

Wymagania edukacyjne z przedmiotu :

Programowanie zaawansowanych aplikacji webowych

w roku szkolnym 2022/2023

Efekty kształcenia

Uczeń:

- rozróżnia pojęcia backend i frontend;
- tworzy użyteczny kod programistyczny;
- wykorzystuje wiedzę dotyczącą planowania projektowo-biznesowego;
- potrafi wyjaśnić różnicę między tradycyjnym podejściem do planowania i zarządzania projektem a filozofią Agile;
- rozumie zasady wizualizacji etapów realizacji projektu;
- potrafi opisać oprogramowanie wspierające zarządzanie projektami;
- rozumie znaczenie profesjonalnej obsługi klienta w skutecznej realizacji projektu informatycznego;
- rozumie podstawy filozofii Agile;
- potrafi zaplanować wygląd i interfejs aplikacji webowej;
- potrafi opisać podstawowe technologie webowe i środowisko informatyczne niezbędne do efektywnego programowania aplikacji internetowych;
- potrafi opisać alternatywne (zaawansowane) technologie, za których pomocą są tworzone aplikacje webowe;

- rozumie zasady działania narzędzi do planowania, takich jak Lean Canvas i Kanban;
- rozróżnia podstawowe stanowiska pracy charakterystyczne dla projektowania i programowania technologii webowych;
- rozumie zasady funkcjonowania zespołu projektowego;
- rozumie zasady wykorzystania HTML, CSS, JavaScript i PHP;
- wykorzystuje środowisko programistyczne dla aplikacji zaawansowanych webowych;
- wykorzystuje frameworki do programowania aplikacji webowych;
- programuje zaawansowane aplikacje webowych;
- dobiera programistyczne do i programowania (np. Visual Studio, Eclipse Jet Brains) , stosuje IDE, RAD 3;
- rozpoznaje wykorzystywane w procesie tworzenia aplikacji webowych;
- stosuje środowisko programistyczne do realizacji projektu aplikacji webowej;
- korzysta z dedykowanych narzędzi dostępnych w środowisku programistycznym przy tworzeniu aplikacji webowej w danym frameworku;
- korzysta z dedykowanych narzędzi dostępnych w środowisku programistycznym przy tworzeniu aplikacji webowej w danym języku programowania;
- implementuje podstawowe operacje w jQuery;
- modyfikuje wygląd aplikacji webowej poprzez kodu w tym HTML poprzez odpowiednią aplikację;

- generuje żądania GET w w jQuery;
- przekazuje nagłówki HTTP w jQuery;
- wczytuje kod XML w jQuery;
- odczytuje dane JSON z zewnętrznego serwera w jQuery;
- implementuje zmianę kolejności elementów przy wykorzystaniu elementów sortowalnych w jQuery UI;
- dostosowuje aplikację webową do urządzeń mobilnych w jQuery;
- implementuje nawigowanie aplikacji webowej przy użyciu jQuery Mobile;
- implementować interakcję aplikacji webowej z użytkownikiem na urządzeniu mobilnym przy użyciu jQuery Mobile;
- korzystanie i modyfikowanie gotowych wtyczek w jQuery;
- możliwość wykorzystania strumieni i HTTP w node.js;
- stworzyć aplikację z wykorzystaniem frameworku express.js w Node.js,
- zainstalowanie MongoDB w Node.js;
- implementowanie podstawowych poleceń Mongo;
- połączyć Node.js z bazą danych;
- dodawać nowe wartości do nazwy danych z poziomu Node.js;
- doraźnie wykorzystać Web Socket w Node.js;
- zainstalować język Python na serwerze;
- dokonywanie edycji zmiennych środowiskowych na serwerze w celu instalacji Django;
- zainstalowanie Django na serwerze;

- tworzenie projektu aplikacji webowej w Django;
- przygotowanie bazy danych do projektu aplikacji wykonanej w Django;
- opracowanie modeli danych stosowanych w aplikacji webowej wykonanej w Django;
- wprowadzić dane do tabel w bazie danych dla aplikacji webowej wykonanej w Django;
- opracowanie formularza logowania się użytkowników do aplikacji webowej wykonanej w Django;
- zautomatyzowanie procesy np. powiadamianie użytkowników w aplikacji webowej wykonanej w wybranym programie;
- zaprogramowanie aplikacji internetowej w wybranym języku programowania np. PHP, C#, Python, JavaScript;
- zaprogramowanie aplikacji webowej wykorzystując mechanizm sesji i ciasteczek;
- zaprogramowanie aplikacji webowej zawierającej dynamiczne formularze;
- zaprogramowanie systemu logowania do aplikacji webowej;
- zaprogramowanie systemu kontroli dostępu do określonych elementów witryny;
- zaprogramowanie aplikacji webowej z dostępem do baz danych;
- zaprogramowanie wybranych elementów e-sklepu;
- zaprogramowanie wybranego elementu portalu społecznościowego;
- zaprogramowanie wybranego elementu serwisu ogłoszeń internetowych,
- zaprogramować wybrane elementy serwisu rezerwacyjnego;

- testowanie zaprogramowanej aplikacji użytkownika;
- dokumentowanie kod zaprogramowanej aplikacji;
- publikowanie aplikacji webowej na serwerze.

