

POLITECHNIKA KRAKOWSKA IM. TADEUSZA KOŚCIUSZKI

KARTA PRZEDMIOTU

obowiązuje studentów rozpoczynających studia w roku akademickim 2014/2015

Wydział Fizyki, Matematyki i Informatyki

Kierunek studiów: Informatyka

Profil: Ogólnoakademicki

Forma studiów: niestacjonarne

Kod kierunku: I

Stopień studiów: II

Specjalności: Teleinformatyka dla inżynierów, Informatyka stosowana dla inżynierów, Grafika komputerowa i multimedia dla inżynierów

1 INFORMACJE O PRZEDMIOCIE

| | |
|---|-------------------------------------|
| NAZWA PRZEDMIOTU | Zaawansowane techniki programowania |
| NAZWA PRZEDMIOTU W JĘZYKU ANGIELSKIM | |
| KOD PRZEDMIOTU | WFMiI I oIIN B3 14/15 |
| KATEGORIA PRZEDMIOTU | Przedmioty podstawowe |
| LICZBA PUNKTÓW ECTS | 5.00 |
| SEMESTRY | 1 |

2 RODZAJ ZAJĘĆ, LICZBA GODZIN W PLANIE STUDIÓW

| SEMESTR | WYKŁAD | ĆWICZENIA | LABORATORIUM | LABORATORIUM KOMPUTERO- WE | SEMINARIUM | PROJEKT |
|---------|--------|-----------|--------------|----------------------------------|------------|---------|
| 1 | 18 | 0 | 18 | 0 | 0 | 0 |

3 CELE PRZEDMIOTU

Cel 1 Zapoznanie słuchaczy ze stanem obecnym i przewidywanymi kierunkami rozwoju w zakresie metod tworzenia (implementacji) i utrzymania (modyfikacji, adaptacji) systemów oprogramowania w całym cyklu życia.

4 WYMAGANIA WSTĘPNE W ZAKRESIE WIEDZY, UMIEJĘTNOŚCI I INNYCH KOMPETENCJI

1 Znajomość obiektowych technik analizy, projektowania i implementacji systemów oprogramowania. Podstawowa znajomość notacji UML. Znajomość języka programowania Java (ewentualnie C#)

5 EFEKTY KSZTAŁCENIA

EK1 Wiedza Rozwój kompetencji w zakresie współczesnych platform tworzenia systemów oprogramowania (J2EE, .Net)

EK2 Wiedza Rozwój kompetencji w zakresie programowania obiektowego i aspektowego

EK3 Umiejętności Opanowanie umiejętności praktycznego stosowania wzorców projektowych

EK4 Umiejętności Opanowanie umiejętności stosowania wzorców architektury (aplikacje wielowarstwowe, MVC, SOA)

6 TREŚCI PROGRAMOWE

| LABORATORIUM | | |
|--------------|---|------------------|
| LP | TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH | LICZBA GODZIN |
| L1 | Singleton, Multiton, Opóźniona inicjalizacja : Singleton ma na celu zapewnienie istnienia co najwyżej jednej instancji klasy oraz dostarczenie globalnego punktu dostępu do niej. Multiton stanowi rozszerzenie singletona o obsługę zbioru (typowo kilku) nazwanych instancji danej klasy. Opóźniona inicjalizacja ma na celu odłożenie procesu tworzenia i inicjalizacji obiektu, do czasu kiedy następuje żądanie realizacji operacji na danej klasie. Strategia ta służy skróceniu czasu inicjalizacji aplikacji (podniesieniu efektywności). | 2 |
| L2 | Metoda wytwórcza, Fabryka abstrakcyjna : Metoda wytwórcza definiuje interfejs tworzenia obiektów, przy jednoczesnym odłożeniu do klas pochodnych decyzji dotyczącej konkretnej klasy do utworzenia. Fabryka abstrakcyjna stanowi uogólnienie metody wytwórczej o dostarczenie interfejsu do tworzenia rodzin zależnych obiektów bez specyfikowania ich konkretnych klas. | 2 |
| L3 | Budowniczy : Budowniczy oddziela konstrukcję złożonego obiektu od jego reprezentacji, tak że ten sam proces konstrukcji może tworzyć różne reprezentacje. | 2 |
| L4 | Obserwator : Obserwator definiuje zależność wiele do jednego (N-1) pomiędzy obiektami, tak że w przypadku zmiany stanu (danych) pojedynczego obiektu, wszystkie jego obiekty zależne są o tym notyfikowane automatycznie. | 2 |
| L5 | Mediator : Mediator definiuje obiekt, który enkapsuluje interakcje pomiędzy zbiorem obiektów. Strategia ta promuje luźne powiązania poprzez uniknięcie bezpośrednich odwołań pomiędzy obiektami, umożliwiając zarazem na niezależne zmiany struktury interakcji. | 2 |
| L6 | Stan, Pełnomocnik :Wzorzec pozwala na zmianę zachowania obiektu w zależności od jego wewnętrznego stanu, wykazując zachowanie jak gdyby obiekt zmienił swoją klasę. Drugi wzorzec dostarcza obiekt pośredniczący lub zaślepkę do innego obiektu w celu kontroli dostępu do obiektu zasadniczego. | 2 |

