

# To-Do List

Projekt Technologie Internetowe

HTML + CSS

Technologie Internetowe

Jakub Jakubowski

125125, lab1

II rok informatyki stacjonarnie

26.01.2024 – v1.0

## Cel aplikacji

Projekt To-Do List ma na celu stworzenie szablonu strony do dodawania codziennych zadań do swojej listy wykorzystując HTML i CSS. Strona będzie umożliwiała użytkownikowi dodawanie, usuwanie i edycję zadań które ma wykonać w ciągu dnia bieżącego, tak aby już nigdy więcej nie zapomnieć o rzeczach ważnych czy mniej ważnych do wykonania w ciągu dnia.

Nie zawsze mamy możliwość skorzystania z komputera w celu sprawdzenia naszych codziennych zadań, dlatego celem jest również aby szablon i cała konstrukcja strony była w pełni responsywna tak aby móc korzystać zarówno z komputera jak i urządzeń mobilnych, w celu dodawania zadań.

## Problem do realizacji

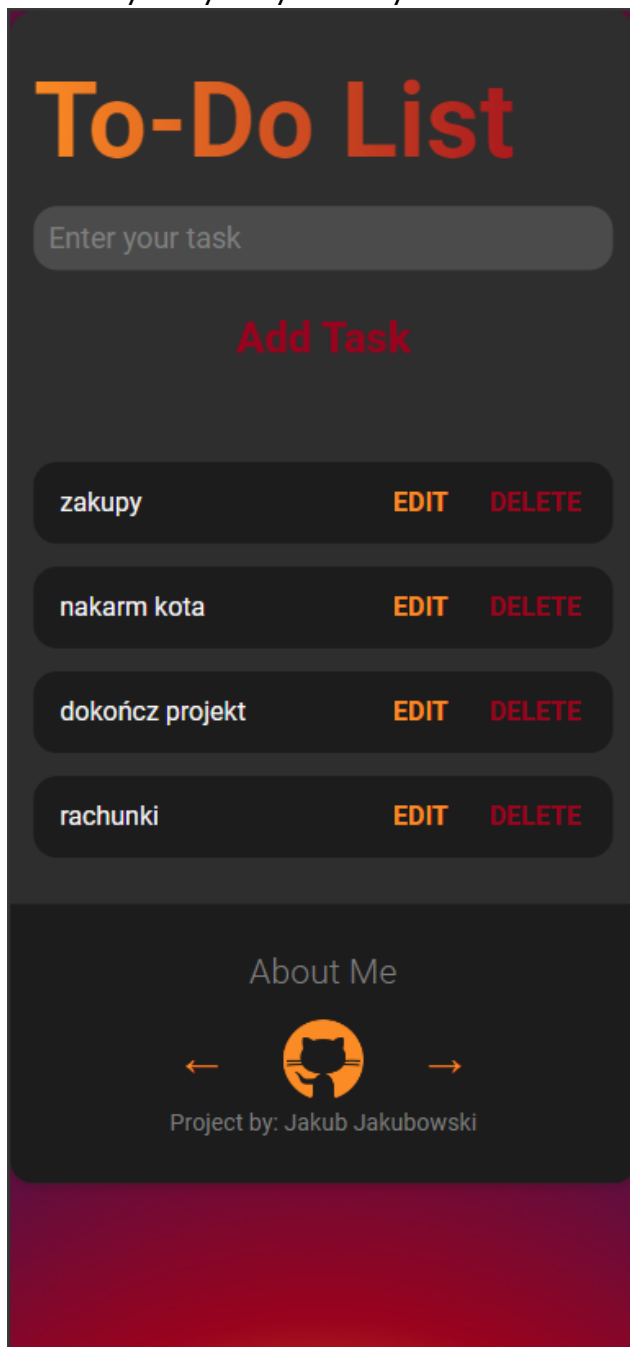
Rozwiązanie problemów poprawi użyteczność aplikacji, ułatwi korzystanie z niej na urządzeniach mobilnych i poprawi ogólną satysfakcję użytkowników. W tym celu należy rozwiązać następujące problemy:

- Przycisk „Add Task” – na urządzeniach o wąskiej szerokości ekranu, przycisk wychodzi za ekran stając się niewidoczny. Problem braku wyświetlającego się przycisku, powoduje że dodawanie zadań do listy staje się niemożliwe
- Lista zadań – zbyt duża ilość zadań (zależna od urządzenia), może powodować że niektóre z nich będą niewidoczne, bez możliwości zarządzania nimi.
- Sekcja „About me” – problem złego wyświetlania listy zadań wymusza na sekcji, wyświetlanie po za dostępnym miejscem wyświetlania strony internetowej. W ten sposób użytkownik mobilny nie widzi tej sekcji

## Sposób rozwiązania

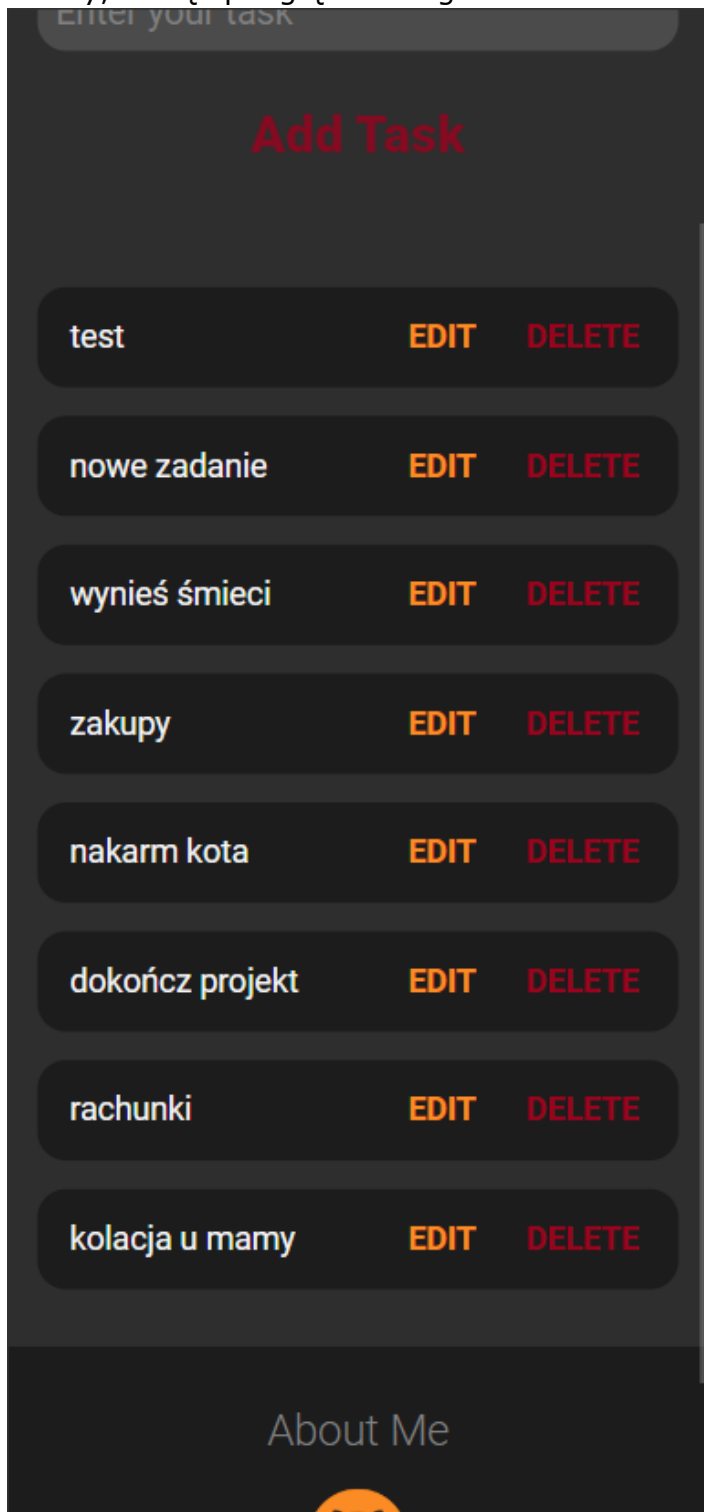
- Przycisk „Add Task” – użycie odpowiednich metod w pliku „MediaQuery.css”, dzięki którym wraz z brakiem miejsca dla przycisku Add Task, przycisk zmienia swoje położenie. W standardowym rozwiązaniu, kiedy ekran jest wystarczająco szeroki to obok pola tekstowego na wpisanie zadania znajduje się przycisk, natomiast kiedy nie ma dla niego miejsca, to przycisk wyświetla się pod polem tekstowym

