

KARTA KURSU (realizowanego w specjalności)**Tworzenie gier komputerowych**

(nazwa specjalności)

Nazwa	Tworzenie gier komputerowych
Nazwa w j. ang.	Creating computer games

Koordynator	mgr inż. Krystian Kurnik	Zespół dydaktyczny
		mgr inż. Krystian Kurnik
Punktacja ECTS*	st. stacjonarne: 5 st. niestacjonarne: 5	

Opis kursu (cele kształcenia)

Celem kształcenia w ramach przedmiotu *Tworzenie Gier Komputerowych* jest nabycie przez studentów praktycznych umiejętności projektowania i implementacji trójwymiarowych gier komputerowych w nowoczesnych silnikach gier (Unity lub Unreal Engine). Studenci zdobędą kompetencje w zakresie podstawowych technik modelowania scen 3D, programowania logiki gry, integracji zasobów multimedialnych oraz testowania i optymalizacji działania aplikacji. Efektem końcowym pracy studenta będzie autorska gra 3D opracowana indywidualnie lub zespołowo, prezentująca zrozumienie i zastosowanie zdobytej wiedzy oraz narzędzi. Przedmiot ma również na celu rozwijanie kreatywności, umiejętności pracy projektowej oraz kompetencji miękkich związanych z prezentacją gotowego produktu.

Warunki wstępne

Wiedza	umiejętność programowania w języku C# lub C++, przynajmniej na poziomie podstawowym (tworzenie klas, metod, obsługa zdarzeń, prosta logika sterująca), podstawowe rozumienie zagadnień grafiki komputerowej, takich jak układ współrzędnych 2D/3D, siatki (meshe), tekstury i światło,
Umiejętności	umiejętność pracy z dowolnym środowiskiem IDE (np. Visual Studio, Rider, Visual Studio Code), umiejętność korzystania z dokumentacji technicznej i samodzielnego wyszukiwania informacji (np. w dokumentacji API, forach, tutorialach), podstawowa umiejętność pracy projektowej: dzielenie zadań, planowanie pracy, dotrzymywanie terminów,
Kursy	Matematyka (geometria, układy współrzędnych, transformacje 3D), Systemy operacyjne (podstawowe zarządzanie, administracja, obsługa procesów), Organizacja i architektura komputerów (podstawowe elementy budowy systemów komputerowych), Podstawy programowania (struktury danych, algorytmy, podstawowe operacje logiczne i matematyczne).

Efekty uczenia się

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Wiedza	Po zakończeniu kursu student: W01: zna pojęcia związane z możliwościami oraz obsługą silników do projektowania i produkcji nowoczesnych interaktywnych aplikacji jakimi są gry komputerowe, W02: zna i rozumie etapy procesu twórczego prowadzącego do powstania gry komputerowej, W03: posiada wiedzę na temat specjalistycznego oprogramowania używanego w procesie projektowania wirtualnych światów,	S1_W01 S1_W03

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Umiejętności	Po zakończeniu kursu student: U01: potrafi wykonać w sposób twórczy wirtualny świat za pomocą wskazanego silnika, U02: potrafi zaplanować realizację elementów na poszczególnych etapach produkcji wirtualnej rzeczywistości,	S1_U01; S1_U02; S1_U03; S1_U05 S1_U06; S1_U10 S1_U11

	Efekt uczenia się dla kursu	Odniesienie do efektów kierunkowych
Kompetencje społeczne	Po zakończeniu kursu student: K01: potrafi pracować w grupie, przyjmując różne role w zespole K02: rozumie potrzebę stałego poszerzania swojej wiedzy i umiejętności oraz śledzenia postępu technologicznego, K03: doskonalili umiejętności związane z dzieleniem się wiedzą, rozumie potrzebę zrozumiałego formułowania treści, potrafi dobrać odpowiednie narzędzia do prezentacji treści, uwypuklić mocne strony swojego projektu	S1_K01 S1_K02 S1_K03 S1_K04

Studia stacjonarne

Organizacja							
Forma zajęć	Wykład (W)	Ćwiczenia w grupach					
		A	K	L	S	P	E
Liczba godzin	25			30			

