

Opis

Powodem napisania książki było rosnące zainteresowanie dziedziną inżynierii wymagań i zagadnień z nią bezpośrednio związanych jak modelowanie biznesowe czy zarządzanie projektem.

Istniejące książki i publikacje z reguły koncentrują się na wybranych aspektach czy narzędziach inżynierii wymagań, brak jest natomiast publikacji opisujących całościowo proces inżynierii wymagań, jego kontekst w wytwarzaniu produktu, czynności i ich praktyczne zastosowanie, możliwe ryzyka i sposoby ich uniknięcia.

Książka skierowana jest do osób zawodowo zajmujących się analizą biznesową i systemową, odpowiedzialnych za jakość oprogramowania i systemów oraz architektów czy kierowników projektów, jak również osób pragnących zrozumieć wyzwania związane z inżynierią wymagań i jej powiązania z innymi procesami w ogólnym procesie wytwarzania produktu.

Dowiesz się:

- jak zarządzać wymaganiami w różnych projektach od formalnych po zwinne (ang. Agile)
- jak przekładać język i potrzeby biznesu na ich realizację w systemach informatycznych
- jak łatwo weryfikować możliwość realizacji wymagań przez rozpoczęciem projektu
- jak zadawać pytania klientowi aby uniknąć problemów na końcu projektu

Powinieneś znać:

- podstawowe zagadnienia inżynierii oprogramowania
- podstawowe zagadnienia zarządzania projektami informatycznymi
- realia pracy z klientami i wyzwania z tym związane

Spis treści

Od Autorów

1. Wprowadzenie do inżynierii wymagań
 - 1.1. Wyzwania związane z projektami IT
 - 1.1.1. Cele i wizja
 - 1.1.2. Złe planowanie projektu
 - 1.1.3. Słaba komunikacja
 - 1.1.4. Złe zarządzanie oczekiwaniami interesariuszy
 - 1.1.5. Problemy z wymaganiami i ich zakresem
 - 1.1.6. Brak umiejętności miękkich
 - 1.1.7. Nierealistyczne oczekiwania
 - 1.1.8. Brak zasobów ludzkich
 - 1.1.9. Brak odpowiedniego wsparcia narzędziowego i metodycznego
 - 1.2. Podstawowe definicje oraz klasyfikacje
 - 1.2.1. Wymagania biznesowe
 - 1.2.2. Wymagania interesariuszy
 - 1.2.3. Wymagania rozwiązania
 - 1.2.4. Wymagania przejścia
 - 1.3. Atrybuty wymagań
 - 1.4. Kryteria jakości wymagań

- 1.5. Wymagania w procesie zapewnienia jakości oprogramowania
- 1.6. Inżynieria wymagań oraz jej znaczenie w projekcie
- 1.7. Podstawowe role w procesie inżynierii wymagań
- 1.8. Koncepcja interesariuszy
- 1.9. Standardy oraz normy
 - 1.9.1. ISO 9000
 - 1.9.2. ISO/IEC 25000 - Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE
 - 1.9.3. ISO 9241
 - 1.9.4. ISO 31000: Risk Management
 - 1.9.5. IEEE 610:1990: Standard Glossary of Software Engineering Terminology
 - 1.9.6. IEEE 828-2012: Standard for Configuration Management in Systems and Software Engineering
 - 1.9.7. IEEE 83-1998: Recommended Practice for Software Requirements Specifications
 - 1.9.8. IEEE 1233-1996: Guide for Developing of System Requirements Specifications
 - 1.9.9. IEEE 1362-1998: Guide for Information Technology - System Definition - Concept of Operations (ConOps) Document
 - 1.9.10. IEEE 29148-2011 - Systems and software engineering - Life cycle processes - Requirements engineering
 - 1.9.11. IEEE 1028:2008 Standard for Software Reviews and Audits
 - 1.9.12. SWEBOK: The Guide to the Software Engineering Body of Knowledge (ISO Technical Report 19759)
 - 1.9.13. CMMI
 - 1.9.14. BABOK - A Guide to the Business Analysis Body of Knowledge
- 1.10. Słowniki

