

# Architektura oprogramowania w praktyce. Wydanie II.

Autorzy: Len Bass, Paul Clements, Rick Kazman

## Twórz doskonale projekty architektoniczne oprogramowania!

- Czym charakteryzuje się dobra architektura oprogramowania?
- Jak przebiega proces jej projektowania?
- Jak ją dokumentować?

Współczesne systemy informatyczne to zaawansowane, skomplikowane mechanizmy, składające się z wielu współdziałających ze sobą komponentów. Ich wyodrębnienie, a także określenie sposobu komunikacji i interakcji między poszczególnymi elementami, jest nie lada wyzwaniem dla architektów. Od ich decyzji zależy, czy system uda się zrealizować, czy będzie on efektywny, stabilny i łatwy w utrzymaniu.

Na szczęście istnieją metodologie, narzędzia oraz sposoby analizy efektów ułatwiające i porządkujące cały ten proces. **W tej książce znajdziesz wszystko, o czym trzeba pamiętać przy projektowaniu oprogramowania.** Poznasz sposoby projektowania z wykorzystaniem Metody Analizy Kompromisów w Architekturze (ATAM) oraz oceniania aspektów finansowych przy użyciu Metody Analizy Kosztów i Korzyści (CBAM). Autorzy przedstawia wiele studiów przypadków, które pozwolą Ci na zapoznanie się z rzeczywistymi problemami i ich rozwiązaniami. Ponadto nauczysz się stosować język UML do wizualnej reprezentacji architektury systemu oraz zobaczysz, jak przygotować dobrą dokumentację projektu. Książka ta sprawdzi się idealnie w rękach każdego architekta oprogramowania.

- Proces wytwarzania oprogramowania a cykl biznesowy architektury
- Wzorce architektury
- Struktury i perspektywy architektury
- Określenie i uzyskanie atrybutów jakościowych
- Projektowanie architektury pod kątem wysokiej dostępności
- Proces projektowania architektury
- Dokumentowanie architektury oprogramowania
- Język UML
- Metody rekonstrukcji architektury i inżynierii odwrotnej
- Metoda Analizy Kompromisów w Architekturze (ATAM)
- Metoda Analizy Kosztów i Korzyści (CBAM)
- Ponowne wykorzystanie elementów architektury
- Dokumentowanie architektury

## Poznaj najlepsze metodologie projektowania architektury!

**Przedmowa (9)**

**Podziękowania (13)**

**Wstęp (15)**

**I. Wizja architektury (21)**

**1. Cykl biznesowy architektury (23)**

- 1.1. Skąd się biorą architektury? (26)
- 1.2. Proces wytwarzania oprogramowania a cykl biznesowy architektury (31)
- 1.3. Czym się charakteryzuje dobra architektura? (33)
- 1.4. Podsumowanie (35)
- 1.5. Pytania do dyskusji (35)

## **2. Czym jest architektura oprogramowania? (37)**

- 2.1. Czym jest, a czym nie jest architektura oprogramowania? (37)
- 2.2. Inne perspektywy (40)
- 2.3. Wzorce architektury, modele referencyjne i architektury referencyjne (41)
- 2.4. Dlaczego architektura jest tak ważna? (43)
- 2.5. Struktury i perspektywy architektury (50)
- 2.6. Podsumowanie (56)
- 2.7. Literatura (57)
- 2.8. Pytania do dyskusji (59)

## **3. System awioniki A-7E - studium wykorzystania struktur architektury (61)**

- 3.1. Położenie w cyklu biznesowym architektury (62)
- 3.2. Wymagania i atrybuty jakościowe (62)
- 3.3. Architektura systemu awioniki A-7E (67)
- 3.4. Podsumowanie (78)
- 3.5. Literatura (79)
- 3.6. Pytania do dyskusji (79)

