



KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Inżynieria oprogramowania [S1IZarz1E>IO]

Przedmiot

Kierunek studiów

Inżynieria zarządzania/Engineering Management

Rok/Semestr

2/4

Studia w zakresie (specjalność)

–

Profil studiów

ogólnoakademicki

Poziom studiów

pierwszego stopnia

Język oferowanego przedmiotu

angielski

Forma studiów

stacjonarne

Wymagalność

obligatoryjny

Liczba godzin

Wykład

15

Laboratorium

0

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

15

Liczba punktów ECTS

2,00

Koordynatorzy

dr hab. Grzegorz Pawłowski

grzegorz.pawlowski@put.poznan.pl

Wykładowcy

Wymagania wstępne

Podstawowy kurs z projektowania informatycznych systemów zarządzania. Sprawne wykorzystywanie narzędzi wspomagania projektowania (Visio) oraz umiejętności z zakresu projektowania baz danych. Rozumienie potrzeby umiejętności projektowania i zarządzania wdrożeniami systemów informatycznych zarządzania.

Cel przedmiotu

Zapoznanie studentów z metodami i CASE-mi inżynierii oprogramowania stosowanymi w projektowaniu informatycznych systemów zarządzania.

Przedmiotowe efekty uczenia się

Wiedza:

Student wyjaśnia i porównuje różne metody i narzędzia do zbierania, przetwarzania, selekcji i dystrybucji informacji w kontekście inżynierii oprogramowania [P6S_WG_08]

Student opisuje cykl życia produktów oprogramowania, w tym jego etapy takie jak tworzenie, implementacja, eksploatacja i modyfikacja [P6S_WG_15]

Student identyfikuje i klasyfikuje podstawowe metody, techniki, narzędzia i materiały stosowane w

inżynierii oprogramowania, takie jak narzędzia do zarządzania wymaganiami, prototypowania i szacowania kosztów [P6S_WG_16]

Student wymienia podstawowe zasady bezpieczeństwa i higieny pracy oraz opisuje ich zastosowanie w środowisku pracy inżyniera oprogramowania [P6S_WG_18]

Umiejętności:

Student projektuje i realizuje eksperymenty oraz symulacje komputerowe, analizuje i interpretuje wyniki w kontekście rozwoju i testowania oprogramowania [P6S_UW_09]

Student stosuje metody analityczne, symulacyjne oraz eksperymentalne do formułowania i rozwiązywania problemów w zakresie inżynierii oprogramowania, w tym w procesie inżynierii wymagań [P6S_UW_10]

Student przejmuje odpowiedzialność za indywidualne i grupowe projekty informatyczne, przestrzegając zasad pracy zespołowej i metodologii zarządzania projektami, np. modelu P-CMM [P6S_UO_01]

Kompetencje społeczne:

Student analizuje i ocenia zależności przyczynowo- skutkowe w procesie tworzenia oprogramowania, podejmując decyzje dotyczące priorytetyzacji zadań i zarządzania zasobami [P6S_KK_02]

Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Ocena formująca z wykładów: zadania problemowe do wykonania na wykładzie, ocena podsumowująca na podstawie egzaminu. Próg zaliczeniowy - 50%.

Projekt: ocena formująca na podstawie bieżącej pracy dotyczącej konstrukcji projektu modelu logicznego aplikacji, ocena podsumowująca na podstawie gotowej dokumentacji modelu logicznego aplikacji. Do oceny uwzględnia się poprawność i kompletność osiągniętych rezultatów.

Treści programowe

Program obejmuje następujące zagadnienia: tworzenie, implementacja, eksploatacja i modyfikacja systemu informatycznego, integracja systemów informatycznych, wymagania funkcjonalne i niefunkcjonalne, proces inżynierii wymagań, zarządzanie wymaganiami, narzędzia inżynierii oprogramowania, prototypowanie oprogramowania z wykorzystaniem języka UML, wdrażanie systemów IT, kastomizacja oprogramowania.

Tematyka zajęć

Wykład:

1. Analiza systemu informatycznego - etapy projektowania, budowy i eksploatacji Systemu Informatycznego, narzędzia CASE wysokiego i niskiego poziomu, wymagania stawiane oprogramowaniu, zarządzanie wymaganiami, schemat FURPS, Model kaskadowy
2. Modelowanie iteracyjne - Model ewolucyjny, Model przyrostowy, Model spiralny, Metodyka RUP: fazy i cykle budowy oprogramowania, Metodyka zwinna (Agile): ekstremalne, Scrum, Kanban
3. Budowa oprogramowania - podejście strukturalne, obiektowe i generyczne
- 4-7. Kurs języka UML - historia modelowania obiektowego, charakterystyka języka UML, modelowanie systemów biznesowych i informatycznych, Diagramy dynamiki i struktury modelu, Diagram przypadków użycia (4), Diagram aktywności (5), Diagram sekwencji (6), Diagram klas i obiektów (7)
8. Wdrażanie systemów IT – prototypowanie, wdrażanie oprogramowania, szkolenia, dobór personelu
9. Kastomizacja oprogramowania - zarządzanie oprogramowaniem, modyfikacje i aktualizacje, serwis

Projekt:

Wstępne modelowanie wymagań wybranego oprogramowania z wykorzystaniem schematu burzy mózgów (np. schemat FURPS) oraz wykonanie zestawu diagramów UML obejmujący: diagram przypadków użycia, diagram aktywności, diagram sekwencji, diagram klas.

Metody dydaktyczne

Wykład - wykład informacyjny, konwersatoryjny, case study

Laboratoria - metoda laboratoryjna, metoda projektu, burza mózgów, metoda demonstracji

Literatura

Podstawowa:

Borucki A. (2012). E-Biznes. Wydawnictwo Politechniki Poznańskiej. Poznań.

Kolbusz E., Olejniczak W., Szyjewski Z. (2005). Inżynieria systemów informatycznych w e-gospodarce. PWE. Warszawa.

Sommerville I. (2003). Inżynieria oprogramowania. WNT. Warszawa.

Jaskiewicz A. (1997). Inżynieria oprogramowania. Helion. Gliwice.

Uzupełniająca:

Szpringer W. (2012). Innowacyjne modele e-biznesu. Difin. Warszawa.

Flasiński M.(2008). Zarządzanie projektami informatycznymi.PWN

Bilans nakładu pracy przeciętnego studenta

	Godzin	ECTS
Łączny nakład pracy	50	2,00
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	30	1,00
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwium/egzaminu, wykonanie projektu)	20	1,00