**Nazwa przedmiotu:**

Podstawy inżynierii oprogramowania

**Koordynator przedmiotu:**

dr inż. Włodzimierz Dąbrowski, wlodzimierz.dabrowski@ee.pw.edu.pl, +48222347075

**Status przedmiotu:**

Obowiązkowy

**Poziom kształcenia:**

Studia I stopnia

**Program:**

Informatyka

**Grupa przedmiotów:**

Wspólne

**Kod przedmiotu:**

**Semestr nominalny:**

2 / rok ak. 2009/2010

**Liczba punktów ECTS:**

4

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

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

**Język prowadzenia zajęć:**

polski

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

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

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

**Wymagania wstępne:**

Podstawy informatyki

**Limit liczby studentów:**

**Cel przedmiotu:**

Po zajęciach student powinien rozumieć procesy projektowania i wytwarzania oprogramowania ; umieć dokonać wyboru narzędzi wspomagających wytwarzanie oprogramowania i modelu procesu wytwarzania oprogramowania, umieć pozyskiwac i specyfkować wymagania na system informatyczny w ujęciu podstawowym, rozumieć procesy przeglądu i testowania oprogramowania i umieć zaplanować i i dokumentować podstawowe testy oprogramowania, umieć planować i zarządzać konfiguracją oraz planować w zakresie podstawowym proste przedsięwzięcie programistyczne. Student powinien równeż rozumieć problematykę zapewnienia jakości oprogramowania, doboru podstawowych metryk oprogramownia oraz posługiwania się podstawowymi wzorcami projektowymi.

**Treści kształcenia:**

Wykład
1. Modele wytwarzania oprogramowania
Wprowadzenie do inżynierii oporgramowania. Miejsce inzynierii oprogramowania w naukach informatycznych. Kryzys oprogramowania. Złożoność systemów informatycznych. Cykle zyciowe oprogramowania. Dobór modelu wytwarzania do rodzaju projektu.
2. Faza strategiczna w produkcji oprogramowania
Identyfikacja celu i zakresu projektu informatycznego. Studium osiągalności. Role w projekcie informatycznym. Analiza wariantów strategicznych. Dokumentowanie projektu i podstawowe dokumenty projektowe. Kryteria sukcesu fazy strategicznej.
3. Pozyskiwanie i dokumentowanie wymagań
Czym są a czym nie są wymagania na system informatyczny. Podział i klasyfikacja wymagań. Metody pozyskiwania wymagań. Wywiady z użytkownikiem. Karty CRC. Metoda przypadków użycia. Pojęcie aktora i użytkownika systemu. Strukturalny opis przypadku użycia. Dokumentowanie wymagań. Cechy dobrych i złych dokumentów wymagań.
4. Języki modelowania systemów informatycznych
Podejście strukturalne. Podejście zorientowane obiektowo. Pojęcie klas i obiektów. Proste asocjacje. Elementy notacji UML. Modelowanie struktury i aspektów behawioralnych systemów. Podstawy diagramu klas.
5. Modelowanie dynamiki i architektury systemów
Kominikaty. Wymina komuniaktów. Proste diagramy sekwencji i diagramy stanu. Obrazowanie komponentów i części składowych systemu. Diagramy wdrożeniowe.
6. Narzędzia inżynierii oprogramowania
Wsparcie planowania i śledzenia przedsięwzięica informatycznego. Diagramy sieciowe i diagramy Gantta. Narzędzia CASE. Przykłady narzędzi CASE i RAD. Modelowanie wizualne. Środowiska IDE.
7. Wersjonowanie i konfiguracja oprogramowania
Pojęcie wersjoowania i konfiguracji. Pozycje konfiguracji. Produkty bazowe i wydania oprogramowania. Zarządzanie konfiguracją. Metody wersjonowania kodu. Narzędzia wspierające zarządzanie konfiguracją.
8. Testowanie i weryfikacja oprogramowania
Rodzaje testów. Przygotowanie i dokumnetowanie testów. meotdy weryfikacji oprogramowania. Przeglądy koleżeńskie, inspekcje i audyty. Elementy zarządzania jakością oprogramowania. Dobre i złe praktyki programistyczne.
9. Metryki oprogramowania
Pomiary w inżynierii oprogramowania. Metryki i rodzaje metryk. Metoda COCOMO, FPA i inne.
10. Projektowanie i programowanie interfejsu użytkownika
Rodzaje interakcji człowiek komputer. Typy interfejsów. Komponnety interfejsu użytkownika. Cechy dobrych i złych interfejsów uzytkownika. Wzorce w projektowaniu interfejsów.
11. Architektura systemów informatycznych
Rózne architektury systemów informatycznych. Cechy architektury. Kryteria doboru architektury systemu. Narzędzia dokumentowania architektury. inżynieria odwrotna.
12. Wzorce projektowe
Pojęcie wzorca projektowego. Potrzeba stosowania i studiowania wzorców projektowych. przykłady prostych wzorców projekowych: MVC, adapter, fasada, fabryka abstrakcyjna.
13. Organizacja przedsięwzięcia programistycznego
Zespół projektowy. Metodyczne podejście do prowadzenia projektów. Plan prjektu. Narzędzia planowania i śledzenia projektu. Praca zespołowa i narzędzia wspomagające pracę zespołową. Kontrola w projekcie.
14. Rozwój inżynierii oprogramowania
Przgląd pozostałych obszarów inżynierii oprogramowania. Problemy i perspektywy inżynierii oprogramowania. Obszary bdawcze inżynierii oprogramowania.
Laboratorium
1. Narzędzia wspomagające wytwarzanie oprogramowania
2. Dokumentowanie wymagań
3. Projektowanie w narzędziach CASE
4. Planowani testów i ich dokumentowanie
5. Metryki oprogramowania

**Metody oceny:**

brak

**Egzamin:**

**Literatura:**

1. W. Dąbrowski, A.Stasiak, M.Wolski, Modelowanie systemów w języku UML 2.1, PWN 2007

**Witryna www przedmiotu:**

**Uwagi:**

## Efekty przedmiotowe