|  |  |  |
| --- | --- | --- |
| **AI1** | Jakubiak Hubert  Album 50985  Grupa 4 | **Wersja 1** |
|  |
| **LAB D** |

REST API Client

# Spis treści

[Spis treści 1](#_Toc147183245)

[Cel zajęć 1](#_Toc147183246)

[Rozpoczęcie 1](#_Toc147183247)

[Uwaga 1](#_Toc147183248)

[Wymagania 2](#_Toc147183249)

[Badanie API 2](#_Toc147183250)

[Implementacja 2](#_Toc147183251)

[Commit projektu do GIT 4](#_Toc147183252)

[Podsumowanie 4](#_Toc147183253)

# Cel zajęć

Celem głównym zajęć jest zdobycie następujących umiejętności:

* pobieranie danych z zewnętrznych zasobów za pomocą REST API
* zdobywanie wiedzy na temat zewnętrznych API za pomocą dokumentacji typu Swagger
* wysyłanie asynchronicznych żądań z wykorzystaniem XMLHttpRequest i Fetch API

W praktycznym wymiarze uczestnicy stworzą dynamiczną stronę HTML pozwalającą na wyświetlanie bieżącej informacji pogodowej oraz prognoz dla zadanej przez użytkownika miejscowości.

# Rozpoczęcie

Rozpoczęcie zajęć. Powtórzenie wykonywania połączeń synchronicznych i asynchronicznych z poziomu JS na stornie.

Wejściówka?

# Uwaga

Ten dokument aktywnie wykorzystuje niestandardowe właściwości. Podobnie jak w LAB A wejdź do Plik -> Informacje -> Właściwości -> Właściwości zaawansowane -> Niestandardowe i zaktualizuj pola. Następnie uruchom ten dokument ponownie lub Ctrl+A -> F9.

# Wymagania

W ramach LAB D przygotowane powinny zostać:

* pojedyncza strona HTML ze skryptem ładowanym z zewnętrznego pliku JS
* pole tekstowe (input typu „text”) do wprowadzania adresu
* przycisk „Pogoda”, po kliknięciu którego wykonywane jest zapytanie asynchroniczne:
  + do API Current Weather: <https://openweathermap.org/current> za pomocą XMLHttpRequest
  + do API 5 day forecast: <https://openweathermap.org/forecast5> za pomocą Fetch API
* obsługa zwrotki z obu API – wypisanie pogody bieżącej oraz prognoz poniżej pola wyszukiwania.

Wygeneruj własny lub wykorzystaj gotowy klucz do API: 7ded80d91f2b280ec979100cc8bbba94

W przypadku blokady można posiłkować się filmem: <https://www.youtube.com/watch?v=WoKp2qDFxKk> jednakże spróbuj rozwiązać ten problem samodzielnie!

Prowadzący omówi powyższe wymagania. Upewnij się, czy wszystko rozumiesz.

Tu umieść swoje notatki:

…notatki…

# Badanie API

Poświęć kilka minut na wykonanie przykładowych zapytań do API z poziomu pasku adresu przeglądarki. Podaj wymagane parametry dla osiągnięcia różnych wyników. Zbadaj odpowiedzi API, aby uzyskać pełen obraz wymagań i możliwości API.

# Implementacja

Tradycyjnie implementację należy zacząć od zbudowania w HTML + CSS wszystkich wymaganych elementów / placeholderów na te elementy. Następnie krok po kroku należy implementować poszczególne zachowania.

Wstaw zrzut ekranu zawierającego stronę ze wszystkimi elementami, tj. pole tekstowe, przycisk, miejsce do wyświetlenia pogody i prognozy:

Obraz zawierający tekst, Czcionka, numer, zrzut ekranu

Opis wygenerowany automatycznie

|  |  |  |
| --- | --- | --- |
| Punkty: | 0 | 1 |

Wstaw zrzut ekranu kodu odpowiedzialnego za wysyłanie żądania do current za pomocą XMLHttpRequest:

Obraz zawierający tekst, zrzut ekranu, Oprogramowanie multimedialne, oprogramowanie

Opis wygenerowany automatycznie

Wstaw zrzut ekranu pokazujący otrzymaną odpowiedź za pomocą console.log() w przeglądarce.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

|  |  |  |
| --- | --- | --- |
| Punkty: | 0 | 1 |

Wstaw zrzut ekranu kodu odpowiedzialnego za wysyłanie żądania do forecast za pomocą Fetch:

Obraz zawierający tekst, Oprogramowanie multimedialne, zrzut ekranu, oprogramowanie

Opis wygenerowany automatycznie

Wstaw zrzut ekranu pokazujący otrzymaną odpowiedź za pomocą console.log() w przeglądarce.

Obraz zawierający zrzut ekranu, wzór, materiał, sztuka

Opis wygenerowany automatycznie

|  |  |  |
| --- | --- | --- |
| Punkty: | 0 | 1 |

Wstaw zrzut ekranu przedstawiającego wizualizację prognoz pogody:

Obraz zawierający tekst, Czcionka, numer, zrzut ekranu

Opis wygenerowany automatycznie

Upewnij się, że widoczne są pasek wyszukiwania ze wskazaną miejscowością, a także zarówno pogoda bieżąca jak i prognozy pogody.

|  |  |  |
| --- | --- | --- |
| Punkty: | 0 | 1 |

# Commit projektu do GIT

Zacommituj i pushnij swoje rozwiązanie do repozytorium GIT.

Upewnij się, czy wszystko dobrze się wysłało. Jeśli tak, to z poziomu przeglądarki utwórz branch o nazwie lab-d na podstawie głównej gałęzi kodu.

Podaj link do brancha lab-d w swoim repozytorium:

…link, np. https://github.com/jakhub21/jakubiak/tree/lab-d/AI1-LAB-D

# Podsumowanie

W kilku zdaniach podsumuj zdobyte podczas tego laboratorium umiejętności.

W trakcie tego laboratorium zdobyłem umiejętności związane z tworzeniem interaktywnego interfejsu webowego do sprawdzania aktualnej pogody. Nauczyłem się integrować zewnętrzne API, takie jak OpenWeatherMap, przy użyciu języka JavaScript. Tworzenie dynamicznych interfejsów, korzystając z XMLHttpRequest oraz funkcji fetch, umożliwiło mi pobieranie danych pogodowych i aktualizowanie interfejsu w czasie rzeczywistym. Ponadto, zyskałem doświadczenie w manipulowaniu elementami HTML i ich stylizacji za pomocą CSS, tworzeniu dynamicznych komponentów przy użyciu JavaScript oraz organizowaniu i prezentowaniu danych w czytelny sposób na stronie internetowej.

Zweryfikuj kompletność sprawozdania. Utwórz PDF i wyślij w terminie.