



**Fundusze  
Europejskie**  
Wiedza Edukacja Rozwój



**Rzeczpospolita  
Polska**

**Unia Europejska**  
Europejski Fundusz Społeczny



**PROGRAM NAUCZANIA**

**KWALIFIKACYJNEGO KURSU ZAWODOWEGO**

w zakresie kwalifikacji

**INF.04 Projektowanie, programowanie i testowanie aplikacji**

wyodrębnionej w zawodzie

**technik programista 351406**

Branża teleinformatyczna (INF)

**Autorzy:**

**mgr inż. Piotr Golonko**

**mgr Robert Fleischer**

**Recenzenci:**

**Recenzent 1** – Recenzja dydaktyczna (nauczyciel uczący w zawodzie, w którym wyodrębniono daną kwalifikację) **mgr inż. Marek Józwiak**

**Recenzent 2** – Recenzja merytoryczna (przedstawiciel pracodawców właściwy dla danego zawodu) **mgr Adam Mazgajczyk**

**Ekspert:**

**mgr inż. Grażyna Mrozińska-Hotłoś**

Program opracowany we współpracy podmiotów z otoczenia społeczno-gospodarczego wskazanego we wniosku o powierzenie grantu na opracowanie modelowego kwalifikacyjnego kursu zawodowego (KKZ):

DGA S.A. (Partner Wiodący) z Gminą Miastem Toruń (Partner) reprezentowaną przez Toruński Ośrodek Doradztwa Metodycznego i Doskonalenia Osoba prowadząca kursy z Torunia przy współpracy z Edukacja i Kształcenie Zawodowe. EKZ. podmiotami otoczenia społeczno-gospodarczego szkół lub placówek systemu oświaty prowadzących kształcenie zawodowe.

Program Operacyjny Wiedza Edukacja Rozwój

Oś priorytetowa II

Efektywne polityki publiczne dla rynku pracy, gospodarki i edukacji

Działanie 2.14. Rozwój narzędzi dla uczenia się przez całe życie

Konkurs nr POWR.02.14.00-IP.02-00-003/19

Opracowanie modelowych programów kwalifikacyjnych kursów zawodowych (kkz)

## Spis treści

<b>PROGRAM NAUCZANIA kwalifikacyjnego kursu zawodowego INF.04 Projektowanie, programowanie i testowanie aplikacji .....</b>	<b>6</b>
<b>1. Wprowadzenie.....</b>	<b>6</b>
<b>2. Plan zajęć kursu umiejętności zawodowych .....</b>	<b>11</b>
2.1. Pogrupowanie efektów kształcenia .....	11
2.2. Określenie liczby godzin na kształcenie zawodowe .....	53
2.3. Plan kursu .....	69
<b>3. Cele kształcenia kursu umiejętności zawodowych .....</b>	<b>70</b>
<b>4. Programy poszczególnych zajęć.....</b>	<b>71</b>
4.1. Program nauczania dla przedmiotu: Bezpieczeństwo pracy programisty .....	71
4.1.1 Cele ogólne przedmiotu to: .....	71
4.1.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	71
4.1.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia.....	71
4.1.4 Procedury osiągania celów kształcenia .....	72
4.1.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	73
4.2. Program nauczania dla przedmiotu: Podstawy informatyki dla programisty .....	74
4.2.1 Cele ogólne przedmiotu to: .....	74
4.2.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	74
4.2.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia.....	74
4.2.4 Procedury osiągania celów kształcenia .....	77
4.2.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	78
4.3. Program nauczania dla przedmiotu: Tworzenie aplikacji.....	78
4.3.1 Cele ogólne przedmiotu to: .....	78
4.3.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	78
4.3.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia.....	79
4.3.4 Procedury osiągania celów kształcenia .....	81
4.3.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	83
4.4. Program nauczania dla przedmiotu: Techniki programowania obiektowego.....	83
4.4.1 Cele ogólne przedmiotu to: .....	83
4.4.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	83
4.4.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia.....	84
4.4.4 Procedury osiągania celów kształcenia .....	87
4.4.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	88
4.5. Program nauczania dla przedmiotu: Tworzenie aplikacji desktopowych.....	89
4.5.1 Cele ogólne przedmiotu to: .....	89
4.5.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	89
4.5.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia.....	89

4.5.4	Procedury osiągania celów kształcenia .....	90
4.5.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	92
4.6.	Program nauczania dla przedmiotu: Tworzenie aplikacji mobilnych .....	92
4.6.1	Cele ogólne przedmiotu to: .....	92
4.6.2	Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	92
4.6.3	Materiał nauczania z uwzględnieniem opisu efektów kształcenia .....	92
4.6.4	Procedury osiągania celów kształcenia .....	95
4.6.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	96
4.7.	Program nauczania dla przedmiotu: Tworzenie aplikacji webowych .....	96
4.7.1	Cele ogólne przedmiotu to: .....	96
4.7.2	Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	97
4.7.3	Materiał nauczania z uwzględnieniem opisu efektów kształcenia .....	97
4.7.4	Procedury osiągania celów kształcenia .....	99
4.7.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	100
4.8.	Program nauczania dla przedmiotu: Dokumentacja i testy aplikacji .....	100
4.8.1	Cele ogólne przedmiotu to: .....	100
4.8.2	Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	101
4.8.3	Materiał nauczania z uwzględnieniem opisu efektów kształcenia .....	101
4.8.4	Procedury osiągania celów kształcenia .....	103
4.8.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	104
4.9.	Program nauczania dla przedmiotu: Język obcy zawodowy .....	104
4.9.1	Cele ogólne przedmiotu to: .....	104
4.9.2	Cele szczegółowe przedmiotu, Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi): .....	105
4.9.3	Materiał nauczania z uwzględnieniem opisu efektów kształcenia: .....	105
4.9.4	Procedury osiągania celów kształcenia .....	109
4.9.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	110
4.10.	Program nauczania dla przedmiotu: Praktyka zawodowa .....	110
4.10.1	Cele ogólne przedmiotu to: .....	111
4.10.2	Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi): .....	111
4.10.3	Materiał nauczania z uwzględnieniem opisu efektów kształcenia .....	113
4.10.4	Procedury osiągania celów kształcenia .....	121
4.10.5	Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika .....	122
<b>5.</b>	<b>Ewaluacja programu Kwalifikacyjnego kursu zawodowego .....</b>	<b>122</b>
<b>6.</b>	<b>Wykaz literatury oraz niezbędnych środków i materiałów dydaktycznych .....</b>	<b>131</b>
6.1.	Wykaz literatury .....	131
6.2.	Wykaz niezbędnych środków i materiałów dydaktycznych .....	133
<b>7.</b>	<b>Sposób i forma zaliczenia kursu .....</b>	<b>134</b>
<b>8.</b>	<b>Sprawdzenie kompletności i poprawności opracowanego programu zajęć .....</b>	<b>134</b>



**Fundusze  
Europejskie**  
Wiedza Edukacja Rozwój



**Rzeczpospolita  
Polska**

**Unia Europejska**  
Europejski Fundusz Społeczny



## **PROGRAM NAUCZANIA kwalifikacyjnego kursu zawodowego INF.04 Projektowanie, programowanie i testowanie aplikacji**

### **1. Wprowadzenie**

#### **Charakterystyka kursu umiejętności zawodowych**

Kurs może być prowadzony przez:

- publiczne i niepubliczne szkoły prowadzące kształcenie zawodowe, z wyjątkiem szkół artystycznych – w zakresie zawodów, w których kształcą, oraz w zakresie innych zawodów przypisanych do branż, do których należą zawody, w których kształci szkoła,
- publiczne i niepubliczne placówki kształcenia ustawicznego i centra kształcenia zawodowego,
- instytucje rynku pracy, o których mowa w art. 6 ustawy z dnia 20 kwietnia 2004 r. o promocji zatrudnienia i instytucjach rynku pracy, prowadzące działalność edukacyjno-szkoleniową,
- podmioty prowadzące działalność oświatową, o której mowa w art. 170 ust. 2, posiadające akredytację, o której mowa w art. 118. ustawy z dnia 14 grudnia 2016 r. Prawo oświatowe (Dz. U. z 2019 r. poz. 1148, z późn. zm.).

Kurs może być realizowany w formie:

- stacjonarnej – w czasie około 18 miesięcy (3 semestry), łącznie 810 godzin – zajęcia odbywają się 3 lub 4 dni w tygodniu po min. 8 godzin dziennie,
- zaocznej – w czasie około 18 miesięcy (65% z 810 godzin = 527 godzin) – zajęcia odbywają się co 2 tygodnie przez 2 dni po 10 godzin dziennie, a w uzasadnionych przypadkach – co tydzień przez 2 dni po 10 godzin dziennie.
- efekty kształcenia wskazane do realizacji w kształceniu teoretycznym mogą być (po spełnieniu wymagań określonych w aktualnych przepisach oświatowych) realizowane w formie kształcenia na odległość, przy czym zaliczenie tych zajęć nie może odbywać się w formie zdalnej. Przedmioty o kształceniu praktycznym ze względu na efekty uczenia wymagające fizycznej interakcji muszą być prowadzone w formie stacjonarnej.

Informacje dodatkowe:

- Kurs jest prowadzony na poziomie IV Polskiej Ramy Kwalifikacji
- Kurs nie jest związany z szczególnymi uwarunkowaniami związanymi z kształceniem w kwalifikacji.
- Ośrodek prowadzący kurs ma obowiązek zgłoszenia okręgowej komisji egzaminacyjnej informacji o jego rozpoczęciu (w przeciągu 14 dni od rozpoczęcia realizacji KKZ).

Kurs jest przeznaczony dla osób chcących:

- podnieść swoje kwalifikacje zawodowe,
- uzupełnić swoje umiejętności zawodowe,
- przygotować się do egzaminu zawodowego.

Wymagania wstępne:

- pozytywne przejście badań lekarskich (brak przeciwwskazań lekarskich do odbycia kursu)

Sposób i forma zaliczenia kursu:

- Kwalifikacyjny kurs zawodowy kończy się zaliczeniem w formie walidacji osiągnięć uczestnika kursu, polegającej na ocenie wykonywanych w trakcie nauki projektów i ćwiczeń oraz na podstawie uzyskanych w trakcie kursu ocen z poszczególnych przedmiotów.
- Do oceny osiągnięć edukacyjnych słuchaczy proponuje się stosowanie testów wielokrotnego wyboru, zadań z luką, ocenę aktywności słuchacza podczas wykonywania zadań w grupie, ocenę jakości wykonania zadań przez słuchacza. Proponuje się, aby osiągnięcia słuchaczy oceniać w zakresie zaplanowanych, uszczegółowionych celów kształcenia na podstawie:
  - obserwacji wykonanych ćwiczeń,
  - testu pisemnego.
- Umiejętności praktyczne proponuje się sprawdzać na podstawie obserwacji czynności wykonywanych przez słuchacza w trakcie realizacji ćwiczeń. Podczas obserwacji należy zwrócić uwagę na:
  - wyszukiwanie i przetwarzanie rzetelnych informacji pozyskanych z różnych źródeł,
  - poprawność merytoryczną wykonanych ćwiczeń praktycznych,
  - umiejętność pracy w zespole.
- Ważne kryteria oceny efektów kształcenia to: zaplanowanie wykonania zadania, dobór elementów oraz sporządzona dokumentacja techniczna. Możliwe są również inne sposoby i formy zaliczenia, takie jak: testy praktyczne, wykonanie projektów, próby pracy, aktywność uczącego się na zajęciach, prezentacje na forum grupy z przeprowadzonych prac.
- Osoba, która uzyskała zaliczenie, otrzymuje zaświadczenie o ukończeniu kursu dodatkowych umiejętności zawodowych.
- Kurs musi się zakończyć co najmniej 6 tygodni przed planowanym egzaminem z kwalifikacji zawodowej.

### **Założenia programowe**

Głównym celem kształcenia w technik programista jest przygotowanie szeroko wykwalifikowanej kadry specjalistów. Przygotowanych do:

- profesjonalnego i rzetelnego wykonywania czynności zawodowych,
- pracy w ciągle zmieniającej się rzeczywistości zawodowej,
- samodzielnego podnoszenia swoich kwalifikacji,
- podejmowania własnej działalności gospodarczej zgodnej z zawodem,
- pracy w zespole,
- szybkiej aktualizacji wiedzy z niezwykle dynamicznej dziedziny, jaką jest branża teleinformatyczna,
- kontynuowania edukacji w szkołach wyższych na kierunkach: elektronika, teleinformatyka, telekomunikacja, informatyka lub zbliżonych.

### **Struktura programu**

Program nauczania jest o strukturze przedmiotowej i spiralnej w układzie treści, z układem materiału nauczania zaczynającym się od zagadnień najprostszych po trudniejsze. Taki układ umożliwia powrót do treści zrealizowanych na początku edukacji, aby je powtórzyć i poszerzyć w kolejnych latach nauki. Utrwala to zarówno wiedzę jak i nabywane umiejętności celem przygotowania do realizacji zadań zawodowych. Dodatkowo taki układ i cykl nauczania w znaczącym stopniu niweluje braki edukacyjne, oraz pozwala na analizę materiału nauczania przez słuchaczy na różnych poziomach umiejętności.

## Cele kierunkowe

Absolwent kursu powinien być przygotowany do wykonywania następujących zadań zawodowych w zakresie:

- tworzenia stron i aplikacji internetowych,
- tworzenia i zarządzania bazami danych,
- tworzenia aplikacji desktopowych,
- tworzenia aplikacji mobilnych,
- testowania i dokumentowania aplikacji.

Rozkład treści nauczania uwzględnia wzajemną korelację pomiędzy przedmiotami, a kolejność zdobywania wiedzy i umiejętności pozwala na nabycie wiedzy teoretycznej, by w krótkim czasie wykorzystać ją praktycznie. Zajęcia są realizowane na przedmiotach kształcenia teoretycznego oraz praktycznego. Dzięki czemu po zakończeniu kursu absolwent potrafi:

- projektować i tworzyć aplikacje desktopowe, mobilne i webowe,
- korzystać z bibliotek i frameworków w czasie tworzenia i modyfikowania aplikacji,
- zarządzać projektami informatycznymi,
- wdrażać aplikacje,
- wdrażać poprawki aplikacji,
- wdrażać nowe wersje aplikacji,
- wdrażać nowe funkcjonalności,
- współpracować z zespołem programistów,
- współpracować z klientem,
- planować testy aplikacji,
- testować aplikacje,
- tworzyć dokumentację z przeprowadzonych testów,
- tworzyć dokumentację zgłoszonych błędów,
- tworzyć dokumentację zapotrzebowania na funkcjonalności,
- tworzyć dokumentację programu,
- tworzyć dokumentację wdrożeniową,
- tworzyć dokumentację (instrukcje) użytkowania aplikacji.

## Opis branży

Branża teleinformatyczna będąc jedną z wielu branż przyszłości składa się z szeregu zawodów ukierunkowanych na świadczenie usług związanych z budową i eksploatacją systemów teleinformatycznych. Branża ta obejmuje szeroki zakres umiejętności i zawodów. Od umiejętności monterskich związanych z mechanicznym i elektrycznym montażem kabli i urządzeń poprzez ich konfigurację i eksploatację aż po tworzenie programów, zarządzania infrastrukturą informatyczną czy też bazami



danych. W części tych zawodów bardzo ważne są kompetencje społeczne zwłaszcza w pracy z osobami niepełnosprawnymi. Branża cechuje się dużą dynamiką rozwoju i ciągłym zapotrzebowaniem na pracowników spowodowanym ciągłym i dynamicznym rozwojem usług i technologii przesyła i przetwarzania informacji.

### **Odniesienie do rynku pracy**

Zawód technik programista 351406, który powstał w oparciu o rekomendacje pracodawców zgłaszających zapotrzebowanie na wykwalifikowanych kandydatów do pracy. Jest nowoczesnym i wymagającym zawodem przyszłości, stawiającym ciągle nowe wyzwania i dającym możliwości samorealizacji i dużej satysfakcji z wykonywanej pracy. Branża teleinformatyczna dynamicznie się rozwija dając możliwość samorozwoju i zapewniając liczne miejsca pracy. W związku z tym istnieje zapotrzebowanie na osoby wykwalifikowane w tym zawodzie. Pracodawcy oczekują absolwenta wyposażonego w wiele kluczowych umiejętności i potrafiącego szybko reagować na zmieniającą się rzeczywistość oraz pogłębiać swoją wiedzę i umiejętności w zakresie nowych rozwiązań konstrukcyjnych i technologii.

Głównym celem kształcenia w zawodzie technik programista jest przygotowanie wykwalifikowanej kadry specjalistów do pracy w dynamicznie zmieniającym się sektorze gospodarki jakim jest branża teleinformatyczna.

Osoba posiadająca wykształcenie w tym zawodzie może pracować:

- przedsiębiorstwa zajmujące się tworzeniem programów desktopowych i aplikacji internetowych;
- przedsiębiorstwa zajmujące się tworzeniem aplikacji mobilnych;
- przedsiębiorstwa zajmujące się projektowaniem UI;
- przedsiębiorstwa zajmujące się modelowaniem, projektowaniem i drukiem 3D.

Poza swoją branżą technik programista 351406 może pracować na stanowiskach związanych z montażem elektrycznym, elektronicznym i mechanicznym sieci i urządzeń teletransmisyjnych związanych z szerokopasmową transmisją danych oraz wykonywaniem instalacji urządzeń teletransmisyjnych i ich uruchamianiem. Do podjęcia pracy w tym zawodzie niezbędna jest wysoka sprawność manualna i dobra koordynacja wzrokowo - ruchowa.

Pracodawcy powinni aktywnie uczestniczyć w cyklu kształcenia poprzez:

- konferencje branżowe,
- seminaria,
- lekcje problemowe,
- lekcje pokazowe,
- praktyki zawodowe,

organizowane wspólnie z szkołą.

### **Powiązanie z zawodami**

Branża teleinformatyczna pokrywa znaczny zakres usług, technik programista 351406 świadczy usługi z pogranicza telekomunikacji i informatyki, które są ukierunkowane na montaż, uruchamianie i utrzymanie sieci teleinformatycznych oraz instalowanie i administrowanie sieciowymi systemami komputerami. Umiejętności praktyczne są wzbogacone wiedzą ogólną z zakresu techniki analogowej i cyfrowej oraz wiedzą szczegółową z zakresu systemów komputerowych, sieciowych systemów operacyjnych, lokalnych sieci komputerowych, urządzeń sieciowych, systemów transmisji danych i systemów komutacyjnych.

Zawód technik programista wiąże się z szeregiem zawodów o zbliżonej tematyce, zakresie wiedzy i umiejętności. Umożliwia to zarówno pogłębianie wiedzy jak i łatwe przebranżowienie się na jeden z zawodów pokrewnych:

- technik teleinformatyk,
- technik informatyk,
- technik szerokopasmowej komunikacji elektronicznej,
- technik telekomunikacji,
- monter sieci i urządzeń telekomunikacyjnych,
- technik tyfloinformatyk.

Dodatkowo technik programista może też się sprawdzić w zawodach z branży związanej z elektroniką takich jak:

- elektronik,
- technik elektronik.

### **Informacje o pozostałych Kursach Umiejętności Zawodowych**

Istnieje możliwość uzupełnienia wiedzy i umiejętności zawodowych poprzez udział w dodatkowych Kursach Umiejętności Za dedykowanych dla zawodu technik programista 351406 (opisanych w odrębnych plikach):

- Bezpieczeństwo sieci komputerowych;
- Bezpieczeństwo systemów komputerowych;
- Budowa i konfiguracja sieci komputerowych;
- Eksploatacja baz danych;
- Grafika 3D i wydruk 3D;
- Programowanie mikrokontrolerów i prostych układów scalonych;
- Programowanie w języku Python;
- Serwis urządzeń techniki komputerowej;
- Tworzenie i testowanie aplikacji.

Program kwalifikacyjnego kursu zawodowego INF.04 Projektowanie, programowanie i testowanie aplikacji oparty jest o podstawę programową kształcenia branżowego w zawodzie **technik programista**, w której to wyodrębniono dla kwalifikacji INF.04 Projektowanie, programowanie i testowanie aplikacji następujące jednostki efektów kształcenia:

- INF.04.1. Bezpieczeństwo i higiena pracy,
- INF.04.2. Podstawy informatyki,
- INF.04.3. Projektowanie oprogramowania,
- INF.04.4. Programowanie obiektowe,
- INF.04.5. Programowanie aplikacji desktopowych,
- INF.04.6. Programowanie aplikacji mobilnych,
- INF.04.7. Programowanie aplikacji zaawansowanych webowych,
- INF.04.8. Testowanie i dokumentowanie aplikacji,

- INF.04.9. Język obcy zawodowy.

oraz efekty kształcenia realizowane na wszystkich obowiązkowych zajęć edukacyjnych z zakresu kształcenia zawodowego związane z nabywaniem kompetencji personalnych i społecznych i organizacji pracy małych zespołów, zgrupowane w jednostkach efektów kształcenia:

- INF.04.10. Kompetencje personalne i społeczne,
- INF.04.11. Organizacja pracy małych zespołów

Kwalifikacje zawodowe realizowane w ramach kursów umiejętności zawodowych (KUZ) w obrębie INF.04 Projektowanie, programowanie i testowanie aplikacji, mogą być osiągnięte z następujących jednostek efektów kształcenia:

- INF.04.1. Bezpieczeństwo i higiena pracy,
- INF.04.2. Podstawy informatyki,
- INF.04.3. Projektowanie oprogramowania,
- INF.04.4. Programowanie obiektowe,
- INF.04.5. Programowanie aplikacji desktopowych,
- INF.04.6. Programowanie aplikacji mobilnych,
- INF.04.7. Programowanie aplikacji zaawansowanych webowych,
- INF.04.8. Testowanie i dokumentowanie aplikacji,
- INF.04.9. Język obcy zawodowy.

