

| <b>TI</b>            | <b>Dokumentacja Projektu JavaScript</b>                    |
|----------------------|--|
| <b>Autor</b>         | Michał Czachor   |
| <b>Kierunek, rok</b> | Informatyka i ekonometria, II rok, st. Stacjonarne (3.5-l) |
| <b>Temat</b>         | Implementacja algorytmu Seam Carving                       |

|                           |   |
|---------------------------|---|
| Tematyka projektu .....   | 1 |
| Realizacja projektu.....  | 1 |
| Testowanie aplikacji..... | 2 |
| Testy.....                | 3 |
| Test optymalizacyjny..... | 3 |
| Test RAM.....             | 4 |
| Raport SEO.....           | 5 |
| Implementacja .....       | 6 |
| Źródła .....              | 6 |
| Prompty AI.....           | 6 |

## Tematyka projektu

Głównym założeniem projektu jest stworzenie aplikacji typu Seam Carving, która umożliwi użytkownikowi inteligentną zmniejszenie rozmiaru obrazu cyfrowego.

Podstawową funkcjonalnością aplikacji jest możliwość ładowania i przetwarzania obrazów w przyjaznym dla użytkownika interfejsie. Użytkownik powinien mieć możliwość łatwego załadowania dowolnego obrazu, a następnie interaktywnego zmniejszania jego wymiarów.

Dodatkowo użytkownik posiada suwak do dostosowania szerokości wyjściowego obrazu. Ponadto obraz jest na bieżąco aktualizowany wraz z czerwonym piorunem (seam), aby użytkownik widział w czasie rzeczywistym działanie algorytmu.

## Realizacja projektu

W celu realizacji założeń projektu, zdecydowano się na wykorzystanie języka JavaScript wraz z Canvas API. Do stworzenia interfejsu użytkownika aplikacji Seam Carving użyto HTML oraz CSS, co zapewniło przejrzystość i intuicyjność obsługi aplikacji. Dzięki

odpowiedniemu projektowi interfejsu użytkownik może łatwo wczytywać obrazy i interaktywnie zmieniać ich wymiary, korzystając z czytelnych suwaków i przycisków.

Proces przetwarzania obrazu realizowany jest za pomocą dwóch canvasów HTML. Pierwszy canvas, niewidoczny dla użytkownika, służy jako bufor przetwarzania, na którym algorytm faktycznie wykonuje operacje usuwania lub dodawania szwów. Drugi canvas, widoczny, pełni funkcję wyświetlacza wynikowego, który na bieżąco renderuje przetworzony obraz w odpowiedniej skali, zapewniając prawidłowe proporcje i zapobiegając wyjściu poza obszar strony.

## Testowanie aplikacji

Proces testowania aplikacji Seam Carving został zaplanowany w celu dokładnej weryfikacji jej funkcjonalności, wydajności oraz zgodności, co zapewni użytkownikom płynne i efektywne korzystanie z narzędzia.