Przycisk zmienił swoją pozycję, na pod polem tekstowym, tak aby był widoczny i czytelny dla użytkownika



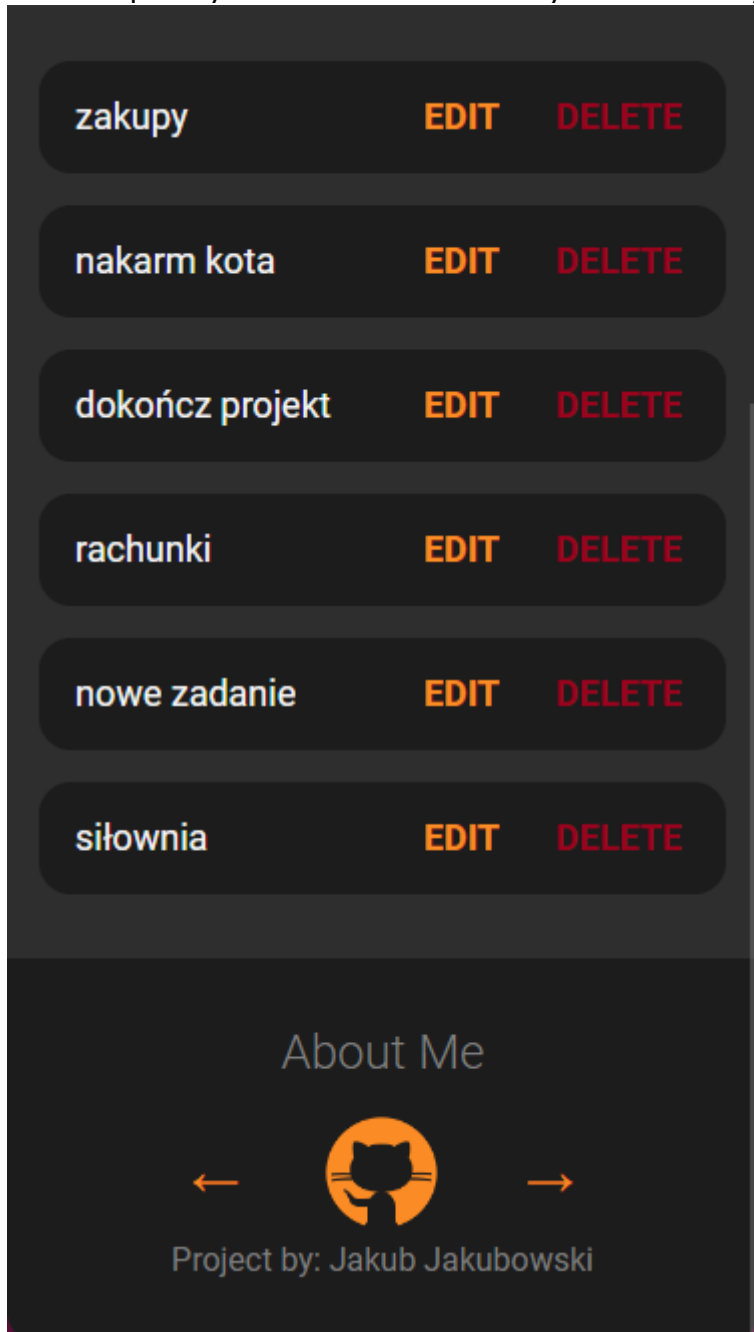
- Lista zadań – użycie metod wyświetlania takich jak „display:flex” oraz zastosowanie dodatkowych metod do pliku „MediaQuery.css” pozwoliły na uporządkowanie dynamicznie dodawanych nowych zadań. Tym samym kiedy zadań jest bardzo dużo, użytkownik przewija stronę w dół aby zobaczyć kolejne zadania, które dodał

Z prawej strony strony, wyświetla się Scrollbar, umożliwiający przewijanie strony, a więc podgląd każdego zadania.



