

Semestr VI

Sylabus przedmiotu / modułu kształcenia:	
Nazwa przedmiotu/modułu kształcenia:	Seminarium dyplomowe
Nazwa w języku angielskim:	Diploma seminar
Język wykładowy:	polski
Kierunek studiów, dla którego przedmiot jest oferowany:	informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	obowiązkowy
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	pierwszego stopnia
Rok studiów:	Trzeci
Semestr:	szósty i siódmy
Liczba punktów ECTS:	0
Imię i nazwisko koordynatora przedmiotu:	prof. dr hab. inż. Andrzej Barczak
Imię i nazwisko prowadzących zajęcia:	Promotorzy, promotorzy pomocniczy lub opiekunowie dyplomantów
Założenia i cele przedmiotu:	<p>Celem zajęć jest zapoznanie studenta z metodyką przygotowania pracy dyplomowej i jej ustnej prezentacji, jak też przygotowanie studenta do egzaminu dyplomowego.</p> <p>Celem przedmiotu jest także weryfikacja umiejętności wykonania przez studentów zaawansowanego projektu inżynierskiego. Realizacja tego celu wiąże się z osiągnięciem następujących celów szczegółowych:</p> <ul style="list-style-type: none"> • umiejętność opracowania prawidłowego harmonogramu prac zarówno pod względem chronologicznym jak kalendarzowym, • umiejętność systematycznej pracy nad projektem, zgodnie z zatwierdzonym harmonogramem, • umiejętność rozwiązywania problemów o charakterze inżynierskim oraz prezentowania opracowanych rozwiązań, - umiejętność raportowania postępów w pracy, <p>umiejętność przygotowania pisemnego opracowania w postaci pracy dyplomowej towarzyszącej dyplomowemu projektowi inżynierskiemu.</p>

Sylabus przedmiotu / modułu kształcenia:		
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	zna i rozumie cywilizacyjne znaczenie informatyki, zna i rozumie jej rolę w życiu społeczeństwa oraz zagrożenia związane z jej zastosowaniami i uwzględnia te aspekty w swojej pracy dyplomowej	K_W03
W_02	Zna i rozumie zagadnienia z zakresu ochrony własności intelektualnej	K_W04
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	potrafi pozyskiwać informacje z literatury, baz danych i innych, w tym zwłaszcza internetowych źródeł; potrafi analizować, interpretować oraz integrować uzyskane informacje, a także oceniać ich użyteczność w aspekcie wykonywanej pracy kwalifikacyjnej, wyciągać wnioski oraz formułować i uzasadniać opinie	K_U01
U_02	potrafi wykorzystać poznane metody i modele matematyczne — w razie potrzeby odpowiednio je modyfikując — do analizy i projektowania systemów informatycznych, w tym zwłaszcza w zakresie wybranej specjalności	K_U02, K_U09
U_03	Potrafi posłużyć się właściwie dobranymi środowiskami programistycznymi i narzędziami komputerowymi wspomagającymi proces projektowania, implementacji i testowania systemów informatycznych, potrafi zaplanować realizację poszczególnych etapów rozwiązywania zadań z zakresu informatyki	K_U12, K_U13
U_04	Potrafi ocenić i porównać rozwiązania projektowe systemów informatycznych ze względu na zadane kryteria użytkowe i ekonomiczne	K_U16
U_05	Potrafi stosować w rozwiązywanych zadaniach standardy i normy inżynierskie i technologie informatyczne wykorzystując zdobyte doświadczenie w środowisku zajmującym się zawodowo działalnością inżynierską	K_U24
U_06	potrafi dostrzegać aspekty pozatechniczne, w tym środowiskowe, ekonomiczne i prawne przy projektowaniu, implementowaniu i wdrażaniu systemów informatycznych	K_U23
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego

Sylabus przedmiotu / modułu kształcenia:		
K_01	jest gotów do odpowiedzialnego pełnienia roli zawodowej informatyka, w tym do przestrzegania zasad etyki zawodowej i wymagania tego od innych i do dbania o dorobek i tradycję zawodu informatyka	K_K04
K_02	jest gotów do krytycznej oceny posiadanej wiedzy oraz jest gotów do uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych z zakresu informatyki	K_K01
K_03	Jest gotów do myślenia i działania w sposób przedsiębiorczy	K_K03
Forma i typy zajęć:	Studia stacjonarne: seminarium semestr VI 15 godz., semestr VII 60 godz., Studia niestacjonarne: seminarium semestr VI 9 godz., semestr VII 45 godz.	
Wymagania wstępne i dodatkowe:		
Znajomość treści nauczania w dotychczasowym przebiegu studiów. Znajomość programów komputerowych do edycji, opracowania wyników oraz ich prezentacji.		
Treści modułu kształcenia:		
<p>Treści modułu kształcenia zawierają elementy wszystkich wykładów przewidzianych kursem I stopnia studiów inżynierskich na studiowanym kierunku i wybranej specjalności. Największy nacisk jest położony na podnoszenie umiejętności i kompetencji w zakresie opracowywania i prezentowania wyników oraz korzystania z fachowej literatury, także w języku angielskim.</p> <p>Główne zagadnienia:</p> <ol style="list-style-type: none"> 1. Rola zajęć seminaryjnych i promotora w tworzeniu pracy. 2. Precyzowanie obszaru zainteresowań dyplomantów. Omówienie przez prowadzącego proponowanej tematyki prac dyplomowych. 3. Sposoby zbierania materiałów źródłowych. Omówienie przez prowadzącego źródeł informacji naukowej oraz sposobów jej wyszukiwania i wykorzystywania. 4. Omówienie przez prowadzącego zasad przygotowywania treści referatów i ich multimedialnej prezentacji. 5. Prezentacja przez studentów referatów z zakresu wstępnie wybranej tematyki i celów ich prac dyplomowych – z wykorzystaniem środków multimedialnych, dyskusja i ewentualna korekta tematów prac. 6. Dyskusja mająca na celu ostateczne sformułowanie tematów prac dyplomowych i określenie ich zakresu. 7. Układ pracy dyplomowej. Konstrukcja poszczególnych rozdziałów i ich rola w całości pracy. <ul style="list-style-type: none"> • Omówienie zasad redagowania pracy dyplomowej i kryteriów jej oceny. • Omówienie poszczególnych etapów procesu realizacji projektów dyplomowych • Wymogi edytorskie – układ tekstu na stronie, typografia tekstu, pisownia nazw obcych i skrótów, składnia i ortografia. Wymogi konstrukcyjne ilustracji (tabel, wykresów, schematów, rycin, fotografii). 		

Sylabus przedmiotu / modułu kształcenia:
<ul style="list-style-type: none"> • Tworzenie bibliografii i zasady powołań literaturowych. Ustalanie spisów treści poszczególnych prac. Dyskusja nad metodyką postępowania w przygotowaniu pracy. Rola adiustacji tekstu. <p>8. Prezentacja wyników realizacji określonych harmonogramami poszczególnych etapów pracy studentów nad projektami dyplomowymi, dyskusja i ewentualne zalecenia korygujące.</p> <p>9. Omówienie zasad postępowania związanych z działaniem Archiwum Prac Dyplomowych (APD).</p> <p>10. Omówienie przebiegu egzaminu dyplomowego i obowiązującego na nim zestawu pytań.</p>
Literatura podstawowa:
<p>1. Specyficzna dla danej dziedziny i konkretnego tematu zlecona przez opiekuna/promotora pracy oraz samodzielnie wyszukane przez studenta w dostępnych źródłach (również elektronicznych) zarówno w języku polskim jak i obcym oraz, m.in.: Zarządzenie nr 26/2016 Rektora Uniwersytetu Przyrodniczo-Humanistycznego w Siedlcach z dnia 22 kwietnia 2016 roku w sprawie określenia warunków, jakim powinna odpowiadać praca dyplomowa oraz zasad ich archiwizowania</p> <p>2. Sobaniec C.: Jak pisać pracę inżynierską/magisterską. www.cs.put.poznan.pl/sobaniec/edu/jak_pisacmgr.pdf</p> <p>3. Starecki T.: Praca dyplomowa – jak realizować, jak pisać i dlaczego. www.ise.pw.edu.pl/impuls/Dyplom.pdf</p> <p>4. Opoka E.: Uwagi o pisaniu i redagowaniu prac dyplomowych na studiach technicznych. Wyd. Politechnika Śląska, Gliwice 2001</p> <p>5. Wytrębowski J.: O poprawności językowej publikacji naukowo-technicznych. w: Zagadnienia Naukoznawstwa, Nr 1(179) 2009</p>
Literatura dodatkowa:
<p>1. Kwaśniewski A.: Jak pisać pracę dyplomową. http://zpt2.tele.pw.edu.pl/~andrzej/TP/wyklad/wyklad-pdf/TPpraca_dypl.pdf</p> <p>2. Drozdowski M. Jak pisać prace dyplomową/magisterską: http://www.cs.put.poznan.pl./mdrozdowski/dyd/txt/jak_mgr.html.</p>
Planowane formy/działania/metody dydaktyczne:
<p>Metoda słowna problemowa, dyskusja, prezentacja multimedialna, prezentacja ustna.</p> <p>Prezentacja celów pracy dyplomowej</p> <p>Prezentacja koncepcji pracy dyplomowej z zastosowaniem informatycznych programów, systemów oraz technik audiowizualnych</p>
Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:
<p>Weryfikację założonych efektów uczenia się dokonują w sposób ciągły prowadzący seminaria dyplomowe na podstawie udziału w dyskusji oraz podczas przedstawianych prezentacji na poszczególnych etapach pracy dyplomowej (I – uzasadnienie wyboru tematu, studia literaturowe, II – postawienie hipotezy, omówienie wybranej metody badawczej i przedstawienie uzyskanych wyników/projektów/rozwiązań, wnioski, weryfikacja postawionej hipotezy)</p> <p>W szczególności efekty uczenia się W_01, U_01, U_02 oraz K_01, K_02 i K_K03 weryfikowane będą w toku zajęć seminaryjnych na podstawie przygotowania, udziału i aktywności poszczególnych studentów w zajęciach.</p>
Forma i warunki zaliczenia:

Sylabus przedmiotu / modułu kształcenia:	
Moduł podlega zaliczeniu (bez oceny). Podstawą zaliczenia jest przygotowanie prezentacji dotyczącej opracowywanej pracy inżynierskiej	
Bilans punktów ECTS:	
Studia stacjonarne	
Aktywność	Obciążenie studenta
Udział w zajęciach	75 godzin
Przygotowanie prezentacji multimedialnej	30 godzin
Konsultacje indywidualne i przygotowanie autoreferatu, korekta pracy dyplomowej	60 godzin
Samodzielne studia literaturowe	30 godzin
Opracowywanie projektu i pisanie pracy dyplomowej	100 godzin
Przygotowanie się do egzaminu dyplomowego	80 godzin
Sumaryczne obciążenie pracą studenta	375 godzin
Punkty ECTS za przedmiot	15 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w zajęciach	54 godzin
Przygotowanie prezentacji multimedialnej	30 godzin
Konsultacje indywidualne i przygotowanie autoreferatu, korekta pracy dyplomowej	50 godzin
Samodzielne studia literaturowe	41 godzin
Opracowywanie projektu i pisanie pracy dyplomowej	100 godzin
Przygotowanie się do egzaminu dyplomowego	100 godzin
Sumaryczne obciążenie pracą studenta	375 godzin
Punkty ECTS za przedmiot	15 ECTS

Sylabus przedmiotu / modułu kształcenia	
Nazwa przedmiotu/modułu kształcenia:	Praktyka zawodowa (III)
Nazwa w języku angielskim:	Apprenticeship III
Język wykładowy:	polski
Kierunek studiów, dla którego przedmiot jest oferowany:	Informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	obowiązkowy
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	pierwszego stopnia
Rok studiów:	trzeci
Semestr :	szósty
Liczba punktów ECTS:	10
Imię i nazwisko koordynatora przedmiotu:	dr Marcin Stępniaik
Imię i nazwisko prowadzących zajęcia:	Osoba delegowana z firmy/institucji
Założenia i cele przedmiotu:	<p>Cele praktyki:</p> <ul style="list-style-type: none"> • pogłębienie umiejętności zawodowych studenta, • pogłębienie specjalistycznej wiedzy, umiejętności i kompetencji związanych z funkcjonowaniem firmy/institucji w zakresie stosowanych systemów informatycznych oraz rozwoju istniejących systemów i wytwarzania nowych aplikacji, • pogłębienie i wykorzystanie w praktyce wiedzy i umiejętności nabytych podczas nauki oraz poprzednich etapów praktyki zawodowej, zwłaszcza tych związanych z wybraną specjalnością, • zdobycie wiedzy i umiejętności związanych ze sposobami organizacji pracy indywidualnej i zespołowej, • nawiązanie kontaktów zawodowych ułatwiających poszukiwanie pracy, firmy/institucji do odbycia kolejnych etapów praktyk, czy realizacji prac dyplomowych.

Symbo l efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Posiada wiedzę w zakresie organizacji i zarządzania czasem oraz budowy harmonogramu pracy indywidualnej i zespołowej.	K_W12
W_02	Posiada pogłębioną specjalistyczną wiedzę dziedzinową związaną z systemami i narzędziami informatycznymi wykorzystywanymi w firmie.	K_W12
W_03	Zna i rozumie w zaawansowanym stopniu zagadnienia związane z wytwarzaniem oprogramowania oraz administrowaniem systemami informatycznymi.	K_W06, K_W07
W_04	Student orientuje się w potrzebach rynku pracy. Zna relację między wymaganiami pracodawców a wiedzą zdobytą w trakcie zajęć.	K_W12, K_W03
Symbo l efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi planować czas pracy, nadawać priorytety zadaniom i je terminowo realizować. Potrafi wykorzystać wiedzę teoretyczną, praktyczną oraz narzędzia informatyczne do realizacji postawionych zadań.	K_U05, K_U23, K_U24
U_02	Potrafi zadbać o własny wizerunek zawodowy. Potrafi zidentyfikować kierunki dalszego rozwoju na podstawie pozyskanej wiedzy, umiejętności oraz doświadczeń zawodowych. Potrafi nawiązywać i utrzymywać kontakty zawodowe.	K_U04, K_U06, K_U23
U_03	Potrafi wykorzystać w praktyce wiedzę i umiejętności związane z realizowaną specjalnością. Potrafi w praktyce stosować zasady bezpieczeństwa i ergonomii pracy.	K_U14, K_U23, K_U25
Symbo l efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	Jest gotów do odpowiedzialnego pełnienia roli zawodowej informatyka, w tym do przestrzegania zasad etyki zawodowej i wymagania tego od innych i do dbania o dorobek i tradycję zawodu informatyka.	K_K04
K_02	Krytycznie ocenia działania własne, działania zespołów, którymi kieruje i organizacji w której uczestniczy. Potrafi dokonać samooceny własnych kompetencji i doskonalenia swoich kwalifikacji zawodowych.	K_K01, K_K03
Forma i typy zajęć:		Studia stacjonarne i niestacjonarne: praktyka (320 godzin)
Wymagania wstępne i dodatkowe:		
1. Podstawowa wiedza z zakresu funkcjonowania, projektowania i implementacji systemów informatycznych.		

2. Wiedza, umiejętności i kompetencje społeczne nabyte podczas praktyki po pierwszym i drugim roku studiów.

Treści modułu kształcenia:

1. Poznanie zasad bezpieczeństwa i higieny pracy na danym stanowisku oraz uwarunkowania prawne i etyczne stosownie do wykonywanych obowiązków.
2. Poznanie specyfiki działania przedsiębiorstwa, w którym jest odbywana praktyka.
3. Rozpoznanie obszarów działalności firmy wspomaganymi komputerowo.
4. Zapoznanie z systemami i narzędziami informatycznymi wspomagającymi działalność firmy, a zwłaszcza wspierającymi zarządzanie i produkcję. W szczególności należy:
 - a. zapoznać się z dokumentacją techniczną sprzętu i oprogramowania,
 - b. rozpoznawać i rozwiązywać problemy związane z eksploatacją sprzętu i oprogramowania,
 - c. studiować możliwości optymalizacji, rozbudowy i modyfikacji infrastruktury teleinformatycznej, zgodnie z aktualnymi tendencjami rozwojowymi.
5. Dokonanie oceny istniejącej infrastruktury i wykorzystywanych technologii informatycznych w przedsiębiorstwie pod kątem zgodności ze standardami oraz możliwości rozwoju i współpracy z innymi rozwiązaniami.
6. Ocena aktualnego stanu oraz przyszłych potrzeb systemów informatycznych.
7. Współdziałanie w projektowaniu nowych i ulepszaniu istniejących systemów informatycznych, biorąc pod uwagę:
 - a. wymagania i cele stawiane przed systemem informatycznym,
 - b. politykę bezpieczeństwa oraz procedury organizacyjne dotyczące wykorzystania infrastruktury informatycznej,
 - c. napotymane ograniczenia techniczne i biznesowe,
 - d. zagadnienia związane ze zwrotem kosztów inwestycji.
8. Realizacja zadań związanych z kierunkiem informatyka, a w szczególności ze specjalnością wybraną przez studenta odbywającego praktyki.
9. Prowadzenie dokumentacji przebiegu praktyk.

Literatura podstawowa:

Według zalecenia w miejscu odbywania praktyki.

Literatura dodatkowa:

Regulamin praktyk

Planowane formy/działania/metody dydaktyczne:

Cykl spotkań informacyjnych odnośnie celów i zakresu praktyki, wymaganych dokumentów i terminów oraz indywidualne konsultacje.

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Wyrywkowa hospitacja w miejscu praktyki, rozmowa ze studentem, ocena przedstawionej dokumentacji.

Forma i warunki zaliczenia:

Podstawą zaliczenia modułu jest ocena wystawiona studentowi w instytucji przyjmującej na praktykę i weryfikowana przez opiekuna praktyk na podstawie rozmowy lub arkusza hospitacyjnego. Ocena ta

obejmuje efekty wykonania przydzielonych zadań, jak również sposób organizacji pracy i podejmowane działania (0-50pkt). Ponadto oceniana jest dokumentacja praktyk zarówno pod kątem merytorycznym jak i formalnym (m.in. kompletność dokumentacji, dotrzymanie terminów; 0-50pkt).

Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Bilans punktów ECTS:

Studia stacjonarne i niestacjonarne

Aktywność	Obciążenie studenta
Udział w zorganizowanej formie pracy na terenie zakładu pracy – miejscu odbywania stażu	320 godzin
Sumaryczne obciążenie pracą studenta	320 godzin
Punkty ECTS za moduł	10 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Uczenie maszynowe
Nazwa w języku angielskim:		Machine learning
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		dr Piotr Świtalski
Imię i nazwisko prowadzących zajęcia:		dr Piotr Świtalski , mgr Mateusz Przychodzki
Założenia i cele przedmiotu:		Celem przedmiotu jest wprowadzenie studentów do metod uczenia maszynowego. Na wykładzie studenci zapoznają się z typowymi technikami klasyfikacji danych, metod regresji oraz redukcji wielowymiarowości. Omówione zostaną podstawy sztucznych sieci neuronowych. Zajęcia laboratoryjne poświęcone będą implementacji i weryfikacji metod oraz algorytmów poznanych na wykładzie.
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie najważniejsze paradygmaty i metody problemu uczenia maszynowego	K_W01, K_W10
W_02	Ma wiedzę na temat algorytmów nadzorowanego i nienadzorowanego uczenia maszynowego	K_W10
W_03	Wie jak ocenić i poprawić jakość modeli uczenia maszynowego	K_W01, K_W10
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi wykorzystać biblioteki implementujące algorytmy uczenia maszynowego do rozwiązania postawionego problemu	K_U02, K_U22

