



UNIwersytet
IM. ADAMA MICKIEWICZA
W POZNANIU

Praktyczne umiejętności projektowe

Sylabus zajęć

Informacje podstawowe

Kierunek studiów Informatyka Specjalność - Jednostka organizacyjna Wydział Matematyki i Informatyki Poziom studiów studia inżynierskie pierwszego stopnia Forma studiów studia stacjonarne Profil studiów profil ogólnoakademicki		Cykl dydaktyczny 2023/24 Kod zajęć 06INFS.320S.02446.23 Języki wykładowe polski Obligatoryjność Obowiązkowy Blok zajęciowy Przedmioty specjalnościowe
Koordynator zajęć	Marcin Witkowski, Patryk Żywica	
Prowadzący zajęcia	Marcin Witkowski	
Okres Semestr 6	Forma zajęć / liczba godzin / forma zaliczenia • Warsztaty: 30, Zaliczenie z oceną	Liczba punktów ECTS 2

Cele kształcenia dla zajęć

Kod	Cel
C1	Zapoznanie studentów z technologiami oraz problemami tworzenia oprogramowania na potrzeby klientów zewnętrznych w oparciu o case studies wybranych zagadnień i scenariuszy.

Wymagania wstępne

Brak.

Efekty uczenia się dla zajęć

Kod	Efekty uczenia się dla zajęć w zakresie	Efekty uczenia się dla kierunku	Metody weryfikacji osiągnięcia efektów uczenia się dla zajęć
Wiedzy - Student/ka:			
W1	Zna i rozumie aspekty bezpieczeństwa danych, zabezpieczenia aplikacji, wykrywania błędów w aplikacji.	INF_K3_W07	Prezentacja multimedialna
W2	Zna i rozumie modele tworzenia aplikacji, strategie budowania aplikacji dla różnych zastosowań praktycznych.	INF_K3_W03	Prezentacja multimedialna
W3	Zna i rozumie proces zbierania wymagań i zarządzania projektem informatycznym	INF_K3_W04_inz	Prezentacja multimedialna
Umiejętności - Student/ka:			
U1	Potrafi wskazać swoją rolę w projekcie i określić związane z nią zadania i odpowiedzialności	INF_K3_U03	Prezentacja multimedialna
U2	Potrafi powiązać przedstawione studium przypadku i określić relacje w stosunku do prowadzonego zespołowego projektu inżynierskiego	INF_K3_U10	Prezentacja multimedialna
Kompetencji społecznych - Student/ka:			
K1	Jest gotów/gotowa zadawać pytania i konsultować się z praktykami na temat własnego zadania projektowego.	INF_K3_K02	Prezentacja multimedialna

Treści programowe dla zajęć

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
1.	Zbieranie wymagań; Priorytetyzacja zakresu; Grooming backlogu i zarządzanie projektem, w tym narzędzia zarządzania projektem	W3	Warsztaty
2.	Testy z użytkownikiem; Testy użyteczności; Scenariusze testowe, wywiad, określenie potrzeb użytkowników	W3	Warsztaty
3.	Testy automatyczne; Testy automatyczne - narzędzia; Testy jednostkowe, pokrycie testami, narzędzia kontroli jakości kodu; Testy integracyjne: narzędzia, projektowanie i implementacja	W1	Warsztaty
4.	Zarządzanie procesem wytwórczym, SCRUM, Kanban, Waterfall, rola Scrum Mastera	W3	Warsztaty
5.	Repozytorium, kodu, przechowywanie dużych plików; aspekty prawne zdobywania i przechowywania danych. Przechowywanie haseł i ustawień. Code review - proces i zasady, narzędzia	W1, W2	Warsztaty
6.	Proces ciągłej integracji, narzędzia ciągłej integracji. Wersjonowanie projektu, tagowanie, wydawanie wersji	W2	Warsztaty
7.	Deploy projektu, serwery aplikacji, ustawienia domeny, https, load balancery, proxy	W2	Warsztaty

Lp.	Treści programowe dla zajęć	Efekty uczenia się dla zajęć	Formy zajęć
8.	Rozwiązania chmurowe, koszty utrzymania aplikacji	W2	Warsztaty
9.	Projektowanie bazy danych, nierelacyjne, relacyjne, czasowe bazy danych. Wydajność bazy danych, cacheowanie danych	W2	Warsztaty
10.	Architektura monolitu; Architektura mikroserwisowa, routing w architekturze mikroserwisowej; Popularne stacki technologiczne i połączenia technologii	W2	Warsztaty
11.	Skalowalność aplikacji (horyzontalne i wertykalne), wydajność, testowanie wydajności, optymalizacja kodu	W2	Warsztaty
12.	Bezpieczeństwo, zabezpieczenie aplikacji: Frontend i Backend, podstawowe podatności i niebezpieczeństwa	W1, W2	Warsztaty
13.	Prezentacje zastosowań w projektach inżynierskich	U1, U2, K1	Warsztaty

Informacje dodatkowe

Forma zajęć	Metody i formy prowadzenia zajęć
Warsztaty	Metoda analizy przypadków, Metoda laboratoryjna

Forma zajęć	Warunki zaliczenia zajęć
Warsztaty	Na końcową ocenę składa się wynik uzyskany z prezentacji. Skala ocen: * bardzo dobry (bdb; 5,0) – od 90% punktów, * dobry plus (db plus; 4,5) – od 80% punktów, * dobry (db; 4,0) – od 70% punktów, * dostateczny plus (dst plus; 3,5) – od 60% punktów, * dostateczny (dst; 3,0) – od 50% punktów, * niedostateczny (ndst; 2,0) – poniżej 50% punktów.

Literatura

Obowiązkowa

1. Vogel, O., Arnold, I., Kehrner, T., Chughtai, A. (2011). Software Architecture: A Comprehensive Framework and Guide for Practitioners. Niemcy: Springer Berlin Heidelberg.

Dodatkowa

1. Hitesh, M. A. K. (2020). Fundamentals of Software Engineering. Indie: BPB Publications.

Nakład pracy studenta i punkty ECTS

Rodzaje zajęć studenta	Średnia liczba godzin* przeznaczonych na zrealizowane rodzaje zajęć
Przygotowanie prezentacji multimedialnej	10
Czytanie wskazanej literatury	10

Warsztaty	30
Łączny nakład pracy studenta	Liczba godzin 50
Liczba punktów ECTS	ECTS 2

* godzina (lekcyjna) oznacza 45 minut

Efekty uczenia się dla kierunku

Kod	Treść
INF_K3_K02	Absolwent/ka jest gotów/gotowa do zasięgania opinii ekspertów przy rozwiązywaniu problemów teoretycznych i praktycznych
INF_K3_U03	Absolwent/ka potrafi pracować indywidualnie i w zespole, zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów
INF_K3_U10	Absolwent/ka potrafi przygotować wystąpienia ustne, w języku polskim i języku angielskim, dotyczące zagadnień teoretycznych i praktycznych informatyki
INF_K3_W03	Absolwent/ka zna i rozumie narzędzia, technologie i urządzenia informatyczne właściwe dla wybranych obszarów zastosowań oraz podstawy ich działania
INF_K3_W04_inz	Absolwent/ka zna i rozumie zaawansowane pojęcia, konstrukcje i procesy związane z językami programowania i inżynierią programowania
INF_K3_W07	Absolwent/ka zna i rozumie zaawansowane pojęcia związane z technologiami sieciowymi, bezpieczeństwem i protokołami kryptograficznymi