

Z programu nauczania można korzystać wyłącznie z podręcznikami Wydawnictwa Helion, celem innego wykorzystania prosimy o kontakt z właścicielem praw autorskich – Wydawnictwem Helion.

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autorzy oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autorzy oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Joanna Zaręba

Wydawnictwo HELION
ul. Kościuszki 1c, 44-100 GLIWICE
tel. 32 231 22 19, 32 230 98 63
e-mail: helion@helion.pl
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!
Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres
<http://helion.pl/user/opinie?tpi34t>
Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzje.

ISBN: 978-83-283-5965-9

Copyright © Helion 2019

PROGRAM NAUCZANIA DLA ZAWODU

TECHNIK PROGRAMISTA
351406

O STRUKTURZE
PRZEDMIOTOWEJ
TYP SZKOŁY: technikum 4-letnie

KWALIFIKACJE WYODRĘBNIONE W ZAWODZIE

INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych

INF.04. Projektowanie, programowanie i testowanie aplikacji

Spis treści

I. PODSTAWY PRAWNE KSZTAŁCENIA ZAWODOWEGO	6
II. CELE I ZADANIA KSZTAŁCENIA ZAWODOWEGO	7
1. OGÓLNE CELE I ZADANIA KSZTAŁCENIA ZAWODOWEGO	7
2. SZCZEGÓŁOWE CELE KSZTAŁCENIA W ZAWODZIE TECHNIK PROGRAMISTA	7
III. KORELACJA MIĘDZYPRZEDMIOTOWA	8
1. PRZEDMIOTY ROZSZERZONE W ZAWODZIE TECHNIK PROGRAMISTA	8
2. KORELACJA PROGRAMU NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA Z PODSTAWĄ PROGRAMOWĄ KSZTAŁCENIA OGÓLNEGO	9
IV. PLANY NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA	10
1. Minimalna liczba godzin kształcenia zawodowego w kwalifikacjach wyodrębnionych w zawodzie	10
2. Plan nauczania dla zawodu technik programista o strukturze przedmiotowej	11
2.1. Plan nauczania dla technika programisty — zajęcia edukacyjne z zakresu kształcenia zawodowego kończą się po pierwszym semestrze klasy IV	11
2.2. Praktyki zawodowe	13
V. WYKAZ PRZEDMIOTÓW WYODRĘBNIONYCH W PROGRAMIE NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA	14
VI. PROGRAMY NAUCZANIA DLA POSZCZEGÓLNYCH PRZEDMIOTÓW W ZAWODZIE TECHNIK PROGRAMISTA	15
1. Programy nauczania określające kompetencje personalne i społeczne	15
1.1. Kompetencje personalne i społeczne	15
1.2. Organizacja pracy małych zespołów	19
2. Programy nauczania dla kwalifikacji INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych	21
2.1. Język angielski zawodowy	21
2.2. Bezpieczeństwo i higiena pracy	27
2.3. Podstawy informatyki	32
2.4. Projektowanie i tworzenie stron internetowych	38
2.5. Projektowanie i administrowanie bazami danych	44

2.6. Programowanie aplikacji internetowych	49
2.7. Pracownia stron internetowych	53
2.8. Pracownia baz danych	60
2.9. Pracownia aplikacji internetowych	65
3. Programy nauczania dla kwalifikacji INF.04. Projektowanie, programowanie i testowanie aplikacji	71
3.1. Projektowanie oprogramowania	71
3.2. Podstawy programowania	77
3.3. Programowanie obiektowe	79
3.4. Programowanie aplikacji mobilnych	84
3.5. Zaawansowane aplikacje webowe	87
3.6. Testowanie i dokumentowanie aplikacji	90
3.7. Pracownia projektowania oprogramowania	93
3.8. Pracownia podstaw programowania	99
3.9. Pracownia programowania obiektowego	102
3.10. Pracownia programowania aplikacji mobilnych	107
3.11. Pracownia zaawansowanych aplikacji webowych	112
3.12. Pracownia testowania i dokumentowania aplikacji	115

I. Podstawy prawne kształcenia zawodowego

Program nauczania dla zawodu opracowany jest zgodnie z poniższymi aktami prawnymi:

- Ustawa z dnia 22 listopada 2018 r. o zmianie ustawy Prawo oświatowe, ustawy o systemie oświaty oraz niektórych innych ustaw (Dz. U. z 30 listopada 2018 r., poz. 2245),
- Ustawa z dnia 22 grudnia 2015 r. o Zintegrowanym Systemie Kwalifikacji (Dz. U. z 14 stycznia 2016 r., poz. 64 z późn. zm.),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 15 lutego 2019 r. w sprawie ogólnych celów i zadań kształcenia w zawodach szkolnictwa branżowego oraz klasyfikacji zawodów szkolnictwa branżowego (Dz. U. z 19 lutego 2019 r., poz. 316),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 16 maja 2019 roku w sprawie podstaw programowych kształcenia w zawodach szkolnictwa branżowego oraz dodatkowych umiejętności zawodowych w zakresie wybranych zawodów szkolnictwa branżowego (Dz. U. z),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 3 kwietnia 2019 roku w sprawie ramowych planów nauczania dla publicznych szkół (Dz. U. z 3 kwietnia 2019r., poz. 639),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 7 lutego 2012 r. w sprawie ramowych planów nauczania w szkołach publicznych (Dz. U. z 22 lutego 2012 r., poz. 204),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 22 lutego 2019 roku w sprawie praktycznej nauki zawodu (Dz. U. z 28 lutego 2019 r., poz. 391),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 8 lipca 2014 r. w sprawie dopuszczania do użytku szkolnego podręczników (Dz. U. z 9 lipca 2014 r., poz. 909),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 10 czerwca 2015 r. w sprawie szczegółowych warunków i sposobu oceniania, klasyfikowania i promowania uczniów i słuchaczy w szkołach publicznych (Dz. U. z 18 czerwca 2015 r., poz. 843 z późn. zm.),
- Rozporządzenie Ministra Edukacji Narodowej z dnia 27 kwietnia 2015 r. w sprawie szczegółowych warunków i sposobu przeprowadzania egzaminu potwierdzającego kwalifikacje w zawodzie (Dz. U. z 18 czerwca 2015 r., poz. 673).

II. CELE I ZADANIA KSZTAŁCENIA ZAWODOWEGO

1. OGÓLNE CELE I ZADANIA KSZTAŁCENIA ZAWODOWEGO

Celem kształcenia zawodowego jest opanowanie przez uczącego się wiedzy i umiejętności w zakresie danego zawodu, rozwijanie jego zainteresowań i zdolności, ważnych przy wykonywaniu danego zawodu oraz przygotowanie go do pracy w wyuczonym zawodzie, z jednoczesnym wdrożeniem do doskonalenia i podnoszenia kwalifikacji zawodowych przez całe życie.

Proces kształcenia zawodowego powinien być zintegrowany i skorelowany z kształceniem ogólnym oraz zapewniać doskonalenie kompetencji kluczowych, nabytych w procesie kształcenia ogólnego na wcześniejszych etapach edukacyjnych.

Realizacja treści nauczania w poszczególnych kwalifikacjach zawodowych odbywa się na bazie oczekiwanych efektów kształcenia oraz kryteriów weryfikacji z podstawy programowej. Osiągnięcie odpowiedniego poziomu wiedzy ogólnej w połączeniu z wiedzą zawodową powinno zapewnić absolwentom szkół kształcących w zawodzie technik programista opanowanie we właściwym stopniu umiejętności zawodowych oraz zagwarantować ich dostosowanie do zmieniającego się rynku pracy.

Zadaniem szkoły oraz innych podmiotów prowadzących kształcenie zawodowe jest stworzenie odpowiednich warunków zbliżonych do realiów otoczenia gospodarczo-społecznego, na które mają wpływ: czynniki gospodarcze, rozwój technologii, mobilność zawodowa, stosowanie innowacyjnych rozwiązań w życiu codziennym, a także oczekiwania pracodawców i rynku pracy w zakresie poziomu wiedzy i umiejętności pracowników.

Niezmiernie ważnym zadaniem jest również wspomaganie rozwoju ucznia, uwzględnienie jego indywidualnych potrzeb rozwojowych i edukacyjnych, dostosowanie warunków, środków, metod i form kształcenia do jego możliwości psychofizycznych oraz stymulowanie i motywowanie go do samorozwoju, samodoskonalenia, rozwijania pasji i zainteresowań technicznych i zawodowych.

Opracowany program nauczania pozwoli na osiągnięcie powyższych celów oraz realizację zadań ogólnych kształcenia zawodowego.

2. SZCZEGÓŁOWE CELE KSZTAŁCENIA W ZAWODZIE TECHNIK PROGRAMISTA

W zawodzie technik programista zostały wyodrębnione następujące kwalifikacje:

INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych,

INF.04. Projektowanie, programowanie i testowanie aplikacji.

Absolwent szkoły kształcącej w zawodzie technik programista po zakończeniu nauki powinien być przygotowany do wykonywania następujących zadań zawodowych:

W zakresie kwalifikacji *INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych*:

- tworzenia i administracji stronami WWW;
- tworzenia, administracji i użytkowania relacyjnych baz danych;
- programowania aplikacji internetowych;
- tworzenia i administracji systemami zarządzania treścią.

W zakresie kwalifikacji *INF.04. Projektowanie, programowanie i testowanie aplikacji*:

- projektowania, programowania i testowania zaawansowanych aplikacji webowych;
- projektowania, programowania i testowania aplikacji desktopowych;
- projektowania, programowania i testowania aplikacji mobilnych.

Do wykonywania wymienionych zadań zawodowych niezbędne jest osiągnięcie efektów kształcenia określonych w podstawie programowej kształcenia w zawodzie technik programista.

Opracowany program nauczania uwzględnia wszystkie elementy podstawy programowej kształcenia w zawodzie. Treści nauczania wynikające z oczekiwanych efektów kształcenia realizowanych w kwalifikacji INF.03, które występują w kwalifikacji INF.04, nie są powtarzane (dotyczy to „Bezpieczeństwa i higieny pracy” oraz „Podstaw informatyki”).

Kształcenie zgodnie z opracowanym programem nauczania pozwoli na osiągnięcie uwzględnionych w podstawie programowej celów kształcenia.

III. KORELACJA MIĘDZYPRZEDMIOTOWA

1. PRZEDMIOTY ROZSZERZONE W ZAWODZIE TECHNIK PROGRAMISTA

Zgodnie z rozporządzeniem MEN w sprawie ramowych planów z dnia 7 lutego 2012 r. w szkołach publicznych uczeń technikum wybiera dwa przedmioty na poziomie rozszerzonym oraz jeden przedmiot uzupełniający.

W programie nauczania dla zawodu technik programista zaleca się, aby przedmiotami realizowanymi w zakresie rozszerzonym były: matematyka i informatyka.

2. KORELACJA PROGRAMU NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA Z PODSTAWĄ PROGRAMOWĄ KSZTAŁCENIA OGÓLNEGO

Program nauczania dla zawodu technik programista uwzględnia aktualny stan wiedzy o zawodzie ze szczególnym zwróceniem uwagi na nowe technologie i osiągnięcia w zakresie programowania aplikacji.

W programie nauczania dla zawodu technik programista zostały uwzględnione powiązania z kształceniem ogólnym polegające na wcześniejszym osiągnięciu efektów kształcenia w zakresie przedmiotów ogólnokształcących, stanowiących podbudowę dla kształcenia w zawodzie. Dotyczy to przede wszystkim takich przedmiotów jak: matematyka, fizyka i informatyka oraz podstawy przedsiębiorczości i edukacji dla bezpieczeństwa.

IV. PLANY NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA

Zgodnie z rozporządzeniem MEN z dnia 7 lutego 2012 r. w sprawie ramowych planów nauczania w szkołach publicznych minimalny wymiar godzin na kształcenie zawodowe wynosi **1470 godzin**, z czego zarówno na kształcenie zawodowe teoretyczne, jak i praktyczne przypada minimum **735 godzin**.

W podstawie programowej dla zawodu technik programista (rozporządzenie MEN z dnia 16 maja 2019 r. w sprawie podstaw programowych kształcenia w zawodach szkolnictwa branżowego oraz dodatkowych umiejętności zawodowych w zakresie wybranych zawodów szkolnictwa branżowego) minimalna liczba godzin na kształcenie zawodowe została określona dla efektów kształcenia i wynosi:

540 godzin na realizację kwalifikacji *INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych*,

810 godzin na realizację kwalifikacji *INF.04. Projektowanie, programowanie i testowanie aplikacji*.

Dodatkowo rozporządzenie MEN z dnia 16 maja 2019 r. w sprawie zawodów szkolnictwa branżowego mówi, że:

„Uczniowie, którzy rozpoczną kształcenie w roku szkolnym 2019/2020 w klasie I dotychczasowego czteroletniego technikum albo w klasie I branżowej szkoły I stopnia w oddziale dla uczniów będących absolwentami dotychczasowego gimnazjum, realizują również efekty kształcenia wspólne dla wszystkich zawodów, w zakresie podejmowania i prowadzenia działalności gospodarczej (PDG), określone w części II załącznika do rozporządzenia Ministra Edukacji Narodowej z dnia 31 marca 2017 r. w sprawie podstawy programowej kształcenia w zawodach” (Dz. U. z poz. 860 oraz z 2018 r., poz. 844).

W związku z tym w planie nauczania dla zawodu technik programista został uwzględniony przedmiot PDG realizowany w wymiarze 30 godz. w cyklu nauczania.

1. Minimalna liczba godzin kształcenia zawodowego w kwalifikacjach wyodrębnionych w zawodzie

INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych	
Nazwa jednostki efektów kształcenia	Liczba godzin
INF.03.1. Bezpieczeństwo i higiena pracy	30
INF.03.2. Podstawy informatyki	30
INF.03.3. Projektowanie i tworzenie stron internetowych	90

INF.03.4. Projektowanie i administrowanie bazami danych	150
INF.03.5. Programowanie aplikacji internetowych	210
INF.03.6. Język obcy zawodowy	30
Razem	540

INF.04. Projektowanie, programowanie i testowanie aplikacji	
Nazwa jednostki efektów kształcenia	Liczba godzin
INF.04.1. Bezpieczeństwo i higiena pracy	15
INF.04.2. Podstawy informatyki	30
INF.04.3. Projektowanie oprogramowania	75
INF.04.4. Programowanie obiektowe	150
INF.04.5. Programowanie aplikacji desktopowych	120
INF.04.6. Programowanie aplikacji mobilnych	150
INF.04.7. Programowanie aplikacji zaawansowanych webowych	150
INF.04.8. Testowanie i dokumentowanie aplikacji	60
INF.04.9. Język obcy zawodowy	60
Razem	810

2. Plan nauczania dla zawodu technik programista o strukturze przedmiotowej

Plan nauczania dla zawodu technik programista jest tworzony zgodnie z ramowym planem nauczania (rozporządzenie MEN z dnia 7 lutego 2012 r. w sprawie ramowych planów nauczania w szkołach publicznych) i zakłada, że obowiązkowe zajęcia edukacyjne z zakresu kształcenia zawodowego zakończą się po pierwszym semestrze klasy czwartej.

2.1. Plan nauczania dla technika programisty — zajęcia edukacyjne z zakresu kształcenia zawodowego kończą się po pierwszym semestrze klasy IV

Lp.	Obowiązkowe zajęcia edukacyjne	Klasa								Liczba godzin tygodniowo w czteroletnim okresie nauczania	Liczba godzin w czteroletnim okresie nauczania
		I		II		III		IV			
		I semestr	II semestr	I semestr	II semestr	I semestr	II semestr	I semestr	II semestr		
Przedmioty ogólnokształcące											
1	Język polski	3	3	3	3	2	2	3	5	12	360
2	Język obcy nowożytny	3	3	2	2	2	2	2	4	10	300
3	Drugi język obcy nowożytny	1	1	1	1	1	1	2	2	5	150

4	Wiedza o kulturze	1	1							1	30
5	Historia	1	1	1	1					2	60
6	Wiedza o społeczeństwie	1	1							1	30
7	Podstawy przedsiębiorczości	1	1	1	1					2	60
8	Geografia	1	1							1	30
9	Biologia	1	1							1	30
10	Chemia	1	1							1	30
11	Fizyka	1	1							1	30
12	Matematyka	3	3	2	2	2	2	2	4	10	300
13	Informatyka	1	1							1	30
14	Wychowanie fizyczne	3	3	3	3	3	3	3	3	12	360
15	Edukacja dla bezpieczeństwa	1	1							1	30
16	Zajęcia z wychowawcą	1	1	1	1	1	1	1	1	4	120
Łączna liczba godzin		24	24	14	14	11	11	13	19	65	1950
Przedmioty realizowane w zakresie rozszerzonym oraz uzupełniające											
1	Przedmiot w zakresie rozszerzonym_1			2	2	3	3	2	4	8	240
2	Przedmiot w zakresie rozszerzonym_2			2	2	2	2	1	3	6	180
3	Przedmiot uzupełniający			1	1	1	1	1	3	4	120
Łączna liczba godzin		0	0	5	5	6	6	4	10	18	540
Przedmioty w kształceniu zawodowym teoretycznym											
1	J. angielski zawodowy			1	1					1	30
2	Bezpieczeństwo i higiena pracy			1	1					1	30
3	Podstawy informatyki	1	1							1	30
4	Projektowanie i tworzenie stron internetowych	2	2							2	60
5	Projektowanie i administrowanie bazami danych	2	2	1	1					3	90
6	Programowanie aplikacji internetowych			2	2	1	1			3	90
1	J. angielski zawodowy					2	2			2	60
2	Projektowanie oprogramowania			2	2					2	60
3	Podstawy programowania			2	2					2	60
4	Programowanie obiektowe					2	2			2	60
5	Programowanie aplikacji mobilnych					2	2			2	60
6	Zaawansowane aplikacje webowe					2	2	2		3	90
7	Testowanie i dokumentowanie aplikacji							2		1	30
Łączna liczba godzin		5	5	9	9	9	9	4	0	25	750

Przedmioty w kształceniu zawodowym praktycznym											
	PDG								2	1	30
1	Pracownia stron internetowych	2	2							2	60
2	Pracownia baz danych	2	2	1	1					3	90
3	Pracownia aplikacji internetowych			3	3	2	2			5	150
1	Pracownia projektowania oprogramowania			1	1					1	30
2	Pracownia podstaw programowania			2	2					2	60
3	Pracownia programowania obiektowego					2	2	2		3	90
4	Pracownia programowania aplikacji mobilnych					2	2	2		3	90
5	Pracownia zaawansowanych aplikacji webowych					2	2	2		3	90
6	Pracownia testowania i dokumentowania aplikacji							4		2	60
Łączna liczba godzin		4	4	7	7	8	8	10	2	25	750
Łączna liczba godzin kształcenia zawodowego		9	9	16	16	17	17	14	2	50	1500
Tygodniowy wymiar godzin obowiązkowych zajęć edukacyjnych		33	33	35	35	34	34	31	31	133	3990
Wymagany tygodniowy wymiar godzin obowiązkowych zajęć edukacyjnych		33	35	34	31						

/1/ (do celów obliczeniowych przyjęto 30 tygodni w ciągu jednego roku szkolnego)

Egzamin potwierdzający pierwszą kwalifikację *INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych* odbywa się pod koniec 2. semestru klasy III.

Egzamin potwierdzający drugą kwalifikację *INF.04. Projektowanie, programowanie i testowanie aplikacji* odbywa się pod koniec 1. semestru klasy IV.

2.2. Praktyki zawodowe

Minimalny wymiar praktyk zawodowych:	Liczba tyg.	Liczba dni	Liczba godz. dziennie	Razem
klasa III	4	20	7	140
klasa IV	4	20	7	140
Razem				280

V. WYKAZ PRZEDMIOTÓW WYODRĘBNIONYCH W PROGRAMIE NAUCZANIA DLA ZAWODU TECHNIK PROGRAMISTA

<i>Nazwa przedmiotu</i>	<i>Liczba godzin dla przedmiotu</i>
Podstawy informatyki	30
BHP	30
Język angielski zawodowy	90
Projektowanie i tworzenie stron internetowych	60
Projektowanie i administrowanie bazami danych	90
Programowanie aplikacji internetowych	90
Pracownia stron internetowych	60
Pracownia baz danych	90
Pracownia aplikacji internetowych	150
Projektowanie oprogramowania	60
Podstawy programowania	60
Programowanie obiektowe	90
Programowanie aplikacji mobilnych	60
Zaawansowane aplikacje webowe	60
Testowanie i dokumentowanie aplikacji	30
Pracownia projektowania oprogramowania	30
Pracownia podstaw programowania	60
Pracownia programowania obiektowego	120
Pracownia programowania aplikacji mobilnych	90
Pracownia zaawansowanych aplikacji webowych	120
Pracownia testowania i dokumentowania aplikacji	60

VI. PROGRAMY NAUCZANIA DLA POSZCZEGÓLNYCH PRZEDMIOTÓW W ZAWODZIE TECHNIK PROGRAMISTA

Do wykonywania zadań zawodowych niezbędne jest osiągnięcie niżej wymienionych zakładanych efektów kształcenia.

1. Programy nauczania określające kompetencje personalne i społeczne

Zaplanowana siatka godzin z poszczególnych kwalifikacji i przedmiotów jest dostosowana do wymiaru godzin określonego w przepisach w sprawie ramowych planów nauczania dla szkół publicznych, przewidzianego dla kształcenia zawodowego w danym typie szkoły. Siatka zachowuje minimalną liczbę godzin wskazanych w tabeli dla efektów kształcenia właściwych dla kwalifikacji wyodrębnionej w zawodzie.

Nauczyciele wszystkich obowiązkowych zajęć edukacyjnych z zakresu kształcenia zawodowego powinni stwarzać uczniom warunki do nabywania określonych w podstawie programowej kształcenia w zawodzie kompetencji personalnych i społecznych oraz umiejętności w zakresie organizacji pracy małych zespołów.

