**Lab - Usar formularios para la entrada del usuario**

Este laboratorio continuo los labs anteriores, y se mejorara el proyecto a través de la adición de una función de pago basada en formularios para recopilar información del usuario como parte del pago.

**Definir el modelo de formulario de pago**

Este paso le muestra cómo configurar el modelo de formulario de pago en la clase de componente. El modelo de formulario determina el estado del formulario.

1. Abrir cart.component.ts.
2. Importar el servicio [FormBuilder](https://angular.io/api/forms/FormBuilder) del paquete @angular/forms. Este servicio proporciona métodos convenientes para generar controles.

**cart.component.ts**

import { [Component](https://angular.io/api/core/Component) } from '@angular/core';

import { [FormBuilder](https://angular.io/api/forms/FormBuilder) } from '@angular/forms';

import { CartService } from '../cart.service';

1. Inyecte el servicio FormBuilder en el constructor CartComponent (). Este servicio es parte del módulo ReactiveFormsModule, que ya ha importado.

***cart.component.ts***

export class CartComponent {

constructor(

private cartService: CartService,

private formBuilder: [FormBuilder](https://angular.io/api/forms/FormBuilder),

) {}

}

1. Para recopilar el nombre y la dirección del usuario, use el método group () de FormBuilder para establecer la propiedad checkoutForm en un modelo de formulario que contenga campos de nombre y dirección.

***cart.component.ts***

export class CartComponent {

items = this.cartService.getItems();

checkoutForm = this.formBuilder.group({

name: '',

address: ''

});

constructor(

private cartService: CartService,

private formBuilder: [FormBuilder](https://angular.io/api/forms/FormBuilder),

) {}

}

5. Defina un método onSubmit () para procesar el formulario. Este método permite a los usuarios enviar su nombre y dirección. Además, este método utiliza el método clearCart () de CartService para restablecer el formulario y borrar el carrito.

Toda la clase de componentes del carrito es la siguiente:

**cart.component.ts**

import { [Component](https://angular.io/api/core/Component) } from '@angular/core';

import { [FormBuilder](https://angular.io/api/forms/FormBuilder) } from '@angular/forms';

import { CartService } from '../cart.service';

@[Component](https://angular.io/api/core/Component)({

selector: 'app-cart',

templateUrl: './cart.component.html',

styleUrls: ['./cart.component.css']

})

export class CartComponent {

items = this.cartService.getItems();

checkoutForm = this.formBuilder.group({

name: '',

address: ''

});

constructor(

private cartService: CartService,

private formBuilder: [FormBuilder](https://angular.io/api/forms/FormBuilder),

) {}

onSubmit(): void {

// Process checkout data here

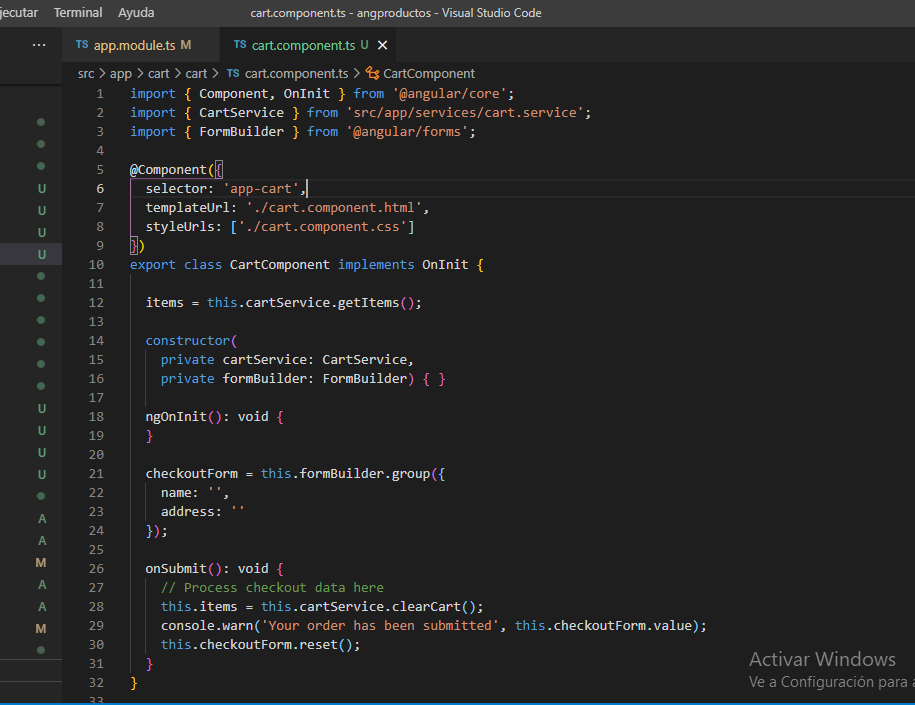
this.items = this.cartService.clearCart();