U_02	Potrafi zaprojektować i przygotować implementację danego algorytmu uczenia maszynowego, odpowiednią dla problemu i dopasowaną do posiadanych danych	K_U07, K_U09
Forma i typy zajęć:	Studia stacjonarne: wykłady (21 godz.), ćwiczenia laboratoryjne (24 godz.) Studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
Wymagania wstępne i dodatkowe:		
Warunkiem uczestnictwa w zajęciach jest znajomość zagadnień z przedmiotów „Analiza danych” oraz „Sztuczna inteligencja” lub znajomość literatury obowiązującej w tych przedmiotach.		
Treści modułu kształcenia:		
<ol style="list-style-type: none"> 1. Wprowadzenie do uczenia maszynowego. Rodzaje systemów uczenia maszynowego. Uczenie nadzorowane i uczenie nienadzorowane. Główne problemy uczenia maszynowego. 2. Ocena przydatności danych. Dane źródłowe. Zbiory danych uczące i testowe. Korelacja. Reprezentatywność danych. Duplikaty. Szeregi czasowe. Przetwarzanie danych. Poprawianie błędnych danych. 3. Klasyfikacja. Uczenie klasyfikatora binarnego. Miary wydajności. Klasyfikacja wieloklasowa. Analiza błędów. Klasyfikacja wieloetykietowa. Klasyfikacja wielowyjściowa. Klasyfikacja probabilistyczna. 4. Drzewa decyzyjne. Uczenie i wizualizowanie drzewa decyzyjnego. Wyliczanie prognoz. Szacowanie prawdopodobieństw przynależności do klas. Algorytm uczący CART. Złożoność obliczeniowa. Hiperparametry regularyzacyjne. Regresja. Niestabilność. 5. Maszyny wektorów nośnych. Liniowa klasyfikacja SVM. Klasyfikacja miękkiego marginesu. Nieliniowa klasyfikacja SVM. Jądro wielomianowe. Cechy podobieństwa. Gaussowskie jądro RBF. Złożoność obliczeniowa. Regresja SVM. Funkcja decyzyjna i prognozy. Programowanie kwadratowe. Problem dualny. Kernelizowane maszyny SVM. Przyrostowe maszyny SVM. 6. Regresja. Model regresji liniowej. Gradient prosty. Regresja wielomianowa. Krzywe uczenia. Regularyzowane modele liniowe. Regresja logistyczna. 7. Redukcja wielowymiarowości. Główne strategie redukcji wymiarowości. Rzutowanie. Uczenie rozmaitościowe. Analiza PCA. Zachowanie wariancji. Główne składowe. 8. Techniki uczenia nienadzorowanego. Analiza skupień. Algorytm centroidów (Algorytm k-średnich). Granice algorytmu centroidów. Analiza skupień w segmentacji obrazu. Analiza skupień w przetwarzaniu wstępnym. Analiza skupień w uczeniu półnadzorowanym. Algorytm DBSCAN. Inne algorytmy analizy skupień. Mieszanki gaussowskie. Wykrywanie anomalii za pomocą mieszanin gaussowskich. Wyznaczanie liczby skupień. Modele bayesowskie mieszanin gaussowskich. Inne algorytmy służące do wykrywania anomalii i nowości. 9. Ocena i poprawa jakości modeli. Reguła powrotu do średniej. Kryteria oceny modeli eksploracji danych. Ocena jakości modeli klasyfikacyjnych. Ocena jakości modeli regresyjnych. Ocena jakości modeli grupujących. Ocena jakości modeli rekomendujących. Ocena jakości modeli prognozujących. Poprawa jakości modeli. Znalezienie optymalnej liczby klastrów. 10. Wprowadzenie do sztucznych sieci neuronowych. Neurony biologiczne. Operacje logiczne przy użyciu neuronów. Perceptron. Perceptron wielowarstwowy i propagacja wsteczna. Regresyjne perceptrony wielowarstwowe. Klasyfikacyjne perceptrony wielowarstwowe. Głębokie uczenie. Dostrajanie hiperparametrów sieci neuronowej. 		
Literatura podstawowa:		
1. Géron A.: Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow. Wydanie III, Helion, 2023		

2. Szeliga M.: Data science i uczenie maszynowe, PWN, 2017
3. Morzy T.: Eksploracja danych – metody i algorytmy, PWN, 2024

Literatura dodatkowa:

1. Raschka S. i Mirjalili V.: Python. Uczenie maszynowe. Wydanie 2, Helion, 2019
2. Sztuczna inteligencja i uczenie maszynowe dla programistów. Praktyczny przewodnik po sztucznej inteligencji, Helion, 2021

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany jest technikami multimedialnymi. Ćwiczenia laboratoryjne – zajęcia praktyczne z wykorzystaniem wybranych narzędzi programowych. Na stronie internetowej prowadzącego zamieszczane są materiały z problemami i zadaniami laboratoryjnymi.

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W_01 do W_03 zostaną zweryfikowane podczas egzaminu pisemnego w trakcie trwania sesji egzaminacyjnej. Na egzaminie pytania będą dotyczyły poznanych zagadnień. Przykładowe pytania egzaminacyjne:

- Przedstaw metodę liniowej klasyfikacji SVM.
- Przedstaw metodę klasyfikacji wieloetykieterowej.
- Czym jest analiza skupień?

Przed egzaminem studenci będą mieli dostęp do przykładowych pytań.

Efekty U_01 do U_02 będą sprawdzane systematycznie na zajęciach laboratoryjnych w postaci wejściówek lub oceny wykonanej implementacji pod koniec zajęć.

Przykładowe zadania:

- Zaimplementuj model regresji liniowej dla podanych danych.
- Dokonaj klasyfikacji danych. Nadaj etykiety dla danych.
- Przeprowadź redukcję wielowymiarowości.

Forma i warunki zaliczenia:

Ocena z przedmiotu składa się z dwóch składowych:

- oceny z zajęć laboratoryjnych,
- oceny z egzaminu końcowego.

Na ocenę z zajęć laboratoryjnych składają się oceny cząstkowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać sumarycznie 40 pkt. Zaliczenie zajęć laboratoryjnych możliwe jest po uzyskaniu sumarycznie co najmniej 51% liczby punktów z tej formy zaliczenia.

W trakcie sesji odbędzie się egzamin końcowy. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Egzamin przewidziany jest w formie pisemnej. Można na nim uzyskać maksymalnie 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 51% liczby punktów z tej formy zaliczenia. Ocena końcowa z przedmiotu, w zależności od sumy uzyskanych punktów (maksymalnie 100 pkt) jest następująca (w nawiasach ocena wg skali ECTS):

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),

- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	21 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Przygotowanie się do egzaminu	25 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	25 godz.
Udział w konsultacjach z przedmiotu	5 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Przygotowanie się do egzaminu	35 godz.
Przygotowanie się do ćwiczeń laboratoryjnych	30 godz.
Udział w konsultacjach z przedmiotu	5 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Testowanie oprogramowania
Nazwa w języku angielskim:	Software testing	
Język wykładowy:	English	
Kierunek studiów, dla którego przedmiot jest oferowany:		Computer Science
Jednostka realizująca:	Faculty of Exact and Natural Sciences	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		Optional
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		First degree
Rok studiów:	Third	
Semestr:	Sixth	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		Dr Waldemar Bartyna
Imię i nazwisko prowadzących zajęcia:		Dr Waldemar Bartyna
Założenia i cele przedmiotu:		The course is intended to provide students with an understanding of crucial testing concepts, various kinds of testing, i.e., functional and non-functional testing, integration testing and system testing, as well as concepts of Test-Driven Development, Continuous Delivery Model and its impact on testing.
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Has ordered and theoretically founded knowledge in core testing concepts, testing types, methods, and objectives appropriate for the software development lifecycle model of a project	K_W06
W_02	Has ordered and theoretically founded knowledge about Test-Driven Development and Continuous Delivery Model and its impact on testing	K_W06
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Is able to plan and apply the appropriate level of testing within the context of a software development. Is able to design specific and measurable test cases to ensure coverage and traceability to requirements	K_U12, K_U19

U_02	Can use problem reporting techniques, metrics, and testing status reports to communicate testing results to colleagues, managers, and end users	K_U12, K_U19
U_03	Can apply principles and practices of test-driven development to improve testing quality and reduce delivery times. Is able to integrate testing processes within a continuous delivery model of software development	K_U12, K_U19
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	Is ready to critically assess their knowledge and is ready to recognize the importance of knowledge in solving cognitive and practical problems in the field of computer science	K_K01
Forma i typy zajęć:	Full-time studies: lectures (21 hours), laboratory classes (24 hours) Part-time studies: lectures (15 hours), laboratory classes (15 hours)	
Wymagania wstępne i dodatkowe:		
<p>The condition for participation in the classes is the prior completion of the following modules:</p> <ul style="list-style-type: none"> • Object-oriented programming • Introduction to WWW technologies • Software engineering • Programming platforms <p>or knowledge of the literature in these modules. Additionally, knowledge of the English language is required.</p>		
Treści modułu kształcenia:		
<ol style="list-style-type: none"> 1. Overview of the software testing process. Testing techniques. Functional and non-functional testing. 2. Test management and planning. Testing phases. Testing roles 3. Unit testing – approach and testing techniques, data requirements. Test engineer role, test cases and defect lifecycles 4. Integration testing – approach and testing techniques, data requirements, test analytics 5. System testing and system integration testing. Test process evaluation 6. User acceptance testing. Test plan example. Operations acceptance testing. 7. Regression testing. Software delivery concepts and configuration management. Metrics. Continuous Integration process 8. Agile testing. Test strategy implementation 9. Deployment pipeline. Automated acceptance test suite. Commit stage, commit stage test suite 10. Automated acceptance testing. Testing nonfunctional requirements 11. Test data management. Continuous Delivery management 		
Literatura podstawowa:		
<ol style="list-style-type: none"> 1. Watkins J., Mills S., Testing IT: An Off-the-Shelf Software Testing Process, 2nd edition, Cambridge University Press, 2011 2. Black, R., Agile Testing Foundations, BCS Learning & Development Ltd: Swindon UK, 2017 3. Whittaker J., Arbon J., Carollo J., How Google Tests Software, Addison-Wesley, 2012 4. Humble J., Farley D., Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Addison-Wesley, 2010 		

Literatura dodatkowa:

1. Roman, A., Testowanie i jakość oprogramowania. Modele, techniki, narzędzia, PWN, Warszawa, 2015
2. Smilgin, R., Zawód tester. Od decyzji do zdobycia doświadczenia, Wydawnictwo Naukowe PWN, Warszawa, 2016
3. Weinberg, G., Perfect Software and Other Illusions about Testing, Dorset House, NY, 2008

Planowane formy/działania/metody dydaktyczne:

Traditional lecture supported by multimedia techniques. Laboratory - practical classes with the use of Java, C #, Python, JavaScript and selected programming environments (e.g. IntelliJ Idea, Visual Studio, PyCharm, WebStorm or alternative) with appropriate plugins and extensions supporting the testing process and dedicated external tools (e.g. Selenium, Postman, Jenkins, JMeter)

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Outcomes W_01, W_02 and K_01 will be verified during oral colloquium. Before the colloquium, students will have access to the list of sample questions.

Sample questions:

- Compare test levels in terms of objectives, test basis, test items, common defects and failures, approaches, and responsibilities
- Discuss the characteristics, commonalities, and the differences between black-box, white-box, and experience-based testing techniques
- Explain how you can derive test cases from a use case
- Discuss benefits of code instruction coverage and decision coverage
- List and characterize the measures used for testing
- Discuss the benefits and risk factors involved with testing automation

The outcomes of U_01, U_02, U_03, and K_01 will be verified during laboratory classes, and team discussions.

Sample exercises and assignments:

- Design test cases based on the given requirements, using decision table method
- Plan the execution of a given set of test cases taking into account the priorities as well as technical and logical dependencies
- Compile a defect report with information on the defects found during testing

Forma i warunki zaliczenia:

The module ends with getting credit with a grade. To pass the module, it is required to pass the practical and theoretical parts. For the practical part of the module (laboratory classes, homework, assignments) one can score a maximum 60 points, while for the theoretical part maximum of 40 points (during oral colloquium). Every part will be credited after obtaining more than half of the maximum number of points (30 and 20 points, respectively). The final grade depends on the sum of points earned during both parts, as follows (in brackets the ECTS grade):

- 0 – 50 p.: 2 ndst (F),
- 51 – 60 p.: 3 dst (E),
- 61 – 70 p.: 3,5 dst+ (D),
- 71 – 80 p.: 4 db (C),
- 81 – 90 p.: 4,5 db+ (B),
- 91 – 100 p.: 5 bdb (A).

