

## ZAAWANSOWANE JS W PRZEGŁĄDARCE - PUZZLE

### SPIS TREŚCI

Spis treści .....	1
Cel zajęć.....	1
Rozpoczęcie.....	1
Uwaga .....	1
Wymagania.....	2
Implementacja .....	2
Commit projektu do GIT.....	8
Podsumowanie.....	9

### CEL ZAJĘĆ

Celem głównym zajęć jest zdobycie następujących umiejętności:

- pobieranie lokalizacji w przeglądarce z wykorzystaniem Geolocation API
- wyświetlanie map z wykorzystaniem biblioteki Leaflet
- pobieranie map statycznych z wykorzystaniem Leaflet
- podział obrazów na sekcje z wykorzystaniem JS
- przestawianie elementów z wykorzystaniem Drag & Drop
- wyświetlanie powiadomień

W praktycznym wymiarze uczestnicy stworzą dynamiczną aplikację – układankę – w której gracz będzie musiał ułożyć 16 elementów uprzednio wskazanej i pobranej mapy.

### ROZPOCZĘCIE

Rozpoczęcie zajęć. Powtóżenie Geolocation API, Leaflet, Drag & Drop, Canvas.

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 C przygotowane powinny zostać:

- pojedyncza strona HTML ze skryptem ładowanym z zewnętrznego pliku JS
- pobranie zgody na lokalizację
- pobranie zgody na wyświetlanie powiadomień
- okno dynamicznej mapy (powiększanie/pomniejszanie, przesuwanie)
- przycisk „Moja lokalizacja” – wyświetla współrzędne oraz oznacza na mapie
- przycisk „Pobierz mapę” – eksportuje mapę w postaci rastrowej
- mapa rastrowa zostaje podzielona na 16 elementów i wymieszana; elementy rozrzucone na „stole”
- użycie mechanizmu drag & drop do przemieszczania elementów na „stole”
- w tle weryfikacja czy element ustawiony na swoim miejscu
- w momencie ustwienia wszystkich elementów na swoim miejscu – wyświetlenie notyfikacji

Wideo z omówieniem działającej aplikacji: [https://www.youtube.com/watch?v=Peb\\_mgDTY0s](https://www.youtube.com/watch?v=Peb_mgDTY0s).

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

Tu umieść swoje notatki:

...notatki...

