

KARTA MODUŁU

I. OGÓLNE INFORMACJE O MODULE								
COLLEGIUM WITELONA UCZELNIA PAŃSTWOWA WYDZIAŁ NAUK TECHNICZNYCH I EKONOMICZNYCH								
Kierunek studiów:		INFORMATYKA						
Poziom studiów:		studia pierwszego stopnia						
Profil studiów:		praktyczny						
Forma studiów:		stacjonarne/niestacjonarne						
Nazwa modułu:		Zaawansowane metody programowania						
Rodzaj modułu:		Moduł do wyboru – specjalność – Programowanie aplikacji mobilnych i internetowych						
Język wykładowy:		Język polski*						
Rok studiów:	3	Formy prowadzenia zajęć wraz z liczbą godzin dydaktycznych:						
Semestr:	6	Wykład**	Laboratorium**					
Liczba punktów ECTS ogółem:	3	15/12	30/12					
Forma zaliczenia:		Egzamin						
Wymagania wstępne:		Znajomość zagadnień z kursów Projektowanie i programowanie systemów internetowych I oraz Projektowanie i programowanie systemów internetowych II.						
II. CELE KSZTAŁCENIA								
Cele kształcenia:								
Cel 1: Przedstawić teorię oraz przykłady wykorzystania dobrych praktyk programistycznych.								
Cel 2: Przedstawić teorię oraz przykłady wykorzystania wzorców projektowych.								
Cel 3: Przedstawić teorię oraz przykłady wykorzystania paradygmatów programowania.								
Cel 4: Przedstawić praktyczne podejście do zespołowej pracy projektowej.								
III. EFEKTY UCZENIA SIĘ WRAZ Z ODNIESIENIEM DO EFEKTÓW KIERUNKOWYCH								
Efekt uczenia się	Student, który zaliczył moduł w zakresie:						Odniesienie do efektów kierunkowych	
wiedzy:								
W01	Rozumie potrzebę wytwarzania oprogramowania zgodnie z zasadami czystego kodu.						K1I_W04	
W02	Opisuje i wskazuje zastosowanie wzorców projektowych.						K1I_W10	
umiejętności:								
U01	Potrafi wykorzystać programowanie ekstremalne.						K1I_U08	
U02	Potrafi wskazać odpowiedzialności wskazanych klas i metod oraz stwierdzić czy łamana jest zasada pojedynczej odpowiedzialności.						K1I_U10	
U03	Potrafi zaproponować elastyczną architekturę kodu opartą o polimorfizm i abstrakcję.						K1I_U12	
U04	Potrafi ocenić kod pod względem wykorzystania dobrych praktyk programistycznych.						K1I_U16	
kompetencji społecznych:								
-	-						-	
IV. TREŚCI PROGRAMOWE								
Treści programowe (tematyka zajęć, zaprezentowana z podziałem na poszczególne formy zajęć z określeniem liczby godzin potrzebnych na ich realizację)								
Wykład:								

Kod	Tematyka zajęć	Liczba godzin S/N
w1	Zapoznanie z treścią karty modułu. Dobór technologii projektowej	2/1
w2	Dobre praktyki programistyczne: zasady SOLID	2/1
w3	Dobre praktyki programistyczne: KISS, DRY, YAGNI i inne	2/1
w4	Czysty kod	2/2
w5	Wzorce projektowe: kreacyjne	2/2
w6	Wzorce projektowe: strukturalne i operacyjne	2/2
w7	12 factor i programowanie ekstremalne	2/2
w8	Praktyczne code review	1/1

Laboratorium**:

Kod	Tematyka zajęć	Liczba godzin S/N
l1	Zapoznanie z treścią karty modułu. Zapoznanie ze środowiskiem pracy i przedstawienie warunków zaliczenia	2/1
l2	Projektowanie rozproszonego systemu informatycznego; dobór technologii projektowej; wytworzenie specyfikacji	6/3
l3	Implementacja wybranych wzorców projektowych	8/4
l4	Refaktoryzacja pod względem dobrych praktyk programistycznych	4/2
l5	Refaktoryzacja pod względem czystego kodu	4/2
l6	Praktyczne wykorzystanie programowania ekstremalnego	2/1
l7	Testowanie integracyjne rozproszonego systemu	2/1
l8	Prezentacja pracy projektowej	2/2

V. METODY KSZTAŁCENIA, NARZĘDZIA DYDAKTYCZNE

1. Metody kształcenia:

Wykład: wykład informacyjny i problemowy

Laboratorium: metoda problemowa, warsztatu i ćwiczeniowa oparta na wykorzystaniu różnych źródeł wiedzy, studium przypadku

2. Narzędzia (środki) dydaktyczne:

Prezentacja multimedialna, teksty źródłowe, tablica i materiały udostępniane przez prowadzącego.

VI. FORMA I KRYTERIA ZALICZENIA MODUŁU

1. Formy zaliczenia:

Wykład: egzamin

Laboratorium: zaliczenie z oceną

2. Sposób weryfikacji i oceniania efektów uczenia się:

Wykład:

- pisemny lub ustny test wiedzy – kryteria oceny: • 51% - 60% - ocena dostateczna, • 61% - 70% - ocena dostateczna plus, • 71% - 80% - ocena dobra, • 81% - 90% - ocena dobra plus, • 91% - 100% - ocena bardzo dobra,
- obserwacja i ocena postaw studenta.

Laboratorium:

- przygotowanie pracy projektowej i sprawozdania,
- obserwacja i ocena postaw studenta.

3. Podstawowe kryteria oceny lub wymagania egzaminacyjne określone są indywidualnie, jednak powinny zachować adekwatność wobec zaplanowanych efektów uczenia się

VII. BILANS PUNKTÓW ECTS - NAKŁAD PRACY STUDENTA

Kategoria	Obciążenie studenta
<i>Liczba godzin realizowanych przy bezpośrednim udziale nauczyciela (godziny kontaktowe)</i>	45/24

Udział w wykładach	15/12
Udział w laboratorium	30/12
Samodzielna praca studenta (godziny niekontaktowe)	30/51
Przygotowanie do wykładu	8/17
Przygotowanie do laboratorium	16/28
Przygotowanie do egzaminu	4/4
Przygotowanie do zaliczenia laboratorium	2/2
Łączna liczba godzin	75
Punkty ECTS za moduł	3

VIII. ZALECANA LITERATURA

Literatura podstawowa:

1. R. C. Martin (2015). Czysty kod. Podręcznik dobrego programisty. Helion
2. E. Gamma, R. Helm, R. Johnson, J. Vlissides (2010). Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku. Helion
3. M. Fowler (2019). Refaktoryzacja. Ulepszanie struktury istniejącego kodu. Wydanie II. Helion

Literatura uzupełniająca:

1. R. C. Martin (2014). Zwinne wytwarzanie oprogramowania. Najlepsze zasady, wzorce i praktyki. Helion
2. D. Thomas, A. Hunt (2021). Pragmatyczny programista. Od czeladnika do mistrza. Wydanie II. Helion

Na kierunkach studiów, na których obowiązują standardy kształcenia oraz odrębne przepisy określone przez właściwego ministra, karty modułów powinny także uwzględniać powyższe uregulowania

*należy odpowiednio wypełnić

** należy wpisać formę/formy przypisane do modułu określone w programie studiów (ćwiczenia, seminarium, konwersatorium, lektorat, laboratorium, warsztat, projekt, zajęcia praktyczne, zajęcia terenowe, zajęcia wychowania fizycznego, praktyka zawodowa, inne)