console.warn('Your order has been submitted', this.checkoutForm.value);

this.checkoutForm.reset();

}

}



**Crea el formulario de pago**

Utilice los siguientes pasos para agregar un formulario de pago en la parte inferior de la vista Carrito.

1. En la parte inferior de cart.component.html, agregue un elemento HTML <form> y un botón Comprar.
2. Utilice un enlace de propiedad formGroup para enlazar checkoutForm al HTML <form>..

***cart.component.html***

<form [formGroup]="checkoutForm">

<button class="button" type="submit">Purchase</button>

</form>

1. En la etiqueta del formulario, use un enlace de evento ngSubmit para escuchar el envío del formulario y llame al método onSubmit () con el valor checkoutForm.

***cart.component.html (cart component template detail)***

<form [formGroup]="checkoutForm" (ngSubmit)="onSubmit()">

</form>

1. Agregue campos <input> para el nombre y la dirección, cada uno con un atributo formControlName que se une a los controles de formulario checkoutForm para el nombre y la dirección de sus campos <input>. El componente completo es el siguiente:

***cart.component.html***

<h3>Cart</h3>

<p>

<a [routerLink](https://angular.io/api/router/RouterLink)="/shipping">Shipping Prices</a>

</p>

<div class="cart-item" \*[ngFor](https://angular.io/api/common/NgForOf)="let item of items">

<span>{{ item.name }} </span>

<span>{{ item.price | [currency](https://angular.io/api/common/CurrencyPipe) }}</span>

</div>

<form [formGroup]="checkoutForm" (ngSubmit)="onSubmit()">

<div>

<label for="name">

Name

</label>

<input id="name" type="text" [formControlName](https://angular.io/api/forms/FormControlName)="name">