1.1. Kompetencje personalne i społeczne

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
---	---	--------------------------------------	------------------------

Uczeń:	Uczeń:	Uczeń:	
1) przestrzega zasad kultury osobistej i etyki zawodowej	1) stosuje zasady kultury osobistej i przestrzega ogólnie przyjętych norm zachowania w środowisku pracy 2) wyjaśnia pojęcie tajemnicy zawodowej 3) przyjmuje odpowiedzialność za powierzone informacje zawodowe 4) respektuje zasady dotyczące przestrzegania tajemnicy zawodowej 5) wyjaśnia, na czym polega zachowanie etyczne w zawodzie	1) zachowywać się odpowiednio do sytuacji, zgodnie z zasadami kultury osobistej i normami zachowania 2) wyjaśnić pojęcia: <ul style="list-style-type: none"> • tajemnicy zawodowej • etyki zawodowej 3) przestrzegać tajemnicy zawodowej 4) zachowywać się zgodnie z zasadami etyki zawodowej 5) ponosić odpowiedzialność za powierzone informacje zawodowe	
2) planuje wykonanie zadania	1) omawia czynności realizowane w ramach czasu pracy 2) określa czas realizacji zadań 3) realizuje działania w wyznaczonym czasie 4) monitoruje realizację zaplanowanych działań 5) dokonuje modyfikacji zaplanowanych działań 6) dokonuje samooceny wykonanej pracy	1) zaplanować realizację zadania zawodowego w czasie 2) omówić zaplanowane działania zawodowe 3) wykonać zaplanowane zadania zawodowe w wyznaczonym czasie 4) nadzorować wykonywanie zaplanowanych działań zawodowych 5) w razie potrzeby wprowadzać zmiany do realizacji zaplanowanych zadań 6) samokrytycznie oceniać wykonanie zaplanowanego działania	
3) ponosi odpowiedzialność za podejmowane działania	1) przewiduje skutki podejmowanych działań, w tym prawne 2) wykazuje świadomość odpowiedzialności za wykonywaną pracę 3) ocenia podejmowane działania	1) określić konsekwencje wykonywanych prac i zadań zawodowych 2) świadomie ponosić odpowiedzialność za realizację zadań zawodowych 3) wyrazić swoją opinię na temat	

	4) przewiduje konsekwencje niewłaściwego wykonywania czynności zawodowych na stanowisku pracy, w tym posługiwania się niebezpiecznymi substancjami i niewłaściwej eksploatacji maszyn i urządzeń na stanowisku pracy	<p>podejmowanych działań zawodowych</p> <p>4) przewidzieć skutki podejmowanych działań zawodowych w związku z:</p> <ul style="list-style-type: none"> • wykorzystaniem na stanowisku pracy niebezpiecznych substancji • niewłaściwą eksploatacją maszyn i urządzeń 	
4) wykazuje się kreatywnością i otwartością na zmiany	<ol style="list-style-type: none"> 1) podaje przykłady wpływu zmiany na różne sytuacje życia społecznego i gospodarczego 2) wskazuje przykłady wprowadzenia zmiany i ocenia skutki jej wprowadzenia 3) proponuje sposoby rozwiązywania problemów związanych z wykonywaniem zadań zawodowych w nieprzewidywalnych warunkach 	<ol style="list-style-type: none"> 1) być otwartym na zmiany, innowacyjne rozwiązania, kreatywne pomysły, oryginalne zastosowania 2) rozwiązywać twórczo problemy dotyczące realizowanych zadań zawodowych 3) wprowadzać kreatywne zmiany do realizowanych zadań zawodowych 4) przewidywać rezultaty wprowadzenia wybranych przez siebie zmian i rozwiązań w wykonywanych zadaniach zawodowych 	
5) stosuje techniki radzenia sobie ze stresem	<ol style="list-style-type: none"> 1) rozpoznaje źródła stresu podczas wykonywania zadań zawodowych 2) wybiera techniki radzenia sobie ze stresem odpowiednio do sytuacji 3) wskazuje najczęstsze przyczyny sytuacji stresowych w pracy zawodowej 4) przedstawia różne formy zachowań asertywnych jako sposobów radzenia sobie ze stresem 5) wyraża swoje emocje, uczucia i poglądy zgodnie z ogólnie przyjętymi normami i zasadami współżycia społecznego 6) rozróżnia techniki rozwiązywania konfliktów związanych z wykonywaniem zadań 	<ol style="list-style-type: none"> 1) rozpoznać źródła stresu związane ze środowiskiem pracy 2) omówić przyczyny i skutki stresu pojawiające się podczas realizacji czynności zawodowych 3) radzić sobie ze stresem podczas pracy zawodowej 4) stosować asertywne zachowania podczas wykonywania czynności zawodowych 5) wymienić różne techniki rozwiązywania konfliktów występujących podczas pracy zawodowej 6) wyrażać swoje emocje, uczucia, opinie z zachowaniem zasad i norm społecznie przyjętych 	

	zawodowych		
	7) określa skutki stresu		
6) doskonalą umiejętności zawodowe	<ol style="list-style-type: none"> 1) pozyskuje informacje zawodowe dotyczące przemysłu z różnych źródeł 2) określa zakres umiejętności i kompetencji niezbędnych w wykonywaniu zawodu 3) analizuje własne kompetencje 4) wyznacza własne cele rozwoju zawodowego 5) planuje drogę rozwoju zawodowego 4) wskazuje możliwości podnoszenia poziomu kompetencji zawodowych, osobistych i społecznych 	<ol style="list-style-type: none"> 1) wyszukiwać w różnych źródłach informacje związane z wykonywanym zawodem 2) nazwać umiejętności i kompetencje zawodowe 3) określić własne umiejętności i kompetencje zawodowe 4) sprecyzować własne cele rozwoju zawodowego 5) zaplanować własną drogę rozwoju zawodowego 6) określić możliwości rozwoju własnych kompetencji zawodowych, osobistych, społecznych 	
7) stosuje zasady komunikacji interpersonalnej	<ol style="list-style-type: none"> 1) identyfikuje sygnały werbalne i niewerbalne 2) stosuje aktywne metody słuchania 3) prowadzi dyskusje 4) udziela informacji zwrotnej 5) wyraża swoje emocje, uczucia i poglądy zgodnie z przyjętymi normami i zasadami współżycia społecznego 	<ol style="list-style-type: none"> 1) rozpoznać sygnały werbalne i niewerbalne 2) stosować techniki aktywnego słuchania 3) zadawać pytania i odpowiadać na pytania 4) brać aktywny udział w dyskusji 5) wyrażać swoje zdanie, uczucia, emocje zgodnie z przyjętymi normami i zasadami współżycia społecznego 	
8) negocjuje warunki porozumień	<ol style="list-style-type: none"> 1) charakteryzuje pożądaną postawę osoby podczas prowadzenia negocjacji 2) wskazuje sposób prowadzenia negocjacji warunków porozumienia 	<ol style="list-style-type: none"> 1) określić odpowiednią postawę osoby podczas prowadzenia negocjacji 2) określić odpowiedni sposób prowadzenia negocjacji 3) negocjować umiejętnie warunki porozumienia 	
9) stosuje metody i techniki rozwiązywania problemów	<ol style="list-style-type: none"> 1) opisuje sposób przeciwdziałania problemom w zespole realizującym zadania 	<ol style="list-style-type: none"> 1) określić odpowiedni sposób przeciwdziałania problemom podczas realizacji zadań 	

	<ol style="list-style-type: none"> 2) opisuje techniki rozwiązywania problemów 3) wskazuje, na wybranym przykładzie, metody i techniki rozwiązywania problemu 	<p>zawodowych</p> <ol style="list-style-type: none"> 2) określić odpowiedni sposób rozwiązywania problemów podczas realizacji zadań zawodowych 3) omówić metody i techniki rozwiązywania problemów na wybranym przykładzie 	
10) współpracuje w zespole	<ol style="list-style-type: none"> 1) pracuje w zespole, ponosząc odpowiedzialność za wspólnie realizowane zadania 2) przestrzega podziału ról, zadań i odpowiedzialności w zespole 3) angażuje się w realizację wspólnych działań zespołu 4) modyfikuje sposób zachowania, uwzględniając stanowisko wypracowane wspólnie z innymi członkami zespołu 	<ol style="list-style-type: none"> 1) współpracować aktywnie w zespole podczas realizacji wspólnych zadań zawodowych 2) ponosić odpowiedzialność za wspólnie realizowane w zespole zadania zawodowe: <ul style="list-style-type: none"> • przestrzegać podziału ról • przestrzegać wykonywania zadań • przewidywać skutki działań 3) dostosować swoje zachowanie i działanie w celu współpracy z innymi członkami zespołu 	

1.2. Organizacja pracy małych zespołów

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) organizuje pracę zespołu w celu wykonania przydzielonych zadań	<ol style="list-style-type: none"> 1) określa strukturę grupy 2) przygotowuje zadania zespołu do realizacji 3) planuje realizację zadań zapobiegających zagrożeniom bezpieczeństwa i ochrony zdrowia 4) oszacowuje czas potrzebny na realizację określonego zadania 	<ol style="list-style-type: none"> 1) stworzyć zespół do wykonania określonego zadania 2) przygotować zespół do wykonania określonego zadania 3) zaplanować realizację zadania w określonym czasie i formie, zapobiegając zagrożeniom bezpieczeństwa i ochrony zdrowia 4) stworzyć harmonogram prac zespołu dla 	

	<ol style="list-style-type: none"> 5) komunikuje się ze współpracownikami 6) wskazuje wzorce prawidłowej współpracy w grupie 7) przydziela zadania członkom zespołu zgodnie z harmonogramem planowanych prac 	<p>poszczególnych jego członków</p> <ol style="list-style-type: none"> 5) porozumieć się z poszczególnymi członkami zespołu w celu wykonania powierzonego zadania zawodowego 6) określić wzory prawidłowej współpracy w zespole 7) przydzielić poszczególnym członkom zespołu określone zadania zawodowe do wykonania, zgodnie z harmonogramem prac zespołu 8) podsumować i zakończyć pracę zespołu 	
2) dobiera osoby do wykonania przydzielonych zadań	<ol style="list-style-type: none"> 1) ocenia przydatność poszczególnych członków zespołu do wykonania zadania 2) rozdziela zadania według umiejętności i kompetencji członków zespołu 	<ol style="list-style-type: none"> 1) określić przydatność poszczególnych członków zespołu do wykonania określonego zadania zawodowego, uwzględniając: <ul style="list-style-type: none"> • umiejętności członków zespołu • kompetencje członków zespołu 2) przydzielić określone funkcje i zadania poszczególnym członkom zespołu z uwzględnieniem: <ul style="list-style-type: none"> • umiejętności członków zespołu • kompetencji członków zespołu 	
3) kieruje wykonaniem przydzielonych zadań	<ol style="list-style-type: none"> 1) ustala kolejność wykonywania zadań zgodnie z harmonogramem prac 2) formułuje zasady wzajemnej pomocy 3) koordynuje realizację zadań zapobiegających zagrożeniom bezpieczeństwa i ochrony zdrowia 4) wydaje dyspozycje osobom wykonującym poszczególne zadania 5) monitoruje proces wykonywania zadań 	<ol style="list-style-type: none"> 1) określić kolejność wykonywania czynności w realizowanym zadaniu zawodowym według przyjętego harmonogramu prac zespołu 2) sprecyzować zasady współpracy i wzajemnej pomocy w zespole podczas wykonywania zadania zawodowego 3) zarządzać wykonywaniem zadań zawodowych, zapobiegając zagrożeniom bezpieczeństwa i ochrony zdrowia 4) komunikować polecenia wykonania danej 	

	6) opracowuje dokumentację dotyczącą realizacji zadania według obowiązujących standardów	czynności poszczególnym członkom zespołu 5) analizować przebieg wykonywania zadania zawodowego przez zespół 6) wykonać opracowanie dokumentacji realizowanego przez zespół zadania zawodowego	
4) ocenia jakość wykonania przydzielonych zadań	1) kontroluje efekty pracy zespołu 2) ocenia pracę poszczególnych członków zespołu pod kątem zgodności z warunkami technicznymi odbioru prac 3) udziela wskazówek w celu prawidłowego wykonania przydzielonych zadań	1) weryfikować efekty pracy całego zespołu oraz poszczególnych jego członków 2) opiniować jakość pracy całego zespołu oraz poszczególnych jego członków 3) sugerować możliwe rozwiązania, zapewniające prawidłowe wykonanie powierzonych zadań zawodowych	
5) wprowadza rozwiązania techniczne i organizacyjne wpływające na poprawę warunków i jakości pracy	1) dokonuje analizy rozwiązań technicznych, warunków organizacyjnych i jakości pracy 2) proponuje rozwiązania techniczne i organizacyjne mające na celu poprawę warunków i jakości pracy	1) analizować rozwiązania techniczne, organizacyjne i jakościowe 2) sugerować rozwiązania techniczne, organizacyjne i jakościowe	

2. Programy nauczania dla kwalifikacji INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych

2.1. Język angielski zawodowy

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		

<p>1) uczeń posługuje się podstawowym zasobem środków językowych w języku obcym nowożytnym (ze szczególnym uwzględnieniem środków leksykalnych) umożliwiającym realizację czynności zawodowych w zakresie tematów związanych:</p> <p>a) ze stanowiskiem pracy i jego wyposażeniem</p> <p>b) z głównymi technologiami stosowanymi w danym zawodzie</p> <p>c) z dokumentacją związaną z danym zawodem</p> <p>d) z usługami świadczonymi w danym zawodzie</p>	<p>1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</p> <p>a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy</p> <p>b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych</p> <p>c) procesów i procedur związanych z realizacją zadań zawodowych</p> <p>d) formularzy, specyfikacji, dokumentacji technicznej oraz innych dokumentów związanych z wykonywaniem zadań zawodowych</p> <p>e) świadczonych usług, w tym obsługi klienta</p>	<p>1) posługiwać się terminologią związaną z branżą IT potrzebną do wykonania określonych zadań zawodowych zgodnie z zasadami bezpieczeństwa i higieny pracy</p> <p>2) przeanalizować obcojęzyczną dokumentację w celu realizacji zadań zawodowych</p> <p>3) zrozumieć i wyjaśnić obcojęzyczne procedury i polecenia związane z obsługą programów i urządzeń IT</p> <p>4) porozumieć się z klientem oraz współpracownikami w języku obcym zawodowym</p>	
<p>2) rozumie proste wypowiedzi ustne artykułowane wyraźnie, w standardowej odmianie języka obcego nowożytnego, a także proste wypowiedzi pisemne w języku obcym nowożytnym, w zakresie umożliwiającym realizację zadań zawodowych:</p> <p>a) rozumie proste wypowiedzi ustne dotyczące czynności zawodowych (np. rozmowy, wiadomości, komunikaty, instrukcje lub filmy instruktażowe, prezentacje), artykułowane wyraźnie, w standardowej odmianie języka</p> <p>b) rozumie proste wypowiedzi</p>	<p>1) określa główną myśl wypowiedzi lub tekstu, ewentualnie fragmentu wypowiedzi lub tekstu</p> <p>2) znajduje w wypowiedzi lub tekście określone informacje</p> <p>3) rozpoznaje związki między poszczególnymi częściami tekstu</p> <p>4) układa informacje w określonym porządku</p>	<p>1) zrozumieć obcojęzyczną wypowiedź ustną lub pisemną związaną z wykonywanym zadaniem zawodowym</p> <p>2) wyszukać w obcojęzycznej wypowiedzi ustnej lub pisemnej informacje istotne pod kątem wykonywania czynności zawodowych</p> <p>3) powiązać ze sobą informacje zawarte w poszczególnych częściach obcojęzycznej wypowiedzi ustnej lub pisemnej</p> <p>4) uporządkować informacje zawarte w obcojęzycznej wypowiedzi ustnej lub pisemnej</p>	

<p>pisemne dotyczące czynności zawodowych (np. napisy, broszury, instrukcje obsługi, przewodniki, dokumentację zawodową)</p>			
<p>3) samodzielnie tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne i pisemne w języku obcym nowożytnym, w zakresie umożliwiającym realizację zadań zawodowych:</p> <p>a) tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne dotyczące czynności zawodowych (np. polecenie, komunikat, instrukcję)</p> <p>b) tworzy krótkie, proste, spójne i logiczne wypowiedzi pisemne dotyczące czynności zawodowych (np. komunikat, e-mail, instrukcję, wiadomość, CV, list motywacyjny, dokument związany z wykonywanym zawodem — wg wzoru)</p>	<p>1) opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi</p> <p>2) przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</p> <p>3) wyraża i uzasadnia swoje stanowisko</p> <p>4) stosuje zasady konstruowania tekstów o różnym charakterze</p> <p>5) stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji</p>	<p>1) opisać w języku obcym przedmioty, działania i zjawiska związane z czynnościami zawodowymi</p> <p>2) posłużyć się językiem obcym do udzielenia instrukcji, wydawania poleceń, formułowania komunikatów słownych i pisemnych w środowisku pracy</p> <p>3) sformułować krótką wypowiedź ustną na temat związany z wykonywaniem czynności zawodowych</p> <p>4) sporządzić krótką wypowiedź pisemną na temat związany z wykonywaniem czynności zawodowych</p> <p>5) zredagować w języku obcym oficjalny dokument umożliwiający komunikowanie się w środowisku pracy</p> <p>6) wyrazić i uzasadnić w formie pisemnej swoje stanowisko za pomocą języka obcego</p>	
<p>4) uczestniczy w rozmowie w typowych sytuacjach związanych z realizacją zadań zawodowych — reaguje w języku obcym nowożytnym w sposób zrozumiały, adekwatnie do sytuacji komunikacyjnej, ustnie lub w formie prostego tekstu:</p> <p>a) reaguje ustnie (np. podczas rozmowy z innym pracownikiem,</p>	<p>1) rozpoczyna, prowadzi i kończy rozmowę</p> <p>2) uzyskuje i przekazuje informacje i wyjaśnienia</p> <p>3) wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób</p> <p>4) prowadzi proste negocjacje związane z czynnościami zawodowymi</p> <p>5) proponuje, zachęca</p>	<p>1) przeprowadzić rozmowę w języku obcym branżowym w sytuacji związanej z wykonywaniem określonego zadania zawodowego</p> <p>2) zdobyć, wyjaśnić i przekazać informację dotyczącą realizacji prac w zawodzie</p> <p>3) brać aktywny udział w dyskusji prowadzonej w języku obcym branżowym:</p> <ul style="list-style-type: none"> • zaprezentować swoje zdanie 	

<p>klientem, kontrahentem, w tym podczas rozmowy telefonicznej) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych</p> <p>b) reaguje w formie prostego tekstu pisanego (np. wiadomości, formularza, e-maila, dokumentu związanego z wykonywanym zawodem) w typowych sytuacjach towarzyszących wykonywaniu czynności zawodowych</p>	<p>6) stosuje zwroty i formy grzecznościowe</p> <p>7) dostosowuje styl wypowiedzi do sytuacji</p>	<ul style="list-style-type: none"> • podać swoją argumentację • wyrazić poparcie lub jego brak dla innych rozmówców <p>4) negocjować warunki realizacji określonych zadań zawodowych w języku obcym branżowym</p> <p>5) konstruować wypowiedzi promujące i zachęcające w języku obcym</p> <p>6) zastosować zwroty grzecznościowe w rozmowach i korespondencji branżowej</p> <p>7) dostosowywać styl wypowiedzi w języku obcym do sytuacji w środowisku pracy</p>	
<p>5) zmienia formę przekazu ustnego lub pisemnego w języku obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych:</p> <p>a) przetwarza tekst ustnie lub pisemnie w typowych sytuacjach związanych z wykonywaniem czynności zawodowych</p>	<p>1) przekazuje w języku obcym nowożytnym informacje zawarte w materiałach wizualnych (np. w wykresach, symbolach, piktogramach, schematach) oraz audiowizualnych (np. w filmach instruktażowych)</p> <p>2) przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym</p> <p>3) przekazuje w języku obcym nowożytnym informacje sformułowane w tym języku lub w języku polskim</p> <p>4) przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację</p>	<p>1) przedstawić w języku obcym branżowym informacje zawarte w materiale wizualnym w formie wykresów, schematów, filmów, slajdów</p> <p>2) wyjaśnić w języku polskim informacje zaprezentowane w materiale w języku obcym branżowym</p> <p>3) wyjaśnić w języku obcym branżowym informacje zaprezentowane w materiale w języku polskim</p> <p>4) zaprezentować na forum w języku obcym branżowym opracowany przez siebie materiał przedstawiony w formie multimedialnej</p>	
<p>6) wykorzystuje strategie służące doskonaleniu własnych umiejętności językowych oraz podnoszące świadomość językową:</p>	<p>1) korzysta ze słownika dwujęzycznego i jednojęzycznego</p> <p>2) współdziała z innymi osobami, realizując zadania językowe</p>	<p>1) posługiwać się słownikami, translatorami, leksykonami celem wzbogacenia swojego zasobu słów w języku obcym branżowym</p> <p>2) współpracować i realizować w zespole zadania językowe związane z branżą IT</p> <p>3) korzystać z obcojęzycznych branżowych</p>	

<p>a) wykorzystuje techniki samodzielnej nauki języka</p> <p>b) współdziała w grupie</p> <p>c) korzysta ze źródeł informacji w języku obcym nowożytnym</p> <p>d) stosuje strategie komunikacyjne i kompensacyjne</p>	<p>3) korzysta z tekstów w języku obcym, również za pomocą technologii informacyjno-komunikacyjnych</p> <p>4) identyfikuje słowa kluczowe, internacjonalizmy</p> <p>5) wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa</p> <p>6) upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznanne słowa innymi, wykorzystuje opis, środki niewerbalne</p>	<p>czasopism, katalogów, zasobów cyfrowych i stron internetowych</p> <p>4) rozumieć z kontekstu znaczenie czytanych tekstów, artykułów, opracowań branżowych</p> <p>5) wyjaśniać własnymi słowami znaczenie czytanych tekstów, artykułów, opracowań branżowych</p>	
--	--	--	--

Planowane zadania

Zadanie 1.

- Przygotuj następujące dokumenty aplikacyjne w języku angielskim: CV, list motywacyjny.
- Napisz w języku angielskim oficjalny e-mail z ofertą promocyjną usługi wykonania aplikacji internetowej.

Zadanie 2.

- Przetłumacz z języka polskiego na język angielski instrukcję instalacji wybranego oprogramowania systemowego lub użytkowego.
- Przetłumacz z języka angielskiego na język polski raport walidacji kodu HTML i CSS witryny internetowej.

Zadanie 3.

Wykonaj samodzielnie lub w małym zespole 2 – 3-osobowym prezentację multimedialną w języku angielskim poświęconą tematyce nowoczesnych technologii i zaprezentuj ją na forum klasy.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Język angielski zawodowy* w części teoretycznej powinny odbywać się w sali przedmiotowej do nauki języków obcych zawodowych, wyposażonej w: komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, telewizor, magnetofon, słowniki, materiały autentyczne, materiały audiowizualne, podręczniki do nauki języka obcego zawodowego, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy i umiejętności językowych z zakresu przedmiotu, tj. rozumienia wypowiedzi i redagowania tekstów w języku obcym

branżowym, tłumaczenia z angielskiego na polski i odwrotnie instrukcji, procedur, poleceń związanych z narzędziami IT, posługiwania się językiem angielskim w sytuacjach związanych z zawodem.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz treści nauczania, np. słowniki, książki do nauki języka obcego, prezentacje multimedialne, filmy dydaktyczne, zestawy ćwiczeń, instrukcje obsługi, pomoce wizualne, artykuły i katalogi w języku angielskim, płyty CD/DVD, karty samooceny. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i komunikacji, np. wykład słowny wsparty ćwiczeniami językowymi i materiałami multimedialnymi, dryle językowe, odgrywanie sytuacji komunikacyjnych, pozyskiwanie informacji ogólnych i szczegółowych z tekstów pisanych, nagrań audio oraz filmów, dyskusja dydaktyczna, odgrywanie ról, wyszukiwanie informacji na branżowych stronach internetowych. Wszystkie metody powinny pomagać w zainteresowaniu uczniów tematem poruszonym na zajęciach dydaktycznych, zachęcić do komunikacji w języku obcym, angażować w proces poznawania i utrwalania nowych słówek, doskonalenia umiejętności komunikowania się w języku angielskim.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych. Część efektów kształcenia powinna być osiągnięta w korelacji z kształceniem zawodowym praktycznym oraz językiem obcym ogólnokształcącym.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod i narzędzi weryfikacji przyrostu wiedzy i umiejętności komunikacyjnych w języku obcym, np. odpowiedzi ustnych, dialogów, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji pracy ucznia, prezentacji na forum klasy. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu. Szczególną uwagę należy zwrócić na to, czy uczeń: rozpoznaje i stosuje różne formy wypowiedzi w języku obcym, ma odpowiedni zasób słownictwa, poprawnie formułuje wypowiedzi w języku angielskim.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o różne źródła wiedzy (podręcznik, artykuły, materiały elektroniczne) oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.2. Bezpieczeństwo i higiena pracy