| LABORATORIUM | | |
|--------------|--|------------------|
| LP | TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH | LICZBA GODZIN |
| L7 | Strategia, Łańcuch zobowiązań : Strategia definiuje rodzinę algorytmów, opakowując każdy z nich w obiekt, czyniąc je wymiennymi. Umożliwia to na zmiany algorytmu niezależnie od używającego je klienta. Łańcuch zobowiązań stanowi rekurencyjne rozszerzenie strategii. Umożliwia uniknięcie ścisłego powiązania zgłoszeniodawcy i zgłoszeniobiorcy wywołania poprzez umożliwienie więcej niż jednemu obiektowi obsługi zgłoszenia. Powoduje to utworzenie łańcucha zgłoszeń, obsługiwanych tak długo, aż zakończy się to sukcesem (lub skończą dalsze możliwości). | 2 |
| L8 | Adapter, Fasada : Adapter zapewnia dopasowanie interfejsu klasy do innego interfejsu oczekiwanego przez klienta. Umożliwia to współpracy klas o różnorodnych interfejsach, które bez zastosowania adaptera nie mogłyby ze sobą współpracować. Fasada dostarcza zunifikowany interfejs do zbioru interfejsów w podsystemie. Zdefiniowanie interfejsu wyższego poziomu ułatwia użycie podsystemów. | 2 |
| L9 | Dekorator : Dekorator dynamicznie dołącza dodatkowe zadania do obiektu, dostarczając łatwej do zmian alternatywy dla dziedziczenia w celu rozszerzenia funkcjonalności. | 2 |

| WYKŁAD | | |
|--------|--|------------------|
| LP | TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH | LICZBA GODZIN |
| W1 | Java zalety i wady. Technologie aplety, serwety, EJB. J2EE koncepcja i charakterystyka platformy. Architektura fizyczna rozwiązań. Architektura oprogramowania. Rozproszenie danych i aplikacji. Niezawodność, skalowalność elastyczność rozwiązań. Wzorce projektowe koncepcja i cele stosowania. Wzorce konstrukcyjne strukturalne i czynnościowe. Wzorce architektury. Architektury trójwarstwowe i wielowarstwowe. Model, widok, kontroler (MVC). Architektury zorientowane na usługi. Wzorce projektowe w warstwie prezentacji (front controler, view dispatcher). Wzorce warstwy logiki aplikacyjnej (application service, session facade, bussines delegate. Warstwa integracji. Efektywność rozwiązań a stosowane praktyki kodowania. Technologia JSP. JavaBeans oraz Enterprise JB. Rola kontenera EJB, oferowane usługi. Pojęcie sesji. Rozwiązania bezstanowe oraz z kontrolą stanu. Przykłady rozwiązań. Transakcyjność w rozwiązaniach rozproszonych. Elementy integracji aplikacyjnej. | 4 |
| W2 | Architektura fizyczna rozwiązań. Architektura oprogramowania. Rozproszenie danych i aplikacji. Niezawodność, skalowalność elastyczność rozwiązań. | 4 |
| W3 | Wzorce projektowe koncepcja i cele stosowania. Wzorce konstrukcyjne strukturalne i czynnościowe. Wzorce architektury. Architektury trójwarstwowe i wielowarstwowe. Model, widok, kontroler (MVC). Architektury zorientowane na usługi. Wzorce projektowe w warstwie prezentacji (front controler, view dispatcher). Wzorce warstwy logiki aplikacyjnej (application service, session facade, bussines delegate. Warstwa integracji. | 4 |