2. Proces inżynierii wymagań

- 2.1. Definicja procesu
- 2.2. Inżynieria wymagań a analiza biznesowa
- 2.3. Zasady tworzenia udanych wymagań
 - 2.3.1. Zrozum krytyczne cele najwyższego poziomu
 - 2.3.2. Koncentruj się na dostarczeniu wartości
 - 2.3.3. Zdefiniuj wymaganie jako "stan końcowy o wartości dla interesariusza"
 - 2.3.4. Wyrażaj wymagania ilościowo
 - 2.3.5. Nie mieszaj środków z celami
 - 2.3.6. Skup się na pożądanej jakości systemu, nie tylko na jego funkcjonalności
 - 2.3.7. Zapewnij "bogatą specyfikację"
 - 2.3.8. Wykonuj kontrolę jakości specyfikacji
 - 2.3.9. Uznaj, że wymagania się zmieniają

3. Inżynieria wymagań a inne procesy

- 3.1. Zarządzanie projektem
- 3.2. Zarządzanie ryzykiem
- 3.3. Testowanie i zapewnienie jakości
- 3.4. Wpływ wymagań na inne artefakty projektu

- 4. Inżynieria wymagań w procesach tworzenia oprogramowania
 - 4.1. Model V jako przykład kaskadowego wytwarzania systemów
 - 4.2. IBM Rational Unified Process
 - 4.2.1. Zarządzania wymaganiami w IBM Rational Unified Process
 - 4.2.2. Przepływ prac dla wymagań w IBM Rational Unified Process
 - 4.2.3. Role i artefakty w IBM Rational Unified Process
 - 4.3. Zwinne metodyki w zarządzaniu wymaganiami
 - 4.4. Programowanie ekstremalne
 - 4.5. Serum (według Scrum.org)
 - 4.5.1. Rejestr produktowy, czyli metoda na zorganizowanie wymagań
 - 4.5.2. Wyzwania związane z migracją do Serum
 - 4.6. Disciplined Agile Delivery
 - 4.7. Przypadek biznesowy
 - 4.7.1. Informacja o firmie i sytuacja rynkowa
 - 4.7.2. Potrzeba
 - 4.7.3. Rozwiązanie
 - 4.7.4. Zyski

- 5. Identyfikacja wymagań
 - 5.1. Źródła wymagań
 - 5.2. Wizja oraz cel projektu
 - 5.3. Identyfikacja interesariuszy projektu
 - 5.4. Techniki identyfikacji wymagań
 - 5.4.1. Warsztat wymagań
 - 5.4.2. Wywiad
 - 5.4.3. Ankieta - kwestionariusz
 - 5.4.4. Samodzielna rejestracja
 - 5.4.5. Reprezentant klienta po stronie dostawcy
 - 5.4.6. Identyfikacja na podstawie istniejących dokumentów
 - 5.4.7. Ponowne użycie specyfikacji
 - 5.4.8. Obserwacja w terenie
 - 5.4.9. Mentorowanie/praktykowanie
 - 5.4.10. Burza mózgów
 - 5.4.11. Prototypowanie
 - 5.4.12. Przypadki użycia
 - 5.4.13. Scenorys
 - 5.5. Wymagania funkcjonalne i нефункционалне

- 6. Analiza wymagań
 - 6.1. Analiza problemu biznesowego
 - 6.2. Organizacja wymagań
 - 6.3. Powiązania i zależności między wymaganiami
 - 6.4. Usuwanie konfliktów i duplikatów wymagań
 - 6.5. Kontrola jakości
 - 6.6. Szacowanie wysiłku

- 6.6.1. Techniki bazujące na algorytmach
- 6.6.2. Techniki bazujące na przybliżeniach
- 6.7. Priorytetyzacja wymagań
- 6.8. Modelowanie rozwiązania
 - 6.8.1. Model dziedziny
 - 6.8.2. Diagram przepływu danych (ang. Data Flow Diagram)
 - 6.8.3. Diagram związków encji (ang. Entity Relationship Diagram)
 - 6.8.4. Modelowanie interfejsu użytkownika
 - 6.8.5. Unified Modeling Language (UML)
 - 6.8.6. System Modeling Language (SysML)
 - 6.8.7. Inne notacje do modelowania
- 6.9. Akceptacja wymagań

