

Aplikacje Internetowe 1

Laboratorium 4: Zaawansowane JS w przeglądarce

Data laboratorium:	28.10.2022
Termin sprawozdania:	16.11.2022
Imię i nazwisko:	Szymon Buckowski

0. Wprowadzenie

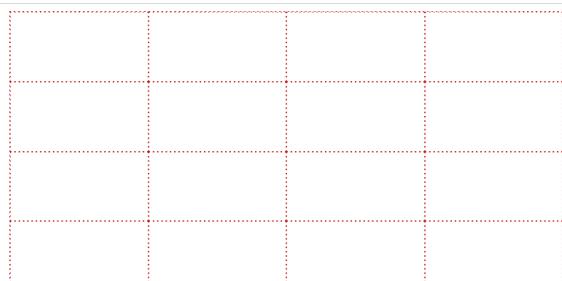
Tutaj należy umieścić 2-3 zdania opisujące zawartość sprawozdania i streszczające przebieg laboratorium. Wypełnienie tej rubryki warunkuje dalsze sprawdzenie sprawozdania.

Celem laboratorium było stworzenie dynamicznej aplikacji - puzzli, zawierającej dynamiczną mapę. Mapa zostaje eksportowana do postaci rastrowej, podzielona na 16 pomieszanych elementów i celem gracza jest ułożyć je w poprawną całość we wskazanym miejscu.

1. Strona HTML ze wszystkimi elementami

Należy przedstawić HTML ze wszystkimi wymaganymi elementami. 1 punkt.

Strona składa się z dynamicznej mapy, dwóch przycisków, ramki na mapę w postaci rastrowej oraz ramki na umiejscowienie elementów pociętej mapy (które będzie można zobaczyć niżej).



2. Dynamiczna mapa

Należy udokumentować działanie dynamicznej mapy, z możliwością zmiany lokalizacji i skali. 1 punkt.

Mapę można dowolnie przesuwać oraz przybliżać/oddalać.



[Get Current Location](#) [Save Raster Map](#)

3. Pobieranie geolokalizacji

Należy przedstawić jak pobierana jest bieżąca lokalizacja użytkownika i centrowana jest mapa. 1 punkt.

Kliknięcie przycisku *Get Current Location* wyświetla prośbę o udostępnienie informacji o położeniu i po zaakceptowaniu centruje mapę w miejscu lokalizacji użytkownika (bez zmiany skali).



[Get Current Location](#) [Save Raster Map](#)



[Get Current Location](#) [Save Raster Map](#)

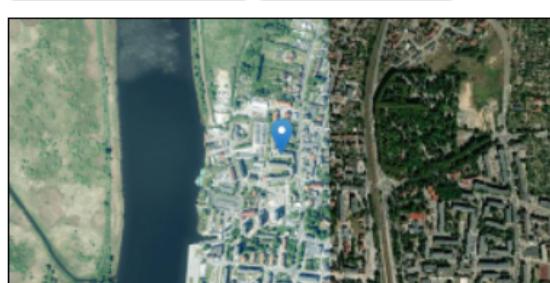
4. Pobieranie mapy rastrowej

Należy zademonstrować funkcjonalność, w której po kliknięciu przycisku, zawartość mapy dynamicznej pobrana zostanie jako statyczny obraz do dalszego wykorzystania. 1 punkt.

Kliknięcie przycisku *Save Raster Map* eksportuje zawartość mapy do postaci rastrowej i wyświetla w ramce poniżej.



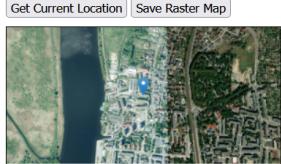
[Get Current Location](#) [Save Raster Map](#)



5. Podział mapy rastrowej na puzzle

Należy zademonstrować podział mapy rastrowej na 16 pomieszanych części. 1 punkt.

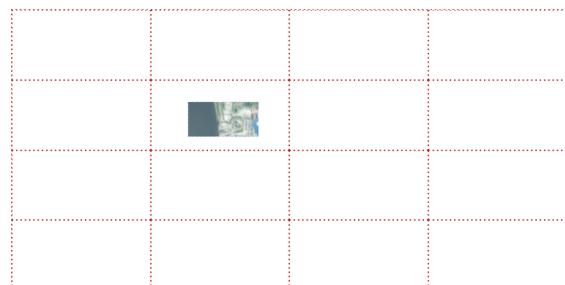
Mapa rastrowa zostaje pocięta i pomieszana na 16 kawałków, kawałki mają rozmiar odpowiadający mapie rastrowej i są ustawione w jednej linii pod spodem.

