



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Angular-Ecommerce: Desarrollo de una librería de
componentes en Angular para mejorar la experiencia de
usuario en entornos E-Commerce

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Rocafull Berbel, Adrián

Tutor/a: Valderas Aranda, Pedro José

CURSO ACADÉMICO: 2022/2023

Resumen

El comercio electrónico ha experimentado un fuerte crecimiento en todo el mundo, lo que ha llevado a la creación de páginas web de venta de productos y servicios en línea [1]. El uso de Angular como framework de desarrollo es cada vez más común debido a su capacidad para construir aplicaciones web escalables y eficientes.

Sin embargo, construir una página web con Angular puede ser complejo y requerir mucho tiempo y esfuerzo, lo que ha llevado a la necesidad de buscar herramientas que agilicen el proceso de desarrollo. Las librerías de componentes son una de esas herramientas y permiten a los desarrolladores web crear páginas web de manera más rápida y eficiente. En este contexto, he desarrollado ngEcommerce, una librería de componentes específicamente diseñada para el desarrollo de páginas web de e-commerce. Esta librería ofrece una amplia variedad de componentes pre-diseñados y listos para usar, lo que permite a los desarrolladores web ahorrar tiempo y esfuerzo en el proceso de construcción de una página web de e-commerce. La librería ha sido diseñada para adaptarse a las necesidades específicas de los desarrolladores web que trabajan en proyectos de e-commerce, incluyendo funcionalidades para la creación de carritos de compra, barra de búsqueda, tarjetas de productos, entre otros elementos fundamentales en este tipo de plataformas. El trabajo presenta un análisis detallado de la librería, destacando sus principales características y funcionalidades, y muestra una aplicación de la misma. Me decanté por este proyecto ya que ayudará a muchos desarrolladores a agilizar el proceso de creación de webs de e-commerce.

Palabras clave: E-commerce, Angular, desarrollo web, librería de componentes.



Tabla de contenidos

- 1. Introducción**
 - 1.1. Motivación personal
- 2. Estudio de componentes**
 - 2.1. Tarjeta de producto
 - 2.2. Barra de búsqueda
 - 2.3. Previsualización de la cesta de compra
 - 2.4. Página de visualización completa de la cesta de compra
 - 2.5. Resumen de pago
 - 2.6. Detalles del producto
- 3. Estudio de mercado**
 - 3.1. Análisis de competidores
 - 3.2. Análisis DAFO
- 4. Diseño de componentes en Figma**
- 5. Desarrollo de ngEcommerce**
 - 5.1. Creación y arquitectura de la librería
 - 5.2. Estructura del componente
 - 5.3. Interfaces
 - 5.4. Selectores HTML
 - 5.5. SetData y GetData
 - 5.6. Flujo de trabajo con SetData y GetData
- 6. Publicación en NPM**

- 7. Construcción de una web utilizando ngEcommerce**
 - 7.1. Instalación de la librería
 - 7.2. Desplegar un componente
 - 7.3. Sitio web a desarrollar
 - 7.4. Implementación de los componentes
 - 7.5. Funcionalidad añadida
- 8. Conclusiones**
- 9. Trabajos futuros**
- 10. Fuentes**
- 11. OD**



1. Introducción

En los últimos años, el comercio electrónico ha experimentado un crecimiento exponencial en todo el mundo, y con ello, la creación de páginas web dedicadas a la venta de productos y servicios en línea.

Además, cada vez es más común el uso de Angular como framework de desarrollo para la creación de estas páginas web, gracias a su capacidad para construir aplicaciones web escalables y eficientes. Sin embargo, el proceso de construcción de una página web con Angular puede ser complejo y requerir mucho tiempo y esfuerzo, lo que ha llevado a la necesidad de buscar herramientas que agilicen el proceso de desarrollo.

En este contexto, surge la importancia de las librerías de componentes, que permiten a los desarrolladores web crear páginas web de manera más rápida y eficiente. En este trabajo se abordará la importancia de las librerías de componentes en el desarrollo de páginas web con Angular, y se explorarán algunas de las opciones disponibles en el mercado actualmente.

En respuesta a esta necesidad, se ha desarrollado una nueva librería de componentes para Angular [2], específicamente diseñada para el desarrollo de páginas web de e-commerce, llamada **ngEcommerce**. Esta librería ha sido creada con el objetivo de agilizar el proceso de creación de estas plataformas, ofreciendo una amplia variedad de componentes pre-diseñados y listos para usar, lo que permitirá a los desarrolladores web ahorrar tiempo y esfuerzo en el proceso de construcción de una página web de e-commerce.

NgEcommerce ha sido diseñada para adaptarse a las necesidades específicas de los desarrolladores web que trabajan en proyectos de e-commerce, incluyendo funcionalidades para la creación de carritos de compra, barra de búsqueda, tarjetas de productos, entre otros elementos fundamentales en este tipo de plataformas.