## **II. Tworzenie architektury (81)**

### **4. Atrybuty jakościowe (83)**

- 4.1. Architektura a funkcje systemu (84)
- 4.2. Architektura a atrybuty jakościowe (84)
- 4.3. Atrybuty jakościowe systemu (85)
- 4.4. Scenariusze atrybutów jakościowych w praktyce (89)
- 4.5. Inne atrybuty jakościowe systemu (103)
- 4.6. Biznesowe atrybuty jakościowe (103)
- 4.7. Atrybuty jakościowe architektury (104)
- 4.8. Podsumowanie (105)
- 4.9. Literatura (105)
- 4.10. Pytania do dyskusji (106)

### **5. Uzyskiwanie atrybutów jakościowych (107)**

- 5.1. Taktyki atrybutów jakościowych (107)
- 5.2. Taktyki dostępności (109)
- 5.3. Taktyki modyfikowalności (112)
- 5.4. Taktyki wydajności (118)
- 5.5. Taktyki bezpieczeństwa (122)
- 5.6. Taktyki testowalności (124)
- 5.7. Taktyki funkcjonalności (126)
- 5.8. Taktyki atrybutów jakościowych a wzorce architektury (128)

- 5.9. Wzorce i style architektury (129)
- 5.10. Podsumowanie (130)
- 5.11. Pytania do dyskusji (131)
- 5.12. Literatura (131)

## **6. Kontrola ruchu lotniczego - projektowanie pod kątem wysokiej dostępności (133)**

- 6.1. Powiązania w cyklu biznesowym architektury (135)
- 6.2. Wymagania i atrybuty jakościowe (135)
- 6.3. Architektura systemu (138)
- 6.4. Podsumowanie (152)
- 6.5. Literatura (152)
- 6.6. Pytania do dyskusji (153)

## **7. Projektowanie architektury (155)**

- 7.1. Architektura w cyklu życia oprogramowania (155)
- 7.2. Projektowanie architektury (157)
- 7.3. Kształtowanie struktury zespołów (167)
- 7.4. Tworzenie systemu szkieletowego (170)
- 7.5. Podsumowanie (171)
- 7.6. Literatura (172)
- 7.7. Pytania do dyskusji (173)

## **8. Symulator lotniczy - architektura ukierunkowana na łatwość integracji (175)**

- 8.1. Powiązania w cyklu biznesowym architektury (176)
- 8.2. Wymagania funkcjonalne i jakościowe (177)
- 8.3. Architektura (180)
- 8.4. Podsumowanie (193)
- 8.5. Literatura (195)
- 8.6. Pytania do dyskusji (195)

## **9. Dokumentacja architektury oprogramowania (197)**

- 9.1. Funkcje dokumentacji (198)
- 9.2. Perspektywy architektury (200)
- 9.3. Wybieranie perspektyw architektury (201)
- 9.4. Opisywanie perspektywy architektury (202)
- 9.5. Ogólna część dokumentacji (208)
- 9.6. Zunifikowany język modelowania - UML (211)
- 9.7. Podsumowanie (220)
- 9.8. Literatura (221)
- 9.9. Pytania do dyskusji (221)

## **10. Rekonstrukcja architektury oprogramowania (223)**

- 10.1. Wprowadzenie (223)
- 10.2. Ekstrakcja informacji (226)
- 10.3. Budowanie bazy danych (228)

- 10.4. Scalanie informacji (230)
- 10.5. Rekonstrukcja (232)
- 10.6. Przykład (237)
- 10.7. Podsumowanie (245)
- 10.8. Literatura (245)
- 10.9. Pytania do dyskusji (246)

### **III. Analiza i weryfikacja architektury (247)**

#### **11. ATAM - kompleksowa metoda analizy architektury (253)**

- 11.1. Uczestnicy procesu ATAM (253)
- 11.2. Materiały wyjściowe procesu ATAM (255)
- 11.3. Fazy procesu ATAM (256)
- 11.4. Studium przypadku - weryfikacja metodą ATAM systemu Nightingale (267)
- 11.5. Podsumowanie (281)
- 11.6. Literatura (282)
- 11.7. Pytania do dyskusji (282)

