu.

Głównym wpływem SCRUM na inżynierię wymagań jest to, że specyfikacje wymagań nie są ukończone i zwalidowane przed rozpoczęciem wytwarzania.

Właściciel Produktu i zespół na podstawie priorytetu biznesowego i wymaganego nakładu pracy uzgadniają, które cechy z Rejestru Produktowego będą uwzględnione w kolejnym przebiegu. Wymagania użytkownika są formułowane przez Właściciela Produktu, jako „Historyjki Użytkownika”, które zawierają informacje o tym, „kto, co, dlaczego” w danym wymaganiu, nie zawierają natomiast informacji „jak”. Na początku przebiegu wybrane funkcjonalności są dzielone na zadania w Rejestrze Przebiegu, a następnie implementowane.

Zaangażowanie Właścicieli Produktu, na przykład poprzez prezentowanie im zaimplementowanych funkcji oprogramowania, pomaga zespołowi i samym Właścicielom Produktu wyjaśnić wymagania.

Do firm szkoleniowych: podać przykłady historyjek użytkownika oraz odpowiadające im elementy rejestrów Produktowego i Przebiegu.

6.2 Modele dojrzałości	40 minut
-------------------------------	-----------------

Proces Inżynierii Wymagań może być doskonały tak, jak każdy inny proces wytwarzania lub utrzymania produktu/rozwiązania. Usprawnić można każdą aktywność Inżynierii Wymagań. Pozyskiwanie wymagań może być usprawnione tak, by efektywniej zbierać wymagania klienta. Aby to osiągnąć, organizacja może wprowadzić odpowiednie techniki zapewniające szybsze zbieranie wymagań, upewniając się, że są kompletne i zaakceptowane przez większość interesariuszy. Te techniki obejmują: wywiad, burze mózgów, wstępne prototypowanie, używanie scenariuszy.

Popularną metodą usprawniania procesu Inżynierii Wymagań jest zastosowanie modeli dojrzałości. Proces ten zakłada ocenę aktualnego poziomu dojrzałości i identyfikację obszarów do poprawy. Modele dojrzałości zawierają poziomy dojrzałości. Poziomy dojrzałości są używane do identyfikacji, oszacowanie i rozwój procesu dojrzałości.

Przykłady modelu dojrzałości, które mogą być zastosowane do procesu inżynierii wymagań to ISO/IEC 15504 (SPICE – *Software Proces Improvement and Capability Determination*) lub *Capability Maturity Model Integrated* (CMMI). Oba modele definiują 5 poziomów procesów czy obszarów i pozwalają na porównywanie poziomu dojrzałości w różnych organizacjach.

ISO/IEC 15504 (SPICE – *Software Process Improvement and Capability Determination*), jest zbiorem standardów technicznych dla procesu wytwarzania oprogramowania oraz powiązanych z nim biznesowych funkcji zarządczych. SPICE może być użyty w celu poprawienia procesu i/lub określenia możliwości (np. ocena możliwości procesu dostawcy). SPICE definiuje procesy podzielone na pięć kategorii: klient- dostawca, inżynieria, wsparcie, zarządzanie i organizacja. Dla każdego z tych procesów określony jest poziom możliwości:

1. Proces wykonany
2. Proces zarządzany
3. Proces ustalony
4. Proces przewidywany
5. Proces optymalizowany

Kluczowe obszary procesów związanych z inżynierią wymagań w modelu SPICE to:

- ACQ1 Przygotowanie Przejęcia BP1-3
- ENG1 Pozyskiwanie Wymagań BP1-6
- ENG2 Analiza Wymagań Systemowych BP1-3
- ENG4 Analiza Wymagań Oprogramowania BP1-6

CMMI definiuje 5 poziomów dojrzałości dla wytwarzania, usług i zakupów:

1. Wstępny
2. Zarządzany
3. Zdefiniowany

4. Zarządzany ilościowo
5. Optymalizowany

Kluczowymi obszarami dla Inżynierii Wymagań według modelu CMMI są:

- REQM Zarządzanie Wymaganiami
- RD Opracowywanie Wymagań

Proces Inżynierii Wymagań jako taki może być traktowany jak środek służący poprawie procesu rozwoju, jako że efektywna Inżynieria Wymagań przyspiesza wysiłek implementacyjny poprzez eliminację czasu zmarnowanego na klaryfikację i wyjaśnianie wymagań, ponadto redukuje liczbę defektów spowodowanych błędami w specyfikacji wymagań i ryzyko niezrozumienia dokumentacji wymagań niskiej jakości. Dobry proces Inżynierii Wymagań zwiększa jakość testowania poprzez dostarczenie dobrze opisanych, dokładnych i testowalnych wymagań. Ponadto czyni testy akceptacyjne znacznie łatwiejszymi i bardziej rzetelnymi.

