

SYLABUS

DOTYCZY CYKLU KSZTAŁCENIA 2024-2028

Rok akademicki 2024/2025

1. PODSTAWOWE INFORMACJE O PRZEDMIOCIE

Nazwa przedmiotu	<i>narzędzia pracy zespołowej</i>
Kod przedmiotu*	
nazwa jednostki prowadzącej kierunek	<i>Kolegium Nauk Przyrodniczych</i>
Nazwa jednostki realizującej przedmiot	<i>Kolegium Nauk Przyrodniczych, Instytut Informatyki</i>
Kierunek studiów	<i>Informatyka i ekonometria</i>
Poziom studiów	<i>studia I stopnia</i>
Profil	<i>praktyczny</i>
Forma studiów	<i>stacjonarne</i>
Rok i semestr/y studiów	<i>rok I, semestr 2</i>
Rodzaj przedmiotu	<i>przedmiot kierunkowy</i>
Język wykładowy	<i>język polski</i>
Koordinator	<i>mgr inż. Jaromir Sarzyński</i>
Imię i nazwisko osoby prowadzącej / osób prowadzących	<i>mgr inż. Jaromir Sarzyński</i>

* -opcjonalnie, zgodnie z ustaleniami w Jednostce

1.1. Formy zajęć dydaktycznych, wymiar godzin i punktów ECTS

Semestr (nr)	Wykł.	Ćw.	Konw.	Lab.	Sem.	ZP	Prakt.	Inne (jakie?)	Liczba pkt. ECTS
2				15					1

1.2. Sposób realizacji zajęć

zajęcia w formie tradycyjnej

1.3 Forma zaliczenia przedmiotu (z toku)

zaliczenie z oceną

2. WYMAGANIA WSTĘPNE

znajomość podstawowych poleceń powłoki systemu Linux, podstawy programowania

3. CELE, EFEKTY UCZENIA SIĘ, TREŚCI PROGRAMOWE I STOSOWANE METODY DYDAKTYCZNE

3.1 Cele przedmiotu

C ₁	Zapoznanie się z systemem kontroli wersji <i>Git</i> oraz narzędziami do jego obsługi.
C ₂	Zapoznanie się z wybranym serwisem hostingowym dla projektów programistycznych.
C ₃	Nabycie podstawowej umiejętności zespołowej pracy nad projektem programistycznym.

3.2 Efekty uczenia się dla przedmiotu

EK (efekt uczenia się)	Treść efektu uczenia się zdefiniowanego dla przedmiotu	Odniesienie do efektów kierunkowych ¹
EK_01	Zna zagadnienia dotyczące procesu kontroli wersji oraz systemów kontroli wersji kodu źródłowego. Zna zasady działania systemu kontroli wersji <i>Git</i> . Zna zagadnienia dotyczące hostingu projektów wykorzystujących system kontroli wersji <i>Git</i> . Zna zestaw komend systemu <i>Git</i> potrzebny do pracy z tym systemem oraz jego współdziałania z wybranym serwisem hostingowym np. <i>GitHub</i> .	K_W03
EK_02	Potrafi wykorzystywać w praktyce system kontroli wersji <i>Git</i> oraz serwis <i>GitHub</i> jako narzędzia wspomagające proces zespołowego wytwarzania oprogramowania.	K_U10

3.3 Treści programowe

Problematyka ćwiczeń laboratoryjnych

Wprowadzenie do systemów kontroli wersji. System <i>Git</i> .
Praca z systemem <i>Git</i> z poziomu linii poleceń – ćwiczenia.
Praca z systemem <i>Git</i> z poziomu GUI – ćwiczenia.
Konwencje (reguły) nazewnictwa.
Zdalne repozytoria na <i>GitHub</i> – ćwiczenia.
Organizacja pracy w zespole z wykorzystaniem <i>Git'a</i> i <i>GitHub'a</i> .
Praca w zespole nad projektem programistycznym – ćwiczenia.
Język <i>Markdown</i> – tworzenie podstawowej dokumentacji.
Kolokwia zaliczeniowe.