## 2. Plan zajęć kursu umiejętności zawodowych

### 2.1. Pogrupowanie efektów kształcenia

**Tabela 1.** Przyporządkowanie efektów kształcenia wraz z kryteriami weryfikacji do poszczególnych przedmiotów.

Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
A	B	C	D	E	F	G	H	I	J	K	L
INF.04.1. Bezpieczeństwo i higiena pracy											



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywne	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
1) określa skutki oddziaływania czynników szkodliwych na organizm człowieka (ew)	3	1) określa zagrożenia występujące w środowisku pracy	x								
		2) określa skutki oddziaływania czynników fizycznych na organizm człowieka	x								
		3) określa skutki oddziaływania czynników psychofizycznych na organizm człowieka	x								
		4) opisuje skutki oddziaływania czynników niebezpiecznych i uciążliwych na organizm człowieka	x								
		5) wyjaśnia pojęcia choroba zawodowa i wypadek przy pracy	x								
2) stosuje środki ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych na stanowisku pracy (ek) 3) udziela pierwszej pomocy w stanach nagłego zagrożenia zdrowotnego (ek)	3	1) identyfikuje środki ochrony zbiorowej	x								
		2) wskazuje środki ochrony zabezpieczające przed hałasem w pracy biurowej	x								
		3) identyfikuje wymagania w zakresie oświetlenia, temperatury i	x								
		4) rozpoznaje środki ochrony zapobiegające porażeniem prądem w pracy biurowej	x								
		5) rozpoznaje środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa	x								
		6) dobiera środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowej	x								
	9	1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego	x								
		2) ocenia sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego	x								
		3) zabezpiecza siebie, poszkodowanego i miejsce wypadku	x								
		4) układa poszkodowanego w pozycji bezpiecznej	x								



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		5) powiadamia odpowiednie służby	x								
		6) prezentuje udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie	x								
		7) prezentuje udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar	x								
		8) wykonuje resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji	x								
INF.04.2. Podstawy informatyki											
1) charakteryzuje parametry sprzętu komputerowego (ew)	2	1) identyfikuje parametry urządzeń techniki komputerowej		x							
		2) porównuje parametry tego samego typu urządzeń techniki komputerowej (np. dwie karty graficzne, dwa dyski twarde)		x							
		3) przelicza jednostki pojemności pamięci masowych		x							
		4) dobiera urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi stanowiska		x							
2) definiuje elementy architektury systemów komputerowych (ew)	2	1) opisuje zasadę działania procesora (rozkazy)		x							
		2) wymienia zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego		x							
	4	1) identyfikuje system informatyczny		x							
		2) podaje przykłady systemów informacji przetwarzanych przez system informatyczny, w tym system PESEL, system postępowania rekrutacyjnego do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany		x							



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Język obcy zawodowy	Dokumentacja i testy aplikacji	Tworzenie aplikacji webowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji desktopowych	Techniki programowania obiektowego	Tworzenie aplikacji	Podstawy informatyki dla programistów	Bezpieczeństwo pracy programisty
3) charakteryzuje systemy informatyczne oraz rozróżnia systemy informatyczne pod względem funkcjonalności (ew)		3) opisuje miejsca przechowywania informacji: serwer lokalny, chmura, nośniki danych								x	
		4) dobiera systemy informatyczne pod względem ich funkcjonalności								x	
		5) opisuje działanie portali społecznościowych								x	
		6) określa zasady bezpiecznego korzystania z portali społecznościowych								x	
		7) podaje przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym e-commerce, e-sklep, e-faktura, systemy rezerwacyjne								x	
4) stosuje zalecenia dotyczące ułatwień dostępności serwisów internetowych dla osób niepełnosprawnych (ew)	2	1) wymienia dostępne udogodnienia dla osób z niepełnosprawnościami								x	
		2) wymienia wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0								x	
5) posługuje się terminologią dotyczącą sieci komputerowych (ek)	6	1) wymienia topologie sieci								x	
		2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej								x	
		3) opisuje sieć bezprzewodową oraz sieć przewodową								x	
		4) oblicza przepustowość sieci								x	
		5) definiuje pojęcia pobieranie i wysyłanie danych								x	
		6) opisuje zasady działania sieci synchronicznej i asynchronicznej								x	
		7) wykazuje różnice w działaniu sieci synchronicznej i asynchronicznej								x	



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci		x							
		9) używa komunikatorów tekstowych, audio-video oraz tablic interaktywnych		x							
		10) stosuje zasadę netykiety		x							
6) stosuje pozycyjne systemy liczbowe (ek)	4	1) przekształca liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym		x							
		2) zapisuje liczby w kodzie uzupełnieniowym do dwóch		x							
		3) wykonuje podstawowe działania logiczne i arytmetyczne na liczbach binarnych		x							
		4) wykorzystuje dostępne narzędzia informatyczne do wykonywania działań na liczbach zapisanych w różnych pozycyjnych systemach liczbowych (np. kalkulatory HEX, DEC, BIN) liczbowych		x							
7) stosuje zasady cyberbezpieczeństwa (ek)	8	1) rozróżnia rodzaje szkodliwego oprogramowania		x							
		2) rozróżnia rodzaje ataków hakerskich		x							
		3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi		x							
		4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni		x							
		5) opisuje zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom		x							





Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		6) przestrzega zasad bezpiecznego przechowywania danych		x							
		7) przestrzega zasad bezpieczeństwa swojego cyfrowego wizerunku i tożsamości		x							
		8) przestrzega zasad prywatności w cyfrowym świecie		x							
		9) wymienia i omawia podstawowe pojęcia związane z ochroną danych osobowych, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony		x							
		10) stosuje zasady dokonywania bezpiecznych transakcji w internecie		x							
8) rozpoznaje właściwe normy i procedury oceny zgodności podczas realizacji zadań zawodowych (ew)	2	1) wymienia cele normalizacji krajowej		x							
		2) wyjaśnia, czym jest norma i wymienia cechy normy		x							
		3) rozróżnia oznaczenie normy międzynarodowej, europejskiej i krajowej		x							
		4) korzysta ze źródeł informacji dotyczących norm i procedur oceny zgodności		x							
INF.04.3. Projektowanie oprogramowania											
1) posługuje się prostymi typami danych (ek)	4	1) rozróżnia typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe			x						
		2) rozpoznaje typ logiczny			x						
		3) rozróżnia typy znakowe i łańcuchowe			x						
		4) posługuje się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym			x						
	6	1) rozróżnia rodzaje złożonych typów danych			x						





Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywne	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
2) posługuje się złożonymi typami danych (ek)		2) posługuje się tablicami jednowymiarowymi i dwuwymiarowymi			x						
		3) posługuje się tablicami dynamicznymi, asocjacyjnymi			x						
		4) posługuje się typem rekordowym, np. struktura, unia			x						
		5) posługuje się typem plikowym			x						
		6) posługuje się typem wskaźnikowym			x						
		7) charakteryzuje cechy kolekcji, w tym znaczenie iteratora			x						
		8) posługuje się kolekcjami, np. listami, kolejkami, stosami, wektorami			x						
		9) projektuje zestawy danych dla problemu programistycznego			x						
3) stosuje metody rozwiązywania problemów za pomocą algorytmów (ek)	15	1) projektuje algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu			x						
		2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe			x						
		3) charakteryzuje algorytmy rekurencyjne			x						
		4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera			x						
		5) określa złożoność obliczeniową algorytmów			x						
		1) charakteryzuje typy sortowania i ich złożoność obliczeniową			x						
		2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj			x						
		3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów			x						
	7	1) określa funkcje narzędzi do zarządzania projektem			x						
		2) stosuje diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta			x						



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
5) dobiera narzędzia i metodologię do planowania i zarządzania projektem (ew)		3) korzysta z programów wspierających zarządzanie projektami, np. Jira, Trello			x						
		4) korzysta z systemu kontroli wersji, np. Git			x						
6) projektuje aplikację (ek)	15	1) analizuje wymagania klienta i tworzy zgodny z nimi projekt			x						
		2) tłumaczy wymagania klienta na specyfikację techniczną dla zespołu programistów			x						
		3) identyfikuje elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widżety			x						
		4) projektuje interfejs użytkownika i wygląd aplikacji			x						
		5) dostosowuje interfejs do różnych platform			x						
		6) projektuje aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym			x						
		7) projektuje aplikację opartą na architekturze klient-serwer			x						
		8) projektuje struktury danych dla aplikacji			x						
		9) projektuje funkcjonalność aplikacji			x						
		10) planuje system zabezpieczeń aplikacji			x						
7) planuje przedsięwzięcie programistyczne (ew)	20	1) określa cel projektu			x						
		2) określa fazy realizacji projektu			x						
		3) charakteryzuje cykl życia projektu informatycznego i jego poszczególne etapy			x						
		4) określa zasoby ludzkie oraz ramy czasowe wykonania projektu			x						
		5) planuje etapy tworzenia aplikacji			x						



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania objektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		6) korzysta z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)			x						
		7) dobiera optymalną metodologię zarządzania projektem			x						
		8) organizuje prace projektowe			x						
		9) stosuje harmonogram czynności w celu efektywnego osiągnięcia celów			x						
8) stosuje wzorce projektowe (ew)	4	1) dobiera wzorzec projektowy do zadania programistycznego			x						
		2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)			x						
9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania (ep)	4	1) rozróżnia autorskie prawa osobiste i majątkowe			x						
		2) określa czas trwania praw autorskich			x						
		3) określa konsekwencje naruszenia prawa autorskiego			x						
		4) charakteryzuje elementy własności intelektualnej (dobra niematerialne, własności przemysłowe)			x						
		5) rozróżnia typy licencji oprogramowania			x						
INF.04.4. Programowanie obiektowe											
1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych (ew)	30	1) rozróżnia kompilatory i interpretery				x					
		2) charakteryzuje zadania kompilatora, interpretera, debuggera				x					
		3) analizuje błędy w kodzie za pomocą debuggera				x					
		4) charakteryzuje etapy kompilacji i interpretacji kodu				x					
		5) charakteryzuje pojęcie biblioteki				x					



<b>Efekty kształcenia</b> Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	<b>Liczba godzin na efekt kształcenia</b>	<b>Kryteria weryfikacji</b>	<b>Bezpieczeństwo pracy programisty</b>	<b>Podstawy informatyki dla programistów</b>	<b>Tworzenie aplikacji</b>	<b>Techniki programowania obiektowego</b>	<b>Tworzenie aplikacji desktopowych</b>	<b>Tworzenie aplikacji mobilnych</b>	<b>Tworzenie aplikacji webowych</b>	<b>Dokumentacja i testy aplikacji</b>	<b>Język obcy zawodowy</b>
		6) kompiluje i uruchamia programy				x					
2) przestrzega zasad programowania (ew)	15	1) dzieli program na funkcje (metody)				x					
		2) stosuje rekurencję				x					
		3) implementuje algorytmy w programie				x					
3) korzysta z typów danych (ek)	16	1) stosuje proste i złożone typy danych				x					
		2) deklaruje własne typy danych				x					
		3) deklaruje zmienne różnych typów danych				x					
		4) wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne				x					
		5) stosuje typy złożone i operacje na nich				x					
4) stosuje wyrażenia, instrukcje i biblioteki (ek)	15	1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe				x					
		2) wykorzystuje priorytety operatorów do właściwego budowania wyrażeń				x					
		3) stosuje instrukcję warunkową i wyboru				x					
		4) stosuje instrukcje pętli				x					
		5) korzysta z wybranych bibliotek języka C++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami				x					
5) stosuje zasady programowania obiektowego (ek)	20	1) stosuje obiektowe podejście do rozwiązywania problemów				x					
		2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm				x					
		3) dzieli zagadnienie na klasy				x					



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		4) powołuje obiekty				x					
		5) planuje aplikację z zastosowaniem hermetyzacji, dziedziczenia i polimorfizmu									
6) definiuje klasy (ek)	20	1) definiuje pola klasy				x					
		2) określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu				x					
		3) definiuje metody klasy				x					
		4) definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy				x					
		5) definiuje instrukcje inicjujące konstruktora				x					
		6) określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu				x					
		7) implementuje funkcjonalność klasy				x					
		8) deklaruje obiekty i odwołuje się obiektem do składowych klasy				x					
		9) definiuje składniki statyczne klasy				x					
		10) rozróżnia klasy dziedziczone i zaprzyjaźnione				x					
		11) tworzy funkcje zaprzyjaźnione z klasą				x					
		12) stosuje składnik statyczny klasy i metody do ich obsługi				x					
7) definiuje klasy pochodne (ew)	14	1) buduje hierarchię dziedziczenia klas w programie				x					
		2) wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia				x					
		3) definiuje klasy bazowe i pochodne				x					
		4) stosuje metody wirtualne, definiuje klasy abstrakcyjne				x					



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiekowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy an aplikacji	Język obcy zawodowy
8) programuje szablony (wzorce) klas (ew)	10	1) definiuje szablony klas dla obsługi prostych typów liczbowych				x					
9) programuje obsługę wyjątków (ew)	10	1) stosuje szkielet obsługi wyjątków z instrukcjami try i catch				x					
		2) stosuje instrukcję throw				x					
		3) opracowuje listę możliwych błędów wykonania aplikacji				x					
		4) definiuje obsługę dla błędów wykonania aplikacji				x					
INF.04.5. Programowanie aplikacji desktopowych											
1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych (ew)	10	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)					x				
		2) stosuje środowiska IDE, RAD					x				
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych					x				
2) wykorzystuje frameworki do programowania aplikacji desktopowych (ek)	50	1) charakteryzuje pojęcie framework					x				
		2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt					x				
3) programuje desktopowe aplikacje okienkowe (ek)	60	1) rozróżnia elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki)					x				
		2) stosuje języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python					x				
		3) stosuje język do projektowania interfejsu użytkownika np. XAML					x				
		4) programuje okna aplikacji					x				
		5) programuje system menu aplikacji					x				
		6) programuje okna dialogowe aplikacji					x				



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywne	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Tworzenie aplikacji	Dokumentacja i testy aplikacji	Język obcy zawodowy
		7) programuje obsługę zdarzeń myszy i klawiatury					x					
INF.04.6. Programowanie aplikacji mobilnych												
1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych (ew)	20	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)						x				
		2) stosuje środowiska IDE, RAD						x				
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych						x				
2) programuje aplikacje mobilne (ek)	130	1) programuje aplikacje w jednym z systemów mobilnych: iOS lub Android						x				
		2) stosuje języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#						x				
		3) programuje przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych						x				
		4) identyfikuje elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)						x				
		5) programuje interfejs użytkownika za pomocą języka XAML						x				
		6) programuje aplikacje mobilne przy wykorzystaniu elementów UI						x				
		7) programuje proste aplikacje mobilne typu zegar, powiadomienie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)						x				
		8) pobiera i wysyła dane z lub do internetu dla aplikacji mobilnej						x				
		9) programuje aplikację mobilną korzystającą z bazy danych						x				





<b>Efekty kształcenia</b> Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	<b>Liczba godzin na efekt kształcenia</b>	<b>Kryteria weryfikacji</b>	<b>Bezpieczeństwo pracy programisty</b>	<b>Podstawy informatyki dla programistów</b>	<b>Tworzenie aplikacji</b>	<b>Techniki programowania obiektowego</b>	<b>Tworzenie aplikacji desktopowych</b>	<b>Tworzenie aplikacji mobilnych</b>	<b>Tworzenie aplikacji webowych</b>	<b>Dokumentacja i testy aplikacji</b>	<b>Język obcy zawodowy</b>
		10) tworzy aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)						x			
		11) uruchamia aplikacje mobilne						x			
		12) przygotowuje aplikacje do publikacji w sklepie						x			
<b>INF.04.7. Programowanie aplikacji zaawansowanych webowych</b>											
1) wykorzystuje środowisko programistyczne dla aplikacji zaawansowanych webowych (ew)	10	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)							x		
		2) stosuje środowiska IDE, RAD							x		
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych							x		
2) wykorzystuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js (ek)	80	1) stosuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js							x		
		2) wykorzystuje bibliotekę jQuery lub jej odpowiednik							x		
3) programuje zaawansowane aplikacje webowych (ek)	60	1) stosuje języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript							x		
		2) stosuje mechanizm sesji użytkownika, ciasteczek							x		
		3) programuje dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny							x		
		4) programuje aplikacje korzystające z bazy danych							x		
		5) programuje wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego							x		
	10	1) dobiera narzędzia i środowisko do testowania programów								x	





Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
1) przeprowadza walidację kodu programu (ek)		2) wykonuje testy tworzonych programów								x	
		3) identyfikuje błędy w tworzonych programach								x	
		4) poprawia błędy w tworzonych programach, skryptach								x	
		5) optymalizuje kod źródłowy								x	
2) dokumentuje tworzoną aplikację (ew)	10	1) stosuje komentarze w kodzie źródłowym programu								x	
		2) pisze dokumentację kodu								x	
		3) pisze dokumenty pomocy programu								x	
		4) pisze instrukcję użytkownika aplikacji								x	
		5) pisze instrukcję użytkownika programu								x	
		6) pisze dokumentację wdrożenia projektu								x	
		7) przygotowuje dokumentację z wykonanych testów								x	
3) przeprowadza testy (ek)	40	1) charakteryzuje metodyki testowania								x	
		2) przygotowuje testy funkcjonalne i нефункционалне								x	
		3) przeprowadza testy funkcjonalne								x	
		4) przeprowadza testy w kolejnych fazach projektu informatycznego								x	
		5) przeprowadza testy нефункционалне: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa								x	
		6) korzysta z systemów raportowania błędów, np. BugZilla, JIRA								x	
		7) przeprowadza testy interfejsu								x	
		8) testuje prototyp projektu interfejsu								x	
		9) przygotowuje środowiska testowe								x	
		10) planuje scenariusze testowania aplikacji								x	



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiekowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
		11) pisze scenariusze testowania aplikacji								x	
		12) raportuje błędy po przeprowadzonych testach								x	
		13) automatyzuje proces testowania								x	
		14) testuje aplikację								x	
		15) weryfikuje zgodność aplikacji z oczekiwaniami klienta								x	
INF.04.9. Język obcy zawodowy											
	10	1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:									x
		a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy									x
		b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych									x
		c) procesów i procedur związanych z realizacją zadań zawodowych									x
		d) formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych									x



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywne	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
1) posługuje się podstawowym zasobem środków językowych w języku obcym nowożytnym (ze szczególnym uwzględnieniem środków leksykalnych) umożliwiającym realizację czynności zawodowych w zakresie tematów związanych: a) ze stanowiskiem pracy i jego wyposażeniem b) z głównymi technologiami stosowanymi w danym zawodzie c) z dokumentacją związaną z danym zawodem d) z usługami świadczonymi w danym zawodzie (ek)		e) świadczonych usług, w tym obsługi klienta									x
	10	1) określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu									x



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Język obcy zawodowy	Dokumentacja i testy aplikacji	Tworzenie aplikacji webowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji desktopowych	Techniki programowania obiekowego	Tworzenie aplikacji	Podstawy informatyki dla programistów	Bezpieczeństwo pracy programisty
		2) znajduje w wypowiedzi lub tekście określone informacje									
		3) rozpoznaje związki między poszczególnymi częściami tekstu	x								



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: a) rozumie proste wypowiedzi ustne dotyczące czynności zawodowych (np. rozmowy, wiadomości, komunikaty, instrukcje lub filmy instruktażowe, prezentacje) artykułowane wyraźnie, w standardowej odmianie języka b) rozumie proste wypowiedzi pisemne dotyczące czynności zawodowych (np. napisy, broszury, instrukcje obsługi, przewodniki, dokumentację zawodową) (ew)		4) układa informacje w określonym porządku											X
3) samodzielnie tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne i pisemne w języku	10	1) opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi											X
		2) przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)											X



<b>Efekty kształcenia</b> Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	<b>Liczba godzin na efekt kształcenia</b>	<b>Kryteria weryfikacji</b>	<b>Bezpieczeństwo pracy programisty</b>	<b>Podstawy informatyki dla programistów</b>	<b>Tworzenie aplikacji</b>	<b>Techniki programowania obiektowego</b>	<b>Tworzenie aplikacji desktopowych</b>	<b>Tworzenie aplikacji mobilnych</b>	<b>Tworzenie aplikacji webowych</b>	<b>Dokumentacja i testy aplikacji</b>	<b>Język obcy zawodowy</b>
obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych: a) tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne dotyczące czynności zawodowych (np. polecenie, komunikat, instrukcję) 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 – według wzoru) (ew)		3) wyraża i uzasadnia swoje stanowisko									x
		4) stosuje zasady konstruowania tekstów o różnym charakterze									x
		5) stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji									x
	10	1) rozpoczyna, prowadzi i kończy rozmowę									x
		2) uzyskuje i przekazuje informacje i wyjaśnienia									x



Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiekowego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
											x
											x
											x
		3) wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób									x
		4) prowadzi proste negocjacje związane z czynnościami zawodowymi									x
		5) stosuje zwroty i formy grzecznościowe									x

Program nauczania kursu umiejętności zawodowych  
**INF.04 Projektowanie, programowanie i testowanie aplikacji**





Efekty kształcenia Stopniowane efektów kształcenia efekt kluczowy ek, efekt ważny ew, efekt pomocniczy ep	Liczba godzin na efekt kształcenia	Kryteria weryfikacji	Bezpieczeństwo pracy programisty	Podstawy informatyki dla programistów	Tworzenie aplikacji	Techniki programowania obiektywego	Tworzenie aplikacji desktopowych	Tworzenie aplikacji mobilnych	Tworzenie aplikacji webowych	Dokumentacja i testy aplikacji	Język obcy zawodowy
obcym nowożytnym w typowych sytuacjach związanym z wykonywaniem czynności zawodowych (ep)		2) przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym									x
		3) przekazuje w języku obcym nowożytnym informacje sformułowane w języku polskim lub w tym języku obcym nowożytnym									x
		4) przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację									x
6) wykorzystuje strategie służące doskonaleniu własnych umiejętności językowych oraz podnoszące świadomość językową: a) wykorzystuje techniki samodzielnej pracy nad językiem b) współdziała w grupie (ep)	10	1) korzysta ze słownika dwujęzycznego i jednojęzycznego									x
		2) współdziała z innymi osobami, realizując zadania językowe									x
		3) korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych									x
		4) identyfikuje słowa kluczowe i internacjonalizmy									x
		5) wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa									x
		6) upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznaną słowa innymi, wykorzystuje opis, środki niewerbalne									x
Razem liczba godzin w jednostce efektów kształcenia	810										

Osoba prowadząca kursy wszystkich obowiązkowych zajęć edukacyjnych powinna stwarzać warunki do nabywania kompetencji personalnych i społecznych oraz umiejętności w zakresie organizacji pracy małych zespołów w ramach godzin przeznaczonych na kształcenie zawodowe.

**Tabela 2.** Grupowanie efektów kształcenia w zajęcia i nadawanie nazwy tym zajęciom

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
A	B	C	D	E	F
INF.04.1. Bezpieczeństwo i higiena pracy (15h)	1) określa skutki oddziaływania czynników szkodliwych na organizm człowieka (ew)	1) określa zagrożenia występujące w środowisku pracy	Bezpieczeństwo pracy programisty (15h)	3	Początek kursu
		2) określa skutki oddziaływania czynników fizycznych na organizm człowieka			
		3) określa skutki oddziaływania czynników psychofizycznych na organizm człowieka			
		4) opisuje skutki oddziaływania czynników niebezpiecznych i uciążliwych na organizm człowieka			
		5) wyjaśnia pojęcia choroba zawodowa i wypadek przy pracy			
	2) stosuje środki ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych na stanowisku pracy (ek)	1) identyfikuje środki ochrony zbiorowe		3	
		2) wskazuje środki ochrony zabezpieczające przed hałasem w pracy biurowej			
		3) identyfikuje wymagania w zakresie oświetlenia, temperatury i			
		4) rozpoznaje środki ochrony zapobiegające porażeniem prądem w pracy biurowej			
		5) rozpoznaje środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa			
		6) dobiera środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowe			
	3) udziela pierwszej pomocy w stanach nagłego zagrożenia zdrowotnego (ek)	1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego		9	
		2) ocenia sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		3) zabezpiecza siebie, poszkodowanego i miejsce wypadku 4) układa poszkodowanego w pozycji bezpiecznej 5) powiadamia odpowiednie służby 6) prezentuje udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie 7) prezentuje udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar 8) wykonuje resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji			
INF.04.2. Podstawy informatyki (30h)	1) charakteryzuje parametry sprzętu komputerowego (ew)	1) identyfikuje parametry urządzeń techniki komputerowej	Podstawy informatyki dla programistów (30h)	2	Początek kursu
		2) porównuje parametry tego samego typu urządzeń techniki komputerowej (np. dwie karty graficzne, dwa dyski twarde)			
		3) przelicza jednostki pojemności pamięci masowych			
		4) dobiera urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi stanowiska			
	2) definiuje elementy architektury systemów komputerowych (ew)	1) opisuje zasadę działania procesora (rozkazy)		2	
		2) wymienia zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego			
		1) identyfikuje system informatyczny		4	

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	3) charakteryzuje systemy informatyczne oraz rozróżnia systemy informatyczne pod względem funkcjonalności (ew)	2) podaje przykłady systemów informacji przetwarzanych przez system informatyczny, w tym system PESEL, system postępowania rekrutacyjnego do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany			
		3) opisuje miejsca przechowywania informacji: serwer lokalny, chmura, nośniki danych			
		4) dobiera systemy informatyczne pod względem ich funkcjonalności			
		5) opisuje działanie portali społecznościowych			
		6) określa zasady bezpiecznego korzystania z portali społecznościowych			
		7) podaje przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym e-commerce, e-sklep, e-faktura, systemy rezerwacyjne			
	4) stosuje zalecenia dotyczące ułatwień dostępności serwisów internetowych dla osób niepełnosprawnych (ew)	1) wymienia dostępne udogodnienia dla osób z niepełnosprawnościami		2	
		2) wymienia wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0			
	5) posługuje się terminologią dotyczącą sieci komputerowych (ek)	1) wymienia topologie sieci		6	
		2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej			
		3) opisuje sieć bezprzewodową oraz sieć przewodową			
		4) oblicza przepustowość sieci			
		5) definiuje pojęcia pobieranie i wysyłanie danych			
		6) opisuje zasady działania sieci synchronicznej i asynchronicznej			



Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		7) wykazuje różnice w działaniu sieci synchronicznej i asynchronicznej			
		8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci			
		9) używa komunikatorów tekstowych, audio-video oraz tablic interaktywnych			
		10) stosuje zasadę netykiety			
	6) stosuje pozycyjne systemy liczbowe (ek)	1) przekształca liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym		4	
		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) liczbowych			
	7) stosuje zasady cyberbezpieczeństwa (ek)	1) rozróżnia rodzaje szkodliwego oprogramowania		8	
		2) rozróżnia rodzaje ataków hakerskich			
		3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi			
		4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		5) opisuje zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom			
		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, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony			
		10) stosuje zasady dokonywania bezpiecznych transakcji w internecie			
	8) rozpoznaje właściwe normy i procedury oceny zgodności podczas realizacji zadań zawodowych (ew)	1) wymienia cele normalizacji krajowej		2	
		2) wyjaśnia, czym jest norma i wymienia cechy normy			
		3) rozróżnia oznaczenie normy międzynarodowej, europejskiej i krajowej			
		4) korzysta ze źródeł informacji dotyczących norm i procedur oceny zgodności			
INF.04.3. Projektowanie oprogramowania (75h)	1) posługuje się prostymi typami danych (ek)	1) rozróżnia typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe	Tworzenie aplikacji (75h)	4	Początek kursu
		2) rozpoznaje typ logiczny			
		3) rozróżnia typy znakowe i łańcuchowe			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		4) posługuje się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym			
	2) posługuje się złożonymi typami danych (ek)	1) rozróżnia rodzaje złożonych typów danych		6	
		2) posługuje się tablicami jednowymiarowymi i dwuwymiarowymi			
		3) posługuje się tablicami dynamicznymi, asocjacyjnymi			
		4) posługuje się typem rekordowym, np. struktura, unia			
		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			
	3) stosuje metody rozwiązywania problemów za pomocą algorytmów (ek)	1) projektuje algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu		15	
		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			
		1) charakteryzuje typy sortowania i ich złożoność obliczeniową			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwycięża			
		3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów			
	5) dobiera narzędzia i metodologie do planowania i zarządzania projektem (ew)	1) określa funkcje narzędzi do zarządzania projektem		7	
		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			
	6) projektuje aplikację (ek)	1) analizuje wymagania klienta i tworzy zgodny z nimi projekt		15	
		2) tłumaczy wymagania klienta na specyfikację techniczną dla zespołu programistów			
		3) identyfikuje elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widgety			
		4) projektuje interfejs użytkownika i wygląd aplikacji			
		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			
	7) planuje przedsięwzięcie programistyczne (ew)	1) określa cel projektu		20	
		2) określa fazy realizacji projektu			



Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		3) charakteryzuje cykl życia projektu informatycznego i jego poszczególne etapy			
		4) określa zasoby ludzkie oraz ramy czasowe wykonania projektu			
		5) planuje etapy tworzenia aplikacji			
		6) korzysta z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)			
		7) dobiera optymalną metodologię zarządzania projektem			
		8) organizuje prace projektowe			
		9) stosuje harmonogram czynności w celu efektywnego osiągnięcia celów			
	8) stosuje wzorce projektowe (ew)	1) dobiera wzorzec projektowy do zadania programistycznego		4	
		2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)			
	9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania (ep)	1) rozróżnia autorskie prawa osobiste i majątkowe		4	
		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			
INF.04.4. Programowanie obiektowe (150h)	1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych (ew)	1) rozróżnia kompilatory i interpretery 2) charakteryzuje zadania kompilatora, interpretera, debuggera 3) analizuje błędy w kodzie za pomocą debuggera	Techniki programowania obiektowego (150h)	30	Semestr 1

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		4) charakteryzuje etapy kompilacji i interpretacji kodu			
		5) charakteryzuje pojęcie biblioteki			
		6) kompiluje i uruchamia programy			
	2) przestrzega zasad programowania (ew)	1) dzieli program na funkcje (metody)		15	
		2) stosuje rekurencję			
		3) implementuje algorytmy w programie			
	3) korzysta z typów danych (ek)	1) stosuje proste i złożone typy danych		16	
		2) deklaruje własne typy danych			
		3) deklaruje zmienne różnych typów danych			
		4) wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne			
		5) stosuje typy złożone i operacje na nich			
	4) stosuje wyrażenia, instrukcje i biblioteki (ek)	1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe		15	
		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++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami			
	5) stosuje zasady programowania obiektowego (ek)	1) stosuje obiektowe podejście do rozwiązywania problemów		20	
		2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm			
		3) dzieli zagadnienie na klasy			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		4) powołuje obiekty			
	6) definiuje klasy (ek)	1) definiuje pola klasy		20	
		2) określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu			
		3) definiuje metody klasy			
		4) definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy			
		5) definiuje instrukcje inicjujące konstruktora			
		6) określa zakres widoczności 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			
		12) stosuje składnik statyczny klasy i metody do ich obsługi			
	7) definiuje klasy pochodne (ew)	1) buduje hierarchię dziedziczenia klas w programie		14	
		2) wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia			
		3) definiuje klasy bazowe i pochodne			
		4) stosuje metody wirtualne, definiuje klasy abstrakcyjne			
	8) programuje szablony (wzorce) klas (ew)	1) definiuje szablony klas dla obsługi prostych typów liczbowych		10	
	9) programuje obsługę wyjątków (ew)	1) stosuje szkielet obsługi wyjątków z instrukcjami try i catch		10	
		2) stosuje instrukcję throw			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		3) opracowuje listę możliwych błędów wykonania aplikacji			
		4) definiuje obsługę dla błędów wykonania aplikacji			
INF.04.5. Programowanie aplikacji desktopowych (120h)	1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)	Tworzenie aplikacji desktopowych (120h)	10	Po zakończeniu podstaw informatyki dla programisty
		2) stosuje środowiska IDE, RAD			
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych			
	2) wykorzystuje frameworki do programowania aplikacji desktopowych (ek)	1) charakteryzuje pojęcie framework		50	
		2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt			
	3) programuje desktopowe aplikacje okienkowe (ek)	1) rozróżnia elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki)		60	
		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			
INF.04.6. Programowanie aplikacji mobilnych (150h)	1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)	Tworzenie aplikacji mobilnych (150h)	20	Po zakończeniu podstaw informatyki dla programisty
		2) stosuje środowiska IDE, RAD			
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	2) programuje aplikacje mobilne (ek)	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, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk) 5) programuje interfejs użytkownika za pomocą języka XAML 6) programuje aplikacje mobilne przy wykorzystaniu elementów UI 7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) 8) pobiera i wysyła dane z lub 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		130	
		1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)	Tworzenie aplikacji webowych (150h)	10	

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
INF.04.7. Programowanie aplikacji zaawansowanych webowych (150h)	1) wykorzystuje środowisko programistyczne dla aplikacji zaawansowanych webowych (ew)	2) stosuje środowiska IDE, RAD			Po zakończeniu podstaw informatyki dla programisty
		3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych		80	
	2) wykorzystuje frameworki do programowania aplikacji webowych (ek)	1) stosuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js			
		2) wykorzystuje bibliotekę jQuery lub jej odpowiednik		60	
	3) programuje zaawansowane aplikacje webowych (ek)	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			
INF.04.8. Testowanie i dokumentowanie aplikacji (60h)	1) przeprowadza walidację kodu programu (ek)	5) programuje wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego	Dokumentacja i testy aplikacji (60h)		Na koniec kursu
		1) dobiera narzędzia i środowisko do testowania programów		10	
		2) wykonuje testy tworzonych programów			
		3) identyfikuje błędy w tworzonych programach			
		4) poprawia błędy w tworzonych programach, skryptach			
		5) optymalizuje kod źródłowy			
	2) dokumentuje tworzoną aplikację (ew)	1) stosuje komentarze w kodzie źródłowym programu		10	
		2) pisze dokumentację kodu			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
		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			
	3) przeprowadza testy (ek)	1) charakteryzuje metodyki testowania		40	
		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			
		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			
INF.04.9. Język obcy zawodowy (60h)		1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:	Język obcy zawodowy (60h)	10	Na początku kursu

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	1) posługuje się podstawowym zasobem środków językowych w języku obcym nowożytnym (ze szczególnym uwzględnieniem środków leksykalnych) umożliwiającym realizację czynności zawodowych w zakresie tematów związanych: a) ze stanowiskiem pracy i jego wyposażeniem b) z głównymi technologiami stosowanymi w danym zawodzie c) z dokumentacją związaną z danym zawodem d) z usługami świadczonymi w danym zawodzie (ek)	a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy			
		b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych			
		c) procesów i procedur związanych z realizacją zadań zawodowych			
		d) formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych			
		e) świadczonych usług, w tym obsługi klienta			
		1) określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu		10	
		2) znajduje w wypowiedzi lub tekście określone informacje			
		3) rozpoznaje związki między poszczególnymi częściami tekstu			



Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	2) rozumie proste wypowiedzi ustne artykułowane wyrażnie, w standardowej odmianie języka obcego nowożytnego, a także proste wypowiedzi pisemne w języku obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych: a) rozumie proste wypowiedzi ustne dotyczące czynności zawodowych (np. rozmowy, wiadomości, komunikaty, instrukcje lub filmy instruktażowe, prezentacje) artykułowane wyrażnie, w standardowej odmianie języka b) rozumie proste wypowiedzi pisemne dotyczące czynności zawodowych (np. napisy, broszury, instrukcje obsługi, przewodniki, dokumentację zawodową) (ew)	4) układa informacje w określonym porządku			
		1) opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi		10	
		2) przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)			
		3) wyraża i uzasadnia swoje stanowisko			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania	
	3) samodzielnie tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne i pisemne w języku obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych: a) tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne dotyczące czynności zawodowych (np. polecenie, komunikat, instrukcję) 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 – według wzoru) (ew)	4) stosuje zasady konstruowania tekstów o różnym charakterze				
		5) stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji				
		1) rozpoczyna, prowadzi i kończy rozmowę				10
		2) uzyskuje i przekazuje informacje i wyjaśnienia				
		3) wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób				
		4) prowadzi proste negocjacje związane z czynnościami zawodowymi				
		5) stosuje zwroty i formy grzecznościowe				

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	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: a) reaguje ustnie (np. podczas rozmowy z innym pracownikiem, klientem, kontrahentem, w tym podczas rozmowy telefonicznej) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych b) reaguje w formie prostego tekstu pisanego (np. wiadomość, formularz, e-mail, dokument związany z wykonywanym zawodem) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)	6) dostosowuje styl wypowiedzi do sytuacji			

Nazwa jednostki efektów kształcenia	Efekt kształcenia wraz z kodowaniem (ek, ew, ep)	Kryteria weryfikacji	Grupowanie efektów kształcenia w przedmioty	Liczba godzin	Okres realizacji w cyklu nauczania
	5) zmienia formę przekazu ustnego lub pisemnego w języku obcym nowożytnym w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)	1) przekazuje w języku obcym nowożytnym informacje zawarte w materiałach wizualnych (np. wykresach, symbolach, piktogramach, schematach) oraz audiowizualnych (np. filmach instruktażowych)		10	
		2) przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym			
		3) przekazuje w języku obcym nowożytnym informacje sformułowane w języku polskim lub w tym języku obcym nowożytnym			
		4) przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację			
	6) wykorzystuje strategie służące doskonaleniu własnych umiejętności językowych oraz podnoszące świadomość językową: a) wykorzystuje techniki samodzielnej pracy nad językiem b) współdziała w grupie (ep)	1) korzysta ze słownika dwujęzycznego i jednojęzycznego		10	
		2) współdziała z innymi osobami, realizując zadania językowe			
		3) korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych			
		4) identyfikuje słowa kluczowe i internacjonalizmy			
		5) wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa			
		6) upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne			

- język obcy uczony jest na poziomie A1 lub A2, w zależności od kompetencji osób przystępujących do kursu,
- organizator kursu może podwyższyć poziom kształcenia Języka obcego zawodowego w zależności od kompetencji słuchaczy.

## 2.2. Określenie liczby godzin na kształcenie zawodowe

**Tabela 3.** Grupowanie efektów kształcenia i nadawanie nazw tym zajęciom.

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
Bezpieczeństwo pracy programisty (15h)	3		1) określa skutki oddziaływania czynników szkodliwych na organizm człowieka (ew)	1) określa zagrożenia występujące w środowisku pracy
				2) określa skutki oddziaływania czynników fizycznych na organizm człowieka
				3) określa skutki oddziaływania czynników psychofizycznych na organizm człowieka
				4) opisuje skutki oddziaływania czynników niebezpiecznych i uciążliwych na organizm człowieka
				5) wyjaśnia pojęcia choroba zawodowa i wypadek przy pracy
	3		2) stosuje środki ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych na stanowisku pracy (ek)	1) identyfikuje środki ochrony zbiorowe
				2) wskazuje środki ochrony zabezpieczające przed hałasem w pracy biurowej
				3) identyfikuje wymagania w zakresie oświetlenia, temperatury i
				4) rozpoznaje środki ochrony zapobiegające porażeniem prądem w pracy biurowej
				5) rozpoznaje środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa
				6) dobiera środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowe
	9		3) udziela pierwszej pomocy w stanach nagłego zagrożenia zdrowotnego (ek)	1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego
				2) ocenia sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego
				3) zabezpiecza siebie, poszkodowanego i miejsce wypadku
				4) układa poszkodowanego w pozycji bezpiecznej

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				5) powiadamia odpowiednie służby
				6) prezentuje udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie
				7) prezentuje udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar
				8) wykonuje resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji
Podstawy informatyki dla programisty (30h)	2		1) charakteryzuje parametry sprzętu komputerowego (ew)	1) identyfikuje parametry urządzeń techniki komputerowej
				2) porównuje parametry tego samego typu urządzeń techniki komputerowej (np. dwie karty graficzne, dwa dyski twarde)
				3) przelicza jednostki pojemności pamięci masowych
				4) dobiera urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi stanowiska
	2		2) definiuje elementy architektury systemów komputerowych (ew)	1) opisuje zasadę działania procesora (rozkazy)
				2) wymienia zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego
	6		3) charakteryzuje systemy informatyczne oraz rozróżnia systemy informatyczne pod względem funkcjonalności (ew)	1) identyfikuje system informatyczny
				2) podaje przykłady systemów informacji przetwarzanych przez system informatyczny, w tym system PESEL, system postępowania rekrutacyjnego do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany
				3) opisuje miejsca przechowywania informacji: serwer lokalny, chmura, nośniki danych

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				4) dobiera systemy informatyczne pod względem ich funkcjonalności
				5) opisuje działanie portali społecznościowych
				6) określa zasady bezpiecznego korzystania z portali społecznościowych
				7) podaje przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym e-commerce, e-sklep, e-faktura, systemy rezerwacyjne
	2		4) stosuje zalecenia dotyczące ułatwień dostępności serwisów internetowych dla osób niepełnosprawnych (ew)	1) wymienia dostępne udogodnienia dla osób z niepełnosprawnościami
	4		5) posługuje się terminologią dotyczącą sieci komputerowych (ek)	2) wymienia wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0
				1) wymienia topologie sieci
				2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej
				3) opisuje sieć bezprzewodową oraz sieć przewodową
				4) oblicza przepustowość sieci
				5) definiuje pojęcia pobieranie i wysyłanie danych
				6) opisuje zasady działania sieci synchronicznej i asynchronicznej
				7) wykazuje różnice w działaniu sieci synchronicznej i asynchronicznej
8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci				
4	6) stosuje pozycyjne systemy liczbowe (ek)	9) używa komunikatorów tekstowych, audio-video oraz tablic interaktywnych		
		10) stosuje zasadę netykiety		
		1) przekształca liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym		

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji	
A	B	C	D	E	
				2) zapisuje liczby w kodzie uzupełnieniowym do dwóch	
	4			7) stosuje zasady cyberbezpieczeństwa (ek)	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)liczbowych
			1) rozróżnia rodzaje szkodliwego oprogramowania		
			2) rozróżnia rodzaje ataków hakerskich		
			3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi		
			4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni		
			5) opisuje zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom		
			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, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony				
	10) stosuje zasady dokonywania bezpiecznych transakcji w internecie				
2			1) wymienia cele normalizacji krajowej		





Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
Tworzenie aplikacji (75h)			8) rozpoznaje właściwe normy i procedury oceny zgodności podczas realizacji zadań zawodowych (ew)	2) wyjaśnia, czym jest norma i wymienia cechy normy
				3) rozróżnia oznaczenie normy międzynarodowej, europejskiej i krajowej
				4) korzysta ze źródeł informacji dotyczących norm i procedur oceny zgodności
		4	1) posługuje się prostymi typami danych (ek)	1) rozróżnia typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe
				2) rozpoznaje typ logiczny
				3) rozróżnia typy znakowe i łańcuchowe
		6	2) posługuje się złożonymi typami danych (ek)	4) posługuje się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym
				1) rozróżnia rodzaje złożonych typów danych
				2) posługuje się tablicami jednowymiarowymi i dwuwymiarowymi
				3) posługuje się tablicami dynamicznymi, asocjacyjnymi
				4) posługuje się typem rekordowym, np. struktura, unia
		15	3) stosuje metody rozwiązywania problemów za pomocą algorytmów (ek)	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
				1) projektuje algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu
				2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				3) charakteryzuje algorytmy rekurencyjne
				4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera
				5) określa złożoność obliczeniową algorytmów
				1) charakteryzuje typy sortowania i ich złożoność obliczeniową
				2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwycięża
				3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów
		7	5) dobiera narzędzia i metodologie do planowania i zarządzania projektem (ew)	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
		15	6) projektuje aplikację (ek)	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, widgety
				4) projektuje interfejs użytkownika i wygląd aplikacji
				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



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				9) projektuje funkcjonalność aplikacji
				10) planuje system zabezpieczeń aplikacji
		2	7) planuje przedsięwzięcie programistyczne (ew)	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 wykonania projektu
				5) planuje etapy tworzenia aplikacji
				6) korzysta z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)
				7) dobiera optymalną metodologię zarządzania projektem
				8) organizuje prace projektowe
				9) stosuje harmonogram czynności w celu efektywnego osiągnięcia celów
			8) stosuje wzorce projektowe (ew)	1) dobiera wzorzec projektowy do zadania programistycznego
				2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)
		4	9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania (ep)	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
		3		1) rozróżnia kompilatory i interpretery



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
Techniki programowania obiektowego (150h)			1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych (ew)	2) charakteryzuje zadania kompilatora, interpretera, debuggera
				3) analizuje błędy w kodzie za pomocą debuggera
				4) charakteryzuje etapy kompilacji i interpretacji kodu
				5) charakteryzuje pojęcie biblioteki
				6) kompiluje i uruchamia programy
				1) dzieli program na funkcje (metody)
		15	2) przestrzega zasad programowania (ew)	2) stosuje rekurencję
				3) implementuje algorytmy w programie
		16	3) korzysta z typów danych (ek)	1) stosuje proste i złożone typy danych
				2) deklaruje własne typy danych
				3) deklaruje zmienne różnych typów danych
				4) wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne
				5) stosuje typy złożone i operacje na nich
		15	4) stosuje wyrażenia, instrukcje i biblioteki (ek)	1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe
				2) wykorzystuje priorytety operatorów do właściwego budowania wyrażeń
				3) stosuje instrukcję warunkową i wyboru
				4) stosuje instrukcje pętli
		2	5) stosuje zasady programowania obiektowego (ek)	5) korzysta z wybranych bibliotek języka C++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami
				1) stosuje obiektowe podejście do rozwiązywania problemów
				2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				3) dzieli zagadnienie na klasy
				4) powołuje obiekty
		2	6) definiuje klasy (ek)	1) definiuje pola klasy
				2) określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu
				3) definiuje metody klasy
				4) definiuje konstruktory, w tym konstruktor kopiujący, i destruktory klasy
				5) definiuje instrukcje inicjujące konstruktora
				6) określa zakres widoczności 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
				12) stosuje składnik statyczny klasy i metody do ich obsługi
		14	7) definiuje klasy pochodne (ew)	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
		1	8) programuje szablony (wzorce) klas (ew)	1) definiuje szablony klas dla obsługi prostych typów liczbowych
		1	9) programuje obsługę wyjątków (ew)	1) stosuje szkielet obsługi wyjątków z instrukcjami try i catch
				2) stosuje instrukcję throw
				3) opracowuje listę możliwych błędów wykonania aplikacji



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
Tworzenie aplikacji desktopowych (120h)		10	1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych (ew)	4) definiuje obsługę dla błędów wykonania aplikacji
				1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)
				2) stosuje środowiska IDE, RAD
		50	2) wykorzystuje frameworki do programowania aplikacji desktopowych (ek)	3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych
				1) charakteryzuje pojęcie framework
				2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt
Tworzenie aplikacji mobilnych (150h)		60	3) programuje desktopowe aplikacje okienkowe (ek)	3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych
				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
		20	1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych (ew)	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
		130	2) programuje aplikacje mobilne (ek)	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#

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				3) programuje przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych
				4) identyfikuje elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)
				5) programuje interfejs użytkownika za pomocą języka XAML
				6) programuje aplikacje mobilne przy wykorzystaniu elementów UI
				7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)
				8) pobiera i wysyła dane z lub 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
Tworzenie aplikacji webowych (150h)		10	1) wykorzystuje środowisko programistyczne dla aplikacji zaawansowanych webowych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)
		80	2) wykorzystuje frameworki do programowania aplikacji webowych (ek)	2) stosuje środowiska IDE, RAD
				3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych
				1) stosuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js
				2) wykorzystuje bibliotekę jQuery lub jej odpowiednik



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
		60	3) programuje zaawansowane aplikacje webowych (ek)	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
Dokumentacja i testy aplikacji (60h)		10	1) przeprowadza walidację kodu programu (ek)	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
		10	2) dokumentuje tworzoną aplikację (ew)	1) stosuje komentarze w kodzie źródłowym programu
				2) pisze dokumentację kodu
				3) pisze dokumenty pomocy programu
				4) pisze instrukcję użytkownika aplikacji
		40	3) przeprowadza testy (ek)	5) pisze instrukcję użytkownika programu
				6) pisze dokumentację wdrożenia projektu
				7) przygotowuje dokumentację z wykonanych testów
				1) charakteryzuje metodyki testowania
				2) przygotowuje testy funkcjonalne i niefunkcjonalne
				3) przeprowadza testy funkcjonalne
				4) przeprowadza testy w kolejnych fazach projektu informatycznego





Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
				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 15) weryfikuje zgodność aplikacji z oczekiwaniami klienta
Język obcy zawodowy (60h)	10		1) posługuje się podstawowym zasobem środków językowych w języku obcym nowożytnym (ze szczególnym uwzględnieniem środków leksykalnych) umożliwiającym realizację czynności zawodowych w zakresie tematów związanych: a) ze stanowiskiem pracy i jego wyposażeniem b) z głównymi technologiami stosowanymi w danym zawodzie c) z dokumentacją związaną z danym zawodem d) z usługami świadczonymi w danym zawodzie (ek)	1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie: a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych c) procesów i procedur związanych z realizacją zadań zawodowych d) formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych e) świadczonych usług, w tym obsługi klienta



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
	10		2) rozumie proste wypowiedzi ustne artykułowane wyrażnie, w standardowej odmianie języka obcego nowożytnego, a także proste wypowiedzi pisemne w języku obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych: a) rozumie proste wypowiedzi ustne dotyczące czynności zawodowych (np. rozmowy, wiadomości, komunikaty, instrukcje lub filmy instruktażowe, prezentacje) artykułowane wyrażnie, w standardowej odmianie języka b) rozumie proste wypowiedzi pisemne dotyczące czynności zawodowych (np. napisy, broszury, instrukcje obsługi, przewodniki, dokumentację zawodową) (ew)	1) określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu
				2) znajduje w wypowiedzi lub tekście określone informacje
				3) rozpoznaje związki między poszczególnymi częściami tekstu
				4) układa informacje w określonym porządku
	10			1) opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi
				2) przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)
				3) wyraża i uzasadnia swoje stanowisko
				4) stosuje zasady konstruowania tekstów o różnym charakterze



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
	10		3) samodzielnie tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne i pisemne w języku obcym nowożytnym w zakresie umożliwiającym realizację zadań zawodowych: a) tworzy krótkie, proste, spójne i logiczne wypowiedzi ustne dotyczące czynności zawodowych (np. polecenie, komunikat, instrukcję) 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 – według wzoru) (ew)	5) stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji
			1) rozpoczyna, prowadzi i kończy rozmowę	
	2) uzyskuje i przekazuje informacje i wyjaśnienia			
	3) wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób			
	4) prowadzi proste negocjacje związane z czynnościami zawodowymi			
	5) stosuje zwroty i formy grzecznościowe			



Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
			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: a) reaguje ustnie (np. podczas rozmowy z innym pracownikiem, klientem, kontrahentem, w tym podczas rozmowy telefonicznej) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych b) reaguje w formie prostego tekstu pisanego (np. wiadomość, formularz, e-mail, dokument związany z wykonywanym zawodem) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)	6) dostosowuje styl wypowiedzi do sytuacji
	10		5) zmienia formę przekazu ustnego lub pisemnego w języku obcym nowożytnym w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)	1) przekazuje w języku obcym nowożytnym informacje zawarte w materiałach wizualnych (np. wykresach, symbolach, piktogramach, schematach) oraz audiowizualnych (np. filmach instruktażowych) 2) przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym 3) przekazuje w języku obcym nowożytnym informacje sformułowane w języku polskim lub w tym języku obcym nowożytnym 4) przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację

Przedmiot	Liczba godzin zajęć teoretycznych	Liczba godzin zajęć praktycznych	Efekty kształcenia	Kryteria weryfikacji
A	B	C	D	E
	10		6) wykorzystuje strategie służące doskonaleniu własnych umiejętności językowych oraz podnoszące świadomość językową: a) wykorzystuje techniki samodzielnej pracy nad językiem b) współdziała w grupie (ep)	1) korzysta ze słownika dwujęzycznego i jednojęzycznego 2) współdziała z innymi osobami, realizując zadania językowe 3) korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych 4) identyfikuje słowa klucze i internacjonalizmy 5) wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa 6) upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznanne słowa innymi, wykorzystuje opis, środki niewerbalne
Razem godzin:	109	701		
SUMA:	810			

Efekty w ramach jednostek:

- INF.04.10. Kompetencje personalne i społeczne
- INF.04.11. Organizacja pracy małych zespołów

Są realizowane w ramach wszystkich zajęć praktycznych, a osoba prowadząca kursy wszystkich obowiązkowych zajęć edukacyjnych z zakresu kształcenia zawodowego powinni stwarzać uczniom warunki do nabywania kompetencji personalnych i społecznych oraz umiejętności w zakresie organizacji pracy małych zespołów.