- Sekcja „About me” – problem z niewidoczną sekcją został naprawiony wraz z dodaniem właściwości strony takich jak „display:flex”, wyświetlania w kolumnach oraz zastosowanie metod do pliku „MediaQuery.css”, w ten sposób na samym końcu listy zadań, sekcja wyświetla się poprawnie

Sekcja jest zawsze widoczna pod listą zadań, w przypadku gdy zadań jest wiele to po użyciu Scrollbar’a na samym dole strony znajduje się owa sekcja



## Sposób wykonania

Strona składa się z takich plików jak:

- index.html – w którym znajduje się szablon strony internetowej, z nagłówkiem gdzie dodajemy zadania oraz sekcji gdzie będą się te zadania wyświetlały
- styles.css – to plik który zawiera zbiór reguł, które definiują wygląd i styl strony. W nim definiujemy kolory, marginesy, czcionki czy cienie. Również obsługa animacji przez najechanie na elementy
- MediaQuery.css – zawiera szereg zasad dla zapytań media, które dostosowują wygląd strony do różnych szerokości ekranu, tak aby na każdym urządzeniu, czy to dekstopowym czy mobilnym wygląd pozostawał niezmienny, bez zaburzeń.

## Efektywność i optymalizacja

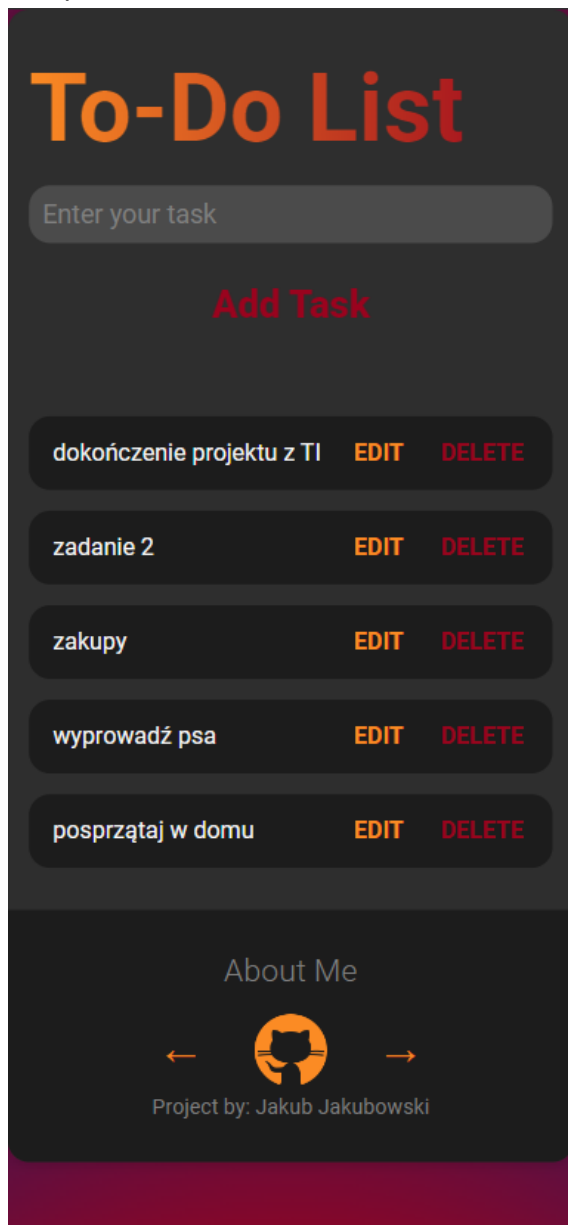
Podczas implementacji projektu skupiono się na efektywności i optymalizacji kodu, aby zapewnić płynność działania oraz zoptymalizowane zużycie zasobów systemowych. Zastosowane techniki optymalizacyjne obejmują m.in. minimalizację zbędnych operacji, efektywne zarządzanie pamięcią oraz unikanie zbędnych powtórzeń kodu.

