



**POLITECHNIKA
BYDGOSKA**

Wydział Telekomunikacji,
Informatyki i Elektrotechniki

Karta przedmiotu AI na urządzeniach brzegowych

1. Informacje podstawowe

Kierunek studiów informatyka stosowana Specjalność: sztuczna inteligencja Jednostka zarządzająca kierunkiem studiów Wydział Telekomunikacji, Informatyki i Elektrotechniki Poziom studiów drugiego stopnia (mgr inż.) Profil studiów Profil ogólnoakademicki Forma studiów studia niestacjonarne		Cykl kształcenia (nabór) 2025/26 Kod przedmiotu 05ISTSZIN.DI2D.3976.25 Języki wykładowe polski Obligatoryjność Obligatoryjny specjalnościowy Blok zajęciowy Przedmioty specjalnościowe	
Wymagania wstępne	Podstawy programowania mikrokontrolerów.		
Przedmioty wprowadzające	Brak przedmiotów wprowadzających.		
Koordynator	Tomasz Marciniak, Jan Baumgart		
Okres Semestr 2	Forma zaliczenia Zaliczenie na ocenę Forma prowadzenia i godziny zajęć Wykład: 18 Ćwiczenia laboratoryjne: 18		Liczba punktów ECTS 4.0

2. Efekty uczenia się dla przedmiotu

Kod	Opis efektów uczenia się	Odniesienie do kierunkowych efektów uczenia się	Odniesienie do charakterystyk PRK
Wiedza:			
W1	Student rozumie podstawowe zasady działania modeli AI na mikrokontrolerach i potrafi wybrać odpowiednią platformę (np. Arduino Nano, ESP32) dla prostych projektów AI.	IST_O2_K_W12, IST_O2_K_W13	P7S_WG, P7S_WG_inż, P7S_WG P7S_WG_inż
W2	Student zna techniki kompresji modeli (kwantyzacja, pruning) i ich zastosowanie w projektach TinyML.	IST_O2_K_W02, IST_O2_K_W05, IST_O2_K_W15	P7S_WG, P7S_WG_inż, P7S_WG, P7S_WG_inż, P7S_WG P7S_WG_inż
W3	Student rozumie ograniczenia pamięciowe i obliczeniowe mikrokontrolerów oraz potrafi zoptymalizować modele AI pod kątem tych ograniczeń.	IST_O2_K_W02, IST_O2_K_W03	P7S_WG, P7S_WG_inż, P7S_WG P7S_WG_inż
W4	Student posiada wiedzę na temat integracji sensorów z modelami AI oraz przetwarzania danych wejściowych na mikrokontrolerach.	IST_O2_K_W04, IST_O2_K_W12	P7S_WG, P7S_WG_inż, P7S_WG P7S_WG_inż
W5	Student zna podstawy bezpieczeństwa danych i prywatności w kontekście urządzeń brzegowych.	IST_O2_K_W07, IST_O2_K_W10	P7S_WG, P7S_WG_inż, P7S_WG P7S_WG_inż
Umiejętności:			
U1	Student implementuje prosty model AI na Arduino Nano lub ESP32	IST_O2_K_U02, IST_O2_K_U03	P7S_UW, P7S_UW_inż, P7S_UK
U2	Student konfiguruje sensory (np. akcelerometr, mikrofon) i integruje dane wejściowe z modelem AI na mikrokontrolerze.	IST_O2_K_U13	P7S_UW P7S_UW_inż
U3	Student potrafi przeprowadzić testy wydajnościowe modelu AI na mikrokontrolerze, oceniając czas inferencji i zużycie pamięci RAM.	IST_O2_K_U07, IST_O2_K_U12	P7S_UW, P7S_UW_inż, P7S_UO
U4	Student projektuje systemy AI uwzględniające ograniczenia energetyczne urządzeń brzegowych.	IST_O2_K_U15, IST_O2_K_U16	P7S_UW, P7S_UW_inż, P7S_UW P7S_UW_inż
U5	Student analizuje dokumentację techniczną narzędzi do pracy z AI na mikrokontrolerach.	IST_O2_K_U01, IST_O2_K_U05	P7S_UW, P7S_UW_inż, P7S_UU
Kompetencje społeczne:			
K1	Student potrafi komunikować wyniki projektu w sposób jasny i zrozumiały dla różnych odbiorców.	IST_O2_K_K04	P7S_KO
K2	Student wykazuje otwartość na współpracę międzydiscyplinarną w projektach IoT i AI.	IST_O2_K_K03	P7S_KO

3. Treści programowe

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
1.	Student rozumie podstawowe pojęcia sztucznej inteligencji, architekturę mikrokontrolerów oraz różnice między AI w chmurze a AI na urządzeniach brzegowych.	Wykład, Ćwiczenia laboratoryjne	W1, U1, K1