Studia niestacjonarne

Organizacja							
Forma zajęć	Wykład (W)	Ćwiczenia w grupach					
		A	K	L	S	P	E
Liczba godzin	15			20			

Opis metod prowadzenia zajęć

Zajęcia prowadzone będą w formie laboratoriów komputerowych, z naciskiem na aktywne uczenie się poprzez praktykę. Wykorzystane metody dydaktyczne obejmują:

- **Instruktaż praktyczny** – krótkie wprowadzenia do poszczególnych zagadnień technicznych (np. interfejs silnika, mechanika fizyki, system animacji) prezentowane przez prowadzącego, bezpośrednio połączone z ćwiczeniami praktycznymi.
- **Zadania warsztatowe** – samodzielne lub zespołowe rozwiązywanie zadań o rosnącym poziomie trudności, obejmujących implementację konkretnych mechanik i funkcjonalności gier 3D.
- **Praca projektowa** – główna forma aktywności, w ramach której studenci realizują własną grę 3D w wybranym silniku (Unity lub Unreal Engine), zgodnie z ustalonym harmonogramem pracy projektowej.
- **Konsultacje i feedback bieżący** – regularne konsultacje indywidualne lub grupowe z prowadzącym, w trakcie których omawiane są postępy, problemy techniczne i koncepcyjne.
- **Prezentacje i przeglądy postępów** – wybrane zajęcia będą wykorzystywane do prezentowania etapów projektu przez studentów, co umożliwi ocenę postępu, wzajemne uczenie się oraz doskonalenie umiejętności komunikacyjnych.
- **Wspieranie samodzielności** – studenci są zachęceni do eksplorowania dokumentacji, korzystania z tutoriali online i samodzielnego rozwiązywania problemów, co rozwija kompetencje niezbędne w pracy w branży gier.

Zajęcia mają charakter praktyczny, a teoria ograniczona jest do niezbędnego minimum służącego zrozumieniu zasad działania wykorzystywanych narzędzi i technologii.

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	Zadania problemowe
W01					X	X	X	X				X	
W02					X	X	X	X				X	
W03					X	X	X	X				X	
U01					X	X	X	X					
U02					X	X	X	X					
K01					X	X	X	X					
K02					X	X	X	X					
K03					X	X	X	X					

Kryteria oceny	Zaliczenie kursu opiera się na ocenie efektów kształcenia osiągniętych przez studenta w ramach pracy indywidualnej lub zespołowej. Warunkiem uzyskania zaliczenia jest spełnienie następujących wymagań: 1. Projekt zaliczeniowy lub ćwiczenia praktyczne a. Student wykonuje projekt zaliczeniowy zgodnie z wytycznymi prowadzącego lub realizuje zadania praktyczne podczas zajęć. b. Forma zaliczenia może obejmować zarówno samodzielnie wykonany projekt, jak i praktyczne ćwiczenia laboratoryjne, w zależności od specyfiki grupy i ustaleń prowadzącego. 2. Test teoretyczny a. Weryfikacja wiedzy teoretycznej odbywa się poprzez test zaliczeniowy lub serię krótszych testów cząstkowych. b. Warunkiem zaliczenia jest uzyskanie co najmniej 50% punktów z testu. Aby zaliczyć kurs, student musi spełnić oba warunki: pomyślnie ukończyć projekt/ćwiczenie oraz osiągnąć wymagany wynik z testu teoretycznego. Szczegółowe wymagania dotyczące formy realizacji projektu oraz harmonogram testów są określone przez prowadzącego.
----------------	---

Uwagi	
-------	--

Treści merytoryczne (wykaz tematów)

Podczas zajęć studenci realizują serię ćwiczeń praktycznych, które stopniowo wprowadzają ich w zagadnienia związane z projektowaniem i implementacją środowisk wirtualnych. Kluczowym elementem kursu jest wykonanie **projektu końcowego** – interaktywnej gry w środowisku 3D.

Gra powinna zawierać:

- **Trójwymiarowy świat**, zaprojektowany przy użyciu wybranego silnika VR (np. Unity, Unreal Engine),
- **Głównego bohatera**, który może się poruszać, oddziaływać ze światem i wykonywać określone akcje,
- **Interakcje i mechaniki gry**, które wzbogacają doświadczenie użytkownika.