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) określa warunki i organizację pracy zapewniające wymagany poziom ochrony zdrowia i życia przed zagrożeniami występującymi w środowisku pracy	1) wymienia obowiązujące przepisy prawa związane z bezpieczeństwem i higieną pracy, ochroną przeciwpożarową, ochroną środowiska i ergonomią 2) wymienia wewnętrzne przepisy prawa związane z bezpieczeństwem i higieną pracy, ochroną przeciwpożarową, ochroną środowiska i ergonomią 3) stosuje wewnętrzne przepisy prawa związane z bezpieczeństwem i higieną pracy, ochroną przeciwpożarową, ochroną środowiska i ergonomią 4) opisuje wymagania dotyczące ergonomii pracy w branży informatycznej	1) omówić obowiązujące przepisy prawa regulujące warunki bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej oraz wymagania dotyczące ergonomii pracy 2) określić wewnętrzne przepisy prawa regulujące warunki bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej oraz wymagania ergonomii pracy 3) posłużyć się przepisami z zakresu bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej, ochrony środowiska oraz ergonomii pracy 4) charakteryzować zasady ergonomicznej postawy ciała i organizacji miejsca pracy osoby pracującej w branży informatycznej	
2) określa zadania i uprawnienia instytucji oraz służb działających w zakresie ochrony pracy i ochrony środowiska w Rzeczypospolitej Polskiej	1) wymienia zadania i uprawnienia w zakresie ochrony pracy 2) wymienia zadania i uprawnienia w zakresie ochrony środowiska naturalnego 3) wymienia zadania i uprawnienia w zakresie ochrony środowiska pracy 4) wymienia zadania i uprawnienia w zakresie stosowania maszyn i urządzeń	1) nazwać instytucje i służby działające w zakresie ochrony pracy i ochrony środowiska w Polsce 2) określić zadania instytucji i służb działających w zakresie ochrony pracy i ochrony środowiska w Polsce 3) określić uprawnienia i kompetencje instytucji i służb działających w zakresie ochrony pracy i ochrony środowiska w Polsce 4) omówić zadania instytucji i służb działających na rzecz ochrony pracy i ochrony środowiska w Polsce w zakresie	

		stosowania maszyn i urządzeń 5) wymienić uprawnienia instytucji i służb działających na rzecz ochrony pracy i ochrony środowiska w Polsce w zakresie stosowania maszyn i urządzeń	
3) określa prawa i obowiązki pracownika oraz pracodawcy w zakresie bezpieczeństwa i higieny pracy	<ol style="list-style-type: none"> 1) wymienia obowiązki pracodawcy w zakresie bezpieczeństwa i higieny pracy 2) wskazuje obowiązki pracowników w zakresie bezpieczeństwa i higieny pracy 3) wskazuje prawa i obowiązki pracownika, który uległ wypadkowi przy pracy, wynikające z przepisów prawa 4) wskazuje rodzaje świadczeń z tytułu wypadku przy pracy 5) wskazuje prawa i obowiązki pracownika, który zachorował na chorobę zawodową, wynikające z przepisów prawa 6) wskazuje rodzaje świadczeń z tytułu choroby zawodowej 	<ol style="list-style-type: none"> 1) scharakteryzować prawa i obowiązki pracodawcy w zakresie bezpieczeństwa i higieny pracy 2) scharakteryzować prawa i obowiązki pracownika w zakresie bezpieczeństwa i higieny pracy 3) określić konsekwencje nieprzestrzegania praw i obowiązków pracodawcy i pracownika w zakresie bezpieczeństwa i higieny pracy 4) omówić prawa i obowiązki pracownika podczas wypadku przy pracy 5) scharakteryzować rodzaje świadczeń z tytułu wypadku przy pracy 6) omówić prawa i obowiązki pracownika, u którego stwierdzono chorobę zawodową 7) scharakteryzować rodzaje świadczeń z tytułu choroby zawodowej 	
4) określa zagrożenia związane z występowaniem różnych czynników w środowisku pracy	<ol style="list-style-type: none"> 1) stosuje klasyfikację czynników występujących w środowisku pracy 2) określa skutki występowania różnych czynników w środowisku pracy 3) ocenia wpływ czynników szkodliwych na zdrowie i bezpieczeństwo pracowników 4) dobiera środki ochrony indywidualnej w zależności od czynników występujących na 	<ol style="list-style-type: none"> 1) scharakteryzować czynniki tworzące środowisko pracy 2) omówić skutki oddziaływania czynników występujących w środowisku pracy na człowieka 3) scharakteryzować zagrożenia i wpływ czynników szkodliwych na bezpieczeństwo, zdrowie i życie człowieka podczas wykonywania pracy 	

	stanowisku pracy	4) wymienić środki ochrony i sposoby przeciwdziałania zagrożeniom występującym w danym środowisku pracy	
5) stosuje środki techniczne ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych	<ol style="list-style-type: none"> 1) określa zasady doboru środków ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych 2) charakteryzuje funkcje odzieży ochronnej 3) ocenia prawidłowość doboru środków ochrony indywidualnej i zbiorowej do wykonywanych zadań zawodowych 	<ol style="list-style-type: none"> 1) zdefiniować zasady stosowania i doboru środków ochrony indywidualnej i zbiorowej 2) scharakteryzować funkcje ochronne odzieży roboczej i środków ochrony indywidualnej 3) dobrać środki ochrony indywidualnej i zbiorowej do wykonania określonych zadań zawodowych 	
6) organizuje stanowisko pracy zgodnie z wymaganiami ergonomii, obowiązującymi przepisami i zasadami bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej i ochrony środowiska	<ol style="list-style-type: none"> 1) określa zasady ergonomicznej organizacji pracy i stanowisk pracy 2) przewiduje wpływ zmian wprowadzanych na poszczególnych etapach procesu produkcyjnego na poziom bezpieczeństwa i higieny pracy 3) określa metody eliminacji źródeł niebezpiecznych i szkodliwych czynników występujących w procesach pracy 4) organizuje działania zapobiegające powstawaniu pożaru lub innego zagrożenia w przedsiębiorstwie 5) organizuje wybrane stanowisko pracy zgodnie z wymogami ergonomii, przepisami bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej i ochrony środowiska 	<ol style="list-style-type: none"> 1) scharakteryzować zasady ergonomicznej organizacji pracy i stanowiska pracy 2) przewidzieć skutki własnych działań związanych z wykonywaniem pracy pod kątem zachowania zasad bezpieczeństwa i higieny pracy 3) wymienić metody likwidacji zagrożeń (mechanicznych, chemicznych, elektrycznych) wynikających z wykonywania zadań zawodowych 4) określić działania ochronne zapobiegające zdarzeniom zagrażającym życiu i zdrowiu człowieka podczas wykonywania zadań zawodowych 5) omówić organizację stanowiska pracy do wykonania określonych zadań zawodowych zgodnie z wymogami ergonomii, przepisami bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej i ochrony środowiska 	
7) udziela pierwszej pomocy w stanach zagrożenia zdrowia i życia	<ol style="list-style-type: none"> 1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowia i życia 2) ocenia sytuację poszkodowanego na 	<ol style="list-style-type: none"> 1) scharakteryzować organizację pierwszej pomocy w przypadkach nagłych 2) określić rodzaj pierwszej pomocy w 	

	<p>podstawie analizy obserwowanych u niego objawów</p> <ol style="list-style-type: none"> 3) zabezpiecza siebie, osobę poszkodowaną i miejsce wypadku 4) układa poszkodowanego w pozycji bezpiecznej 5) powiadamia odpowiednie służby 6) udziela pierwszej pomocy w urazowych stanach zagrożenia życia i zdrowia, takich jak krwotoki, zmiążdżenia, amputacje, złamania, oparzenia 7) udziela pierwszej pomocy w nieurazowych stanach zagrożenia życia i zdrowia, takich jak omdlenie, zawał, udar 8) wykonuje na fantomie resuscytację krążeniowo-oddechową zgodnie z wytycznymi Polskiej i Europejskiej Rady Resuscytacji 	<p>zależności od urazu i objawów zaobserwowanych u poszkodowanego</p> <ol style="list-style-type: none"> 3) zaprezentować wykonanie ułożenia poszkodowanego w pozycji bocznej ustalonej 4) zaprezentować wykonanie resuscytacji krążeniowo-oddechowej 5) określić działania wykonywane w toku czynności udzielania pierwszej pomocy w urazowych stanach zagrożenia życia i zdrowia 6) określić działania wykonywane w toku czynności udzielania pierwszej pomocy w nieurazowych stanach zagrożenia życia i zdrowia 	
--	---	--	--

Planowane zadania

Zadanie 1.

- Określ podstawowe obowiązki pracownika w zakresie BHP i higieny pracy.
- Określ podstawowe obowiązki pracodawcy w zakresie BHP i higieny pracy.

Zadanie 2.

- Omów organizację stanowiska pracy zgodnie z wymogami ergonomii, przepisami bezpieczeństwa i higieny pracy oraz ochrony przeciwpożarowej i ochrony środowiska. Uwzględnij właściwą pozycję ciała na stanowisku pracy siedzącej.

Zadanie 3.

Scharakteryzuj organizację pierwszej pomocy w nagłych wypadkach, a następnie zaprezentuj wykonanie u osoby poszkodowanej resuscytacji krążeniowo-oddechowej zgodnie z wytycznymi Polskiej i Europejskiej Rady Resuscytacji. Przedstaw prawidłowe ułożenie poszkodowanego w pozycji bocznej ustalonej.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Bezpieczeństwo i higiena pracy* w części teoretycznej powinny odbywać się w sali przedmiotowej, wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, telewizor, plansze dydaktyczne z zakresu bezpieczeństwa i higieny pracy oraz pierwszej pomocy, modele, makiety, schematy, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat przepisów prawa regulujących zasady dotyczące bezpiecznych i higienicznych warunków pracy, ochrony zdrowia i życia, ochrony przeciwpożarowej oraz ergonomii pracy.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, rysunki, ilustracje, zestawy aktualnych przepisów prawa dotyczącego bezpieczeństwa i higieny pracy, ochrony przeciwpożarowej, ochrony środowiska oraz wymagań ergonomii, środków ochrony indywidualnej i zbiorowej. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład informacyjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz z instruktążem, ćwiczenia, odgrywanie scenek sytuacyjnych. Każde ćwiczenie powinno zostać poprzedzone szczegółowym instruktążem i pokazem. Jedną z metod powinna być metoda tekstu przewodniego, która wymaga od nauczyciela przed przystąpieniem do lekcji opracowania „przewodnika” dla ucznia, z pytaniami prowadzącymi go po wybranym obszarze wiedzy. Inną metodą może być metoda projektu, której celem będzie rozwijanie umiejętności pracy w małych zespołach oraz przypisanie ról poszczególnym członkom grupy. Wszystkie metody powinny pomagać w zainteresowaniu uczniów tematem poruszonym na zajęciach dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych. Część efektów kształcenia powinna być osiągnięta w korelacji z kształceniem zawodowym praktycznym oraz edukacją dla bezpieczeństwa.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod i narzędzi weryfikacji przyrostu wiedzy i znajomości zagadnień oraz umiejętności praktycznych z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji pracy ucznia podczas wykonywania ćwiczeń, udzielania pierwszej pomocy. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu. Szczególną uwagę należy zwrócić na to, czy uczeń przestrzega obowiązujących

instrukcji i przepisów BHP, potrafi zidentyfikować czynniki zagrażające zdrowiu i życiu oraz wskazać realne zagrożenia występujące w środowisku pracy, odpowiednio dobiera środki ochrony indywidualnej i zbiorowej oraz czy umiejętnie stosuje procedury udzielania pierwszej pomocy poszkodowanym.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o różne źródła wiedzy (podręcznik, artykuły, materiały elektroniczne) oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.3. Podstawy informatyki

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) charakteryzuje parametry sprzętu komputerowego	1) identyfikuje parametry urządzeń techniki komputerowej 2) porównuje parametry tego samego typu urządzeń techniki komputerowej (np. dwóch kart graficznych, dwóch dysków twardych) 3) przelicza jednostki pojemności pamięci masowych 4) dobiera urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi dla danego stanowiska (np. grafika komputerowego)	1) rozpoznać parametry sprzętu komputerowego 2) porównać parametry sprzętu komputerowego i jego podzespołów 3) przeliczać jednostki pojemności pamięci masowej (pamięci operacyjnej, nośników danych) 4) dobrać sprzęt i jego parametry odpowiednio do wymagań i przeznaczenia	
2) definiuje elementy architektury systemów komputerowych	1) opisuje zasadę działania procesora (rozkazy) 2) wymienia zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego	1) wyjaśnić zasadę działania procesora (CPU) 2) scharakteryzować funkcje elementów systemu komputerowego: <ul style="list-style-type: none"> • procesora • pamięci operacyjnej 	

		<ul style="list-style-type: none"> • płyty głównej • układów wejścia-wyjścia 	
3) charakteryzuje systemy informatyczne oraz rozróżnia systemy informatyczne pod względem funkcjonalności	<ol style="list-style-type: none"> 1) identyfikuje systemy informatyczne, podaje przykłady systemów informacji przetwarzanych elektronicznie, w tym: system PESEL, nabór elektroniczny do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany 2) opisuje miejsca przechowywania informacji: serwer lokalny, chmurę, nośniki danych 3) dobiera systemy informatyczne pod względem ich funkcjonalności 4) opisuje działanie portali społecznościowych 5) określa zasady bezpiecznego korzystania z portali społecznościowych 6) podaje przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym: e-commerce, e-sklep, e-faktura, systemy rezerwacyjne 	<ol style="list-style-type: none"> 1) wymienić elementy składające się na systemy informatyczne 2) scharakteryzować rodzaje systemów informacji przetwarzanych elektronicznie, bazując na przykładach takich jak: <ul style="list-style-type: none"> • system PESEL • nabór elektroniczny do szkół • e-dziennik • system bankowości elektronicznej • profil zaufany 3) omówić zastosowania systemów informatycznych w działalności biznesowej na przykładach takich jak: <ul style="list-style-type: none"> • e-commerce • e-sklep • e-faktura • systemy rezerwacyjne 4) omówić miejsca przechowywania informacji takie jak: <ul style="list-style-type: none"> • serwer lokalny • chmura informatyczna • nośniki danych 5) scharakteryzować funkcjonowanie internetowych serwisów społecznościowych 	

		6) określić zasady bezpiecznego korzystania z internetowych serwisów społecznościowych	
4) stosuje zalecenia dotyczące ułatwień w zakresie dostępności serwisów internetowych dla osób niepełnosprawnych	<ol style="list-style-type: none"> 1) wymienia dostępne udogodnienia dla osób z niepełnosprawnościami 2) wymienia wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0 	<ol style="list-style-type: none"> 1) wymienić udogodnienia dla osób z niepełnosprawnościami zgodne ze standardem WCAG 2.0 2) podać wymagania dla różnych poziomów dostępności zgodnych ze standardem WCAG 2.0 	
5) posługuje się terminologią dotyczącą sieci komputerowych	<ol style="list-style-type: none"> 1) wymienia topologie sieci 2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej 3) opisuje sieć bezprzewodową oraz sieć przewodową 4) stosuje programy monitorujące łącze internetowe 5) definiuje pojęcia: pobieranie i wysyłanie danych 6) opisuje zasady działania sieci synchronicznej i asynchronicznej 7) wskazuje różnice w działaniu sieci synchronicznej i asynchronicznej 8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci 9) używa komunikatorów tekstowych, audio/wideo oraz tablic interaktywnych 10) stosuje zasady netykiety 	<ol style="list-style-type: none"> 1) odróżnić topologię logiczną od topologii fizycznej sieci komputerowej (porównać ich funkcje) 2) scharakteryzować warstwy modelu TCP/IP 3) scharakteryzować protokoły komunikacji sieciowej 4) opisać sieć bezprzewodową oraz sieć przewodową 5) wymienić/dobrać odpowiednie oprogramowanie do monitorowania łącza internetowego 6) definiować pojęcia: <ul style="list-style-type: none"> • pobieranie danych • wysyłanie danych 7) rozróżnić i scharakteryzować sieć synchroniczną i asynchroniczną 8) wymienić i stosować zasady bezpiecznego korzystania z internetu 9) omówić i stosować zasady przyzwoitego zachowania w internecie (netykietę) 10) obsługiwać komunikatory tekstowe, 	

		dźwiękowe, wideo oraz tablice interaktywne	
6) stosuje pozycyjne systemy liczbowe	<ol style="list-style-type: none"> 1) przekształca liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym 2) zapisuje liczby w kodzie uzupełnieniowym do dwóch 3) wykonuje podstawowe działania logiczne i arytmetyczne na liczbach binarnych 4) wykorzystuje dostępne narzędzia informatyczne do wykonywania działań na liczbach zapisanych w różnych pozycyjnych systemach liczbowych (np. kalkulatory HEX, DEC, BIN) 	<ol style="list-style-type: none"> 1) przeliczać i zapisywać liczby zamiennie w różnych pozycyjnych systemach liczbowych 2) zapisywać liczby w kodzie uzupełnieniowym do dwóch 3) wykonywać operacje arytmetyczne w systemie dwójkowym (dodawanie, odejmowanie, mnożenie, dzielenie) 4) wykonywać operacje logiczne w systemie dwójkowym (takie jak negacja, suma logiczna, iloczyn logiczny, różnica symetryczna) 5) wykonywać działania na liczbach zapisanych w różnych systemach liczbowych za pomocą konwertera systemów liczbowych (kalkulatora programisty) lub innych dostępnych narzędzi informatycznych 	
7) stosuje zasady cyberbezpieczeństwa	<ol style="list-style-type: none"> 1) rozróżnia rodzaje szkodliwego oprogramowania 2) rozróżnia rodzaje ataków hakerskich 3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami 4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej wynikające z przebywania w cyberprzestrzeni 5) opisuje zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom 	<ol style="list-style-type: none"> 1) rozpoznać rodzaje szkodliwego oprogramowania 2) rozpoznać rodzaje ataków hakerskich 3) określić środki ochrony przed złośliwym oprogramowaniem i cyberatakami 4) nazwać zagrożenia związane z korzystaniem z internetu z uwzględnieniem sfery: <ul style="list-style-type: none"> • psychicznej (emocjonalnej) • fizycznej • społecznej • poznawczej 5) scharakteryzować zagrożenia związane w 	

	<ol style="list-style-type: none"> 6) przestrzega zasad bezpiecznego przechowywania danych 7) przestrzega zasad bezpieczeństwa swojego cyfrowego wizerunku i tożsamości 8) przestrzega zasad prywatności w cyfrowym świecie 9) wymienia i omawia podstawowe pojęcia związane z ochroną danych osobowych, informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony 10) stosuje zasady dokonywania bezpiecznych transakcji w internecie 	<p>korzystaniem z internetu z uwzględnieniem sfery:</p> <ul style="list-style-type: none"> • psychicznej (emocjonalnej) • fizycznej • społecznej • poznawczej <ol style="list-style-type: none"> 6) określić sposoby przeciwdziałania zagrożeniom w cyberprzestrzeni 7) wymieni i stosować zasady bezpiecznego i odpowiedzialnego korzystania z internetu w zakresie: <ul style="list-style-type: none"> • bezpiecznego przechowywania danych • bezpieczeństwa cyfrowego wizerunku • ochrony tożsamości cyfrowej • ustalania zasad prywatności • dokonywania bezpiecznych transakcji cyfrowych 8) zdefiniować pojęcia i wyjaśnić potrzebę: <ul style="list-style-type: none"> • ochrony danych osobowych • ochrony informacji • ochrony praw autorskich • ochrony własności intelektualnej 	
<ol style="list-style-type: none"> 8) rozpoznaje właściwe normy i procedury oceny zgodności podczas realizacji zadań zawodowych 	<ol style="list-style-type: none"> 1) wymienia cele normalizacji krajowej 2) wyjaśnia, czym jest norma, i wymienia cechy normy 3) rozróżnia oznaczenie normy międzynarodowej, europejskiej i krajowej 	<ol style="list-style-type: none"> 1) wyjaśnić pojęcia: <ul style="list-style-type: none"> • norma • normalizacja 2) określić cele i zasady normalizacji krajowej 	

	4) korzysta ze źródeł informacji dotyczących norm i procedur oceny zgodności	3) scharakteryzować różnice w oznaczeniach normy międzynarodowej, europejskiej i krajowej 4) wyszukiwać informacje na temat norm i procedur oceny zgodności	
--	--	--	--

Planowane zadania

Zadanie 1.

- Wykonaj prezentację multimedialną na temat „Rodzaje szkodliwego oprogramowania”.

Zadanie 2.

- Przelicz z systemu dziesiętnego na binarny, np. $16_{(10)} = \dots_{(2)}$.
- Przelicz liczby binarne na dziesiętne, np. $10_{(2)} = \dots_{(10)}$.
- Przelicz liczby binarne na szesnastkowe, np. $101110_{(2)} = \dots_{(16)}$.
- Wykonaj działania bez przeliczania na system dziesiętny, np.
 - $10101_{(2)} + 10101_{(2)} = \dots$
 - $1001_{(2)} - 11_{(2)} = \dots$
 - $10010_{(2)} * 101_{(2)} = \dots$
 - $10110_{(2)} : 10_{(2)} = \dots$

Zadanie 3.

Wyjaśnij zasadę działania CPU i omów typowe rozkazy wykonywane przez procesor.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Podstawy informatyki* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat urządzeń techniki

komputerowej i architektury systemów komputerowych, zastosowania pozycyjnych systemów liczbowych, podstawowych zagadnień dotyczących sieci komputerowych, stosowania zasad cyberbezpieczeństwa, działania systemów informatycznych oraz zasad normalizacji i procedur oceny zgodności.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, fotografie, zestawy ćwiczeń. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład informacyjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz, zestawy zadań. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, wystąpień uczniów z prezentacją na forum klasy. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.4. Projektowanie i tworzenie stron internetowych

Efekty kształcenia	Kryteria weryfikacji	Po zrealizowaniu zajęć uczeń potrafi	Material w
---------------------------	-----------------------------	---	-------------------

z podstawy programowej	z podstawy programowej	z podstawy programowej	podręczniku
Uczeń:	Uczeń:		
1) posługuje się hipertekstowymi językami znaczników (HTML — ang. <i>HyperText Markup Language</i>)	1) korzysta ze standardów dokumentów hipertekstowych 2) stosuje znaczniki języka HTML 3) definiuje strukturę dokumentu hipertekstowego, korzystając ze znaczników sekcji 4) definiuje hierarchię treści, stosując znaczniki nagłówków i paragrafu 5) definiuje elementy strony internetowej: listy, tabele, obrazy, odnośniki, kontrolki 6) wykonuje formularze na stronie internetowej	1) scharakteryzować standardy języków hipertekstowych 2) określić znaczenie i zastosowanie poszczególnych znaczników języka HTML5 i ich atrybutów 3) opisać strukturę strony internetowej opartą na znacznikach sekcji 4) zdefiniować hierarchiczną strukturę tekstu witryny internetowej za pomocą znaczników nagłówków różnego stopnia i paragrafu 5) zdefiniować elementy składowe strony internetowej (listy, tabele, obrazy, odsyłacze, formularze, multimedia, zdarzenia, gotowe skrypty) 6) rozróżnić typy pól formularzy i ich funkcje	Rozdział 1. Wprowadzenie Rozdział 2. Język HTML5
2) stosuje kaskadowe arkusze stylów do tworzenia responsywnych stron internetowych	1) stosuje style lokalne, wewnętrzne i zewnętrzne 2) stosuje wykorzystuje kaskadowość stylów 3) rozróżnia selektory elementów, atrybutów, specjalne, pseudoklas i pseudoelementów 4) rozpoznaje selektory CSS 5) stosuje selektory CSS, ich własności i wartości 6) projektuje wygląd strony internetowej przy użyciu języka CSS 7) wykonuje responsywne strony internetowe z wykorzystaniem CSS	1) scharakteryzować rodzaje stylów CSS 2) wyjaśnić pojęcie kaskadowości stylów 3) rozróżnić i opisać rodzaje selektorów 4) wyjaśnić definicję reguły CSS zastosowaną do sformatowania elementów na stronie internetowej 5) omówić budowę szablonu strony wraz z zastosowanym stylem w języku CSS 6) scharakteryzować responsywny układ strony	Rozdział 3. Kaskadowe arkusze stylów (CSS)
3) stosuje systemy zarządzania treścią (CMS — ang. <i>Content</i>)	1) określa funkcje systemów zarządzania treścią	1) scharakteryzować najpopularniejsze systemy zarządzania treścią (CMS)	Rozdział 9.