### 2.3. Plan kursu

Kurs może być realizowany w formie:

- stacjonarnej – w czasie około 18 miesięcy (3 semestry), łącznie 810 godzin co stanowi 100% liczby godzin wynikającej z podstawy programowej – zajęcia odbywają się 3 lub 4 dni w tygodniu po min. 8 godzin dziennie,
- zaocznej – w czasie około 18 miesięcy (65% z 570 godzin = 371 godzin) – zajęcia odbywają się co 2 tygodnie przez 2 dni po 10 godzin dziennie, a w uzasadnionych przypadkach – co tydzień przez 2 dni po 10 godzin dziennie.

- efekty kształcenia wskazane do realizacji w kształceniu teoretycznym mogą być (po spełnieniu wymagań określonych w aktualnych przepisach oświatowych) realizowane w formie kształcenia na odległość, przy czym zaliczenie tych zajęć nie może odbywać się w formie zdalnej. Ze względu na specyfikę zawodu programisty część zajęć może być prowadzona zdalnie o ile uczestnicy kursu dysponują odpowiednim zapleczem technicznym i programowym pozwalającym na realizację efektów kursu.

**Tabela 4.** Określenie liczby godzin poszczególnych zajęć z podziałem na zajęcia teoretyczne i praktyczne

Lp.	Powiązanie z podstawą programową	Przedmioty	Liczba godzin	Uwagi
Kształcenie teoretyczne				Zajęcia teoretyczne powinny odbywać się przez zajęciami praktycznymi. Realizacja przedmiotów praktycznych powinna zapewniać korelacje pomiędzy efektami.
1	INF.04.1. Bezpieczeństwo i higiena pracy	Bezpieczeństwo pracy programisty	15	
2	INF.04.2. Podstawy informatyki	Podstawy informatyki dla programisty	30	
3	INF.04.9. Język obcy zawodowy	Język obcy zawodowy	60	
Kształcenie praktyczne				
4	INF.04.3. Projektowanie oprogramowania	Tworzenie aplikacji	75	
5	INF.04.4. Programowanie obiektowe	Techniki programowania obiektowego	150	
6	INF.04.5. Programowanie aplikacji desktopowych	Tworzenie aplikacji desktopowych	120	
7	INF.04.6. Programowanie aplikacji mobilnych	Tworzenie aplikacji mobilnych	150	
8	INF.04.7. Programowanie aplikacji zaawansowanych webowych	Tworzenie aplikacji webowych	150	
9	INF.04.8. Testowanie i dokumentowanie aplikacji	Dokumentacja i testy aplikacji	60	
			Łączna liczba godzin:	810
<b>Planowany termin praktyki zawodowej</b> – w przypadku kwalifikacyjnego kursu zawodowego (jeżeli w podstawie programowej, w którym wyodrębniono daną kwalifikację przewidziano praktykę zawodową): <ul style="list-style-type: none"> <li>— Praktyka zawodowa odbywa się w II połowie kursu, w wymiarze 140 godzin.</li> <li>— Praktyka zawodowa musi odbywać w trakcie trwania kursu</li> </ul>				Praktyka zawodowa powinna odbywać się po zakończeniu przedmiotów teoretycznych
<b>Planowany termin egzaminu:</b> <ul style="list-style-type: none"> <li>— Egzamin potwierdzający kwalifikację INF.04 Projektowanie, programowanie i testowanie aplikacji odbywa się po zakończeniu KKZ.</li> <li>— Termin egzaminu zgodny z harmonogramem ogłaszany przez Dyrektora Centralnej Komisji Egzaminacyjnej.</li> </ul>				KKZ musi się zakończyć co najmniej 6 tygodni przed planowanym terminem egzaminu potwierdzającego kwalifikację.

### 3. Cele kształcenia kursu umiejętności zawodowych

Absolwent kursu powinien być przygotowany do wykonywania następujących zadań zawodowych w zakresie:

- wdrażania projektów programistycznych;
- testowania aplikacji;
- przygotowania dokumentacji z wykonanych testów aplikacji.

## 4. Programy poszczególnych zajęć

### 4.1. Program nauczania dla przedmiotu: Bezpieczeństwo pracy programisty

#### 4.1.1 Cele ogólne przedmiotu to:

- Nabywanie umiejętności stosowania zasad bezpiecznej pracy.
- Nabywanie umiejętności określania zagrożeń dla zdrowia i życia człowieka podczas wykonywania zadań zawodowych.
- Kształtowanie umiejętności identyfikowania czynników niebezpiecznych, szkodliwych i uciążliwych podczas wykonywania zadań zawodowych.
- Poznanie i stosowanie środków ochrony indywidualnej i zbiorowej.
- Doskonalenie umiejętności udzielania pierwszej pomocy poszkodowanym podczas wykonywania zadań zawodowych.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

#### 4.1.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):

- rozróżniać pojęcia związane z bezpieczeństwem i higieną pracy,
- rozróżniać i charakteryzować czynniki szkodliwe, uciążliwe i niebezpieczne w środowisku pracy,
- rozróżniać i charakteryzować środki ochrony indywidualnej i zbiorowej podczas wykonywania prac zawodowych,
- określać i stosować zasady udzielania pierwszej pomocy,
- udzielać pierwszej pomocy poszkodowanym w wypadkach przy pracy oraz w stanach zagrożenia zdrowia i życia.

#### 4.1.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Ochrona pracownika	Ochrona pracownika	6	<ul style="list-style-type: none"> <li>— określać zagrożenia występujące w środowisku pracy</li> <li>— opisywać skutki oddziaływania czynników niebezpiecznych i uciążliwych na organizm człowieka</li> <li>— identyfikować środki ochrony zbiorowe</li> <li>— wskazywać środki ochrony zabezpieczające przed hałasem w pracy biurowej</li> <li>— identyfikować wymagania w zakresie oświetlenia, temperatury i</li> <li>— rozpoznawać środki ochrony zapobiegające porażeniem prądem w pracy biurowej</li> <li>— rozpoznawać środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— określać skutki oddziaływania czynników fizycznych na organizm człowieka</li> <li>— określać skutki oddziaływania czynników psychofizycznych na organizm człowieka</li> <li>— wyjaśniać pojęcia choroba zawodowa i wypadek przy pracy</li> <li>— dobrać środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowe</li> </ul>
Niesienie pomocy	Niesienie pomocy	9	<ul style="list-style-type: none"> <li>— opisywać podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego</li> <li>— powiadamiać odpowiednie służby</li> <li>— oceniać sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego</li> <li>— zabezpieczać siebie, poszkodowanego i miejsce wypadku</li> <li>— układać poszkodowanego w pozycji bezpiecznej</li> <li>— prezentować udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie</li> <li>— prezentować udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar</li> <li>— wykonywać resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji</li> </ul>

#### 4.1.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.



## **Obudowa dydaktyczna**

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21”,

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- zagrożeń w miejscu pracy,
- przeciwdziałania zagrożeniom,
- niesienia pomocy,
- dodatkowo w ilości 1 zestaw na 2 kursantów:
- zestawy pierwszej pomocy do ćwiczeń,
- zestaw do ćwiczenia RKO.

Wyposażenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne

## **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

### **4.1.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## **4.2. Program nauczania dla przedmiotu: Podstawy informatyki dla programisty**

### **4.2.1 Cele ogólne przedmiotu to:**

- Posługiwanie się i systemami pozycyjnymi.
- Poznanie podstaw budowy i działania sprzętu komputerowego.
- Poznanie podstaw budowy i działania sieci komputerowych.
- Nabycie umiejętności do rozróżniania systemów informatycznych pod względem funkcjonalności.
- Poznanie podstaw bezpieczeństwa w internecie.
- Korzystanie z norm krajowych, europejskich i międzynarodowych.
- Nabywanie umiejętności pracy w grupach i zespołach.
- Rozwijanie kompetencji personalnych i społecznych.

### **4.2.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):**

- posługiwać się systemami liczbowymi,
- przeliczać wartości pomiędzy systemami,
- identyfikować parametry sprzętu komputerowego,
- dobierać urządzenia komputerowe z uwzględnieniem ich parametrów,
- obliczać przepustowości sieci,
- opisywać protokoły sieciowe,
- stosować zasady netykiety,
- znać i stosować normy stosowane w pracy technika programisty,
- identyfikować zagrożenia dla systemów informatycznych,
- przeciwdziałać zagrożeniom systemów informatycznych.

### **4.2.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia**

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>
Sprzęt i sieci	Systemy liczbowe	4	<ul style="list-style-type: none"> <li>— obliczać przepustowość sieci</li> <li>— zapisuje liczby w kodzie uzupełnieniowym do dwóch</li> <li>— wykorzystywać dostępne narzędzia informatyczne do wykonywania działań na liczbach zapisanych w różnych pozycyjnych systemach liczbowych (np. kalkulatory HEX, DEC, BIN)</li> </ul>

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>
			<ul style="list-style-type: none"> <li>— przeliczać jednostki pojemności pamięci masowych</li> <li>— przekształcać liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym</li> <li>— wykonywać podstawowe działania logiczne i arytmetyczne na liczbach binarnych</li> </ul>
Sprzęt i sieci	Sprzęt komputerowy	4	<ul style="list-style-type: none"> <li>— identyfikować parametry urządzeń techniki komputerowej</li> <li>— porównywać parametry tego samego typu urządzeń techniki komputerowej (np. dwie karty graficzne, dwa dyski twarde)</li> <li>— wymieniać zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego</li> <li>— przeliczać jednostki pojemności pamięci masowych</li> <li>— dobierać urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi stanowiska</li> <li>— opisywać zasadę działania procesora (rozkazy)</li> </ul>
Sprzęt i sieci	Sieci komputerowe	6	<ul style="list-style-type: none"> <li>— Wymieniać topologie sieci</li> <li>— opisywać sieć bezprzewodową oraz sieć przewodową</li> <li>— obliczać przepustowość sieci</li> <li>— definiować pojęcia pobieranie i wysyłanie danych</li> <li>— wykazywać różnice w działaniu sieci synchronicznej i asynchronicznej</li> <li>— identyfikować cechy modelu TCP/IP i protokołów komunikacji sieciowej</li> <li>— opisywać zasady działania sieci synchronicznej i asynchronicznej</li> <li>— wymieniać i stosuje zasady bezpieczeństwa przy korzystaniu z sieci</li> <li>— używać komunikatorów tekstowych, audio-video oraz tablic interaktywnych</li> <li>— stosować zasadę netykiety</li> </ul>
Systemy informatyczne	Systemy informatyczne	6	<ul style="list-style-type: none"> <li>— identyfikować system informatyczny</li> <li>— podawać przykłady systemów informacji przetwarzanych przez system informatyczny, w tym system PESEL, system postępowania rekrutacyjnego do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany</li> <li>— opisywać miejsca przechowywania informacji: serwer lokalny, chmura, nośniki danych</li> <li>— wymieniać dostępne udogodnienia dla osób z niepełnosprawnościami</li> <li>— dobierać systemy informatyczne pod względem ich funkcjonalności</li> <li>— opisywać działanie portali społecznościowych</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— określać zasady bezpiecznego korzystania z portali społecznościowych</li> <li>— podawać przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym e-commerce, e-sklep, e-faktura, systemy rezerwacyjne</li> <li>— wymieniać wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0</li> </ul>
Systemy informatyczne	Bezpieczeństwo w informatyce	8	<ul style="list-style-type: none"> <li>— rozróżniać rodzaje szkodliwego oprogramowania</li> <li>— rozróżniać rodzaje ataków hakerskich</li> <li>— wymieniać zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni</li> <li>— przestrzegać zasad bezpiecznego przechowywania danych</li> <li>— przestrzegać zasad prywatności w cyfrowym świecie</li> <li>— wymieniać środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi</li> <li>— opisywać zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom</li> <li>— przestrzegać zasad bezpieczeństwa swojego cyfrowego wizerunku i tożsamości</li> <li>— wymieniać podstawowe pojęcia związane z ochroną danych osobowych, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony</li> <li>— omawiać podstawowe pojęcia związane z ochroną danych osobowych, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony</li> <li>— stosować zasady dokonywania bezpiecznych transakcji w Internecie</li> </ul>
Systemy informatyczne	Normy w informatyce	2	<ul style="list-style-type: none"> <li>— wymieniać cele normalizacji krajowej</li> <li>— wyjaśniać, czym jest norma i wymienia cechy normy</li> <li>— rozróżniać oznaczenie normy międzynarodowej, europejskiej i krajowej</li> <li>— korzystać ze źródeł informacji dotyczących norm i procedur oceny zgodności</li> </ul>

#### **4.2.4 Procedury osiągania celów kształcenia**

##### **Propozycje metod nauczania,**

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### **Obudowa dydaktyczna**

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- budowy komputera i jego parametrów,
- protokołów sieciowych,
- sieci komputerowej,
- bezpieczeństwa IT,
- norm w informatyce.

Wypożyczenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

## **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

### **4.2.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## **4.3. Program nauczania dla przedmiotu: Tworzenie aplikacji**

### **4.3.1 Cele ogólne przedmiotu to:**

- Stosowanie różnych typów danych.
- Stosowanie różnych algorytmów.
- Tworzenie projektu aplikacji na podstawie wymagań klienta.
- Planowanie prac projektowych.
- Licencjonowanie.
- Nabywanie umiejętności pracy w grupach i zespołach.
- Nabycie kompetencji personalnych i społecznych.

### **4.3.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):**

- zastosować różne typy danych,
- zastosować algorytmy,
- zastosować narzędzia wspomagające projektowanie aplikacji,
- współpracować z klientem,
- opracowywać funkcjonalność aplikacji,
- projektować aplikację,
- tworzyć harmonogramy pracy,
- określać zasady licencjonowania,
- zaplanować system zabezpieczeń licencyjnych w aplikacjach.

#### 4.3.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Wstęp do programowania	Wstęp do programowania, typy danych	10	<ul style="list-style-type: none"> <li>— rozróżniać typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe</li> <li>— rozpoznawać typ logiczny</li> <li>— rozróżniać typy znakowe i łańcuchowe</li> <li>— posługiwać się tablicami jednowymiarowymi i dwuwymiarowymi</li> <li>— posługiwać się typem plikowym</li> <li>— posługiwać się typem wskaźnikowym</li> <li>— posługiwać się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym</li> <li>— rozróżniać rodzaje złożonych typów danych</li> <li>— posługiwać się tablicami dynamicznymi, asocjacyjnymi</li> <li>— posługiwać się typem rekordowym, np. struktura, unia</li> <li>— charakteryzować cechy kolekcji, w tym znaczenie iteratora</li> <li>— posługiwać się kolekcjami, np. listami, kolejkami, stosami, wektorami</li> <li>— projektować zestawy danych dla problemu programistycznego</li> </ul>
Wstęp do programowania	Rozwiązywanie problemów przy użyciu algorytmów	15	<ul style="list-style-type: none"> <li>— charakteryzować algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe</li> <li>— charakteryzować algorytmy rekurencyjne</li> <li>— określać złożoność obliczeniową algorytmów</li> <li>— charakteryzować typy sortowania i ich złożoność obliczeniową</li> <li>— projektować algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu</li> <li>— charakteryzować problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera</li> <li>— stosować różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj</li> <li>— stosować algorytmy wyszukiwania dla tablic, list, kolejek, stosów</li> </ul>
Projektowanie aplikacji	Narzędzia wspomagające projektowanie aplikacji	5	<ul style="list-style-type: none"> <li>— określać funkcje narzędzi do zarządzania projektem</li> <li>— korzystać z programów wspierających zarządzanie projektami, np. Jira, Trello</li> <li>— stosować diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta</li> <li>— korzystać z systemu kontroli wersji, np. Git</li> </ul>



<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>
Projektowanie aplikacji	Projekt aplikacji na podstawie wymagań klienta	15	<ul style="list-style-type: none"> <li>— tłumaczyć wymagania klienta na specyfikację techniczną dla zespołu programistów</li> <li>— identyfikować elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widgety</li> <li>— projektować aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym</li> <li>— projektować funkcjonalność aplikacji</li> <li>— określać cel projektu</li> <li>— analizować wymagania klienta i tworzy zgodny z nimi projekt</li> <li>— projektować interfejs użytkownika i wygląd aplikacji</li> <li>— dostosowywać interfejs do różnych platform</li> <li>— projektować aplikację opartą na architekturze klient-serwer</li> <li>— projektować struktury danych dla aplikacji</li> <li>— planować system zabezpieczeń aplikacji</li> <li>— dobierać wzorzec projektowy do zadania programistycznego</li> <li>— stosować wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)</li> </ul>
Projektowanie aplikacji	Planowanie prac w projekcie	25	<ul style="list-style-type: none"> <li>— korzystać z programów wspierających zarządzanie projektami, np. Jira, Trello</li> <li>— tłumaczyć wymagania klienta na specyfikację techniczną dla zespołu programistów</li> <li>— identyfikować elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widgety</li> <li>— projektować aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym</li> <li>— projektować funkcjonalność aplikacji</li> <li>— określać cel projektu</li> <li>— określać zasoby ludzkie oraz ramy czasowe wykonania projektu</li> <li>— określać etapy tworzenia aplikacji</li> <li>— organizować prace projektowe</li> <li>— stosować harmonogram czynności w celu efektywnego osiągnięcia celów</li> <li>— projektować zestawy danych dla problemu programistycznego</li> <li>— stosować diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta</li> </ul>



Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— korzystać z systemu kontroli wersji, np. Git</li> <li>— analizować wymagania klienta i tworzyć zgodny z nimi projekt</li> <li>— projektować interfejs użytkownika i wygląd aplikacji</li> <li>— dostosowywać interfejs do różnych platform</li> <li>— projektować aplikację opartą na architekturze klient-serwer</li> <li>— projektować struktury danych dla aplikacji</li> <li>— planować system zabezpieczeń aplikacji</li> <li>— określać fazy realizacji projektu</li> <li>— charakteryzować cykl życia projektu informatycznego i jego poszczególne etapy</li> <li>— korzystać z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)</li> <li>— dobierać optymalną metodologię zarządzania projektem</li> <li>— dobierać wzorzec projektowy do zadania programistycznego</li> <li>— stosować wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)</li> </ul>
Projektowanie aplikacji	Prawo autorskie i licencje	5	<ul style="list-style-type: none"> <li>— rozróżniać autorskie prawa osobiste i majątkowe</li> <li>— określać czas trwania praw autorskich</li> <li>— rozróżniać typy licencji oprogramowania</li> <li>— planować system zabezpieczeń aplikacji \</li> <li>— określać konsekwencje naruszenia prawa autorskiego</li> <li>— charakteryzować elementy własności intelektualnej (dobra niematerialne, własności przemysłowe)</li> </ul>

#### 4.3.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,

- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

### **Obudowa dydaktyczna**

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym projektowanie aplikacji.

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:

- dostępem do internetu,
- pakietem programów biurowych
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- projektowania aplikacji,
- typów danych,
- algorytmów,
- frameworków.

Wypożyczenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

## **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

### **4.3.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## **4.4. Program nauczania dla przedmiotu: Techniki programowania obiektowego**

### **4.4.1 Cele ogólne przedmiotu to:**

- Programowanie obiektowe.
- Stosowanie bibliotek.
- Obsługiwanie błędów.
- Programowanie dziedziczenia i hierarchii.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

### **4.4.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):**

- stosować różne typy danych,
- kompilować kod programu,
- debiutować kod programu,
- obsługiwać błędy i wyjątki w programie,
- stosować biblioteki programistyczne,
- tworzyć funkcje,
- tworzyć klasy,
- tworzyć obiekty,
- określać zasięg zmiennych,
- zarządzać hierarchią dziedziczenia,

- implementować algorytmy,
- wyszukiwać i poprawiać błędy.

