

Technikum w Zespole Szkół
im. Armii Krajowej
Obwodu "Głuszczyk" - Grójec
w Grójcu

Wymagania edukacyjne
na poszczególne oceny szkolne z przedmiotu:
Strony internetowe

I. Podstawa prawna

1. Ustawa z dnia 7 września 1991 r. o systemie oświaty (tekst jednolity: Dz.U. z 2024 r., poz. 750) - Rozdział 3a
2. Ustawa z dnia 14 grudnia 2016 r. Prawo oświatowe (Dz.U.2023 poz.900)
3. Rozporządzenie Ministra Edukacji Narodowej z dnia 22 lutego 2019 r. w sprawie oceniania, klasyfikowania i promowania uczniów i słuchaczy w szkołach publicznych (tekst jedn.: Dz.U. z 2023 r., poz. 2572)
4. Statut Technikum w Zespole Szkół im. Armii Krajowej Obwodu "Głuszec" - Grójec w Grójcu.
5. Program nauczania dla zawodu Technik Informatyków 351203
6. Program nauczania dla zawodu Technik Programista 351406

Efekty kształcenia z podstawy programowej	Po zrealizowaniu zajęć uczeń potrafi
Uczeń:	
Tworzenie stron internetowych	wymienić polecenia (znaczniki) języka HTML (standard HTML5), rozróżnić polecenia języka HTML i ich atrybuty, scharakteryzować znaczniki języka stron internetowych i ich atrybuty, zastosować polecenia języka HTML do tworzenia elementów strony (akapity, nagłówki, obrazy, hiperłącza wewnętrzne i zewnętrzne, mapy odsyłaczy, listy, tabele, formularze, pływające ramki), zastosować elementy strukturalne do tworzenia układu strony
Ocena niedostateczna	<ul style="list-style-type: none">• Uczeń nie potrafi stworzyć najprostszej strony internetowej w HTML.• Uczeń wykazuje brak znajomości podstawowych elementów HTML, takich jak tagi <html>, <head>, <body>, <p>, <h1>, <a>, .• Uczeń nie rozumie struktury dokumentu HTML ani sposobu tworzenia hiperłączy, nagłówków czy list.• Uczeń nie podejmuje prób ani nie jest w stanie poprawić nawet najprostszych błędów w kodzie HTML.
Ocena dopuszczająca	<ul style="list-style-type: none">• Uczeń, z pomocą nauczyciela, potrafi stworzyć prostą stronę internetową w HTML z podstawową strukturą (tagi <html>, <head>, <body>).• Uczeń potrafi używać podstawowych tagów HTML do tworzenia tekstu, prostych list i nagłówków, ale potrzebuje wsparcia w bardziej złożonych elementach (np. obrazki,

	<p>hiperłącza).</p> <ul style="list-style-type: none"> • Uczeń rozumie strukturę dokumentu HTML, ale wykonuje zadania w sposób niepewny i często popełnia błędy. • Uczeń wymaga pomocy w naprawianiu błędów w kodzie HTML.
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie stworzyć prostą stronę internetową w HTML, zawierającą tekst, nagłówki, listy oraz podstawowe hiperłącza. • Uczeń potrafi używać podstawowych tagów (np. <p>, <h1>, <a>,) do wstawiania treści, ale może mieć trudności z bardziej zaawansowanymi funkcjami (np. tabelami, formularzami). • Uczeń zna podstawy formatowania tekstu w HTML oraz potrafi umieścić obrazy i tworzyć proste linki, ale może mieć trudności z estetyką i układem strony. • Uczeń rozumie, że HTML służy do tworzenia struktury strony, ale nie opanował jeszcze koncepcji semantycznych tagów (np. <article>, <section>).
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie stworzyć i poprawnie sformatować stronę internetową w HTML, używając zarówno prostych, jak i bardziej zaawansowanych tagów (np. <table>, <form>, <input>). • Uczeń potrafi rozszerzyć funkcjonalność strony, dodając hiperłącza, obrazy, tabele, formularze oraz stosując style inline (np. style w atrybucie style). • Uczeń rozumie podstawowe zasady semantyki HTML (np. używanie odpowiednich tagów dla różnych sekcji strony) oraz zachowuje poprawność kodu (np. poprawne otwieranie i zamykanie tagów). • Uczeń potrafi naprawiać błędy w kodzie HTML i sprawdzić poprawność swojej pracy, choć może wymagać dodatkowego wsparcia w zakresie walidacji kodu.
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć złożone strony internetowe w HTML, korzystając z pełnej gamy dostępnych tagów (np. strukturalnych, semantycznych, multimedialnych). • Uczeń potrafi wdrożyć zaawansowane elementy, takie jak formularze z różnymi typami pól (<input>, <textarea>), osadzać wideo i dźwięk oraz tworzyć dobrze sformatowane tabele. • Uczeń stosuje najlepsze praktyki kodowania, np. poprawnie używa tagów semantycznych, dba o strukturę i czytelność kodu oraz przestrzega standardów dostępności (np. odpowiednie etykietowanie elementów formularzy). • Uczeń potrafi wprowadzać elementy interakcji z użytkownikiem (np. formularze) i łączyć je z arkuszami stylów CSS lub skryptami