En este trabajo se presentará un análisis detallado de la librería de componentes, destacando sus principales características y funcionalidades, y se mostrará una aplicación de la misma.

1.1. Motivación personal

Me decanté por este proyecto ya que esto ayudará a muchos desarrolladores a agilizar el proceso de creación de dichas webs y me permitirá indagar en un nuevo framework como es Angular. Llevo menos de un año investigando los frameworks más demandados para desarrollo web y no podía dejar pasar la oportunidad de estudiar Angular.

2. Estudio de componentes


Para el correcto desarrollo de la librería, se ha requerido un proceso de licitación y extracción de datos de algunas de las más utilizadas webs de comercio electrónico.

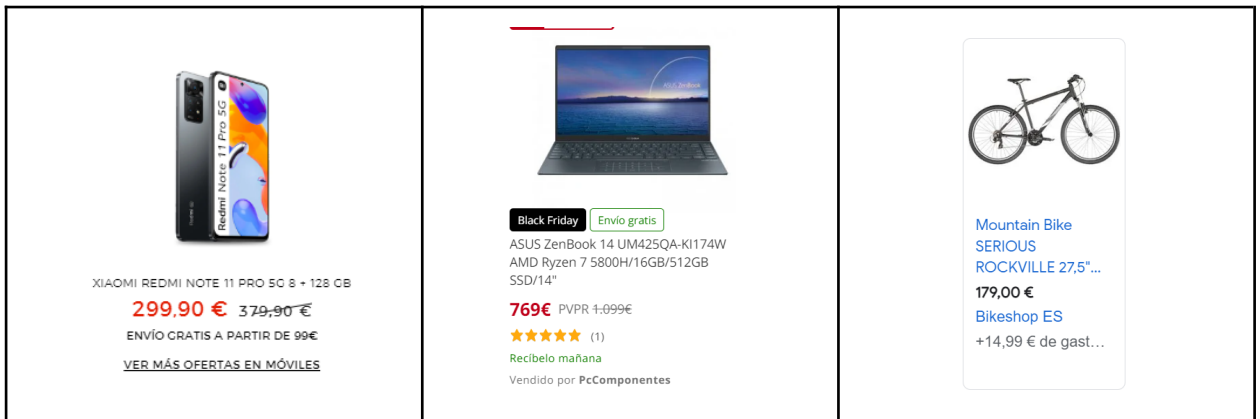
De ellas hemos obtenido algunos de los componentes más utilizados y que son comunes a prácticamente cualquier e-commerce.

Una vez extraídos, han sido comparados para poder tener una mejor visión de lo que es necesario e imprescindible en cada tipo de componente.

1. Tarjeta de producto

Una tarjeta de producto es un componente cuya finalidad es mostrar información resumida sobre un producto específico. Ésta proporciona una vista previa rápida y atractiva de los productos disponibles, lo que facilita a los usuarios navegar y encontrar lo que están buscando. Además, muchas veces permite añadir el producto a la cesta de la compra.

AMAZON	EBAY	ALIEXPRESS
 <p>#2</p>  <p>Cecotec Freidora sin Aceite 6 L de Aire Cecofry Experience Window White 6000. 1300 W, Tecnología PerfectCook, Temperatura...</p> <p>★★★★☆ 580</p> <p>89,90 €</p>	  <p>51,99 EUR</p>	  <p>2€ 07 2,50€</p> <p>Destacado</p>
EL CORTE INGLÉS	PC COMPONENTES	GOOGLE SHOPPING



CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLÉS	ALIEXPRESS	PC COMPONENTES	GOOGLE SHOPPING
PRECIO	✓	✓	✓	✓	✓	✓
NOMBRE	✓	✗	✓	✗	✓	✓
IMAGEN	✓	✓	✓	✓	✓	✓
VALORACION	✓	✗	✗	✗	✓	✗
LINK A PRODUCTOS SIMILARES	✗	✗	✓	✗	✗	✗

Descripción

Según lo visto en estas 6 tiendas online, su tarjeta de producto ofrece características similares.

Sin embargo, las predominantes son:

- Precio
- Nombre
- Imagen

Estas características deberán verse reflejadas en el componente final.

2. Barra de búsqueda

La barra de búsqueda es una funcionalidad común en los sitios web de comercio electrónico que permite a los usuarios buscar productos o información específica dentro del sitio. Los usuarios pueden ingresar palabras clave, nombres de productos o categorías en la barra de búsqueda, y el sistema del comercio electrónico devolverá resultados relevantes que coincidan con la consulta. La barra de búsqueda ayuda a los usuarios a encontrar rápidamente los productos que desean sin tener que navegar manualmente a través de las categorías y páginas del sitio.