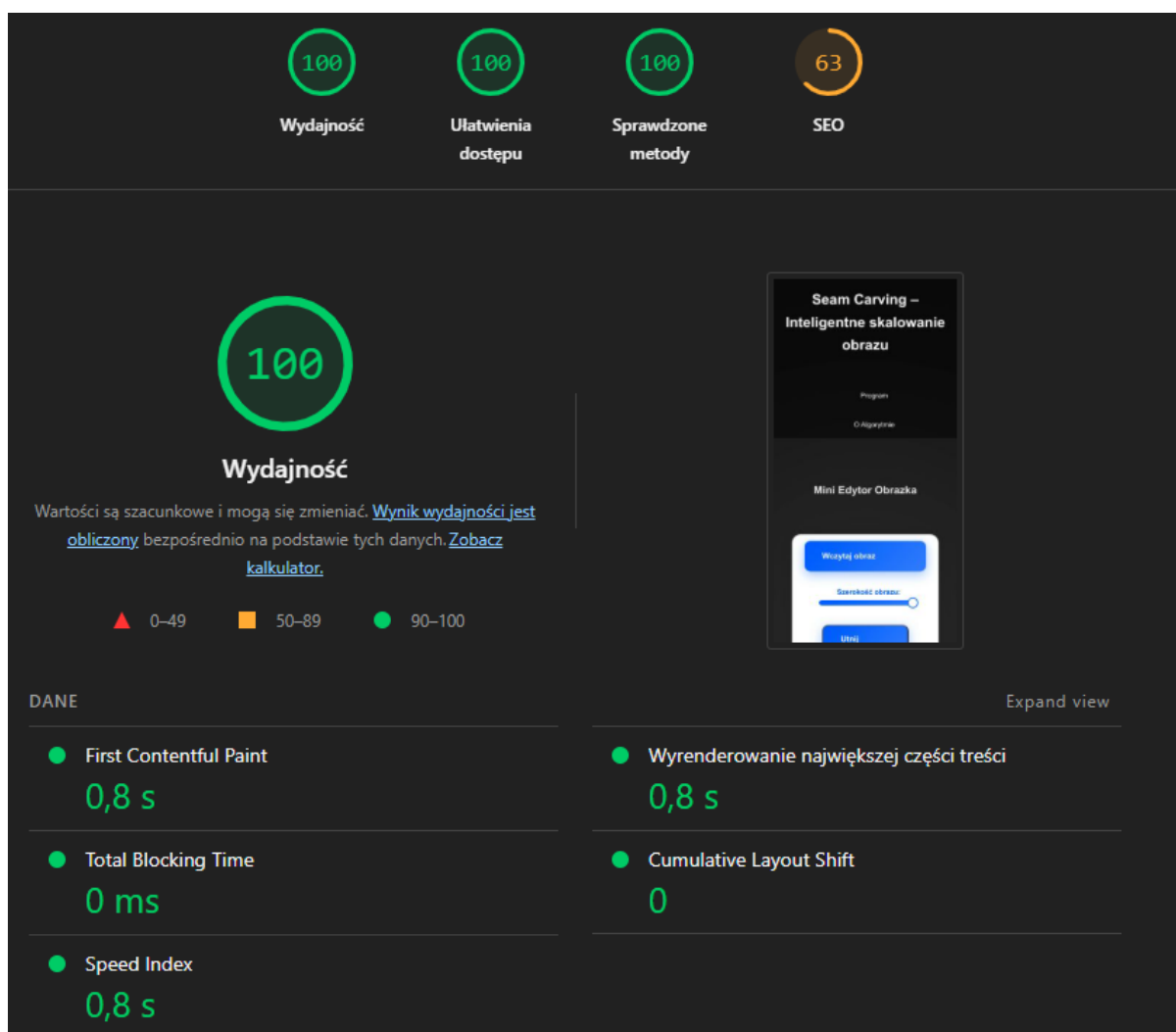
Pierwszym krokiem jest sprawdzenie dostępności i responsywności interfejsu na różnych urządzeniach. Aplikacja będzie testowana pod kątem poprawnego wyświetlania na komputerach, tabletach i smartfonach, aby upewnić się, że canvasy, przyciski i kontrolki są czytelne i funkcjonalne w każdej rozdzielczości ekranu.

Kolejnym krokiem jest optymalizacja wydajności algorytmu. W ramach tych testów zostanie przeanalizowany czas przetwarzania obrazów o różnej rozdzielczości oraz zużycie pamięci przez aplikację. Narzędzia takie jak Chrome DevTools Performance monitor oraz Memory profiler umożliwią ocenę efektywności implementacji Seam Carving i wskażą obszary wymagające optymalizacji.

Innym ważnym elementem testów jest walidacja poprawności algorytmu. Zostanie ona przeprowadzona z użyciem zestawu testowych obrazów o znanej strukturze, aby zweryfikować, czy Seam Carving prawidłowo identyfikuje i zachowuje istotne obszary.

Ostatnim, ale równie ważnym elementem testów, jest analiza kompatybilności międzyprzeglądarkowej. Zostanie ocenione działanie aplikacji w różnych przeglądarkach (Chrome, Firefox, Safari, Edge) z uwzględnieniem specyficznych implementacji Canvas API.

