

	Nazwa modułu. Blok przedmiotów wybieralnych						Kod modułu: M23
Wypełnia Zespół Kierunku	Nazwa przedmiotu: Budowa aplikacji na platformie .NET						Kod przedmiotu:
	Nazwa jednostki prowadzącej przedmiot / moduł: INSTYTUT INFORMATYKI STOSOWANEJ						
	Nazwa kierunku: INFORMATYKA						
	Forma studiów: niestacjonarne			Profil kształcenia: PRAKTYCZNY		Specjalność: Projektowanie baz danych i oprogramowanie użytkowe	
	Rok / semestr: 3/5			Status przedmiotu / modułu: obowiązkowy		Język przedmiotu / modułu: polski	
	Forma zajęć	wykład	ćwiczenia	ćwiczenia laboratoryjne	konwersatorium	seminarium	inne (wpisać jakie)
	Wymiar zajęć	15		30			
	Koordynator przedmiotu / modułu		mgr inż. Marzanna Skowrońska				
Prowadzący zajęcia		mgr inż. Marzanna Skowrońska					
Cel przedmiotu / modułu		<p>Przedstawienie zagadnień związanych z stosowaniem nowych technologii informatycznych w wytwarzaniu oprogramowania;</p> <p>Nauczenie programowania w języku zorientowanym obiektowo C#, posługiwanie się środowiskiem wytwarzania oprogramowania Microsoft Visual Studio. NET, implementacji komponentów aplikacji, tworzenia kompletnych aplikacji desktopowych</p>					
Wymagania wstępne		podstawy programowania					
EFEKTY KSZTAŁCENIA						Odniesienie do efektów dla programu	
Nr	Wiedza						
01	operuje gramatyką języka obiektowego C# w środowisku Visual Studio, dobierając składnię adekwatnie do zadanego problemu					K_W11	
02	modeluje rozwiązanie problemu zgodnie z paradygmatem obiektowym, budową komponentową, modelem warstwowym aplikacji					K_W13	
	Umiejętności						
03	sprawnie porusza się po zintegrowanym środowisku programistycznym, bezbłędnie wykonując wszystkie niezbędne operacje związane z tworzeniem, testowaniem, uruchamianiem kodu, czytając i wyciągając wnioski z komentarzy oraz błędów w języku angielskim					K_U16, K_U13, K_U05	
04	tworzy desktopowe aplikacje w języku C# dość złożone pod względem architektonicznym					K_U17, K_U19, K_U12	
05	rozwija umiejętności sięgania po różnego typu zasoby literaturowe także w języku angielskim, poszukując optymalnego rozwiązania zadanego problemu					K_U01, K_U05	
06	dba o ergonomiczny interfejs użytkownika z myślą o przyszłych użytkownikach jego programów					K_U11	
	Kompetencje społeczne						
07	prezentuje publicznie efekty swojej pracy					K_U04	
08	komentuje kod źródłowy oraz komponuje go w poprawnych porcjach logicznych z myślą o innych programistach, którzy mogliby rozwijać jego kod					K_K02	
TREŚCI PROGRAMOWE							
Forma zajęć – WYKŁAD							
<ol style="list-style-type: none"> Wprowadzenie do nowego podejścia do kompilacji. Charakterystyka platformy Microsoft .NET Framework; architektura, CLR, BCL. Wyjaśnienie zagadnień: podzespoły, języki programowania, zarządzanie pamięcią, wspólne typy danych. Wytwarzanie aplikacji .NET. Podstawy języka C# , analogia do języka Java. Tworzenie programów, kompilacja, uruchamianie, debugowanie z wykorzystaniem prostego edytora i kompilatora oraz zintegrowanego środowiska programistycznego Visual Studio. Konstrukcje autonomiczne w języku C# jak: przestrzenie nazw, indexer'y, delegat'y, event'y, indeksy tablic. Wzorzec 							