Zakres tematyczny zajęć:

- 1. Wprowadzenie do rzeczywistości wirtualnej (VR) i rozszerzonej (AR)**
 - a. Definicja i zastosowania VR i AR w różnych branżach
 - b. Przegląd popularnych technologii VR i silników gier
- 2. Architektura silników VR**
 - a. Struktura i komponenty silnika do tworzenia rzeczywistości wirtualnej
 - b. Praca z edytorami 3D oraz narzędziami do modelowania scen
- 3. Modelowanie świata gry**
 - a. Tworzenie i importowanie modeli 3D
 - b. Projektowanie krajobrazu, struktur i elementów otoczenia
 - c. Optymalizacja grafiki i zasobów w celu poprawy wydajności
- 4. Sterowanie postaciami i interakcja z otoczeniem**
 - a. Implementacja systemu poruszania się postaci w świecie 3D
 - b. Animacja ruchu bohatera i integracja z fizyką gry
 - c. Kolizje i interakcje z obiektami wirtualnymi
- 5. Mechaniki gry i logika interakcji**
 - a. Systemy sterowania kamerą (widok pierwszoosobowy i trzecioosobowy)
 - b. Reakcje na działania użytkownika (np. skakanie, podnoszenie przedmiotów)
 - c. Implementacja mechaniki grawitacji i detekcji przeszkód
- 6. Zaawansowane efekty w rzeczywistości wirtualnej**
 - a. Oświetlenie i cieniowanie w środowisku VR
 - b. Implementacja efektów specjalnych (mgła, cząsteczki, dynamiczne światła)
 - c. Integracja dźwięku przestrzennego
- 7. Fizyka obiektów i interakcje wirtualne**
 - a. Wprowadzenie do silników fizycznych w VR
 - b. Kolizje, siły i interakcja z obiektami dynamicznymi

- c. Symulacja efektów fizycznych (np. grawitacja, sprężystość, tarcie)
- 8. Projektowanie interfejsu użytkownika (UI) w VR**
 - a. Tworzenie HUD (Heads-Up Display) i elementów interakcji
 - b. Implementacja interfejsów użytkownika w wirtualnej rzeczywistości
 - c. Sterowanie gestami i kontrolerami VR
- 9. Optymalizacja wydajności i testowanie gry**
 - a. Strategie optymalizacji zasobów i renderowania w VR
 - b. Debugowanie i testowanie środowiska na różnych urządzeniach VR
 - c. Poprawa płynności i stabilności działania aplikacji
- 10. Finalizacja projektu i prezentacja wyników**
 - o Integracja wszystkich komponentów gry w gotowy projekt
 - o Przygotowanie dokumentacji i prezentacji projektu
 - o Analiza wyzwań i przyszłych możliwości rozwoju gry

Mechaniki do zaimplementowania w projekcie

Każdy projekt powinien zawierać kluczowe mechaniki, które uczynią grę interaktywną i angażującą:

- **System ruchu postaci** (chód, sprint, skok)
- **Kamera gracza** (widok FPP lub TPP)
- **Kolizje z otoczeniem** (detekcja przeszkód, ograniczenia ruchu)
- **Podstawowa interakcja** (np. podnoszenie i przesuwanie przedmiotów)
- **Zarządzanie fizyką** (grawitacja, odbicia, popychanie obiektów)
- **Światło i cienie** (symulacja warunków oświetleniowych)
- **Efekty dźwiękowe i wizualne** (np. kroki, zmiany środowiskowe)

Dzięki połączeniu ćwiczeń praktycznych i realizacji kompleksowego projektu, studenci zdobywają umiejętności niezbędne do projektowania interaktywnych aplikacji VR. Kurs zapewnia solidne podstawy do pracy nad bardziej zaawansowanymi projektami w branży gier komputerowych oraz innych dziedzinach wykorzystujących rzeczywistość wirtualną.