| WYKŁAD | | |
|--------|---|------------------|
| LP | TEMATYKA ZAJĘĆ OPIS SZCZEGÓŁOWY BLOKÓW TEMATYCZNYCH | LICZBA GODZIN |
| W4 | Efektywność rozwiązań a stosowane praktyki kodowania. Technologia JSP. Java-Beans oraz Enterprise JB. Rola kontenera EJB, oferowane usługi. Pojęcie sesji. Rozwiązania bezstanowe oraz z kontrolą stanu. Przykłady rozwiązań. Transakcyjność w rozwiązaniach rozproszonych. Elementy integracji aplikacyjnej. | 3 |
| W5 | Lekkie metodyki tworzenia oprogramowania - zasady eXtreme Programming. Tworzenie oprogramowania sterowane testami. Refactoring kodu - wprowadzenie. | 3 |

7 NARZĘDZIA DYDAKTYCZNE

N1 Ćwiczenia laboratoryjne

8 OBCIĄŻENIE PRACĄ STUDENTA

| FORMA AKTYWNOŚCI | ŚREDNIA LICZBA GODZIN NA ZREALIZOWANIE AKTYWNOŚCI |
|--|---|
| Godziny kontaktowe z nauczycielem akademickim, w tym: | |
| Godziny wynikające z planu studiów | 0 |
| Konsultacje przedmiotowe | 0 |
| Egzaminy i zaliczenia w sesji | 0 |
| Godziny bez udziału nauczyciela akademickiego wynikające z nakładu pracy studenta, w tym: | |
| Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury | 90 |
| Opracowanie wyników | 0 |
| Przygotowanie raportu, projektu, prezentacji, dyskusji | 24 |
| SUMARYCZNA LICZBA GODZIN DLA PRZEDMIOTU WYNIKAJĄCA Z CAŁEGO NAKŁADU PRACY STUDENTA | 114 |
| SUMARYCZNA LICZBA PUNKTÓW ECTS DLA PRZEDMIOTU | 5.00 |