Do firm szkoleniowych: zademonstrować typowe wymagania dla procesu inżynierii wymagań w określonym modelu dojrzałości

7 Wsparcie narzędziowe**80 minut****Terminologia**

Narzędzia do zarządzania defektami, narzędzie do identyfikacji wymagań, narzędzie do modelowania, narzędzie do zarządzania projektem, narzędzie do zarządzania wymaganiami, narzędzie do zarządzania zmianami, narzędzie do prototypowania, projekt pilotażowy, uzasadnienie koncepcji (POC), kategorie wymagań.

Cele kształcenia dla Poziomu Podstawowego Inżynierii Wymagań

Cele określają wiedzę i umiejętności, uzyskiwane po ukończeniu każdego modułu

7.1 Korzyści z narzędzi

- LO-7.1.1 Wyjaśnij korzyści z wykorzystywania narzędzi do wspierania różnych aktywności w Inżynierii Wymagań (K2)

7.2 Kategorie narzędzi

- LO-7.2.1 Zrozum cel i sposób wykorzystania różnych kategorii narzędzi wspierających (K2)

7.3 Wybór narzędzi

- LO-7.3.1 Wyjaśnij, które ważne czynniki należy uwzględnić podczas wyboru narzędzia (K2)
- LO-7.3.2 Wyjaśnij proces wprowadzania narzędzia do organizacji (K2)

7.1 Korzyści z narzędzi**30 minut**

Różne narzędzia wspierają różne specyficzne aktywności związane z Inżynierią Wymagań.

Narzędzia do przechowywania i administracji wymagań ułatwiają Inżynierię Wymagań. Większość narzędzi do zarządzania wymagań udostępnia wspólne repozytorium dla wymagań, w którym to repozytorium wymagania mogą być organizowane i utrzymywane (ze śledzeniem zmian). Umożliwia to utrzymywanie spójności i aktualności dla nawet bardzo skomplikowanych i zmieniających się dokumentów. Takie narzędzia mogą automatyzować pewne ręczne aktywności oraz mogą służyć do generowania różnych statystyk i raportów. Narzędzia służące do modelowania wymagań wykorzystują określone notacje np. UML lub BPMN. Bardzo często narzędzia modelujące potrafią generować specyfikację wymagań na podstawie modeli.

Narzędzia mogą wspierać następujące aktywności Inżynierii Wymagań:

- Pozyskiwanie i przechowywanie wymagań.
- Definiowanie i utrzymywanie śledzenia wymagań
- Modelowanie wymagań i rozwiązania (wraz z prototypowaniem i modelowaniem biznesowym)
- Dokumentowanie wymagań (tworzenie specyfikacji wymagań)
- Raportowanie statusu
- Zarządzanie konfiguracją i zmianami

Niektóre z korzyści wykorzystywania narzędzi to:

- Zapewnienie, że wszystkie wymagania są przechowywane w jednym miejscu i dostępne dla wszystkich interesariuszy
- Wspieranie śledzenia wymagań (np. do przypadków testowych) oraz umożliwienie weryfikacji odpowiedniego pokrycia wymagań
- Umożliwienie zarządzania zmianami wymagań w łatwy sposób
- Poprawa jakości specyfikacji wymagań poprzez wymuszanie stosowania określonych szablonów dokumentów i notacji modelowania
- Oszczędność czasu dzięki automatyzacji niektórych działań (takich jak generowanie pełnych specyfikacji z narzędzia)
- Dostarczanie informacji dla podejmujących decyzję (np. poprzez metryki zawierających liczbę zaimplementowanych wymagań)

7.2 Kategorie narzędzi**20 minut**

Narzędzia wspierające aktywności związane z Inżynierią Wymagań mogą być podzielone w następujący sposób:

- Narzędzia do zarządzania wymaganiami
- Narzędzia do pozyskiwania wymagań
- Narzędzia do modelowania
- Narzędzia do prototypowania
- Narzędzia do zarządzania defektami
- Narzędzia do zarządzania konfiguracją i zmianami
- Narzędzia do zarządzania projektem

Wiele narzędzi może być wykorzystanych do więcej niż jednego zadania, np. oprócz modelowania może oferować przechowywanie wymagań w repozytorium a ponadto konfigurację i zarządzanie zmianą, może wspierać różne notacje, generowanie dokumentacji i statystyk.