Dodatkowo, zdecydowano się na zastosowanie podejścia responsywnego, co umożliwiło dostosowanie interfejsu projektu do różnych rozdzielczości ekranu. To ważne działanie pozwala użytkownikom korzystać z To-Do List na różnych urządzeniach, bez utraty jakości doświadczenia użytkownika. Interfejs responsywny dostarcza optymalne wrażenia zarówno na dużych ekranach komputerów, jak i na mniejszych ekranach urządzeń mobilnych.

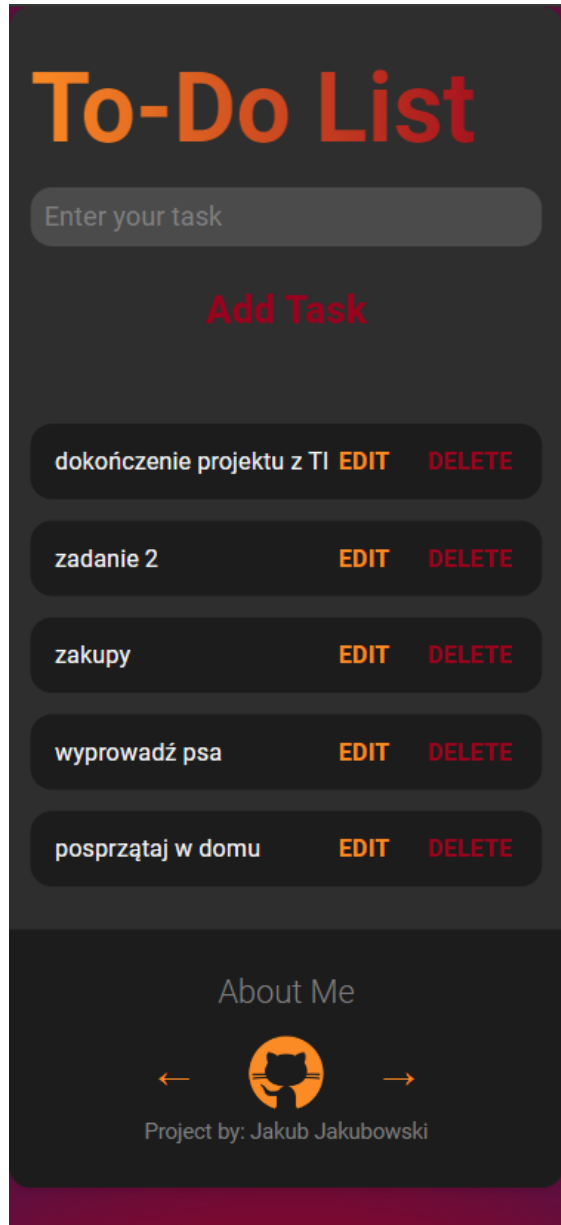
W rezultacie, połączenie dbałości o efektywność kodu i zastosowanie podejścia responsywnego sprawia, że witryna nie tylko działa płynnie, ale także dostarcza przyjemność użytkownikom na różnych platformach i urządzeniach.

