

---

## SZCZEGÓŁOWA INFORMACJA O PRZEDMIOCIE

---

### Wersja przedmiotu

<b>Kod przedmiotu:</b>	<b>PIK</b>
<b>Nr wersji:</b>	<b>1</b>
<b>Nazwa:</b>	Wytwarzanie aplikacji internetowych i korporacyjnych
<b>JD:</b>	4
<b>ECTS:</b>	5
<b>JK:</b>	30
<b>Zaliczanie:</b>	B
<b>Ocenianie:</b>	5
<b>Semestr wprowadzenia:</b>	14L
<b>Osoba odpowiedzialna:</b>	dr inż. Jakub Janusz Koperwas
<b>Opis:</b>	Studia I stopnia - ISI (obieralny), etap B4

### Wymiary

Typ zajęć	Wymiar
W	2
P	2

### Poprzedniki

Typ poprzedzania	Nr poprzednika	Przedmiot poprzedzający	Nazwa
Z		1 BD	Bazy danych 1
Z		1 PROZ	Programowanie zdarzeniowe (I)
Z		1 TIN	Techniki internetowe
W		2 IOP	Inżynieria oprogramowania
W		2 PROI	Programowanie obiektowe (I)

### Przedmioty podobne

---

## Ostatnie realizacje

Semestr	Realizacja	Prowadzący	Instytut	Liczba miejsc
14L	A	dr inż. Jakub Janusz Koperwas	IN	24

## Przynależność do klas tematycznych i programowych

Klasa	Nazwa	Typ klasy
MINI	Metody inżynierii informatycznej (I)	Programowa
OSK	Oprogramowanie systemów komputerowych	Tematyczna
OT	Przedmioty obieralne techniczne	Programowa

## Konspekt

**Streszczenie** Celem przedmiotu jest praktyczne przygotowanie do pracy w zawodzie projektanta/programisty w zakresie aplikacji internetowych i korporacyjnych. Wykład jest poświęcony praktycznym zagadnieniom architektury, projektowania i implementacji współczesnych aplikacji internetowych oraz aplikacji klasy korporacyjnej. Na wykładzie omówione zostaną różne aspekty projektowania poszczególnych warstw aplikacji rzutujące na dobór technologii. Wyżej wymienione zagadnienia zostaną uzupełnione przeglądem najpopularniejszych i najistotniejszych na rynku technologii skupionych w, oraz wokół platformy Java (w tym Java Enterprise Edition, standardy JCP) ale także innych standardów (OMG, W3C) lub rozwiązań popularnych, ale nie będących standardami (np. rozwiązania firmy Google). Omówiony zostanie też współczesny warsztat pracy programisty. W ramach projektu studenci zaprojektują i zaimplementują aplikację z użyciem wybranego stosu technologicznego.

### Treść wykładu

- Pojęcia podstawowe(1): aplikacja internetowa, aplikacja korporacyjna, Java Enterprise Edition i jej składowe oraz inne rozwiązania. Warsztat pracy programisty:
  - zintegrowane środowiska programistyczne (Eclipse, NetBeans, itd.),
  - podstawowe informacje dotyczące narzędzi wspomagających proces wytwarzania aplikacji: automatycznego budowania (m.in. ant, maven), zarządzania kodem źródłowym SCM i DSCM (svn, mercurial, i in.), zarządzania projektem/zadaniami. Rozwiązania komercyjne a rozwiązania otwartego oprogramowania (ang. open source): rodzaje licencji i ich konsekwencje w projekcie IT. Praca programisty w aspekcie społecznościowym.
- Architektura aplikacji(1): architektura wielowarstwowa, architektura zorientowana na usługi, szyna korporacyjna, aspekty doboru architektury
- Tworzenie warstwy klienckiej aplikacji(1). Różne aspekty związane z GUI:

klient gruby i cienki, programowanie deklaratywne a imperatywne interfejsów użytkownika, różne sposoby dystrybucji aplikacji klienckiej. Podstawowe technologie aplikacji WWW: http, html, javascript, json, Ajax.