	JavaScript.
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi stworzyć profesjonalną stronę internetową w HTML, dbając o optymalizację kodu, jego poprawność, dostępność i zgodność ze standardami W3C. • Uczeń potrafi tworzyć rozbudowane strony, zawierające multimedia, formularze, tabele, osadzone treści i elementy interaktywne, oraz integrować je z innymi technologiami webowymi (np. CSS, JavaScript). • Uczeń rozumie i stosuje zasady dostępności i SEO (Search Engine Optimization) podczas tworzenia strony, np. dba o odpowiednie opisy obrazków (atrybuty alt), używa poprawnych nagłówków oraz strukturę przyjazną wyszukiwarkom. • Uczeń potrafi tworzyć czysty, czytelny kod, dbać o jego strukturę i optymalizację oraz korzystać z narzędzi do walidacji kodu HTML.
Kaskadowe arkusze stylów (CSS)	<p>wymienić podstawowe właściwości CSS elementów strony (np.: czcionki, tekstu, tła, obramowania, odsyłaczy, list, kursora, marginesów wewnętrznych i zewnętrznych, rozmiarów elementów),</p> <p>skonstruować regułę CSS,</p> <p>rozdzielić selektory CSS (elementów, atrybutów, klas, identyfikatorów, pseudoklas, pseudoelementów, kontekstowe)</p> <p>scharakteryzować rodzaje stylów,</p> <p>zdefiniować strukturę arkusza stylów,</p> <p>dołączyć styl do strony internetowej,</p> <p>objaśnić kaskadowość stylów,</p> <p>zastosować reguły CSS do formatowania strony internetowej</p>
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie potrafi wyjaśnić, czym są kaskadowe arkusze stylów (CSS). • Uczeń nie umie zidentyfikować podstawowej składni CSS. • Uczeń nie rozróżnia elementów HTML od reguł CSS. • Uczeń nie potrafi przypisać prostych stylów (np. kolor tła, rozmiar czcionki) do elementów HTML.
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi wymienić podstawowe właściwości CSS (np. color, font-size, background-color). • Uczeń potrafi wskazać, że CSS służy do stylizowania stron internetowych. • Uczeń rozpoznaje i rozumie podstawową składnię reguły CSS (selekcja elementu, właściwość, wartość). • Uczeń umie wskazać miejsce umieszczania CSS w kodzie HTML (wewnątrz znacznika <style>, wewnątrz pliku zewnętrznego).

Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi opisać, jak CSS wpływa na wygląd stron internetowych. • Uczeń umie zinterpretować prostą regułę CSS i wyjaśnić jej działanie. • Uczeń potrafi użyć prostych selektorów (np. selektor typu, klasy, identyfikatora) do przypisywania stylów. • Uczeń jest w stanie opisać różnice między stylami wewnętrznymi, zewnętrznymi i inline (wewnątrz znaczników HTML). • Uczeń umie zastosować CSS do zmiany koloru, tła, wielkości czcionki i marginesów.
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie tworzyć pliki CSS i podłączać je do dokumentu HTML. • Uczeń poprawnie stosuje różne rodzaje selektorów (np. potomków, atrybutów, grup). • Zna i stosuje zasady dziedziczenia oraz kaskadowości w CSS (priorytet reguł). • Umie stosować CSS do pozycjonowania elementów na stronie (np. float, position, display). • Potrafi tworzyć proste układy stron z użyciem stylów, korzystając z modelu pudełkowego (box model). • Zna różnice między jednostkami bezwzględными a względnymi (np. px, %, em).
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi analizować złożone arkusze stylów i diagnozować problemy ze stylowaniem. • Uczeń umie optymalizować i upraszczać kod CSS (np. przez łączenie selektorów, wykorzystywanie zmiennych CSS). • Uczeń stosuje zaawansowane selektory (np. nth-child, pseudo-klasy, pseudo-elementy). • Uczeń umie pracować z responsywnym designem, wykorzystując media queries do stylizowania stron na różne urządzenia. • Uczeń potrafi stosować zaawansowane właściwości CSS, takie jak cienie (box-shadow), gradienty (background-gradient), przejścia (transition), transformacje (transform).
Ocena celująca	<ul style="list-style-type: none"> • Uczeń tworzy skomplikowane, złożone układy stron internetowych, korzystając z nowoczesnych technik CSS (np. Flexbox, CSS Grid). • Uczeń w pełni kontroluje responsywność stron z wykorzystaniem media queries oraz nowoczesnych metod projektowania (mobile-first). • Uczeń potrafi tworzyć animacje CSS i stosować je w praktyce. • Uczeń samodzielnie rozwija projekty front-endowe, wykorzystując preprocesory CSS (np. SASS, LESS) oraz narzędzia optymalizacyjne. • Uczeń umie oceniać i rozwiązywać problemy z wydajnością styli oraz analizować wpływ CSS na User Experience. • Uczeń potrafi krytycznie oceniać i wdrażać nowe techniki i narzędzia związane z CSS w

	bieżących projektach.
Systemy zarządzania treścią (CMS)	scharakteryzować systemy zarządzania treścią (CMS), dobrać system zarządzania treścią do określonego zastosowania
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie potrafi wyjaśnić, czym jest system zarządzania treścią (CMS). • Uczeń nie rozróżnia popularnych systemów CMS. • Uczeń nie potrafi opisać głównych funkcji CMS-ów. • Uczeń nie rozumie roli CMS-ów w zarządzaniu treścią stron internetowych.
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi wymienić przykłady popularnych systemów CMS (np. WordPress, Joomla, Drupal). • Uczeń potrafi zdefiniować, co to jest CMS (system zarządzania treścią). • Uczeń rozpoznaje podstawowe elementy CMS, takie jak edytor treści, motywy, wtyczki i panele administratora. • Uczeń wie, że CMS jest używany do tworzenia i zarządzania treścią na stronach internetowych.
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi opisać, w jaki sposób CMS pomaga w zarządzaniu stronami internetowymi bez potrzeby zaawansowanej znajomości programowania. • Uczeń rozumie różnice między statyczną stroną internetową a stroną opartą na CMS. • Uczeń potrafi zainstalować i skonfigurować CMS (np. WordPress) na lokalnym serwerze lub hostingu. • Uczeń zna podstawowe funkcje CMS, takie jak tworzenie stron i postów, zarządzanie użytkownikami i instalacja wtyczek. • Uczeń potrafi wyjaśnić, w jaki sposób CMS upraszcza aktualizację i zarządzanie treścią.
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć i edytować strony oraz wpisy w CMS, używając narzędzi edytora wizualnego i edytora kodu HTML. • Uczeń umie instalować i konfigurować wtyczki oraz motywy, aby rozszerzać funkcjonalność i zmieniać wygląd strony. • Uczeń potrafi zarządzać użytkownikami CMS (tworzenie nowych kont, przypisywanie ról, nadawanie uprawnień). • Uczeń zna zasady zarządzania mediami (obrazy, filmy) w CMS i umie je dodawać do treści. • Umie zarządzać strukturą nawigacji strony (menu, kategorie). • Uczeń potrafi przeprowadzać podstawowe kopie zapasowe i przywracanie danych w CMS.
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi analizować i diagnozować problemy związane z funkcjonowaniem CMS (np. problemy z wydajnością, błędy w wtyczkach). • Uczeń umie optymalizować stronę