#### **12. CBAM - ilościowe podejście do decyzji konstrukcyjnych (283)**

- 12.1. Kontekst podejmowania decyzji (284)
- 12.2. Podstawy metody CBAM (285)
- 12.3. Stosowanie metody CBAM (289)
- 12.4. Studium przypadku - projekt ECS w agencji NASA (291)
- 12.5. Rezultaty analizy CBAM (298)
- 12.6. Podsumowanie (299)
- 12.7. Literatura (299)
- 12.8. Pytania do dyskusji (299)

#### **13. Współdziałanie w World Wide Web - studium przypadku (301)**

- 13.1. Powiązania z cyklem biznesowym architektury (301)
- 13.2. Wymagania funkcjonalne i atrybuty jakościowe (303)
- 13.3. Architektura (307)
- 13.4. Nowy cykl ABC - ewolucja architektur handlu elektronicznego w WWW (313)
- 13.5. Wymagania jakościowe (317)
- 13.6. Współczesny cykl biznesowy architektury (318)
- 13.7. Podsumowanie (319)
- 13.8. Literatura (320)
- 13.9. Pytania do dyskusji (321)

### **IV. Od jednego systemu do wielu (323)**

#### **14. Rodziny produktów - ponowne użycie elementów architektury (325)**

- 14.1. Wprowadzenie (325)
- 14.2. Co sprawia, że linia oprogramowania jest udana? (326)
- 14.3. Określanie zakresu (328)
- 14.4. Architektury linii produktów (331)
- 14.5. Co sprawia, że rozwijanie linii oprogramowania jest trudne? (334)
- 14.6. Podsumowanie (337)

- 14.7. Literatura (337)
- 14.8. Pytania do dyskusji (337)

### **15. CelsiusTech - studium przypadku rodziny produktów (339)**

- 15.1. Związki z cyklem ABC (339)
- 15.2. Wymagania i atrybuty jakościowe (355)
- 15.3. Rozwiązanie architektoniczne (357)
- 15.4. Podsumowanie (364)
- 15.5. Literatura (365)
- 15.6. Pytania do dyskusji (365)

### **16. J2EE/EJB. Studium przypadku - standardowa dla branży infrastruktura obliczeniowa (367)**

- 16.1. Związki z cyklem biznesowym architektury (368)
- 16.2. Wymagania i atrybuty jakościowe (368)
- 16.3. Rozwiązanie architektoniczne (371)
- 16.4. Decyzje związane z wdrażaniem systemu (383)
- 16.5. Podsumowanie (388)
- 16.6. Literatura (388)
- 16.7. Pytania do dyskusji (388)

### **17. Architektura Luther. Studium przypadku - aplikacje przenośne oparte na J2EE (389)**

- 17.1. Związki z cyklem ABC (390)
- 17.2. Wymagania i atrybuty jakościowe (393)
- 17.3. Rozwiązanie architektoniczne (396)
- 17.4. Jak w architekturze Luther zrealizowano cele z obszaru jakości? (410)
- 17.5. Podsumowanie (410)
- 17.6. Literatura (411)
- 17.7. Pytania do dyskusji (411)

### **18. Budowanie systemów z gotowych komponentów (413)**

- 18.1. Wpływ komponentów na architekturę (415)
- 18.2. Niedopasowanie architektury (416)
- 18.3. Budowa z gotowych komponentów jako proces poszukiwań (421)
- 18.4. Przykład - system ASEILM (424)
- 18.5. Podsumowanie (433)
- 18.6. Literatura (433)

### **19. Przyszłość architektury oprogramowania (435)**

- 19.1. Cykl biznesowy architektury (436)
- 19.2. Budowa architektury (437)
- 19.3. Architektura w cyklu życia oprogramowania (438)
- 19.4. Wpływ komponentów komercyjnych (439)
- 19.5. Podsumowanie (441)

**Skróty (443)**  
**Bibliografia (449)**  
**Skorowidz (455)**