#### 4.4.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Wstęp do programowania	Środowiska programistyczne, programowanie strukturalne	10	<ul style="list-style-type: none"> <li>— rozróżnia kompilatory i interpretery</li> <li>— charakteryzuje zadania kompilatora, interpretera, debuggera</li> <li>— charakteryzuje etapy kompilacji i interpretacji kodu</li> <li>— kompiluje i uruchamia programy</li> <li>— stosuje rekurencję</li> <li>— stosuje proste i złożone typy danych</li> <li>— deklaruje własne typy danych</li> <li>— deklaruje zmienne różnych typów danych</li> <li>— stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe</li> <li>— stosuje instrukcję warunkową i wyboru</li> <li>— stosuje instrukcje pętli</li> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— dzieli program na funkcje (metody)</li> <li>— implementuje algorytmy w programie</li> <li>— wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne</li> <li>— stosuje typy złożone i operacje na nich</li> <li>— wykorzystuje priorytety operatorów do właściwego budowania wyrażeń</li> </ul>
Wstęp do programowania	Programowanie z wykorzystaniem bibliotek	10	<ul style="list-style-type: none"> <li>— rozróżnia kompilatory i interpretery</li> <li>— charakteryzuje zadania kompilatora, interpretera, debuggera</li> <li>— charakteryzuje etapy kompilacji i interpretacji kodu</li> <li>— charakteryzuje pojęcie biblioteki</li> <li>— kompiluje i uruchamia programy</li> <li>— stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe</li> <li>— stosuje instrukcję warunkową i wyboru</li> <li>— stosuje instrukcje pętli</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— wykorzystuje priorytety operatorów do właściwego budowania wyrażeń</li> <li>— korzysta z wybranych bibliotek języka C++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami</li> </ul>
Wstęp do programowania	Niezawodność oprogramowania	20	<ul style="list-style-type: none"> <li>— kompiluje i uruchamia programy</li> <li>— stosuje instrukcję throw</li> <li>— definiuje obsługę dla błędów wykonania aplikacji</li> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— stosuje szkielet obsługi wyjątków z instrukcjami try i catch</li> <li>— opracowuje listę możliwych błędów wykonania aplikacji</li> </ul>
Programowanie obiektowe	Wstęp do programowania obiektowego	30	<ul style="list-style-type: none"> <li>— rozróżnia kompilatory i interpretery</li> <li>— charakteryzuje zadania kompilatora, interpretera, debuggera</li> <li>— charakteryzuje pojęcie biblioteki</li> <li>— kompiluje i uruchamia programy</li> <li>— stosuje rekurencję</li> <li>— stosuje proste i złożone typy danych</li> <li>— deklaruje własne typy danych</li> <li>— deklaruje zmienne różnych typów danych</li> <li>— stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe</li> <li>— stosuje instrukcję warunkową i wyboru</li> <li>— stosuje instrukcje pętli</li> <li>— stosuje obiektowe podejście do rozwiązywania problemów</li> <li>— charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm</li> <li>— powołuje obiekty</li> <li>— definiuje pola klasy</li> <li>— określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu</li> <li>— definiuje metody klasy</li> <li>— definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— definiuje instrukcje inicjujące konstruktora</li> <li>— definiuje składniki statyczne klasy</li> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— implementuje algorytmy w programie</li> <li>— wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne</li> <li>— stosuje typy złożone i operacje na nich</li> <li>— wykorzystuje priorytety operatorów do właściwego budowania wyrażeń</li> <li>— korzysta z wybranych bibliotek języka C++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami</li> <li>— dzieli zagadnienie na klasy</li> <li>— określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu</li> <li>— implementuje funkcjonalność klasy</li> <li>— deklaruje obiekty i odwołuje się obiektem do składowych klasy</li> </ul>
Programowanie obiektowe	Programowanie obiektowe	40	<ul style="list-style-type: none"> <li>— charakteryzuje pojęcie biblioteki</li> <li>— kompiluje i uruchamia programy</li> <li>— definiuje pola klasy</li> <li>— określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu</li> <li>— definiuje metody klasy</li> <li>— definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy</li> <li>— definiuje instrukcje inicjujące konstruktora</li> <li>— definiuje składniki statyczne klasy</li> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu</li> <li>— implementuje funkcjonalność klasy</li> <li>— deklaruje obiekty i odwołuje się obiektem do składowych klasy</li> </ul>
Programowanie obiektowe	Dziedziczenie i hierarchia	40	<ul style="list-style-type: none"> <li>— kompiluje i uruchamia programy</li> <li>— definiuje pola klasy</li> <li>— określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu</li> <li>— definiuje metody klasy</li> <li>— definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— definiuje instrukcje inicjujące konstruktora</li> <li>— definiuje składniki statyczne klasy</li> <li>— rozróżnia klasy dziedziczone i zaprzyjaźnione</li> <li>— stosuje składnik statyczny klasy i metody do ich obsługi</li> <li>— wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia</li> <li>— definiuje klasy bazowe i pochodne</li> <li>— definiuje szablony klas dla obsługi prostych typów liczbowych</li> <li>— analizuje błędy w kodzie za pomocą debuggera</li> <li>— określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu</li> <li>— implementuje funkcjonalność klasy</li> <li>— deklaruje obiekty i odwołuje się obiektem do składowych klasy</li> <li>— tworzy funkcje zaprzyjaźnione z klasą</li> <li>— buduje hierarchię dziedziczenia klas w programie</li> <li>— stosuje metody wirtualne, definiuje klasy abstrakcyjne</li> </ul>

#### 4.4.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna

stanowisko komputerowe dla osoby prowadzącej kurs z:

Program nauczania kursu umiejętności zawodowych  
**INF.04 Projektowanie, programowanie i testowanie aplikacji**

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:

- dostępem do internetu,
- pakietem programów biurowych
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- typów zmiennych,
- klas i obiektów,
- typowych języków programowania,
- frameworków,

Wyposażenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

### **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

#### **4.4.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.



## 4.5. Program nauczania dla przedmiotu: Tworzenie aplikacji desktopowych

### 4.5.1 Cele ogólne przedmiotu to:

- Programowanie aplikacji na komputery osobiste.
- Programowanie aplikacji okienkowych bazując na API dostarczonym przez system operacyjny.
- Programowanie aplikacji okienkowych bazując na frameworkach.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

### 4.5.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):

- zastosować środowiska IDE i RAD,
- stworzyć aplikacje desktopowe,
- stworzyć aplikacje okienkowe,
- stworzyć aplikacje okienkowe z typowymi elementami graficznych interfejsów użytkownika,
- zastosować język XAML do budowy interfejsów,
- zastosować frameworki do budowy aplikacji desktopowych.

### 4.5.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Aplikacje desktopowe	Środowiska i tworzenie aplikacji desktopowych	20	<ul style="list-style-type: none"> <li>— dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)</li> <li>— stosować środowiska IDE, RAD</li> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych</li> </ul>
Aplikacje okienkowe	Aplikacje okienkowe bazujące na OS API	30	<ul style="list-style-type: none"> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych</li> <li>— rozróżniać elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki)</li> <li>— stosować języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python</li> <li>— stosować język do projektowania interfejsu użytkownika np. XAML</li> <li>— programować okna aplikacji</li> <li>— programować system menu aplikacji</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— programować okna dialogowe aplikacji</li> <li>— programować obsługę zdarzeń myszy i klawiatury</li> </ul>
Aplikacje okienkowe	Aplikacje okienkowe bazujące na frameworkach	70	<ul style="list-style-type: none"> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych</li> <li>— charakteryzować pojęcie framework</li> <li>— rozróżniać elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki)</li> <li>— stosować języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python</li> <li>— stosować język do projektowania interfejsu użytkownika np. XAML</li> <li>— stosować frameworki typowe dla aplikacji desktopowych, np. WPF, Qt</li> <li>— programować okna aplikacji</li> <li>— programować system menu aplikacji</li> <li>— programować okna dialogowe aplikacji</li> <li>— programować obsługę zdarzeń myszy i klawiatury</li> </ul>

#### 4.5.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna

stanowisko komputerowe dla osoby prowadzącej kurs z:

Program nauczania kursu umiejętności zawodowych  
**INF.04 Projektowanie, programowanie i testowanie aplikacji**

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:

- dostępem do internetu,
- pakietem programów biurowych
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- tworzenia aplikacji okienkowych,
- elementów graficznego interfejsu użytkownika,
- typowych języków programowania,
- frameworków,

Wypożyczenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

### **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

#### 4.5.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

#### 4.6. Program nauczania dla przedmiotu: Tworzenie aplikacji mobilnych

##### 4.6.1 Cele ogólne przedmiotu to:

- Nabywanie umiejętności pracy w środowiskach programistycznych do tworzenia aplikacji mobilnych.
- Nabywanie umiejętności tworzenia aplikacji na urządzenia mobilne.
- Przygotowanie aplikacji do dystrybucji.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

##### 4.6.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):

- zastosować środowiska IDE i RAD,
- programować w językach typowych dla aplikacji mobilnych,
- stworzyć proste aplikacje mobilne,
- stworzyć aplikacje korzystające z zasobów urządzenia mobilnego,
- stworzyć aplikacje korzystające z baz danych,
- stworzyć aplikacje korzystające z Internetu,
- przygotowywać aplikacje do dystrybucji w sklepach.

##### 4.6.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Wstęp do aplikacji mobilnych	Środowiska do tworzenia aplikacji mobilnych	10	<ul style="list-style-type: none"> <li>— dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)</li> <li>— stosować środowiska IDE, RAD _1</li> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych</li> <li>— stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</li> <li>— programować aplikacje w jednym z systemów mobilnych: iOS lub Android</li> </ul>

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>
Wstęp do aplikacji mobilnych	Proste aplikacje mobilne	20	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _1</li> <li>— stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</li> <li>— programować interfejs użytkownika za pomocą języka XAML</li> <li>— programować aplikacje mobilne przy wykorzystaniu elementów UI</li> <li>— uruchamiać aplikacje mobilne</li> <li>— programować aplikacje w jednym z systemów mobilnych: iOS lub Android</li> <li>— identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)</li> <li>— programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</li> <li>— tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</li> </ul>
Aplikacje mobilne	Aplikacje z wykorzystaniem lokalnych nośników i baz danych	40	<ul style="list-style-type: none"> <li>— uruchamiać środowiska IDE, RAD _1</li> <li>— stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</li> <li>— programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych</li> <li>— programować interfejs użytkownika za pomocą języka XAML</li> <li>— programować aplikacje mobilne przy wykorzystaniu elementów UI</li> <li>— uruchamiać aplikacje mobilne</li> <li>— programować aplikacje w jednym z systemów mobilnych: iOS lub Android</li> <li>— identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)</li> <li>— programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</li> <li>— programować aplikację mobilną korzystającą z bazy danych</li> <li>— tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</li> </ul>
Aplikacje mobilne	Aplikacje z wykorzystaniem połączenia internetowego	40	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _1</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</li> <li>— programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych</li> <li>— programować interfejs użytkownika za pomocą języka XAML</li> <li>— programować aplikacje mobilne przy wykorzystaniu elementów UI</li> <li>— uruchamiać aplikacje mobilne</li> <li>— programować aplikacje w jednym z systemów mobilnych: iOS lub Android</li> <li>— identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)</li> <li>— programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</li> <li>— pobierać i wysyłać dane z lub do internetu dla aplikacji mobilnej</li> <li>— programować aplikację mobilną korzystającą z bazy danych</li> <li>— tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</li> </ul>
Aplikacje mobilne	Projekt, aplikacja przygotowana do dystrybucji w sklepie	40	<ul style="list-style-type: none"> <li>— dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)</li> <li>— stosować środowiska IDE, RAD_1</li> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych</li> <li>— stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</li> <li>— programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych</li> <li>— programować interfejs użytkownika za pomocą języka XAML</li> <li>— programować aplikacje mobilne przy wykorzystaniu elementów UI</li> <li>— uruchamiać aplikacje mobilne</li> <li>— programować aplikacje w jednym z systemów mobilnych: iOS lub Android</li> <li>— identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)</li> <li>— programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— dobierać i wysyłać dane z lub do internetu dla aplikacji mobilnej</li> <li>— programować aplikację mobilną korzystającą z bazy danych</li> <li>— tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</li> <li>— przygotowywać aplikacje do publikacji w sklepie</li> </ul>

#### 4.6.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji mobilnych
- zainstalowane emulatory urządzeń mobilnych z systemem Android.

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:



- dostępem do internetu,
- pakietem programów biurowych
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji mobilnych,
- zainstalowane emulatory urządzeń mobilnych z system Android lub fizyczne urządzenia.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- tworzenia aplikacji mobilnych,
- elementów interfejsu użytkownika,
- typowych języków programowania,
- frameworków.

Wyposażenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

### Warunki realizacji

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

### 4.6.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## 4.7. Program nauczania dla przedmiotu: Tworzenie aplikacji webowych

### 4.7.1 Cele ogólne przedmiotu to:

- Praca w środowiskach programistycznych do tworzenia aplikacji webowych.
- Tworzenie aplikacji webowych.



- Tworzenie aplikacji współpracujących z bazami danych.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

#### 4.7.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):

- zastosować środowiska IDE i RAD,
- programować w językach typowych dla aplikacji webowych zarówno po stronie serwera jak i klienta,
- wykorzystywać technologię AJAX,
- wykorzystywać JQuery,
- wykorzystywać frameworki,
- stworzyć aplikacje internetowe.

#### 4.7.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Aplikacje WEB	Środowiska do tworzenia aplikacji webowych	10	<ul style="list-style-type: none"> <li>— dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)</li> <li>— stosować środowiska IDE, RAD _2</li> <li>— rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji webowych</li> <li>— stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript</li> <li>— stosować mechanizm sesji użytkownika, ciasteczek</li> </ul>
Aplikacje WEB	Aplikacje działające po stronie klienta	30	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _2</li> <li>— wykorzystywać bibliotekę jQuery lub jej odpowiednik</li> <li>— programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny</li> <li>— programować wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego</li> </ul>
Aplikacje WEB	Aplikacje działające po stronie serwera	40	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _2</li> <li>— stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript</li> <li>— stosować mechanizm sesji użytkownika, ciasteczek</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— stosować frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js</li> <li>— programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny</li> <li>— programować aplikacje korzystające z bazy danych</li> </ul>
Projekt aplikacji	Aplikacje wykorzystujące AJAX	20	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _2</li> <li>— wykorzystywać bibliotekę jQuery lub jej odpowiednik</li> <li>— stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript</li> <li>— stosować mechanizm sesji użytkownika, ciasteczek</li> <li>— stosować frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js</li> <li>— programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny</li> <li>— programować aplikacje korzystające z bazy danych</li> </ul>
Projekt aplikacji	Projekt, aplikacja webowa	50	<ul style="list-style-type: none"> <li>— stosować środowiska IDE, RAD _2</li> <li>— wykorzystywać bibliotekę jQuery lub jej odpowiednik</li> <li>— stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript</li> <li>— stosować mechanizm sesji użytkownika, ciasteczek</li> <li>— stosować frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js</li> <li>— programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny</li> <li>— programować aplikacje korzystające z bazy danych</li> <li>— programować wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego</li> </ul>

#### 4.7.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji webowych,
- zainstalowanymi serwerami z obsługą PHP/Python/JS itp. i baz danych.

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:

- dostępem do internetu,
- pakietem programów biurowych
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji webowych,
- zainstalowanymi serwerami z obsługą PHP/Python/JS itp. i baz danych.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- tworzenia aplikacji webowych,
- AJAX,
- typowych języków programowania,
- frameworków,

Wyposażenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

### **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia. Warunkiem skutecznego kształcenia przedmiotu jest odpowiednie przygotowanie dydaktyczne osoby prowadzącej kurs, tworzenie aplikacji webowych jest stosunkowo nową dziedziną cechującą się niezwykle dużą dynamiką zmian technologicznych. Równie istotne jest wyposażenie pracowni w odpowiedni sprzęt i technologie związane z specyfiką tworzenia tego typu aplikacji.

### **4.7.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## **4.8. Program nauczania dla przedmiotu: Dokumentacja i testy aplikacji**

### **4.8.1 Cele ogólne przedmiotu to:**

- Nabywanie umiejętności tworzenia dokumentacji.
- Usuwanie błędów w aplikacjach.
- Optymalizowanie kodu.
- Testowanie aplikacji.
- Nabycie kompetencji personalnych i społecznych.

— Nabywanie umiejętności pracy w grupach i zespołach.

#### 4.8.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):

- udokumentować kod programu,
- udokumentować wyniki testów,
- stworzyć dokumentację użytkownika,
- stworzyć dokumentację wdrożeniową,
- zaplanować testy,
- przeprowadzać testy,
- wyszukiwać, analizować i poprawiać błędy.

#### 4.8.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Dokumentacja kodu i optymalizacja	Dokumentacja kodu programu	5	<ul style="list-style-type: none"> <li>— dobierać narzędzia i środowisko do testowania programów</li> <li>— stosować komentarze w kodzie źródłowym programu</li> <li>— pisać dokumentację kodu</li> <li>— przygotowywać dokumentację z wykonanych testów</li> </ul>
Dokumentacja kodu i optymalizacja	Usuwanie błędów i optymalizacja	10	<ul style="list-style-type: none"> <li>— dobierać narzędzia i środowisko do testowania programów</li> <li>— wykonywać testy tworzonych programów</li> <li>— identyfikować błędy w tworzonych programach</li> <li>— stosować komentarze w kodzie źródłowym programu</li> <li>— przeprowadzać testy w kolejnych fazach projektu informatycznego</li> <li>— poprawiać błędy w tworzonych programach, skryptach</li> <li>— optymalizować kod źródłowy</li> </ul>
Testowanie aplikacji	Testy funkcjonalności i interfejsów	10	<ul style="list-style-type: none"> <li>— wykonywać testy tworzonych programów</li> <li>— charakteryzować metodyki testowania</li> <li>— przeprowadzać testy funkcjonalne</li> <li>— przeprowadzać testy w kolejnych fazach projektu informatycznego</li> <li>— korzystać z systemów raportowania błędów, np. BugZilla, JIRA</li> <li>— przygotowywać środowiska testowe</li> <li>— raportować błędy po przeprowadzonych testach</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— poprawiać błędy w tworzonych programach, skryptach</li> <li>— przygotowywać testy funkcjonalne i нефункционалне</li> <li>— przeprowadzać testy нефункционалне: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa</li> <li>— testować prototyp projektu interfejsu</li> <li>— automatyzować proces testowania</li> <li>— testować aplikację</li> <li>— weryfikować zgodność aplikacji z oczekiwaniami klienta</li> </ul>
Testowanie aplikacji	Planowanie testów	5	<ul style="list-style-type: none"> <li>— przygotowywać dokumentację z wykonanych testów</li> <li>— pisać scenariusze testowania aplikacji</li> <li>— planować scenariusze testowania aplikacji</li> <li>— weryfikować zgodność aplikacji z oczekiwaniami klienta</li> </ul>
Testowanie aplikacji	Testy aplikacji	20	<ul style="list-style-type: none"> <li>— wykonywać testy tworzonych programów</li> <li>— przygotowywać dokumentację z wykonanych testów</li> <li>— przeprowadzać testy funkcjonalne</li> <li>— przeprowadzać testy w kolejnych fazach projektu informatycznego</li> <li>— korzystać z systemów raportowania błędów, np. BugZilla, JIRA</li> <li>— przeprowadzać testy interfejsu</li> <li>— przygotowywać środowiska testowe</li> <li>— pisać scenariusze testowania aplikacji</li> <li>— raportować błędy po przeprowadzonych testach</li> <li>— poprawiać błędy w tworzonych programach, skryptach</li> <li>— przygotowywać testy funkcjonalne i нефункционалне</li> <li>— przeprowadzać testy нефункционалне: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa</li> <li>— testować prototyp projektu interfejsu</li> <li>— planować scenariusze testowania aplikacji</li> <li>— automatyzować proces testowania</li> <li>— testować aplikację</li> <li>— weryfikować zgodność aplikacji z oczekiwaniami klienta</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
Testowanie aplikacji	Dokumentacja wdrożeniowa i instrukcje obsługi	10	<ul style="list-style-type: none"> <li>— pisać dokumentację kodu</li> <li>— pisać dokumenty pomocy programu</li> <li>— pisać instrukcję użytkownika aplikacji</li> <li>— pisać instrukcję użytkownika programu</li> <li>— pisać dokumentację wdrożenia projektu</li> </ul>

#### 4.8.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- metoda projektu (zalecana),
- metoda tekstu przewodniego,
- symulacje,
- gry dydaktyczne,
- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia praktyczne (zalecana),
- praca w grupach (zalecana),
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- monitorem minimum 21",
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

stanowisko komputerowe dla kursanta (1 stanowisko na osobę) z:

- dostępem do internetu,

- pakietem programów biurowych
- monitorem minimum 21”,
- zainstalowanymi środowiskami programistycznymi,
- zainstalowanym oprogramowaniem wspierającym testowanie.

W sali lekcyjnej powinny znajdować się e-booki, plansze dydaktyczne, schematy, grafiki i plansze interaktywne, gry edukacyjne, filmy instruktażowe (tutoriale), filmy edukacyjne, sekwencje filmowe, wizualizacje lub animacje 2D/3D, galerie zdjęć, symulatory, umożliwiające realizowanie treści w formie zdalnej i zawierające treści dotyczące:

- testowania aplikacji,
- dokumentacji,
- frameworków,

Wypożyczenie sali powinno obejmować między innymi zestawy instrukcji eksploatacji używanych na zajęciach urządzeń i przyrządów, zestawy ćwiczeń, instrukcje do ćwiczeń, komputerowe programy demonstracyjne i symulacyjne, czasopisma branżowe, katalogi, schematy ideowe i montażowe, normy ISO i PN, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne.

### **Warunki realizacji**

Zajęcia powinny być prowadzone z wykorzystaniem różnych form pracy aktywizującej słuchaczy np. praca w grupach po 2-3 słuchaczy. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb. W czasie prowadzenia zajęć w pracowni należy stosować zasadę, iż nieudane ćwiczenie też może być wysoko ocenione pod warunkiem, iż uczestnik kursu potrafi wyjaśnić przyczyny niepowodzenia oraz wskazać jak powinno ono przebiegać w prawidłowy sposób. Pozwoli to na indywidualizację prowadzonych działań przez słuchaczy oraz pokaże, iż doświadczenie można zdobyć nie tylko poprzez udane doświadczenia.

### **4.8.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

## **4.9. Program nauczania dla przedmiotu: Język obcy zawodowy**

### **4.9.1 Cele ogólne przedmiotu to:**

- Komunikowanie się w celu realizacji zadań zawodowych.
- Poznanie specjalistycznego słownictwa technicznego.
- Posługiwanie się terminologią i wiedzą specjalistyczną w języku angielskim.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.



#### 4.9.2 Cele szczegółowe przedmiotu, Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):

- posługiwać się dokumentacją techniczną w języku obcym,
- zrozumieć ze słuchu instruktażowe materiały wideo,
- przeprowadzić pisemną korespondencję techniczno-handlową,
- przeprowadzić konserwację związaną z realizacją zadań zawodowych,
- przeprowadzić negocjacje z klientami,
- opisywać wykonywane czynności zawodowe,
- skorzystać ze słowników technicznych i literatury specjalistycznej,
- przedstawiać swoje umiejętności i cechy osobowe.

#### 4.9.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia:

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
komunikacja w języku obcym	Słownictwo w branży informatycznej	10	<ul style="list-style-type: none"> <li>— rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</li> <li>— narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych</li> <li>— formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych</li> <li>— korzysta ze słownika dwujęzycznego i jednojęzycznego</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych</li> <li>— rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</li> <li>— czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy</li> <li>— procesów i procedur związanych z realizacją zadań zawodowych</li> <li>— świadczonych usług, w tym obsługi klienta</li> <li>— przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> <li>— upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
	Obsługa klienta w języku obcym	10	<ul style="list-style-type: none"> <li>— rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</li> <li>— narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych</li> <li>— formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych</li> <li>— określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu znajduje w wypowiedzi lub tekście określone informacje</li> <li>— rozpoczyna, prowadzi i kończy rozmowę</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— uzyskuje i przekazuje informacje i wyjaśnienia</li> <li>— opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi</li> <li>— stosuje zwroty i formy grzecznościowe</li> <li>— korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych</li> <li>— rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</li> <li>— czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy</li> <li>— procesów i procedur związanych z realizacją zadań zawodowych</li> <li>— e) świadczonych usług, w tym obsługi klienta</li> <li>— rozpoznaje związki między poszczególnymi częściami tekstu</li> <li>— przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</li> <li>— wyraża i uzasadnia swoje stanowisko</li> <li>— wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób</li> <li>— prowadzi proste negocjacje związane z czynnościami zawodowymi</li> <li>— dostosowuje styl wypowiedzi do sytuacji</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> <li>— wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa</li> <li>— upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne</li> </ul>



Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
	Komunikacja za pomocą słowa pisanego	10	<ul style="list-style-type: none"> <li>— określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu</li> <li>— znajduje w wypowiedzi lub tekście określone informacje</li> <li>— układa informacje w określonym porządku</li> <li>— stosuje zasady konstruowania tekstów o różnym charakterze</li> <li>— uzyskuje i przekazuje informacje i wyjaśnienia</li> <li>— stosuje zwroty i formy grzecznościowe</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych</li> <li>— rozpoznaje związki między poszczególnymi częściami tekstu</li> <li>— przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</li> <li>— wyraża i uzasadnia swoje stanowisko</li> <li>— stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji</li> <li>— wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób</li> <li>— prowadzi proste negocjacje związane z czynnościami zawodowymi</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> <li>— wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa</li> <li>— upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne</li> </ul>
	Poszukiwanie pracy	10	<ul style="list-style-type: none"> <li>— rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:</li> <li>— narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych</li> <li>— formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych</li> <li>— znajduje w wypowiedzi lub tekście określone informacje</li> <li>— układa informacje w określonym porządku</li> <li>— uzyskuje i przekazuje informacje i wyjaśnienia</li> <li>— stosuje zasady konstruowania tekstów o różnym charakterze</li> <li>— rozpoczyna, prowadzi i kończy rozmowę</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— stosuje zwroty i formy grzecznościowe</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych</li> <li>— rozpoznaje związki między poszczególnymi częściami tekstu</li> <li>— przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</li> <li>— wyraża i uzasadnia swoje stanowisko</li> <li>— stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji</li> <li>— wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób</li> <li>— dostosowuje styl wypowiedzi do sytuacji</li> <li>— przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> <li>— wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa</li> <li>— upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne</li> </ul>
<b>dokumentacja techniczna</b>	Pozyskiwanie informacji z źródeł obcojęzycznych	10	<ul style="list-style-type: none"> <li>— znajduje w wypowiedzi lub tekście określone informacje</li> <li>— uzyskuje i przekazuje informacje i wyjaśnienia</li> <li>— przekazuje w języku obcym nowożytnym informacje zawarte w materiałach wizualnych (np. wykresach, symbolach, piktogramach, schematach) oraz audiowizualnych (np. filmach instruktażowych)</li> <li>— przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym</li> <li>— korzysta ze słownika dwujęzycznego i jednojęzycznego</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— rozpoznaje związki między poszczególnymi częściami tekstu</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> </ul>
	Tworzenie dokumentacji technicznej	10	<ul style="list-style-type: none"> <li>— układa informacje w określonym porządku</li> <li>— opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi</li> <li>— stosuje zasady konstruowania tekstów o różnym charakterze</li> </ul>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji
			<ul style="list-style-type: none"> <li>— uzyskuje i przekazuje informacje i wyjaśnienia</li> <li>— korzysta ze słownika dwujęzycznego i jednojęzycznego</li> <li>— współdziała z innymi osobami, realizując zadania językowe</li> <li>— przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)</li> <li>— stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji</li> <li>— przekazuje w języku obcym nowożytnym informacje sformułowane w języku polskim lub w tym języku obcym nowożytnym</li> <li>— przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację</li> <li>— identyfikuje słowa kluczowe i internacjonalizmy</li> <li>— wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa</li> <li>— upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznanne słowa innymi, wykorzystuje opis, środki niewerbalne</li> </ul>

#### 4.9.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- ćwiczenia,
- inscenizacja,
- symulacja,
- metoda gier dydaktycznych,
- metoda projektów,
- metody doskonalące kompetencje komunikacyjne,
- użycie platform dydaktycznych,
- nauczanie na odległość z wykorzystaniem platform edukacyjnych.