3.4 Metody dydaktyczne

Laboratorium: praca indywidualna, praca zespołowa, wykonywanie zadań praktycznych przy komputerze.

¹ W przypadku ścieżki kształcenia prowadzącej do uzyskania kwalifikacji nauczycielskich uwzględnić również efekty uczenia się ze standardów kształcenia przygotowującego do wykonywania zawodu nauczyciela.

4. METODY I KRYTERIA OCENY

4.1 Sposoby weryfikacji efektów uczenia się

Symbol efektu	Metody oceny efektów uczenia się (np.: kolokwium, egzamin ustny, egzamin pisemny, projekt, sprawozdanie, obserwacja w trakcie zajęć)	Forma zajęć dydaktycznych (w, ćw, ...)
EK_01	kolokwium	laboratorium
EK_02	kolokwium	laboratorium

4.2 Warunki zaliczenia przedmiotu (kryteria oceniania)

Zaliczenie laboratorium:

Następuje na podstawie dwóch kolokwίων. Każde z nich jest ocenianie w skali 2.0 – 5.0. Ocena uzyskiwana jest proporcjonalnie do liczby uzyskanych punktów za kolokwium. Na ocenę 3.0 „dostateczny” próg punktów do uzyskania wynosi 50%.

Laboratorium jest zaliczone, gdy student z obu kolokwίων otrzyma ocenę przynajmniej „dostateczny”.

Efekt EK_01 jest zaliczany w ramach pierwszego kolokwium w formie testu, w którym student udziela odpowiedzi na pytania m.in. z zakresu teorii odnośnie systemów kontroli wersji, znajomości funkcjonowania systemu *Git*, znajomości komend systemu *Git*, znajomości stosowania systemu *Git* jako narzędzia pracy w zespole (w tym jego użycia powiązanego z serwisem *GitHub*).

Efekt EK_02 jest zaliczany w ramach drugiego kolokwium w formie praktycznej, w której studenci wykorzystują system *Git* (umiejętność używania komend) oraz serwis *GitHub* do pracy nad projektem programistycznym w wybranym języku programowania. W ramach wykonywanych zadań do uzyskania są punkty m.in. za umiejętność obsługi serwisu *GitHub*, stosowanie ustalonych zasad pracy w zespole, poprawne wykonanie powierzonych zadań (w tym programistycznych), przestrzeganie konwencji nazewnictwa wykorzystywanego języka.

5. CAŁKOWITY NAKŁAD PRACY STUDENTA POTRZEBNY DO OSIĄGNIĘCIA ZAŁOŻONYCH EFEKTÓW W GODZINACH ORAZ PUNKTACH ECTS

Forma aktywności	Średnia liczba godzin na zrealizowanie aktywności
Godziny z harmonogramu studiów	15
Inne z udziałem nauczyciela akademickiego (udział w konsultacjach, egzaminie)	2
Godziny niekontaktowe – praca własna studenta (przygotowanie do zajęć, egzaminu, napisanie referatu itp.)	10
SUMA GODZIN	27
SUMARYCZNA LICZBA PUNKTÓW ECTS	1

* Należy uwzględnić, że 1 pkt ECTS odpowiada 25-30 godzin całkowitego nakładu pracy studenta.

6. PRAKTYKI ZAWODOWE W RAMACH PRZEDMIOTU

wymiar godzinowy	–
zasady i formy odbywania praktyk	–

7. LITERATURA

Literatura podstawowa:

- Chacon S., Straub B.: „Pro Git”, Apress, 2014
<https://git-scm.com/book/pl/v2>
- Tsitoara M.: *Git i GitHub. Kontrola wersji, zarządzanie projektami i zasady pracy zespołowej.*, Gliwice, Helion, 2022
- Jacek Drąg.: *Git. Od koncepcji do praktyki.*, Gliwice, Helion, 2024

Literatura uzupełniająca:

- dokumentacja systemu Git:
<https://git-scm.com/docs>
- poradnik od Atlassian:
<https://www.atlassian.com/git/tutorials>
- poradniki od GitHub'a:
<https://docs.github.com/en/get-started/getting-started-with-git>
<https://docs.github.com/en/get-started/using-git>