Do firm szkoleniowych: zaprezentować narzędzie wspierające przynajmniej 2 aktywności związane z inżynierią wymagań.

7.3 Wybór narzędzia**30 minut**

Wybór narzędzia powinien nastąpić zanim produkt zacznie powstawać – zapobiegnie to potencjalnym problemom ze zmianą narzędzi w trakcie projektu.

Koszty zakupu narzędzi są bardzo zróżnicowane. Narzędzia komercyjne mogą być bardzo drogie, podczas gdy narzędzia open source są bezpłatne. Wybór narzędzia musi być zatem dokonany bardzo ostrożnie. Przed wyborem narzędzia, powinna być przeprowadzona analiza. Analiza powinna obejmować następujące problemy:

- Ocena dojrzałości organizacji, mocnych i słabych stron oraz korzyści (potencjalnego ulepszenia procesu inżynierii wymagań) wynikających z użycia narzędzia.
- Wymagania wobec funkcjonalności narzędzia.
- POC (ang. *proof of concept*), przy użyciu narzędzia będącego przedmiotem oceny w celu sprawdzenia, czy narzędzie może działać efektywnie z danym procesem inżynierii wymagań oraz z danym środowiskiem w celu określenia potencjalnych miejsc w infrastrukturze, które wymagają zmiany.
- Oszacowanie zwrotu z inwestycji ROI (ang. *return of investment*) oparta na konkretnym przypadku biznesowym, zawierającym np. analizę kosztów narzędzia dla wybranego projektu wszystkich projektów.
- Czy narzędzie może być zintegrowane z innymi potrzebnymi narzędziami (takimi jak narzędzie do zarządzania defektami, narzędzie do zarządzania projektem)?
- Czy narzędzie umożliwi wymianę informacji z narzędziem wykorzystywanym przez organizację klienta (w niektórych przypadkach klient wykonuje własną analizę wymagań, a produkty szczegółowej analizy wymagań dostawcy powinny być następnie przeniesione do środowiska klienta)
- Łatwość obsługi i uczenia się (potencjalne koszty szkolenia), dostępność pomocy online, podręczników, samouczków, innego wsparcia

Wybór narzędzia dla organizacji powinien rozpocząć się projektem pilotażowym. Podstawowe cele przeprowadzenia projektu pilotażowego:

- Poznać szczegóły narzędzia
- Sprawdzić, jak narzędzie wpisuje się w istniejący proces i praktyki, ustalić, co należałoby zmienić
- Ustalić wspólne zasady używania, zarządzania, przechowywania i utrzymywania narzędzia oraz zasobów Inżynierii Wymagań

Poziom podstawowy

Projekt pilotażowy obniża ryzyko wprowadzenia narzędzi, które jeśli zostały przyjęte bez jakiegokolwiek analizy, mogą się okazać bardzo drogie. Poniżej przedstawiono przykładowe problemy z drogimi narzędziami:

- Wybór narzędzia, które nie odpowiada wymaganiom interesariuszy i nie spełnia swojego celu
- Kupno drogiego narzędzia, które jest używane tylko w jednym projekcie lub zamawianie drogiego narzędzia, jeśli istnieje tańszy odpowiednik
- Koszt szkolenia w przypadku zakupu narzędzia bez wystarczającego systemu pomocy (zwłaszcza, gdy od narzędzia wymaga się tylko podstawowej funkcjonalności)
- Koszt rozszerzenia zakupionego narzędzia o dodatkowe niewspierane funkcje (gdy były też inne narzędzia o takiej funkcjonalności)
- Integracja narzędzia z innymi używanymi w organizacji