	<p>internetową opartą na CMS pod kątem szybkości działania (np. optymalizacja obrazów, buforowanie).</p> <ul style="list-style-type: none"> • Uczeń potrafi analizować bezpieczeństwo CMS, instalować aktualizacje i zabezpieczenia (np. wtyczki antyspamowe, firewalle). • Uczeń zna zasady tworzenia kopii zapasowych i zarządzania nimi. • Uczeń umie ocenić różne CMS-y i wskazać ich zalety i wady w zależności od potrzeb projektu. • Uczeń potrafi analizować potrzeby klienta i wybrać odpowiedni system CMS do realizacji danego projektu.
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie tworzyć zaawansowane strony internetowe oparte na CMS, z wykorzystaniem dostosowanych motywów i zaawansowanych funkcji. • Uczeń umie tworzyć i edytować motywy oraz wtyczki, aby dostosować stronę do specyficznych potrzeb. • Uczeń potrafi integrować CMS z innymi systemami (np. CRM, e-commerce, systemy mailingowe) za pomocą API i wtyczek. • Uczeń krytycznie ocenia działanie strony, przeprowadza testy wydajności, optymalizuje ją pod kątem SEO i UX. • Uczeń potrafi samodzielnie zarządzać serwisem opartym na CMS, włącznie z zarządzaniem bazą danych, serwerem oraz procesami automatyzacji zadań. • Uczeń tworzy dokumentację i instrukcje dla innych użytkowników systemu oraz potrafi prowadzić szkolenia z obsługi CMS.
Grafika komputerowa	<p>scharakteryzować grafikę rastrową, scharakteryzować grafikę wektorową, scharakteryzować typy plików graficznych, scharakteryzować modele barw, dobierać oprogramowanie do obróbki grafiki komputerowej, wykorzystać kodowanie kolorów na stronach www, objaśnić funkcje edytorów grafiki rastrowej i edytorów grafiki wektorowej,</p>
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie posiada wiedzy na temat grafiki komputerowej. • Uczeń nie posiada wiedzy na temat podstawowych narzędzi do tworzenia grafiki. • Uczeń nie posiada umiejętności korzystania z podstawowych funkcji edytorów grafiki (np. brak umiejętności otwarcia, edycji lub zapisania pliku graficznego).
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi wymienić podstawowe formaty plików graficznych (np. JPEG, PNG, GIF). • Uczeń potrafi rozpoznać narzędzia do tworzenia grafiki komputerowej (np. GIMP, Photoshop, Paint).

	<ul style="list-style-type: none"> • Uczeń potrafi wskazać elementy interfejsu wybranego edytora grafiki (np. pasek narzędzi, warstwy, paleta kolorów).
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi wyjaśnić różnice między rastrową a wektorową grafiką komputerową. • Uczeń potrafi opisać podstawowe cechy formatów plików graficznych (np. kompresja stratna vs bezstratna). • Uczeń potrafi stworzyć prostą grafikę komputerową, korzystając z podstawowych narzędzi edytora (np. pędzla, wypełnienia, tekstu). • Uczeń potrafi zastosować podstawowe operacje na grafice (np. przycięcie obrazu, zmiana rozmiaru, zapis w wybranym formacie).
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć bardziej zaawansowane grafiki, używając warstw, efektów i filtrów. • Uczeń potrafi edytować grafikę, stosując techniki retuszu (np. usuwanie niedoskonałości, poprawianie kolorów, praca z maskami). • Uczeń potrafi pracować z grafiką wektorową (np. tworzyć kształty, edytować węzły, skalować grafiki bez utraty jakości). • Uczeń potrafi zastosować proste techniki kompresji i optymalizacji plików graficznych do różnych zastosowań (np. na potrzeby stron www).
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi przeanalizować różnice między różnymi formatami plików graficznych w kontekście ich zastosowań (np. PNG dla przezroczystości, JPEG dla zdjęć, SVG dla grafiki wektorowej). • Uczeń potrafi oceniać jakość grafiki w zależności od jej kompresji i optymalizacji, sugerując odpowiednie formaty i ustawienia dla określonych celów. • Uczeń potrafi używać zaawansowanych funkcji edytorów grafiki, takich jak praca z warstwami, zaawansowane narzędzia do selekcji, maski warstw, korekcie kolorystyczne.
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć złożone projekty graficzne (np. plakaty, logotypy, materiały reklamowe) od podstaw, stosując zaawansowane techniki grafiki rastrowej i wektorowej. • Uczeń potrafi przeprowadzać pełną optymalizację plików graficznych pod kątem

