

KARTA KURSU (realizowanego w specjalności)

CYBERBEZPIECZEŃSTWO

(nazwa specjalności)

Nazwa	Tworzenie bezpiecznych aplikacji
Nazwa w j. ang.	Security coding

Koordynator	dr hab. prof. UKEN Volodymyr Alekseyev	Zespół dydaktyczny
		dr hab. prof. UKEN Volodymyr Alekseyev
Punktacja ECTS*	st. stacjonarne: 3 st. niestacjonarne: 3	

Opis kursu (cele kształcenia)

Celem kursu jest zapoznanie studentów z:

- tworzenie bezpiecznych aplikacji internetowych, cecha rozwoju aplikacji desktopowych i mobilnych;
- cechami cyklu rozwoju oprogramowania i metodologią DevSecOps;
- metodami ochrony interfejsu programistycznego aplikacji (mikroserwisów).

Kurs prowadzony jest w języku polskim i angielskim.

Warunki wstępne

Wiedza	Działanie i funkcjonowanie sieci. Protokoły internetowe. Podstawy programowania obiektowego.
Umiejętności	Programowanie w jednym z popularnych języków do tworzenia aplikacji webowych, np. PHP, Python, Java, C#.

Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
Wiedza	W01: zna zasady działania głównych narzędzi kryptograficznych służących do przechowywania poufnych danych (hasel) w programie (aplikacji) oraz narzędzi szyfrujących interfejs programowania aplikacji (API).	SC_W01, SC_W02, SC_W03
	W02: zna cykl życia oprogramowania w kontekście wymagań bezpieczeństwa i metodologii DevSecOps.	
	W03: zna zagadnienia związane z tworzeniem aplikacji, a także zasadami ich organizacji i administrowania, z naciskiem na bezpieczeństwo systemów serwerowych i rozwiązań chmurowych.	

	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
Umiejętności	U01: wdraża i stosuje kryptograficzne metody i środki bezpieczeństwa informacji.	SC_U01, SC_U02, SC_U03, SC_U04, SC_U07.
	U02: potrafi budować algorytmy i pisać indywidualne aplikacje w oparciu o języki programowania aplikacji internetowych z uwzględnieniem zasad bezpieczeństwa.	
	U03: umie wykorzystać specjalistyczne środowiska programistyczne wraz z wybranymi bibliotekami do efektywnego i bezpiecznego tworzenia aplikacji desktopowych, mobilnych lub webowych.	
	U04: potrafi analizować i projektować interfejsy API, stosując odpowiednie metody, techniki i narzędzia oraz uwzględniając aspekty związane z bezpieczeństwem ich użytkowania.	
	U07: potrafi rozwiązywać problemy związane z analizą kodu programu pod kątem możliwych zagrożeń.	

	Efekt uczenia się dla kursu	Odniesienie do efektów dla specjalności (określonych w karcie programu studiów dla specjalności)
Kompetencje społeczne	K02: potrafi sformułować koncepcję tworzenia nowoczesnych aplikacji, określić środki wspierające cykl życia aplikacji, wybrać optymalny API, ze szczególnym uwzględnieniem aspektów cyberbezpieczeństwa.	SC_K02.

Studia stacjonarne

Organizacja												
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		Z
Liczba godzin	15					30						

Studia niestacjonarne

Organizacja												
Forma zajęć	Wykład (W)	Ćwiczenia w grupach										
		A		K		L		S		P		Z
Liczba godzin	10					20						

Opis metod prowadzenia zajęć

Kurs składa się z wykładów oraz zajęć praktycznych, które prowadzone są w formie zajęć laboratoryjnych. W ramach zajęć laboratoryjnych studenci opracowują i tworzą zadane programy w PHP i JavaScript, które następnie są omawiane, korzystając z nowoczesnych narzędzi wspierających cykl życia aplikacji. Oprócz zajęć tradycyjnych studenci uczestniczą w zajęciach z wykorzystaniem platformy e-learningowej (Microsoft Teams).