Improvement of the result in the practical part: at the latest within 3 weeks of the given classes/assignment. Improvement is not possible after the end of the semester.

Improvement of the colloquium result: during the make-up session.	
Bilans punktów ECTS:	
Studia stacjonarne	
Aktywność	Obciążenie studenta
Participation in lectures	21 h
Participation in laboratory classes	24 h
Self-preparation for laboratory classes, homework and assignments	30 h
Participation in the consultation hours	6 h
Individual studies on topics presented during lectures	9 h
Preparing for the colloquium and attending the colloquium	10 h
Total student workload	100 h
ECTS credits	4 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Participation in lectures	15 h
Participation in laboratory classes	15 h
Self-preparation for laboratory classes, homework and assignments	30 h
Participation in the consultation hours	3 h
Individual studies on topics presented during lectures	17 h
Preparing for the colloquium and attending the colloquium	20 h
Total student workload	100 h
ECTS credits	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Zintegrowane systemy informatyczne zarządzania
Nazwa w języku angielskim:		Integrated management information systems
Język wykładowy:		Polski
Kierunek studiów, dla którego przedmiot jest oferowany:		informatyka
Jednostka realizująca:		Wydział Nauk Ścisłych i Przyrodniczych
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:		Trzeci
Semestr:		szósty
Liczba punktów ECTS:		4
Imię i nazwisko koordynatora przedmiotu:		dr Ewa Szczepanik
Imię i nazwisko prowadzących zajęcia:		dr Marek Piłski
Założenia i cele przedmiotu:		Celem kursu jest przedstawienie zagadnień związanych z zintegrowanymi systemami zarządzania, min. to: struktura, architektura, standardy oraz nabycie praktycznych umiejętności związanych z wykorzystaniem
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie rolę systemu informatycznego w przedsiębiorstwie.	K_W07, K_W09
W_02	Zna i rozumie zagadnienia z zakresu typologii, standardów, struktury i architektury typowych systemów informatycznych zarządzania.	K_W09
W_03	Zna i rozumie zagadnienia z zakresu złożoności realizacyjnej SI oraz zna sposoby (scenariusze) realizacji zintegrowanych systemów informatycznych zarządzania.	K_W07, K_W12
W_04	Zna i rozumie zagadnienia z zakresu wdrażania do przedsiębiorstwa zintegrowanych systemów informatycznych zarządzania, zarządzania przedsięwzięciami gospodarczymi i informatycznymi (opracowanie harmonogramu, śledzenie realizacji zadań, itp.).	K_W12, K_W13
W_05	Zna i rozumie zagadnienia z zakresu zastosowania wybranych branżowych systemów informatycznych zarządzania.	K_W07
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi na podstawie ogólnodostępnej informacji (literatura, informacja w Internecie) formułować wnioski dotyczące najnowszych rozwiązań systemów komputerowych.	K_U01
U_02	Potrafi zbudować harmonogram prac uwzględniając jego zadania, czas, budżet, zasoby i zarządzać realizacją przedsięwzięcia.	K_U13, K_U05

U_03	Potrafi wskazać rodzaj systemu informatycznego do określonych zastosowań w praktyce zarządzania przedsiębiorstwem.	K_U23
U_04	Potrafi wykorzystać wybrany SI do realizacji prostych zadań biznesowych.	K_U23
U_05	Potrafi przygotować prezentację dotyczącą charakterystyki i właściwości funkcjonalno-użytkowych wybranego SIZ	K_U01, K_U02
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	jest gotów do wyrażenia oceny zasadności zastosowania SIZ dla danego sektora lub branży	K_K01
K_02	jest gotów do przekazywania informacji i opinii dotyczących funkcjonalności i zastosowania SIZ w sposób zrozumiały	K_K02
K_03	jest gotów do wykorzystania w sposób przedsiębiorczy SIZ	K_K03
Forma i typy zajęć:	studia stacjonarne: wykłady (21 godz.), ćwiczenia laboratoryjne (24 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
Wymagania wstępne i dodatkowe:		
1. Umiejętność wykorzystania podstaw inżynierii tworzenia systemów informatycznych oraz znajomość ogólnych problemów społecznych i zawodowych informatyki		
Treści modułu kształcenia		
<ol style="list-style-type: none"> Zarządzanie przedsięwzięciami. Pojęcia: projekt, zadanie, ścieżka krytyczna, zapas czasu, punkty kontrolne, zadania sumaryczne i składowe, zadania o stałym czasie trwania, zadania zależne od nakładu pracy, zadania cykliczne, zasoby, alokacja zasobów, bilansowanie zasobów. Harmonogram. Budowa harmonogramu, lista zadań, typy zależności pomiędzy zadaniami, typy zasobów, przydział zasobów do zadań, fazy projektu (cel i zakres projektu, planowanie, korekta, śledzenie), warianty harmonogramu, kalendarze projektu, wykres Gantta, diagram sieciowy PERT, trójkąt projektu. Rola zasobów informacyjnych w przedsiębiorstwie. Systemy organizacji. Zasoby informacyjne w przedsiębiorstwie. Procesy informacyjno-decyzyjne, pętla OODA. Informacja a procesy podejmowania decyzji. Model zastosowań technologii informatycznej w organizacji. Zastosowanie pojedynczych aplikacji IT. Wewnętrzna integracja. Reorganizacja procesów biznesowych. Reorganizacja sieci gospodarczej. Zmiana zakresu działalności gospodarczej. Typologia i ewolucja informatycznych systemów zarządzania i przegląd wybranych informatycznych systemów zarządzania. Systemy informatyczne zarządzania. Typologia SIZ. Standardy SIZ. Struktury i architektura SIZ. Struktury SIZ. Architektury SIZ Podejście procesowe w zarządzaniu i jego wpływ na kierunki rozwoju SIZ. Istota podejścia procesowego w zarządzaniu. Związek podejścia procesowego z koncepcjami współczesnych SIZ. Złożoność realizacyjna SIZ i scenariusze realizacji SIZ. Czynniki występujących podczas realizacji systemu. Składniki systemu jako produktu końcowego. Tworzenie ZSI od podstaw przez służby informatyczne przedsiębiorstwa. Tworzenie ZSI od podstaw przez zewnętrzne firmy informatyczne. Wybór, zakup i wdrożenie wyrobu gotowego. Procedura wyboru gotowego ZSI. Kryteria wyboru ZSI. Opis procedury. Fazy procedury wyboru gotowego ZSI. Organizacja prac wdrożeniowych ZSI. Model i etapy wdrażania ZSI. Realizacja ZSI. Działania integrujące (integrator wdrożeniowy). Przykładowy model integracji działań Współczesne zintegrowane systemy informatyczne zarządzania. Klasyfikacja i podział współczesnych zintegrowanych SIZ. Przegląd dostępnych na rynku polskim SIZ. Tendencje rozwojowe współczesnych SIZ Zapoznanie z działaniem wybranych systemów informatycznych zarządzania. 		

13. **Przegląd wybranych informatycznych systemów zarządzania.** Prezentacja wybranej aplikacji dla różnych branż i sektorów przez studentów, np. wspomaganie zarządzania instytucją oświatową, administracji państwowej, administracji gospodarczej, służby zdrowia itp.

Literatura podstawowa:

1. Carl Chatfield, Timothy Johnson, (tł. Marcin Kowalczyk): *Microsoft Office Project 2007 : krok po kroku*, wersja polska, RM, Warszawa, 2008.
2. Barczak A., Florek J., Sydoruk T.: *Projektowanie zintegrowanych systemów informatycznych zarządzania*. Wydawnictwo Akademii Podlaskiej, Siedlce 2006
3. Jerzy Kisielnicki: *Systemy informatyczne zarządzania*. Placet, Warszawa, 2013

Literatura dodatkowa:

4. Jabłoński W., Bartkiewicz W.: *Systemy informatyczne zarządzania : klasyfikacja i charakterystyka systemów*, Bydgoszcz, 2006
1. Mirosława Kopertowska, Witold Sikorski: *MS Project : kurs podstawowy*, PWN, Warszawa, 2007

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, laboratoria wspomagane technikami komputerowymi i przykładowymi systemami informatycznymi zarządzania. Zamieszczanie na stronach internetowych problemów i zadań laboratoryjnych.

Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:

Efekty W_01 - W_05 sprawdzane będą na semestralnym kolokwium – forma pisemna (test wiedzy).

Efekt U_02 - U_04 będą systematycznie sprawdzane na zajęciach laboratoryjnych w trakcie wykonywania zadań i sprawdzianów. Przykładowe zadania:

1. Omów pojęcia: „zadanie krytyczne” i „ścieżka krytyczna” w kontekście planowania przedsięwzięć gospodarczych
2. Zdefiniuj nowy zasób typu „praca” i przydziel go do wybranego zadania projektowego w wymiarze 50% dostępnych jednostek w narzędziu MS Project
3. Przedstaw dostępne typy relacji pomiędzy zadaniami projektowymi w harmonogramach Gantta i podaj stosowne przykłady zadań pasujące do omawianych typów relacji
4. Podaj przykład zadania cyklicznego i dodaj je w MS Project
5. Wykonaj bilans otwarcia dnia z 100 sztukami wskazanego towaru w narzędziu Optima firmy Comarch
6. Sprzedaj dany towar nowemu kontrahentowi wystawiając fakturę płatną w dwóch równych ratach w narzędziu Optima firmy Comarch
7. Zatrudnij nowego pracownika w wymiarze ½ etatu, ustal mu miesięczne wynagrodzenie oraz premie w narzędziu Optima firmy Comarch

Zadania na każde następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami.

Efekty U_01, U_05 oraz K_01 - K_03 weryfikowane, w oparciu o posiadaną wiedzę i umiejętności w czasie zajęć laboratoryjnych, sprawdzianów oraz podczas zaprezentowanej przez studenta prezentacji.

Przykładowe zadania

1. Przedstaw funkcjonalność wybranego narzędzia z sektora służby zdrowia, pokaż realizację dwóch wybranych/kluczowych funkcji aplikacji.
2. Przedstaw funkcjonalność wybranego narzędzia z branży IT, pokaż realizację dwóch wybranych/kluczowych funkcji aplikacji.

