

Завдання:

Імплементувати дизайн у робочий Vue.js веб-додаток

Дизайн:

https://drive.google.com/drive/folders/16_IV-3IyMMjhdi65tE2BoPr--Ugqv4GR?usp=sharing

Виконання:

Приготування проекту і старт роботи

1. Створити Vue.js проект за допомогою терміналу:

<https://cli.vuejs.org/guide/creating-a-project.html>

```
vue create hello-world
```

sh

2. Очистити проект від демонстраційного коду
3. Додати в проект [Bootstrap-Vue](#)

```
# With npm
npm install vue bootstrap bootstrap-vue

# With yarn
yarn add vue bootstrap bootstrap-vue
```

Then, register BootstrapVue in your app entry point (typically `app.js` or `main.js`):

```
import Vue from 'vue'
import { BootstrapVue, IconsPlugin } from 'bootstrap-vue'

// Import Bootstrap and BootstrapVue CSS files (order is important)
import 'bootstrap/dist/css/bootstrap.css'
import 'bootstrap-vue/dist/bootstrap-vue.css'

// Make BootstrapVue available throughout your project
Vue.use(BootstrapVue)
// Optionally install the BootstrapVue icon components plugin
Vue.use(IconsPlugin)
```

4. Створити необхідні роути (routes) у **Vue Router** та компоненти (views) для цих роутів
 - **Home**
 - **Catalog**
 - **Product**
 - **Cart**
 - **Delivery (доставка)**
 - **Payments (оплата)**
5. Створити компоненти для хедера та футер і включити їх у **App.vue**

Головна сторінка

1. Створити компонент для каруселі ([vue-carousel](#)) та підключити його в **Home.vue**. Весь слайд можна еспортувати зображенням.
2. Отримати 4 будь-які товари за допомогою **fetch** та <https://fakestoreapi.com/docs>
3. Створити компонент для картки товару у списку та за допомогою директиви-циклу **v-for** відобразити отримані товари в п.7 на головній сторінці **Home.vue**

```
1
2 <div class="col-3" v-for="p in products" :key="p.id">
3   <productItem :product="p"/>
4 </div>
5
```

Сторінка каталогу

1. Роут каталогу буде мати вхідний параметр **:category**

```
{
  path: '/catalog/:category',
  name: 'Catalog',
  component: Catalog
}
```

2. При завантаженні сторінки зчитуємо параметр **:category**

```
var category = this.$route.params.category;
```

3. Також при завантаженні сторінки отримуємо список усіх доступних категорій на <https://fakestoreapi.com/docs> та відображаємо ці категорії у вигляді списку із **<router-link>** за допомогою директиви-циклу **v-for** у сайдбарі сторінки згідно дизайну. Кожен **<router-link>** це посилання на цю ж сторінку але з іншим параметром **:category**

Get all categories

```
fetch('https://fakestoreapi.com/products/categories')
  .then(res=>res.json())
  .then(json=>console.log(json))
```

4. Також, паралельно з отриманням категорій отримуємо список товарів відповідної категорії із <https://fakestoreapi.com/docs> , згідно зі зчитаним параметром **:category** з посилання.

Get products in a specific category

```
fetch('https://fakestoreapi.com/products/category/jewelery')
  .then(res=>res.json())
  .then(json=>console.log(json))
```



You can also use `limit(Number)` and `sort(asc|desc)` as a query string to get your ideal results

Show output

Сторінка товару

1. При завантаженні сторінки зчитуємо параметр роуту **:id** та згідно цього значення отримуємо інформацію про товар від <https://fakestoreapi.com/docs>

Get a single product

```
fetch('https://fakestoreapi.com/products/1')
  .then(res=>res.json())
  .then(json=>console.log(json))
```

Show output

2. Після отримання інформації про товар, отримуємо ще 4 товари цієї ж категорії та відображаємо згідно дизайну. Категорії поточного товару можна отримати за ключем "**category**" з об'єкту отриманого товару.

```
{
  id:8,
  title:'...',
  price:'...',
  category:'jewelery',
  description:'...',
  image:'...'
}
```

Додавання товару до кошика

1. Кошик містить айді одного товару
2. При додавання кожного наступного товару, попереднє значення перезаписується
3. Після кожного додавання товару до кошика слід показати Модальне вікно згідно дизайну. Компонент модального вікна потрібно взяти з [Bootstrap-Vue](#)

Код компоненту модального вікна:

```
3
9  <b-modal id="item-added" hide-footer centered hide-header>
0    //Код вмісту модального вікна
1  </b-modal>
2
```

Функція для відкривання вікна з методу:

```
this.$bvModal.show('item-added');
```

Функція для **ЗАКРИВАННЯ** вікна з методу:

```
this.$bvModal.hide('item-added');
```

Сторінка кошику

1. При завантаженні сторінки слід перевірити чи є дані про доданий товар в window.localStorage
2. Якщо товар є - отримати його по ID і відобразити згідно дизайну + завантажити ще 4 товари тієї ж категорії за таким же принципом як на сторінці товару
3. Якщо товару немає - показати відповідне повідомлення згідно дизайну.

Як відображати різний вміст відповідно до умов за допомогою директиви **v-if**:

```
<div v-if="cartHasItems">
  //Код вмісту, якщо корзина містить товар
</div>
<div v-if="!cartHasItems">
  //Код вмісту, якщо корзина пуста
</div>
```