|  |  |  |
| --- | --- | --- |
| **AI** | Mielniczuk Marcin  Album 51621  Grupa 1 | **Wersja 1** |
|  |
| **LAB D** |

REST API Client

# Spis treści

[Spis treści 1](#_Toc188183835)

[Cel zajęć 1](#_Toc188183836)

[Rozpoczęcie 1](#_Toc188183837)

[Uwaga 1](#_Toc188183838)

[Wymagania 2](#_Toc188183839)

[Badanie API 2](#_Toc188183840)

[Implementacja 2](#_Toc188183841)

[Commit projektu do GIT 7](#_Toc188183842)

[Podsumowanie 7](#_Toc188183843)

# 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 klucz do API. Ponieważ aktywacja może chwilę potrwać, na czas trwania laboratorium możesz wykorzystać „służbowy” klucz: 7ded80d91f2b280ec979100cc8bbba94. **UWAGA!** Klucz zostanie dezaktywowany niedługo po zajęciach. Musisz wygenerować swój własny.

W przypadku blokady twórczej 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, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

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

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

Obraz zawierający tekst, elektronika, zrzut ekranu, wyświetlacz

Opis wygenerowany automatycznie

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

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

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

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

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

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



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

Wstaw zrzut ekranu przedstawiającego wizualizację prognoz pogody:

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

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:

. <https://github.com/mm51621/main/tree/main/AI/labD>

# Podsumowanie

W kilku zdaniach podsumuj zdobyte podczas tego laboratorium umiejętności.

Podczas tego laboratorium nauczyłem się, jak korzystać z API OpenWeatherMap do pobierania danych o pogodzie. Zrozumiałem, jak działa wysyłanie żądań HTTP za pomocą zarówno starszego obiektu XMLHttpRequest, jak i nowoczesnej metody fetch. Dowiedziałem się także, jak przetwarzać odpowiedzi z API w formacie JSON oraz wyświetlać je w przeglądarce, tworząc dynamiczne elementy HTML. Dzięki dodaniu parametru lang=pl mogłem dostosować język odpowiedzi API do polskiego.

…podsumowanie…

Zweryfikuj kompletność sprawozdania. Utwórz PDF i wyślij w terminie.