Forma i warunki zaliczenia:

Moduł kończy się zaliczeniem na ocenę. Ocena końcowa jest wystawiana na podstawie zdobytych punktów z zajęć laboratoryjnych, wygłoszonej prezentacji i kolokwium pisemnego przeprowadzonego na ostatnim wykładzie. Na zaliczenie laboratorium składają się oceny cząstkowe wg schematu:

- Regularne zajęcia z MS Project – 10 pkt.,

- Sprawdzian z MS Project – 20 pkt.,
- Regularne zajęcia z Optima – 10 pkt.,
- Sprawdzian z Optima – 20 pkt.,
- Wykonanie i wygłoszenie prezentacji – 20 pkt.,
- Kolokwium – 20 pkt.

Punkty sumuje się. Razem za wszystkie oceniane formy zajęć można uzyskać 100 pkt.

Zaliczenie odpowiednio:

- 0 – 50 pkt: niedostateczna (F),
- 51 – 60 pkt: dostateczna (E),
- 61 – 70 pkt: dostateczna plus (D),
- 71 – 80 pkt: dobra (C),
- 81 – 90 pkt: dobra plus (B),
- 91 – 100 pkt: bardzo dobra (A).

Poprawy:

Każde oceniane spotkanie laboratoryjne można poprawić jeden raz. Jednorazowa poprawa kolokwium w trakcie zajęć w semestrze.

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	21 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	30 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do kolokwium pisemnego i obecność na kolokwium	20 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach laboratoryjnych	15 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	42 godz.
Udział w konsultacjach godz. z przedmiotu	3 godz.

Przygotowanie się do kolokwium pisemnego i obecność na kolokwium	25 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Rozpoznawanie obrazów
Nazwa w języku angielskim:		Image recognition
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka
Jednostka realizująca:	Instytut Informatyki	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		dr Andrzej Salamończyk
Imię i nazwisko prowadzących zajęcia:		dr Andrzej Salamończyk
Założenia i cele przedmiotu:		Celem kursu jest zapoznanie studentów z podstawowymi zagadnieniami związanymi z algorytmami i metodami rozpoznawania obrazów oraz zdobycie praktycznych umiejętności w implementacji tych algorytmów i metod oraz korzystania z odpowiednich bibliotek.
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Zna i rozumie zagadnienia związane z metodami i algorytmami rozpoznawania obrazów opartymi o metody probabilistyczne i metody wzorców	K_W08
W_02	Zna i rozumie zagadnienia związane z metodami i algorytmami rozpoznawania obrazów opartymi o metody aproksymacyjne	K_W08
W_03	Zna i rozumie zagadnienia związane z metodami i algorytmami rozpoznawania obrazów opartymi o metody sztucznej inteligencji	K_W08
W_04	Zna i rozumie zagadnienia związane z metodami i algorytmami rozpoznawania obrazów opartymi o metody ciągowe	K_W08

Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi pozyskiwać informacje dotyczące z literatury, baz danych i innych źródeł, potrafi integrować pozyskane informacje z innymi zadaniami	K_U01
U_02	Potrafi samodzielnie planować i realizować uczenie się przez całe życie, m.in. w celu podnoszenia kompetencji zawodowych	K_U06
U_03	Potrafi wykorzystać właściwie dobrane środowiska programistyczne i narzędzia komputerowe do projektowania i weryfikacji systemów informatycznych związanych z przetwarzaniem obrazów	K_U11
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	Jest gotów do krytycznej oceny posiadanej wiedzy z zakresu rozpoznawania obrazów	K_K01
K_02	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu problemów praktycznych i poznawczych z zakresu rozpoznawania obrazów	K_K01
Forma i typy zajęć:	studia stacjonarne: wykłady (21 godz.), ćwiczenia laboratoryjne (24 godz.) studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
Wymagania wstępne i dodatkowe:		
1. Znajomość analizy matematycznej, algebry, podstaw programowania, technologii WWW oraz zagadnień sztucznej inteligencji.		
Treści modułu kształcenia:		

1. Wstęp. Cyfrowa reprezentacja obrazu. Etapy zadania rozpoznawania obrazów: preprocessing, segmentacja, ekstrakcja cech, klasyfikacja, interpretacja. Kryteria oceny i zastosowania systemów rozpoznających obrazy.
2. Formaty plików graficznych (obrazów). Modele barwowe. Podstawy języka Python.
3. Przydatne biblioteki w rozpoznawaniu obrazów: Python: NumPy OpenCV, Scikit-Image, Tensorflow.
4. Preprocessing. Metody aproksymacyjne. Przetwarzanie obrazu z przykładami w języku Python.
5. Segmentacja. Metody punktowe: progowanie i klasteryzacja, metody krawędziowe, metody obszarowe: rozrost/podział/łączenie obszarów, metody hybrydowe i wododziałowe.
6. Detekcja i ekstrakcja cech. Deskryptory kształtu, HOG, SIFT, BRIEF, FAST. Kody łańcuchowe.
7. Klasyfikacja. Segmentacja i klasyfikacja tekstur. Deskryptory tekstury. Metody k-NN.
8. Rozpoznawanie wzorców. Metoda PCA. Maszyna wektorów nośnych SVM. Metody statystyczne, klasyfikacja bayesowska.
9. Sieci neuronowe i głębokie sieci neuronowe. Wykorzystanie sieci neuronowych i konwolucyjnych sieci neuronowych w rozpoznawaniu obrazów. Wykorzystanie metod i narzędzi sztucznej inteligencji.
10. Analiza sekwencji obrazów. Detekcja ruchu i śledzenie obiektów.
11. Tendencje w rozpoznawaniu obrazów. Kolokwium.

Laboratorium (zajęcia rozpoczynają się w czwartym tygodniu, po trzecim wykładzie)

1. Preprocessing, przetwarzanie wstępne obrazów w języku Python (scikit-image)
2. Preprocessing, przetwarzanie wstępne obrazów w języku Python (numpy)
3. Preprocessing, przetwarzanie wstępne obrazów w języku Python (opencv)

4. Przetwarzanie obrazów przechwyconych z kamery i plików video w języku Python
5. Segmentacja, rozpoznawanie tekstu
6. Detekcja cech, deskryptory, szukanie odpowiadających sobie punktów na obrazach
7. Detekcja cech, Wykrywanie obiektów, wykrywanie linii (np. pasa ruchu na drodze)
8. Detekcja cech, Wykrywanie twarzy na obrazach
9. Detekcja cech, Śledzenie obiektów
10. Klasyfikacja, klasyfikacja z wykorzystaniem sztucznych sieci neuronowych
11. Klasyfikacja, wykorzystanie konwolucyjnych sieci neuronowych do rozpoznawania obrazów
12. Przegląd innych ciekawych bibliotek i narzędzi w rozpoznawaniu obrazów.

Literatura podstawowa:

1. W. Kasprzak. Rozpoznawanie obrazów i sygnałów mowy. Oficyna Wydawnicza Politechniki Warszawskiej, 2009.
2. K. Stąpor, Automatyczna klasyfikacja obrazów, EXIT, 2005
3. R. Tadeusiewicz, M. Flasiński, Rozpoznawanie obrazów, WNT, 1991.

Literatura dodatkowa:

1. A. Kaehler, G. Bradski, OpenCV 3: Komputerowe rozpoznawanie obrazów w C++ przy użyciu biblioteki OpenCV, Helion 2017.
2. J. Koronacki, J. Ćwik, Statystyczne systemy uczące się, WNT, Warszawa, 2005,
3. M. Domański, Obraz cyfrowy, WKŁ 2009.

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi. Zamieszczanie na stronach internetowych zadań i materiałów do laboratoriów.

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W_01 – W_04 będą sprawdzane na egzaminie. Zadania będą dotyczyły wybranych problemów przetwarzania obrazów, przykładowe zadania:

- Omów metodę wzorców, podaj przykłady jej wykorzystania.
- Omów metodę kodów łańcuchowych Freemana, Jakie algorytmy są wykorzystywane w tej metodzie.

Efekty U_01 - U_02 będą systematycznie sprawdzane na zajęciach. Zadania na następne laboratorium muszą być dostępne co najmniej tydzień przed zajęciami. Student, na podstawie podanej literatury, musi się do nich samodzielnie lub korzystając z konsultacji przygotować.

Efekt U_03 będzie sprawdzany systematycznie na zajęciach laboratoryjnych, przykładowe zadanie:

- Dany jest obraz, stosując algorytm k-NN znajdź wybrane obszary.

Efekty K_K01, K_K02 będą sprawdzane na egzaminie. Przykładowe zadanie:

- W jaki sposób kody łańcuchowe Freemana można wykorzystać do rozpoznawania pisma.

Forma i warunki zaliczenia:

Moduł kończy się egzaminem. Ocena końcowa jest wystawiana na podstawie zajęć laboratoryjnych i egzaminu pisemnego. Na zaliczenie laboratorium składają się oceny cząstkowe uzyskane na laboratoriach (oprócz pierwszego i ostatniego).

- Oceniane laboratoria (10 zajęć po 5 pkt.)

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania co najmniej połowy punktów tj. co najmniej 26pkt Ostateczny wynik punktowy modułu oblicza się wg wzoru: $P=L+E$,

gdzie P-końcowy wynik punktowy (maksymalnie 100pkt.) , L-punkty uzyskane z części laboratoryjnej (maksymalnie 50), E-punktowy wynik egzaminu (maksymalnie 50)

Ocena z zajęć zależy od końcowego wyniku punktowego i wyznacza się w następujący sposób.