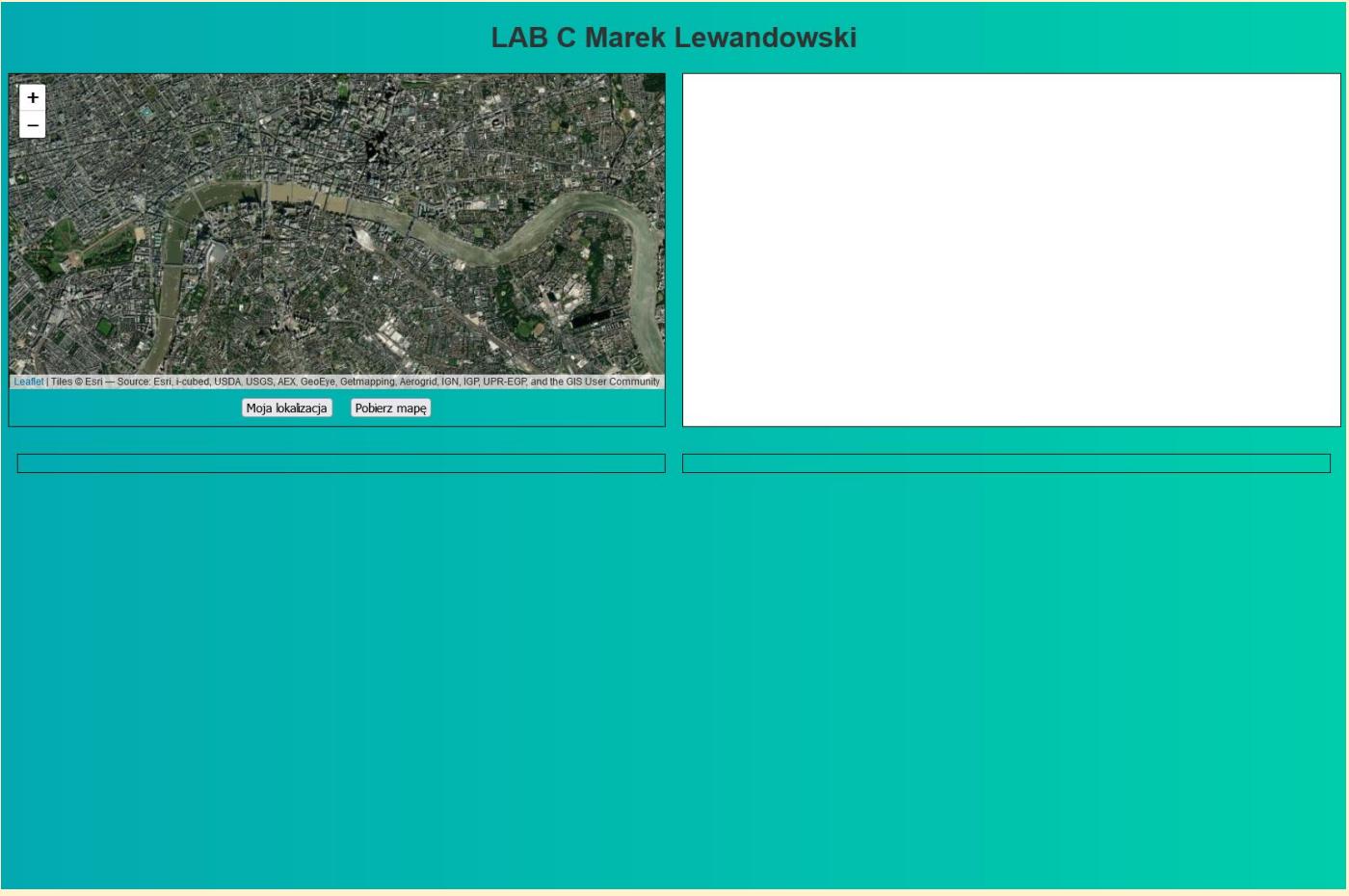
## IMPLEMENTACJA

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

Dopiero po skończeniu implementacji całości zadania zrób i powklejaj zrzuty ekranu.

**UWAGA!** Większość kodu jest już zrobiona! Wystarczy przejrzeć kody źródłowe prezentowane na wykładach 😊

Wstaw zrzut ekranu zawierającego stronę ze wszystkimi elementami, tj. okno dynamicznej mapy, przycisk „Moja lokalizacja”, przycisk „Pobierz mapę”, przestrzeń z rozsypanymi puzzlami, przestrzeń do układania puzzli. Wygląd może być odmienny od zaprezentowanego na wideo:



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

## AI1 LAB C – Lewandowski Marek – Wersja 1

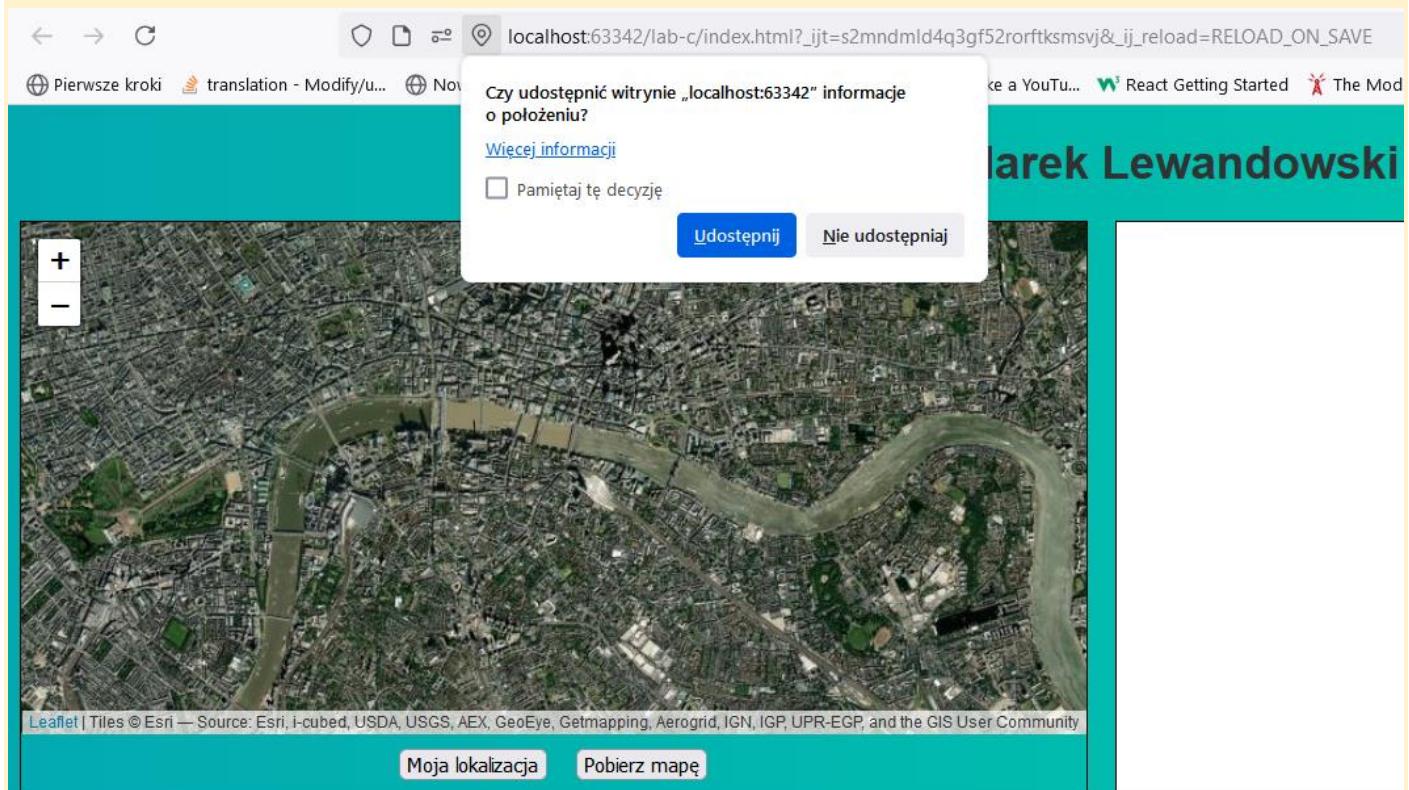
Wstaw zrzuty ekranu z załadowaną dynamiczną mapą, inną lokalizacją na mapie, innym przybliżeniem mapy:

The image displays three separate screenshots of a map application interface, likely Leaflet, arranged vertically. Each screenshot shows a map with various geographical features like fields, roads, and bodies of water. The top two screenshots have a teal header bar with the text "LAB C Marek Lewandowski". The bottom screenshot has a larger teal header bar with the same text. Each map includes standard controls such as zoom (+/-), orientation (compass rose), and location (geolocation icon). At the bottom of each map view, there are buttons for "Moja lokalizacja" and "Pobierz mapę". The middle screenshot shows a map of a rural area with green fields and brown paths. The bottom screenshot shows a close-up view of a grassy field with some brown patches.