- Szkielety (ang. frameworks) (2): rola szkieletów, typowe problemy tworzenia aplikacji internetowej, Koncepcja IoC/DI, MVC, MVP, EventBus, charakterystyka wybranych szkieletów IoC/DI oraz internetowych m.in.: spring, seam, CDI, spring mvc, struts.
- Przegląd popularnych technologii warstwy klienckiej (3): JSF, GWT, Flex, Vaadin, biblioteki i szkielety javascript (backbone, jquery, node.js) i inne, charakterystyka aplikacji portalowych.
- Tworzenie warstwy logiki biznesowej (2): logika osadzona w aplikacji, logika zdalna z użyciem protokołów binarnych i tekstowych: EJB, usługi internetowe (ang. web services) oparte na SOAP (W3C), "lekkie" usługi oparte na paradygmacie REST, inne podejścia: xml-rpc, hessian.
- Tworzenie warstwy dostępu do danych (1): model relacyjny a obiektowy, technologie realizujące odwzorowanie relacyjno-obiektowe(ORM): Hibernate, JPA i in. Techniki zwiększania wydajności: cache, sharding. Podejścia typu NoSQL (Cassandra, MongoDB i in.) ,systemy repozytoryjne (JCR).
- Architektura zorientowana na usługi/zagadnienia integracji (1) (SOA), szyna korporacyjna (ESB), aranżacja usług internetowych: BPEL, wzorce EAI
- Modelowanie i zarządzanie logiką podlegającą częstym zmianom (1): procesy biznesowe i reguły biznesowe: modelowanie (notacja BPMN), uruchamianie i zarządzanie w obliczu zmian.
- Zawansowane aspekty (1): bezpieczeństwo aplikacji internetowych, zagadnienia wydajności i skalowalności, testowanie i kontrola jakości projektu, zagadnienia ciągłej integracji

## **Zakres projektu**

Celem realizowanego zadania semestralnego jest projekt i implementacja przydzielonego tematu (aplikacja internetowa, aplikacja korporacyjna, integracja) z dużym naciskiem położonym na proces wytwórczy, warsztat pracy, projekt systemu.

Projekt przeznaczony jest dla zespołów 3-4 osobowych, zawiera jednak również elementy oceniane indywidualnie.

Projekt składa się z 4 etapów:

Etap 1: Analizy postawionego problemu, dobór architektury i technologii z uzasadnieniem

Etap 2: Zapoznania z technologią (oceniane indywidualnie)

Etap 3: Stworzenia warsztatu pracy i zrębu projektu

Etap 4: Zaprojektowania i implementacji

<b>Warunki zaliczenia</b>	<p><i>Dwa sprawdziany : 2x20 punktów = 40 punktów</i></p> <p>Projekt: 60 punktów</p> <p>Etap 1: 10p</p> <p>Etap 2: 10p</p> <p>Etap 3: 10p</p> <p>Etap 4: 30p (z czego 15 punktów - realizacja funkcjonalności i jakość implementacji; 7 punktów - rozliczenie czasu pracy członków zespołu w oparciu o dostępne narzędzia kontroli projektu i logów SCM; 8 punktów - prezentacja systemu i zapewnienie jakości projektu: testy, dokumentacja)</p> <p>Końcowy etap projektu będzie rozliczany na podstawie krótkiej prezentacji połączonej z demonstracją działania systemu, a także dokumentacji projektowej.</p>
<b>Literatura</b>	<ol style="list-style-type: none"><li>1. Krzysztof Sacha (2010), Inżynieria Oprogramowania. Wydawnictwo Naukowe PWN ISBN: 978-83-01-16179-8</li><li>2. Gamma, Erich; Richard Helm, Ralph Johnson, and John Vlissides (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley. ISBN 0-201-63361-2.</li><li>3. Fowler, Martin (2002). Patterns of Enterprise Application Architecture. Addison-Wesley. ISBN 978-0-321-12742-6.</li><li>4. Freeman, Eric; Elisabeth Freeman, Kathy Sierra, and Bert Bates (2004). Head First Design Patterns. O'Reilly Media. ISBN 0-596-00712-4.</li><li>5. Christian Bauer and Gavin King (2006). Java Persistence with Hibernate, Manning Publications Co ISBN: 1-932394-88-5</li><li>6. Craig Walls (2011). Spring in Action, Third Edition, Manning Publications Co ISBN 9781935182351</li><li>7. Inne książki wydawnictwa Manning dotyczące omawianych technologii.</li><li>8. Dan Pilone, Russ Miles (2008) Head First Software Development O'Reilly Media; ISBN 978-0596527358</li><li>9. Inne książki z serii Head First</li><li>10. Gregor Hohpe and Bobby Woolf (2003) Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions</li><li>11. Addison-Wesley Professional; ISBN 978-0321200686</li><li>12. Thomas Allweyer (2010) BPMN 2.0, BoD ISBN 978-3839149850</li></ol>
<b>Streszczenie (po angielsku)</b>	<p>The aim of the course is practical preparation for the web and enterprise class application developer profession. The lecture is devoted to practical issues of architecture, design and implementation of modern web and enterprise applications. The lecture will discuss the various aspects of designing application tiers to bear on the choice of technology. The above issues will be complemented by an overview of the most popular and most relevant technologies concentrated in and around Java (including Java Enterprise Edition, JCP standards) but also to other standards (OMG, W3C), or selected popular, but non-standard technologies (e.g. Google solutions). Developer workshop tools will also be discussed. A</p>

project task is to design and implement application using the selected technological stack.

---

**ERES2** *Wydział Elektroniki i Technik Informatycznych PW*