

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

KARTA OPISU PRZEDMIOTU

Wydział		Informatyki	
Kierunek		Informatyka	
Specjalność		Programowanie urządzeń mobilnych	
Semestr	V	Program studiów, dla którego obowiązuje sylabus	2025/2026
Stopień studiów	I		

Nazwa przedmiotu	Programowane iOS			
Kod przedmiotu	PIOS			
Łączna liczba godzin	30	Tryb	stacjonarny	niestacjonarny
Profil kształcenia	Ogólnoakademicki (A)		Praktyczny (P)	
Forma zajęć	laboratorium			
Język przedmiotu	polski			
Liczba punktów ECTS	2			

Prowadzący zajęcia	
Forma prowadzonych zajęć	Laboratorium
Wymiar zajęć	30 h
Stopień (tytuł) naukowy	
Imię	
Nazwisko	

Wymagania wstępne	Podstawowa znajomość języków programowania, takich jak Java lub C#, znajomość podstawowych pojęć z zakresu inżynierii oprogramowania, znajomość podstawowych algorytmów i struktur danych.
Założenia i cele przedmiotu	Przedmiot ma na celu wprowadzenie do tworzenia aplikacji na platformę iOS. Studenci poznają architekturę systemu iOS, środowisko Xcode oraz język Swift. Nauczą się tworzyć interfejs użytkownika z wykorzystaniem Storyboards, UIKit i Auto Layout, zarządzać widokami i nawigacją między nimi oraz pozyskają podstawową wiedzę o przechowywaniu danych na urządzeniach z iOS.
Metody dydaktyczne	<ol style="list-style-type: none"> 1. Prezentacje multimedialne. 2. Pokazy przykładowych rozwiązań problemów. 3. Rozwiązywanie zadań praktycznych.

Efekty uczenia się (odniesienie do charakterystyk poziomów Polskiej Ramy Kwalifikacji)		Odniesienie do efektów dla kierunku	Odniesienie do efektów uczenia się wg Polskiej Ramy Kwalifikacji
WIEDZA – absolwent zna i rozumie:	W01. Architekturę systemu iOS oraz podstawy działania aplikacji w środowisku Apple. W02. Podstawy języka Swift, jego składnię i semantykę oraz różnice względem Javy czy	K_W02 K_W06 K_W10 K_W13 K_W14	P6S_WG P6S_WG_INŻ

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

	<p>Kotlina.</p> <p>W03. Zasady projektowania interfejsów użytkownika w iOS (Storyboards, UIKit, Auto Layout) z uwzględnieniem estetyki i użyteczności.</p> <p>W04. Rolę View Controllers oraz mechanizmy nawigacji i zarządzania przepływem między ekranami aplikacji iOS.</p> <p>W05. Podstawowe metody przechowywania danych w iOS (UserDefaults, pliki, podstawy Core Data).</p>		
UMIEJĘTNOŚCI – absolwent potrafi:	<p>U01. Wykorzystać środowisko Xcode do stworzenia prostej aplikacji iOS w języku Swift.</p> <p>U02. Zaprojektować i zaimplementować interfejs użytkownika aplikacji iOS z użyciem Auto Layout i podstawowych komponentów.</p> <p>U03. Opracować plan testowania aplikacji iOS i przeprowadzić testy podstawowych funkcjonalności.</p> <p>U04. Analizować i optymalizować interfejs i logikę aplikacji iOS, proponując usprawnienia w zakresie wydajności i użyteczności.</p> <p>U05. Uwzględnić podstawowe aspekty prawne (np. licencje) i ergonomiczne przy projektowaniu prostej aplikacji iOS.</p>	<p>K_U01 K_U02 K_U03 K_U04 K_U09 K_U10 K_U15 K_U17 K_U23</p>	<p>P6S_UW P6S_UW_INŻ P6S_UO P6S_KK P6S_UK</p>
KOMPETENCJE SPOŁECZNE – absolwent jest gotów do	<p>K01. Pracy w zespole, przyjmując w nim różne role.</p> <p>K02. Krytycznej oceny możliwości urządzeń, oprogramowania i systemów dostępnych na rynku IT.</p> <p>K03. Ciągłego samokształcenia się w celu dostosowywania się do dynamicznie zmieniających się technologii.</p>	<p>K_K04 K_K05 K_K06</p>	<p>P6S_UO P6S_KR P6S_KK</p>

Lp.	Tematyka zajęć	Liczba godzin
Forma zajęć – laboratorium		
1	Podstawy systemu iOS: architektura, środowisko deweloperskie Xcode.	2
2	Podstawy języka Swift. Składnia, typy danych, kontrola przepływu.	10
3	Interfejs użytkownika w iOS: Storyboards, UIKit, Auto Layout.	4
4	View Controllers i nawigacja. Zarządzanie ekranami, przejścia między widokami.	4
5	Przechowywanie danych: UserDefaults, pliki, wprowadzenie do Core Data.	2

AKADEMIA TECHNICZNO-INFORMATYCZNA W NAUKACH STOSOWANYCH

6	Tworzenie prostej aplikacji na iOS z interfejsem graficznym. Zaliczenie.	8
---	--	---

Forma i warunki zaliczenia przedmiotu	Wykonanie projektów. Częstkowe prezentacje, zdawanie raportów, obrona projektów.	
Metody weryfikacji efektów uczenia się		Nr efektu uczenia się z sylabusu
	Ocena projektów i częściowych prezentacji.	W01-W05, U01-U05, K01-K03

Literatura podstawowa	<ol style="list-style-type: none"> 1. J. Hoffman, <i>Swift 4. Koduj jak mistrz</i>, Helion, Gliwice 2018. 2. I. Sommerville, <i>Inżynieria oprogramowania</i>, WNT, Warszawa 2020. 3. M. Śmiałek, K. Rybiński, <i>Inżynieria oprogramowania w praktyce. Od wymagań do kodu z językiem UML</i>, Helion, Gliwice 2004.
Literatura uzupełniająca	<ol style="list-style-type: none"> 1. N. Wirth, <i>Algorytmy + struktury danych = programy</i>, WNT, Warszawa 2002.

Nakład pracy studenta	
	Liczba godzin
Zajęcia dydaktyczne	30
Przygotowanie się do zajęć	5
Studiowanie literatury	5
Udział w konsultacjach	2
Przygotowanie projektu / eseju / prezentacji itp.	18
Przygotowanie się do egzaminu / zaliczenia	-
Inne	-
ŁĄCZNY nakład pracy studenta w godz.	60
Liczba punktów ECTS	2