ETML - C294 - GCR 01/03/2025

# Vue Gadget Shop - step7-solution



# Création du composant GadgetDisplay.js

Nous allons maintenant réorganiser le code en créant un composant Vue pour afficher chaque gadget proprement.

#### ♦ 1. Création du fichier GadgetDisplay.js

Dans le dossier /components, crée un fichier GadgetDisplay.js :

/components/GadgetDisplay.js

```
app.component("gadget-display", {
 props: ["gadget"], // Le gadget est passé en tant que prop depuis le parent
 template: /*html*/
         <div class="gadget">
            <img v-bind:src="gadget.image" alt="Gadget">
            <h2>{{ gadget.name }}</h2>
            Prix : {{ gadget.price }}€
            !gadget.inStock}">
                {{ gadget.inStock ? '✓ En stock' : '✗ En rupture de stock' }}
            <button @click="addToCart" :disabled="!gadget.inStock">Ajouter au
panier</button>
        </div>
 methods: {
   addToCart() {
     this.$emit("add-to-cart", this.gadget); // Émet un événement pour ajouter au
panier
   },
 },
});
```

#### 

- props: ['gadget'] → Réceptionne les données du gadget envoyées par le parent.
- template: /html/ → Définit le code HTML du gadget.
- @click="addToCart" → Lorsqu'on clique, la fonction addToCart() est appelée.
- this.\$emit('add-to-cart', this.gadget) → Émet un événement vers le parent pour ajouter l'article au panier.

### ♦ 2. Mise à jour de main.js pour utiliser le composant

Ajoute l'import du composant dans main.js :

ETML - C294 - GCR 01/03/2025



```
const app = Vue.createApp({
  data() {
    return {
      title: "Vue Gadget Shop",
      description:
        "Découvrez les derniers gadgets électroniques de haute technologie !",
      gadgets: [
        {
          id: 1,
          name: "Smartphone XZ",
          price: 799,
          image: "./assets/phone.jpg",
         inStock: true,
        },
        {
          id: 2,
          name: "Laptop Pro",
          price: 1299,
          image: "./assets/laptop.jpg",
          inStock: false,
        },
        {
          id: 3,
          name: "Écouteurs Bluetooth",
          price: 199,
          image: "./assets/earbuds.jpg",
          inStock: true,
        },
      cart: [], // Panier initialisé comme un tableau vide
   };
 },
  computed: {
   cartTotal() {
      return this.cart.reduce((total, item) => total + item.price, 0);
   },
    discountedTotal() {
      return this.cartTotal >= 1000 ? this.cartTotal * 0.9 : this.cartTotal;
   },
 },
 methods: {
    addToCart(gadget) {
     this.cart.push(gadget); // Ajoute l'article au panier
    removeFromCart(index) {
     this.cart.splice(index, 1); // Supprime un article du panier par son index
   },
 },
});
```

ETML - C294 - GCR 01/03/2025

#### ♦ 3. Mise à jour de index.html pour utiliser le composant

Modifie la liste des gadgets en utilisant le composant :

index.html

```
<!DOCTYPE html>
<html lang="fr">
 <head>
   <meta charset="UTF-8" />
   <meta name="viewport" content="width=device-width, initial-scale=1.0" />
   <title>Vue Gadget Shop</title>
   <script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
   <link rel="stylesheet" href="style.css" />
 </head>
 <body>
   <div id="app">
     <h1>{{ title }}</h1>
     {{ description }}
     <div class="cart">
        0">
        {{ item.name }} - {{ item.price }}€
          <button class="remove-btn" @click="removeFromCart(index)">
            X
          </button>
        Votre panier est vide.
       <!-- Affichage du total -->
       <h3 v-if="cart.length > 0">

   Total : <span v-if="cartTotal < 1000">{{ cartTotal }}€</span>

        <span v-else>
          <s>{{ cartTotal }}€</s> →
          <strong>{{ discountedTotal }}€ (-10%)</strong>
       </h3>
     </div>
     <div class="gadget-container">
       <gadget-display</pre>
        v-for="gadget in gadgets"
        :key="gadget.id"
        :gadget="gadget"
        @add-to-cart="addToCart"
       </gadget-display>
     </div>
   </div>
   <script src="./main.js"></script>
```

ETML - C294 - GCR 01/03/2025

## 

- $\bullet \quad \text{v-for="gadget in gadgets"} \rightarrow \text{It\`ere sur chaque gadget}.$
- :gadget="gadget" → Passe les données du gadget au composant via props.
- @add-to-cart="addToCart" → Écoute l'événement \$emit venant du composant enfant et appelle addToCart().