Ogólne czynniki sukcesu wdrożenia narzędzia są następujące:

- Adaptowanie i poprawienie procesów w celu dopasowania do narzędzia
- Przyrostowe wdrażanie narzędzia do organizacji
- Dostarczanie szkoleń i coachingu/mentoringu dla nowych użytkowników, włączając w to definiowanie przewodników użycia
- Implementowanie sposobu zbierania informacji o używaniu narzędzia przez dotychczasowych użytkowników oraz ich doświadczenia.
- Monitorowanie użycia narzędzi i korzyści

8 Tabele/Rysunki

Tabele

Tabela 1 Cele programu Poziomu Podstawowego, jego korzyści oraz główny nacisk.....	9
Tabela 2 Przegląd czynności, wejść oraz wyjść dla procesów Inżynierii Wymagań	34
Tabela 3 Techniki identyfikacji wymagań.....	68
Tabela 4 Przykładowe metody i modele analizy	76
Tabela 5 Diagramy UML i ich zastosowanie	78

Rysunki

Rysunek 1 Kontekst Inżynierii Wymagań.....	27
Rysunek 2 Podstawowy cykl życia produktu (PLC).....	86

Standardy

IEEE 610.12-1990 IEEE Standard Glossary of Software Engineering Terminology

IEEE 830-1998 IEEE Recommended Practice for Software Requirements Specifications

IEEE 1233-1998 IEEE Guide for Developing System Requirements Specifications

IEEE 1362-1998 IEEE Guide for Information Technology-System Definition – Concept of Operations (ConOps) Document

IEEE 830, IEEE 1233 and IEEE 1362 zostały zastąpione przez ISO/IEC/IEEE 29148:2011

ISO 9000 Quality management

ISO 12207 Systems and software engineering — Software life cycle processes

ISO 15288 System Life Cycle Processes

ISO 15504 Information technology — Process assessment, also known as SPICE (Software Process Improvement and Capability Determination),

ISO 31000 Risk Management - Principles and Guidelines on Implementation

ISO/EIC 25000 (previously ISO/IEC 9126 Software engineering — Product quality)

ISO/IEC 14598-1:1999 Information technology -- Software product evaluation

SWEBOK - The Guide to the Software Engineering Body of Knowledge:

<http://www.computer.org/portal/web/swebok/home>

SEBOK – Systems Engineering Body of Knowledge: <http://www.sebokwiki.org/>

9 Książki i publikacje

Cockburn, A.: *Writing Effective Use Cases*. Amsterdam, 2000

Cohn M.: *Estimating With Use Case Points*. Fall 2005 issue of Methods & Tools

Cohn M.: *User Stories Applied: For Agile Software Development*. Addison-Wesley Professional, 2004

Cotterell, M. and Hughes, B.: *Software Project Management*. International Thomson Publishing, 1995

Davis, A. M.: *Just Enough Requirements Management. Where Software Development Meets Marketing*. Dorset House, 2005, ISBN 0932633641

DeMarco, T.: *Controlling Software Projects: Management, Measurement and Estimates*. Prentice Hall, 1986

Doran, G. T.: *There's a S.M.A.R.T. way to write management's goals and objectives*. Management Review, Volume 70, Issue 11, 1981, (AMA FORUM), pp. 35-36.

Evans, E. J.: *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Amsterdam, 2003

Graham, D. et al: *Foundations of Software Testing*. London, 2007

Gilb, T.; Graham, D.: *Software Inspection*. Reading, MA, 1993

Gilb, T.: *What's Wrong with Requirements Specification*. See: www.gilb.com

Hull, E. et al: *Requirements Engineering*. Oxford, 2005

IQBBA: *Certified Business Analyst, Foundation Level Syllabus*, version 2011

IBUQ: *Certified Professional for Usability Engineering, Foundation Level Syllabus*, version 2011

ISTQB: *ISTQB Glossary of Testing Terms v2.2*

ISTQB: *Certified Tester, Foundation Level Syllabus*, version 2011

Jacobson, I. et al.: *Object-Oriented Software Engineering. A Use Case Driven Approach*. Addison-Wesley, 1993

Lauesen, S.: *Software Requirements: Styles and Techniques*. London, 2002

Pfleeger, S.L. and J.M. Atlee: *Software Engineering: Theory and Practice*. Upper Saddle River, New Jersey, USA, Prentice Hall, 2006.