Lp.	Treści programowe	Formy zajęć	Efekty uczenia się dla przedmiotu
2.	Student zna techniki kompresji modeli (kwantyzacja, pruning) i ich zastosowanie w projektach TinyML, potrafi zoptymalizować model pod kątem ograniczeń sprzętowych.	Wykład, Ćwiczenia laboratoryjne	W2, U3, K2
3.	Student rozumie ograniczenia pamięciowe i obliczeniowe mikrokontrolerów oraz potrafi zoptymalizować modele AI pod kątem tych ograniczeń.	Wykład, Ćwiczenia laboratoryjne	W3, U4, K2
4.	Student posiada wiedzę na temat integracji sensorów z modelami AI oraz przetwarzania danych wejściowych na mikrokontrolerach, konfiguruje sensory i integruje dane z modelem.	Wykład, Ćwiczenia laboratoryjne	W4, U2, K1
5.	Student zna podstawy bezpieczeństwa danych i prywatności w kontekście urządzeń brzegowych, potrafi wskazać zagrożenia i zaproponować podstawowe środki ochrony.	Wykład, Ćwiczenia laboratoryjne	W5, U5
6.	Student implementuje i testuje prosty model AI na Arduino/ESP32 w MicroPython, przeprowadza testy wydajnościowe i analizuje wyniki działania modelu.	Wykład, Ćwiczenia laboratoryjne	W1, U1, K2
7.	Student projektuje system AI z uwzględnieniem ograniczeń energetycznych urządzenia, analizuje wpływ optymalizacji na zużycie energii.	Wykład, Ćwiczenia laboratoryjne	W3, U4
8.	Student analizuje dokumentację techniczną narzędzi do AI na mikrokontrolerach i prezentuje wyniki projektu w sposób jasny i zrozumiały dla różnych odbiorców.	Wykład, Ćwiczenia laboratoryjne	W3, U4, K1

4. Metody prowadzenia zajęć, weryfikacji efektów uczenia się i warunki zaliczenia

Forma zajęć		
Wykład	Metody prowadzenia zajęć:	
	Wykład	
	Metody (sposoby) weryfikacji:	Udział:
	Test wielokrotnego wyboru	100%
	Warunki zaliczenia przedmiotu:	
	Forma zaliczenia obejmuje test wielokrotnego wyboru, w którym student wybiera jedną lub więcej poprawnych odpowiedzi spośród kilku podanych wariantów. Za każdą poprawną odpowiedź przyznawane są punkty. Łączna liczba uzyskanych punktów jest przeliczana na procenty, a następnie oceniana zgodnie z procentowymi zakresami ocen określonymi w regulaminie studiów. Warunkiem zaliczenia jest uzyskanie minimalnej liczby punktów odpowiadającej ocenie pozytywnej. Forma zaliczenia zajęć może również zostać zmieniona przez prowadzącego	

Ćwiczenia laboratoryjne	Metody prowadzenia zajęć:	
	Ćwiczenia laboratoryjne	
	Metody (sposoby) weryfikacji:	Udział:
	Sprawozdanie	100%
	Warunki zaliczenia przedmiotu:	
	Oddanie wszystkich sprawozdań i uzyskanie min. oceny dostatecznej za sprawozdanie, ocena końcowa średnia arytmetyczna ocen cząstkowych	

Efekt uczenia się dla przedmiotu	Metody (sposoby) weryfikacji	
	Test wielokrotnego wyboru	Sprawozdanie
W1	x	
W2	x	
W3	x	
W4	x	
W5	x	
U1		x
U2		x
U3		x
U4		x
U5		x
K1		x
K2		x

5. Literatura

Literatura podstawowa

1. Warden, P., Situnayake, D., 2019. TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly Media.
2. Banzi, M., Shiloh, M., 2022. Getting Started with Arduino: The Open Source Electronics Prototyping Platform. Maker Media.
3. Lobur, M., Lobur, O., 2021. MicroPython for Microcontrollers: Getting Started with MicroPython on ESP32 and ESP8266. Apress.
4. Atul Krishna Gupta, 2023 Dr. Siva Prasad Nandyala, Deep Learning on Microcontrollers, BPB Publications

Literatura uzupełniająca

1. Latif, S., Rana, R., 2021. Machine Learning for Embedded Systems: A Practical Approach to TinyML and Edge AI. Packt Publishing.
2. Khan, M., 2022. Practical MicroPython: Building Real-World Embedded Systems. Apress.
3. Espressif, 2024. ESP32 Technical Reference Manual. Espressif Systems, dostęp online.

6. Nakład pracy studenta - bilans godzin i punktów ECTS

Aktywność studenta		Obciążenie studenta Liczba godzin
Zajęcia prowadzone z bezpośrednim udziałem nauczyciela akademickiego lub innych osób prowadzących zajęcia	Wykład	18
	Ćwiczenia laboratoryjne	18
Praca własna studenta	Studiowanie literatury	9
	Konsultacje	3
	Przygotowanie do zajęć	49
	Przygotowanie sprawozdania	15
	Przygotowanie do zaliczenia	8
Łączny nakład pracy studenta		120
Liczba punktów ECTS		4

* Godzina (dydaktyczna) oznacza 45 minut