	<p>różnych mediów (druk, web, multimedia), świadomie dobierając formaty, kolory, rozdzielczość.</p> <ul style="list-style-type: none"> • Uczeń potrafi tworzyć grafiki 3D lub korzysta z narzędzi do grafiki trójwymiarowej (jeśli jest to częścią programu). • Uczeń potrafi oceniać projekty graficzne pod kątem estetyki, funkcjonalności i wydajności (np. oceniać UX/UI, ergonomię projektów graficznych dla stron www).
Multimedia na stronie www	<p>dobrac oprogramowanie do edycji dźwięku i obrazu ruchomego na potrzeby stron internetowych, wyjaśnić zasady tworzenia animacji, osadzić elementy multimedialne na stronie internetowej</p>
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie posiada wiedzy na temat multimedii na stronie www. • Uczeń nie posiada umiejętności wstawiania jakichkolwiek multimedii na stronę. • Uczeń nie rozumie roli multimedii w kontekście użytkownika i funkcjonalności strony.
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi zdefiniować podstawowe rodzaje multimedii (grafika, wideo, audio). • Uczeń potrafi wymienić narzędzia lub technologie używane do osadzania multimedii na stronie www (np. HTML, CSS, formaty plików). • Uczeń potrafi wskazać podstawowe tagi HTML do osadzania multimedii, np. , <audio>, <video>.
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi wyjaśnić, jakie są funkcje poszczególnych typów multimedii na stronie (np. wideo jako element wizualny, audio jako element dźwiękowy). • Uczeń potrafi opisać, jak wstawienie różnych multimedii wpływa na stronę internetową (np. poprawa interaktywności, ale wzrost czasu ładowania strony). • Uczeń potrafi zastosować podstawowe multimedia na stronie (np. osadzić obrazek i wideo za pomocą tagów HTML).
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie osadzić różne multimedia (grafikę, wideo, audio) na stronie internetowej i poprawnie dostosować je do responsywnego wyświetlania na różnych urządzeniach. • Uczeń potrafi skonfigurować podstawowe atrybuty mediów (np. szerokość, wysokość, odtwarzanie wideo, napisy, alt text). • Uczeń potrafi używać prostych narzędzi lub bibliotek (np. CSS do stylizacji obrazów,