<i>Management System)</i>	<ol style="list-style-type: none"> 2) określa funkcje panelu administratora w systemach zarządzania treścią 3) instaluje systemy zarządzania treścią (Joomla! i WordPress) 4) konfiguruje systemy zarządzania treścią (Joomla! i WordPress) 5) administruje systemem zarządzania treścią (Joomla! i WordPress) 6) wykorzystuje gotowe szablony dla systemów CMS 7) aktualizuje systemy CMS 	<ol style="list-style-type: none"> 2) opisać dostępne funkcje administracyjne systemów zarządzania treścią 3) wymienić czynności wykonywane podczas instalacji i aktualizacji systemów zarządzania treścią 4) omówić konfigurację systemów zarządzania treścią 5) zaplanować stworzenie strony internetowej z wykorzystaniem gotowych szablonów systemów zarządzania treścią 	Systemy zarządzania treścią (CMS)
4) projektuje grafikę komputerową	<ol style="list-style-type: none"> 1) rozróżnia podstawowe pojęcia dotyczące grafiki komputerowej rastrowej i wektorowej 2) przestrzega zasad cyfrowego zapisu obrazu 3) dobiera oprogramowanie do obróbki grafiki komputerowej 4) identyfikuje różne formaty plików graficznych 5) stosuje różne modele barw 6) osadza tekst na grafice oraz dobiera jego krój i styl 7) korzysta z funkcji edytora grafiki wektorowej 8) korzysta z funkcji edytora grafiki rastrowej 9) wykonuje edycję plików graficznych na potrzeby stron internetowych 10) projektuje elementy graficzne dla strony internetowej 	<ol style="list-style-type: none"> 1) scharakteryzować cechy i zastosowanie grafiki rastrowej i wektorowej 2) omówić zasady cyfrowego zapisu obrazu oraz tworzenia grafiki na potrzeby strony internetowej 3) rozpoznać oprogramowanie użytkowe i jego narzędzia do obróbki grafiki komputerowej 4) określić cechy i zastosowanie formatów plików graficznych 5) scharakteryzować modele barw oraz sposoby zapisu koloru za ich pomocą 	Rozdział 5. Grafika na stronie internetowej
5) wykorzystuje elementy multimedialne na stronach	1) określa zasady komputerowego przetwarzania wideo i dźwięku przygotowanego na potrzeby	1) scharakteryzować cechy i zastosowanie wideo, dźwięku i animacji	Rozdział 7. Animacja na stronie

internetowych	strony internetowej 2) dobiera oprogramowanie do edycji obrazu ruchomego i dźwięku 3) wykonuje animacje na potrzeby stron internetowych 4) wykonuje materiały wideo na potrzeby stron internetowych 5) edytuje wideo i dźwięk na potrzeby stron internetowych 6) osadza materiały multimedialne na stronach internetowych 7) importuje materiały multimedialne do systemów zarządzania treścią (CMS)	2) omówić zasady cyfrowego przetwarzania wideo i dźwięku oraz tworzenia animacji komputerowej na potrzeby strony internetowej 3) rozpoznać oprogramowanie użytkowe do tworzenia i edycji wideo, dźwięku i animacji 4) określić cechy i zastosowanie formatów plików wideo, dźwiękowych i animacji	internetowej Rozdział 8. Dźwięk i wideo
6) wykonuje strony internetowe zgodnie z projektami	1) projektuje układ sekcji na stronie internetowej 2) analizuje projekt strony internetowej 3) tworzy strukturę strony internetowej zgodnie z projektem 4) dobiera paletę barw dla strony internetowej 5) dobiera czcionki dla strony internetowej 6) uwzględnia potrzeby użytkowników z różnymi niepełnosprawnościami przy projektowaniu stron internetowych, dostosowując np. kontrast, powiększenie, inne elementy wspomagające niepełnosprawnych 7) opisuje zasady i znaczenie wytycznych dotyczących ułatwień w dostępie do treści publikowanych w internecie 8) tworzy stronę zgodną z wytycznymi dotyczącymi ułatwień w dostępie do treści publikowanych w internecie	1) omówić standardy budowy graficznego projektu strony internetowej 2) określić zasady doboru barw na stronie internetowej 3) scharakteryzować różnice w zastosowaniu różnych jednostek wielkości czcionek na stronie internetowej 4) omówić zasady i wytyczne dotyczące dostępności treści internetowych (WCAG 2.0) 5) wymienić dostosowania do potrzeb osób niepełnosprawnych stosowane na stronach internetowych	Rozdział 6. Graficzny projekt strony internetowej

<p>7) stosuje reguły testowania, walidacji i optymalizacji stron internetowych</p>	<ol style="list-style-type: none"> 1) testuje stronę internetową w różnych przeglądarkach 2) testuje responsywność strony internetowej 3) określa proces walidacji strony internetowej 4) dobiera narzędzia walidacji strony internetowej 5) dokonuje walidacji strony internetowej 6) optymalizuje stronę internetową 7) określa proces pozycjonowania strony internetowej 8) stosuje zasady dostępności (WCAG) i pozycjonowania strony internetowej 	<ol style="list-style-type: none"> 1) scharakteryzować etapy procesu testowania i sprawdzania poprawności strony internetowej 2) omówić funkcje walidatorów i przebieg procesu walidacji strony internetowej 3) opisać metody i sposoby pozycjonowania strony internetowej 4) określić wymagania w zakresie optymalizacji strony internetowej 5) scharakteryzować narzędzia wykorzystywane do testowania szybkości wczytywania strony i jej funkcjonalności 	<p>Rozdział 10. Testowanie i publikowanie strony</p>
<p>8) publikuje witryny i aplikacje internetowe</p>	<ol style="list-style-type: none"> 1) opisuje usługi hostingu 2) dobiera usługi hostingu w zależności od potrzeb użytkownika 3) opisuje operacje na domenach internetowych 4) wykonuje operacje na domenach internetowych 5) rozpoznaje etapy publikacji witryn i aplikacji internetowych 6) opisuje funkcje programów wykorzystywanych do przesyłania danych na serwer 7) dobiera program do przesyłania danych na serwer 8) przesyła dane na serwer 9) dobiera pakiety serwerowe WWW 10) sprawdza poprawność publikowanych stron WWW 	<ol style="list-style-type: none"> 1) scharakteryzować usługi hostingu stron internetowych 2) omówić operacje wykonywane na domenach internetowych 3) opisać protokoły sieciowe wykorzystywane do publikacji strony internetowej 4) rozróżnić sposoby i etapy publikacji witryn i aplikacji internetowych 5) scharakteryzować działanie programów z klientem FTP 	<p>Rozdział 10. Testowanie i publikowanie strony</p>

	11) publikuje witryny internetowe		
--	-----------------------------------	--	--

* *Jolanta Pokorska, Podręcznik do nauki zawodu technik informatyk, technik programista. Część 1., Tworzenie stron internetowych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu*

Planowane zadania

Zadanie 1.

Rozszyfruj skróty:

- HTML, XHTML, CSS, FTP, HTTP, CMS, JS, RWD, SEO, WCAG.

Zadanie 2.

Wymień cechy i zastosowanie:

- hipertekstowego języka znaczników (HTML),
- kaskadowych arkuszy stylów (CSS),
- obrazów rastrowych/obrazów wektorowych,
- modelu barw RGB/CMYK.

Zadanie 3.

Przedstaw strukturę dokumentu hipertekstowego i omów znaczenie poszczególnych elementów jego budowy.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Projektowanie i tworzenie stron internetowych* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat tworzenia witryn internetowych za pomocą języków HTML i CSS, zagadnień dotyczących tworzenia i modyfikowania elementów graficznych i multimedialnych na potrzeby stron internetowych oraz systemów zarządzania treścią w celu wykonania witryny internetowej za ich pomocą.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, rysunki, ilustracje, fotografie, nagrania dźwiękowe, nagrania wideo, animacje komputerowe. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.5. Projektowanie i administrowanie bazami danych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		

<p>1) posługuje się pojęciami dotyczącymi baz danych</p>	<ol style="list-style-type: none"> 1) określa pojęcia związane z bazami danych: encja, związki encji, atrybuty encji, klucz relacji 2) określa typy danych używanych w bazach danych 3) stosuje odpowiednie typy danych przy definiowaniu encji 4) rozpoznaje postacie normalne baz danych 5) opisuje cechy relacyjnej bazy danych 	<ol style="list-style-type: none"> 1) scharakteryzować modele baz danych 2) określić cechy relacyjnego modelu baz danych 3) scharakteryzować pojęcia związane z algebrą relacji, takie jak selekcja, projekcja, połączenie 4) scharakteryzować pojęcia związane z bazami danych, takie jak klucz podstawowy, klucz obcy, integralność danych, typy związków 5) objaśnić pojęcia encji, atrybutów encji, dziedziny 6) określić typy danych używanych w bazach danych 7) scharakteryzować postacie normalne bazy danych 8) opisać cechy relacyjnej bazy danych 	
<p>2) tworzy diagramy E/R (ang. <i>Entity-Relationship Diagram</i>)</p>	<ol style="list-style-type: none"> 1) charakteryzuje typy notacji diagramów E/R 2) rozróżnia bloki składowe diagramów E/R 3) analizuje diagramy E/R 4) definiuje encje i atrybuty encji 5) definiuje związki między encjami i określa ich liczebność 6) dobiera typ danych do określonych atrybutów encji 7) określa klucz główny dla encji 	<ol style="list-style-type: none"> 1) określić podstawowe zasady projektowania baz danych 2) scharakteryzować typy notacji diagramów ERD 3) rozróżnić bloki składowe diagramów ERD 4) rozpoznać podstawowe funkcje narzędzi CASE 5) analizować diagramy ERD 6) definiować encje i atrybuty encji 7) definiować związki między encjami 8) dobierać typ danych atrybutów encji 9) określać klucz główny dla encji 	
<p>3) korzysta z systemów zarządzania</p>	<ol style="list-style-type: none"> 1) rozróżnia dostępne SZBD 	<ol style="list-style-type: none"> 1) scharakteryzować architekturę systemu baz 	

bazami danych SZBD (ang. <i>Database Management System</i>)	<ol style="list-style-type: none"> 2) dobiera SZBD do określonego zastosowania 3) instaluje SZBD 4) konfiguruje SZBD do pracy w środowisku wielu użytkowników 5) aktualizuje SZBD 	danych <ol style="list-style-type: none"> 2) scharakteryzować różne SZBD 	
<ol style="list-style-type: none"> 4) stosuje strukturalny język zapytań SQL (ang. <i>Structured Query Language</i>) 	<ol style="list-style-type: none"> 1) opisuje polecenia języka SQL 2) stosuje polecenia języka SQL 3) definiuje struktury baz danych przy użyciu instrukcji języka zapytań 4) wyszukuje informacje w bazie danych przy użyciu języka SQL 5) zmienia rekordy w bazie danych przy użyciu języka SQL 6) usuwa rekordy w bazie danych przy użyciu języka SQL 7) tworzy skrypty w strukturalnym języku zapytań 	<ol style="list-style-type: none"> 1) opisać standardy języka SQL 2) scharakteryzować składnię języka SQL 3) określać typy danych języka SQL 4) scharakteryzować polecenia tworzące strukturę bazy danych — język DDL 5) scharakteryzować sposoby modyfikowania danych w bazie — język DML 6) scharakteryzować sposoby usuwania danych z bazy 7) scharakteryzować sposoby wyszukiwania informacji w bazie danych przy użyciu języka SQL 	
<ol style="list-style-type: none"> 5) tworzy formularze, zapytania i raporty do przetwarzania danych 	<ol style="list-style-type: none"> 1) tworzy formularze do wprowadzania i modyfikowania danych 2) identyfikuje rodzaje zapytań 3) tworzy zapytania i podzapytania do tabel bazy danych 4) tworzy raporty w bazie danych 	<ol style="list-style-type: none"> 1) scharakteryzować rodzaje zapytań 2) określić klauzule instrukcji SELECT 3) pogrupować dane i używać funkcji agregujących 4) określić rodzaje połączeń w języku SQL 5) scharakteryzować sposoby definiowania więzów integralności w języku SQL 6) wymienić sposoby łączenia wyników zapytań w języku SQL 7) opisać sposoby tworzenia podzapytań do tabel bazy danych 	

6) modyfikuje struktury baz danych	<ol style="list-style-type: none"> 1) analizuje strukturę bazy danych w celu jej modyfikacji 2) rozbudowuje strukturę bazy danych, tworząc tabele, pola, relacje i atrybuty 3) weryfikuje poprawność struktury bazy danych po rozbudowie 4) usuwa elementy struktury bazy danych oraz dane 5) modyfikuje strukturę bazy danych oraz dane 	<ol style="list-style-type: none"> 1) scharakteryzować strukturę bazy danych w celu jej modyfikacji 2) określić sposoby rozbudowy i modyfikacji struktury bazy danych 3) zweryfikować poprawność struktury bazy danych po modyfikacji 4) określić sposoby usuwania, modyfikowania i dodawania danych do bazy 	
7) zarządza systemem bazy danych	<ol style="list-style-type: none"> 1) tworzy użytkowników bazy danych 2) określa uprawnienia dla użytkowników 3) kontroluje spójność bazy danych 4) tworzy kopię zapasową struktury bazy danych 5) weryfikuje poprawność kopii zapasowej bazy danych 6) przywraca dane z kopii zapasowej bazy danych 7) importuje i eksportuje tabele bazy danych 8) diagnozuje i naprawia bazę danych 	<ol style="list-style-type: none"> 1) określić sposoby tworzenia użytkowników bazy danych 2) scharakteryzować role w bazie danych 3) scharakteryzować rodzaje uprawnień dla użytkowników 4) określić sposoby nadawania i odebrania uprawnień użytkownikowi 5) opisać mechanizm dziedziczenia uprawnień 6) scharakteryzować mechanizm replikacji bazy danych 7) określić strategię wykonania kopii bezpieczeństwa 8) określić sposób przywracania danych z kopii zapasowej bazy danych 9) określić sposoby importowania i eksportowania tabel i danych 	

* Jolanta Pokorska, Podręcznik do nauki zawodu technik informatyk, technik programista. Część 2. Tworzenie i administrowanie bazami danych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu

Planowane zadania

Zadanie 1.

Podaj definicję encji.

Zadanie 2.

Wyjaśnij pojęcia związane z algebrą relacji, takie jak selekcja, projekcja, połączenie.

Zadanie 3.

Wymień i omów podstawowe cechy baz danych.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Projektowanie i administrowanie bazami danych* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy z programem MS Access, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat projektowania baz danych i administrowania nimi, zagadnień dotyczących relacyjnych baz danych oraz cech baz danych.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, projekty baz danych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.6. Programowanie aplikacji internetowych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) stosuje zasady programowania	1) analizuje problemy programistyczne 2) stosuje algorytmy 3) stosuje zasady programowania strukturalnego	1. scharakteryzować podstawowe pojęcia: aplikacja internetowa, skrypt, aplet 2. opisać reprezentacje algorytmów 3. opisać narzędzia programistyczne typu edytor, debugger, IDE 4. opisać strukturę aplikacji internetowej	
2) stosuje skryptowe języki programowania	1) stosuje języki JavaScript oraz jeden z: Python/ASP.NET/PHP/JSP do tworzenia aplikacji internetowych 2) identyfikuje skryptowe języki programowania 3) implementuje algorytmy w języku interpretowanym 4) posługuje się typami prostymi i złożonymi,	1. scharakteryzować skryptowe języki programowania 2. scharakteryzować typy proste i złożone stosowane przy tworzeniu aplikacji 3. scharakteryzować zmienne stosowane przy tworzeniu aplikacji 4. scharakteryzować operatory stosowane przy	

	<p>zmiennymi i operatorami w językach skryptowych</p> <p>5) stosuje instrukcje sterujące języków skryptowych</p> <p>6) stosuje funkcje oraz wybrane biblioteki języków skryptowych</p> <p>7) tworzy strony internetowe wykorzystujące skryptowe języki programowania</p>	<p>tworzeniu aplikacji</p> <p>5. scharakteryzować instrukcje sterujące języków skryptowych</p> <p>6. scharakteryzować funkcje oraz wybrane biblioteki języków skryptowych</p>	
<p>3) programuje skrypty wykonywane po stronie klienta</p>	<p>1) programuje w języku JavaScript</p> <p>2) stosuje w programowaniu obsługę zdarzeń myszy i klawiatury</p> <p>3) stosuje biblioteki wykorzystywane w skryptach po stronie klienta</p> <p>4) definiuje skrypty obsługujące formularze i kontrolki HTML</p> <p>5) wykorzystuje walidację formularzy HTML za pomocą mechanizmów HTMLS</p> <p>6) korzysta z funkcji modelu DOM</p> <p>7) korzysta z bibliotek i frameworków języka JavaScript, w tym z biblioteki jQuery, Angular, React</p>	<p>1. opisać składnię języka JavaScript</p> <p>2. opisać funkcje języka JavaScript</p> <p>3. scharakteryzować obiektowość języka JavaScript</p> <p>4. opisać obiekty przeglądarki</p> <p>5. opisać obiekty języka JavaScript</p> <p>6. opisać obiekty dokumentu DOM</p> <p>7. scharakteryzować obsługę zdarzeń myszy i klawiatury</p> <p>8. opisać sposoby obsługi formularzy i kontrolek HTML</p> <p>9. scharakteryzować działanie wyrażeń regularnych</p> <p>10. scharakteryzować bibliotekę jQuery</p> <p>11. scharakteryzować bibliotekę AngularJS</p> <p>12. scharakteryzować bibliotekę React</p>	
<p>4) programuje skrypty wykonywane po stronie serwera</p>	<p>1) programuje w jednym z języków Python/ASP.NET/PHP/JSP</p> <p>2) stosuje wbudowane instrukcje, funkcje</p> <p>3) stosuje metody przesyłania danych z formularza</p>	<p>1. opisać składnię języka PHP</p> <p>2. opisać funkcje języka PHP</p> <p>3. opisać funkcje obsługi plików</p> <p>4. opisać sposoby przekazywania danych z</p>	

	<p>4) programuje wysyłanie danych z formularza HTML</p> <p>5) stosuje biblioteki do obsługi bazy danych, odpowiednie dla języka i frameworka</p> <p>6) korzysta z funkcji do obsługi plików</p> <p>7) korzysta z funkcji do obsługi ciasteczek (ang. <i>cookies</i>) oraz sesji</p>	<p>formularza</p> <p>5. opisać zastosowanie plików <i>cookies</i></p> <p>6. opisać zastosowanie sesji</p> <p>7. opisać sposób współpracy aplikacji z bazą danych</p> <p>8. opisać sposób definiowania zapytań do bazy danych</p> <p>9. opisać bibliotekę PDO</p>	
5) stosuje środowisko programistyczne i uruchomieniowe aplikacji internetowych	<p>1) opisuje funkcje środowiska programistycznego</p> <p>2) dobiera środowisko programistyczne do określonych zadań i języka programowania</p> <p>3) tworzy programy w wybranym środowisku programistycznym</p> <p>4) instaluje i konfiguruje serwer WWW</p> <p>5) instaluje i konfiguruje serwer baz danych</p> <p>6) korzysta z gotowych pakietów dla aplikacji internetowych, np. phpMyAdmin</p>	<p>1. scharakteryzować środowiska programistyczne aplikacji internetowych</p> <p>2. opisać funkcje środowiska programistycznego</p>	
6) przeprowadza walidację kodu programu	<p>1) analizuje błędy w kodzie źródłowym programu</p> <p>2) wykonuje testy tworzonych programów</p> <p>3) poprawia błędy w tworzonych programach</p> <p>4) stosuje debugger w przeglądarce internetowej</p>	<p>1. analizować błędy w kodzie źródłowym programu</p> <p>2. opisać sposoby testowania tworzonych programów</p>	
7) dokumentuje tworzoną aplikację	<p>1) stosuje komentarze w kodzie źródłowym programu</p> <p>2) tworzy dokumentację programu</p> <p>3) tworzy instrukcję użytkownika programu</p>	<p>1. opisać sposoby umieszczania komentarzy w kodzie źródłowym programu</p> <p>2. opisać sposoby dokumentowania programu</p> <p>3. opisać elementy instrukcji użytkownika</p>	

* Jolanta Pokorska, *Podręcznik do nauki zawodu technik informatyk, technik programista. Część 3., Tworzenie aplikacji internetowych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu*

Planowane zadania

Zadanie 1.

Z jakich warstw składa się podstawowa architektura aplikacji internetowej?

Zadanie 2.

Wyjaśnij znaczenie wszystkich atrybutów znacznika `<form>` podanego niżej polecenia:

```
<form action="http://localhost/form.php" method="post">
```

Zadanie 3.

W jaki sposób w aplikacji internetowej wykorzystywane są pliki *cookies*?

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Programowanie aplikacji internetowych* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat architektury aplikacji internetowych, identyfikowania skryptowych języków programowania oraz ich funkcjonalności, zasad korzystania z bibliotek i frameworków.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, wzorce aplikacji internetowych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.7. Pracownia stron internetowych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) posługuje się hipertekstowymi językami znaczników (HTML — ang. <i>HyperText Markup Language</i>)	1) korzysta ze standardów dokumentów hipertekstowych 2) stosuje znaczniki języka HTML 3) definiuje strukturę dokumentu hipertekstowego, korzystając ze znaczników sekcji 4) definiuje hierarchię treści, stosując znaczniki nagłówków i paragrafu 5) definiuje elementy strony internetowej: listy, tabele, obrazy, odnośniki, kontrolki 6) wykonuje formularze na stronie internetowej	1) zastosować standardy języków hipertekstowych 2) posłużyć się poszczególnymi znacznikami języka HTML5 i ich atrybutami 3) budować strukturę strony internetowej opartą na znacznikach sekcji 4) tworzyć hierarchiczną strukturę tekstu witryny internetowej za pomocą znaczników nagłówków i paragrafu 5) wykorzystywać różne elementy (listy, tabele, obrazy, odsyłacze, formularze, multimedia, zdarzenia, gotowe skrypty) do budowy strony internetowej 6) stosować różne typy pól do budowy formularzy na stronach internetowych	Rozdział 1. Wprowadzenie Rozdział 2. Język HTML5

<p>2) stosuje kaskadowe arkusze stylów do tworzenia responsywnych stron internetowych</p>	<ol style="list-style-type: none"> 1) stosuje style lokalne, wewnętrzne i zewnętrzne 2) stosuje kaskadowość stylów 3) rozróżnia selektory elementów, atrybutów, specjalne, pseudoklas i pseudoelementów 4) rozpoznaje selektory CSS 5) stosuje selektory CSS, ich własności i wartości 6) projektuje wygląd strony internetowej przy wykorzystaniu języka CSS 7) wykonuje responsywne strony internetowe z wykorzystaniem CSS 	<ol style="list-style-type: none"> 1) tworzyć różne rodzaje stylów CSS 2) wykorzystywać kaskadowość stylów 3) wykorzystywać różne rodzaje selektorów 4) budować reguły CSS w celu formatowania elementów na stronie internetowej 5) tworzyć różnorodne szablony stron internetowych z wykorzystaniem stylów CSS 6) projektować responsywny układ strony 	<p>Rozdział 3. Kaskadowe arkusze stylów (CSS)</p>
<p>3) stosuje systemy zarządzania treścią (CMS — ang. <i>Content Management System</i>)</p>	<ol style="list-style-type: none"> 1) określa funkcje systemów zarządzania treścią 2) określa funkcje panelu administratora w systemach zarządzania treścią 3) instaluje systemy zarządzania treścią (Joomla! i WordPress) 4) konfiguruje systemy zarządzania treścią (Joomla! i WordPress) 5) administruje systemem zarządzania treścią (Joomla! i WordPress) 6) wykorzystuje gotowe szablony dla systemów CMS 7) aktualizuje systemy CMS 	<ol style="list-style-type: none"> 1) instalować, konfigurować i aktualizować systemy zarządzania treścią 2) administrować systemem zarządzania treścią 3) instalować motywy i wtyczki w systemach zarządzania treścią 4) tworzyć strony internetowe z wykorzystaniem gotowych szablonów systemów zarządzania treścią 	<p>Rozdział 9. Systemy zarządzania treścią (CMS)</p>
<p>4) projektuje grafikę komputerową</p>	<ol style="list-style-type: none"> 1) rozróżnia podstawowe pojęcia dotyczące grafiki komputerowej rastrowej i wektorowej 2) przestrzega zasad cyfrowego zapisu obrazu 3) dobiera oprogramowanie do obróbki grafiki 	<ol style="list-style-type: none"> 1) dobierać i obsługiwać oprogramowanie użytkowe do edycji obrazu 2) stosować narzędzia i funkcje programu do obróbki grafiki rastrowej i wektorowej 	<p>Rozdział 5. Grafika na stronie internetowej</p>