## Testy

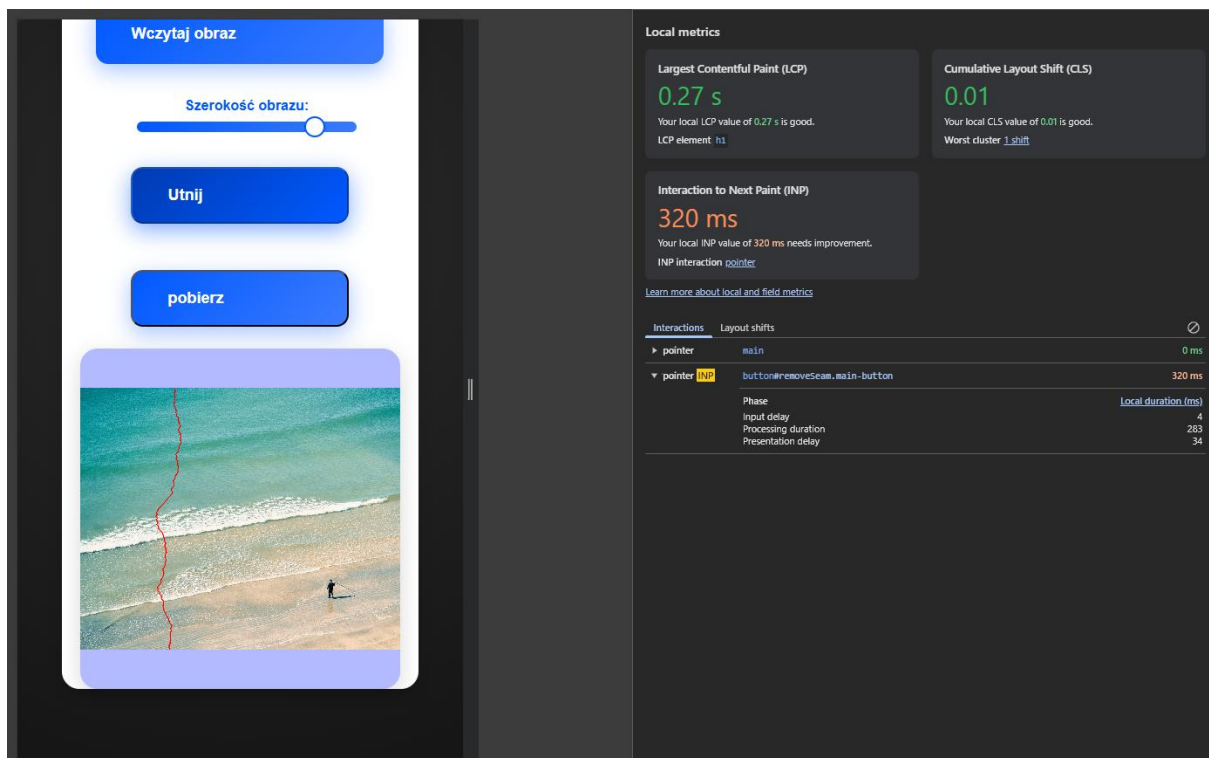


Rysunek 1. Wynik testu z narzędzia lighthouse.

Wyniki testu optymalizacji wykazują, że strona internetowa jest w dużej mierze zoptymalizowana pod względem wydajności, z uwzględnieniem kluczowych aspektów takich jak czas ładowania, minimalizacja zasobów oraz optymalizacja mediów.

Dodatkowo, testy sugerują, że strona mogłaby zyskać na dalszej optymalizacji pod kątem wyszukiwarek. Implementacja tych usprawnień poprawiłaby ogólną dostępność strony, czyniąc ją bardziej dostosowaną do potrzeb szerszej grupy użytkowników.