9 SPOSOBY OCENY

OCENA FORMUJĄCA

F1 Sprawozdanie z ćwiczenia laboratoryjnego

OCENA PODSUMOWUJĄCA

P1 Egzamin pisemny

P2 Egzamin ustny

KRYTERIA OCENY

| EFEKT KSZTAŁCENIA 1 | |
|---------------------|--|
| NA OCENĘ 2.0 | — |
| NA OCENĘ 3.0 | Wiedza podstawowa o zasadach projektowania i implementacji systemów komponentowych z wykorzystaniem przynajmniej jednej z platform (.Net, J2EE), znajomość podstawowych koncepcji i technologii, rozumienie zagadnień technicznych związanych ze skalowalnością, bezpieczeństwem, wysoką dostępnością. |
| NA OCENĘ 3.5 | Zdolność do samodzielnego analizowania i doboru właściwych rozwiązań technologicznych w oparciu o otrzymaną specyfikację wymagań. |
| NA OCENĘ 4.0 | Zdolność do samodzielnego wyspecyfikowania wymagań, analizy oraz doboru odpowiedniej technologii realizacji w oparciu o otrzymane założenia techniczne dotyczące wymaganych do osiągnięcia parametrów wydajnościowych, niezawodnościowych, funkcjonalnych. |
| NA OCENĘ 4.5 | Zdolność do samodzielnego wyboru właściwych rozwiązań projektowych w zakresie transakcyjności (w tym transakcji rozproszonych) oraz metod i środków komunikacji pomiędzy komponentami. |
| NA OCENĘ 5.0 | Zdolność do samodzielnego opracowania zarysu projektu systemu komponentowego z uwzględnieniem zagadnień bezpieczeństwa aplikacyjnego. |
| EFEKT KSZTAŁCENIA 2 | |
| NA OCENĘ 2.0 | — |
| NA OCENĘ 3.0 | Znajomość budowy modelu obiektowego systemu rozproszonego, z uwzględnieniem wymagań dotyczących możliwości realizacji w oparciu o podejście komponentowe, z wykorzystaniem jednej z poznanych technologii. |
| NA OCENĘ 3.5 | Znajomość podstawowych reguł budowy rozwiązań rozproszonych w warstwie aplikacji, zasad doboru proponowanej technologii realizacji systemu komponentowego, zasad doboru serwera aplikacyjnego oraz metod implementacji w przyjętym środowisku. |
| NA OCENĘ 4.0 | Zdolność do uzasadnienia przyjętych rozwiązań projektowych oraz krytycznego porównania podstawowych cech wybranej technologii realizacji z co najmniej jednym ze znanych rozwiązań alternatywnych. |
| NA OCENĘ 4.5 | Umiejętność wskazania elementów rozwiązania uzasadniających zastosowanie podejścia aspektowego oraz uzasadnienia ewentualnych korzyści wynikające z przyjęcia tej techniki implementacji. |
| NA OCENĘ 5.0 | Umiejętność wyboru właściwych rozwiązań technologicznych (.Net, J2EE) oraz stworzenia zarysu projektu systemu komponentowego z wykorzystaniem uwzględnieniem rozproszenia w warstwie danych. |
| EFEKT KSZTAŁCENIA 3 | |

| | |
|---------------------|---|
| NA OCENĘ 2.0 | — |
| NA OCENĘ 3.0 | Znajomość podstawowych wzorców projektowych, zakresu ich stosowalności, praktycznych metod implementacji. Umiejętność specyfikowania wzorców z wykorzystaniem notacji UML. |
| NA OCENĘ 3.5 | Umiejętność identyfikacji obszarów rozwiązania wskazujących na konieczność zastosowania standardowych wzorców projektowych, praktyczna umiejętność implementacji z wykorzystaniem wskazanego do zastosowania wzorca projektowego. |
| NA OCENĘ 4.0 | Praktyczna znajomość zasad projektowania aplikacji z wykorzystaniem typowych wzorców projektowych, umiejętność poprawnej implementacji przyjętych rozwiązań z wykorzystaniem jednego z języków obiektowych. |
| NA OCENĘ 4.5 | Umiejętność praktycznego stosowania zasad implementacji z wykorzystaniem techniki 'od testu do programu (test driven development) oraz zasad refactoringu tworzonych kodu. |
| NA OCENĘ 5.0 | Umiejętność samodzielnego zaprojektowania kompletnego rozwiązania wymagającego doboru i zastosowania więcej niż jednego wzorca projektowego, zdolność do uzasadnienia przyjętych rozwiązań projektowych, umiejętność dyskusji rozwiązań alternatywnych. |
| EFEKT KSZTAŁCENIA 4 | |
| NA OCENĘ 2.0 | — |
| NA OCENĘ 3.0 | Znajomość podstawowych wzorców architektury systemów rozproszonych, zakresu ich stosowalności, praktycznych metod implementacji. |
| NA OCENĘ 3.5 | Umiejętność praktycznego określenia właściwych rozwiązań w zakresie architektury aplikacji, z uwzględnieniem wymagań wynikających z transakcyjności, skalowalności, wysokiej dostępności oraz zagadnień bezpieczeństwa aplikacyjnego. |
| NA OCENĘ 4.0 | Praktyczna umiejętność dekompozycji modelu obiektowego umożliwiającej skuteczną implementację w modelu trójwarstwowym lub wielowarstwowym. Umiejętność wyodrębnienia co najmniej elementów warstwy prezentacji oraz warstwy integracji w celu precyzyjnego określenia granic warstwy logiki biznesowej. |
| NA OCENĘ 4.5 | Praktyczna umiejętność realizacji warstwy integracji w przypadku tworzenia rozwiązań z rozproszonym repozytorium danych. |
| NA OCENĘ 5.0 | Praktyczna umiejętność doboru koncepcji rozwiązania (SOA, MVC) oraz dyskusji cech proponowanego rozwiązania w stosunku do znanych rozwiązań alternatywnych. |