	<p>komputerowej</p> <ol style="list-style-type: none"> 4) identyfikuje różne formaty plików graficznych 5) stosuje różne modele barw 6) osadza tekst na grafice oraz dobiera jego krój i styl 7) korzysta z funkcji edytora grafiki wektorowej 8) korzysta z funkcji edytora grafiki rastrowej 9) wykonuje edycję plików graficznych na potrzeby stron internetowych 10) projektuje elementy graficzne dla strony internetowej 	<ol style="list-style-type: none"> 3) stosować zasady cyfrowego zapisu obrazu 4) tworzyć i edytować elementy graficzne na potrzeby strony internetowej 5) łączyć tekst z grafiką oraz wybierać jego parametry (krój, styl) 6) eksportować i zapisywać grafikę w różnych formatach graficznych 7) definiować kolor za pomocą nazwy, kodu szesnastkowego, zapisu dziesiętnego 8) przeliczać zdefiniowany kolor z zapisu szesnastkowego na dziesiętny i z dziesiętnego na szesnastkowy 	
<ol style="list-style-type: none"> 5) wykorzystuje elementy multimedialne na stronach internetowych 	<ol style="list-style-type: none"> 1) określa zasady komputerowego przetwarzania wideo i dźwięku przygotowanego na potrzeby strony internetowej 2) dobiera oprogramowanie do edycji obrazu ruchomego i dźwięku 3) wykonuje animacje na potrzeby stron internetowych 4) wykonuje materiały wideo na potrzeby stron internetowych 5) edytuje wideo i dźwięk na potrzeby stron internetowych 6) osadza materiały multimedialne na stronach internetowych 7) importuje materiały multimedialne do systemów zarządzania treścią (CMS) 	<ol style="list-style-type: none"> 1) dobierać i obsługiwać oprogramowanie użytkowe do edycji wideo, dźwięku i animacji 2) stosować narzędzia i funkcje programu do obróbki wideo, dźwięku i animacji 3) stosować zasady cyfrowego przetwarzania wideo i dźwięku oraz tworzenia animacji komputerowej na potrzeby strony internetowej 4) tworzyć i edytować wideo, dźwięk i animację na potrzeby strony internetowej 5) łączyć ze sobą wideo, dźwięk i animację 6) eksportować i zapisywać wideo, dźwięk i animację w różnych formatach multimedialnych 7) rozróżniać formaty plików multimedialnych 8) dołączać pliki multimedialne do stron wykonanych w systemach zarządzania 	<p>Rozdział 7. Animacja na stronie internetowej</p> <p>Rozdział 8. Dźwięk i wideo</p>

		treścią	
6) wykonuje strony internetowe zgodnie z projektami	<ol style="list-style-type: none"> 1) projektuje układ sekcji na stronie internetowej 2) analizuje projekt strony internetowej 3) tworzy strukturę strony internetowej zgodnie z projektem 4) dobiera paletę barw dla strony internetowej 5) dobiera czcionki dla strony internetowej 6) uwzględnia potrzeby użytkowników z różnymi niepełnosprawnościami przy projektowaniu stron internetowych, dostosowując np. kontrast, powiększenie, inne elementy wspomagające niepełnosprawnych 7) opisuje zasady i znaczenie wytycznych dotyczących ułatwień w dostępie do treści publikowanych w internecie 8) tworzy stronę zgodną z wytycznymi dotyczącymi ułatwień w dostępie do treści publikowanych w internecie 	<ol style="list-style-type: none"> 1) wykonywać projekt strony internetowej według różnych wzorów i szablonów 2) stosować zasady doboru barw na stronie internetowej 3) dobierać czcionki i jednostki ich wielkości na stronie internetowej 4) uwzględniać w projektach stron internetowych zasady i wytyczne dotyczące dostępności treści internetowych (WCAG 2.0) pod kątem osób niepełnosprawnych 5) dostosowywać strony internetowe do potrzeb osób niepełnosprawnych 	<p>Rozdział 6.</p> <p>Graficzny projekt strony internetowej</p>
7) stosuje reguły testowania, walidacji i optymalizacji stron internetowych	<ol style="list-style-type: none"> 1) testuje stronę internetową w różnych przeglądarkach 2) testuje responsywność strony internetowej 3) określa proces walidacji strony internetowej 4) dobiera narzędzia walidacji strony internetowej 5) dokonuje walidacji strony internetowej 6) optymalizuje stronę internetową 7) określa proces pozycjonowania strony 	<ol style="list-style-type: none"> 1) Stosować kolejne etapy procesu testowania i sprawdzania poprawności strony internetowej 2) wykorzystywać programy walidujące do kontroli poprawności kodu strony (HTML i CSS) 3) stosować metody i sposoby pozycjonowania strony internetowej 4) wykonywać optymalizację strony internetowej 	<p>Rozdział 10.</p> <p>Testowanie i publikowanie strony</p>

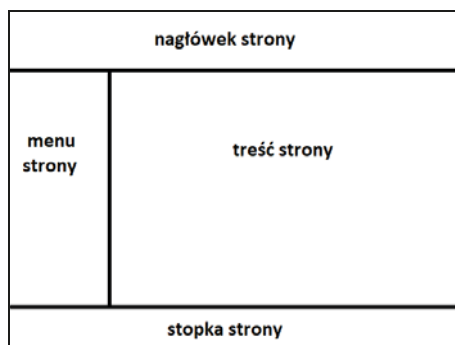
	internetowej 8) stosuje zasady dostępności (WCAG) i pozycjonowania strony internetowej	5) testować funkcjonalność i szybkość wczytywania strony 6) testować responsywność strony internetowej	
8) publikuje witryny i aplikacje internetowe	1) opisuje usługi hostingu 2) dobiera usługi hostingu w zależności od potrzeb użytkownika 3) opisuje operacje na domenach internetowych 4) wykonuje operacje na domenach internetowych 5) rozpoznaje etapy publikacji witryn i aplikacji internetowych 6) opisuje funkcje programów wykorzystywanych do przesyłania danych na serwer 7) dobiera program do przesyłania danych na serwer 8) przesyła dane na serwer 9) dobiera pakiety serwerowe WWW 10) sprawdza poprawność publikowanych stron WWW 11) publikuje witryny internetowe	1) korzystać z usług hostingowych stron internetowych 2) dobierać usługi hostingowe w zależności od potrzeb użytkownika 3) wykonywać operacje na domenach internetowych 4) stosować protokoły sieciowe do przesyłania i publikacji witryny internetowej 5) dobierać, instalować, konfigurować i obsługiwać programy z klientem FTP 6) publikować witryny i aplikacje internetowe 7) sprawdzać poprawność wyświetlania się i działania opublikowanych witryn i aplikacji internetowych	Rozdział 10. Testowanie i publikowanie strony

* Jolanta Pokorska, *Podręcznik do nauki zawodu technik informatyk, technik programista. Część 1., Tworzenie stron internetowych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu*

Planowane zadania

Zadanie 1.

- Wykonaj witrynę według następującego schematu (rysunek 1.), do budowy strony wykorzystaj znaczniki sekcji oraz zewnętrzne arkusze stylu.



Rysunek 1. Schemat strony internetowej

Zadanie 2.

Zastosuj następujące formatowanie tekstu dla danego akapitu:

- ustal wielkość czcionki na 1.5em,
- ustal wyjustowanie tekstu w akapicie,
- ustal pogrubienie czcionki,
- wprowadź wcięcie tekstu na poziomie 20px,
- ustal rodzaj czcionki Verdana,
- ustal kolor niebieski czcionki w zapisie heksadecymalnym.

Zadanie 3.

Zbuduj prosty formularz do kontaktu, zawierający następujące pola:

- dwa pola typu tekstowego (imię, nazwisko),
- jedno pole z datą urodzenia,
- jedno pole do wpisania adresu e-mail,
- jedno pole tekstowe do wpisania komentarza,
- dwa pola typu wyczyść i wyślij.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia stron internetowych* w części praktycznej powinny odbywać się w *Pracowni stron WWW, baz danych i aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji internetowych po stronie serwera i klienta w wybranych językach programowania, pakiety oprogramowania zawierające serwer WWW, SQL, PHP, serwer hostingowy do testowania projektów webowych), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętność tworzenia witryn internetowych za pomocą języków HTML i CSS, sprawnego tworzenia i modyfikowania elementów graficznych i multimedialnych na potrzeby stron internetowych oraz administrowania systemami zarządzania treścią w celu wykonania witryny internetowej za ich pomocą.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktazem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.8. Pracownia baz danych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) posługuje się pojęciami dotyczącymi baz danych	1) określa pojęcia związane z bazami danych: encja, związki encji, atrybuty encji, klucz relacji 2) określa typy danych używanych w bazach danych 3) stosuje odpowiednie typy danych przy definiowaniu encji 4) rozpoznaje postacie normalne baz danych 5) opisuje cechy relacyjnej bazy danych	1) posługiwać się pojęciami dotyczącymi baz danych, takimi jak klucz podstawowy, klucz obcy, integralność danych, typy związków 2) posługiwać się typami danych używanych w bazach danych 3) stosować postacie normalne bazy danych	
1) tworzy diagramy E/R (ang. <i>Entity-Relationship Diagram</i>)	1) charakteryzuje typy notacji diagramów E/R 2) rozróżnia bloki składowe diagramów E/R 3) analizuje diagramy E/R 4) definiuje encje i atrybuty encji 5) definiuje związki między encjami i określa ich liczebność 6) dobiera typ danych do określonych atrybutów encji 7) określa klucz główny dla encji	1) stosować podstawowe zasady projektowania baz danych 2) wykorzystywać podstawowe funkcje narzędzi CASE 3) tworzyć projekt bazy danych z zastosowaniem bloków składowych diagramów ERD 4) analizować diagramy ERD 5) definiować encje i atrybuty encji	

		<ul style="list-style-type: none"> 6) definiować związki między encjami 7) dobierać typ danych atrybutów encji 8) określać klucz główny dla encji 	
2) korzysta z systemów zarządzania bazami danych SZBD (ang. <i>Database Management System</i>)	<ul style="list-style-type: none"> 1) rozróżnia dostępne SZBD 2) dobiera SZBD do określonego zastosowania 3) instaluje SZBD 4) konfiguruje SZBD do pracy w środowisku wielu użytkowników 5) aktualizuje SZBD 	<ul style="list-style-type: none"> 1) zainstalować serwer MySQL i MS SQL 2) skonfigurować MySQL i MS SQL do pracy w środowisku wielu użytkowników 3) aktualizować MySQL i MS SQL 	
3) stosuje strukturalny język zapytań SQL (ang. <i>Structured Query Language</i>)	<ul style="list-style-type: none"> 1) opisuje polecenia języka SQL 2) stosuje polecenia języka SQL 3) definiuje struktury baz danych przy użyciu instrukcji języka zapytań 4) wyszukuje informacje w bazie danych przy użyciu języka SQL 5) zmienia rekordy w bazie danych przy użyciu języka SQL 6) usuwa rekordy w bazie danych przy użyciu języka SQL 7) tworzy skrypty w strukturalnym języku zapytań 	<ul style="list-style-type: none"> 1) stosować składnię języka SQL 2) stosować typy danych języka SQL 3) stosować polecenia tworzące strukturę bazy danych — język DDL 4) zmodyfikować dane w bazie przy użyciu języka SQL 5) usuwać dane z bazy przy użyciu języka SQL 6) wyszukiwać informacje w bazie danych przy użyciu języka SQL 7) tworzyć skrypty przy użyciu języka SQL 	
4) tworzy relacyjne bazy danych zgodnie z projektem	<ul style="list-style-type: none"> 1) definiuje tabele w bazie danych na podstawie projektu 2) definiuje typy danych oraz atrybuty kolumn 3) wprowadza dane do bazy danych 4) programuje skrypty automatyzujące proces tworzenia struktury bazy danych 	<ul style="list-style-type: none"> 1) utworzyć bazę danych na serwerze 2) utworzyć tabele w bazie danych zgodnie z projektem 3) zdefiniować typy danych oraz atrybuty kolumn 4) utworzyć skrypty do automatycznego 	

	<p>5) importuje dane z pliku</p> <p>6) eksportuje strukturę bazy danych i dane do pliku</p>	<p>tworzenia struktury bazy danych</p> <p>5) wprowadzić dane do bazy danych</p> <p>6) zaimportować dane z pliku</p> <p>7) wyeksportować strukturę bazy danych i dane do pliku</p>	
<p>5) tworzy formularze, zapytania i raporty do przetwarzania danych</p>	<p>1) tworzy formularze do wprowadzania i modyfikowania danych</p> <p>2) identyfikuje rodzaje zapytań</p> <p>3) tworzy zapytania i podzapytania do tabel bazy danych</p> <p>4) tworzy raporty w bazie danych</p>	<p>1) utworzyć formularze do wprowadzania i modyfikowania danych</p> <p>2) stworzyć zapytania do tabel bazy danych</p> <p>3) stosować klauzule instrukcji SELECT</p> <p>4) stosować funkcje agregujące</p> <p>5) pogrupować dane</p> <p>6) tworzyć podsumowania w bazie danych</p> <p>7) definiować połączenia w języku SQL</p> <p>8) definiować więzy integralności w języku SQL</p> <p>9) łączyć wyniki zapytań w języku SQL</p> <p>10) tworzyć podzapytania do tabel bazy danych</p>	
<p>6) modyfikuje struktury baz danych</p>	<p>1) analizuje strukturę bazy danych w celu jej modyfikacji</p> <p>2) rozbudowuje strukturę bazy danych, tworząc tabele, pola, relacje i atrybuty</p> <p>3) weryfikuje poprawność struktury bazy danych po rozbudowie</p> <p>4) usuwa elementy struktury bazy danych oraz dane</p> <p>5) modyfikuje strukturę bazy oraz dane bazy</p>	<p>1) analizować strukturę bazy danych w celu modyfikacji tej struktury</p> <p>2) zmodyfikować strukturę bazy</p> <p>3) rozbudować strukturę bazy danych, tworząc:</p> <ul style="list-style-type: none"> • nowe tabele • nowe pola w tabeli <p>4) modyfikować relacje i atrybuty kolumn</p> <p>5) usunąć elementy struktury bazy danych</p> <p>6) zweryfikować poprawność struktury bazy</p>	

		danych po modyfikacji 7) usunąć dane 8) zmodyfikować dane 9) dodać nowe dane	
7) zarządza systemem bazy danych	1) tworzy użytkowników bazy danych 2) określa uprawnienia dla użytkowników 3) kontroluje spójność bazy danych 4) tworzy kopię zapasową struktury bazy danych 5) weryfikuje poprawność kopii zapasowej bazy danych 6) przywraca dane z kopii zapasowej bazy danych 7) importuje i eksportuje tabele bazy danych 8) diagnozuje i naprawia bazę danych	1) tworzyć użytkowników bazy danych 2) tworzyć role w bazie danych 3) zdefiniować uprawnienia dla użytkowników 4) odebrać uprawnienia użytkownikowi 5) opracować strategię wykonania kopii bezpieczeństwa 6) skontrolować spójność bazy danych 7) utworzyć kopię zapasową bazy danych 8) zweryfikować poprawność kopii zapasowej bazy danych 9) przywrócić dane z kopii zapasowej bazy danych 10) zaimportować tabele bazy danych 11) wyeksportować tabele bazy danych do pliku 12) zdiagnozować i naprawić bazę danych	

* Jolanta Pokorska, *Podręcznik do nauki zawodu technik informatyk, technik programista. Część 2., Tworzenie i administrowanie bazami danych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu*

Planowane zadania

Zadanie 1.

Wykonaj projekt graficzny bazy danych zawierającej dane pracowników małej firmy. W projekcie powinny zostać zdefiniowane zbiory encji (przydzielone im atrybuty i typy danych) oraz rodzaje związków zachodzących między nimi. W bazie danych zostały wyodrębnione encje: *Pracownicy* (dane osobowe pracowników firmy), *Stanowiska* (informacje na temat stanowisk dostępnych w firmie: nazwa stanowiska, płaca na zajmowanym stanowisku), *Wykształcenie* (stopień wykształcenia), *Języki obce*.

Jeden pracownik może zajmować tylko jedno stanowisko, ale na jednym stanowisku może pracować wiele osób. Pracownik może mieć np. wykształcenie techniczne i ogólne, wiele osób może mieć takie samo wykształcenie. Pracownik może znać wiele języków, jeden język może znać wiele osób.

Zadanie 2.

Tabela *Klient* zawiera kolumny: *Nazwisko* (typ tekstowy), *Imię* (typ tekstowy), *Data_urodzenia* (typ tekstowy). Za pomocą poleceń języka SQL zmodyfikuj jej strukturę, dodając kolumnę *Id_klienta*, która będzie kluczem podstawowym automatycznie numerowanym od wartości 1, oraz zmień typ w kolumnie *Data_urodzenia* na typ daty.

Zadanie 3.

Baza danych *Magazyn* zawiera tabelę *Towar* z kolumnami *Id_towaru*, *Nazwa_towaru*, *Cena*, *Rok_produkcji*, *Ilość* oraz tabelę *Producent* z kolumnami *Id_producenta*, *Nazwa_producenta*, *Adres_firmy*, *Prezes*. Za pomocą poleceń języka SQL wyświetl informację o towarach, które zostały wyprodukowane w roku 2019. Podaj nazwę towaru, jego cenę, nazwę producenta oraz nazwisko prezesa firmy.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia baz danych* w części praktycznej powinny odbywać się w *Pracowni stron WWW, baz danych i aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy z programem MS Access lub Base, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji internetowych po stronie serwera i klienta w wybranych językach programowania, pakiety oprogramowania zawierające serwery baz danych), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności projektowania, tworzenia baz danych i administrowania nimi.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktazem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

2.9. Pracownia aplikacji internetowych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) stosuje zasady programowania	1) analizuje problemy programistyczne 2) stosuje algorytm 3) stosuje zasady programowania strukturalnego	1) scharakteryzować podstawowe pojęcia: aplikacja internetowa, skrypt, aplet 2) zastosować algorytm do opisu problemu programistycznego 3) wykorzystywać narzędzia programistyczne	
2) stosuje skryptowe języki programowania	1) stosuje języki JavaScript oraz jeden z: Python/ASP.NET/PHP/JSP do tworzenia aplikacji internetowych 2) identyfikuje skryptowe języki	1) zastosować język JavaScript do tworzenia aplikacji internetowych 2) zastosować język PHP do tworzenia aplikacji internetowych	

	<p>programowania</p> <ol style="list-style-type: none"> 3) implementuje algorytmy w języku interpretowanym 4) posługuje się typami prostymi i złożonymi, zmiennymi i operatorami w językach skryptowych 5) stosuje instrukcje sterujące języków skryptowych 6) stosuje funkcje oraz wybrane biblioteki języków skryptowych 7) tworzy strony internetowe wykorzystujące skryptowe języki programowania 	<ol style="list-style-type: none"> 3) zastosować język C# do tworzenia aplikacji internetowych 4) posługiwać się typami prostymi i złożonymi oraz zmiennymi przy tworzeniu aplikacji w językach JavaScript, PHP oraz C# 5) posługiwać się operatorami przy tworzeniu aplikacji w językach JavaScript, PHP oraz C# 6) zastosować instrukcje sterujące języków skryptowych 7) zastosować funkcje oraz wybrane biblioteki języków skryptowych 8) stworzyć strony internetowe wykorzystujące skryptowe języki programowania 	
<ol style="list-style-type: none"> 3) programuje skrypty wykonywane po stronie klienta 	<ol style="list-style-type: none"> 1) programuje w języku JavaScript 2) stosuje w programowaniu obsługę zdarzeń myszy i klawiatury 3) stosuje biblioteki wykorzystywane w skryptach po stronie klienta 4) definiuje skrypty obsługujące formularze i kontrolki HTML 5) wykorzystuje walidację formularzy HTML za pomocą mechanizmów HTMLS 6) korzysta z funkcji modelu DOM 7) korzysta z bibliotek i frameworków języka JavaScript, w tym z biblioteki jQuery, Angular, React 	<ol style="list-style-type: none"> 1) zastosować funkcje języka JavaScript 2) stworzyć obiekty w języku JavaScript 3) zastosować obiekty przeglądarki 4) zastosować obiekty języka JavaScript 5) zastosować obiekty dokumentu DOM 6) zastosować obsługę zdarzeń myszy i klawiatury 7) zastosować skrypty obsługujące formularze i kontrolki HTML 8) zastosować wyrażenia regularne do walidacji formularza 9) zastosować obsługę zdarzeń myszy i klawiatury za pomocą biblioteki jQuery 10) zastosować walidację formularzy z wykorzystaniem biblioteki jQuery 	

		11) tworzyć skrypty z wykorzystaniem biblioteki AngularJS 12) tworzyć skrypty z wykorzystaniem biblioteki React	
4) programuje skrypty wykonywane po stronie serwera	1) programuje w jednym z języków Python/ASP.NET/PHP/JSP 2) stosuje wbudowane instrukcje, funkcje 3) stosuje metody przesyłania danych z formularza 4) programuje wysyłanie danych z formularza HTML 5) stosuje biblioteki do obsługi bazy danych odpowiednie dla języka i frameworka 6) korzysta z funkcji do obsługi plików 7) korzysta z funkcji do obsługi ciasteczek (ang. <i>cookies</i>) oraz sesji	1) zastosować skrypt PHP w kodzie HTML 2) zastosować funkcje języka PHP 3) zastosować funkcje obsługi plików 4) tworzyć skrypty przekazujące dane z formularza 5) tworzyć pliki <i>cookies</i> 6) tworzyć skrypty wykorzystujące mechanizm sesji 7) zdefiniować połączenie z bazą danych 8) definiować zapytania do bazy danych 9) zastosować bibliotekę PDO do obsługi bazy danych	
5) stosuje środowisko programistyczne i uruchomieniowe aplikacji internetowych	1) opisuje funkcje środowiska programistycznego 2) dobiera środowisko programistyczne do określonych zadań i języka programowania 3) tworzy programy w wybranym środowisku programistycznym 4) instaluje i konfiguruje serwer WWW 5) instaluje i konfiguruje serwer baz danych 6) korzysta z gotowych pakietów dla aplikacji internetowych, np. phpMyAdmin	1) dobrać środowisko programistyczne do określonych zadań i języka programowania 2) tworzyć programy w wybranym środowisku programistycznym (XAMPP, Visual Studio) 3) instalować i konfigurować serwer WWW (Apache, ASP.NET) 4) instalować i konfigurować serwer MS SQL 5) instalować i konfigurować serwer MySQL 6) korzystać z pakietu phpMyAdmin 7) korzystać z pakietu Management Studio	
6) przeprowadza walidację kodu	1) analizuje błędy w kodzie źródłowym	1) analizować błędy w kodzie źródłowym	

programu	programu 2) wykonuje testy tworzonych programów 3) poprawia błędy w tworzonych programach 4) stosuje debugger w przeglądarce internetowej	programu 2) wykonać testy tworzonych programów 3) poprawić błędy w tworzonych programach 4) zastosować debugger w przeglądarce internetowej	
7) dokumentuje tworzoną aplikację	1) stosuje komentarze w kodzie źródłowym programu 2) tworzy dokumentację programu 3) tworzy instrukcję użytkownika programu	1) zastosować komentarze w kodzie źródłowym programu 2) utworzyć dokumentację programu 3) utworzyć instrukcję użytkownika programu	

** Jolanta Pokorska, Podręcznik do nauki zawodu technik informatyk, technik programista. Część 3., Tworzenie aplikacji internetowych. Kwalifikacja INF.03. Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, Helion — podręcznik w przygotowaniu*

Planowane zadania

Zadanie 1.

Przygotuj stronę internetową. W prawym górnym rogu strony umieść dowolną grafikę. Po wciśnięciu przycisku myszy nad grafiką powinna nastąpić zmiana na inną grafikę. Po zwolnieniu przycisku powinien nastąpić powrót do poprzedniej grafiki. W utworzonym skrypcie wykorzystaj metodę `getElementById()` języka JavaScript i zdarzenia myszy.

Zadanie 2.

W języku HTML opracuj formularz do rejestracji klientów sklepu internetowego w podanej postaci (rysunek 2.):

Dane osobowe

Nazwisko:
A] × Proszę o podanie przynajmniej 3 znaków.

Imię:
J Proszę o podanie przynajmniej 2 znaków.

Zawód:
Mu Proszę o podanie przynajmniej 3 znaków.

Adres e-mail:
kowal Proszę o podanie prawidłowego adresu email.

Telefon:
2345 Proszę o podanie przynajmniej 9 znaków.

Pracuję:

Wiadomość:
 To pole jest wymagane.

Rysunek 2. Formularz do rejestracji klientów

Za pomocą poleceń języka JavaScript oraz biblioteki jQuery przeprowadź walidację formularza. Po wpisaniu nieprawidłowych danych zostaną wyświetlone komunikaty podobne do pokazanych na rysunku.

Zadanie 3.

