

Nazwa projektu: brick-lister

Opis projektu: program do listowania części wraz ze zdjęciami dla budowanych zestawów z klocków, wyliczania kosztów oraz generowania listy części do pliku PDF

Wstęp

Projekt web-aplikacji `Brick lister` bazuje na systemie sprzedażowym lego.com Pick a Brick oraz BrickLink.com i jest połączeniem obu jako przykład wymagań stojących przed deweloperami dzisiejszych aplikacji internetowych, którzy muszą sprostać oczekiwaniom klienta.

Funkcjonalności programu

- menu programu: Zestawy (Sets), Listy części (parts), Pomoc (faq)
- importowanie części z pliku CSV (katalog z plikami CSV)
- listowanie części z wyświetlaniem: obrazka, nazwy, numeru, ilości, ceny (szt), wartość
- zapisywanie listy części jako plik CSV oraz jako SET (np. w WebStorage lub BD)
- eksportowanie listy części do pliku CSV (z podaniem nazwy)
- lista skompletowanych zestawów zbudowana w postaci kafelek z numerem zestawu, obrazkiem oraz wskazaniem pliku CSV
- dla zestawu można wyświetlić link do strony lub do pliku PDF z instrukcją budowy
- strona pomocy złożona ze statycznych danych teleadresowych oraz formularza kontaktowego wysyłającego informację na adres e-mail

Projekt UI aplikacji


a) Wizualizacja strony głównej:

Komponent: Menu	
Zestawy (sets) Części (parts) Pomoc (help)	
Routing	Wyświetlanie zawartości podstron

Zadania:

- stworzyć komponent `Menu` z odnośnikami statycznymi oraz z Routing`iem aplikacji,
- stworzyć komponent **strony** (page) `Intro`
- załączyć oba komponenty w głównym komponencie aplikacji `App`

b) Wizualizacja podstrony Zestawy (Sets)

Komponent: Menu			
Zestawy (sets) Części (parts) Pomoc (help)			
Strona: Sets			
	6263 Lista części Instrukcja	6265 Lista części	6266 Instrukcja
	6267 Lista części Instrukcja	 6277 Lista części	

Zadania:

- stworzyć komponent **strony** (page) **`Sets`**
- lista skompletowanych zestawów zbudowana w postaci kafelek na podstawie plików z katalogu ``media`` (Dane w ZIP do pobrania z panelu listy materiałów),
- wyświetlanie zestawów w postaci kafelek z informacjami:
numerem zestawu, obraz zestawu (plik JPG lub PNG), odnośnik do podstrony Parts ładujący pliku CSV, odnośnik do instrukcji (jeżeli istnieje plik PDF),
- dla zestawu można wyświetlić link, z instrukcją budowy, do strony lub do pliku PDF.

Listowanie zawartości katalogu:

- odnieść się do tematu #8 ``Odczyt i zapis do pliku`` z zajęć z Node.js,
- listując katalog ``media`` należy odczytać każdy rodzaj pliku (obraz, instrukcja PDF, lista części CSV) i sprawdzić jakie numery zestawów są dostępne łącząc pliki w zestawy, np. pliki: ``6541_parts.csv`` i ``6541_preview.jpg`` tworzą jeden kafelek z zestawem.

c) Wizualizacja podstrony Części (Parts)

Komponent: Menu				
Zestawy (sets) Części (parts) Pomoc (help)				
Strona: Lista			Pobierz CSV	
Zestaw: 6277			Generuj PDF	
<i>Obraz</i>	<i>Nazwa / Numer</i>	<i>Ilość</i>	<i>Cena</i>	<i>Wartość</i>
[obraz]	Nazwa klocka Numer klocka	(- [1] +)	0,12 PLN	Ilość * Cena PLN

Zadania:

- stworzyć komponent **strongy** (page) ``Parts``
- stworzyć komponent **szablonu** (layout) ``RegularList``
- stworzyć komponent **szablonu** (layout) ``ListItem``
- zaimplementować funkcjonalności dla tej podstrony.

Funkcjonalności:

- odczytywanie z adresu URL identyfikatora zestawu i wyświetlenie go w **Węźle**,
- odczytywanie (jeśli istnieje) pliku CSV z listą części tego zestawu (przeanalizuj plik CSV oraz strukturę wpisu tabeli – `ListItem`, kolor fioletowy ramki),
- obsługa, przy użyciu **React Hooks**, zliczania Wartości pojedynczego klocka (pole `Ilość * Cena`, z dopisaniem waluty),
- pobieranie z tabeli listy części w formacie CSV (pakiet np.: ``react-csv-downloader``),
- zapisywanie listy części w WebStorage (ochrona przed zamknięciem przeglądarki),
- (inne zajęcia) generowanie z tabeli dokumentu PDF.

d) Wizualizacja podstrony Pomoc (Help)

Komponent: Menu	
Zestawy (sets) Części (parts) Pomoc (help)	
Strona: Pomoc (zmień układ)	
Tematy pomocy: <ol style="list-style-type: none">Pytanie-1Pytanie-2Pytanie-N	Wyślij e-mail do autora: Autor: [____] Tytuł: [____] Treść: [____] (Wyślij)

Zadania:

- stworzyć komponent **strongy** (page) **`Help`**
- stworzyć komponent **szablonu** (layout) **`SplitScreen`**
- stworzyć komponent **strongy** (page) **`Faq`**
- stworzyć komponent **strongy** (page) **`Form`**
- wskazać komponenty **`Faq`** oraz **`Form`** do załadowania w komponencie szablonu **`SplitScreen`**
- ustawić Properties dla komponentu **`SplitScreen`**, by można było kontrolować układ (pionowy/poziomy) oraz szerokość poszczególnych kolumn
- dodać przycisk zmiany układu wyświetlania podstrony dla komponentu **`SplitScreen`** (pionowy/poziomy)
- zaimplementować funkcjonalności dla tej podstrony.

Funkcjonalności FAQ:

- odczyt listy pytań z (wybierz: plik JSON, katalog z plikami pomocy, BD)
- odczyt wybranego pytania i jego odpowiedzi

Funkcjonalności Form:

- obsługa formularza pomocy przy użyciu React Hooks,
- wysyłka danych zebranych z formularza do Bazy danych SQLite lub MySQL