- 0-50 punktów – 2
- 51-60 punktów – 3
- 61-70 punktów - 3,5
- 71-80 punktów – 4
- 81-90 punktów – 4,5
- 91-100 punktów – 100

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	21 godzin
Udział w ćwiczeniach laboratoryjnych	24 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	30 godzin
Udział w konsultacjach godz. z przedmiotu	2 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	23 godzin
Sumaryczne obciążenie pracą studenta	100 godzin
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godzin
Udział w ćwiczeniach laboratoryjnych	15 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	35 godzin

Udział w konsultacjach godz. z przedmiotu	2 godziny
Przygotowanie się do egzaminu i obecność na egzaminie	33 godziny

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:		Programowanie zaawansowane
Nazwa w języku angielskim:		Advanced programming
Język wykładowy:	polski	
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia
Rok studiów:	trzeci	
Semestr:	szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:		Dr Waldemar Bartyna
Imię i nazwisko prowadzących zajęcia:		Dr Waldemar Bartyna
Założenia i cele przedmiotu:		Celem przedmiotu jest zapoznanie studentów z projektowaniem i implementacją aplikacji wykorzystujących zaawansowane konstrukcje i mechanizmy programistyczne dostępne w języku C# na platformie .NET. Studenci będą dokładnie rozumieć znaczenie i sposób działania takich konstrukcji i mechanizmów jak obsługa wyjątków, polimorfizm, budowanie bazy wspólnego kodu, delegaty, refleksja, serializacja, LINQ i innych, oraz będą potrafili prawidłowo je wykorzystywać w procesie tworzenia aplikacji różnego typu.
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	zna i rozumie zagadnienia dotyczące zasad programowania obiektowego, między innymi, hermetyzacji, dziedziczenia i polimorfizmu	K_W06
W_02	zna i rozumie zaawansowane mechanizmy programistyczne, w tym: strukturalną obsługę wyjątków, interfejsy; generyczność, refleksję i serializację	K_W06
W_03	zna i rozumie elementy składające się na platformę .NET, język programowania C# i jego zaawansowane konstrukcje w tym: właściwości automatyczne, metody rozszerzeniowe, niejawnie typowanie	K_W06

W_04	zna i rozumie sposoby komunikowania się między obiektami poprzez interfejsy i delegaty	K_W06
W_05	zna i rozumie różne sposoby utrwalania danych poprzez bazy danych, pliki XML, oraz sposoby korzystania z tych źródeł danych (ADO.NET i LINQ to XML)	K_W06
W_06	zna i rozumie zasady korzystania i mechanizmy związane z językiem zapytań zintegrowanym z językiem programowania (LINQ)	K_W06
W_07	zna i rozumie zagadnienia związane z tworzeniem aplikacji sieciowych bazujących na paradygmacie MVC na platformie .NET	K_W06, K_W011
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	potrafi korzystać w efektywny sposób ze środowiska programistycznego Visual Studio i implementować w nim cztery podstawowe rodzaje projektów (biblioteki klas, aplikacje konsolowe, desktopowe i sieciowe)	K_U10, K_U11
U_02	potrafi projektować i implementować aplikacje zgodnie z zasadami programowania obiektowego	K_U10, K_U11
U_03	potrafi efektywnie korzystać z zaawansowanych mechanizmów i konstrukcji (obsługi wyjątków, interfejsów, generyczności, delegatów, LINQ, refleksji i serializacji)	K_U10, K_U11
U_04	potrafi ocenić przydatność rutynowych metod i narzędzi służących do rozwiązywania prostych zadań inżynierskich, typowych dla projektowania i programowania aplikacji oraz wybierać i stosować właściwe metody i narzędzia związane z zaawansowanymi technikami programistycznymi	K_U10, K_U11
U_05	potrafi zaprojektować, zaimplementować oraz przetestować aplikacje sieciowe z dostępem do baz danych i wykorzystaniem wzorca MVC	K_U19
U_06	potrafi opracować i zaimplementować algorytm w języku C# wykorzystując odpowiednie narzędzia informatyczne	K_U22
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	jest gotów do podejmowania decyzji i przeprowadzania analizy efektów tych decyzji w ramach rozwiązywania zadań programistycznych	K_K01
K_02	jest gotów do uznania znaczenia wiedzy w rozwiązywaniu problemów programistycznych oraz do konstruktywnej krytyki powstałych rozwiązań	K_K01
Forma i typy zajęć:	studia stacjonarne: wykłady (30 godzin), ćwiczenia laboratoryjne (45 godzin) studia niestacjonarne: wykłady (18 godzin), ćwiczenia laboratoryjne (24 godzin)	
Wymagania wstępne i dodatkowe:		

1. Znajomość podstaw programowania obiektowego.
2. Znajomość podstaw języka C#.
3. Znajomość wzorca MVC.

Treści modułu kształcenia:

1. **Platforma .NET.** Omówienie wspólnego silnika uruchomieniowego, wspólnego systemu typów, wspólnej specyfikacji języka, wielojęzykowości i niezależności od platformy.
2. **Obiektowe programowanie w języku C#.** Znaczenie filarów programowania obiektowego (obiekty, klasy, dziedziczenie, hermetyzacja, abstrakcja i polimorfizm) z naciskiem na elementy składni charakterystycznej dla języka C#.
3. **Strukturalna obsługa wyjątków.** Znaczenie mechanizmu wyjątków. Sposoby implementacji strukturalnej obsługi wyjątków w języku C#.
4. **Interfejsy.** Znaczenie interfejsu jako warstwy abstrakcji, definiowanie własnych interfejsów, implementowanie i korzystanie z bazy wspólnego kodu, różnice między interfejsami a klasami abstrakcyjnymi.
5. **Generyczność.** Znaczenie typów i metod generycznych, sposoby definiowania i wykorzystywania własnych szablonów, definiowanie ograniczeń dla parametrów generycznych.
6. **Delegaty.** Wykorzystanie delegatów do komunikacji między obiektami, silnie typowane sposoby przekazywania wskaźników do metod, metody anonimowe, wyrażenia lambda.
7. **Zaawansowane konstrukcje języka C#.** Niejawne typowanie zmiennych lokalnych, właściwości automatyczne, metody rozszerzeniowe, metody częściowe, inicjalizator obiektów i typy anonimowe.
8. **Technologia LINQ.** Język zapytań zintegrowany z językiem programowania jako ujednolicony sposób wyszukiwania danych niezależnie od ich źródła. Operatory zapytań i ich wewnętrzna reprezentacja.
9. **ADO.NET – warstwa połączeniowa.** Architektura dostawców danych, wspólne interfejsy i klasy bazowe, korzystanie z obiektów połączenia, polecenia i odczytywania danych.
10. **ADO.NET – warstwa bezpołączeniowa.** Rola warstwy bezpołączeniowej, typy obiektów wykorzystywane do lokalnego reprezentowania bazy danych, sposoby kontroli zmian w lokalnej wersji.
11. **LINQ to ADO.NET.** Wykorzystanie technologii LINQ do wyszukiwania danych w bazie danych lub jej lokalnej reprezentacji. Automatyczne generowanie klas reprezentujących tabele i klas wspomagających.
12. **Refleksja.** Znaczenie metadanych, klasy i metody związane z mechanizmem refleksji; dynamiczne ładowanie pakietów, późne wiązanie, definiowanie własnych atrybutów.
13. **Serializacja.** Znaczenie serializacji, dostępne formaty (binarny, SOAP, XML), konfigurowanie procesu serializacji.
14. **LINQ to XML.** Tworzenie dokumentów XML, ich edytowanie; parsowanie i wyszukiwanie danych w dokumentach XML z wykorzystaniem nowoczesnego API – LINQ to XML.
15. **LINQ to Entity.** Entity Framework, podejście Code First, obiekty encji i związane z nimi atrybuty, przykład wykorzystania w ramach aplikacji sieciowej ASP.NET Core MVC.

Literatura podstawowa:

1. Mark Michaelis, C# 8.0. Kompletny przewodnik dla praktyków. Wydanie VII, Helion 2021
2. Adam Freeman, ASP.NET Core 3. Zaawansowane programowanie. Wydanie VIII, Helion 2021

Literatura dodatkowa:

1. Joseph Albahari, C# 9.0 w pigułce, Helion 2021

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia laboratoryjne wspomagane technikami multimedialnymi. Zamieszczanie na stronach internetowych problemów i zadań ćwiczeniowych.

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty **W_01 – W_07** będą sprawdzane podczas ćwiczeń i na egzaminie ustnym. Student będzie odpowiadał na pytania dotyczące zaawansowanych pojęć i konstrukcji w programowaniu obiektowym w języku C#:

- Czym jest interfejs? Podaj przykład jego użycia,
- Czym są delegaty i do czego ich używamy?
- Na czym polega mechanizm refleksji? Podaj przykłady jego zastosowania.

Studenci zapoznają się z listą wszystkich pytań podczas pierwszego wykładu.

Efekt **U_01 - U_06** będą systematycznie sprawdzane na zajęciach. Wszystkie zadania laboratoryjne udostępnione są studentom na początku kursu. Studenci mogą przygotować się do laboratoriów znając treści zadań i na podstawie silnie powiązanego z laboratoriami wykładu. Przykładowe zadania:

- Dana jest lista osób o strukturze z poprzedniego zadania zapisana w liście genrycznej. Napisz odpowiednie zapytania LINQ umożliwiające wyszukanie podanych niżej informacji.
- Zaprojektuj i zaimplementuj interfejsy umożliwiające realizację opisanej poniżej funkcjonalności na poziomie bazy wspólnego kodu.
- Zaprojektuj aplikację, którą można będzie rozszerzać za pomocą wtyczek. Opracuj odpowiedni interfejs i mechanizm dynamicznego ładowania pakietów z wtyczkami.

Efekty **K_01 i K_02** będą weryfikowane w oparciu o odpowiedzi na pytania zadawane w czasie ćwiczeń laboratoryjnych oraz podczas zaliczania zadania indywidualnego.

Forma i warunki zaliczenia:

Przedmiot kończy się egzaminem ustnym. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie ćwiczeń laboratoryjnych. Na zaliczenie ćwiczeń laboratoryjnych składają się oceny częściowe uzyskane na ćwiczeniach z nauczycielem akademickim oraz ocena z samodzielnie wykonanego zadania indywidualnego według schematu:

- Ćwiczenia – 140 punktów (10 punktów za każde z ćwiczeń oprócz ostatniego ćwiczenia przeznaczonego na obrony zadań indywidualnych),
- Zadanie indywidualne – 60 punktów.

Ćwiczenia laboratoryjne będą zaliczone wyłącznie w wypadku uzyskania powyżej połowy punktów z każdych punktowanych ćwiczeń i powyżej połowy punktów z zadania indywidualnego. Na tej formie zajęć student może uzyskać maksymalnie 200 punktów.

Podczas egzaminu ustnego można uzyskać maksymalnie 100 punktów. Egzamin będzie zaliczony w przypadku uzyskania powyżej połowy punktów za każde pytanie. Ocena końcowa z przedmiotu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 300 punktów) jest ustalana na podstawie poniższych zakresów (w nawiasach ocena wg skali ECTS):

- 0 – 150 punktów: niedostateczna (F),
- 151 – 180 punktów: dostateczna (E),
- 181 – 210 punktów: dostateczna plus (D),

- 211 – 240 punktów: dobra (C),
- 241 – 270 punktów: dobra plus (B),
- 271 – 300 punktów: bardzo dobra (A).