Wykorzystując pliki *cookies* napisz skrypt, który po otwarciu strony wyświetli komunikat *Witamy ponownie na naszej stronie. Ostatni raz odwiedziłeś nas <data ostatniej wizyty>*. W miejscu *<data ostatniej wizyty>* zostanie wstawiona wartość przechowywana w pliku *cookie*.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia aplikacji internetowych* w części praktycznej powinny odbywać się w *Pracowni stron WWW, baz danych i aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji internetowych po stronie serwera i klienta w wybranych językach programowania, pakiety oprogramowania zawierające serwery baz danych), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności projektowania i tworzenia aplikacji internetowych.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3. Programy nauczania dla kwalifikacji INF.04. Projektowanie, programowanie i testowanie aplikacji

3.1. Projektowanie oprogramowania

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) posługuje się prostymi typami danych	1) rozróżnia typy numeryczne stało- i zmiennoprzecinkowe 2) rozpoznaje typ logiczny 3) rozróżnia typy znakowe i łańcuchowe 4) posługuje się typami liczbowymi stało- i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym	1) omówić klasyfikację prostych typów danych 2) zidentyfikować daną typu prostego i określić jej charakter (numeryczna, znakowa, łańcuchowa, logiczna) 3) uściślić reprezentację i własności prostych typów danych 4) podać przykłady użycia różnych prostych typów danych	
2) posługuje się złożonymi typami danych	1) rozróżnia rodzaje złożonych typów danych 2) posługuje się tablicami jedno- i dwuwymiarowymi 3) posługuje się tablicami dynamicznymi, asocjacyjnymi 4) posługuje się typem rekordowym, np. strukturą, unią 5) posługuje się typem plikowym 6) posługuje się typem wskaźnikowym 7) charakteryzuje cechy kolekcji, w tym znaczenie iteratora	1) omówić klasyfikację złożonych typów danych 2) zidentyfikować daną typu złożonego i określić jej charakter (tablica, rekord, lista, plik i in.) 3) uściślić reprezentację i własności złożonych typów danych 4) podać przykłady użycia poszczególnych złożonych typów danych 5) określić specyfikację typów danych dla problemu programistycznego	

	<ul style="list-style-type: none"> 8) posługuje się kolekcjami, np. listami, kolejkami, stosami, wektorami 9) projektuje zestawy danych dla problemu programistycznego 		
3) stosuje metody rozwiązywania problemów za pomocą algorytmów	<ul style="list-style-type: none"> 1) projektuje algorytmy za pomocą różnych metod i narzędzi: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu 2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe 3) charakteryzuje algorytmy rekurencyjne 4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera 5) określa złożoność obliczeniową algorytmów 	<ul style="list-style-type: none"> 1) rozwiązać zadanie algorytmiczne 2) zapisać algorytm w wybranej formie/notacji (schemat blokowy, lista kroków, drzewo decyzyjne, pseudokod) 3) określić własności algorytmu (złożoność obliczeniową, własność STOP-u) 4) porównać efektywność różnych rozwiązań tego samego problemu programistycznego 5) rozróżnić i scharakteryzować algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe, rekurencyjne 6) podać klasyfikację technik projektowania algorytmów (np. metoda przyrostowa, dziel i zwyciężaj, programowanie dynamiczne, algorytmy heurystyczne, algorytmy ewolucyjne) 7) scharakteryzować problem obliczeniowy i podać algorytm jego rozwiązania (np. problem sortowania, wydawania reszty, wyznaczania najkrótszej ścieżki, wyznaczania maksymalnego przepływu, komiwojażera) 	
4) stosuje algorytmy sortowania i wyszukiwania	<ul style="list-style-type: none"> 1) charakteryzuje typy sortowania i ich złożoność obliczeniową 2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj oraz inne 	<ul style="list-style-type: none"> 1) scharakteryzować różne algorytmy sortowania i oszacować ich złożoność obliczeniową 2) ocenić efektywność różnych algorytmów sortowania 	

	<ul style="list-style-type: none"> 3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów 	<ul style="list-style-type: none"> 3) rozwiązać problem sortowania z wykorzystaniem różnych metod sortowania (np. bąbelkowego, szybkiego, kubełkowego, przez wstawianie, zachłannego, przez scalanie) 4) omówić różne metody wyszukiwania (np. sekwencyjne, binarne, interpolacyjne) 5) rozwiązać problem wyszukiwania dla różnych typów danych przy użyciu algorytmów wyszukiwania 	
<ul style="list-style-type: none"> 5) dobiera narzędzia i metodologie do planowania projektu i zarządzania nim 	<ul style="list-style-type: none"> 1) określa funkcje narzędzi do zarządzania projektem 2) stosuje diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta 3) korzysta z programów wspierających zarządzanie projektami, np. Jira, Trello 4) korzysta z systemu kontroli wersji, np. Git 	<ul style="list-style-type: none"> 1) wymienić i scharakteryzować główne aspekty zarządzania projektem (np. zasoby, czas, jakość) 2) określić działania wchodzące w skład zarządzania projektem (np. planowanie, wykonanie, kontrola) 3) identyfikować i charakteryzować narzędzia wspomagające zarządzanie projektem programistycznym (np. Trello, Wrike, Clarizen, Jira, ViewPath, KanbanTool) 4) podać przykłady narzędzi wizualizacji w zarządzaniu projektem (np. diagramy, dashboards) oraz narzędzi do wersjonowania oprogramowania (np. Git, SVN) 	
<ul style="list-style-type: none"> 6) projektuje aplikację 	<ul style="list-style-type: none"> 1) analizuje wymagania klienta i tworzy zgodny z nimi projekt 2) tłumaczy wymagania klienta na specyfikację techniczną dla zespołu programistów 3) identyfikuje elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widżety 4) projektuje interfejs użytkownika i wygląd 	<ul style="list-style-type: none"> 1) podać definicję i przykłady wymagań funkcjonalnych i нефункциональных aplikacji 2) wymienić i scharakteryzować elementy dokumentacji wymagań dla aplikacji (np. słownik, architekturę, indeks, wymagania systemowe, wymagania klienta, scenariusze) 3) ustalić wymagania klienta i na ich podstawie stworzyć opis wymagań dla projektu 	

	<p>aplikacji</p> <ol style="list-style-type: none"> 5) dostosowuje interfejs do różnych platform 6) projektuje aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym 7) projektuje aplikację opartą na architekturze klient-serwer 8) projektuje struktury danych dla aplikacji 9) projektuje funkcjonalność aplikacji 10) planuje system zabezpieczeń aplikacji 	<p>programistycznego</p> <ol style="list-style-type: none"> 4) opisać specyfikację techniczną dla zespołu programistów na podstawie wymagań klienta 5) określić składowe interfejsu użytkownika i zaprojektować jego wygląd 6) zaplanować funkcjonalność aplikacji 7) omówić rodzaje zabezpieczeń dla aplikacji 8) identyfikować i charakteryzować różne paradygmaty programowania (np. imperatywne, obiektowe, funkcyjne) 9) opisać budowę i zasadę działania aplikacji opartej na architekturze klient-serwer 10) określić struktury danych, algorytmy i zabezpieczenia dla danej aplikacji 	
<ol style="list-style-type: none"> 7) planuje przedsięwzięcie programistyczne 	<ol style="list-style-type: none"> 1) określa cel projektu 2) określa fazy realizacji projektu 3) charakteryzuje cykl życia projektu informatycznego i jego poszczególne etapy 4) określa zasoby ludzkie oraz ramy czasowe 5) planuje etapy tworzenia aplikacji 6) korzysta z metodologii zarządzania projektem: modelu kaskadowego (ang. <i>waterfall</i>), modelu przyrostowego, modelu prototypowego, metodyk zwinnych (Agile oraz przynajmniej jednej z: Scrum, Lean, Kanban) 7) dobiera optymalną metodologię zarządzania projektem 8) organizuje prace projektowe 	<ol style="list-style-type: none"> 1) opisać cel projektu 2) zaplanować fazy realizacji projektu 3) scharakteryzować pojęcie cyklu życia projektu informatycznego i omówić jego etapy 4) określić zakres, koszty, zasoby ludzkie i ramy czasowe dla projektu 5) przypisać role i zaplanować w czasie etapy prac w procesie tworzenia aplikacji (stworzyć harmonogram przedsięwzięcia) 6) rozróżnić i scharakteryzować metodologie zarządzania projektem informatycznym (metodyki zwinne Agile oraz np. RUP, Kanban, Scrum, Lean) 7) wskazać optymalną metodologię zarządzania dla danego projektu i ocenić przewagę 	

	9) stosuje harmonogram czynności dla efektywnego osiągnięcia celów	obranej metodologii nad innymi	
8) stosuje wzorce projektowe	<ol style="list-style-type: none"> 1) dobiera wzorzec projektowy do zadania programistycznego 2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metodę szablonową (ang. <i>Template method</i>), Fasadę (ang. <i>Facade</i>), Kompozyt (ang. <i>Composite</i>) 	<ol style="list-style-type: none"> 1) sklasyfikować różne wzorce projektowe (np. Prototyp, Fasadę, Kompozyt, Most, Szablon, Obserwatora) i scharakteryzować je 2) podać przykłady użycia wzorców projektowych w zadaniach programistycznych o różnym charakterze 	
9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania	<ol style="list-style-type: none"> 1) rozróżnia autorskie prawa osobiste i majątkowe 2) określa czas trwania praw autorskich 3) określa konsekwencje naruszenia prawa autorskiego 4) charakteryzuje elementy własności intelektualnej (dobra niematerialne, własności przemysłowe) 5) rozróżnia typy licencji oprogramowania 	<ol style="list-style-type: none"> 1) zidentyfikować zakres odpowiedzialności zawodowej programisty 2) rozróżnić prawa autorskie osobiste i majątkowe 3) wymienić konsekwencje naruszeń w zakresie praw autorskich osobistych i majątkowych 4) zidentyfikować i omówić elementy własności intelektualnej (jak dobra niematerialne, własność przemysłowa) 5) sklasyfikować różne typy licencji oprogramowania 6) uwzględnić w projektowaniu aplikacji kanony postępowania (ang. <i>codes of ethics</i>) 	

Planowane zadania

Zadanie 1.

Wyjaśnij zasady dostępu do elementów w kolekcjach (listach, stosach, wektorach) i wskaż kluczowe różnice.

Zadanie 2.

Omów na wybranym przykładzie technikę projektowania algorytmów dziel i zwyciężaj.

Zadanie 3.

Omów podstawowe cechy wzorców projektowych Fasada i Most. Wskaż ich przykładowe użycie w aplikacjach.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Projektowanie oprogramowania* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat projektowania algorytmów i aplikacji za pomocą narzędzi wspierających projektowanie, a także zarządzania projektami informatycznymi z zastosowaniem przeznaczonych do tego narzędzi i metod.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, wzorce aplikacji internetowych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.2. Podstawy programowania

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio) 2) stosuje środowiska IDE, RAD 3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych	1) scharakteryzować środowiska programistyczne (Visual Studio, CodeBlocks) 2) dobrać środowisko programistyczne do programowania w języku C++ 3) dobrać środowisko programistyczne do programowania w języku C#	
2) wykorzystuje frameworki do programowania aplikacji desktopowych	1) charakteryzuje pojęcie framework 2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt	1) scharakteryzować pojęcie framework 2) scharakteryzować frameworki WPF, Qt 3) podać przykłady zastosowania frameworków WPF, Qt	
3) programuje desktopowe aplikacje okienkowe	1) rozróżnia elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki) 2) stosuje języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python 3) stosuje język do projektowania interfejsu użytkownika, np. XAML	1) scharakteryzować elementy interfejsu użytkownika 2) scharakteryzować funkcje języka C++ 3) scharakteryzować funkcje języka C# 4) podać przykłady stosowania interfejsu użytkownika 5) podać sposoby projektowania menu aplikacji	

	4) programuje okna aplikacji 5) programuje system menu aplikacji 6) programuje okna dialogowe aplikacji 7) programuje obsługę zdarzeń myszy i klawiatury	6) scharakteryzować sposób tworzenia okien dialogowych aplikacji	
--	---	--	--

Planowane zadania

Zadanie 1.

Na czym polega rzutowanie zmiennych w języku C++? Wymień sposoby rzutowania w języku C++.

Zadanie 2.

Omów sposoby wykorzystania frameworków do tworzenia aplikacji.

Zadanie 3.

Omów przeznaczenie kontrolki do tworzenia aplikacji okienkowej w języku C#.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Podstawy programowania* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat różnych środowisk programistycznych, funkcji wybranych języków programowania aplikacji okienkowych, zasad korzystania z frameworków.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, wzorce aplikacji internetowych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.3. Programowanie obiektowe

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych	1) rozróżnia kompilatory i interpretery 2) charakteryzuje zadania kompilatora, interpretera, debugera 3) analizuje błędy w kodzie za pomocą	1) objaśnić pojęcia kompilator, interpreter oraz zasady ich działania 2) zidentyfikować i scharakteryzować zadania realizowane przez kompilator, interpreter i debuger	

	<p>debugera</p> <p>4) charakteryzuje etapy kompilacji i interpretacji kodu</p> <p>5) charakteryzuje pojęcie biblioteki</p> <p>6) kompiluje i uruchamia programy</p>	<p>3) opisać etapy kompilacji i interpretacji kodu</p> <p>4) objaśnić pojęcie biblioteka (moduł) oraz omówić rolę biblioteki w programowaniu</p>	
2) przestrzega zasad programowania	<p>1) dzieli program na funkcje (metody)</p> <p>2) stosuje rekurencję</p> <p>3) implementuje algorytmy w programie</p>	<p>1) podać definicję funkcji (metody) oraz opisać prototyp funkcji w danym języku programowania</p> <p>2) identyfikować funkcję rekurencyjną i określać poziom zagnieżdżenia rekurencji</p> <p>3) projektować program o strukturze modułowej</p>	
3) korzysta z typów danych	<p>1) stosuje proste i złożone typy danych</p> <p>2) deklaruje własne typy danych</p> <p>3) deklaruje zmienne różnych typów</p> <p>4) wykonuje operacje na zmiennych: wejścia-wyjścia, arytmetyczne, logiczne</p> <p>5) stosuje typy złożone i operacje na nich</p>	<p>1) identyfikować proste i złożone typy danych</p> <p>2) określać własne typy danych</p> <p>3) stosować zmienne różnych typów, zarówno wbudowanych, jak i własnych</p> <p>4) określać sposoby wymiany danych z programem (operacje wejścia-wyjścia)</p> <p>5) stosować elementarne operacje na zmiennych (arytmetyczne, logiczne)</p>	
4) stosuje wyrażenia, instrukcje i biblioteki	<p>1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, bitowe, operatory do obsługi łańcuchów</p> <p>2) wykorzystuje priorytety operatorów do właściwego budowania wyrażeń</p> <p>3) stosuje instrukcję warunkową i wyboru</p> <p>4) stosuje instrukcje pętli</p> <p>5) korzysta z wybranych bibliotek języka C++/ lub C# lub Python lub innego popularnego</p>	<p>1) stosować różne rodzaje operatorów (przypisania, arytmetyczne, logiczne, porównania, bitowe, specjalne)</p> <p>2) omówić priorytety operatorów i zasady poprawnego budowania wyrażeń złożonych</p> <p>3) określić składnię instrukcji sterowania przepływem danych (warunkową, wyboru, pętle)</p> <p>4) identyfikować wybrane biblioteki języków</p>	

	języka programowania: biblioteki standardowej, biblioteki z funkcjami matematycznymi, biblioteki z podstawowymi algorytmami	programowania, np. C++, C#, Java, Python lub innego popularnego języka programowania (m.in. bibliotekę standardową, bibliotekę funkcji matematycznych, bibliotekę z podstawowymi algorytmami)	
5) stosuje zasady programowania obiektowego	<ol style="list-style-type: none"> 1) stosuje obiektowe podejście do rozwiązywania problemów 2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm 3) dzieli zagadnienie na klasy 4) powołuje obiekty 5) planuje aplikację z zastosowaniem hermetyzacji, dziedziczenia i polimorfizmu 	<ol style="list-style-type: none"> 1) stosować paradygmaty programowania obiektowego 2) wyjaśnić i zilustrować przykładami kluczowe pojęcia z zakresu programowania obiektowego (klasa, obiekt, metoda, dziedziczenie, hermetyzacja, polimorfizm) 3) określić klasy reprezentujące zbiory obiektów w danym zagadnieniu 4) zdefiniować obiekty jako instancje klasy oraz wywołać metody klasy na rzecz obiektu 5) projektować aplikację, stosując dziedziczenie, hermetyzację i polimorfizm 	
6) definiuje klasy	<ol style="list-style-type: none"> 1) definiuje pola klasy 2) określa zakres widzialności pól klasy i definiuje kwalifikatory dostępu 3) definiuje metody klasy 4) definiuje konstruktory (w tym kopiujący) i destruktor klasy 5) definiuje instrukcje inicjujące konstruktora 6) określa zakres widzialności metod klasy i definiuje kwalifikatory dostępu 7) implementuje funkcjonalność klasy 8) deklaruje obiekty i odwołuje się obiektem do składowych klasy 	<ol style="list-style-type: none"> 1) zdefiniować pola (właściwości) klasy 2) zdefiniować metody klasy 3) zdefiniować konstruktor i destruktor klasy oraz wyjaśnić ich rolę w programowaniu obiektowym 4) posługiwać się modyfikatorami zmiennych, klas i metod w celu definiowania dostępności pól, klas i metod 5) deklarować obiekty i operować ich składowymi 6) rozróżnić klasy bazowe, potomne oraz określić cechy klas (abstrakcyjnej, statycznej, zaprzyjaźnionej) 	

	<ul style="list-style-type: none"> 9) definiuje składniki statyczne klasy 10) rozróżnia klasy dziedziczone i zaprzyjaźnione 11) tworzy funkcje zaprzyjaźnione z klasą, stosuje składnik statyczny klasy i metody do ich obsługi 	<ul style="list-style-type: none"> 7) definiować klasy i funkcje zaprzyjaźnione oraz posługiwać się nimi 	
7) definiuje klasy pochodne	<ul style="list-style-type: none"> 1) buduje hierarchię dziedziczenia klas w programie 2) wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia 3) definiuje klasy bazowe i pochodne 4) stosuje metody wirtualne, definiuje klasy abstrakcyjne 	<ul style="list-style-type: none"> 1) projektować strukturę programu z zastosowaniem hierarchii dziedziczenia klas 2) określać przynależność pól i metod do odpowiednich klas w hierarchii dziedziczenia 3) definiować klasy bazowe i klasy potomne 4) posługiwać się metodami wirtualnymi i klasami abstrakcyjnymi 	
8) programuje szablony (wzorce) klas	<ul style="list-style-type: none"> 1) definiuje szablony klas do obsługi prostych typów liczbowych 	<ul style="list-style-type: none"> 1) wyjaśnić rolę szablonów klas w programowaniu obiektowym 2) opisywać szablony klas dla prostych typów liczbowych 	
9) programuje obsługę wyjątków	<ul style="list-style-type: none"> 1) stosuje szkielet obsługi wyjątków z instrukcjami <code>try</code> i <code>catch</code> 2) stosuje instrukcję <code>throw</code> 3) opracowuje listę możliwych błędów wykonania aplikacji 4) definiuje obsługę błędów wykonania aplikacji 	<ul style="list-style-type: none"> 1) omówić mechanizm obsługi wyjątków w programie 2) określić szkielet obsługi wyjątków z instrukcjami <code>try</code> i <code>catch</code> 3) zdefiniować wyjątki przy użyciu instrukcji <code>throw</code> 4) sklasyfikować rodzaje wyjątków w aplikacji 5) ustalić listę potencjalnych błędów w działaniu programu 6) określić obsługę błędów wykonania programu 	

Planowane zadania

Zadanie 1.

Wyjaśnij na przykładzie, na czym polega dziedziczenie klas w programowaniu obiektowym.

Zadanie 2.

Wymień i scharakteryzuj modyfikatory dostępu w programowaniu obiektowym.

Zadanie 3.

Omów właściwości oraz zastosowanie klas i metod abstrakcyjnych.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Programowanie obiektowe* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat kluczowych wyznaczników programowania obiektowego, jakimi są dziedziczenie, hermetyzacja i polimorfizm, zasad określania widoczności pól, metod i klas za pomocą modyfikatorów dostępu oraz stosowania klas do obiektowego tworzenia aplikacji.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, wzorce aplikacji internetowych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.4. Programowanie aplikacji mobilnych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode) 2) stosuje środowiska IDE, RAD 3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych	1) wymienić różne środowiska programistyczne do tworzenia aplikacji mobilnych 2) rozróżniać środowiska programistyczne do tworzenia aplikacji mobilnych ze względu na różne urządzenia i platformy systemowe 3) wyjaśnić pojęcia zintegrowane środowisko programistyczne (IDE) i środowisko do szybkiego tworzenia oprogramowania (RAD) 4) dobrać środowisko i narzędzia programistyczne w celu wytworzenia aplikacji mobilnej zgodnie z określonym systemem operacyjnym i urządzeniami 5) stosować zintegrowane środowiska programistyczne (IDE) oraz narzędzia do szybkiego tworzenia aplikacji użytkowych	

		<p>(RAD) w procesie tworzenia aplikacji mobilnej</p> <ol style="list-style-type: none"> 6) opisać strukturę aplikacji mobilnej i jej komponenty 7) wymienić kluczowe parametry aplikacji mobilnej i narzędzia ich monitorowania 8) dobrać system kontroli wersji dla wybranego środowiska programistycznego 9) omówić sposoby przygotowania aplikacji do dystrybucji z wykorzystaniem środowiska programistycznego 	
2) programuje aplikacje mobilne	<ol style="list-style-type: none"> 1) programuje aplikacje w jednym z systemów mobilnych: iOS lub Android 2) stosuje języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C# 3) programuje przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych 4) identyfikuje elementy UI aplikacji mobilnej (przyciski, nawigację, okna dialogowe, listy, formularze, paski narzędziowe, grafikę, animacje, dźwięk) 5) programuje interfejs użytkownika za pomocą języka XAML 6) programuje aplikacje mobilne z wykorzystaniem elementów UI 7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) 	<ol style="list-style-type: none"> 1) klasyfikować języki programowania dla aplikacji mobilnych ze względu na systemy mobilne 2) stosować konstrukcje wybranego języka programowania do zapisu kodu natywnej aplikacji mobilnej na system iOS lub Android 3) omówić sposoby przechowywania danych i dostępu do nich w aplikacji mobilnej (m.in. prostych danych, danych wewnętrznych i zewnętrznych, plików zasobów, baz danych) 4) stosować operacje na danych i plikach z danymi dostępnymi zarówno wewnętrznie, jak i zewnętrznie 5) przedstawić sposoby walidacji i obsługi danych wejściowych 6) dobrać API do połączeń z bazą danych 7) scharakteryzować różne szablony interfejsu aplikacji mobilnej i elementy UI (przyciski, nawigację, okna dialogowe, listy, formularze, paski narzędziowe, grafikę, animacje, 	

	8) pobiera i wysyła dane z/do internetu dla aplikacji mobilnej 9) programuje aplikację mobilną korzystającą z bazy danych 10) tworzy aplikacje dostosowane do danej platformy mobilnej (np.: system Android smartfon, system Android Tablet, IPAD, iPhone) 11) uruchamia aplikacje mobilne 12) przygotowuje aplikacje do publikacji w sklepie	dźwięk) 8) wyjaśnić sposoby tworzenia animacji na potrzeby aplikacji mobilnych 9) stosować język hipertekstowy XAML do opisu interfejsu graficznego aplikacji mobilnej 10) zidentyfikować i opisać zasoby aplikacji mobilnej oraz zależności między nimi 11) zidentyfikować i scharakteryzować sensory i sprzętowe kontrolery mediów w urządzeniach mobilnych 12) omówić etapy przygotowania aplikacji mobilnej do publikacji w sklepie	
--	---	--	--

Planowane zadania

Zadanie 1.

Omów strukturę kodu aplikacji mobilnej oraz zasady jego uruchamiania w urządzeniu mobilnym.

Zadanie 2.

Scharakteryzuj sposoby dostępu do danych w aplikacji mobilnej.

Zadanie 3.

Wymień i scharakteryzuj główne kategorie sensorów obsługiwanych przez urządzenia i aplikacje mobilne.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Programowanie aplikacji mobilnych* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat programowania aplikacji mobilnych w wybranych językach programowania i środowiskach programistycznych, zasad projektowania efektywnego interfejsu graficznego aplikacji, tworzenia i optymalizacji graficznych elementów UI, wykorzystania API w celu integracji aplikacji z bazami danych oraz zagadnień dotyczących programowania funkcjonalności aplikacji związanych ze sprzętowymi właściwościami urządzeń mobilnych oraz preferencjami użytkownika.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, rysunki, ilustracje, fotografie, nagrania dźwiękowe, nagrania wideo, animacje komputerowe. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.5. Zaawansowane aplikacje webowe

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		

<p>1) wykorzystuje środowisko programistyczne dla zaawansowanych aplikacji webowych</p>	<p>1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)</p> <p>2) stosuje środowiska IDE, RAD</p> <p>3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych</p>	<p>1) scharakteryzować środowiska programistyczne aplikacji internetowych (Visual Studio, Eclipse)</p> <p>2) opisać wybrane narzędzia programistyczne stosowane po stronie klienta i po stronie serwera (XAMPP, Visual Studio)</p>	
<p>2) wykorzystuje frameworki do programowania aplikacji webowych</p>	<p>1) wykorzystuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js (do wyboru po jednym do technologii backend i technologii frontend)</p> <p>2) wykorzystuje bibliotekę jQuery lub jej odpowiednik</p>	<p>1) scharakteryzować platformy ASP.NET 4.x i ASP.NET Core</p> <p>2) scharakteryzować framework Django</p> <p>3) scharakteryzować bibliotekę jQuery</p> <p>4) scharakteryzować bibliotekę AngularJS</p> <p>5) scharakteryzować bibliotekę React</p>	
<p>3) programuje zaawansowane aplikacje webowe</p>	<p>1) stosuje języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript</p> <p>2) stosuje mechanizm sesji użytkownika, ciasteczek</p> <p>3) programuje dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny</p> <p>4) programuje aplikacje korzystające z bazy danych</p> <p>5) programuje wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego i innych</p>	<p>1) opisać funkcje języka C#</p> <p>2) opisać funkcje języka PHP</p> <p>3) scharakteryzować pliki <i>cookies</i></p> <p>4) scharakteryzować mechanizm sesji</p> <p>5) opisać sposób połączenia z bazą danych</p> <p>6) opisać sposób definiowania zapytań do bazy danych</p> <p>7) scharakteryzować bibliotekę PDO</p>	

Planowane zadania

Zadanie 1.