10 MACIERZ REALIZACJI PRZEDMIOTU

| EFEKT KSZTAŁCENIA | ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁO- WYCH EFEKTÓW ZDEFINIOWA- NYCH DLA PROGRAMU | CELE PRZEDMIOTU | TREŚCI PROGRAMOWE | NARZĘDZIA DYDAKTYCZNE | SPOSOBY OCENY |
|----------------------|---|--------------------|----------------------|--------------------------|---------------|
| EK1 | I2_W01, I2_W02, I2_W03, I2_W04, I2_W05, I2_W06, I2_W07, I2_W08, I2_U01, I2_U02, I2_U03, I2_U04, I2_U05, I2_U06, I2_U07, I2_U08, I2_U09, I2_U10, I2_U11, I2_K01, I2_K02, I2_K03, I2_K04 | Cel 1 | L1 L2 L3 L4 L5 | N1 | F1 P1 P2 |
| EK2 | I2_W01, I2_W02, I2_W03, I2_W04, I2_W05, I2_W06, I2_W07, I2_W08, I2_U01, I2_U02, I2_U03, I2_U04, I2_U05, I2_U06, I2_U07, I2_U08, I2_U09, I2_U10, I2_U11, I2_K01, I2_K02, I2_K03, I2_K04 | Cel 1 | L1 L2 L3 L4 L5 | N1 | F1 P1 P2 |

| EFEKT KSZTAŁCENIA | ODNIESIENIE DANEGO EFEKTU DO SZCZEGÓŁO- WYCH EFEKTÓW ZDEFINIOWA- NYCH DLA PROGRAMU | CELE PRZEDMIOTU | TREŚCI PROGRAMOWE | NARZĘDZIA DYDAKTYCZNE | SPOSOBY OCENY |
|----------------------|---|--------------------|----------------------|--------------------------|---------------|
| EK3 | I2_W01, I2_W02, I2_W03, I2_W04, I2_W05, I2_W06, I2_W07, I2_W08, I2_U01, I2_U02, I2_U03, I2_U04, I2_U05, I2_U06, I2_U07, I2_U08, I2_U09, I2_U10, I2_U11, I2_K01, I2_K02, I2_K03, I2_K04 | Cel 1 | | N1 | F1 |
| EK4 | I2_W01, I2_W02, I2_W03, I2_W04, I2_W05, I2_W06, I2_W07, I2_W08, I2_U01, I2_U02, I2_U03, I2_U04, I2_U05, I2_U06, I2_U07, I2_U08, I2_U09, I2_U10, I2_U11, I2_K01, I2_K02, I2_K03, I2_K04 | Cel 1 | | N1 | F1 |

11 WYKAZ LITERATURY

LITERATURA PODSTAWOWA

- [1] **GoF** — *Design Patterns Elements Of Reusable Object Oriented Software*, —, 2000, —
- [2] **GoF** — *Design Patterns with examples using Java and UML2*, —, 2000, —
- [3] **James William Cooper** — *Java. Wzorce projektowe*, —, 2001, Helion
- [4] **Allan Shalloway, James R. Trott** — *Projektowanie zorientowane obiektowo*, —, 2001, Helion

12 INFORMACJE O NAUCZYCIELACH AKADEMICKICH

OSOBA ODPOWIEDZIALNA ZA KARTĘ

dr inż. Jerzy Jaworowski (kontakt: jrj@pk.edu.pl)

OSOBY PROWADZĄCE PRZEDMIOT

1 dr inż. Jerzy Jaworowski (kontakt: jrj@pk.edu.pl)

13 ZATWIERDZENIE KARTY PRZEDMIOTU DO REALIZACJI

(miejsowość, data)

(odpowiedzialny za przedmiot)

(dziekan)

PRZYJMUJĘ DO REALIZACJI (data i podpisy osób prowadzących przedmiot)

.....