##### Obudowa dydaktyczna,

W pracowni, w której prowadzone będą zajęcia edukacyjne, powinny znajdować się: sprzęt audiowizualny, tablica multimedialna/interaktywna, odtwarzacz CD/DVD (lub inny odtwarzacz plików dźwiękowych), słowniki jedno – i dwujęzyczne ogólne oraz techniczne, komputer z dostępem do Internetu, zestawy ćwiczeń, pakiety edukacyjne dla słuchaczy, obcojęzyczna dokumentacja techniczna urządzeń i instalacji elektronicznych, obcojęzyczna literatura branży teleinformatycznej, schematy układów

elektronicznych, katalogi elementów i układów elektronicznych. Wskazane jest, aby zajęcia prowadzone były w pracowni eksploatacji urządzeń elektronicznych. Nauka języka obcego może przebiegać z wykorzystaniem środków kształcenia na odległość. Dodatkowym atutem jest oswojenie uczestników kursu z coraz bardziej popularną formą kontaktu w biznesie i nauce w postaci e-komunikatorów, e-maili, platform społecznościowych itd.

### **Warunki realizacji**

Zajęcia powinny odbywać się w grupach do 12 osób, z podziałem na zespoły 2-3 osobowe. Dominująca forma organizacyjna pracy słuchaczy: indywidualna, zróżnicowana. Zajęcia powinny być prowadzone z wykorzystaniem zróżnicowanych form nauczania. Praca z większą grupą jest formą najbardziej efektywną podczas wprowadzania nowego materiału oraz pracy z materiałem audiowizualnym. Technika pracy w parach będzie najefektywniejsza podczas prowadzenia dialogów lub prezentowania inscenizacji. W przygotowaniu projektów najlepiej sprawdzi się metoda pracy w małej grupie. Praca indywidualna pozwoli na uczenie się i samodzielne wykonanie ćwiczeń we własnym tempie oraz wybraną przez siebie metodą.

Osoba prowadząca kurs realizujący przedmiot powinien współpracować z kadrą uczącą języka ogólnego, gdyż tylko dobra znajomość podstaw językowych może przybliżyć słuchacza do poznania języka specjalistycznego i posługiwania się nim podczas realizacji przyszłych zadań zawodowych. Jednocześnie należy zdawać sobie sprawę, że kurs języka obcego zawodowego w szkole ponadgimnazjalnej, z racji relatywnie małej liczby godzin, nie pozwoli słuchaczowi nabyć niezbędnej kompetencji językowej, a jedynie pozwoli na poznanie podstaw specjalistycznej komunikacji i słownictwa. Dalsza samoedukacja i zachęcenie słuchaczy do pogłębiania swojej wiedzy w tym zakresie będzie zatem jednym z kluczowych celów na tym etapie nauki.

W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb.

### **4.9.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika**

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych ćwiczeń. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną ćwiczeń, ich poprawność, formy przedstawienia. Sprawdzanie osiągnięć powinno odbywać się przez cały okres realizacji programu zajęć na podstawie kryteriów przedstawionych na początku kursu.

### **4.10. Program nauczania dla przedmiotu: Praktyka zawodowa**

Praktyka zawodowa odbywa się w semestrze II w wymiarze 140 godzin (Kolejne 140 godzin realizacji praktyki zawodowej dla zawodu technik programista w obrębie pierwszej kwalifikacji odbywają się w ramach KKZ INF.03 Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych opisanych w odrębnych plikach).

Miejsce realizacji praktyk zawodowych:

- firmy tworzące oprogramowanie,
- firmy zajmujące się rozwojem oprogramowania,
- firmy wdrażające i testujące oprogramowanie.

Forma praktyk zawodowych:

- wolontariat u pracodawcy,

- projekt edukacyjny we współpracy z pracodawcą,
- staż zawodowy u pracodawcy.

#### **4.10.1 Cele ogólne przedmiotu to:**

- Stosowanie różnych typów danych.
- Stosowanie różnych algorytmów.
- Umiejętność tworzenia projektu aplikacji na podstawie wymagań klienta.
- Planowanie prac projektowych.
- Licencjonowanie.
- Programowanie aplikacji na komputery osobiste.
- Programowanie aplikacji okienkowych bazując na API dostarczonym przez system operacyjny.
- Programowanie aplikacji okienkowych bazując na frameworkach.
- Nabycie umiejętności pracy w środowiskach programistycznych do tworzenia aplikacji mobilnych.
- Tworzenie aplikacji na urządzenia mobilne.
- Przygotowanie aplikacji do dystrybucji.
- Nabycie umiejętności pracy w środowiskach programistycznych do tworzenia aplikacji webowych.
- Tworzenie aplikacji webowych.
- Tworzenie aplikacji współpracujących z bazami danych.
- Tworzenie dokumentacji.
- Nabycie umiejętności testowania aplikacji.
- Nabycie kompetencji personalnych i społecznych.
- Nabywanie umiejętności pracy w grupach i zespołach.

#### **4.10.2 Cele szczegółowe przedmiotu (słuchacz/uczestnik potrafi):**

- zastosować różne typy danych,
- zastosować algorytmy,
- zastosować narzędzia wspomagające projektowanie aplikacji,
- współpracować z klientem,
- opracowywać funkcjonalność aplikacji,
- zaprojektować aplikację,
- stworzyć harmonogramy pracy,
- określać zasady licencjonowania,





- zaplanować system zabezpieczeń licencyjnych w aplikacjach.
- zastosować środowiska IDE i RAD,
- stworzyć aplikacje desktopowe,
- stworzyć aplikacje okienkowe,
- stworzyć aplikacje okienkowe z typowymi elementami graficznych interfejsów użytkownika,
- zastosować język XAML do budowy interfejsów,
- zastosować frameworki do budowy aplikacji desktopowych.
- zastosować środowiska IDE i RAD,
- stworzyć program językach typowych dla aplikacji mobilnych,
- stworzyć proste aplikacje mobilne,
- stworzyć aplikacje korzystające z zasobów urządzenia mobilnego,
- stworzyć aplikacje korzystające z baz danych,
- stworzyć aplikacje korzystające z Internetu,
- przygotowywać aplikacje do dystrybucji w sklepach.
- zastosować środowiska IDE i RAD,
- stworzyć program w językach typowych dla aplikacji webowych zarówno po stronie serwera jak i klienta,
- zastosować technologię AJAX,
- zastosować JQuery,
- zastosować frameworki,
- stworzyć aplikacje internetowe.
- udokumentować kod programu,
- udokumentować wyniki testów,
- stworzyć dokumentację użytkownika,
- stworzyć dokumentację wdrożeniową,
- nabyć umiejętności przeprowadzania testów oprogramowania,
- wyszukiwać, analizować i poprawiać błędy.



#### 4.10.3 Materiał nauczania z uwzględnieniem opisu efektów kształcenia

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji	
Wstęp do programowania	Rozwiązywanie problemów przy użyciu algorytmów	8	<p>charakteryzować algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe</p> <p>charakteryzować algorytmy rekurencyjne</p> <p>określać złożoność obliczeniową algorytmów</p> <p>charakteryzować typy sortowania i ich złożoność obliczeniową</p>	<p>projektować algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu</p> <p>charakteryzować problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera</p> <p>stosować różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj</p> <p>stosować algorytmy wyszukiwania dla tablic, list, kolejek, stosów</p>
Projektowanie aplikacji	Narzędzia wspomagające projektowanie aplikacji	4	<p>określać funkcje narzędzi do zarządzania projektem</p> <p>korzystać z programów wspierających zarządzanie projektami, np. Jira, Trello</p>	<p>stosować diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta</p> <p>korzystać z systemu kontroli wersji, np. Git</p>

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>	
Projektowanie aplikacji	Projekt aplikacji na podstawie wymagań klienta	8	<p>tłumaczyć wymagania klienta na specyfikację techniczną dla zespołu programistów</p> <p>identyfikować elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widżety</p> <p>projektować aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym</p> <p>projektować funkcjonalność aplikacji</p> <p>określać cel projektu</p>	<p>analizować wymagania klienta i tworzy zgodny z nimi projekt</p> <p>projektować interfejs użytkownika i wygląd aplikacji</p> <p>dostosowywać interfejs do różnych platform</p> <p>projektować aplikację opartą na architekturze klient-serwer</p> <p>projektować struktury danych dla aplikacji</p> <p>planować system zabezpieczeń aplikacji</p> <p>dobierać wzorzec projektowy do zadania programistycznego</p> <p>stosować wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)</p>
Projektowanie aplikacji	Planowanie prac w projekcie	25	<p>korzystać z programów wspierających zarządzanie projektami, np. Jira, Trello</p> <p>tłumaczyć wymagania klienta na specyfikację techniczną dla zespołu programistów</p> <p>identyfikować elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widżety</p> <p>projektować aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym</p> <p>projektować funkcjonalność aplikacji</p> <p>określać cel projektu</p> <p>określać zasoby ludzkie oraz ramy czasowe wykonania projektu</p> <p>określać etapy tworzenia aplikacji</p> <p>organizować prace projektowe</p> <p>stosować harmonogram czynności w celu efektywnego osiągnięcia celów</p>	<p>projektować zestawy danych dla problemu programistycznego</p> <p>stosować diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta</p> <p>korzystać z systemu kontroli wersji, np. Git</p> <p>analizować wymagania klienta i tworzy zgodny z nimi projekt</p> <p>projektować interfejs użytkownika i wygląd aplikacji</p> <p>dostosowywać interfejs do różnych platform</p> <p>projektować aplikację opartą na architekturze klient-serwer</p> <p>projektować struktury danych dla aplikacji</p> <p>planować system zabezpieczeń aplikacji</p> <p>określać fazy realizacji projektu</p> <p>charakteryzować cykl życia projektu informatycznego i jego poszczególne etapy</p>

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji	
				<p>korzystać z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)</p> <p>dobierać optymalną metodologię zarządzania projektem</p> <p>dobierać wzorzec projektowy do zadania programistycznego</p> <p>stosować wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)</p>
Projektowanie aplikacji	Prawo autorskie i licencje	5	<p>rozdzielić autorskie prawa osobiste i majątkowe</p> <p>określać czas trwania praw autorskich</p> <p>rozdzielić typy licencji oprogramowania</p>	<p>planować system zabezpieczeń aplikacji \</p> <p>określać konsekwencje naruszenia prawa autorskiego</p> <p>charakteryzować elementy własności intelektualnej (dobra niematerialne, własności przemysłowe)</p>
INF.04.5. Programowanie aplikacji desktopowych				
Aplikacje desktopowe	Środowiska i tworzenie aplikacji desktopowych	4	<p>dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)</p> <p>stosować środowiska IDE, RAD</p> <p>rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych</p>	

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>	
Aplikacje okienkowe	Aplikacje okienkowe bazujące na OS API	20	rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych rozdzielać elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki) stosować języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python stosować język do projektowania interfejsu użytkownika np. XAML	programować okna aplikacji programować system menu aplikacji programować okna dialogowe aplikacji programować obsługę zdarzeń myszy i klawiatury
Aplikacje okienkowe	Aplikacje okienkowe bazujące na frameworkach	16	rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych charakteryzować pojęcie framework rozdzielać elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki) stosować języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python stosować język do projektowania interfejsu użytkownika np. XAML	stosować frameworki typowe dla aplikacji desktopowych, np. WPF, Qt programować okna aplikacji programować system menu aplikacji programować okna dialogowe aplikacji programować obsługę zdarzeń myszy i klawiatury
<b>INF.04.6. Programowanie aplikacji mobilnych</b>				
Wstęp do aplikacji mobilnych	Środowiska do tworzenia aplikacji mobilnych	2	dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode) stosować środowiska IDE, RAD _1 rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#	programować aplikacje w jednym z systemów mobilnych: iOS lub Android
Wstęp do aplikacji mobilnych	Proste aplikacje mobilne	8	stosować środowiska IDE, RAD _1 stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#	programować aplikacje w jednym z systemów mobilnych: iOS lub Android

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>	
			programować interfejs użytkownika za pomocą języka XAML programować aplikacje mobilne przy wykorzystaniu elementów UI uruchamiać aplikacje mobilne	identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk) programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)
Aplikacje mobilne	Aplikacje z wykorzystaniem lokalnych nośników i baz danych	10	uruchamiać środowiska IDE, RAD _1 stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C# programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych programować interfejs użytkownika za pomocą języka XAML programować aplikacje mobilne przy wykorzystaniu elementów UI uruchamiać aplikacje mobilne	programować aplikacje w jednym z systemów mobilnych: iOS lub Android identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk) programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) programować aplikację mobilną korzystającą z bazy danych tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)
Aplikacje mobilne	Aplikacje z wykorzystaniem połączenia internetowego	10	stosować środowiska IDE, RAD _1 stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C# programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych	programować aplikacje w jednym z systemów mobilnych: iOS lub Android identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)

Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji	
			<p>programować interfejs użytkownika za pomocą języka XAML</p> <p>programować aplikacje mobilne przy wykorzystaniu elementów UI</p> <p>uruchamiać aplikacje mobilne</p>	<p>programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</p> <p>pobierać i wysyłać dane z lub do internetu dla aplikacji mobilnej</p> <p>programować aplikację mobilną korzystającą z bazy danych</p> <p>tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</p>
Aplikacje mobilne	Projekt, aplikacja przygotowana do dystrybucji w sklepie	5	<p>dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)</p> <p>stosować środowiska IDE, RAD _1</p> <p>rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych</p> <p>stosować języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#</p> <p>programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych</p> <p>programować interfejs użytkownika za pomocą języka XAML</p> <p>programować aplikacje mobilne przy wykorzystaniu elementów UI</p> <p>uruchamiać aplikacje mobilne</p>	<p>programować aplikacje w jednym z systemów mobilnych: iOS lub Android</p> <p>identyfikować elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)</p> <p>programować proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)</p> <p>dobierać i wysyłać dane z lub do internetu dla aplikacji mobilnej</p> <p>programować aplikację mobilną korzystającą z bazy danych</p> <p>tworzyć aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)</p> <p>przygotowywać aplikacje do publikacji w sklepie</p>
INF.04.7. Programowanie aplikacji zaawansowanych webowych				

<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>	
Aplikacje WEB	Środowiska do tworzenia aplikacji webowych	2	1) dobierać środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains) 2) stosować środowiska IDE, RAD _2 3) rozpoznawać narzędzia wykorzystywane w procesie tworzenia aplikacji webowych 1) stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript 2) stosować mechanizm sesji użytkownika, ciasteczek	
Aplikacje WEB	Aplikacje działające po stronie klienta	10	2) stosować środowiska IDE, RAD _2 2) wykorzystywać bibliotekę JQuery lub jej odpowiednik	3) programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny 5) programować wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego
Aplikacje WEB	Aplikacje działające po stronie serwera	10	2) stosować środowiska IDE, RAD _2 1) stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript 2) stosować mechanizm sesji użytkownika, ciasteczek	1) stosować frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js 3) programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny 4) programować aplikacje korzystające z bazy danych



<b>Dział programowy</b>	<b>Tematy jednostek metodycznych</b>	<b>liczba godzin</b>	<b>Opis efektów kształcenia uwzględniający kryteria weryfikacji</b>	
Projekt aplikacji	Aplikacje wykorzystujące AJAX	8	2) stosować środowiska IDE, RAD _2 2) wykorzystywać bibliotekę jQuery lub jej odpowiednik 1) stosować języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript 2) stosować mechanizm sesji użytkownika, ciasteczek	1) stosować frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js 3) programować dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny 4) programować aplikacje korzystające z bazy danych
<b>INF.04.8. Testowanie i dokumentowanie aplikacji</b>				
Dokumentacja kodu i optymalizacja	Dokumentacja kodu programu	5	1) dobierać narzędzia i środowisko do testowania programów 1) stosować komentarze w kodzie źródłowym programu 2) pisać dokumentację kodu 7) przygotowywać dokumentację z wykonanych testów	
Testowanie aplikacji	Testy funkcjonalności i interfejsów	5	2) wykonywać testy tworzonych programów 1) charakteryzować metodyki testowania 3) przeprowadzać testy funkcjonalne 4) przeprowadzać testy w kolejnych fazach projektu informatycznego 6) korzystać z systemów raportowania błędów, np. BugZilla, JIRA 9) przygotowywać środowiska testowe 12) raportować błędy po przeprowadzonych testach	4) poprawiać błędy w tworzonych programach, skryptach 2) przygotowywać testy funkcjonalne i niefunkcjonalne 5) przeprowadzać testy niefunkcjonalne: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa 8) testować prototyp projektu interfejsu 13) automatyzować proces testowania 14) testować aplikację 15) weryfikować zgodność aplikacji z oczekiwaniami klienta



Dział programowy	Tematy jednostek metodycznych	liczba godzin	Opis efektów kształcenia uwzględniający kryteria weryfikacji	
Testowanie aplikacji	Testy aplikacji	5	2) wykonywać testy tworzonych programów 7) przygotowywać dokumentację z wykonanych testów 3) przeprowadzać testy funkcjonalne 4) przeprowadzać testy w kolejnych fazach projektu informatycznego 6) korzystać z systemów raportowania błędów, np. BugZilla, JIRA 7) przeprowadzać testy interfejsu 9) przygotowywać środowiska testowe 11) pisać scenariusze testowania aplikacji 12) raportować błędy po przeprowadzonych testach	4) poprawiać błędy w tworzonych programach, skryptach 2) przygotowywać testy funkcjonalne i niefunkcjonalne 5) przeprowadzać testy niefunkcjonalne: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa 8) testować prototyp projektu interfejsu 10) planować scenariusze testowania aplikacji 13) automatyzować proces testowania 14) testować aplikację 15) weryfikować zgodność aplikacji z oczekiwaniami klienta

#### 4.10.4 Procedury osiągnięcia celów kształcenia

##### Propozycje metod nauczania,

- pokaz z objaśnieniem,
- pokaz z instruktażem,
- ćwiczenia w zakładzie pracy w naturalnych warunkach pracy.
- ćwiczenia w terenie w naturalnych warunkach pracy.

##### Obudowa dydaktyczna

Uczestnik kwalifikacyjnego kursu zawodowego podczas praktyki zawodowej powinien realizować jej treści na wyposażeniu przedsiębiorstwa, firmy bądź instytucji, do której uczęszcza. Może realizować praktykę zawodową m. in:

- w firmach tworzących oprogramowanie,
- w firmach zajmujących się rozwojem oprogramowania,
- w firmach wdrażających oprogramowanie,
- w firmach testujących oprogramowanie.

Wymienione miejsca realizacji praktyki zawodowej powinny posiadać wyposażenie umożliwiające realizację efektów kształcenia podczas praktyki zawodowej.

## Warunki realizacji

Praktyka zawodowa powinna być prowadzona w pracowniach szkolnych, warsztatach szkolnych, Centrach Kształcenia Praktycznego lub u pracodawców mających możliwość realizacji programu praktyk, tzn. mających odpowiednie urządzenia, narzędzia i wyposażenie oraz odpowiednią kadrę z obszaru programowania, zapewniające rzeczywiste warunki pracy właściwe dla technika programisty, a także kontakt z nowoczesnymi technikami i technologiami.

Zajęcia należy prowadzić metodą ćwiczeń praktycznych oraz stosując metody aktywizujące słuchaczy. Z uwagi na bezpieczeństwo słuchaczy zajęcia powinny być prowadzone w grupach nie większych niż 8 osób, a podczas wykonywania ćwiczeń słuchacze powinni pracować w grupach max. 2-osobowych. W trakcie prac ze słuchaczami należy pozostawiać im dodatkowy czas na własne prace związane z realizowanymi celami kształcenia. Dodatkowy czas należy też poświęcić na indywidualizowanie pracy słuchaczy w zależności od ich możliwości i potrzeb.

### 4.10.5 Proponowane metody sprawdzania osiągnięć edukacyjnych słuchacza/uczestnika

Sprawdzanie opanowania przez słuchacza/uczestnika wymagań programowych będzie przeprowadzone na podstawie wykonanych prac i zadań. W ocenie należy uwzględnić następujące kryteria ogólne: zawartość merytoryczną zadań, ich poprawność, formy przedstawienia. Osiągnięcia ucznia oceniać na bieżąco będzie opiekun praktyki. Na zakończenie praktyki uczestnik kursu musi przedłożyć opiekunowi dziennik praktyki oraz przygotowane portfolio z dokumentacją wykonywanych podczas praktyki zadań.