## Testowanie

Urządzenie : iPhone 14 Pro Max  
A51/71



Urządzenie: Samsung Galaxy



Urządzenie: Samsung Galaxy S8+

# To-Do List

Enter your task

**Add Task**

dokończenie projektu **EDIT** **DELETE**

zadanie 2 **EDIT** **DELETE**

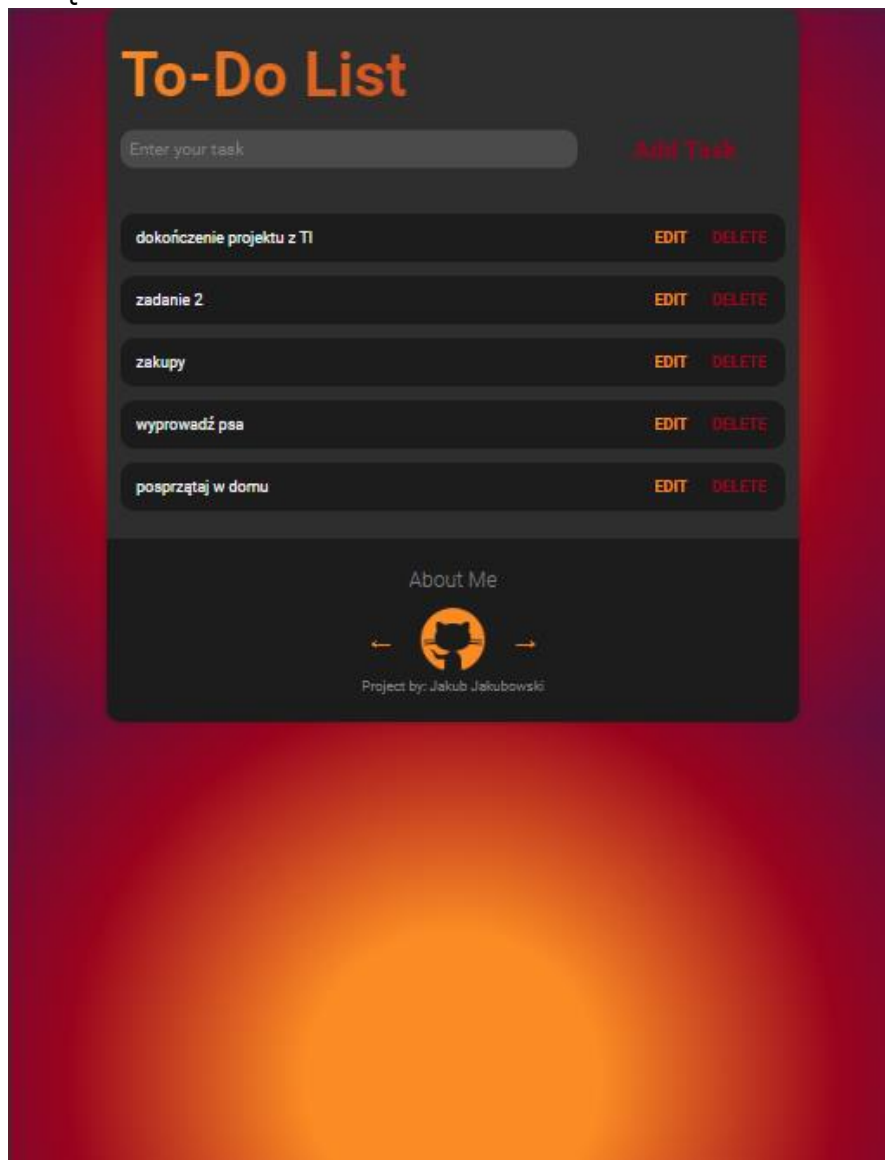
zakupy **EDIT** **DELETE**

wyprowadź psa **EDIT** **DELETE**

posprzątaj w domu **EDIT** **DELETE**



Urządzenie: iPad Pro



Testowano responsywność strony jak i jej poprawne działanie na różnych urządzeniach. Testy przebiegły pomyślnie, jak widać w przypadku za małej długości ekranu, mamy możliwość przewijania w dół ekranu w celu zobaczenia wszystkich zadań na liście.

## Dokumentacja KSS

Dokumentacja KSS(Knyle Style Sheets), która jest dokumentacją plików stylu dla witryny znajduje się pod tym [linkiem](#).

## Linki

Projekt To-Do List korzysta z źródeł zewnętrznych, dokładniej do grafiki, ikon. Wszystkie użyte zasoby są zgodne z prawem autorskim, a ich źródła są udokumentowane poniżej.

- Ikony: <https://icons8.com/icons>