<p>projektowy <i>fabryka</i>.</p> <ol style="list-style-type: none"> 3. Tworzenie konstrukcji złożonych desktopowych aplikacji okienkowych w Visual Studio: solution'y, projekty, elementy projektów różnych typów. Analiza startu aplikacji (<i>entry point, set as startup</i>). Cykl życia formy. Przekazywanie sterowania między formami poprzez przeciążanie konstruktora i rzutowanie obiektu. 4. Poznanie bazowych kontrolek środowiska dostarczonych przez producenta, w tym menu, pasek narzędzi, okna dialogowe. Walidacja danych. Obsługa błędów. 5. Tworzenie aplikacji typu <i>single</i> i <i>multiplay document</i>. Podsumowanie poznanych treści na przykładzie dość złożonego edytora. 6. Budowa różnych typów kontrolek. Tworzenie bibliotek komponentów. Model trójwarstwowy aplikacji. 7. Metody dostępu do danych, idea ADO.NET. Tworzenie aplikacji z dostępem do repozytorium danych w trybie połączeniowym i bezpołączeniowym ze wsparciem automatycznego kreatora obiektów dostępowych i z poziomu kodu 8. Użytkowanie generator raportów wbudowanego w środowisko. 9. Tworzenie wydruków z poziomu kodu. 10. Wsparcie użytkownika, dostosowywanie aplikacji do użytkownika niepełnosprawnego. 11. Aplikacje w wielu wersjach językowych. 12. Zabezpieczenie aplikacji, zarządzanie konfiguracją, tworzenie aplikacji instalacyjnej. 		
<p>Forma zajęć – PROJEKT</p> <p>Celem laboratorium jest implementacja zagadnień tworzenia aplikacji w języku C# przy wykorzystaniu środowiska Microsoft Visual Studio. NET.</p> <p>Na wstępnych zajęciach studenci poznają język C# i tworzą małe programy przy użyciu zwykłego edytora i kompilatora. Następnie nabywają umiejętności posługiwania się zintegrowanym środowiskiem programistycznym Microsoft Visual Studio.NET.</p> <p>Później tworzą fragmenty kodu według zadanego scenariusza, uzupełniając odpowiednie do każdego tematu projekty, a także dość złożone aplikacje od zera na podsumowanie omawianego bloku zagadnień.</p> <p>Zajęcia laboratoryjne prowadzone są w oparciu o materiały kursu Microsoft IT Academy "Developing Microsoft .NET Applications for Windows(Visual C#.NET)" oraz "Programming in C#" w postaci statycznych plików .pdf i multimedialnych umieszczonych na platformie http://itacademy.microsoftlearning.com/.</p>		
Metody kształcenia	Wykład z prezentacją multimedialną, ćwiczenia laboratoryjne, dyskusja, rozwiązywanie zadań, praca indywidualna i zespołowa w laboratorium, konsultacja indywidualna z wykładowcą;	
Metody weryfikacji efektów kształcenia		Nr efektu kształcenia z sylabusu
1.	Na każdym zajęciach laboratoryjnych Student implementuje rozwiązanie zadań zdefiniowanych przez prowadzącego.	02,03,04,05
2.	Student ustnie prezentuje przed nauczycielem swoje rozwiązanie i odpowiada na pytania dotyczące kodu, uzasadniając swoją koncepcję lub referując poznana konstrukcję kodową, o ile była ona narzucona.	01
3.	Na koniec semestru Student tworzy autorską aplikację z możliwością adopcji komponentów programowych autorstwa innej osoby bez naruszenia jej praw autorskich.	02,03,04,06,08
4.	Student prezentuje publicznie swoją aplikację od strony funkcjonalnej i architektonicznej.	07
5.	Student udziela wyczerpującej odpowiedzi na pytania dotyczące kodu swojej aplikacji, implementującego wyrywkowo wskazaną funkcję widzianą z perspektywy użytkownika.	01
Forma i warunki zaliczenia	Podstawą zaliczenia laboratorium jest systematyczny, aktywny udział w zajęciach i zdobywanie punktów za wykonane zadania oraz udzielanie wyjaśnień odnośnie implementowanych zagadnień w odpowiedzi na pytania zadawane przez prowadzącego; oceny powyżej 72% upoważniają do otrzymania świadectwa ukończenia kursu Microsoft. Podstawą zaliczenia wykładu jest prezentacja autorskiej aplikacji o zadanej specyfikacji funkcjonalnej, implementującej zagadnienia omawiane na wykładzie a przećwiczone na laboratorium.	
Literatura podstawowa	<ol style="list-style-type: none"> 1. http://msdn.microsoft.com/en-us/library 2. Workbook – "Developing Microsoft .NET Applications for Windows(Visual C#.NET)", 2002 Microsoft Corporation (plik formatu 'pdf'). 3. Workbook – "Programming in C#.", 2002 Microsoft Corporation (plik formatu 'pdf'). 4. http://itacademy.microsoftlearning.com/ 	
Literatura uzupełniająca	Perry S.: <i>C# i .NET</i> . Gliwice, Helion 2006.	

NAKLAD PRACY STUDENTA:

	Liczba godzin
Udział w wykładach	15
Samodzielne studiowanie tematyki wykładów	15
Udział w ćwiczeniach audytoryjnych i laboratoryjnych*	30
Samodzielne przygotowywanie się do ćwiczeń*	30
Przygotowanie projektu / eseju / itp.*	40
Przygotowanie się do egzaminu / zaliczenia	15
Udział w konsultacjach	5
Inne	
ŁĄCZNY nakład pracy studenta w godz.	150
Liczba punktów ECTS za przedmiot	6 ECTS
Obciążenie studenta związane z zajęciami praktycznymi*	90 godz. 3,6 pkt ECTS
Obciążenie studenta na zajęciach wymagających bezpośredniego udziału nauczycieli akademickich	50 godz. 2 pkt ECTS