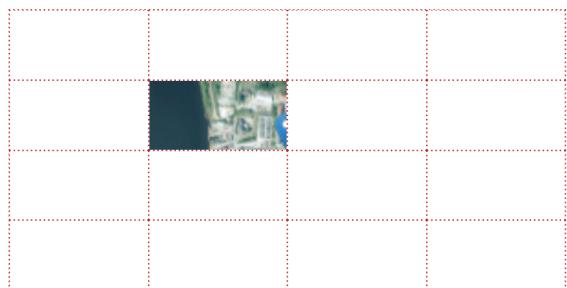
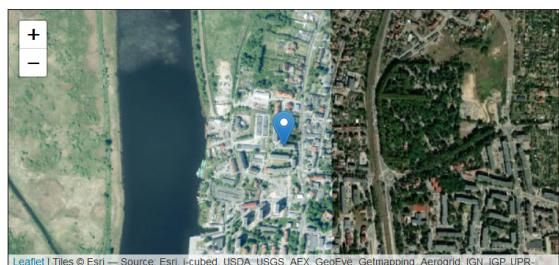


6. Drag & drop

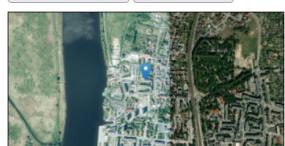
Należy udokumentować funkcjonowanie mechanizmu drag & drop do przestawiania puzzli. 1 punkt.

Kliknięcie i przytrzymanie wybranego kawałka pozwala na przesunięcie go na jedno z pól przeznaczonych do układania mapy i po puszczeniu przypina go do tego pola.





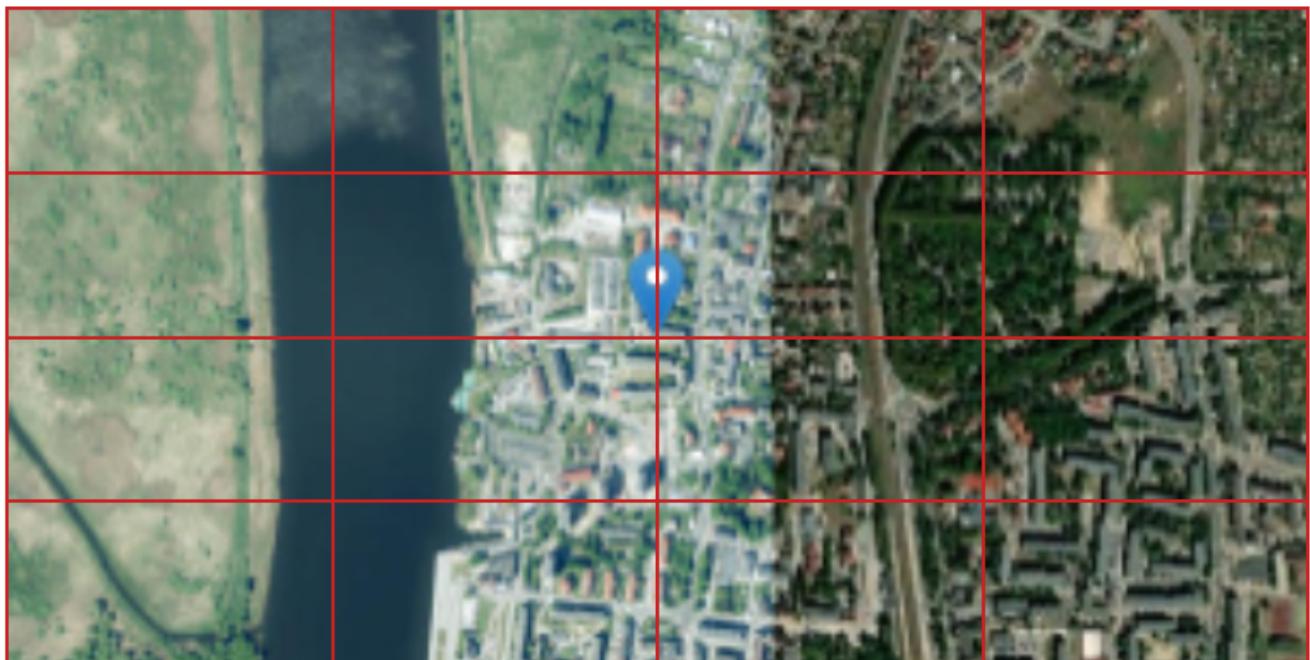
[Get Current Location](#) [Save Raster Map](#)



7. Wykrywanie dobrze ułożonej mapy

Należy udokumentować, że skrypt rozpoznaje moment, w którym mapa została poprawnie ułożona. Przykładowo z wykorzystaniem komunikatu w konsoli. 1 punkt.

Po poprawnym ułożeniu mapy obramowania pól zmieniają się w solidną linię i wyświetlany jest komunikat w konsoli.



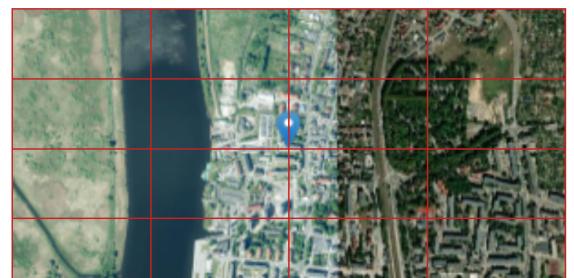
Congratulations, You Won!

»

8. Notyfikacja po dobrze ułożonej mapie

Należy udokumentować, że skrypt po poprawnym ustawieniu puzzli wyświetli notyfikację z wykorzystaniem Notification API. 1 punkt.

Oprócz komunikatu w konsoli użytkownik dostaje również notyfikację.



9. Linki i uwagi

W tej sekcji należy umieścić link do działającej aplikacji listy zadań oraz dowolne uwagi.

...