**Nazwa przedmiotu:**

Architektura i integracja systemów

**Koordynator przedmiotu:**

Rajmund Kożuszek

**Status przedmiotu:**

Fakultatywny dowolnego wyboru

**Poziom kształcenia:**

Studia II stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Przedmioty techniczne - zaawansowane

**Kod przedmiotu:**

AIS

**Semestr nominalny:**

2 / rok ak. 2021/2022

**Liczba punktów ECTS:**

4

**Liczba godzin pracy studenta związanych z osiągnięciem efektów uczenia się:**

1. liczba godzin kontaktowych – 55 godz., w tym
obecność na wykładach 30 godz.,
obecność na zajęciach projektowych 15 godz.,
udział w konsultacjach 10 godz.
2. praca własna studenta – 50 godz., w tym
przygotowanie do projektu 25 godz.,
przygotowanie do kolokwium 25 godz.
Łączny nakład pracy studenta wynosi 105 godz., co odpowiada 4 pkt. ECTS.

**Liczba punktów ECTS na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich:**

2,09 pkt. ECTS, co odpowiada 55 godz. kontaktowym

**Język prowadzenia zajęć:**

polski

**Liczba punktów ECTS, którą student uzyskuje w ramach zajęć o charakterze praktycznym:**

1,52 pkt. ECTS, co odpowiada 25 godz. realizacji projektu plus 15 godz. spotkań projektowych

**Formy zajęć i ich wymiar w semestrze:**

|  |  |
| --- | --- |
| Wykład: | 30h |
| Ćwiczenia: | 0h |
| Laboratorium: | 0h |
| Projekt: | 15h |
| Lekcje komputerowe: | 0h |

**Wymagania wstępne:**

Znajomość UML. Umiejętność projektowania prostych elementów systemów informatycznych – modułów i interfejsów oprogramowania.

**Limit liczby studentów:**

30

**Cel przedmiotu:**

Celem przedmiotu jest przedstawienie problemu integracji złożonych systemów na poziomie architektonicznym, miejsca i roli systemów informatycznych w architekturze korporacyjnej oraz sposobu wykorzystania architektury usługowej. Wykład przedstawia też cele i korzyści wynikające z zarządzania architekturą złożonych systemów oraz metody modelowania decyzji architektonicznych.

**Treści kształcenia:**

Wykład:
1. Architektura oprogramowania (4h).
a. Udziałowcy, widoki i punkty widzenia.
b. Dokumentowanie architektury zgodnie ze standardem IEEE 42010.
c. Modularyzacja i jej wpływ na właściwości oprogramowania.
d. Rola architektury i architekta.
2. Metody oceny architektury (4h).
a. Wpływ architektury na atrybuty jakości oprogramowania.
b. Ocena architektury metodą analizy kompromisów architektonicznych.
c. Inne metody oceny architektury oprogramowania.
3. Projektowanie architektury(4h) oprogramowania i systemów IT.
a. Style architektoniczne i wzorce projektowe.
b. Typowe architektury systemów informatycznych – scentralizowana, warstwowa, rozproszona, bezpostaciowa.
c. Obliczenia w chmurze.
d. Podstawowe problemy i dylematy architektoniczne.
4. Decyzje i wiedza architektoniczna(3h)
a. Decyzje architektoniczne i ich dokumentowanie.
b. Modelowanie i języki opisu architektury (Architecture Description Language – ADL).
c. Trendy rozwojowe w dziedzinie architektury.
5. Architektura korporacyjna (4h).
a. Poziomy architektury: architektura korporacyjna, architektura systemów IT, architektura systemu, architektura oprogramowania.
b. Architektura korporacyjna – cele, narzędzia, udziałowcy, procesy. Rola architekta w procesie wytwarzania i rozwoju systemów informatycznych.
c. Pojęcia i standardy – metoda TOGAF, siatka Zachmana. Architektury referencyjne (eTOM, SID, DoDAF).
d. Język modelowania architektury korporacyjnej Archimate.
6. Integracja systemów (Enterprise Integration) (4h).
a. Rozwój korporacji, systemy dziedziczone (legacy), integracja systemów,
b. Korporacyjna szyna usług (Enterprise Service Bus – ESB).
c. Brokery.
d. Systemy wysokoskalowalne.
e. Problemy integracji systemów wysokoskalowalnych.
f. Wybrane wzorce i technologie integracyjne.
7. Architektura usługowa (Service Oriented Architecture – SOA) (3h).
a. Usługi sieciowe, protokoły komunikacyjne
b. Orkiestracja i choreografia usług.
c. Proces biznesowy i jego implementacja, język BPEL.
d. Architektura usług sieciowych.
e. Modelowanie i projektowania architektury usługowej (SOMA, SOMF, SOAD).
8. Architektura mikroserwisów (Microservices)(3h).
a. Koncepcja mikroserwisów.
b. Wybrane problemy architektury mikroserwisów.
c. Wzorce architektonicznie przeznaczone dla mikroserwisów.
d. Wybrane rozwiązania technologiczne wspierające architekturę mikroserwisów.
e. Wzorce i antywzorce mikroserwisowe.
Projekt:
Celem projektu jest praktyczne zapoznanie z problematyką projektowania architektury. W trakcie projektu słuchacze tworzą architekturę przykładowego systemu, przy użyciu wybranego języka opisu architektury (ADL). Założeniem koncepcyjnym projektu architektury jest wykorzystanie architektury usługowej oraz integracji złożonych systemów.

**Metody oceny:**

W ramach realizacji przedmiotu studenci opracowują projekt architektury wybranego, złożonego systemu informatycznego. Projekt jest realizowany w zespołach 4-osobowych. Studenci mają za zadanie:
● zebrać szczegółowe wymagania
● zidentyfikować kluczowych interesariuszy
● zaprojektować architekturę w kilku wybranych widokach architektonicznych
● przedyskutować wybrane wzorce i taktyki architektoniczne
● przedyskutować podjęte decyzje architektoniczne
● przedstawić wyniki projektu w zwartej i czytelnej formie

**Egzamin:**

nie

**Literatura:**

● Bass, Len, Paul Clements, and Rick Kazman. Software architecture in practice. Addison-Wesley Professional, 2003.
● Clements, Paul, et al. "Documenting software architectures: views and beyond." 25th International Conference on Software Engineering, 2003. Proceedings.. IEEE, 2003.
● Lankhorst, Marc. Enterprise architecture at work. Vol. 352. Berlin: Springer, 2009.
● Tanenbaum, Andrew S., and Maarten Van Steen. Distributed systems: principles and paradigms. Prentice-Hall, 2007.
● Narzędzie modelowania w języku Archimate: https://www.archimatetool.com/

**Witryna www przedmiotu:**

https://usosweb.usos.pw.edu.pl/kontroler.php?\_action=katalog2/przedmioty/pokazPrzedmiot&prz\_kod=103A-INxxx-MSP-AIS

**Uwagi:**

(-)

## Charakterystyki przedmiotowe