	odtwarzacz wideo HTML5).
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi przeanalizować i ocenić, jakie multimedia są odpowiednie dla konkretnej strony internetowej pod kątem użyteczności, estetyki i wydajności. • Uczeń potrafi zidentyfikować potencjalne problemy związane z wydajnością strony wynikające z użycia multimediów (np. zbyt duży rozmiar plików) i zasugerować optymalizacje. • Uczeń potrafi używać zewnętrznych narzędzi do optymalizacji multimediów (np. kompresja obrazów, streaming wideo).
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi zoptymalizować stronę pod kątem użycia multimediów, np. poprzez zastosowanie technik lazy loading, CDN, kompresji plików, itp. • Uczeń potrafi tworzyć interaktywne elementy multimedialne (np. galerie zdjęć, animacje, interaktywne mapy) za pomocą JavaScript lub gotowych bibliotek (np. Lightbox, Plyr). • Uczeń potrafi oceniać wpływ użycia multimediów na SEO, szybkość ładowania strony i doświadczenie użytkownika (UX), proponując zaawansowane rozwiązania.
Projektowanie stron internetowych	zaprojektować układ sekcji na stronie internetowej, analizować projekt strony internetowej, stworzyć strukturę strony wg projektu, doskonalić jakość wykonywanych działań; ułatwiać dostęp do informacji osobom niepełnosprawnym;
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie posiada znajomości podstawowych terminów związanych z projektowaniem stron internetowych (HTML, CSS, itp.). • Uczeń nie posiada umiejętności utworzenia prostego kodu HTML. • Uczeń nie posiada zdolności do stworzenia nawet najprostszej struktury strony (np. nagłówki, paragraf).
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi wymienić podstawowe technologie stosowane do tworzenia stron internetowych (HTML, CSS, JavaScript). • Uczeń potrafi rozpoznać podstawowe tagi HTML (np. <h1>, <p>, <a>,). • Uczeń potrafi zidentyfikować narzędzia do projektowania stron internetowych (np. edytory tekstu, przeglądarki internetowe).
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi wyjaśnić różnice między HTML a CSS (np. HTML służy do struktury strony, CSS do stylizacji).

	<ul style="list-style-type: none"> • Uczeń potrafi opisać podstawową strukturę strony internetowej (nagłówek, treść, stopka). • Uczeń potrafi tworzyć proste strony internetowe, używając podstawowych tagów HTML (np. osadzenie obrazka, utworzenie linku). • Uczeń potrafi stosować podstawowe style CSS (np. zmiana koloru, wielkości czcionki, marginesów).
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć strony internetowe, które są bardziej złożone, z zastosowaniem zaawansowanych stylów CSS (np. pozycjonowanie elementów, flexbox, grid). • Uczeń potrafi wykorzystywać podstawowe funkcje JavaScript, np. do obsługi przycisków lub prostych animacji. • Uczeń potrafi tworzyć strony, które działają responsywnie na różnych urządzeniach (RWD). • Uczeń potrafi stosować zewnętrzne biblioteki CSS, takie jak Bootstrap, w celu szybkiego projektowania interfejsów.
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi przeanalizować i ocenić, jak różne elementy strony wpływają na jej użyteczność i wydajność (np. struktura HTML, ilość zewnętrznych zasobów, optymalizacja obrazów). • Uczeń potrafi diagnozować błędy w kodzie i je poprawiać (np. błędy wyświetlania, niespójności w stylach, problemy z responsywnością). • Uczeń potrafi optymalizować strony internetowe pod kątem szybkości ładowania (np. minifikacja CSS/JavaScript, optymalizacja obrazów). • Uczeń potrafi projektować strony internetowe zgodnie z dobrymi praktykami dostępności (WCAG) oraz SEO.
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć złożone, wielostronicowe witryny internetowe, korzystając z HTML, CSS, JavaScript oraz frameworków (np. React, Vue.js). • Uczeń potrafi tworzyć dynamiczne elementy strony z wykorzystaniem JavaScript (np. formularze, interaktywne mapy, animacje). • Uczeń potrafi implementować zaawansowane techniki optymalizacji stron internetowych, takie jak lazy loading, CDN, cache. • Uczeń potrafi przeprowadzać testy