Bilans punktów ECTS:

Studia stacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	30 godzin
Udział w ćwiczeniach laboratoryjnych	45 godziny
Udział w konsultacjach z przedmiotu	2 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godzin
Samodzielne realizacja zadania indywidualnego	10 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	3 godziny
Sumaryczne obciążenie pracą studenta	100 godzin
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	18 godzin
Udział w ćwiczeniach laboratoryjnych	24 godziny
Udział w konsultacjach godz. z przedmiotu	2 godziny
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	30 godzin
Samodzielne realizacja zadania indywidualnego	20 godzin
Przygotowanie się do egzaminu i obecność na egzaminie	6 godzin
Sumaryczne obciążenie pracą studenta	100 godzin
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia		
Nazwa przedmiotu/modułu kształcenia:	Aplikacje internetowe i rozproszone	
Nazwa w języku angielskim:	web and distributed applications	
Język wykładowy:	Polski	
Kierunek studiów, dla którego przedmiot jest oferowany:	informatyka	
Jednostka realizująca:	Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):	fakultatywny	
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):	pierwszego stopnia	
Rok studiów:	Trzeci	
Semestr:	Szósty	
Liczba punktów ECTS:	4	
Imię i nazwisko koordynatora przedmiotu:	Dr Dariusz Mikułowski	
Imię i nazwisko prowadzących ćwiczenia:	Dr Dariusz Mikułowski	
Założenia i cele przedmiotu:	Celem kursu jest zapoznanie studentów z projektowaniem i implementacją aplikacji opartych o zaawansowane techniki i technologie rozproszone i internetowe	
Symbol efektu	Efekt uczenia się: WIEDZA	Symbol efektu kierunkowego
W_01	Student zna i rozumie funkcjonowanie języków opisu dokumentu HTML i XML i ich zastosowania.	K_W06, K_W11
W_02	Zna i rozumie zasady działania języków opisu struktury dokumentu DTD oraz XMLSchema	K_W06, K_W11
W_03	Zna i rozumie protokoły komunikacyjne używane w aplikacjach internetowych; zna zasady działania aplikacji rozproszonych takich jak poczta elektroniczna, wyszukiwarki internetowe, pająki i roboty.	K_W06, K_W11
W_04	Zna i rozumie działanie języków zapytań do dokumentów XML: XPath oraz XQuery.	K_W06, K_W11
W_05	Zna i rozumie koncepcje architektur otwartych systemów internetowych: Webservice, WS-REST, technologii semantycznych oraz P2P. oraz ich zastosowania.	K_W06, K_W11
Symbol efektu	Efekt uczenia się: UMIEJĘTNOŚCI	Symbol efektu kierunkowego
U_01	Potrafi pisać programy posługujące się pocztą internetową jako medium komunikacji	K_U02, K_U10, K_U11, K_U19

U_02	Potrafi wykorzystać w komunikacji między aplikacjami języki i formaty takie jak JSON, XML oraz HTML i specyfikować język komunikacji w DTD i XMLSchema.	K_U02, K_U10, K_U11, K_U19
U_03	Potrafi napisać aplikację wyszukiwarki Internetowej oraz prostego pająka Internetowego,.	K_U02, K_U10, K_U11, K_U19
U_04	Posługuje się narzędziami do wyszukiwania informacji w dokumentach XML/HTML oraz do ich przetwarzania z wykorzystaniem parserów XML DOM i JAXP,	K_U02, K_U10, K_U11
Symbol efektu	Efekt uczenia się: KOMPETENCJE SPOŁECZNE	Symbol efektu kierunkowego
K_01	Pojmuje naukę jako postępowy rozwój teoretycznych uogólnień wynikający z obserwacji i doświadczeń i prowadzący do nowych obserwacji i nowych doświadczeń.	K_K01
K_02	rozumie jedność teorii i praktyki oraz konieczność pogłębiania swej wiedzy w sposób ukierunkowany, by stawiać czoła konkretnym wyzwaniom projektowym	K_K01, K_K02, K_K03
K_03	Potrafi oceniać wartość informacji, szczególnie dostępnej w Internecie, nie tylko na poziomie statystycznym, ale i syntaktycznym, semantycznym, pragmatycznym.	K_K01
Forma i typy zajęć:	studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (45 godz.) studia niestacjonarne: wykłady (18 godz.), ćwiczenia laboratoryjne (24 godz.)	
Wymagania wstępne i dodatkowe:		
Umiejętność programowania		
Treści modułu kształcenia:		
<ol style="list-style-type: none"> Infrastruktura i architektury dla aplikacji internetowych: model wielowarstwowy ISO-OSI sieci komputerowych, wybrane protokoły internetowe, poczta elektroniczna WWW jako przykład architektury klient-serwer: HTTP, wybrane serwery WWW Język XML (eXtensible Markup Language) jako język wymiany informacji między aplikacjami: XML jako szczególny przypadek SGML, budowa dokumentu, przestrzenie nazw, zastosowania XML Język XML (eXtensible Markup Language) - interfejsy programistyczne: model DOM, model SAX, realizacje w Javie Metainformacje dla XML - słaba kontrola składni: DTD, XSchema Język ścieżek XPATH oraz język transformacji dokumentów XML (XSLT): XPath, Koncepcja transformacji XSL (XML Stylesheet Language);, Budowa wzorców XSLT; , Środki programistyczne dostępne w XSLT. repozytoria XML i języki zapytań: repozytoria XML, XQUERY, inne języki zapytań dla XML Problemy technologiczne wyszukiwarek internetowych: Budowa wyszukiwarki, Technologie pająków, Technologie Indeksatorów, Metody wyszukiwania Java beans: struktura EJB, zastosowania Otwarte architektury aplikacji: SOAP i Web-Serwisy: Oprogramowanie pośredniczące, WSDL, ebXML Otwarte architektury aplikacji: REST: Wady koncepcji Webservisów, Naturalność architektury REST Otwarte architektury aplikacji: Modele "koleżeńskie": architektura P2P, wymagania technologiczne, JXTA, zastosowania w biznesie Technologie semantyczne i pojęcie sieci semantycznej (Semantic Web), pojęcie ontologii, języki do wyrażania ontologii (RDF, OWL), zastosowania technologii semantycznych do budowania aplikacji Internetowych, Programowanie w chmurze: środki programistyczne, Koncepcja mikroserwisów, kontrola zasobów, aspekty ekonomiczne. Elementy projektowania uniwersalnego w aplikacjach Internetowych 		

Literatura podstawowa:	
<ol style="list-style-type: none"> 1. Mikrousługi w javie poradnik eksperta. Sourabh Sharma. Helion Gliwice 2018 2. XML w programowaniu aplikacji internetowych / Dariusz Mikułowski ; Akademia Podlaska. Instytut Informatyki. 	
Literatura dodatkowa:	
<ol style="list-style-type: none"> 1. B. Eckel: Thinking in Java Prentice-Hall, December 2002 2. M.A.Kłopotek: Inteligentne wyszukiwarki internetowe. Akademicka Oficyna Wydawnicza Exit, Warszawa 2001, 332 strony, ISBN 83-87674-31-1 3. XML : księga eksperta / Elliotte Rusty Harold ; [tł. Tomasz Żmijewski] Grupa Wydawnicza Helion. 2000 4. XSLT dla każdego Autor: Michiel van Otegem Tłumaczenie: Tomasz Żmijewski ISBN: 83-7197-785-9 Tytuł oryginału: TY XSLT in 21 Days Format: B5, stron: 576 Data wydania: 05/2003 	
Planowane formy/działania/metody dydaktyczne:	
Wykład tradycyjny wspomagany technikami multimedialnymi, ćwiczenia rachunkowe wspomagane technikami multimedialnymi. Zamieszczanie na stronach internetowych problemów i zadań ćwiczeniowych.	
Sposoby weryfikacji efektów kształcenia osiągniętych przez studenta:	
Podczas ćwiczeń laboratoryjnych sprawdzane będą efekty U_01 - U_04 oraz K_01- K_03. Podczas egzaminu efekty U_01 - U_04 oraz W_01-W_05.	
Forma i warunki zaliczenia:	
<p>Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się oceny częściowe uzyskane na regularnych zajęciach z nauczycielem akademickim, za które można uzyskać maksymalnie 140 pkt. Zaliczenie zajęć laboratoryjnych i dopuszczenie do egzaminu jest możliwe po uzyskaniu co najmniej 71 pkt.</p> <p>Egzamin jest egzaminem pisemnym. Można na nim uzyskać do 60 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 30 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w 0 – 100 pkt: niedostateczna (F),</p> <ul style="list-style-type: none"> • 101 – 120 pkt: dostateczna (E), • 121 – 140 pkt: dostateczna plus (D), • 141 – 160 pkt: dobra (C), • 161 – 180 pkt: dobra plus (B), • 181 – 200 pkt: bardzo dobra (A).Poprawy: <p>Jednorazowa poprawa egzaminu w sesji. Jednorazowe poprawy do 4 ćwiczeń laboratoryjnych w trakcie semestru.</p>	
Bilans punktów ECTS:	
Studia stacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach laboratoryjnych	45 godz.

Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	10 godz.
Udział w konsultacjach godz. z przedmiotu	5 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS
Studia niestacjonarne	
Aktywność	Obciążenie studenta
Udział w wykładach	18 godz.
Udział w ćwiczeniach laboratoryjnych	24 godz.
Samodzielne przygotowanie się do ćwiczeń laboratoryjnych	37 godz.
Udział w konsultacjach godz. z przedmiotu	3 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	18 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Sylabus przedmiotu / modułu kształcenia			
Nazwa przedmiotu/modułu kształcenia:		Systemy baz danych	
Nazwa w języku angielskim:		Database Systems	
Język wykładowy:	polski		
Kierunek studiów, dla którego przedmiot jest oferowany:		Informatyka	
Jednostka realizująca:		Wydział Nauk Ścisłych i Przyrodniczych	
Rodzaj przedmiotu/modułu kształcenia (obowiązkowy/fakultatywny):		fakultatywny	
Poziom modułu kształcenia (np. pierwszego lub drugiego stopnia):		pierwszego stopnia	
Rok studiów:	trzeci		
Semestr:	szósty		
Liczba punktów ECTS:	4		
Imię i nazwisko koordynatora przedmiotu:		prof. dr hab. inż. Andrzej Barczak	
Imię i nazwisko prowadzących zajęcia:		prof. dr hab. inż. Andrzej Barczak mgr Wojciech Nabiałek, mgr Zbigniew Młynarski	
Założenia i cele przedmiotu:		Celem zajęć jest zapoznanie studentów z zagadnieniami związanymi z systemami baz danych oraz zaprezentowanie najnowszych tendencji ich rozwoju.	
Symbol efektu	Efekty uczenia się		Symbol efektu kierunkowego
	WIEDZA		
W_01	Zna i rozumie w zaawansowanym stopniu zagadnienia z zakresu teoretycznych podstaw baz danych, systemów baz danych, wykorzystywanych modeli oraz projektowania systemów relacyjnych i obiektowo-relacyjnych baz danych.		K_W08
W_02	Zna aktualny stan oraz najnowsze trendy w rozwoju systemów baz danych relacyjnych i relacyjno-obiektowych.		K_W08
UMIĘJĘTNOŚCI			
U_01	Potrafi pozyskiwać informacje na temat systemów baz danych z literatury i innych źródeł, w tym zwłaszcza internetowych; potrafi analizować, interpretować, porządkować, agregować i integrować oraz oceniać pod względem użyteczności uzyskane informacje, a także wyciągać wnioski na podstawie wyników swojej pracy.		K_U01
U_02	Posiada umiejętności samodzielnego pozyskiwania wiedzy w dziedzinie systemów baz danych.		K_U01
U_03	Potrafi ocenić przydatność dostępnych metod i narzędzi, a zwłaszcza środowisk programistycznych służących do rozwiązywania zadań inżynierskich z zakresu systemów baz danych oraz dobrać i stosować właściwe dla rozwiązania określonego problemu metody i narzędzia.		K_U10, K_U11
U_04	Potrafi właściwie oceniać i porównywać systemy baz danych ze względu na wybrane parametry.		K_U15, K_U16, K_U17