Project Management Institute: *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*. PMI, 2004

Robertson, S.; Robertson, J.: *Mastering the Requirements Process*. Harlow, 1999

Rupp, C.: *Requirements-Engineering und Management. Professionelle, Iterative Anforderungsanalyse in der Praxis*. Munich, 2007

Sharp H., Finkelstein A. and Galal G.: *Stakeholder Identification in the Requirements Engineering Process*. 1999

Sommerville, I.: *Requirements Engineering*. West Sussex, 2004

Sommerville, I.: *Software Engineering 8*. Harlow, 2007

Sommerville, I.; Sawyer, P.: *Requirements Engineering: A Good Practice Guide*. Chichester, 1997

Sommerville, I.; Kotonya, G.: *Requirements Engineering: Processes and Techniques*. Chichester, 1998

Thayer, R. H.; Dorfman, M.: *Software Requirements Engineering, 2nd edition*. Los Alamitos, CA, 1997

Wieggers, K.E.: *First Things First: Prioritizing Requirements*. Software Development, September, 1999

Wieggers, K. E.: *Software Requirements*. Redmond, 2005

Young, R. R.: *Effective Requirements Practices*. Addison-Wesley, 2001

10 Indeks

- analiza biznesowa, 26, 27, 28, 29
- Analiza Punktów Funkcyjnych, 56, 76
- Analiza Punktów Przypadków Użycia, 56, 76
- analiza ryzyka, 35, 39, 44
- analiza wpływu, 39, 46, 50, 76
- analiza wymagań, 9, 33, 34, 56, 58, 74, 76, 85
- atributy jakościowe wymagań, 54, 71
- BPMN, 56, 79, 81, 85, 98
- burza mózgów, 56, 62, 71, 95
- cel, 60
- CMMI, 21, 22, 25, 89, 95, 96
- cykl życia produktu, 89, 90
- Deweloper Wymagań, 37
- dokumentowanie wymagań, 34
- dostawca, 18, 31, 60, 95
- historijka użytkownika, 73, 84, 93
- historijki użytkownika, 94
- identyfikacja na podstawie istniejących dokumentów, 56, 61
- identyfikacja ryzyka, 39, 44
- identyfikacja wymagań, 33, 56, 58, 59, 66, 70, 76
- IEEE 1233, 19, 24, 103
- IEEE 1362, 24, 103
- IEEE 610, 15, 48, 103
- IEEE 830, 18, 47, 103
- interesariusz, 8, 15, 20, 21, 23, 31, 34, 35, 36, 42, 44, 50, 56, 59, 61, 65, 70, 72, 74, 77, 83, 84, 87, 91, 95, 101
- inżynier wymagań, 31, 37, 38, 65
- Inżynieria Wymagań, 8, 9, 13, 14, 19, 22, 26, 27, 28, 29, 37, 38
- ISO 12207, 25, 103
- ISO 15288, 25, 103
- ISO 15504, 25, 103
- ISO 9000, 24, 52, 103
- ISO 9126, 24, 71
- ISO/IEC 25000, 24, 71, 72
- ISO/IEC/IEEE 29148:2011, 18, 19, 24, 47, 103
- klient, 8, 15, 16, 17, 18, 19, 20, 21, 23, 31, 36, 46, 50, 59, 60, 61, 65, 69, 77, 78, 83, 84, 92, 93, 95, 100
- Komitet Kontroli Zmian, 39, 40, 49, 50
- kontrakt, 16, 42, 60, 87
- kontrola jakości, 9, 39, 52, 54
- kryterium akceptacji, 60, 87
- krytyczność, 13, 18, 19
- kwestionariusz, 56, 61, 63, 75
- łagodzenie ryzyka, 39, 44
- macierz oceny ryzyka, 45
- menedżer wymagań, 31
- metoda delficka, 56
- metryka, 39, 54