Wymień i omów popularne środowiska programistyczne aplikacji internetowych.

Zadanie 2.

Wymień i omów podstawowe cechy platformy ASP.NET Core.

Zadanie 3.

W jaki sposób w aplikacji internetowej realizowane jest połączenie z bazą danych?

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Zaawansowane aplikacje webowe* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat architektury aplikacji internetowych, identyfikowania skryptowych języków programowania oraz ich funkcjonalności, zasad korzystania z bibliotek i frameworków.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, zrzuty ekranu, wzorce aplikacji internetowych. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomagać w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.6. Testowanie i dokumentowanie aplikacji

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) przeprowadza walidację kodu programu	1) dobiera narzędzia i środowisko do testowania programów 2) wykonuje testy tworzonych programów 3) identyfikuje błędy w tworzonych programach 4) poprawia błędy w tworzonych programach, skryptach 5) optymalizuje kod źródłowy	1) wymienić narzędzia i środowisko do testowania danego rodzaju aplikacji 2) omówić główne cele i fazy procesu testowania aplikacji (statyczna inspekcja kodu, projektu i wymagań, dynamiczne testowanie, alfa i beta testy) 3) rozróżnić i wyjaśnić pojęcia weryfikacja i walidacja oprogramowania 4) opisać rolę wariantów testowych 5) identyfikować cechy kodu generujące błędy (semantyczne i syntaktyczne) 6) omówić sposoby optymalizacji kodu źródłowego z uwzględnieniem norm	
2) dokumentuje tworzoną aplikację	1) stosuje komentarze w kodzie źródłowym	1) wymienić i scharakteryzować rodzaje	

	<p>programu</p> <ol style="list-style-type: none"> 2) pisze dokumentację kodu 3) pisze dokumenty pomocy programu 4) pisze instrukcję użytkownika aplikacji 5) pisze instrukcję użytkownika programu 6) pisze dokumentację wdrożenia projektu 7) przygotowuje dokumentację z wykonanych testów 	<p>dokumentacji aplikacji</p> <ol style="list-style-type: none"> 2) omówić zasady tworzenia poszczególnych rodzajów dokumentacji 3) stosować komentarze w kodzie źródłowym aplikacji zgodnie z danym językiem programowania oraz konwencją DocBlocks 4) wymienić narzędzia automatyzacji tworzenia dokumentacji technicznej do danej aplikacji 	
3) przeprowadza testy	<ol style="list-style-type: none"> 1) charakteryzuje metodyki testowania 2) przygotowuje testy funkcjonalne i niefunkcjonalne 3) przeprowadza testy funkcjonalne 4) przeprowadza testy w kolejnych fazach projektu informatycznego 5) przeprowadza testy niefunkcjonalne: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa 6) korzysta z systemów raportowania błędów, np. BugZilla, JIRA 7) przeprowadza testy interfejsu 8) testuje prototyp projektu interfejsu 9) przygotowuje środowiska testowe 10) planuje scenariusze testowania aplikacji 11) pisze scenariusze testowania aplikacji 12) raportuje błędy po przeprowadzonych testach 13) automatyzuje proces testowania 14) testuje aplikację 	<ol style="list-style-type: none"> 1) sklasyfikować i scharakteryzować rodzaje testów aplikacji 2) rozróżnić techniki testowania aplikacji 3) omówić fazy w procesie testowania i ich role 4) scharakteryzować testy funkcjonalne i niefunkcjonalne 5) wymienić różne systemy raportowania błędów 6) określić dokumenty dla poszczególnych faz procesu testowania (listy kontrolne, scenariusze, raporty i in.) 7) wskazać narzędzia do zarządzania testami i frameworki do automatyzacji procesu testowania 	

	15) weryfikuje zgodność aplikacji z oczekiwaniami klienta		
--	---	--	--

Planowane zadania

Zadanie 1.

Omów cele i sposoby tworzenia dokumentacji technicznej aplikacji.

Zadanie 2.

Wymień i scharakteryzuj testy funkcjonalne i нефункционалне aplikacji.

Zadanie 3.

Scharakteryzuj sposoby, narzędzia i cele automatyzacji testów.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Testowanie i dokumentowanie aplikacji* w części teoretycznej powinny odbywać się w sali wykładowej wyposażonej w komputer z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne), oraz w projektor, ekran, głośniki, co ma ułatwić nauczycielowi stosowanie różnych metod przekazu materiału nauczania, ze szczególnym uwzględnieniem metod i środków aktywizujących uczniów w procesie dydaktycznym. Stworzone warunki powinny umożliwić opanowanie przez ucznia wiedzy teoretycznej z zakresu przedmiotu, tj. wiedzy na temat testowania i dokumentowania różnego rodzaju aplikacji zarówno metodą ręczną, jak też zautomatyzowaną w celu sprawnego wykrywania defektów, a także na temat metod optymalizowania kodu aplikacji zgodnie z przyjętymi normami i wytwarzania dokumentacji na potrzeby zespołu programistów, wdrożeniowców i użytkowników.

Środki dydaktyczne

Podczas realizacji zajęć teoretycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, które umożliwią i wzmocnią przekaz wzrokowo-słuchowy, np. prezentacje multimedialne, filmy dydaktyczne, rysunki, ilustracje, fotografie, nagrania dźwiękowe, nagrania wideo, animacje komputerowe. Stosowane środki dydaktyczne powinny pomóc w zapamiętaniu i zrozumieniu zagadnień poruszanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć teoretycznych powinny dominować metody oparte na słowie i na obserwacji, np. wykład konwersatoryjny, wykład problemowy, pogadanka, dyskusja dydaktyczna, pokaz multimedialny, quiz. Metody te powinny pomóc w zainteresowaniu uczniów tematem poruszonym podczas zajęć dydaktycznych, zachęcić do poszukiwania odpowiedzi na pojawiające się pytania, angażować w proces poznawania i utrwalania nowych wiadomości.

Formy organizacyjne

Zajęcia teoretyczne powinny być prowadzone w formie pracy z całym zespołem klasowym, z możliwością pracy w małych zespołach lub pracy indywidualnej. Wymagania edukacyjne powinny być dostosowane do zróżnicowanego potencjału uczniów wchodzących w skład zespołu klasowego. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy, co pozwoli na rzeczywiste rozwinięcie potencjału wszystkich uczniów przy zachowaniu możliwości indywidualizacji pracy z uczniem podczas zajęć edukacyjnych.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji przyrostu wiedzy i znajomości zagadnień z przedmiotu, np. odpowiedzi ustnych, sprawdzianów wiedzy, oceny udziału w dyskusji, ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do nauki, przygotowywać materiał dydaktyczny w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.7. Pracownia projektowania oprogramowania

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) posługuje się prostymi typami danych	1) rozróżnia typy numeryczne stało- i zmiennoprzecinkowe 2) rozpoznaje typ logiczny 3) rozróżnia typy znakowe i łańcuchowe 4) posługuje się typami liczbowymi stało- i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym	1) stosować typy numeryczne stało- i zmiennoprzecinkowe do reprezentacji wartości liczbowych 2) wykorzystać typ logiczny w projektowaniu oprogramowania 3) stosować typ znakowy i łańcuchowy do reprezentacji wartości napisowych 4) zastosować różne typy danych (numeryczny stało- i zmiennoprzecinkowy, logiczny,	

		znakowy, łańcuchowy) odpowiednio do reprezentowanych wartości w projektowaniu oprogramowania	
2) posługuje się złożonymi typami danych	<ol style="list-style-type: none"> 1) rozróżnia rodzaje złożonych typów danych 2) posługuje się tablicami jedno- i dwuwymiarowymi 3) posługuje się tablicami dynamicznymi, asocjacyjnymi 4) posługuje się typem rekordowym, np. strukturą, unią 5) posługuje się typem plikowym 6) posługuje się typem wskaźnikowym 7) charakteryzuje cechy kolekcji, w tym znaczenie iteratora 8) posługuje się kolekcjami, np. listami, kolejkami, stosami, wektorami 9) projektuje zestawy danych dla problemu programistycznego 	<ol style="list-style-type: none"> 1) stosować złożone typy danych do reprezentacji złożonych struktur danych w projektowaniu oprogramowania 2) wykorzystywać tablice jedno- i dwuwymiarowe, tablice dynamiczne i asocjacyjne w projektowaniu oprogramowania 3) stosować typ rekordowy (np. strukturę, unię) do reprezentowania struktur danych 4) stosować typ plikowy do projektowania obsługi plików 5) stosować typ wskaźnikowy do danych tworzonych dynamicznie 6) zastosować kolekcje (np. listy, kolejki, stosy, wektory) zgodnie z ich cechami 7) zastosować zestawy danych odpowiednie dla problemu programistycznego w projektowaniu oprogramowania 	
3) stosuje metody rozwiązywania problemów za pomocą algorytmów	<ol style="list-style-type: none"> 1) projektuje algorytmy za pomocą różnych metod i narzędzi: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu 2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe 3) charakteryzuje algorytmy rekurencyjne 4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera 	<ol style="list-style-type: none"> 1) projektować algorytmy, stosując różne metody/formy zapisu (schemat blokowy, listę kroków, drzewo decyzyjne, pseudokod) 2) opracować i wykorzystać algorytmy iteracyjne i rekurencyjne 3) zastosować algorytmy tablicowe, tekstowe i szyfrowania 4) stosować techniki projektowania algorytmów (np. metodę przyrostową, dziel i zwyciężaj, metodę heurystyczną, programowanie dynamiczne) 	

	5) określa złożoność obliczeniową algorytmów	5) oszacować złożoność obliczeniową algorytmów i wykazać własność STOP-u algorytmu	
4) stosuje algorytmy sortowania i wyszukiwania	<ol style="list-style-type: none"> 1) charakteryzuje typy sortowania i ich złożoność obliczeniową 2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj oraz inne 3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów 	<ol style="list-style-type: none"> 1) dobiera i stosować różne metody sortowania (np. bąbelkowe, szybkie, kubełkowe, przez wstawianie, zachłanne, przez scalanie) w projektowaniu oprogramowania 2) stosować algorytmy wyszukiwania dla różnych zestawów danych (tablic, list, kolejek, stosów) 	
5) dobiera narzędzia i metodologie do planowania projektu i zarządzania nim	<ol style="list-style-type: none"> 1) określa funkcje narzędzi do zarządzania projektem 2) stosuje diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta 3) korzysta z programów wspierających zarządzanie projektami, np. Jira, Trello 4) korzysta z systemu kontroli wersji, np. Git 	<ol style="list-style-type: none"> 1) dobrać narzędzia do zarządzania projektem 2) zastosować narzędzia wizualizacji w zarządzaniu etapami projektu, zadaniami i czasem (np. diagramy Gantta, dashboardy) 3) wykorzystać programy wspierające zarządzanie projektami programistycznymi (np. Trello, Wrike, Clarizen, Jira, ViewPath, KanbanTool) 4) zastosować narzędzia wersjonowania oprogramowania (np. Git, SVN) 	
6) projektuje aplikację	<ol style="list-style-type: none"> 1) analizuje wymagania klienta i tworzy zgodny z nimi projekt 2) tłumaczy wymagania klienta na specyfikację techniczną dla zespołu programistów 3) identyfikuje elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widżety 4) projektuje interfejs użytkownika i wygląd aplikacji 5) dostosowuje interfejs do różnych platform 	<ol style="list-style-type: none"> 1) analizować wymagania klienta względem aplikacji oraz projektować zgodnie z nimi aplikację 2) tworzyć specyfikację techniczną projektu dla zespołu programistów na podstawie wymagań klienta 3) projektować funkcjonalność aplikacji 4) projektować interfejs użytkownika i wygląd aplikacji z wykorzystaniem odpowiednich elementów interfejsu (jak okna, dialogi, kontrolki, formularze, paski narzędziowe, 	

	<ul style="list-style-type: none"> 6) projektuje aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym 7) projektuje aplikację opartą na architekturze klient-serwer 8) projektuje struktury danych dla aplikacji 9) projektuje funkcjonalność aplikacji 10) planuje system zabezpieczeń aplikacji 	<ul style="list-style-type: none"> widgety) 5) dostosować interfejs aplikacji do różnych platform 6) projektować aplikacje zgodnie z paradygmatami programowania (np. imperatywnym, obiektowym, funkcyjnym) 7) projektować aplikację opartą na architekturze klient-serwer 8) projektować struktury danych dla aplikacji 9) dobierać mechanizmy zabezpieczeń aplikacji 	
7) planuje przedsięwzięcie programistyczne	<ul style="list-style-type: none"> 1) określa cel projektu 2) określa fazy realizacji projektu 3) charakteryzuje cykl życia projektu informatycznego i jego poszczególne etapy 4) określa zasoby ludzkie oraz ramy czasowe 5) planuje etapy tworzenia aplikacji 6) korzysta z metodologii zarządzania projektem: modelu kaskadowego (ang. <i>waterfall</i>), modelu przyrostowego, modelu prototypowego, metodyk zwinnych (Agile oraz przynajmniej jedna z: Scrum, Lean, Kanban) 7) dobiera optymalną metodologię zarządzania projektem 8) organizuje prace projektowe 9) stosuje harmonogram czynności dla efektywnego osiągnięcia celów 	<ul style="list-style-type: none"> 1) zdefiniować cel/misję projektu 2) wyodrębnić fazy realizacji projektu 3) zdefiniować zakres, koszt, ramy czasowe i zasoby ludzkie dla projektu 4) wyznaczyć fazy tworzenia projektu 5) stosować metodologię zarządzania projektem informatycznym (metodyki zwinne Agile oraz np. RUP, Kanban, Scrum, Lean) 6) dobierać optymalną metodologię zarządzania projektem 7) zdefiniować role i wyznaczyć w czasie etapy prac w procesie tworzenia aplikacji (stworzyć harmonogram przedsięwzięcia) 8) stosować harmonogram czynności dla efektywnego osiągnięcia celów 	
8) stosuje wzorce projektowe	<ul style="list-style-type: none"> 1) dobiera wzorzec projektowy do zadania programistycznego 	<ul style="list-style-type: none"> 1) dobrać odpowiedni wzorzec projektowy do zadania programistycznego 	

	2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metodę szablonową (ang. <i>Template method</i>), Fasadę (ang. <i>Facade</i>), Kompozyt (ang. <i>Composite</i>)	2) stosować różne wzorce projektowe w programowaniu obiektowym (np. Prototyp, Fasadę, Kompozyt, Most, Szablon, Obserwatora)	
9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania	<ol style="list-style-type: none"> 1) rozróżnia autorskie prawa osobiste i majątkowe 2) określa czas trwania praw autorskich 3) określa konsekwencje naruszenia prawa autorskiego 4) charakteryzuje elementy własności intelektualnej (dobra niematerialne, własności przemysłowe) 5) rozróżnia typy licencji oprogramowania 	<ol style="list-style-type: none"> 1) przestrzegać przepisów prawa w zakresie autorskich praw osobistych i majątkowych w praktyce zawodowej 2) stosować oprogramowanie zgodnie z typem licencji 3) stosować w projektowaniu aplikacji kanony postępowania (ang. <i>codes of ethics</i>) 	

Planowane zadania

Zadanie 1.

Zaprojektuj algorytm, który dla zestawu danych wejściowych w postaci tablicy liczb całkowitych T (o wartościach losowych) zwróci wynik będący liczbą określającą, ile jest liczb parzystych w tablicy T . Oszacuj złożoność tego algorytmu i wykaż własność STOP-u oraz dobierz odpowiednie typy danych do realizacji tego algorytmu.

Zadanie 2.

Zaprojektuj algorytm, który zbada strukturę wejściowego łańcucha znakowego pod względem zgodności ze strukturą adresu e-mail. Wynikiem działania algorytmu ma być odpowiedź, czy podany łańcuch znakowy jest zgodny z formatem adresu e-mail, czy też nie. Dobierz odpowiednie typy danych do reprezentacji danych występujących w algorytmie oraz techniki przeszukiwania tekstu.

Zadanie 3.

Dokonaj analizy wymagań klienta podanych w formie historyjek użytkownika i sporządź projekt aplikacji zgodnej z jego wymaganiami. W opisie projektu określ architekturę aplikacji, docelową platformę sprzętowo-programową i funkcjonalność aplikacji. Wykonaj prototyp interfejsu użytkownika aplikacji. Określ metodologię wytwarzania aplikacji. Stosując program wspierający zarządzanie projektem, utwórz harmonogram prac tworzenia aplikacji i przedstaw graficznie w postaci np. diagramu etapy realizacji przedsięwzięcia w czasie.

Historyjki użytkownika:

- Klienci aplikacji mogą zakupić online miesięczne karnety do siłowni.

- Karnet miesięczny może być opłacony kartą kredytową.
- Karnet miesięczny może być opłacony przez system PayPal.
- Trenerzy personalni mogą wprowadzać notatki w indywidualnych kartach klienta.
- Klienci mogą otrzymać własny podstawowy zestaw i plan ćwiczeń.
- Klienci mogą zamówić specjalny plan treningowy i dietę.
- Klienci mogą zapisać się wyłącznie na treningi, do których nie mają przeciwwskazań.
- Harmonogram klienta jest dostępny online w standardowej przeglądarce po zalogowaniu.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia projektowania oprogramowania* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji w wybranych językach programowania, programy wspierające zarządzanie projektami programistycznymi, narzędzia wersjonowania oprogramowania) oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności projektowania algorytmów i aplikacji za pomocą narzędzi wspierających projektowanie, a także zarządzania projektami informatycznymi z zastosowaniem przeznaczonych do tego narzędzi i metod.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.8. Pracownia podstaw programowania

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio) 2) stosuje środowiska IDE, RAD 3) 3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych	1) dobrać środowisko programistyczne do programowania w języku C++ 2) dobrać środowisko programistyczne do programowania w języku C# 3) zastosować środowisko programistyczne dla języka C++ 4) zastosować środowisko programistyczne dla języka C#	
2) wykorzystuje frameworki do	1) charakteryzuje pojęcie framework	1) zastosować framework Qt do tworzenia	

programowania aplikacji desktopowych	2) wykorzystuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt	aplikacji w języku C++ 2) zastosować framework WPF do tworzenia aplikacji w języku C#	
3) programuje desktopowe aplikacje okienkowe	<ol style="list-style-type: none"> 1) rozróżnia elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki) 2) stosuje języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python 3) stosuje język do projektowania interfejsu użytkownika, np. XAML 4) programuje okna aplikacji 5) programuje system menu aplikacji 6) programuje okna dialogowe aplikacji 7) programuje obsługę zdarzeń myszy i klawiatury 	<ol style="list-style-type: none"> 1) zastosować funkcje języka C++ do tworzenia aplikacji desktopowych 2) zastosować funkcje języka C# do tworzenia aplikacji okienkowych 3) zaprojektować interfejs użytkownika 4) utworzyć interfejs użytkownika 5) zaprojektować okna aplikacji 6) utworzyć okna aplikacji 7) zaprojektować menu aplikacji 8) utworzyć menu aplikacji 9) zaprojektować okna dialogowe aplikacji 10) utworzyć okna dialogowe aplikacji 11) zaprojektować obsługę zdarzeń myszy i klawiatury 12) zaprogramować obsługę zdarzeń myszy i klawiatury 	

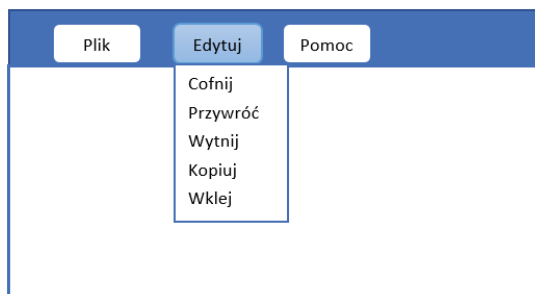
Planowane zadania

Zadanie 1.

Napisz program, który pozwoli na wprowadzenie ocen z matematyki 10 uczniów. Dla każdego ucznia można wprowadzić do 8 ocen. Po wprowadzeniu danych program policzy średnią ocen każdego ucznia i wyświetli informację o najwyższej i najniższej średniej.

Zadanie 2.

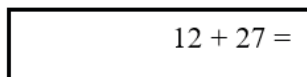
Wykorzystując jeden z poznanych frameworków, napisz program tworzący menu w postaci podanej na rysunku 3.



Rysunek 3. Okno menu tworzonej aplikacji

Zadanie 3.

Napisz program, którego zadaniem będzie wykonywanie podstawowych obliczeń na dwóch liczbach wprowadzonych przez użytkownika. Utworzona aplikacja będzie zawierała pole, w którym zostaną wyświetlone wprowadzone liczby oraz wykonywane na nich działanie w postaci podanej na rysunku 4. Dostępne będą również przyciski operacji: dodawania, odejmowania, dzielenia i mnożenia, oraz przycisk równa się, po którego naciśnięciu w polu aplikacji zostanie wyświetlony wynik wykonanego działania.