7. Specyfikacja wymagań

- 7.1. Pojęcie specyfikacji
- 7.2. Rodzaje specyfikacji
 - 7.2.1. Specyfikacja wymagań
 - 7.2.2. Specyfikacja rozwiązania
 - 7.2.3. Specyfikacja techniczna
- 7.3. Szablony dla specyfikacji wymagań (na podstawie IEEE 830)
 - 7.3.1. IEEE 830
 - 7.3.2. Wzorzec Volere
 - 7.3.3. Historyjki użytkownika
 - 7.3.4. Przypadki użycia jako sposób na wymagania funkcjonalne
- 7.4. Jakość specyfikacji wymagań

Rozdział 8. Zarządzanie wymaganiami

- 8.1. Śledzenie wymagań
- 8.2. Zarządzanie Konfiguracją
- 8.3. Zarządzanie Zmianą
- 8.4. Zarządzanie wymaganiami dotyczącymi projektu oraz systemu
- 8.5. Plan Zarządzania Wymaganiami
- 8.6. Przypadek biznesowy - wdrożenie procesu zarządzania wymaganiami
 - 8.6.1. Informacja o firmie i sytuacja rynkowa
 - 8.6.2. Potrzeba
 - 8.6.3. Rozwiązanie
 - 8.6.4. Zyski

9. Wymagania a zarządzanie jakością

- 9.1. Planowanie jakości
- 9.2. Kontrola jakości
 - 9.2.1. Przeglądy
 - 9.2.2. Inspekcje
 - 9.2.3. Listy kontrolne
- 9.3. Miary jakości wymagań

9.4. Doskonalenie procesu

10. Narzędzia wspierające proces inżynierii wymagań

10.1. Narzędzia w zarządzaniu wymaganiami

10.1.1. IBM Rational Requirements Composer

10.1.2. Borland Caliber RM

10.1.3. Serene Dimensions

10.1.4. Rational DOORS (Dynamie Object Oriented Requirements System)

10.1.5. Blueprint Requirements Center

10.1.6. Open Source Requirements Management Tool/aNimble Platform

10.1.7. Cechy dobrego narzędzia do zarządzania wymaganiami

10.1.8. Wdrożenie narzędzia do zarządzania wymaganiami

10.2. Czynniki mające znaczenie przy doborze odpowiednich narzędzi

10.3. Narzędzia w obszarze modelowaniu wymagań

10.3.1. Sparx Enterprise Architect

10.3.2. IBM Rational Software Architect

10.3.3. StarUML

10.4. Narzędzia w obszarze modelowania procesów biznesowych

10.4.1. Boc Group Adonis

10.4.2. iGrafx Process

10.4.3. BizAgi Process Modeler

10.4.4. Rational System Architect

10.5. Narzędzia wsparcia dla zarządzania konfiguracją

10.5.1. GIT

10.5.2. Subversion

10.5.3. IBM ClearCase

10.6. Narzędzia wsparcia dla zarządzania zmianami

10.6.1. Atlassian Jira

10.6.2. IBM Rational Team Concert

10.7. Zarządzanie procesem testowania oprogramowania

10.7.1. HP Quality Center

10.7.2. IBM Rational Quality Manager

10.7.3. Testia Tarantula

10.7.4. Requirements Testing Hub

10.7.5. TestLink

10.8. Ryzyko związane ze złym zakupem narzędzia

Podsumowanie

Przypadki biznesowe

Projekt 1 - Wdrażanie procesu inżynierii wymagań

Informacja o firmie i sytuacja rynkowa

Potrzeba

Rozwiązanie

Zyski

Projekt 2 - Integracja narzędzi w procesie wytwarzania

Informacja o firmie i sytuacja rynkowa

Potrzeba

Rozwiązanie

Etap 1 - Integracja wymagań z procesem zarządzania testami

Etap 2 - Integracja wymagań z zarządzaniem konfiguracją

Etap 3 - Integracja wymagań z zarządzaniem zmianami

Zyski

Projekt 3 - Kontrola jakości wymagań na wczesnych etapach projektu

Informacja o firmie i sytuacja rynkowa

Potrzeba

Skutek

Przyczyna

Rozwiązanie

Projekt 4 - Zarządzanie wymaganiami przy użyciu historyjek użytkownika

Informacja o firmie i sytuacja rynkowa

Potrzeba

Rozwiązanie

Zyski

Bibliografia

Spis rysunków

Spis tabel

Indeks