Poziom podstawowy

model dojrzałości, 89, 95, 96	prototypy pionowe/wertykalne, 82
model Kano, 19	prototypy scenariuszowe, 82
model koncepcyjny, 56, 78	przedstawiciel klienta, 9, 61
model procesu, 89	przypadki użycia, 21, 56, 62, 69, 93
model rozwiązania, 56, 58, 78	Rational Unified Process, 72, 89, 92
model wymagań, 56, 78	rejestr produktu, 70, 89, 94
modelowanie GUI, 81	rozwiązanie, 8, 13, 15, 16, 17, 28, 36, 42, 49, 57, 60, 70, 71, 78, 85, 87
modelowanie rozwiązania, 77, 85	Rozwijanie Wymagań, 37
narzędzia, 34, 35, 48, 97, 98, 99, 100, 101	ryzyko, 19, 39, 42, 43, 45, 61, 68, 92, 101
obserwacja polowa, 68	ryzyko produktowe, 39, 87
ogólny proces Inżynierii Wymagań, 31, 32, 33	ryzyko projektowe, 39, 42, 53
ograniczenia biznesowe, 13, 16	S.M.A.R.T., 56, 60
ograniczenia techniczne, 13, 16	samo-rejestracja, 61, 65
ograniczenie, 13, 16, 34, 80, 83	SEBOK, 25
plan zarządzania ryzykiem, 39, 44	specyfikacja, 21, 56, 83, 84
POC (ang. <i>proof of concept</i>), 100	specyfikacja formalna, 86
podjęcie iteracyjne, 89, 91, 92	specyfikacja nieformalna, 86
podjęcie sekwencyjne, 89	specyfikacja rozwiązania, 56, 83
podjęcie zwinne, 89, 92, 93	specyfikacja rozwiązania/wymagań, 87
Poker Planistyczny, 56, 75	specyfikacja średnio-formalna, 86
ponowne użycie, 46, 56, 61, 67	specyfikacja wymagań, 21, 22, 33, 34, 36, 53, 56, 57, 58, 67, 83, 84, 85
potrzeby biznesowe, 15, 34	SPICE, 25, 89, 95
priorytet, 13, 18, 19, 20, 53, 70, 76, 77, 93, 94	SWEBOK, 18, 22, 25, 103
priorytetyzacja, 10, 56, 57, 74, 76, 77	SysML, 56, 79, 81
problem biznesowy, 10, 15	system, 13, 15, 16, 17, 20, 28, 36, 43, 49, 50, 53, 59, 61, 69, 71, 72, 74, 76, 78, 79, 80, 81, 85, 86, 87
produkt, 8, 10, 13, 15, 16, 18, 19, 20, 21, 22, 23, 28, 29, 32, 34, 36, 41, 48, 100	śledzenie, 17, 29, 33, 35, 39, 41, 46, 53
projekt pilotażowy, 97, 100	śledzenie pionowe/wertykalne, 39, 47
prototypy horyzontalne/poziome, 82	

Poziom podstawowy

śledzenie poziome/horyzontalne, 39, 47	wymaganie, 13, 15, 17, 18, 20, 21, 29, 72, 87
terminowanie, 56, 62, 69	Wymaganie, 37
testowalność, 87	wywiad, 61, 64, 68, 95
testowanie, 22, 29, 33, 40, 42, 53, 54	zapewnienie jakości, 33, 39, 41, 50, 52, 53
UML, 56, 57, 78, 79, 80, 81, 85, 86, 98	Zarządzający Wymaganiami, 37
walidacja, 13, 21, 22, 52, 87	zarządzanie konfiguracją, 22, 33, 39, 41, 48, 52, 98
walidacja i weryfikacja wymagań, 33, 34, 57, 87	zarządzanie projektem, 22, 29, 39, 42, 52
warsztaty, 56, 62, 70	zarządzanie ryzykiem, 22, 29, 39, 43, 44, 52
weryfikacja, 13, 22, 52, 87	zarządzanie wymaganiami, 8, 10, 13, 14, 22, 35, 39, 41, 48, 65, 94
wizja, 56, 60, 61	Zarządzanie Wymaganiami, 37
Właściciel Produktu, 50, 89, 93, 94	zarządzanie zmianą, 22, 39, 42, 48, 52, 99
wymagania biznesowe, 42, 72, 85	zmiana, 39, 49, 50
wymagania funkcjonalne, 13, 16, 71, 72, 80	zobowiązanie, 13, 18
wymagania нефункционалне, 13, 16, 69, 71, 72	żądanie zmiany, 39, 40, 46
wymagania produktowe, 13, 16	