	wydajnościowe i dostępnościowe (np. narzędzia typu Lighthouse) i wprowadzać odpowiednie poprawki.
Testowanie i publikowanie stron	<p>scharakteryzować etapy procesu testowania strony internetowej,</p> <p>opisać zasady testowania responsywności strony internetowej,</p> <p>opisać proces walidacji strony internetowej,</p> <p>dobrać narzędzia walidacji strony internetowej,</p> <p>opisać usługę hostingowa,</p> <p>wyjaśnić pojęcia: domena, subdomena, serwer wirtualny, serwer dedykowany,</p> <p>opisać operacje na domenach internetowych,</p> <p>określić etapy publikacji witryn i aplikacji internetowych,</p> <p>opisać funkcje programów służących do przesyłania danych na serwer,</p> <p>dobrać program do przesyłania danych na serwer,</p> <p>ponosić odpowiedzialność za podejmowane działania,</p> <p>dokonać analizy i oceny podejmowanych działań,</p>
Ocena niedostateczna	<ul style="list-style-type: none"> • Uczeń nie posiada wiedzy na temat narzędzi i metod testowania stron internetowych. • Uczeń nie posiada umiejętności przeprowadzenia podstawowych testów strony (np. brak próby testowania strony na różnych przeglądarkach). • Uczeń nie posiada wiedzy na temat publikowania stron internetowych i niewykonanie prób publikacji.
Ocena dopuszczająca	<ul style="list-style-type: none"> • Uczeń potrafi wymienić podstawowe narzędzia i metody testowania stron internetowych (np. narzędzia deweloperskie w przeglądarkach, testowanie responsywności). • Uczeń potrafi rozpoznać różne przeglądarki i urządzenia, na których strony powinny być testowane. • Uczeń potrafi wymienić podstawowe kroki w procesie publikowania strony (np. wybór hostingu, domena, przesyłanie plików na serwer).
Ocena dostateczna	<ul style="list-style-type: none"> • Uczeń potrafi wyjaśnić znaczenie testowania strony na różnych przeglądarkach i urządzeniach (np. komputer, tablet, smartfon). • Uczeń potrafi opisać proces publikacji strony na serwerze (np. przesyłanie plików przez FTP lub używanie GitHub Pages). • Uczeń potrafi przeprowadzić podstawowe testy strony (np. sprawdzenie wyświetlania elementów na różnych rozdzielczościach)

	<p>ekranu).</p> <ul style="list-style-type: none"> • Uczeń potrafi wykonać podstawową publikację strony na darmowym hostingu.
Ocena dobra	<ul style="list-style-type: none"> • Uczeń potrafi samodzielnie przeprowadzić testy responsywności strony oraz sprawdzić jej działanie na różnych przeglądarkach (Chrome, Firefox, Safari). • Uczeń potrafi zastosować narzędzia do testowania wydajności strony (np. Google Lighthouse, GTMetrix) oraz poprawić podstawowe problemy wydajności (np. optymalizacja obrazów). • Uczeń potrafi opublikować stronę na wybranym serwerze, korzystając z narzędzi do przesyłania plików (np. FTP, SFTP, Git). • Uczeń potrafi zastosować podstawowe zasady bezpieczeństwa podczas publikacji strony (np. SSL, zabezpieczenie danych użytkowników).
Ocena bardzo dobra	<ul style="list-style-type: none"> • Uczeń potrafi analizować wyniki testów wydajnościowych i wprowadzać optymalizacje (np. skrócenie czasu ładowania strony, kompresja zasobów). • Uczeń potrafi diagnozować i naprawiać problemy wynikające z niezgodności z różnymi przeglądarkami (cross-browser compatibility). • Uczeń potrafi monitorować wydajność opublikowanej strony oraz analizować jej ruch (np. Google Analytics, narzędzia monitorujące uptime). • Uczeń potrafi implementować zaawansowane techniki testowania i publikowania (np. testowanie A/B, CDN, automatyzacja publikacji przy użyciu systemów kontroli wersji).
Ocena celująca	<ul style="list-style-type: none"> • Uczeń potrafi tworzyć kompleksowe plany testowania strony, uwzględniające testy funkcjonalne, wydajnościowe, bezpieczeństwa i UX. • Uczeń potrafi wprowadzać zaawansowane optymalizacje w oparciu o szczegółowe analizy (np. użycie cache, lazy loading, minimalizacja zasobów). • Publikować i zarządzać stroną w profesjonalnym środowisku produkcyjnym (np. z użyciem CI/CD, narzędzi do automatycznego wdrażania). • Oceniać wpływ różnych metod publikacji na wydajność strony (np. porównanie hostingów, wdrażanie technik load balancing,

CDN).