## 5. Ewaluacja programu Kwalifikacyjnego kursu zawodowego

<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
INF.04.1. Bezpieczeństwo i higiena pracy			
2) stosuje środki ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych na stanowisku pracy (ek)	1) identyfikuje środki ochrony zbiorowej 2) wskazuje środki ochrony zabezpieczające przed hałasem w pracy biurowej 3) identyfikuje wymagania w zakresie oświetlenia, temperatury i 4) rozpoznaje środki ochrony zapobiegające porażeniem prądem w pracy biurowej 5) rozpoznaje środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa 6) dobiera środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowe	wykład informacyjny, pokaz z objaśnieniem, wykład problemowy, film dydaktyczny, dyskusja dydaktyczna, burza mózgów, pokaz z instruktażem, pokaz z objaśnieniem, ćwiczenia przedmiotowe, ćwiczenia produkcyjne,	W czasie realizacji programu nauczania podczas trwania kursu dodatkowych umiejętności zawodowych



<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
3) udziela pierwszej pomocy w stanach nagłego zagrożenia zdrowotnego (ek)	1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego 2) ocenia sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego 3) zabezpiecza siebie, poszkodowanego i miejsce wypadku 4) układa poszkodowanego w pozycji bezpiecznej 5) powiadamia odpowiednie służby 6) prezentuje udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie 7) prezentuje udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar 8) wykonuje resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji	metoda projektów, metoda przewodniego tekstu, metoda przypadków, metoda sytuacyjna, inscenizacja, dyskusja dydaktyczna, gry dydaktyczne, teksty zamknięte, próby pracy, testy zamknięte, praca w grupie	
INF.04.2. Podstawy informatyki			
5) posługuje się terminologią dotyczącą sieci komputerowych (ek)	1) wymienia topologie sieci 2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej 3) opisuje sieć bezprzewodową oraz sieć przewodową 4) oblicza przepustowość sieci 5) definiuje pojęcia pobieranie i wysyłanie danych 6) opisuje zasady działania sieci synchronicznej i asynchronicznej 7) wykazuje różnice w działaniu sieci synchronicznej i asynchronicznej 8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci		



<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
	9) używa komunikatorów tekstowych, audio-video oraz tablic interaktywnych		
	10) stosuje zasadę netykiety		
6) stosuje pozycyjne systemy liczbowe (ek)	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) liczbowych		
7) stosuje zasady cyberbezpieczeństwa (ek)	1) rozróżnia rodzaje szkodliwego oprogramowania		
	2) rozróżnia rodzaje ataków hakerskich		
	3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi		
	4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni		
	5) opisuje zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom		
	6) przestrzega zasad bezpiecznego przechowywania danych		
	7) przestrzega zasad bezpieczeństwa swojego cyfrowego wizerunku i tożsamości		
	8) przestrzega zasad prywatności w cyfrowym świecie		



Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)	Wskaźniki potwierdzające osiągnięcie efektu kształcenia	Metody/techniki badania	Termin badania
	9) wymienia i omawia podstawowe pojęcia związane z ochroną danych osobowych, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony 10) stosuje zasady dokonywania bezpiecznych transakcji w internecie		
INF.04.3. Projektowanie oprogramowania			
1) posługuje się prostymi typami danych (ek)	1) rozróżnia typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe 2) rozpoznaje typ logiczny 3) rozróżnia typy znakowe i łańcuchowe 4) posługuje się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym		
2) posługuje się złożonymi typami danych (ek)	1) rozróżnia rodzaje złożonych typów danych 2) posługuje się tablicami jednowymiarowymi i dwuwymiarowymi 3) posługuje się tablicami dynamicznymi, asocjacyjnymi 4) posługuje się typem rekordowym, np. struktura, unia 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		
3) stosuje metody rozwiązywania problemów za pomocą algorytmów (ek)	1) projektuje algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu 2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe		



Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)	Wskaźniki potwierdzające osiągnięcie efektu kształcenia	Metody/techniki badania	Termin badania
	3) charakteryzuje algorytmy rekurencyjne 4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera 5) określa złożoność obliczeniową algorytmów 1) charakteryzuje typy sortowania i ich złożoność obliczeniową 2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwyciężaj 3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów		
6) projektuje aplikację (ek)	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, widgety 4) projektuje interfejs użytkownika i wygląd aplikacji 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		
INF.04.4. Programowanie obiektowe			
3) korzysta z typów danych (ek)	1) stosuje proste i złożone typy danych 2) deklaruje własne typy danych 3) deklaruje zmienne różnych typów danych 4) wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne 5) stosuje typy złożone i operacje na nich		



<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
4) stosuje wyrażenia, instrukcje i biblioteki (ek)	1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe 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++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami		
5) stosuje zasady programowania obiektowego (ek)	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		
6) definiuje klasy (ek)	1) definiuje pola klasy 2) określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu 3) definiuje metody klasy 4) definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy 5) definiuje instrukcje inicjujące konstruktora 6) określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu 7) implementuje funkcjonalność klasy 8) deklaruje obiekty i odwołuje się obiektem do składowych klasy		





<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
	9) definiuje składniki statyczne klasy 10) rozróżnia klasy dziedziczone i zaprzyjaźnione 11) tworzy funkcje zaprzyjaźnione z klasą 12) stosuje składnik statyczny klasy i metody do ich obsługi		
INF.04.5. Programowanie aplikacji desktopowych			
2) wykorzystuje frameworki do programowania aplikacji desktopowych (ek)	1) charakteryzuje pojęcie framework 2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt		
3) programuje desktopowe aplikacje okienkowe (ek)	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		
INF.04.6. Programowanie aplikacji mobilnych			
2) programuje aplikacje mobilne (ek)	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, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)		





<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
	5) programuje interfejs użytkownika za pomocą języka XAML 6) programuje aplikacje mobilne przy wykorzystaniu elementów UI 7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej) 8) pobiera i wysyła dane z lub 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		
<b>INF.04.7. Programowanie aplikacji zaawansowanych webowych</b>			
2) wykorzystuje frameworki do programowania aplikacji webowych (ek)	1) stosuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js 2) wykorzystuje bibliotekę jQuery lub jej odpowiednik		
3) programuje zaawansowane aplikacje webowych (ek)	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		
<b>INF.04.8. Testowanie i dokumentowanie aplikacji</b>			
1) przeprowadza walidację kodu programu (ek)	1) dobiera narzędzia i środowisko do testowania programów		



Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)	Wskaźniki potwierdzające osiągnięcie efektu kształcenia	Metody/techniki badania	Termin badania
	2) wykonuje testy tworzonych programów 3) identyfikuje błędy w tworzonych programach 4) poprawia błędy w tworzonych programach, skryptach 5) optymalizuje kod źródłowy		
3) przeprowadza testy (ek)	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 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		
INF.04.9. Język obcy zawodowy			
	1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie: a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych		

<b>Efekt kształcenia z podstawy programowej (oznaczony w programie kursu jako kluczowy dla kwalifikacji lub jednostki efektów) (czy Cele szczegółowe przedmiotu to (słuchacz/uczestnik potrafi):)</b>	<b>Wskaźniki potwierdzające osiągnięcie efektu kształcenia</b>	<b>Metody/techniki badania</b>	<b>Termin badania</b>
1) posługuje się podstawowym zasobem środków językowych w języku obcym nowożytnym (ze szczególnym uwzględnieniem środków leksykalnych) umożliwiającym realizację czynności zawodowych w zakresie tematów związanych: a) ze stanowiskiem pracy i jego wyposażeniem b) z głównymi technologiami stosowanymi w danym zawodzie c) z dokumentacją związaną z danym zawodem d) z usługami świadczonymi w danym zawodzie (ek)	c) procesów i procedur związanych z realizacją zadań zawodowych		
	d) formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych		
	e) świadczonych usług, w tym obsługi klienta		

## 6. Wykaz literatury oraz niezbędnych środków i materiałów dydaktycznych

### 6.1. Wykaz literatury

#### Proponowane podręczniki:

- Kwalifikacja INF.04. Projektowanie, programowanie i testowanie aplikacji. Część 1. Inżynieria programowania - projektowanie oprogramowania, testowanie i dokumentowanie aplikacji. Podręcznik do nauki zawodu technik programista,
- Testowanie oprogramowania. Podręcznik dla początkujących, Rafał Pawlak,
- Zarządzanie projektami informatycznymi. Subiektywne spojrzenie programisty, Joel Spolsky,
- Zarządzanie projektami IT, Joseph Philips,
- Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides.

#### Literatura dodatkowa:

- Krzysztof Szczęch, Wanda Buwała, Bezpieczeństwo i higiena pracy, Podręcznik do kształcenia zawodowego. WSiP. Warszawa 2016.
- Marcin Czerwonka, Zenon Nowocień Kwalifikacja INF.02. Administracja i eksploatacja systemów komputerowych, urządzeń peryferyjnych i lokalnych sieci komputerowych. Część 1. Systemy komputerowe. Podręcznik do nauki zawodu technik informatyk, wyd. Helion,

- Jolanta Pokorska, Kwalifikacja INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych. Część 2. Projektowanie i administrowanie bazami danych. Podręcznik do nauki zawodu technik informatyk i technik programista, wyd. Helion,
- Jolanta Pokorska, Podręcznik do zawodu technik informatyk, technik programista, część 1, Tworzenie stron internetowych, Kwalifikacja INF.03.Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych, wyd. Helion –podręcznik w przygotowaniu.
- Adam Freeman, HTML5. Przewodnik encyklopedyczny, wyd. Helion
- David Sawyer McFarland, CSS3 nieoficjalny podręcznik, wyd. Helion
- David Sawyer McFarland, JavaScript i jQuery. Nieoficjalny podręcznik, wyd. Helion,
- Eric T. Freeman, Elisabeth Robson, Programowanie w Javascript. Rusz głową!, wyd.Helion
- Luke Welling, Laura Thomson, PHP i MySQL. Tworzenie stron WWW. Vademecum profesjonalisty, wyd. Helion
- R. Sama, K. Sama, Język angielski zawodowy w branży informatycznej, wyd. WSiP, Warszawa 2016
- Bjarne Stroustrup, Język C++. Kompendium wiedzy, wyd. Helion,
- Stephen Prata, Język C++. Szkoła programowania. Wydanie VI, wyd. Helion,
- Grębosz Jerzy, Opus magnum C++11, Programowanie w języku C++ (komplet), Wyd.: Grębosz Jerzy,
- Python dla każdego. Podstawy programowania – Michael Dawson, wyd. Helion,
- Eric Matthes, Python. Instrukcje dla programisty, wyd. Helion,
- Al Sweigart, Automatyzacja nudnych zadań z Pythonem, wyd. Helion,
- Dawn Griffiths, David Griffiths, Android. Programowanie aplikacji. Rusz głową! Wydanie II, wyd. Helion,
- Marcin Płonkowski, Android Studio. Tworzenie aplikacji mobilnych. wyd. Helion,
- Matt Neuburg, iOS 12. Wprowadzenie do programowania w Swiftie. Wydanie V, wyd. Helion,
- Steven F. Daniel, Xamarin. Tworzenie interfejsów użytkownika, wyd. Helion.
- Gniewomir Sarbicki, Python. Kurs dla osoba prowadząca kursy i studentów, wyd. Helion,
- Adriaan de Jonge, Phillip Dutson, jQuery, jQuery UI oraz jQuery Mobile. Receptury, wyd. Helion,
- Jon Duckett, JavaScript i jQuery. Interaktywne strony WWW dla każdego. Podręcznik Front-End Developera wyd. Helion,
- David Herron, Platforma Node.js. Przewodnik webdevelopera. Wydanie III, wyd. Helion,
- Antonio Mele, Django 2. Praktyczne tworzenie aplikacji sieciowych. Wydanie II, wyd. Helion.
- <http://codecademy.com>
- <http://w3schools.com>
- <http://www.cplusplus.com/>
- <https://docs.python.org/3/tutorial/>
- <https://doc.qt.io/qtcreator/index.html>
- <http://www-cs.cuny.cuny.edu/~wolberg/cs221/qt/books/C++-GUI-Programming-with-Qt-4-1st-ed.pdf>
- [https://qmlbook.github.io/assets/qt5\\_cadaques.pdf](https://qmlbook.github.io/assets/qt5_cadaques.pdf)

### **Czasopisma branżowe:**

- Programista (programista.pl),

## **6.2. Wykaz niezbędnych środków i materiałów dydaktycznych**

Pracownia programistyczna:

stanowisko komputerowe dla osoby prowadzącej kurs z:

- dostępem do internetu,
- pakietem programów biurowych,
- projektorem multimedialnym lub tablicą multimedialną,
- urządzenie wielofunkcyjne sieciowe (drukarka z skanerem),
- środowiska programistyczne,
- monitorem minimum 24",

Stanowiska komputerowe kursantów (1 stanowisko na 1 osobę) powinny być wyposażone następująco:

- komputer z połączeniem do internetu z zainstalowanymi środowiskami programistycznymi i ekranem minimum 21",
- zainstalowane stosowne środowiska programistyczne,
- zainstalowane oprogramowanie biurowe,
- zainstalowane oprogramowanie do obsługi zgłoszeń i błędów,
- zainstalowane oprogramowanie do zarządzania i kontroli wersji oprogramowania,
- zainstalowane oprogramowanie wspierające testowanie aplikacji,
- zainstalowane oprogramowanie wspierające projektowanie aplikacji,
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji webowych,
- zainstalowanymi serwerami z obsługą PHP/Python/JS itp. i baz danych.
- zainstalowanymi bibliotekami i frameworkami do tworzenia aplikacji mobilnych,
- zainstalowane emulatory urządzeń mobilnych z system Android.

W salach lekcyjnych oraz pomieszczeniach praktycznej nauki zawodu powinny znajdować się przepisy BHP dotyczące pracy z urządzeniami oraz instrukcje obsługi i konserwacji tych urządzeń. Niezbędne są również zestawy ćwiczeń, instrukcje do ćwiczeń, pakiety edukacyjne dla słuchaczy, karty samooceny, karty pracy dla słuchaczy i prezentacje multimedialne związane z treściami kształcenia, czasopisma branżowe, katalogi, normy ISO i PN.

Dodatkowo do dyspozycji wskazane są tematyczne e-booki, animacje 2D/3D, atlasy interaktywne, plansze interaktywne, gry edukacyjne, filmy edukacyjne, symulatory, wirtualne laboratoria, programy ćwiczeniowe do projektowania przez dobieranie umożliwiające realizowanie treści w formie zdalnej dostosowane treściami do poszczególnych przedmiotów teoretycznych i praktycznych.

## 7. Sposób i forma zaliczenia kursu

Kwalifikacyjny kurs zawodowy kończy się zaliczeniem w formie walidacji osiągnięć uczestnika kursu, polegającej na ocenie wykonywanych w trakcie nauki projektów i ćwiczeń oraz na podstawie uzyskanych w trakcie kursu ocen z poszczególnych przedmiotów.

Do oceny osiągnięć edukacyjnych słuchaczy proponuje się stosowanie testów wielokrotnego wyboru, zadań z luką, ocenę aktywności słuchacza podczas wykonywania zadań w grupie, ocenę jakości wykonania zadań przez słuchacza. Proponuje się, aby osiągnięcia słuchaczy oceniać w zakresie zaplanowanych, uszczegółowionych celów kształcenia na podstawie:

- obserwacji wykonanych ćwiczeń,
- testu pisemnego.

Umiejętności praktyczne proponuje się sprawdzać na podstawie obserwacji czynności wykonywanych przez słuchacza w trakcie realizacji ćwiczeń. Podczas obserwacji należy zwrócić uwagę na:

- wyszukiwanie i przetwarzanie rzetelnych informacji pozyskanych z różnych źródeł,
- poprawność merytoryczną wykonanych ćwiczeń praktycznych,
- umiejętność pracy w zespole.

Ważne kryteria oceny efektów kształcenia to: zaplanowanie wykonania zadania, dobór elementów oraz sporządzona dokumentacja techniczna. Możliwe są również inne sposoby i formy zaliczenia, takie jak: testy praktyczne, wykonanie projektów, próby pracy, aktywność uczącego się na zajęciach, prezentacje na forum grupy z przeprowadzonych prac.

Osoba, która uzyskała zaliczenie, otrzymuje zaświadczenie o ukończeniu kwalifikacyjnego kursu zawodowego. Osoba, która ukończyła kwalifikacyjny kurs zawodowy i otrzymała zaświadczenie o jego ukończeniu może przystąpić do egzaminu potwierdzającego kwalifikację INF.04 Projektowanie, programowanie i testowanie aplikacji.

## 8. Sprawdzenie kompletności i poprawności opracowanego programu zajęć

**Tabela 5.** Tabela weryfikacji programu nauczania kursu dodatkowych umiejętności zawodowych pod kątem zgodności z przepisami prawa oświatowego

Lp.	Program kursu dodatkowych umiejętności zawodowych uwzględnia	Zawartość opracowanego programu zajęć (T/N)
1.	Cele kształcenia (zadania zawodowe)	T
2.	Efekty kształcenia	T
3.	Kryteria weryfikacji	T
4.	Warunki realizacji kształcenia w kwalifikacji (lub niezbędne do realizacji danej jednostki efektów)	T
5.	Minimalna liczba godzin kształcenia zawodowego dla kwalifikacji wyodrębnionej w zawodzie lub jednostki efektów	T

**Tabela 6.** Tabela weryfikacji programu kursu dodatkowych umiejętności zawodowych pod kątem kompletności efektów kształcenia

<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
<b>INF.04.1. Bezpieczeństwo i higiena pracy</b>		
1) określa skutki oddziaływania czynników szkodliwych na organizm człowieka (ew)	1) określa zagrożenia występujące w środowisku pracy	Ochrona pracownika
	2) określa skutki oddziaływania czynników fizycznych na organizm człowieka	Ochrona pracownika
	3) określa skutki oddziaływania czynników psychofizycznych na organizm człowieka	Ochrona pracownika
	4) opisuje skutki oddziaływania czynników niebezpiecznych i uciążliwych na organizm człowieka	Ochrona pracownika
	5) wyjaśnia pojęcia choroba zawodowa i wypadek przy pracy	Ochrona pracownika
2) stosuje środki ochrony indywidualnej i zbiorowej podczas wykonywania zadań zawodowych na stanowisku pracy (ek)	1) identyfikuje środki ochrony zbiorowe	Ochrona pracownika
	2) wskazuje środki ochrony zabezpieczające przed hałasem w pracy biurowej	Ochrona pracownika
	3) identyfikuje wymagania w zakresie oświetlenia, temperatury i	Ochrona pracownika
	4) rozpoznaje środki ochrony zapobiegające porażeniem prądem w pracy biurowej	Ochrona pracownika
	5) rozpoznaje środki ochrony zapobiegające pogorszeniu wzroku i zniekształceniu kręgosłupa	Ochrona pracownika
	6) dobiera środki ochrony zbiorowej do rodzaju zagrożeń w pracy biurowe	Ochrona pracownika
3) udziela pierwszej pomocy w stanach nagłego zagrożenia zdrowotnego (ek)	1) opisuje podstawowe symptomy wskazujące na stany nagłego zagrożenia zdrowotnego	Niesienie pomocy
	2) ocenia sytuację poszkodowanego na podstawie analizy objawów obserwowanych u poszkodowanego	Niesienie pomocy
	3) zabezpiecza siebie, poszkodowanego i miejsce wypadku	Niesienie pomocy
	4) układa poszkodowanego w pozycji bezpiecznej	Niesienie pomocy
	5) powiadamia odpowiednie służby	Niesienie pomocy
	6) prezentuje udzielanie pierwszej pomocy w urazowych stanach nagłego zagrożenia zdrowotnego, np. krwotok, zmiążdżenie, amputacja, złamanie, oparzenie	Niesienie pomocy





Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	7) prezentuje udzielanie pierwszej pomocy w nieurazowych stanach nagłego zagrożenia zdrowotnego, np. omdlenie, zawał, udar	Niesienie pomocy
	8) wykonuje resuscytację krążeniowo-oddechową na fantomie zgodnie z wytycznymi Polskiej Rady Resuscytacji i Europejskiej Rady Resuscytacji	Niesienie pomocy
INF.04.2. Podstawy informatyki		
1) charakteryzuje parametry sprzętu komputerowego (ew)	1) identyfikuje parametry urządzeń techniki komputerowej	Sprzęt komputerowy
	2) porównuje parametry tego samego typu urządzeń techniki komputerowej (np. dwie karty graficzne, dwa dyski twarde)	Sprzęt komputerowy
	3) przelicza jednostki pojemności pamięci masowych	Systemy liczbowe Sprzęt komputerowy
	4) dobiera urządzenia techniki komputerowej zgodnie z wymaganiami technicznymi stanowiska	Sprzęt komputerowy
2) definiuje elementy architektury systemów komputerowych (ew)	1) opisuje zasadę działania procesora (rozkazy)	Sprzęt komputerowy
	2) wymienia zależności między pamięcią operacyjną, procesorem i pozostałymi elementami systemu komputerowego	Sprzęt komputerowy
3) charakteryzuje systemy informatyczne oraz rozróżnia systemy informatyczne pod względem funkcjonalności (ew)	1) identyfikuje system informatyczny	Systemy informatyczne
	2) podaje przykłady systemów informacji przetwarzanych przez system informatyczny, w tym system PESEL, system postępowania rekrutacyjnego do szkół, e-dziennik, system bankowości elektronicznej, profil zaufany	Systemy informatyczne
	3) opisuje miejsca przechowywania informacji: serwer lokalny, chmura, nośniki danych	Systemy informatyczne
	4) dobiera systemy informatyczne pod względem ich funkcjonalności	Systemy informatyczne
	5) opisuje działanie portali społecznościowych	Systemy informatyczne
	6) określa zasady bezpiecznego korzystania z portali społecznościowych	Systemy informatyczne





Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	7) podaje przykłady zastosowań systemów informatycznych w działalności biznesowej, w tym e-commerce, e-sklep, e-faktura, systemy rezerwacyjne	Systemy informatyczne
4) stosuje zalecenia dotyczące ułatwień dostępności serwisów internetowych dla osób niepełnosprawnych (ew)	1) wymienia dostępne udogodnienia dla osób z niepełnosprawnościami	Systemy informatyczne
	2) wymienia wymagania dotyczące poziomu dostępności według wytycznych WCAG 2.0	Systemy informatyczne
5) posługuje się terminologią dotyczącą sieci komputerowych (ek)	1) wymienia topologie sieci	Sieci komputerowe
	2) identyfikuje cechy modelu TCP/IP i protokołów komunikacji sieciowej	Sieci komputerowe
	3) opisuje sieć bezprzewodową oraz sieć przewodową	Sieci komputerowe
	4) oblicza przepustowość sieci	Systemy liczbowe Sieci komputerowe
	5) definiuje pojęcia pobieranie i wysyłanie danych	Sieci komputerowe
	6) opisuje zasady działania sieci synchronicznej i asynchronicznej	Sieci komputerowe
	7) wykazuje różnice w działaniu sieci synchronicznej i asynchronicznej	Sieci komputerowe
	8) wymienia i stosuje zasady bezpieczeństwa przy korzystaniu z sieci	Sieci komputerowe
	9) używa komunikatorów tekstowych, audio-video oraz tablic interaktywnych	Sieci komputerowe
	10) stosuje zasadę netykiety	Sieci komputerowe
6) stosuje pozycyjne systemy liczbowe (ek)	1) przekształca liczby zapisane w różnych pozycyjnych systemach liczbowych: dwójkowym, ósemkowym, szesnastkowym, dziesiętnym	Systemy liczbowe
	2) zapisuje liczby w kodzie uzupełnieniowym do dwóch	Systemy liczbowe
	3) wykonuje podstawowe działania logiczne i arytmetyczne na liczbach binarnych	Systemy liczbowe



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	4) wykorzystuje dostępne narzędzia informatyczne do wykonywania działań na liczbach zapisanych w różnych pozycyjnych systemach liczbowych (np. kalkulatory HEX, DEC, BIN) liczbowych	Systemy liczbowe
7) stosuje zasady cyberbezpieczeństwa (ek)	1) rozróżnia rodzaje szkodliwego oprogramowania	Bezpieczeństwo w informatyce
	2) rozróżnia rodzaje ataków hakerskich	Bezpieczeństwo w informatyce
	3) wymienia środki zabezpieczeń przed złośliwym oprogramowaniem oraz atakami hakerskimi	Bezpieczeństwo w informatyce
	4) wymienia zagrożenia dla sfery psychicznej (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni	Bezpieczeństwo w informatyce
	5) opisuje zagrożenia dla sfery psychicznej człowieka (emocjonalnej), fizycznej, społecznej, poznawczej człowieka, wynikające z przebywania w cyberprzestrzeni i sposoby przeciwdziałania tym zagrożeniom	Bezpieczeństwo w informatyce
	6) przestrzega zasad bezpiecznego przechowywania danych	Bezpieczeństwo w informatyce
	7) przestrzega zasad bezpieczeństwa swojego cyfrowego wizerunku i tożsamości	Bezpieczeństwo w informatyce
	8) przestrzega zasad prywatności w cyfrowym świecie	Bezpieczeństwo w informatyce
	9) wymienia i omawia podstawowe pojęcia związane z ochroną danych osobowych, ochroną informacji, prawami autorskimi i własnością intelektualną oraz wyjaśnia potrzebę ich ochrony	Bezpieczeństwo w informatyce
	10) stosuje zasady dokonywania bezpiecznych transakcji w internecie	Bezpieczeństwo w informatyce
8) rozpoznaje właściwe normy i procedury oceny zgodności podczas realizacji zadań zawodowych (ew)	1) wymienia cele normalizacji krajowej	Normy w informatyce
	2) wyjaśnia, czym jest norma i wymienia cechy normy	Normy w informatyce
	3) rozróżnia oznaczenie normy międzynarodowej, europejskiej i krajowej	Normy w informatyce
	4) korzysta ze źródeł informacji dotyczących norm i procedur oceny zgodności	Normy w informatyce
INF.04.3. Projektowanie oprogramowania		