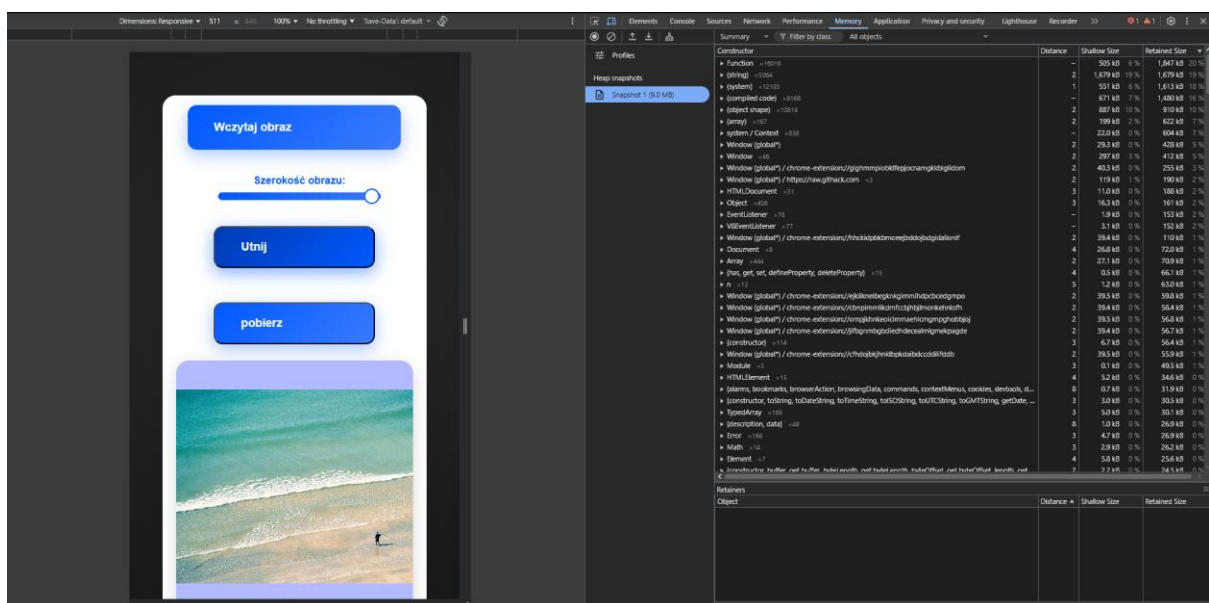
## Test optymalizacyjny



Rysunek 2. Rysunek testu optymalizacyjnego

Wyniki testu wskazują na poprawne działanie strony pod kątem odpowiadania, jedynie aspekt INP wymaga lekkiej poprawy, jednakże przycisk o którym jest mowa ma opcję blokowania po naciśnięciu do momentu zakończenia usuwania SEAM-ów, co nie powinno prowadzić do znacznych opóźnień po stronie użytkownika.