</div>

<div>

<label for="address">

Address

</label>

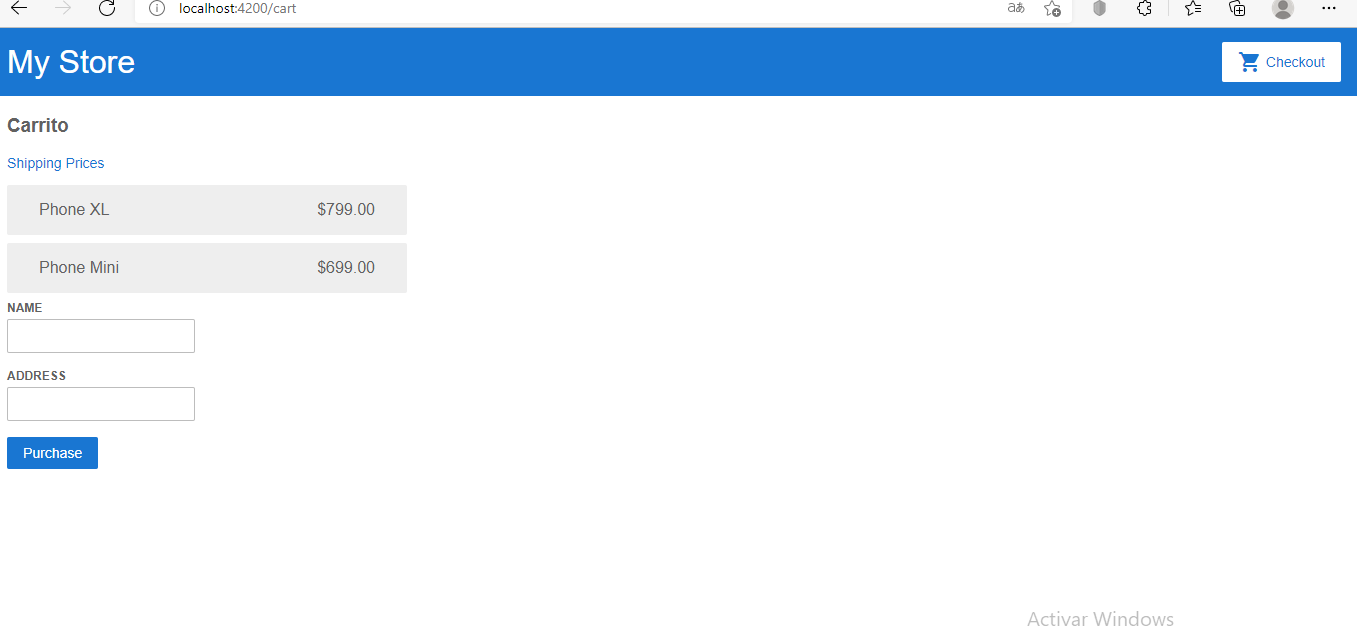
<input id="address" type="text" [formControlName](https://angular.io/api/forms/FormControlName)="address">

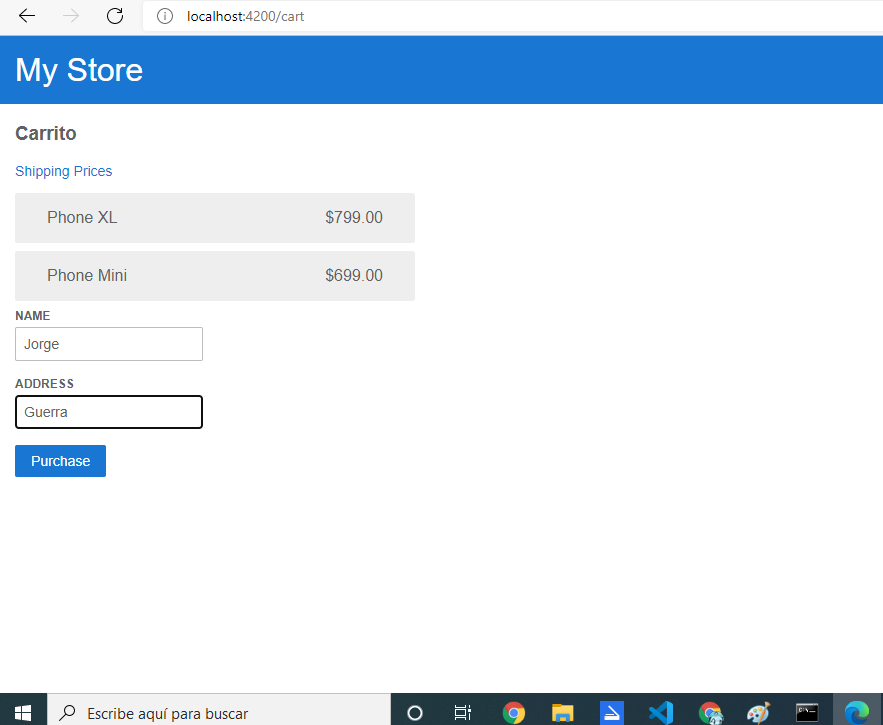
</div>

<button class="button" type="submit">Purchase</button>

</form>

Después de poner algunos artículos en el carrito, los usuarios pueden revisar sus artículos, ingresar su nombre y dirección y enviar su compra.





Para confirmar el envío, abra la consola para ver un objeto que contiene el nombre y la dirección que envió.