I. Zalecane warunki i sposób realizacji

Szkoła prowadząca kształcenie w zawodzie zapewnia pomieszczenia dydaktyczne z wyposażeniem odpowiadającym technologii i technice stosowanej w zawodzie, aby zapewnić osiągnięcie wszystkich efektów kształcenia określonych w podstawie programowej kształcenia w zawodzie szkolnictwa branżowego oraz umożliwić przygotowanie absolwenta do wykonywania zadań zawodowych.

II. Sposoby sprawdzania i oceniania osiągnięć edukacyjnych uczniów

Podczas egzekwowania osiągnięć uczniów decydującą rolę odgrywa jasność, przejrzystość i precyzyjność w ustalaniu kryteriów wystawianych ocen. Uczeń wie, za co jaką ocenę może otrzymać, o jego postępach są również na bieżąco informowani rodzice. Ocenie podlega zarówno wiedza teoretyczna, jak i nabyte w trakcie nauki umiejętności. Lekcje w dużej części polegające na przyswajaniu sobie umiejętności posługiwania się różnego typu narzędziami i programami z opcjami o różnym stopniu trudności, pozwolą dokładnie omawiane kryteria opracować.

Ponieważ w programowaniu często spotykamy się z istnieniem kilku rozwiązań tego samego zadania czy problemu, warto premiować wysokimi ocenami uczniów proponujących rozwiązania nie omawiane na lekcji, do których doszli w wyniku własnej pomysłowości i dociekliwości albo też dzięki pogłębionemu zaznajomieniu się z omawianym tematem.

Wiedzę teoretyczną sprawdzać można standardowo przez odpytanie ucznia, również poprzez kartkówki, sprawdziany pisemne lub przede wszystkim praktyczne, ale też grupowe projekty, wymagające posiłkowania się taką wiedzą.

Doceniać należy aktywność uczniów, aby podtrzymać zaangażowanie podczas kolejnych lekcji. Uczniowie, którzy wykonali ćwiczenie szybciej niż inni, mogą otrzymać zadania dodatkowe, które warto oddzielnie punktować. Liczne ćwiczenia, całonocne kilkugodzinne projekty i wszelkie praktyczne zadania oceniać należy według kryteriów omówionych wyżej.

Oceniane są:

1. Ćwiczenia, zadania wykonywane na lekcji.

Ocenie podlega:

- wykonanie wszystkich poleceń zgodnie z treścią;
- stopień samodzielności wykonywania zadania;
- pilność;
- końcowy efekt pracy (jakość pracy).
- umiejętność pracy w zespole;

2. Odpowiedzi ustne.

Oceniany jest sposób rozumienia oraz stosowania podstawowych terminów informatycznych: najczęściej odpowiedź ustna związana jest z wypowiedziami uczniów w trakcie dyskusji i pracy z dokumentacją projektową.

3. Aktywność podczas pracy na lekcji.

Ocenie podlega:

- aktywność ucznia w czasie zajęć;
- stopień zaangażowania podczas wykonywania zajęć;
- zainteresowanie tematem lekcji;

- przygotowanie dodatkowych materiałów do lekcji.

4. Kartkówki, sprawdziany pisemne lub praktyczne.

5. Zadania domowe, zadania dodatkowe, prace projektowe.

W warunkach pracy zdalnej:

a) postępy uczniów są monitorowane na podstawie bieżących osiągnięć w pracy zdalnej (nauczanie w czasie rzeczywistym) lub na podstawie wykonanych m.in.: ćwiczeń, prac, quizów, projektów, zadań w czasie odroczonym (nauczanie w czasie odroczonym).

b) ocenianie bieżące polega na wystawieniu oceny zgodnie z przyjętą skalą stopni, z uwzględnieniem w szczególności: co uczeń zrobił dobrze, co wymaga poprawy, a także wysiłku wkładanego w wykonanie zadania przez ucznia za:

1) odpowiedzi ustne w czasie zajęć on-line lub w trakcie rozmów telefonicznych;

2) wypowiedzi uczniów na czacie tekstowym, na forum dyskusyjnym;

3) wypowiedzi uczniów w czasie wideokonferencji/webinariów i innych form komunikowania się on-line;