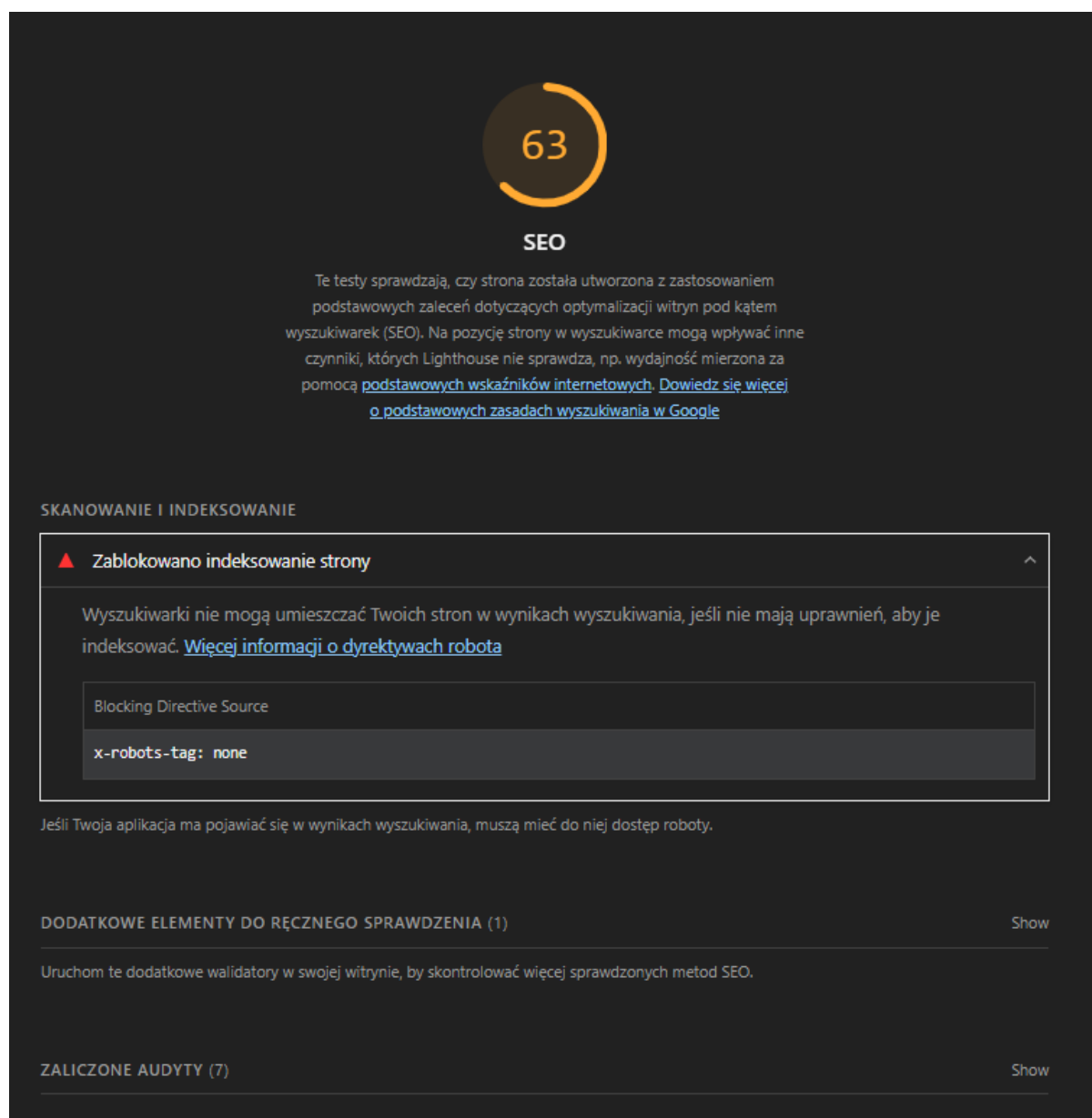
## Test RAM



Rysunek 2. Rysunek testu zajmowania pamięci RAM

Pamięć potrzeba w typowych aplikacjach z użyciem API canvas wahają się między 20-50 MB, więc aplikacja nie zużywa znacznej wielkości pamięci ram, jednakże dla wysoko rozdzielczych obrazów test wskazuje na zajęcie Pamięci RAM na poziomie 100 MB, dla takich obrazów należy w przyszłości zoptymalizować algorytm.

## Raport SEO



Rysunek 3. Raport SEO

Strona jest poprawna semantycznie i poprawnie funkcjonuje, jednakże strona nie jest przystosowana do wprowadzenia na witryny wyszukiwarek. Strona nie posiada plików wspomagających dostęp dla robotów.

## Implementacja

Seam carving to metoda inteligentnej zmiany rozmiaru obrazu poprzez usuwanie najmniej istotnych ścieżek pikseli (tzw. seams), tak aby ważne elementy obrazu pozostały nienaruszone.

Zaimplantowany algorytm potrafi odnaleźć i usunąć wyżej wymienione seam-y, ich ilość jest obliczana na podstawie suwaka w procentach, aby potem przygotować plik z rozszerzeniem .PNG, aby użytkownik mógł zachować produkt końcowy algorytmu.

```
function drawSeam(seam) {  
  ctx.strokeStyle = "red";  
  ctx.lineWidth = 4;  
  ctx.beginPath();  
  
  seam.forEach((x, y) => {  
    if (y === 0) {  
      ctx.moveTo(x, y);  
    } else {  
      ctx.lineTo(x, y);  
    }  
  });  
  
  ctx.stroke();  
}
```



Rysunek 1. Kod z implementacją funkcji drawSeam i przykładem jej wywołania.

## Źródła

<https://github.com/vivianhylee/seam-carving>

[https://en.wikipedia.org/wiki/Seam\\_carving](https://en.wikipedia.org/wiki/Seam_carving)

<https://icons8.com/icons>

## Prompty AI

Prompt:

“podaj implementację algorytmu Seam Carving w Języku JavaScript oraz wytłumacz szczegółowo funkcję jakich on używa”

Cel: Zaznajomienie ze strukturą algorytmu.

Zastosowany wynik: Częściowa implementacja algorytmu.

[https://www.ur.edu.pl/files/ur/import/private/113/Zarz%C4%85dzenie%20Rektora%20nr%20164%20\\_2023%20w%20sprawie%20wykorzystania%20w%20procesie%20kszta%C5%82cenia%20narz%C4%99dzi%20opartych%20na%20sztucznej%20inteligencji/Zarz%C4%85dzenie%20164\\_2023%20z%2029.11.2023%20r.%20-%20wykorzystywanie%20w%20procesie%20kszta%C5%82cenia%20SI.pdf](https://www.ur.edu.pl/files/ur/import/private/113/Zarz%C4%85dzenie%20Rektora%20nr%20164%20_2023%20w%20sprawie%20wykorzystania%20w%20procesie%20kszta%C5%82cenia%20narz%C4%99dzi%20opartych%20na%20sztucznej%20inteligencji/Zarz%C4%85dzenie%20164_2023%20z%2029.11.2023%20r.%20-%20wykorzystywanie%20w%20procesie%20kszta%C5%82cenia%20SI.pdf)