Wykaz literatury podstawowej

1. **Jeremy Gibson Bond** – *Projektowanie gier przy użyciu środowiska Unity i języka C#*, Wydanie II, Helion, 2018.
2. **Joanna Pasek** – *3ds Max 2010: Animacja 3D od podstaw*, Helion, 2010.
3. **David M. Bourg, Amber Mac** – *Fizyka dla programistów gier*, Helion, 2003.
4. **Ernest Adams** – *Projektowanie gier. Podstawy*, Wydanie II, Helion, 2010.
5. **Steven M. LaValle** – *Virtual Reality*, Cambridge University Press, 2017. (Podstawy VR, matematyka, algorytmy, interakcja)
6. **Tony Parisi** – *Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile*, O'Reilly Media, 2015.
7. **Jason Gregory** – *Game Engine Architecture*, Wydanie III, CRC Press, 2018.
8. **Alan B. Craig** – *Understanding Augmented Reality: Concepts and Applications*, Morgan Kaufmann, 2013.
9. **Zacharias Voulgaris** – *AI for Game Developers: Creating Intelligent Behavior in Games*, O'Reilly Media, 2020.
10. **Robert Scoble, Shel Israel** – *The Fourth Transformation: How Augmented Reality & Artificial Intelligence Change Everything*, Patrick Brewster Press, 2016.

Wykaz literatury uzupełniającej

1. **John Vince** – *Virtual Reality Systems*, Addison-Wesley, 1995. (Klasyczna pozycja wprowadzająca do VR)
2. **William R. Sherman, Alan B. Craig** – *Understanding Virtual Reality: Interface, Application, and Design*, Morgan Kaufmann, 2018.
3. **Jason Jerald** – *The VR Book: Human-Centered Design for Virtual Reality*, Morgan & Claypool, 2015.
4. **John C. Russ** – *The Image Processing Handbook*, CRC Press, 2020. (Przetwarzanie obrazu w VR)
5. **Mick West** – *Game Programming Patterns*, Genever Benning, 2014. (Optymalizacja kodu w grach VR)
6. **Pablo C. Figueroa** – *Augmented and Virtual Reality: Theory and Practice*, Springer, 2022.
7. **Dieter Schmalstieg, Tobias Hollerer** – *Augmented Reality: Principles & Practice*, Addison-

Wesley, 2016.

8. **Tobias Kunz, Daniel Roth** – *Developing Virtual Reality Applications: Foundations of Effective Design*, Springer, 2021.
9. **Benjamin W. Wah (ed.)** – *Encyclopedia of Computer Science and Engineering*, Vol. 1-4, Wiley, 2008.
10. **Toni Parisi** – *WebGL Up and Running: Building 3D Graphics for the Web*, O'Reilly Media, 2012. (VR w przeglądarkach)

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia stacjonarne**

Liczba godzin w kontakcie z prowadzącymi	Wykład	20
	Konwersatorium (ćwiczenia, laboratorium itd.)	30
	Pozostałe godziny kontaktu studenta z prowadzącym	10
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	20
	Realizacja zadań domowych (problemowych) po zapoznaniu się z niezbędną literaturą przedmiotu	10
	Przygotowanie projektu lub prezentacji na podany temat (praca indywidualna lub w grupie)	20
	Przygotowanie do egzaminu/zaliczenia	25
Ogółem bilans czasu pracy		125
Liczba punktów ECTS w zależności od przyjętego przelicznika		5

Bilans godzinowy zgodny z CNPS (Całkowity Nakład Pracy Studenta) – **studia niestacjonarne**

Liczba godzin w kontakcie z prowadzącymi	Wykład	10
	Konwersatorium (ćwiczenia, laboratorium itd.)	20
	Pozostałe godziny kontaktu studenta z prowadzącym	10
Liczba godzin pracy studenta bez kontaktu z prowadzącymi	Lektura w ramach przygotowania do zajęć	15
	Realizacja zadań domowych (problemowych) po zapoznaniu się z niezbędną literaturą przedmiotu	15
	Przygotowanie projektu lub prezentacji na podany temat (praca indywidualna lub w grupie)	30
	Przygotowanie do egzaminu/zaliczenia	25
Ogółem bilans czasu pracy		125
Liczba punktów ECTS w zależności od przyjętego przelicznika		5