U_05	Potrafi, zgodnie ze specyfikacją, zaplanować proces realizacji informatycznego systemu bazodanowego.	K_U09, K_U11
KOMPETENCJE SPOŁECZNE		
K_01	Jest gotów do podejmowania decyzji, krytycznej oceny działań własnych, działań zespołów, którymi kieruje i organizacji, w których uczestniczy związanych z wykorzystaniem systemów baz i do ponoszenia odpowiedzialności za skutki swoich działań.	K_K01
K_02	Jest gotów do uznania znaczenia wiedzy w rozwiązywaniu problemów związanych z systemami baz danych oraz jest gotów do konstruktywnej krytyki w stosunku do działań swoich i innych osób.	K_K01
Forma i typy zajęć:	Studia stacjonarne: wykłady (30 godz.), ćwiczenia laboratoryjne (30 godz.) Studia niestacjonarne: wykłady (15 godz.), ćwiczenia laboratoryjne (15 godz.)	
Wymagania wstępne i dodatkowe:		
Wiedza z zakresu baz danych, systemów operacyjnych oraz inżynierii oprogramowania.		
Treści modułu kształcenia:		
<ol style="list-style-type: none"> 1. Język PL/SQL (cz.1). Wstęp do programowania w PL/SQL. Typy danych. Tworzenie bloków PL/SQL. 2. Język PL/SQL (cz. 2). Wyrażenia, operatory, funkcje i wyjątki. Struktury sterujące. 3. Język PL/SQL (cz. 3). Polecenia SQL w programie PL/SQL. Wyzwalacze. Przechowywanie procedur i funkcji. Pakiety. Kursory. Przegląd funkcji wbudowanych. 4. Język T-SQL Omówienie różnic między T-SQL a PL/SQL. Microsoft Common Table Expressions, rekursja w zapytaniach, konwersja wyników do XML/JSON, rankingi. 5. System zarządzania bazą danych. Zarządzanie plikami, wyszukiwanie informacji. Jądro SZBD, model pamięci zewnętrznej. Pliki nieuporządkowane, pliki sekwencyjne. Pliki haszowane, pliki indeksowe, indeks a struktura B-drzewa. 6. System zarządzania bazą danych (cz. 1). Zarządzanie transakcjami. Współbieżność transakcji - sytuacje konfliktowe. Metody zapewniające współbieżność transakcji. Metody blokowania. Zarządzanie integralnością bazy danych. 7. System zarządzania bazą danych (cz. 2). Niezawodność i odtwarzanie baz danych po awarii. Obrazy przed i po transakcji, uaktualnianie segmentów pamięci. Tabele słownikowe. Zarządzanie zapytaniami, optymalizacja zapytań. 8. Inteligentne bazy danych (cz. 1). Inteligencja, a bazy danych. Logika i bazy danych. Definicja danych w katalogu, operowanie danymi w Datalogu. Obsługa DDL w PL/SQL i T-SQL. 9. Inteligentne bazy danych (cz. 2). Datalog i relacyjny model danych. Różnice między Datalogiem i Prologiem. Systemy hipermedialne, geograficzne systemy informacyjne. Zasady budowy i działania aplikacji pracującej w architekturze trójwarstwowej. 10. Architektura bazy danych ORACLE (cz. 1). Baza danych i instancje. Wewnętrzna struktura bazy danych. Wewnętrzne obszary pamięci. Zasady budowy formularzy i przetwarzania wprowadzonych danych. 11. Architektura bazy danych ORACLE (cz. 2). Procesy drugoplanowe. Podstawowa konfiguracja bazy danych. Model danych i tworzenie bazy danych. Autoryzacja w aplikacji bazodanowej. 12. Architektura bazy danych MS SQL Server. Baza danych i instancje. Wewnętrzna struktura bazy danych. Wewnętrzne obszary pamięci. Podstawowa konfiguracja bazy danych. Model danych i tworzenie bazy danych. Autoryzacja w aplikacji bazodanowej. 13. Konfiguracja sprzętowa baz danych ORACLE i MS SQL Server (cz. 1). Połączone bazy danych. Zdalna modyfikacja danych. Zalety rozproszenia. 		

14. Konfiguracja sprzętowa baz danych ORACLE i MS SQL Server (cz. 2). Serwery klastrowe, konfiguracje wieloprocesorowe. Aplikacje typu klient-serwer. Architektura trójwarstwowa. Operacje sieciowe z użyciem wbudowanych pakietów.

15. Porównanie systemów bazodanowych. Różnice i podobieństwa w tworzeniu systemów bazodanowych m. in. w ORACLE, Microsoft SQL Server, PostgreSQL, MariaDB. Wybór odpowiedniego narzędzia do rozwiązywania określonych problemów bazodanowych.

Literatura podstawowa:

1. Connolly T., Begg C.: Systemy baz danych - Praktyczne metody projektowania, implementacji i zarządzania. Tom 1, 2; Wydawnictwo RM, 2004
2. Darwen H., Date C.J.: SQL. Omówienie standardu języka; Wydawnictwa Naukowo Techniczne, 2000
3. Elmasri R., Navathe S. B.: Wprowadzenie do systemów baz danych; Wydawnictwo Helion, 2005
4. Allen S.: Modelowanie danych; Wydawnictwo Helion, 2006C. J. Date , Wprowadzenie do systemów baz danych, WNT, 2000
5. Garcia-Molina H., Ullman J. D., Widom J.; Systemy baz danych Pełny wykład; Wydawnictwa Naukowo Techniczne, 2006
6. Dokumentacja Oracle: Application Developer's Guide - Fundamentals, PL/SQL User's Guide and Reference, PL/SQL Web Toolkit Reference, Using the PL/SQL Gateway.

Literatura dodatkowa:

1. Andrzej Barczak, Dariusz Zacharczuk, Stanisław Jastrzebowski, Comparative analysis of database access technology w Studia Informatica. System and information technology, Wyd. AP, Siedlce 2009 (vol. 12)
2. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 1. Tuning instance. w Studia Informatica.1-2 (16) 2012 , Wyd.UPH, Siedlce
3. Andrzej Barczak, Dariusz Zacharczuk, Anna Korzeniecka, The influence of indexing methods on effective functioning of the database w Studia Informatica.1-2 (17) 2013 , Wyd.UPH, Siedlce
4. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 2. Tuning of hardware applications and SQL queries w Studia Informatica.1 (18) 2014 , Wyd.UPH, Siedlce
5. Andrzej Barczak, Dariusz Zacharczuk, Damian Pluta, Tools and methods of databases optimization In Oracle 10g. Part 3. Theory in practice. w Studia Informatica.1 (18) 2014 , Wyd.UPH, Siedlce

Planowane formy/działania/metody dydaktyczne:

Wykład tradycyjny wspomagany technikami multimedialnymi, laboratoria – praktyczna praca na komputerze. Zamieszczanie na stronach internetowych zagadnień teoretycznych i zadań ćwiczeniowych

Sposoby weryfikacji efektów uczenia się osiągniętych przez studenta:

Efekty W_01 i W_02 będą weryfikowane na egzaminie pisemnym.

Efekty U_01 - U_05 weryfikowane będą w trakcie zajęć laboratoryjnych.

Efekty K_01 - K_02 będą weryfikowane, w oparciu o posiadaną wiedzę i umiejętności, w czasie zajęć laboratoryjnych. W trakcie zaliczania zadania indywidualnego.

Przykładowe pytania egzaminacyjne:

- Omów funkcje jądra SZBD,
- Omów wewnętrzne obiekty zaawansowanej bazy danych (ORACLE, SQL Serwer),
- Przedstaw architekturę zaawansowanej bazy danych (ORACLE, SQL Serwer),
- Zdefiniuj wyzwalacz aktualizujący dane zawarte w zaawansowanej bazie danych,
- Obsłuż sytuacje wyjątkowe zaistniałe przy przetwarzaniu danych w bazie danych.

Przed egzaminem studenci będą mieli dostęp do pełnej listy pytań.

Forma i warunki zaliczenia:

Moduł kończy się egzaminem. Do egzaminu mogą przystąpić osoby, które uzyskały zaliczenie laboratorium. Na zaliczenie laboratorium składają się średnie ważone punktów uzyskanych na regularnych laboratoriach (60%) oraz punkty z wykonanego zadania indywidualnego (40%).

Zajęcia laboratoryjne będą zaliczone w wypadku uzyskania:

- minimum 51% z regularnych zajęć (61/120 pkt)
- minimum 51% z zadania indywidualnego (16/30 pkt).

Maksymalne z tej części zajęć można uzyskać 150 pkt.

Egzamin jest egzaminem pisemnym. Można na nim uzyskać maksymalnie 50 pkt. Egzamin będzie zaliczony w przypadku uzyskania co najmniej 25 pkt. Ocena końcowa z modułu (wystawiana po zaliczeniu wszystkich części składowych), w zależności od sumy uzyskanych punktów (maksymalnie 200 pkt) jest następująca (w nawiasach ocena w skali ECTS):

- 0-100 pkt – ndst (F),
- 101- 120 pkt – dst (E),
- 121-140 pkt – dst+ (D),
- 141-160 pkt – db (C),
- 161-180 pkt – db+ (B),
- 181-200 pkt – bdb (A).

Poprawy: jednorazowa poprawa egzaminu pisemnego. Jednorazowe poprawy do 2 ćwiczeń laboratoryjnych w trakcie trwania semestru.

Bilans punktów ECTS:**Studia stacjonarne**

Aktywność	Obciążenie studenta
Udział w wykładach	30 godz.
Udział w ćwiczeniach	30 godz.
Udział w konsultacjach z przedmiotu	5 godz.
Samodzielne przygotowanie się do ćwiczeń	25 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	10 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS

Studia niestacjonarne

Aktywność	Obciążenie studenta
Udział w wykładach	15 godz.
Udział w ćwiczeniach	15 godz.
Udział w konsultacjach z przedmiotu	3 godz.
Samodzielne przygotowanie się do ćwiczeń	37 godz.
Przygotowanie się do egzaminu i obecność na egzaminie	30 godz.
Sumaryczne obciążenie pracą studenta	100 godz.
Punkty ECTS za przedmiot	4 ECTS