

Wymagania edukacyjne i PSO

Przedmiot: Programowanie aplikacji internetowych (klasa 4C, 4GH)

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

Rodzaj zajęć: zajęcia praktyczne w pracowni komputerowej

Nauczyciel: Michał Bednarek

1. Cele kształcenia

Uczeń w wyniku kształcenia powinien:

- obsługiwać zdarzenia w JavaScript i jQuery,
- znać zasady BHP i regulamin pracowni komputerowej,
- walidować i przetwarzać formularze za pomocą JavaScript,
- stosować zasady programowania obiektowego w JavaScript,
- tworzyć aplikacje obiektowe,
- korzystać z biblioteki jQuery (zdarzenia, efekty, animacje),
- wykorzystywać technologię AJAX do dynamicznej komunikacji aplikacji z serwerem,
- stosować element **Canvas** do tworzenia grafik i animacji,
- projektować i implementować pełne aplikacje internetowe w JavaScript,
- przygotowywać dokumentację techniczną aplikacji,
- znać podstawy frameworków JS (React) i tworzyć w nich proste projekty.

2. Wymagania edukacyjne

Ocena celująca (6)

Uczeń:

- wykazuje wiedzę i umiejętności wykraczające poza program nauczania,
- samodzielnie tworzy złożone aplikacje z użyciem JS, jQuery, AJAX i React,
- stosuje wzorce projektowe i dobre praktyki programowania,
- tworzy rozbudowane animacje i grafiki w Canvas,
- aktywnie rozwija swoje umiejętności (np. udział w konkursach, dodatkowe projekty).

Ocena bardzo dobra (5)

Uczeń:

- samodzielnie realizuje wszystkie wymagania programowe,
- poprawnie obsługuje zdarzenia i formularze w JS i jQuery,
- stosuje obiektowe podejście do tworzenia aplikacji,
- implementuje AJAX w prostych projektach,
- tworzy aplikacje z wykorzystaniem Canvas i frameworka React,
- przygotowuje kompletną dokumentację aplikacji.

Ocena dobra (4)

Uczeń:

- poprawnie realizuje większość zadań praktycznych, czasem korzysta z pomocy nauczyciela,
- zna i stosuje podstawowe techniki OOP w JS,
- potrafi użyć jQuery do prostych efektów i animacji,
- implementuje podstawowe zapytania AJAX,
- potrafi stworzyć prostą aplikację w React.

Ocena dostateczna (3)

Uczeń:

- zna podstawowe metody obsługi zdarzeń i formularzy,
- wykonuje proste zadania programistyczne przy wsparciu nauczyciela,
- stosuje proste elementy jQuery i AJAX,
- potrafi stworzyć bardzo prostą aplikację w JS lub React, ale z błędami.

Ocena dopuszczająca (2)

Uczeń:

- zna jedynie elementarne pojęcia związane z JS, jQuery i AJAX,
- potrafi uruchomić przykładowe skrypty, ale nie rozumie w pełni ich działania,
- nie radzi sobie z projektowaniem aplikacji bez pomocy nauczyciela.

Ocena niedostateczna (1)

Uczeń:

- nie opanował podstawowych umiejętności,
- nie podejmuje prób wykonania zadań praktycznych,
- nie zna kluczowych zagadnień (zdarzenia, OOP, AJAX, React),
- nie stosuje zasad pracy w pracowni komputerowej.

3. Obszary oceniania

1. **Umiejętności praktyczne** – zadania w JS, jQuery, AJAX, Canvas, React.
2. **Dokumentacja i projektowanie** – dokumentacja aplikacji, opisy kodu, schematy.
3. **Postawa i aktywność** – praca na lekcji, systematyczność, współpraca w grupie.

4. Kryteria oceniania

- Zadania praktyczne i projekty – **80%**

- Dokumentacja, aktywność i praca na lekcji – **20%**

5. Zasady poprawy i zaliczeń

- Uczeń ma prawo poprawić ocenę ze sprawdzianu lub projektu w terminie ustalonym z nauczycielem.
- Nieobecność na sprawdzianie należy usprawiedliwić i napisać w dodatkowym terminie przewidzianym w Statucie Zespołu Szkół nr 4 w Nowym Sączu
- Brak wykonania zadania praktycznego skutkuje oceną niedostateczną, którą można poprawić.

6. Metody sprawdzania osiągnięć uczniów

- sprawdziany teoretyczne,
- zadania praktyczne w JS, jQuery, AJAX, React,
- ocena projektów indywidualnych i grupowych,
- obserwacja pracy ucznia na zajęciach,
- ocena dokumentacji aplikacji.

7. Szczegółowe wymagania do tematów

Uczeń powinien umieć:

- **Obsługa zdarzeń i formularzy** – reagować na zdarzenia użytkownika, walidować dane formularzy, przetwarzać je w JS.
- **Programowanie obiektowe** – stosować klasy, obiekty, metody, tworzyć aplikacje obiektowe.
- **jQuery** – wykorzystywać selektory, zdarzenia, efekty, animacje.
- **AJAX** – wysyłać i odbierać dane z serwera bez przeładowania strony.
- **Canvas** – rysować kształty, tworzyć animacje.
- **Projektowanie aplikacji** – analizować wymagania, projektować strukturę, implementować w JS i React.
- **React** – znać podstawy środowiska, potrafić utworzyć i modyfikować projekt, tworzyć proste komponenty.
- **Dokumentacja** – przygotować opis działania aplikacji, schematy i instrukcję obsługi.