



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Technologie dla aplikacji klasy enterprise [S2Inf1-TPD>ENTERP]

Przedmiot

Kierunek studiów
Informatyka

Rok/Semestr
1/1

Studia w zakresie (specjalność)
Technologie przetwarzania danych

Profil studiów
ogólnoakademicki

Poziom studiów
drugiego stopnia

Język oferowanego przedmiotu
polski

Forma studiów
stacjonarne

Wymagalność
obligatoryjny

Liczba godzin

Wykład
30

Laboratorium
30

Inne
0

Ćwiczenia
0

Projekty/seminaria
0

Liczba punktów ECTS

5,00

Koordynatorzy

dr hab. inż. Marek Wojciechowski prof. PP
marek.wojciechowski@put.poznan.pl

Wykładowcy

dr hab. inż. Marek Wojciechowski prof. PP
marek.wojciechowski@put.poznan.pl

Wymagania wstępne

Efekty uczenia się ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_W1-2, K_W4, K_W6-15, weryfikowane w procesie rekrutacji na studia 2 stopnia – efekty te prezentowane są w serwisie internetowym wydziału. Efekty uczenia się ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_U1-2, K_U4, K_U7-8, K_U14-20, K_U22-23, K_U26, weryfikowane w procesie rekrutacji na studia 2 stopnia – efekty te prezentowane są w serwisie internetowym wydziału. Efekty uczenia się ze studiów I stopnia zdefiniowane w Uchwale Senatu PP, a szczególnie efekty K_K1-9, weryfikowane w procesie rekrutacji na studia 2 stopnia – efekty te prezentowane są w serwisie internetowym wydziału. Ponadto w zakresie kompetencji społecznych student musi prezentować takie postawy jak uczciwość, odpowiedzialność, wytrwałość, ciekawość poznawcza, kreatywność, kultura osobista, szacunek dla innych ludzi.

Cel przedmiotu

1. Przekazanie studentom podstawowej wiedzy dotyczącej tworzenia nowoczesnych aplikacji korporacyjnych w architekturze internetowej ze szczególnym naciskiem na aplikacje komponentowe i wielowarstwowe, w zakresie najważniejszych platform do tworzenia tego typu aplikacji: Java Enterprise Edition (Java EE / Jakarta EE) i Microsoft .NET, ale również z uwzględnieniem istotnych rozwiązań z ekosystemu języka JavaScript, głównie frontendowych. 2. Rozwijanie u studentów umiejętności rozwiązywania problemów związanych z tworzeniem nowoczesnych aplikacji korporacyjnych w architekturze komponentowej i wielowarstwowej oraz wyboru odpowiednich technologii dla tworzonego systemu i rozwiązywania problemów.

Przedmiotowe efekty uczenia się

Wiedza:

1. ma zaawansowaną i pogłębioną wiedzę z zakresu aplikacji wielowarstwowych i rozproszonych, wzorców architektonicznych dotyczących tego typu aplikacji, narzędzi, frameworków, bibliotek i środowisk programistycznych wykorzystywanych do ich implementacji (k2st_w1)
2. ma zaawansowaną wiedzę szczegółową dotyczącą wybranych zagadnień z zakresu aplikacji wielowarstwowych i rozproszonych (k2st_w3)
3. ma zaawansowaną i szczegółową wiedzę o procesach zachodzących w cyklu życia aplikacji internetowych (k2st_w5)
4. zna zaawansowane metody, techniki i narzędzia stosowane przy tworzeniu złożonych aplikacji internetowych klasy enterprise (k2st_w6)

Umiejętności:

1. potrafi pozyskiwać informacje z dokumentacji dotyczącej technologii, frameworków i narzędzi do tworzenia aplikacji internetowych (w języku polskim i angielskim), analizując ich silne i słabe strony i oceniając adekwatność w kontekście tworzonego systemu informatycznego (k2st_u1)
2. potrafi przy tworzeniu zaawansowanych aplikacji korporacyjnych integrować wiedzę z dziedzin baz danych, technologii internetowych, systemów rozproszonych, a także zasad projektowania interfejsu użytkownika (k2st_u5)
3. potrafi ocenić przydatność i możliwość wykorzystania nowych technologii, frameworków i narzędzi do tworzenia aplikacji klasy enterprise (k2st_u6)
4. potrafi dokonać krytycznej analizy istniejących wzorców architektonicznych, technologii i frameworków stosowanych w kontekście aplikacji klasy enterprise i wskazać możliwości ich ulepszenia (k2st_u8)
5. potrafi ocenić przydatność metod i narzędzi służących do rozwiązania zadania inżynierskiego, polegającego na budowie aplikacji internetowej klasy enterprise, w tym dostrzec ograniczenia tych metod i narzędzi (k2st_u9)
6. potrafi, zgodnie z zadaną specyfikacją, uwzględniającą aspekty pozatechniczne, zaprojektować aplikację internetową używając metod, technik i narzędzi właściwych dla aplikacji klasy enterprise (k2st_u11)

Kompetencje społeczne:

1. rozumie, że w obszarze aplikacji internetowych i rozproszonych wiedza i umiejętności bardzo szybko stają się przestarzałe (k2st_k1)
2. rozumie znaczenie wykorzystywania najnowszych architektur, wzorców i technologii do tworzenia aplikacji klasy enterprise (k2st_k2)

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca:

a) w zakresie wykładów:

- na podstawie odpowiedzi na pytania dotyczące materiału omówionego na poprzednich wykładach,

b) w zakresie laboratoriów:

- na podstawie oceny bieżącego postępu realizacji zadań,

Ocena podsumowująca:

a) w zakresie wykładów weryfikowanie założonych efektów kształcenia realizowane jest przez:

- ocenę wiedzy i umiejętności wykazanych na sprawdzianie pisemnym w formie testu (test obejmuje 16 równo punktowanych pytań, na ocenę 3.0 konieczne jest uzyskanie co najmniej połowy możliwych do

uzyskania punktów)

- aktywny udział w wykładach premiowany dodatkowymi punktami na sprawdzianie,
 - b) w zakresie laboratoriów weryfikowanie założonych efektów kształcenia realizowane jest przez:
 - ocenę realizacji zadań w formie ćwiczeń programistycznych przy komputerach,
 - ocenę i obronę przez studenta realizowanych samodzielnie projektów,
- Uzyskiwanie punktów dodatkowych za aktywność podczas zajęć, a szczególnie za:
- omówienie dodatkowych aspektów zagadnienia,
 - uwagi związane z udoskonaleniem materiałów dydaktycznych.

Treści programowe

Program modułu obejmuje następujące zagadnienia:

1. Architektury współczesnych aplikacji internetowych.
2. Platformy programistyczne, technologie i frameworki dla aplikacji internetowych klasy enterprise.
3. Współpraca aplikacji internetowych z bazami danych.

Tematyka zajęć

Program wykładu obejmuje następujące zagadnienia:

Wprowadzenie do aplikacji wielowarstwowych, architektura wielowarstwowa a wzorzec Model-View-Controller (MVC), architektura monolityczna a mikroserwisy. Platforma Java Enterprise Edition (Java EE / Jakarta EE) i jej technologie składowe. Przegląd technologii warstwy prezentacji i warstwy logiki biznesowej platformy Java EE. Serwlety i JavaServer Pages (JSP) jako klasyczne technologie prezentacji. JavaServer Faces (JSF) jako technologia warstwy prezentacji i implementacja wzorca MVC. Enterprise JavaBeans (EJB) jako klasyczna technologia do implementacji logiki biznesowej: rodzaje komponentów EJB, klienci komponentów EJB, zarządzanie transakcjami w technologii EJB. Architektura Message-Oriented Middleware, komunikacja asynchroniczna w aplikacjach Java EE, standard Java Message Service (JMS), komunikatowe komponenty EJB. Odzworowanie obiektowo-relacyjne jako podstawowe obecnie podejście do komunikacji z bazami danych SQL, przegląd technologii odpowiedzialnych za komunikację z bazą danych na platformie Java EE, standard Java Persistence API (JPA). Integracja warstw aplikacji Java EE, technologia Contexts and Dependency Injection (CDI). Wzorce projektowe wykorzystywane na platformie Java EE. Framework Spring i związany z nim ekosystem projektów (w tym Spring Boot). Platforma Microsoft .NET i jej podstawowe założenia architektoniczne i technologie składowe. Microsoft .NET Core. Frameworki ASP.NET MVC oraz ASP.NET Web API. Komunikacja z bazą danych w aplikacjach Microsoft .NET, zapytania LINQ, technologia ADO.NET Entity Framework jako implementacja koncepcji odzworowania obiektowo-relacyjnego. Model Ajax aplikacji internetowych, jego zalety i wady, przegląd rozwiązań z nim związanych. Architektura Single Page Application (SPA). Framework Angular. Usługi Web Services typu SOAP i REST, podstawowe standardy, implementacja usług i klientów na platformach Java EE i Microsoft .NET.