Formy sprawdzania efektów uczenia się

	E – learning	Gry dydaktyczne	Ćwiczenia w szkole	Zajęcia terenowe	Praca laboratoryjna	Projekt indywidualny	Projekt grupowy	Udział w dyskusji	Referat	Praca pisemna (esej)	Egzamin ustny	Egzamin pisemny	Inne (Zadania problemowe)
W01	X							X					X
W02	X				X								X
W03	X				X								
U01	X				X								
U02					X								
U03	X				X			X					
U04	X				X								X
U07					X			X					
K02	X				X			X					X

Kryteria oceny	<p>Pomyślne ukończenie kursu obejmuje tworzenie przez studenta projektu w celu wdrożenia bezpiecznej aplikacji, która w znacznym stopniu wykorzystuje wiedzę zdobytą podczas kursu.</p> <p>Ocena "dobry" lub "bardzo dobry" może zostać przyznana studentowi, który:</p> <ul style="list-style-type: none"> - zna nowoczesne metody budowania bezpiecznych aplikacji; - potrafi zdefiniować algorytm i architekturę aplikacji, określić sposoby wsparcia cyklu życia produktu; - jest świadomy zagrożeń cyberbezpieczeństwa dla współczesnych aplikacji; - potrafi formułować wymagania dotyczące tworzenia bezpiecznych aplikacji.
----------------	---

Uwagi	<p>Kurs ma na celu zapoznanie studentów z najlepszymi praktykami tworzenia bezpiecznych aplikacji. Kurs koncentruje się na aplikacjach internetowych, w tym hybrydowych aplikacjach mobilnych i desktopowych (opracowanych na stosie technologii JavaScript, CSS, HTML). Tworzenie aplikacji po stronie serwera jest rozważane w języku PHP. Analizowany jest cykl życia aplikacji internetowej.</p>
-------	--

Treści merytoryczne (wykaz tematów)

1. Rozwój aplikacji internetowych. Główne luki w aplikacjach internetowych i metody opracowywania bezpiecznych rozwiązań (OWASP Top 10).
2. Zrozumienie architektury mikrousług. Budowanie bezpiecznego API.
3. Tworzenie aplikacji przy użyciu metodologii DevSecOps.
4. Właściwości tworzenia bezpiecznych aplikacji mobilnych.
5. Rozwój aplikacji desktopowych. Bezpieczne tworzenie skomplikowanych aplikacji.

Wykaz literatury podstawowej

1. Alicja i Bob. Bezpieczeństwo aplikacji w praktyce, Tanya Janca, Helion 2022.
2. Bezpieczeństwo aplikacji internetowych dla programistów. Rzeczywiste zagrożenia, praktyczna ochrona, Malcolm McDonald, Helion 2021.
3. Bezpieczeństwo kontenerów w DevOps. Zabezpieczanie i monitorowanie kontenerów Docker, Jose Manuel Ortega Candel, Helion 2021.
4. PHP 8 i SQL. Programowanie dla początkujących w 43 lekcjach, Mariusz Duka, Helion 2021.
5. Bezpieczne programowanie. Aplikacje hakeroodporne, Jacek Ross, Helion 2009.

Wykaz literatury uzupełniającej

1. Cyberbezpieczeństwo dla zaawansowanych. Skuteczne zabezpieczenia systemu Windows, Linux, IoT i infrastruktury w chmurze, Cesar Bravo, Helion 2023.
2. Algorytmy kryptograficzne. Przewodnik po algorytmach w blockchain, kryptografii kwantowej, protokołach o wiedzy zerowej oraz szyfrowaniu homomorficznym, Massimo Bertaccini, Helion 2023.
3. Bug Bounty Bootcamp. Przewodnik po tropieniu i zgłaszaniu luk w zabezpieczeniach, Vickie Li, Helion 2022.

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) Studia stacjonarne

Ilość godzin w kontakcie z prowadzącymi	Wykład	15
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	5
Ilość godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	5
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	15
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	5
	Przygotowanie do egzaminu	-
Ogółem bilans czasu pracy		75
Ilość punktów ECTS w zależności od przyjętego przelicznika		3

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) Studia niestacjonarne

Ilość godzin w kontakcie z prowadzącymi	Wykład	10
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	5
Ilość godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Przygotowanie krótkiej pracy pisemnej lub referatu po zapoznaniu się z niezbędną literaturą przedmiotu	20
	Przygotowanie projektu lub prezentacji na podany temat (praca w grupie)	5
	Przygotowanie do egzaminu	-
Ogółem bilans czasu pracy		75
Ilość punktów ECTS w zależności od przyjętego przelicznika		3