4) wykonanych w domu zadań zleconych przez nauczyciela, przesłanych poprzez Dziennik elektroniczny, platformę Moodle (<http://kursy.blich.pl>) lub Teams, lub w szczególnych przypadkach inną drogą elektroniczną w uzgodnieniu z nauczycielem

c) Nauczyciel może wymagać od uczniów wykonania określonych poleceń, zadań, prac, projektów umieszczonych w Internecie, np. na zintegrowanych platformach edukacyjnych lub poprosić o samodzielne wykonanie pracy w domu i udokumentowanie jej, np. w postaci zdjęcia przesłanego na adres e-mail nauczyciela.

d) Nauczyciele w pracy zdalnej wskazują dokładny czas i ostateczny termin wykonania zadania, określając jednocześnie warunki ewentualnej poprawy, jeśli zadanie nie zostało wykonane w sposób prawidłowy lub zawiera błędy.

e) Nauczyciel informuje ucznia o postępach w nauce i ocenach podczas bieżącej pracy z dzieckiem lub po jej zakończeniu przez e-dziennik.

f) na ocenę osiągnięć ucznia z danego przedmiotu nie mogą mieć wpływu czynniki związane z ograniczonym dostępem do sprzętu komputerowego i do Internetu.

g) na ocenę osiągnięć ucznia z danego przedmiotu nie może mieć wpływu poziom jego kompetencji cyfrowych. Nauczyciel ma obowiązek wziąć pod uwagę zróżnicowany poziom umiejętności obsługi narzędzi informatycznych i dostosować poziom trudności wybranego zadania oraz czas jego wykonania do możliwości psychofizycznych ucznia.

III. WYMAGANIA NA POSZCZEGÓLNE OCENY:

Ocena niedostateczna

Uczeń:

- nie opanował wiedzy i umiejętności z zakresu materiału programowego;
- nie zna terminologii ani oprogramowania wspierającego zarządzanie projektami;
- nie jest w stanie napisać aplikacji;
- rozwiązuje testy zawodowe i próbne egzaminy na poziomie poniżej 50%.

Ocena dopuszczająca

Uczeń:

- nie wykazuje zainteresowania przedmiotem;
- posiada minimalny wymagany zasób wiedzy teoretycznej;
- z pomocą nauczyciela, często niezbyt chętnie, posługuje zasadami wykorzystania HTML, CSS, JavaScript i PHP w zakresie minimalnym;
- z dużą pomocą nauczyciela pisze proste aplikacje;

- ćwiczenia, prace i projekty wykonuje niestarannie, z dużymi brakami w stosunku do zaprezentowanego przez nauczyciela wzoru lub przykładu, z wykorzystaniem najprostszych opcji i narzędzi;
- zna podstawowe środowisko programistyczne do realizacji projektu aplikacji webowej;
- prawidłowo rozróżnia podstawowe stanowiska pracy charakterystyczne dla projektowania i programowania technologii webowych;
- nazywa podstawowe elementy kodowania;
- rozwiązuje testy zawodowe i próbne egzaminy na poziomie 50-60%.

Ocena dostateczna

Uczeń:

- nie wykazuje zbytniego zainteresowania przedmiotem, niemniej zadane ćwiczenia i prace stara się, mimo trudności, wykonać jak najlepiej;
- w posiadanej wiedzy teoretycznej prezentuje duże braki, niemniej większość materiału ma opanowaną;
- z niewielką pomocą nauczyciela posługuje się programami i dokumentacją projektową zaawansowanych aplikacji webowych;
- z niewielką pomocą nauczyciela rozróżnia elementy aplikacji i potrafi opracować model danych stosowanych w aplikacji webowej;
- wykonuje ćwiczenia, prace i projekty z niedbałością, prostotą, brakiem zastosowania wielu opcji i efektów;
- potrafi opisać podstawowe technologie webowe i środowisko informatyczne niezbędne do efektywnego programowania aplikacji internetowych;
- potrafi opisać alternatywne (zaawansowane) technologie, za których pomocą są tworzone aplikacje webowe;

- rozumie zasady działania narzędzi do planowania, takich jak Lean Canvas i Kanban;
- rozróżnia podstawowe stanowiska pracy charakterystyczne dla projektowania i programowania technologii webowych;
- rozumie zasady funkcjonowania zespołu projektowego• rozwiązuje testy zawodowe i próbne egzaminy na poziomie 60-70%.