Program laboratorium obejmuje następujące zagadnienia:

Platforma Java Enterprise Edition (Java EE) i jej technologie składowe: implementacja i konfiguracja serwletów, tworzenie prostych stron JavaServer Pages (JSP), projektowanie i implementacja interfejsu użytkownika w technologii JavaServer Faces (JSF), implementacja logiki biznesowej w oparciu o sesyjne komponenty Enterprise JavaBeans (EJB) oraz komponenty Contexts and Dependency Injection (CDI), komunikacja z bazą danych w oparciu o odzworowanie obiektowo-relacyjne w technologii Java Persistence API (JPA), implementacja komunikacji asynchronicznej w oparciu o komunikatowe komponenty EJB. Tworzenie aplikacji Java z wykorzystaniem frameworka Spring. Platforma Microsoft .NET i jej technologie składowe: tworzenie aplikacji wielowarstwowych przy użyciu frameworka ASP.NET MVC z obsługą komunikacji z bazą danych w technologii ADO.NET Entity Framework. Usługi Web Services typu RESTful na platformach Java EE i Microsoft .NET. Model Ajax aplikacji internetowych. Wykorzystanie frameworków Angular, React i Vue do implementacji aplikacji w architekturze SPA.

Metody dydaktyczne

Wykład: prezentacja multimedialna, dyskusja.

Ćwiczenia laboratoryjne: ćwiczenia praktyczne, dyskusja, demonstracja, samodzielnie realizowane projekty.

Literatura

Podstawowa

1. Java Platform, Enterprise Edition, The Java EE Tutorial, Release 8, Oracle, 2017
2. Microsoft Docs, www.msdn.com
3. Spring Guides, <https://spring.io/guides>
4. Angular Docs, <https://angular.io/docs>

Uzupełniająca

1. Oracle ADF i JBoss Seam - dwa skrajnie różne podejścia do współpracy JSF z EJB, B. Mordaka, M. Wojciechowski, Materiały XV konf. PLOUG, 2009
2. Co nowego w Java EE?, M. Wojciechowski, Materiały XII konf. PLOUG, 2006
3. Java EE 8. Wzorce projektowe i najlepsze praktyki, R. Rocha, J. Purificação, 2019
4. Architectural Styles and the Design of Network-based Software Architectures, R. Fielding, University of California, 2000
5. Ajax w akcji, D. Crane, E. Pascarello, D. James, Helion, 2007
6. ASP.NET Core MVC 2. Zaawansowane programowanie. Wydanie VII, A. Freeman, Helion, 2018.
7. Microsoft .NET: Architecting Applications for the Enterprise, D. Esposito, A. Saltarello, Microsoft Press, 2008
8. Web Services: Principles and Technology, M. Papazoglou, Prentice Hall, 2007
9. Spring w Akcji. Wydanie V, Craig Walls, Helion, 2019
10. Angular w akcji, Jeremy Wilken, Helion, 2019

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	125	5,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	60	2,50
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	65	2,50