

Tienda Virtual Etapa 2

REQUERIMIENTOS RF-01, RF-03 Y RF-05
JOSE MARÍA ESTRADA ZEPEDA

DOCENTE | María De Jesús Sánchez Solís

Contenido

Requerimiento RF-01: Página de Detalle de Producto	2
Código relevante:	2
Requerimiento RF-03: Abrir la Página de Detalles en la Misma Pestaña.....	3
Código relevante:	3
Requerimiento RF-05: Sección de Cantidad de Productos.....	4
Código relevante:	4
Conclusión.....	5

Requerimiento RF-01: Página de Detalle de Producto

Descripción: Crear una página para mostrar el detalle de un producto, con un diseño profesional y acorde al tema de la API utilizada.

Solución: Se creó un archivo HTML (detalles.html) con la estructura para mostrar todos los detalles del producto, incluyendo nombre, imagen, categoría, descripción, y precio. Se utilizó Bootstrap para lograr un diseño atractivo y consistente con el tema del juego de cartas Digimon TCG.

Código relevante:

```
<!-- detalles.html -->
<div class="container">
  <div class="row">
    <div class="col-md-6">
      <div class="card shadow-lg p-3 mb-5 bg-body-tertiary rounded" style="background: linear-gradient(135deg, #ffffff, #f5f5f5);">
        
      </div>
    </div>
    <div class="col-md-6">
      <div class="detalles-producto bg-light p-4 rounded" style="box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);">
        <h2 class="display-5 text-primary">{{ producto.name }}</h2>
        <p><strong>Categoría:</strong> Nivel {{ producto.level }}</p>
        <p><strong>Tipo:</strong> {{ producto.type }}</p>
        <p><strong>ID:</strong> {{ producto.id || 'Desconocido' }}</p>
        <p><strong>Descripción:</strong> {{ producto.description || 'No disponible' }}</p>
        <p><strong>Precio:</strong> ${{ producto.price }}</p>
      </div>
    </div>
  </div>
</div>
```

En este fragmento se utiliza Vue.js para mostrar dinámicamente la información del producto que se obtiene del almacenamiento local del navegador.

Requerimiento RF-03: Abrir la Página de Detalles en la Misma Pestaña

Descripción: Al dar clic en la imagen de un producto en el catálogo, se debe abrir la página de detalle en la misma pestaña.

Solución: En la página de inicio (index.html), se implementó un botón "Ver Detalles" que redirige a la página detalles.html. Al hacer clic en la imagen del producto, se guarda la información del producto seleccionado en el almacenamiento local y se redirige al archivo detalles.html.

Código relevante:

```
<!-- index.html -->
<div class="card" @click="viewDetails(card)">
  
  <div class="card-body text-center">
    <h5 class="card-title">{{ card.name }}</h5>
    <p class="card-text">Nivel: {{ card.level }}</p>
    <button class="btn btn-primary" @click.stop="redirectToDetails(card)">Ver Detalles</button>
  </div>
</div>

// app.js
methods: {
  viewDetails(card) {
    // Asignar tipo aleatorio si aún no está asignado
    if (!card.type || card.type === "Desconocido") {
      const tipos = ["Agua", "Fuego", "Tierra", "Aire", "Luz", "Oscuridad"];
      card.type = tipos[Math.floor(Math.random() * tipos.length)];
    }
    localStorage.setItem('selectedProduct', JSON.stringify(card));
    window.location.href = 'detalles.html';
  }
}
```

La función redirectToDetails guarda los datos del producto en el almacenamiento local del navegador y redirige a la página detalles.html.

Requerimiento RF-05: Sección de Cantidad de Productos

Descripción: En la página de detalle del producto, se debe incluir una sección para definir la cantidad de artículos a comprar, con un mínimo de 1 y un máximo de 6.

Solución: En la página de detalles del producto (detalles.html), se agregó una sección de cantidad con botones de incremento y decremento. Se utilizan métodos de Vue.js para controlar la cantidad seleccionada.

Código relevante:

```
html
<!-- detalles.html -->
<div class="d-flex align-items-center my-4">
  <button class="btn btn-outline-secondary" @click="disminuirCantidad" :disabled="cantidad === 1"></button>
  <input type="text" class="form-control text-center mx-2" :value="cantidad" readonly style="width: 50px;">
  <button class="btn btn-outline-secondary" @click="aumentarCantidad" :disabled="cantidad === 6">+</button>
</div>
```

```
// app.js
methods: {
  disminuirCantidad() {
    if (this.cantidad > 1) {
      this.cantidad--;
    }
  },
  aumentarCantidad() {
    if (this.cantidad < 6) {
      this.cantidad++;
    }
  }
}
```

Este fragmento permite controlar la cantidad de productos a comprar, asegurando que el valor se encuentre entre 1 y 6, como lo requiere el RF-05.

Conclusión

- **RF-01** se resolvió creando una página (detalles.html) que muestra toda la información relevante del producto, con un diseño profesional utilizando Bootstrap.
- **RF-03** se logró redirigiendo a la página de detalles al hacer clic en la imagen del producto, utilizando el almacenamiento local para mantener la información del producto seleccionado.
- **RF-05** se implementó agregando una sección de control de cantidad de productos, utilizando botones para limitar entre 1 y 6 unidades.