Rysunek 4. Okno tworzonej aplikacji

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia podstaw programowania* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji w wybranych językach programowania, pakiety oprogramowania zawierające serwer WWW, SQL, PHP, serwer hostingowy do testowania projektów webowych), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności stosowania różnych środowisk programistycznych oraz frameworków, programowania aplikacji okienkowych w języku C++ i C#.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.9. Pracownia programowania obiektowego

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych	1) rozróżnia kompilatory i interpretery 2) charakteryzuje zadania kompilatora,	1) korzystać z kompilatorów i interpreterów w środowiskach programistycznych	

	<p>interpretera, debugera</p> <ol style="list-style-type: none"> 3) analizuje błędy w kodzie za pomocą debugera 4) charakteryzuje etapy kompilacji i interpretacji kodu 5) charakteryzuje pojęcie biblioteki 6) kompiluje i uruchamia programy 	<ol style="list-style-type: none"> 2) uruchamiać kody źródłowe programów, stosując kompilatory i interpretery odpowiednio do języka programowania 3) analizować kod źródłowy programu przy użyciu debugera 4) tworzyć programy przy użyciu bibliotek (modułów) programowych odpowiednich dla danego języka programowania 	
2) przestrzega zasad programowania	<ol style="list-style-type: none"> 1) dzieli program na funkcje (metody) 2) stosuje rekurencję 3) implementuje algorytmy w programie 	<ol style="list-style-type: none"> 1) tworzyć program z podziałem na bloki kodu z zastosowaniem funkcji 2) programować przy użyciu rekurencji 3) implementować algorytmy w programie 	
3) korzysta z typów danych	<ol style="list-style-type: none"> 1) stosuje proste i złożone typy danych 2) deklaruje własne typy danych 3) deklaruje zmienne różnych typów 4) wykonuje operacje na zmiennych: wejścia-wyjścia, arytmetycznych, logicznych 5) stosuje typy złożone i operacje na nich 	<ol style="list-style-type: none"> 1) definiować zmienne różnych typów w programie (prostych i złożonych) 2) definiować własne typy danych na potrzeby programu 3) stosować zmienne różnych typów (prostych i złożonych) oraz operacje na nich 4) wykonywać operacje przepływu danych z użyciem zmiennych (operacje wejścia-wyjścia) 5) stosować elementarne operacje arytmetyczne i logiczne na zmiennych 	
4) stosuje wyrażenia, instrukcje i biblioteki	<ol style="list-style-type: none"> 1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, bitowe, operatory do obsługi łańcuchów, 2) wykorzystuje priorytety operatorów do właściwego budowania wyrażeń 3) stosuje instrukcję warunkową i wyboru 4) stosuje instrukcje pętli 5) korzysta z wybranych bibliotek języka C++/ lub C#/ lub Python lub innego popularnego języka programowania: biblioteki 	<ol style="list-style-type: none"> 1) wykonywać w programie operacje: <ul style="list-style-type: none"> • przypisania • arytmetyczne • porównania • logiczne • bitowe • do obsługi łańcuchów 2) posługiwać się operatorami zgodnie z ich 	

	standardowej, biblioteki z funkcjami matematycznymi, biblioteki z podstawowymi algorytmami	<p>priorytetami do właściwego budowania wyrażeń</p> <p>3) tworzyć kod programu przy użyciu instrukcji:</p> <ul style="list-style-type: none"> • warunkowej • wyboru • pętli <p>4) programować przy użyciu wybranych bibliotek (modułów) danego języka programowania (np. C++, C#, Java, Python lub innego popularnego języka programowania), w tym:</p> <ul style="list-style-type: none"> • biblioteki standardowej • biblioteki funkcji matematycznych • biblioteki z podstawowymi algorytmami 	
5) stosuje zasady programowania obiektowego	<p>1) stosuje obiektowe podejście do rozwiązywania problemów</p> <p>2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm</p> <p>3) dzieli zagadnienie na klasy</p> <p>4) powołuje obiekty</p> <p>5) planuje aplikację z zastosowaniem hermetyzacji, dziedziczenia i polimorfizmu</p>	<p>1) deklarować klasy i powoływać obiekty klas</p> <p>2) stosować modyfikatory dostępu dla klas, metod i pól</p> <p>3) tworzyć prototyp programu/aplikacji z zastosowaniem dziedziczenia, hermetyzacji i polimorfizmu</p> <p>4) implementować aplikację zgodnie z podejściem obiektowym</p>	
6) definiuje klasy	<p>1) definiuje pola klasy</p> <p>2) określa zakres widzialności pól klasy i definiuje kwalifikatory dostępu</p> <p>3) definiuje metody klasy</p>	<p>1) zapisywać definicję klasy, określając:</p> <ul style="list-style-type: none"> • pola klasy • modyfikatory dostępu dla pól klasy • metody klasy 	

	<ol style="list-style-type: none"> 4) definiuje konstruktory (w tym kopiujący) i destruktor klasy 5) definiuje instrukcje inicjujące konstruktora 6) określa zakres widzialności metod klasy i definiuje kwalifikatory dostępu 7) implementuje funkcjonalność klasy 8) deklaruje obiekty i odwołuje się obiektem do składowych klasy 9) definiuje składniki statyczne klasy 10) rozróżnia klasy dziedziczone i zaprzyjaźnione 11) tworzy funkcje zaprzyjaźnione z klasą, stosuje składnik statyczny klasy i metody do ich obsługi 	<ul style="list-style-type: none"> • konstruktory klasy (w tym kopiujący) • destruktor klasy • modyfikatory dostępu dla metod klasy • kod funkcjonalności klasy <ol style="list-style-type: none"> 2) deklarować obiekty klasy i operować na składowych klasy tych obiektów 3) stosować modyfikatory <code>static</code> oraz <code>friend</code> w tworzeniu klas, metod, pól i funkcji 	
7) definiuje klasy pochodne	<ol style="list-style-type: none"> 1) buduje hierarchię dziedziczenia klas w programie 2) wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia 3) definiuje klasy bazowe i pochodne 4) stosuje metody wirtualne, definiuje klasy abstrakcyjne 	<ol style="list-style-type: none"> 1) implementować program z wykorzystaniem dziedziczenia klas 2) definiować klasy bazowe i klasy potomne 3) stosować polimorfizm w definiowaniu metod odpowiednich klas w hierarchii dziedziczenia 4) stosować modyfikatory <code>abstract</code>, <code>final</code> oraz <code>virtual</code> przy tworzeniu odpowiednich klas i metod 	
8) programuje szablony (wzorce) klas	<ol style="list-style-type: none"> 1) definiuje szablony klas do obsługi prostych typów liczbowych 	<ol style="list-style-type: none"> 1) definiować szablony klas 2) parametryzować klasy prostymi typami liczbowymi przy użyciu szablonów klas 	
9) programuje obsługę wyjątków	<ol style="list-style-type: none"> 1) stosuje szkielet obsługi wyjątków z instrukcjami <code>try</code> i <code>catch</code> 2) stosuje instrukcję <code>throw</code> 3) opracowuje listę możliwych błędów wykonania aplikacji 	<ol style="list-style-type: none"> 1) definiować w programie bloki kodu przy użyciu instrukcji <code>try</code> i <code>catch</code> 2) ustalać i wykrywać w kodzie miejsca powstawania błędów wykonania aplikacji 	

	4) definiuje obsługę błędów wykonania aplikacji	3) określać w kodzie programu wyjątki przy użyciu instrukcji <code>throw</code>	
		4) definiować w programie kod obsługi błędów	

Planowane zadania

Zadanie 1.

Zdefiniuj nową klasę o nazwie *egzamin*, obejmującą pola: *nazwa_przedmiotu*, *data*, *godzina_rozporządzenia*, *godzina_zakończenia*, oraz metody: *start()*, która ustawia wartość pola *godzina_rozporządzenia*, i *stop()*, która ustawia wartość pola *godzina_zakończenia*, dodając 120 minut do wartości ustawionej w polu *godzina_rozporządzenia*. Pole *nazwa_przedmiotu* ustaw jako publiczne, do pozostałych pól określ dostęp prywatny.

Zadanie 2.

Stosując definicję klasy *egzamin* z zadania 1., zdefiniuj konstruktor i destruktor tej klasy. Następnie utwórz obiekt tej klasy o nazwie *e_programowanie* i nadaj poszczególnym polom wartości zgodnie z następującym opisem słownym: *egzamin z przedmiotu „Programowanie aplikacji” rozpocznie się dzisiaj o godz. 12:00 i zgodnie z regulaminem zakończy się po upływie 120 minut.*

Zadanie 3.

Stosując definicję klasy *egzamin* z zadania 1., zdefiniuj klasę potomną o nazwie *egz_praktyczny*, obejmującą dodatkowo pola *forma* i *opis* oraz własną metodę o nazwie *stop()*, która ustawia wartość pola *godzina_zakończenia*, dodając 150 minut do wartości ustawionej w polu *godzina_rozporządzenia*.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia programowania obiektowego* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji w wybranych językach programowania, oprogramowanie wspomagające wersjonowanie kodu źródłowego), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. tworzenie programów w wybranych językach programowania z uwzględnieniem paradygmatu programowania obiektowego.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.10. Pracownia programowania aplikacji mobilnych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)	1) instalować środowiska programistyczne do tworzenia aplikacji mobilnych	

	<ul style="list-style-type: none"> 2) stosuje środowiska IDE, RAD 3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych 	<ul style="list-style-type: none"> 2) dostosować i skonfigurować środowisko programistyczne do pracy nad aplikacjami mobilnymi 3) zarządzać pakietami rozszerzeń 4) stosować funkcje i usługi dostępne w środowiskach i narzędziach programistycznych w procesie wytwarzania aplikacji mobilnej 5) uruchomić wirtualne urządzenia i zarządzać nimi 6) strukturyzować foldery i pliki aplikacji 7) optymalizować kod aplikacji i monitorować jej wydajność za pomocą wbudowanych narzędzi 8) integrować środowisko programistyczne z systemem kontroli wersji 9) przygotować aplikację do dystrybucji 	
<ul style="list-style-type: none"> 2) programuje aplikacje mobilne 	<ul style="list-style-type: none"> 1) programuje aplikacje w jednym z systemów mobilnych: iOS lub Android 2) stosuje języki programowania przeznaczone dla aplikacji mobilnych w systemach iOS lub Android: Objective-C lub Swift lub Java lub C# 3) programuje przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych 4) identyfikuje elementy UI aplikacji mobilnej (przyciski, nawigację, okna dialogowe, listy, formularze, paski narzędziowe, grafikę, animacje, dźwięk) 5) programuje interfejs użytkownika za pomocą 	<ul style="list-style-type: none"> 1) stosować odpowiedni język programowania dla aplikacji mobilnych ze względu na systemy mobilne 2) zapisywać kod natywnej aplikacji mobilnej na system iOS lub Android w wybranym języku programowania 3) dobrać sposób przechowywania danych i dostępu do nich w aplikacji mobilnej (m.in. prostych danych, danych wewnętrznych i zewnętrznych, plików zasobów, baz danych) 4) wykonać operacje na danych i plikach z danymi dostępnymi wewnątrz i zewnątrz 5) walidować i obsługiwać dane wejściowe 	

	<p>języka XAML</p> <ol style="list-style-type: none"> 6) programuje aplikacje mobilne przy użyciu elementów UI 7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) 8) pobiera i wysyła dane z/do internetu dla aplikacji mobilnej 9) programuje aplikację mobilną korzystającą z bazy danych 10) tworzy aplikacje dostosowane do danej platformy mobilnej (np.: system Android smartfon, system Android Tablet, IPAD iPhone) 11) uruchamia aplikacje mobilne 12) przygotowuje aplikacje do publikacji w sklepie 	<ol style="list-style-type: none"> 6) stosować API do połączeń z bazami danych oraz do obsługi danych wejściowych od użytkownika 7) zaprojektować szablon interfejsu aplikacji mobilnej oraz jej funkcjonalność 8) przygotować elementy UI (przyciski, nawigację, okna dialogowe, listy, formularze, paski narzędziowe, grafikę, animacje, dźwięk) zgodnie z szablonem interfejsu aplikacji i jej funkcjonalnością 9) opracować różne szablony interfejsu aplikacji mobilnej i optymalizować je 10) przygotować interfejs graficzny aplikacji mobilnej z wykorzystaniem języka hipertekstowego XAML 11) stworzyć grafikę i animacje na potrzeby aplikacji mobilnej w oparciu o sceny i przejścia oraz biblioteki graficzne dla systemów wbudowanych (np. OpenGL ES, Core Graphics/Quartz) 12) implementować ustrukturyzowane zasoby aplikacji mobilnej dostosowane do mobilnej platformy systemowej i sprzętowej 13) zastosować monitoring zdarzeń i zaprogramować ich obsługę 14) implementować interakcję i obsługę użytkownika w oparciu o dane wejściowe z różnych sensorów (środowiskowych, pozycyjnych, ruchu) i sprzętowe kontrolery mediów (np. ekran dotykowy, kamerę) 15) zaprogramować alerty i powiadomienia różnymi metodami wizualnymi (jak paski, 	
--	--	--	--

		<p>dialogi) i niewizualnymi (światło, wibracja, dźwięk)</p> <p>16) zoptymalizować elementy graficzne aplikacji mobilnej z uwzględnieniem sprzętowych zasobów pamięciowych i ich konsumpcji</p> <p>17) zoptymalizować kod aplikacji mobilnej z uwzględnieniem parametrów sprzętowych urządzeń</p> <p>18) uruchomić aplikację mobilną na urządzeniu wirtualnym i fizycznym</p> <p>19) przygotować aplikację mobilną do publikacji w sklepie</p>	
--	--	---	--

Planowane zadania

Zadanie 1.

Stosując narzędzia szybkiego prototypowania, zaprojektuj interfejs użytkownika aplikacji mobilnej „Regularny trening”, obejmujący:

- stronę logowania,
- stronę historii ostatnich 10 aktywności sportowych,
- stronę dodawania aktywności sportowej.

Opisz związki i przejścia między stronami.

Zadanie 2.

Zaprogramuj menu główne aplikacji mobilnej, zawierające następujące pozycje:

- Mój profil,
- Historia aktywności,
- Wyloguj.

Zadanie 3.

Zaprogramuj prostą aplikację mobilną, która będzie odczytywać z modułu GPS bieżącą lokalizację użytkownika i wyświetlać na ekranie w sposób dynamiczny aktualną długość i szerokość geograficzną.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia programowania aplikacji mobilnych* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji mobilnych w wybranych językach programowania na wybrane platformy systemowe, pakiety oprogramowania obejmujące odpowiednie zintegrowane środowiska programistyczne oraz środowiska programistyczne typu RAD, narzędzia i frameworki wspomagające programowanie aplikacji mobilnych, środowisko uruchomieniowe dla aplikacji mobilnych, oprogramowanie serwerowe dla systemów baz danych z dostępem do API wymiany danych), oraz w urządzenia mobilne, projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności programowania aplikacji mobilnych w wybranych językach programowania i środowiskach programistycznych, projektowania efektywnego interfejsu graficznego aplikacji, tworzenia i optymalizacji graficznych elementów UI, wykorzystania API w celu integracji aplikacji z bazami danych oraz zrozumienia zagadnień dotyczących programowania funkcjonalności aplikacji związanych ze sprzętowymi właściwościami urządzeń mobilnych oraz preferencjami użytkownika.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.11. Pracownia zaawansowanych aplikacji webowych

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) wykorzystuje środowisko programistyczne dla zaawansowanych aplikacji webowych	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains) 2) stosuje środowiska IDE, RAD 3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych	1) dobrać środowisko programistyczne do określonych zadań i języka programowania 2) zastosować wybrane środowisko IDE, RAD 3) wykorzystać poznane narzędzia do tworzenia aplikacji webowych	
2) wykorzystuje frameworki do programowania aplikacji webowych	1) wykorzystuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js (do wyboru po jednym do technologii backend i technologii frontend) 2) wykorzystuje bibliotekę jQuery lub jej odpowiednik	1) zastosować platformę ASP.NET 4.x i ASP.NET Core do programowania aplikacji webowych 2) zastosować bibliotekę jQuery do tworzenia dynamicznej aplikacji internetowej 3) zastosować bibliotekę AngularJS do tworzenia aplikacji internetowej	

<p>3) programuje zaawansowane aplikacje webowe</p>	<ol style="list-style-type: none"> 1) stosuje języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript 2) stosuje mechanizm sesji użytkownika, ciasteczek 3) programuje dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny 4) programuje aplikacje korzystające z bazy danych 5) programuje wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego i innych 	<ol style="list-style-type: none"> 1) zastosować język C# do tworzenia zaawansowanych aplikacji internetowych 2) zastosować język PHP do tworzenia zaawansowanych aplikacji internetowych 3) tworzyć skrypty wykorzystujące pliki <i>cookies</i> 4) tworzyć skrypty wykorzystujące mechanizm sesji 5) programować dynamiczne formularze w językach PHP i C# 6) programować systemy logowania i kontroli dostępu do określonych elementów witryny w językach PHP i C# 7) definiować połączenie z bazą danych 8) definiować zapytania do bazy danych 9) zastosować bibliotekę PDO do obsługi bazy danych 10) tworzyć aplikacje korzystające z bazy danych 11) programować zaawansowane aplikacje internetowe 	
--	--	---	--

Planowane zadania

Zadanie 1.

Utwórz stronę internetową zawierającą menu. Wykorzystując bibliotekę jQuery, zaprojektuj dla utworzonego menu podpowiedzi w postaci dymków pojawiających się po ustawieniu myszy nad wybraną jego opcją.

Zadanie 2.

Utwórz stronę internetową zawierającą formularz logowania do aplikacji. Utwórz bazę danych przechowującą informacje na temat loginów i haseł użytkowników aplikacji. Po wprowadzeniu loginu i hasła sprawdzana jest poprawność logowania w systemie i wyświetlany jest komunikat: „Zostałeś zalogowany w naszej aplikacji”, gdy dane użytkownika występują w bazie, lub komunikat „Błędny login lub hasło”, gdy danych użytkownika brak w bazie.

Zadanie 3.

Napisz aplikację, która pozwoli klientowi sklepu internetowego utworzyć listę artykułów wybranych w sklepie. Artykuły dostępne w sklepie powinny zostać wyświetlone w formularzu w postaci listy. Po wybraniu przez klienta produktów lista tych produktów powinna zostać wyświetlona w drugim formularzu. Klient powinien mieć możliwość usunięcia artykułu z listy wybranych przez siebie artykułów. Do stworzenia aplikacji wykorzystaj skrypty języka PHP oraz mechanizm sesji.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia zaawansowanych aplikacji webowych* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji internetowych po stronie serwera i klienta w wybranych językach programowania, pakiety oprogramowania zawierające serwery baz danych), oraz w projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności projektowania i tworzenia aplikacji internetowych.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktazem oraz gotowego projektu.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących oraz poszukiwania ciekawych pomysłów na rozwiązania zaproponowanych ćwiczeń i zadań praktycznych.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji projektów przez uczniów. Podczas oceniania postępów w

naucze nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

3.12. Pracownia testowania i dokumentowania aplikacji

Efekty kształcenia z podstawy programowej	Kryteria weryfikacji z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi	Materiał w podręczniku
Uczeń:	Uczeń:		
1) przeprowadza walidację kodu programu	1) dobiera narzędzia i środowisko do testowania programów 2) wykonuje testy tworzonych programów 3) identyfikuje błędy w tworzonych programach 4) poprawia błędy w tworzonych programach, skryptach 5) optymalizuje kod źródłowy	1) dobrać narzędzia i środowisko do testowania danej aplikacji 2) dokonać statycznej inspekcji kodu, projektu i wymagań w celu wykrycia problemów 3) uruchomić program z danymi testowymi i ocenić poprawność jego zachowania 4) walidować wymagania нефункционалне poprzez warianty testów 5) usuwać defekty z własnych kodów źródłowych 6) optymalizować kod źródłowy z uwzględnieniem standardów i rekomendacji	
2) dokumentuje tworzoną aplikację	1) stosuje komentarze w kodzie źródłowym programu 2) pisze dokumentację kodu 3) pisze dokumenty pomocy programu	1) stosować zasady tworzenia poszczególnych rodzajów dokumentacji 2) umieszczać komentarze w kodzie źródłowym aplikacji odpowiednio do języka programowania	

	<ul style="list-style-type: none"> 4) pisze instrukcję użytkownika aplikacji 5) pisze instrukcję użytkownika programu 6) pisze dokumentację wdrożenia projektu 7) przygotowuje dokumentację z wykonanych testów 	<ul style="list-style-type: none"> 3) opracować w kodzie źródłowym komentarz w stylu DocBlocks 4) tworzyć dokumentację techniczną do danej aplikacji, korzystając z narzędzi automatycznego generowania 5) tworzyć ręcznie dokumentację (instrukcję) użytkownika zgodną z aktualną wersją aplikacji lub programu według ogólnie przyjętych zasad 6) opracować słownik pojęć niejednoznacznych na potrzeby dokumentacji użytkownika 7) opracować dokumentację wdrożenia projektu 8) przygotować dokumentację z przeprowadzonych testów aplikacji 	
<ul style="list-style-type: none"> 3) przeprowadza testy 	<ul style="list-style-type: none"> 1) charakteryzuje metodyki testowania 2) przygotowuje testy funkcjonalne i нефункционалне 3) przeprowadza testy funkcjonalne 4) przeprowadza testy w kolejnych fazach projektu informatycznego 5) przeprowadza testy нефункционалне: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa 6) korzysta z systemów raportowania błędów, np. BugZilla, JIRA 7) przeprowadza testy interfejsu 8) testuje prototyp projektu interfejsu 9) przygotowuje środowiska testowe 10) planuje scenariusze testowania aplikacji 	<ul style="list-style-type: none"> 1) sklasyfikować i scharakteryzować rodzaje testów aplikacji 2) stosować różne techniki testowania aplikacji 3) planować poszczególne fazy procesu testowania 4) opracować i przeprowadzić testy funkcjonalne i нефункционалне (użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa) 5) stosować systemy raportowania błędów 6) przygotować scenariusze testowania aplikacji 7) tworzyć specyfikację testową i dokumenty dla poszczególnych faz procesu testowania (listy kontrolne, scenariusze, raporty i in.) 8) oceniać zgodność aplikacji z wymaganiami klienta 	

	11) pisze scenariusze testowania aplikacji 12) raportuje błędy po przeprowadzonych testach 13) automatyzuje proces testowania 14) testuje aplikację 15) weryfikuje zgodność aplikacji z oczekiwaniami klienta	9) stosować narzędzia do zarządzania testami i frameworki do automatyzacji procesu testowania	
--	---	---	--

Planowane zadania

Zadanie 1.

Utwórz dokumentację wymagań klienta do aplikacji.

Zadanie 2.

Dodaj komentarze w formacie DocBlocks w kodzie źródłowym aplikacji i za pomocą wybranego narzędzia wygeneruj automatycznie dokumentację techniczną własnej aplikacji.

Zadanie 3.

Przygotuj scenariusze testowania aplikacji i przeprowadź zgodnie z nimi testy aplikacji. Po wykonaniu testów sporządź dokumentację testów aplikacji i opisz zgodność aplikacji z wymaganiami klienta.

Warunki osiągnięcia efektów kształcenia, w tym środki dydaktyczne, metody, formy organizacyjne

Zajęcia z przedmiotu *Pracownia testowania i dokumentowania aplikacji* w części praktycznej powinny odbywać się w *Pracowni programowania aplikacji*, wyposażonej w stanowisko komputerowe dla nauczyciela oraz stanowiska komputerowe dla uczniów (jedno stanowisko dla jednego ucznia) z dostępem do internetu, z zainstalowanym oprogramowaniem użytkowym (pakiet biurowy, programy graficzne, odtwarzacze multimedialne, oprogramowanie umożliwiające tworzenie aplikacji desktopowych, internetowych i mobilnych w wybranych językach programowania, pakiety oprogramowania obejmujące odpowiednie zintegrowane środowiska programistyczne, frameworki do automatyzacji testowania, narzędzia do zarządzania testami, oprogramowanie do automatyzacji tworzenia dokumentacji aplikacji, środowisko uruchomieniowe dla aplikacji webowych i mobilnych, serwer hostingowy do testowania projektów webowych), oraz w urządzenia mobilne, projektor, ekran, głośniki, urządzenie wielofunkcyjne. Stworzone warunki powinny umożliwić opanowanie przez ucznia umiejętności praktycznych z zakresu przedmiotu, tj. umiejętności testowania i dokumentowania różnego rodzaju aplikacji zarówno metodą ręczną, jak też zautomatyzowaną w celu sprawnego wykrywania defektów, poznania i wyćwiczenia metod optymalizowania kodu aplikacji zgodnie z przyjętymi normami i wytwarzania dokumentacji na potrzeby zespołu programistów, wdrożeniowców i użytkowników.

Środki dydaktyczne

Podczas realizacji zajęć praktycznych nauczyciel powinien wykorzystywać zróżnicowane środki dydaktyczne, w postaci: zestawów ćwiczeń i zadań, przykładowych rozwiązań, prezentacji multimedialnych lub materiałów elektronicznych. Stosowane środki dydaktyczne powinny pomóc w wykonaniu ćwiczeń i zadań kontrolnych prezentowanych podczas zajęć lekcyjnych z przedmiotu.

Zalecane metody dydaktyczne

Podczas zajęć praktycznych powinny dominować metody oparte na ćwiczeniach, umożliwiających zarówno nabywanie nowych umiejętności, jak i doskonalenie sprawności posiadanych już przez uczniów. Na zajęciach mogą być realizowane ćwiczenia w oparciu o instrukcję, ćwiczenia wspomagane metodą tekstu przewodniego oraz rekomendowaną metodą projektów. Proces kształcenia należy wzbogacać metodą pokazu z instruktążem oraz gotowego projektu i przypadków.

Nauczyciel powinien angażować i zachęcać uczniów do wykonywania ćwiczeń dodatkowych i doskonalących.

Formy organizacyjne

Zajęcia praktyczne powinny być prowadzone w formie pracy z podziałem klasy na grupy, gdzie na jednego ucznia przypada jedno stanowisko komputerowe. Praca w małych grupach ma zapewnić możliwość indywidualizacji pracy z uczniem oraz rozwijania rzeczywistego potencjału wszystkich uczniów. Zajęcia lekcyjne powinny być zorganizowane z uwzględnieniem wykorzystania aktywizujących metod i form pracy.

Propozycje kryteriów oceny i metod sprawdzania efektów kształcenia

Do oceny osiągnięć i postępów edukacyjnych uczniów w trakcie realizacji poszczególnych efektów kształcenia zaleca się stosowanie różnych metod weryfikacji opanowania i zwiększania umiejętności, np. odpowiedzi ustnych, sprawdzianów umiejętności praktycznych, testów wyboru (Moodle, platformy edukacyjne), ćwiczeń i zadań wykonywanych indywidualnie lub w małym zespole, obserwacji ucznia podczas zajęć dydaktycznych, prezentacji prac przez uczniów. Podczas oceniania postępów w nauce nauczyciel może stosować elementy oceniania kształtującego, które umożliwi uczniowi zdobycie informacji zwrotnej na temat tego, co już opanował, co powinien poprawić oraz jak powinien dalej pracować na zajęciach z danego przedmiotu.

Formy indywidualizacji pracy uczniów

Warunki, środki, metody i formy kształcenia powinny być dobierane z uwzględnieniem indywidualnych potrzeb rozwojowych i edukacyjnych każdego ucznia oraz powinny być dostosowane do jego możliwości psychofizycznych. Nauczyciel powinien dostosować stopień trudności materiału do możliwości i potrzeb uczniów, motywować uczniów do wykonywania ćwiczeń i zadań, przygotowywać zadania dydaktyczne w oparciu o wybrany podręcznik do nauki zawodu oraz zachęcać uczniów do korzystania z różnych źródeł wiedzy podczas nauki przedmiotu.

Wydawnictwo Helion