<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
1) posługuje się prostymi typami danych (ek)	1) rozróżnia typy numeryczne stałoprzecinkowe i zmiennoprzecinkowe	Wstęp do programowania, typy danych
	2) rozpoznaje typ logiczny	Wstęp do programowania, typy danych
	3) rozróżnia typy znakowe i łańcuchowe	Wstęp do programowania, typy danych
	4) posługuje się typami liczbowymi stałoprzecinkowe i zmiennoprzecinkowymi, typem logicznym, typem znakowym i łańcuchowym	Wstęp do programowania, typy danych
2) posługuje się złożonymi typami danych (ek)	1) rozróżnia rodzaje złożonych typów danych	Wstęp do programowania, typy danych
	2) posługuje się tablicami jednowymiarowymi i dwuwymiarowymi	Wstęp do programowania, typy danych
	3) posługuje się tablicami dynamicznymi, asocjacyjnymi	Wstęp do programowania, typy danych
	4) posługuje się typem rekordowym, np. struktura, unia	Wstęp do programowania, typy danych
	5) posługuje się typem plikowym	Wstęp do programowania, typy danych
	6) posługuje się typem wskaźnikowym	Wstęp do programowania, typy danych
	7) charakteryzuje cechy kolekcji, w tym znaczenie iteratora	Wstęp do programowania, typy danych
	8) posługuje się kolekcjami, np. listami, kolejkami, stosami, wektorami	Wstęp do programowania, typy danych
	9) projektuje zestawy danych dla problemu programistycznego	Wstęp do programowania, typy danych Planowanie prac w projekcie
3) stosuje metody rozwiązywania problemów za pomocą algorytmów (ek)	1) projektuje algorytmy za pomocą różnych metod: schematów blokowych, listy kroków, drzew decyzyjnych, pseudokodu	Rozwiązywanie problemów przy użyciu algorytmów
	2) charakteryzuje algorytmy iteracyjne, tekstowe i szyfrowania, tablicowe	Rozwiązywanie problemów przy użyciu algorytmów
	3) charakteryzuje algorytmy rekurencyjne	Rozwiązywanie problemów przy użyciu algorytmów
	4) charakteryzuje problemy i metody ich rozwiązywania, np. algorytmy heurystyczne, problem komiwojażera	Rozwiązywanie problemów przy użyciu algorytmów
	5) określa złożoność obliczeniową algorytmów	Rozwiązywanie problemów przy użyciu algorytmów
	1) charakteryzuje typy sortowania i ich złożoność obliczeniową	Rozwiązywanie problemów przy użyciu algorytmów



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	2) stosuje różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawianie, szybkie, metodą dziel i zwycięża	Rozwiązywanie problemów przy użyciu algorytmów
	3) stosuje algorytmy wyszukiwania dla tablic, list, kolejek, stosów	Rozwiązywanie problemów przy użyciu algorytmów
5) dobiera narzędzia i metodologie do planowania i zarządzania projektem (ew)	1) określa funkcje narzędzi do zarządzania projektem	Narzędzia wspomagające projektowanie aplikacji
	2) stosuje diagramy do zarządzania etapami projektu, zadaniami i czasem, np. diagram Gantta	Narzędzia wspomagające projektowanie aplikacji Planowanie prac w projekcie
	3) korzysta z programów wspierających zarządzanie projektami, np. Jira, Trello	Narzędzia wspomagające projektowanie aplikacji Planowanie prac w projekcie
	4) korzysta z systemu kontroli wersji, np. Git	Narzędzia wspomagające projektowanie aplikacji Planowanie prac w projekcie
6) projektuje aplikację (ek)	1) analizuje wymagania klienta i tworzy zgodny z nimi projekt	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	2) tłumaczy wymagania klienta na specyfikację techniczną dla zespołu programistów	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	3) identyfikuje elementy interfejsu użytkownika, np. okna, dialogi, kontrolki, formularze, paski narzędziowe, widgety	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	4) projektuje interfejs użytkownika i wygląd aplikacji	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	5) dostosowuje interfejs do różnych platform	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	6) projektuje aplikacje w różnych paradygmatach programowania: strukturalnym, obiektowym	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	7) projektuje aplikację opartą na architekturze klient-serwer	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	8) projektuje struktury danych dla aplikacji	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	9) projektuje funkcjonalność aplikacji	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	10) planuje system zabezpieczeń aplikacji	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie Prawo autorskie i licencje
7) planuje przedsięwzięcie programistyczne (ew)	1) określa cel projektu	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	2) określa fazy realizacji projektu	Planowanie prac w projekcie
	3) charakteryzuje cykl życia projektu informatycznego i jego poszczególne etapy	Planowanie prac w projekcie
	4) określa zasoby ludzkie oraz ramy czasowe wykonania projektu	Planowanie prac w projekcie
	5) planuje etapy tworzenia aplikacji	Planowanie prac w projekcie
	6) korzysta z metodologii zarządzania projektem: model kaskadowy (waterfall), model przyrostowy, model prototypowy, metodyki zwinne (Agile oraz przynajmniej jedną z Scrum, Lean, Kanban)	Planowanie prac w projekcie
	7) dobiera optymalną metodologię zarządzania projektem	Planowanie prac w projekcie
	8) organizuje prace projektowe	Planowanie prac w projekcie



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	9) stosuje harmonogram czynności w celu efektywnego osiągnięcia celów	Planowanie prac w projekcie
8) stosuje wzorce projektowe (ew)	1) dobiera wzorzec projektowy do zadania programistycznego	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
	2) stosuje wzorce projektowe w programowaniu obiektowym, np. Metoda szablonowa (Template method), Fasada (Facade), Kompozyt (Composite)	Projekt aplikacji na podstawie wymagań klienta Planowanie prac w projekcie
9) stosuje zagadnienia prawa autorskiego w dziedzinie programowania (ep)	1) rozróżnia autorskie prawa osobiste i majątkowe	Prawo autorskie i licencje
	2) określa czas trwania praw autorskich	Prawo autorskie i licencje
	3) określa konsekwencje naruszenia prawa autorskiego	Prawo autorskie i licencje
	4) charakteryzuje elementy własności intelektualnej (dobra niematerialne, własności przemysłowe)	Prawo autorskie i licencje
	5) rozróżnia typy licencji oprogramowania	Prawo autorskie i licencje
INF.04.4. Programowanie obiektowe		
1) wykorzystuje środowisko programistyczne dla obiektowych aplikacji konsolowych (ew)	1) rozróżnia kompilatory i interpretery	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	2) charakteryzuje zadania kompilatora, interpretera, debuggera	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	3) analizuje błędy w kodzie za pomocą debuggera	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Niezwodność oprogramowania Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia

<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
	4) charakteryzuje etapy kompilacji i interpretacji kodu	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek
	5) charakteryzuje pojęcie biblioteki	Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego Programowanie obiektowe
	6) kompiluje i uruchamia programy	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Niezawodność oprogramowania Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
2) przestrzega zasad programowania (ew)	1) dzieli program na funkcje (metody)	Środowiska programistyczne, programowanie strukturalne
	2) stosuje rekurencję	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
	3) implementuje algorytmy w programie	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
3) korzysta z typów danych (ek)	1) stosuje proste i złożone typy danych	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
	2) deklaruje własne typy danych	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
	3) deklaruje zmienne różnych typów danych	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	4) wykonuje operacje na zmiennych: wejścia i wyjścia, arytmetyczne, logiczne	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
	5) stosuje typy złożone i operacje na nich	Środowiska programistyczne, programowanie strukturalne Wstęp do programowania obiektowego
4) stosuje wyrażenia, instrukcje i biblioteki (ek)	1) stosuje operatory arytmetyczne, przypisania, porównania, logiczne, operatory do obsługi łańcuchów, bitowe	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	2) wykorzystuje priorytety operatorów do właściwego budowania wyrażeń	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	3) stosuje instrukcję warunkową i wyboru	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	4) stosuje instrukcje pętli	Środowiska programistyczne, programowanie strukturalne Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
	5) korzysta z wybranych bibliotek języka C++, C#, Python lub innego języka programowania: biblioteka standardowa, biblioteka z funkcjami matematycznymi, biblioteka z podstawowymi algorytmami	Programowanie z wykorzystaniem bibliotek Wstęp do programowania obiektowego
5) stosuje zasady programowania obiektowego (ek)	1) stosuje obiektowe podejście do rozwiązywania problemów	Wstęp do programowania obiektowego
	2) charakteryzuje pojęcia klasa, obiekt, metoda, pole, dziedziczenie, hermetyzacja, polimorfizm	Wstęp do programowania obiektowego
	3) dzieli zagadnienie na klasy	Wstęp do programowania obiektowego
	4) powołuje obiekty	Wstęp do programowania obiektowego





Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	5) planuje aplikację z zastosowaniem hermetyzacji, dziedziczenia i polimorfizmu	
6) definiuje klasy (ek)	1) definiuje pola klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	2) określa zakres widoczności pól klasy i definiuje kwalifikatory dostępu	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	3) definiuje metody klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	4) definiuje konstruktory, w tym konstruktor kopiujący, i destruktor klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	5) definiuje instrukcje inicjujące konstruktora	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	6) określa zakres widoczności metod klasy i definiuje kwalifikatory dostępu	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	7) implementuje funkcjonalność klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	8) deklaruje obiekty i odwołuje się obiektem do składowych klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	9) definiuje składniki statyczne klasy	Wstęp do programowania obiektowego Programowanie obiektowe Dziedziczenie i hierarchia
	10) rozróżnia klasy dziedziczone i zaprzyjaźnione	Dziedziczenie i hierarchia
	11) tworzy funkcje zaprzyjaźnione z klasą	Dziedziczenie i hierarchia



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	12) stosuje składnik statyczny klasy i metody do ich obsługi	Dziedziczenie i hierarchia
7) definiuje klasy pochodne (ew)	1) buduje hierarchię dziedziczenia klas w programie	Dziedziczenie i hierarchia
	2) wydziela metody i pola do odpowiednich klas w hierarchii dziedziczenia	Dziedziczenie i hierarchia
	3) definiuje klasy bazowe i pochodne	Dziedziczenie i hierarchia
	4) stosuje metody wirtualne, definiuje klasy abstrakcyjne	Dziedziczenie i hierarchia
8) programuje szablony (wzorce) klas (ew)	1) definiuje szablony klas dla obsługi prostych typów liczbowych	Dziedziczenie i hierarchia
9) programuje obsługę wyjątków (ew)	1) stosuje szkielet obsługi wyjątków z instrukcjami try i catch	Niezawodność oprogramowania
	2) stosuje instrukcję throw	Niezawodność oprogramowania
	3) opracowuje listę możliwych błędów wykonania aplikacji	Niezawodność oprogramowania
	4) definiuje obsługę dla błędów wykonania aplikacji	Niezawodność oprogramowania
INF.04.5. Programowanie aplikacji desktopowych		
1) wykorzystuje środowisko programistyczne dla aplikacji desktopowych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio)	Środowiska i tworzenie aplikacji desktopowych
	2) stosuje środowiska IDE, RAD	Środowiska i tworzenie aplikacji desktopowych
	3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji desktopowych	Środowiska i tworzenie aplikacji desktopowych Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
2) wykorzystuje frameworki do programowania aplikacji desktopowych (ek)	1) charakteryzuje pojęcie framework	Aplikacje okienkowe bazujące na frameworkach
	2) stosuje frameworki typowe dla aplikacji desktopowych, np. WPF, Qt	Aplikacje okienkowe bazujące na frameworkach
3) programuje desktopowe aplikacje okienkowe (ek)	1) rozróżnia elementy interfejsu użytkownika (okno, dialog modalny i niemodalny, kontrolki)	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
	2) stosuje języki programowania odpowiednie dla aplikacji desktopowych, np. C++, C#, Visual Basic, Java, Python	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	3) stosuje język do projektowania interfejsu użytkownika np. XAML	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
	4) programuje okna aplikacji	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
	5) programuje system menu aplikacji	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
	6) programuje okna dialogowe aplikacji	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
	7) programuje obsługę zdarzeń myszy i klawiatury	Aplikacje okienkowe bazujące na OS API Aplikacje okienkowe bazujące na frameworkach
INF.04.6. Programowanie aplikacji mobilnych		
1) wykorzystuje środowisko programistyczne dla aplikacji mobilnych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Android Studio, XCode)	Środowiska do tworzenia aplikacji mobilnych Projekt, aplikacja przygotowana do dystrybucji w sklepie
	2) stosuje środowiska IDE, RAD	Środowiska do tworzenia aplikacji mobilnych Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji mobilnych	Środowiska do tworzenia aplikacji mobilnych Projekt, aplikacja przygotowana do dystrybucji w sklepie



<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
2) programuje aplikacje mobilne (ek)	1) programuje aplikacje w jednym z systemów mobilnych: iOS lub Android	Środowiska do tworzenia aplikacji mobilnych Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	2) stosuje języki programowania dedykowane dla aplikacji mobilnych na systemach iOS lub Android: Objective-C lub Swift lub Java lub C#	Środowiska do tworzenia aplikacji mobilnych Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	3) programuje przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych	Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	4) identyfikuje elementy UI aplikacji mobilnej (przyciski, nawigacja, okna dialogowe, listy, formularze, paski narzędziowe, grafika, animacje, dźwięk)	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie



<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
	5) programuje interfejs użytkownika za pomocą języka XAML	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	6) programuje aplikacje mobilne przy wykorzystaniu elementów UI	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	7) programuje proste aplikacje mobilne typu zegar, powiadamianie, kalendarz, formularz, lokalizacja (system nawigacji satelitarnej)	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	8) pobiera i wysyła dane z lub do internetu dla aplikacji mobilnej	Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	9) programuje aplikację mobilną korzystającą z bazy danych	Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	10) tworzy aplikacje dostosowane do danej platformy mobilnej (np. system Android smartfon, system Android Tablet, IPAD iPhone)	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	11) uruchamia aplikacje mobilne	Proste aplikacje mobilne Aplikacje z wykorzystaniem lokalnych nośników i baz danych Aplikacje z wykorzystaniem połączenia internetowego Projekt, aplikacja przygotowana do dystrybucji w sklepie
	12) przygotowuje aplikacje do publikacji w sklepie	Projekt, aplikacja przygotowana do dystrybucji w sklepie
INF.04.7. Programowanie aplikacji zaawansowanych webowych		
1) wykorzystuje środowisko programistyczne dla aplikacji zaawansowanych webowych (ew)	1) dobiera środowisko programistyczne do określonych zadań i języka programowania (np. Visual Studio, Eclipse Jet Brains)	Środowiska do tworzenia aplikacji webowych
	2) stosuje środowiska IDE, RAD	Środowiska do tworzenia aplikacji webowych Aplikacje działające po stronie klienta Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
	3) rozpoznaje narzędzia wykorzystywane w procesie tworzenia aplikacji webowych	Środowiska do tworzenia aplikacji webowych
2) wykorzystuje frameworki do programowania aplikacji webowych (ek)	1) stosuje frameworki typowe dla aplikacji webowych: ASP.NET Core, Django, Angular, React.js, Node.js	Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	2) wykorzystuje bibliotekę jQuery lub jej odpowiednik	Aplikacje działające po stronie klienta Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
3) programuje zaawansowane aplikacje webowych (ek)	1) stosuje języki programowania odpowiednie dla aplikacji webowych, np. PHP, C#, Python, JavaScript	Środowiska do tworzenia aplikacji webowych Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
	2) stosuje mechanizm sesji użytkownika, ciasteczek	Środowiska do tworzenia aplikacji webowych Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
	3) programuje dynamiczne formularze, systemy logowania i kontroli dostępu do określonych elementów witryny	Aplikacje działające po stronie klienta Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
	4) programuje aplikacje korzystające z bazy danych	Aplikacje działające po stronie serwera Aplikacje wykorzystujące AJAX Projekt, aplikacja webowa
	5) programuje wybrane elementy funkcjonalności e-sklepu, portalu społecznościowego, serwisu ogłoszeniowego, serwisu rezerwacyjnego	Aplikacje działające po stronie klienta Projekt, aplikacja webowa
INF.04.8. Testowanie i dokumentowanie aplikacji		
1) przeprowadza walidację kodu programu (ek)	1) dobiera narzędzia i środowisko do testowania programów	Dokumentacja kodu programu Usuwanie błędów i optymalizacja
	2) wykonuje testy tworzonych programów	Usuwanie błędów i optymalizacja Testy funkcjonalności i interfejsów Testy aplikacji
	3) identyfikuje błędy w tworzonych programach	Usuwanie błędów i optymalizacja



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	4) poprawia błędy w tworzonych programach, skryptach	Usuwanie błędów i optymalizacja Testy funkcjonalności i interfejsów Testy aplikacji
	5) optymalizuje kod źródłowy	Usuwanie błędów i optymalizacja
2) dokumentuje tworzoną aplikację (ew)	1) stosuje komentarze w kodzie źródłowym programu	Dokumentacja kodu programu Usuwanie błędów i optymalizacja
	2) pisze dokumentację kodu	Dokumentacja kodu programu Dokumentacja wdrożeniowa i instrukcje obsługi
	3) pisze dokumenty pomocy programu	Dokumentacja wdrożeniowa i instrukcje obsługi
	4) pisze instrukcję użytkownika aplikacji	Dokumentacja wdrożeniowa i instrukcje obsługi
	5) pisze instrukcję użytkownika programu	Dokumentacja wdrożeniowa i instrukcje obsługi
	6) pisze dokumentację wdrożenia projektu	Dokumentacja wdrożeniowa i instrukcje obsługi
	7) przygotowuje dokumentację z wykonanych testów	Dokumentacja kodu programu Planowanie testów Testy aplikacji
3) przeprowadza testy (ek)	1) charakteryzuje metodyki testowania	Testy funkcjonalności i interfejsów
	2) przygotowuje testy funkcjonalne i нефункционалне	Testy funkcjonalności i interfejsów Testy aplikacji
	3) przeprowadza testy funkcjonalne	Testy funkcjonalności i interfejsów Testy aplikacji
	4) przeprowadza testy w kolejnych fazach projektu informatycznego	Usuwanie błędów i optymalizacja Testy funkcjonalności i interfejsów Testy aplikacji
	5) przeprowadza testy нефункционалне: użyteczności, wydajnościowe, obciążeniowe, zgodności, bezpieczeństwa	Testy funkcjonalności i interfejsów Testy aplikacji





Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
	6) korzysta z systemów raportowania błędów, np. BugZilla, JIRA	Testy funkcjonalności i interfejsów Testy aplikacji
	7) przeprowadza testy interfejsu	Testy aplikacji
	8) testuje prototyp projektu interfejsu	Testy funkcjonalności i interfejsów Testy aplikacji
	9) przygotowuje środowiska testowe	Testy funkcjonalności i interfejsów Testy aplikacji
	10) planuje scenariusze testowania aplikacji	Planowanie testów Testy aplikacji
	11) pisze scenariusze testowania aplikacji	Planowanie testów Testy aplikacji
	12) raportuje błędy po przeprowadzonych testach	Testy funkcjonalności i interfejsów Testy aplikacji
	13) automatyzuje proces testowania	Testy funkcjonalności i interfejsów Testy aplikacji
	14) testuje aplikację	Testy funkcjonalności i interfejsów Testy aplikacji
	15) weryfikuje zgodność aplikacji z oczekiwaniami klienta	Testy funkcjonalności i interfejsów Planowanie testów Testy aplikacji
INF.04.9. Język obcy zawodowy		
	1) rozpoznaje oraz stosuje środki językowe umożliwiające realizację czynności zawodowych w zakresie:	Słownictwo w branży teleinformatycznej Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice



<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
<p>1) 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 (ek)</p>	a) czynności wykonywanych na stanowisku pracy, w tym związanych z zapewnieniem bezpieczeństwa i higieny pracy	<p>Słownictwo w branży teleinformatycznej</p> <p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p>
	b) narzędzi, maszyn, urządzeń i materiałów koniecznych do realizacji czynności zawodowych	<p>Słownictwo w branży teleinformatycznej</p> <p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p>
	c) procesów i procedur związanych z realizacją zadań zawodowych	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p>
	d) formularzy, specyfikacji oraz innych dokumentów związanych z wykonywaniem zadań zawodowych	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p>
	e) świadczonych usług, w tym obsługi klienta	<p>Obsługa klienta w języku obcym</p> <p>Poszukiwanie pracy</p>
	1) określa główną myśl wypowiedzi lub tekstu lub fragmentu wypowiedzi lub tekstu	<p>Słownictwo w branży teleinformatycznej</p> <p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p>

<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
<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 pisemne dotyczące czynności zawodowych (np. napisy, broszury, instrukcje obsługi, przewodniki, dokumentację zawodową) (ew)</p>	2) znajduje w wypowiedzi lub tekście określone informacje	<p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p> <p>Oznaczenia i symbole w elektronice</p>
	3) rozpoznaje związki między poszczególnymi częściami tekstu	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</p>
	4) układa informacje w określonym porządku	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Pozyskiwanie informacji z źródeł obcojęzycznych</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 – według wzoru) (ew)</p>	1) opisuje przedmioty, działania i zjawiska związane z czynnościami zawodowymi	<p>Słownictwo w branży teleinformatycznej</p> <p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p>
	2) przedstawia sposób postępowania w różnych sytuacjach zawodowych (np. udziela instrukcji, wskazówek, określa zasady)	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p>
	3) wyraża i uzasadnia swoje stanowisko	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p>
	4) stosuje zasady konstruowania tekstów o różnym charakterze	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p>
	5) stosuje formalny lub nieformalny styl wypowiedzi adekwatnie do sytuacji	<p>Obsługa klienta w języku obcym</p> <p>Komunikacja za pomocą słowa pisanego</p> <p>Poszukiwanie pracy</p>

<b>Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie</b>		<b>Zawartość opracowanego programu zajęć (temat zajęć)</b>
<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, 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. wiadomość, formularz, e-mail, dokument związany z wykonywanym zawodem) w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)</p>	1) rozpoczyna, prowadzi i kończy rozmowę	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	2) uzyskuje i przekazuje informacje i wyjaśnienia	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	3) wyraża swoje opinie i uzasadnia je, pyta o opinie, zgadza się lub nie zgadza z opiniami innych osób	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	4) prowadzi proste negocjacje związane z czynnościami zawodowymi	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	5) stosuje zwroty i formy grzecznościowe	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	6) dostosowuje styl wypowiedzi do sytuacji	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
<p>5) zmienia formę przekazu ustnego lub pisemnego w języku obcym nowożytnym w typowych sytuacjach związanych z wykonywaniem czynności zawodowych (ep)</p>	1) przekazuje w języku obcym nowożytnym informacje zawarte w materiałach wizualnych (np. wykresach, symbolach, piktogramach, schematach) oraz audiowizualnych (np. filmach instruktażowych)	Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	2) przekazuje w języku polskim informacje sformułowane w języku obcym nowożytnym	Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	3) przekazuje w języku obcym nowożytnym informacje sformułowane w języku polskim lub w tym języku obcym nowożytnym	Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	4) przedstawia publicznie w języku obcym nowożytnym wcześniej opracowany materiał, np. prezentację	Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice



Efekty kształcenia oraz kryteria weryfikacji określone w podstawie programowej kształcenia w zawodzie		Zawartość opracowanego programu zajęć (temat zajęć)
<p>6) wykorzystuje strategie służące doskonaleniu własnych umiejętności językowych oraz podnoszące świadomość językową:</p> <p>a) wykorzystuje techniki samodzielnej pracy nad językiem</p> <p>b) współdziała w grupie (ep)</p>	1) korzysta ze słownika dwujęzycznego i jednojęzycznego	Pozyskiwanie informacji z źródeł obcojęzycznych
	2) współdziała z innymi osobami, realizując zadania językowe	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy
	3) korzysta z tekstów w języku obcym nowożytnym, również za pomocą technologii informacyjno-komunikacyjnych	Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	4) identyfikuje słowa klucze i internacjonalizmy	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	5) wykorzystuje kontekst (tam, gdzie to możliwe), aby w przybliżeniu określić znaczenie słowa	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Pozyskiwanie informacji z źródeł obcojęzycznych Oznaczenia i symbole w elektronice
	6) upraszcza (jeżeli to konieczne) wypowiedź, zastępuje nieznane słowa innymi, wykorzystuje opis, środki niewerbalne	Obsługa klienta w języku obcym Komunikacja za pomocą słowa pisanego Poszukiwanie pracy