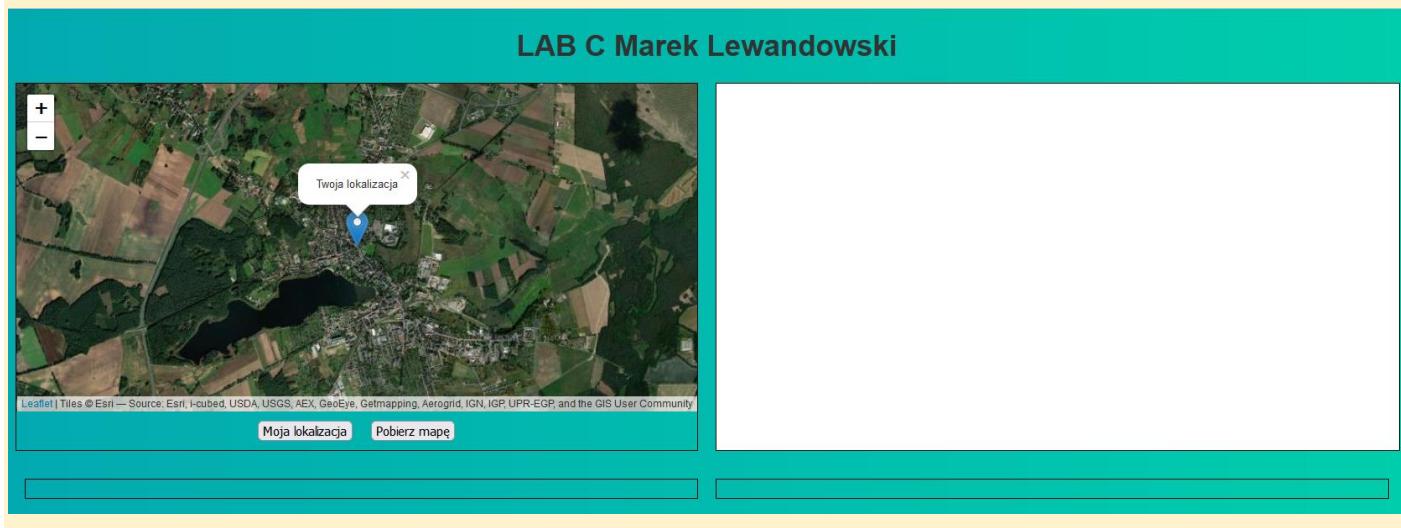
Punkty:

0	1
---	---

Przedstaw zrzut ekranu przeglądarki proszącej o zgodę na udostępnienie geolokalizacji:



Wstaw zrzut ekranu wycentrowanej mapki na pobranej geolokalizacji:



Punkty:

0

1

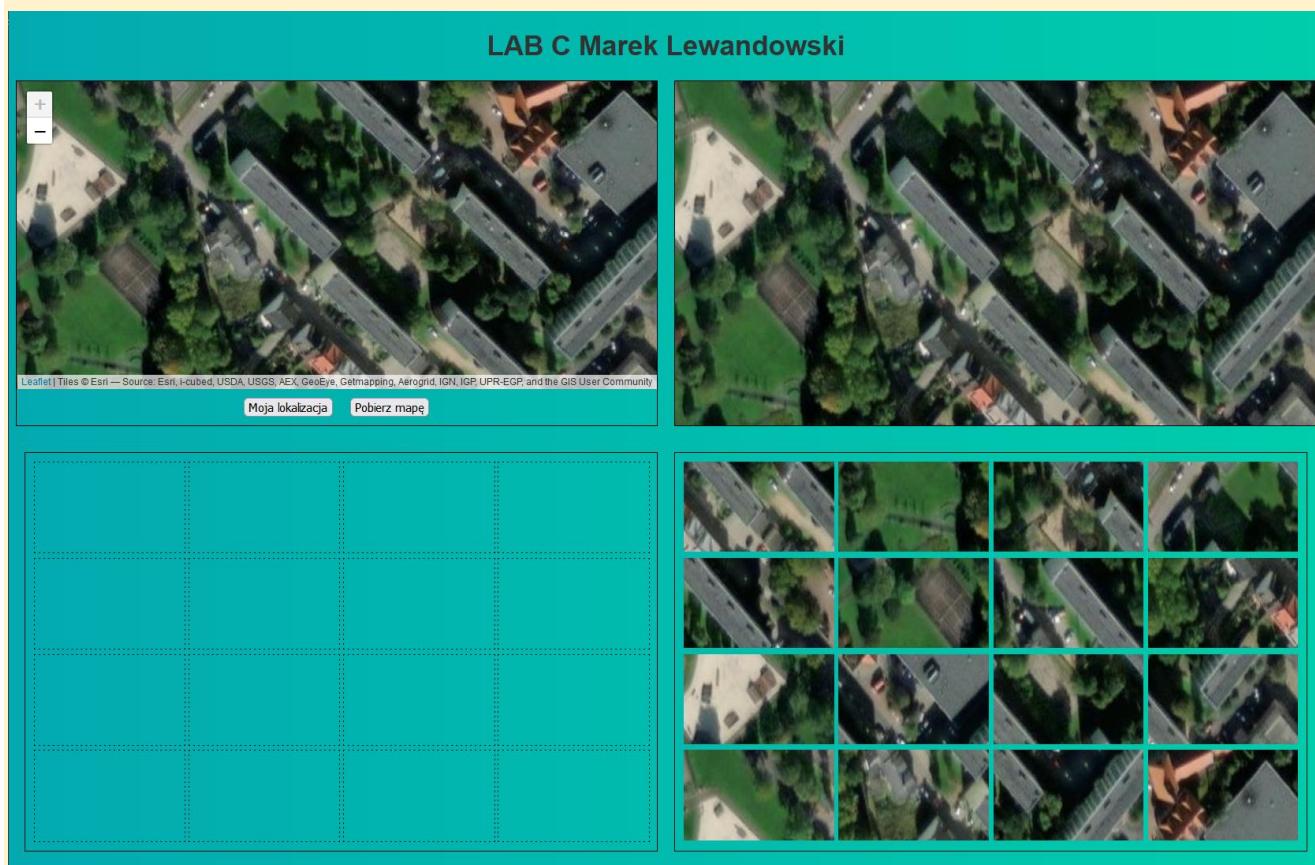
Wstaw zrzut ekranu obrazujący zapisanie ustawionego fragmentu mapy dynamicznej do rastra w canvas:



Punkty:

0	1
---	---

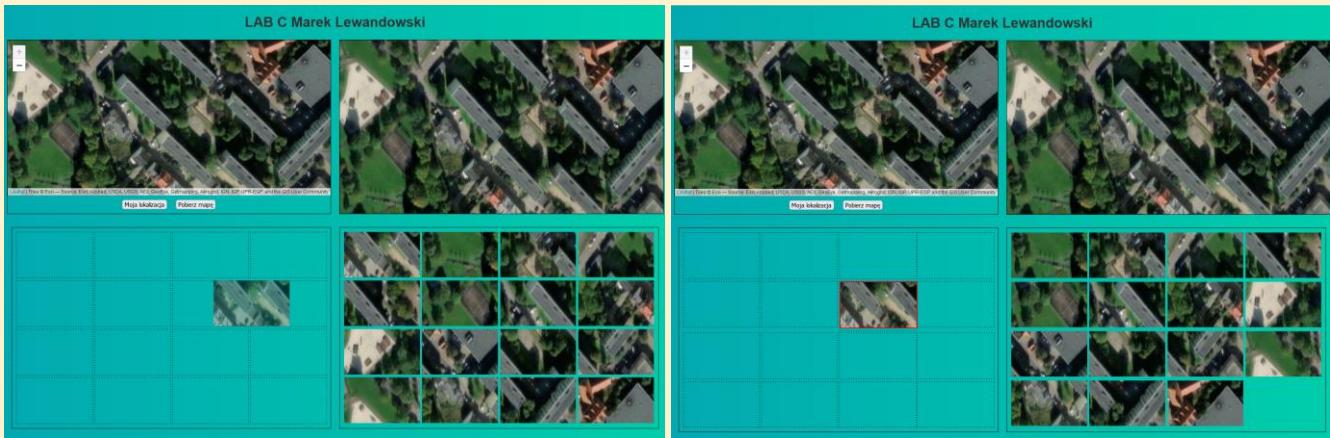
Wstaw zrzut ekranu obrazujący podział mapy rastrowej na puzzle:



Punkty:

0	1
---	---

Wstaw dwa zrzuty ekranu obrazujące działanie mechanizmu Drag-and-Drop na puzzlach:

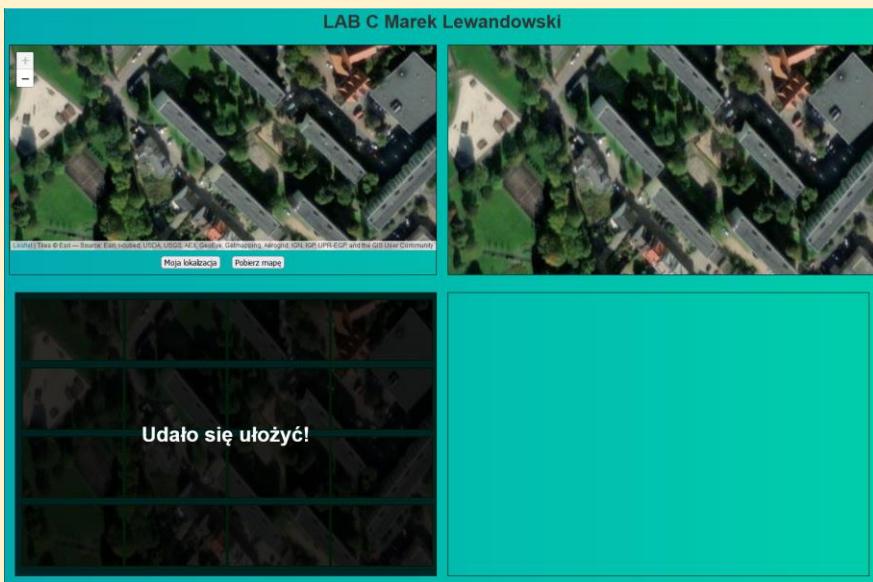


Punkty:

0

1

Wstaw zrzut ekranu obrazujący działający mechanizm wykrywania poprawnego ułożenia wszystkich puzzli. Można ograniczyć się do wydrukowania komunikatu za pomocą `console.log()`:

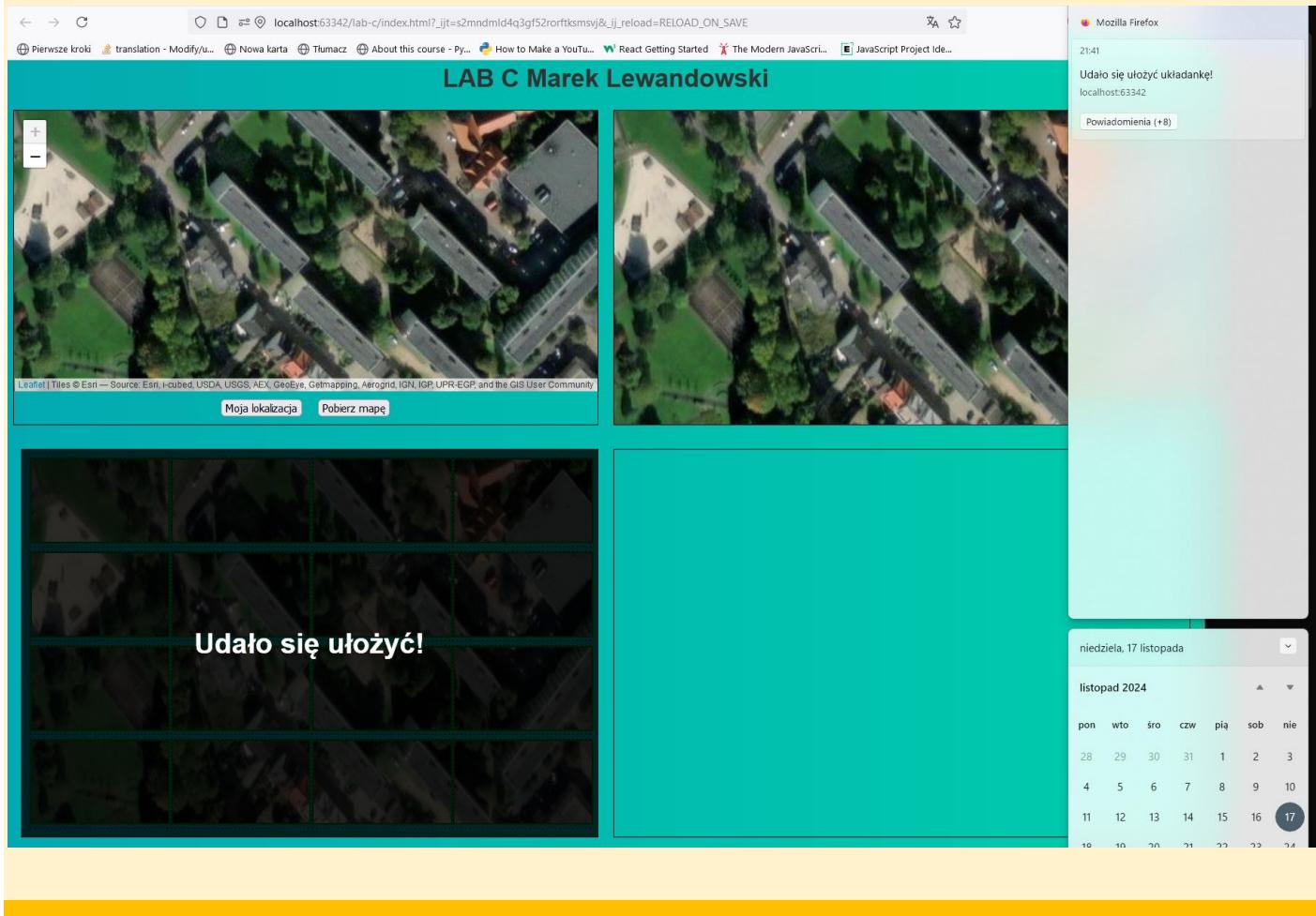


Punkty:

0

1

Wstaw zrzut ekranu obrazujący wyświetlenie notyfikacji systemowej po poprawnym ułożeniu puzzli:



Punkty:

0

1

## COMMIT PROJEKTU DO GIT

Zacommittuj 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-c na podstawie głównej gałęzi kodu.

Podaj link do brancha lab-c w swoim repozytorium:

<https://github.com/Marek-Lewandowski01/Aplikacje-Internetowe1/tree/lab-c>

## PODSUMOWANIE

W kilku zdaniach podsumuj zdobyte podczas tego laboratorium umiejętności.

Ciekawe zadanie, trochę się namęczyłem z układem 2x2 (równość kontenerów) i z tym, żeby puzzle były równej wielkości ale udało się ogarnąć temat. Wiem, że mój układ trochę różni się od przedstawionego na YT przez Pana Doktora, ale wyżej było napisane że układ może się różnić, a dla mnie przedstawienie poszczególnych elementów w taki sposób był bardziej przejrzysty i łatwiejszy do implementacji. Ogólnie zadanie bardzo fajne i miało sporą ilość wyzwań! I dziękuję Panu Doktorowi za umieszczenie przydatnych fragmentów kodu na GitHub-ie, to ułatwiło wykonanie zadania.

Zweryfikuj kompletność sprawozdania. Utwórz PDF i wyślij w terminie.