Ocena dobra

Uczeń:

- rozróżnia pojęcia backend i frontend;
- tworzy użyteczny kod programistyczny;
- wykorzystuje wiedzę dotyczącą planowania projektowo-biznesowego;
- potrafi wyjaśnić różnicę między tradycyjnym podejściem do planowania i zarządzania projektem a filozofią Agile;
- rozumie zasady wizualizacji etapów realizacji projektu;
- potrafi opisać oprogramowanie wspierające zarządzanie projektami;
- rozumie znaczenie profesjonalnej obsługi klienta w skutecznej realizacji projektu informatycznego;
- rozumie podstawy filozofii Agile;
- potrafi zaplanować wygląd i interfejs aplikacji webowej;
- potrafi opisać podstawowe technologie webowe i środowisko informatyczne niezbędne do efektywnego programowania aplikacji internetowych;
- potrafi opisać alternatywne (zaawansowane) technologie, za których pomocą są tworzone aplikacje webowe;
- rozumie zasady działania narzędzi do planowania, takich jak Lean Canvas i Kanban;

- rozróżnia podstawowe stanowiska pracy charakterystyczne dla projektowania i programowania technologii webowych;
- rozumie zasady funkcjonowania zespołu projektowego• rozwiązuje testy zawodowe i próbne egzaminy na poziomie 70-80%.

Ocena bardzo dobra

Uczeń:

- programuje aplikacje webowe zawierające dynamiczne formularze;
- programuje systemy logowania do aplikacji webowej;
- stosuje system kontroli dostępu do określonych elementów witryny;
- potrafi zaprogramować aplikacje webowe z dostępem do baz danych;
- wykonuje ćwiczenia zaprogramowania wybranych elementów e-sklepu;
- wykonuje ćwiczenia wybranych elementu portalu społecznościowego;
- stosuje w projektach wybrane elementy serwisu ogłoszeń internetowych;
- wykorzystuje w ćwiczeniach testowanie zaprogramowanej aplikacji użytkownika;
- dokumentuje kod zaprogramowanej aplikacji;
- wykonuje ćwiczenia, prace i projekty z dużą starannością i dokładnością w odtworzeniu zaprezentowanego przez nauczyciela wzoru, przykładu;
- szczegółowo omawia programy i języki programowania oraz metody tworzenia projektów zaawansowanych aplikacji webowych;
- charakteryzuje etapy procesu testowania aplikacji webowej;
- stosuje zasady testowania oprogramowania zaawansowanej aplikacji webowej;
- rozwiązuje testy zawodowe i próbne egzaminy na poziomie 80-90%.

Ocena celująca

Uczeń:

- zna wymagane pojęcia i terminologię oprogramowania wspierającego zarządzanie projektami;
- posiada wymaganą na tym etapie nauczania przedmiotu wiedzę teoretyczną;
- perfekcyjnie i z dużą swobodą posługuje się narzędziami i dokumentacją aplikacji webowych;
- korzysta z dedykowanych narzędzi dostępnych w środowisku programistycznym przy tworzeniu aplikacji webowej w danym języku programowania;
- implementuje podstawowe operacje w jQuery;
- modyfikuje wygląd aplikacji webowej poprzez kod w tym HTML poprzez odpowiednią aplikację;
- zautomatyzowanie procesy np. powiadamianie użytkowników w aplikacji webowej wykonanej w wybranym programie;
- zaprogramowanie aplikacji internetowe w wybranym języku programowania np. PHP, C#, Python, JavaScript;
- zaprogramowanie aplikacji webowej wykorzystując mechanizm sesji i ciasteczek;
- zaprogramowanie aplikacji webowej zawierającej dynamiczne formularze;
- zaprogramowanie systemu logowania do aplikacji webowej;
- samodzielnie rozwiązuje przedstawione na zajęciach problemy programistyczne;
- wykonuje ćwiczenia, prace i projekty z dużym stopniem samodzielności i własnej inwencji, złożonością oraz pomysłowością, oryginalnością;
- do swoich prac pozyskuje materiał z bardzo różnych źródeł wiedzy;

- wyróżnia się starannością i solidnością podczas wykonywania powierzonych zadań oraz aktywnością na lekcjach;
- wykazuje ponadprzeciętne zainteresowanie przedmiotem, mogące objawiać się poszerzoną wiedzą i umiejętnościami zdobywanymi we własnym zakresie;
- bierze udział w konkursach IT;
- rozwiązuje testy zawodowe i próbne egzaminy na poziomie powyżej 90%.

Znak '+' dodawany jest do oceny podstawowej w ocenianiu bieżącym (oprócz oceny celującej) gdy uczeń nie spełnił

wymaganych kryteriów na daną ocenę, a poziom jego wiedzy i umiejętności przekracza według nauczyciela wymagania na ocenę niższą – do 50% wymagań.

Znak '-' dodawany jest do oceny podstawowej w ocenianiu bieżącym (oprócz oceny niedostatecznej) gdy uczeń według

nauczyciela nie spełnił wymaganych kryteriów na daną ocenę- brak 25% wymaganych kryteriów.