AMAZON	
EBAY	
EL CORTE INGLES	
ALIEXPRESS	
PC COMPONENTES	
GOOGLE SHOPPING	NO TIENE

CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLÉS	ALIEXPRESS	PC COMPONENTES	GOOGLE SHOPPING
FILTRADO POR CATEGORÍAS	✓	✓	✗	✓	✓	✗
ICONO BÚSQUEDA	✓	✗	✓	✓	✓	✗
TEXTO PLACEHOLDER	✗	✓	✓	✓	✓	✗

Descripción

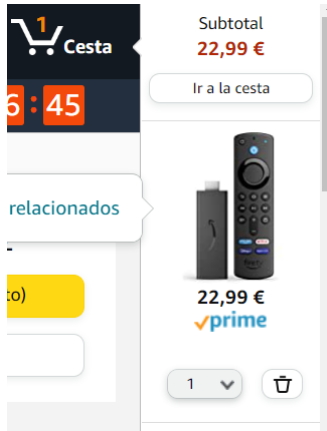
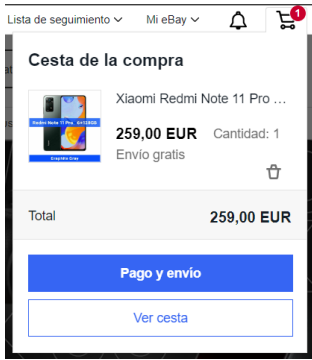
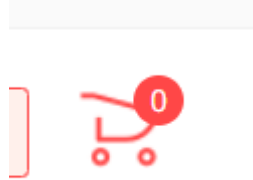
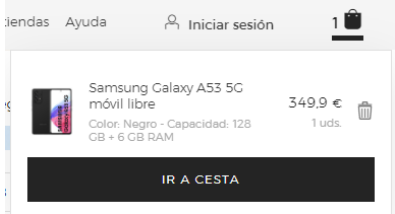
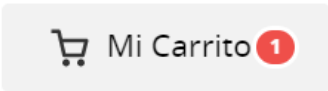
En este caso Google shopping no contaría con una barra de búsqueda como tal, sino que obtendría el producto buscado desde la propia barra de google y desde la pestaña “shopping” se puede acceder a ella. Además, encontramos características similares en todas las barras de búsqueda.

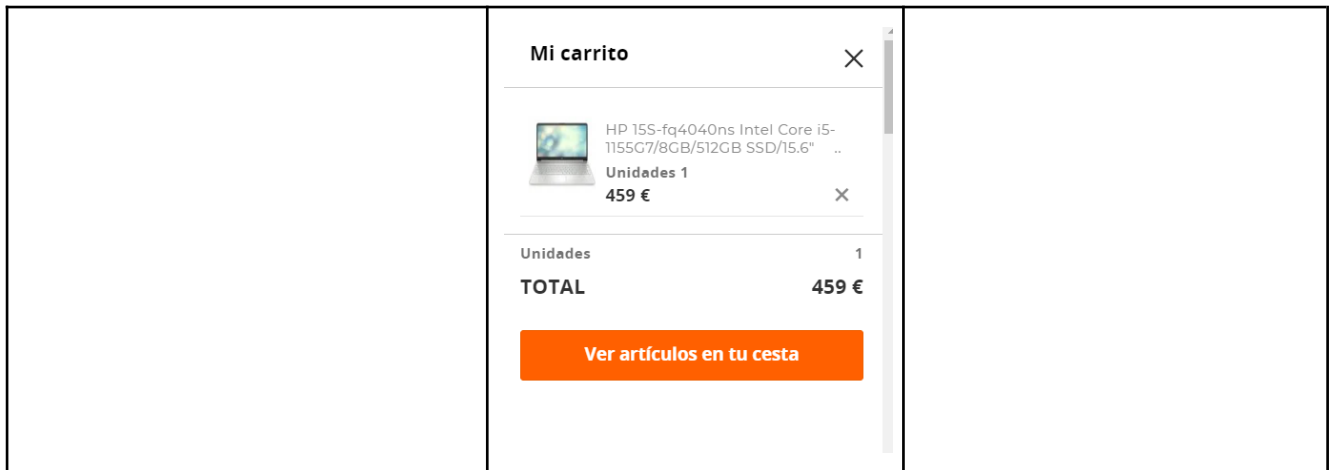
- Filtrado por categorías, previo al avanzado.
- Un icono significativo de búsqueda (lupa).
- Texto placeholder para facilitar el reconocimiento del usuario.

Estas características deberán verse reflejadas en el componente final.

3. Previsualización de la cesta de compra

La previsualización de la cesta de compra muestra un resumen de los artículos seleccionados por un usuario y agregados a su carrito de compras antes de finalizar la compra. Por lo general, se muestra en un área pequeña y visible en el sitio web, generalmente en forma de un ícono de carrito o bolsa de compras en la parte superior de la página. La previsualización de la cesta de compra puede mostrar el número de artículos en el carrito y el costo total aproximado, lo que permite a los usuarios tener una idea rápida de lo que han seleccionado antes de pasar al proceso de pago.

AMAZON	EBAY	ALIEXPRESS
		
EL CORTE INGLES	PC COMPONENTES	GOOGLE SHOPPING
		<p>No tiene</p>








CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLÉS	ALIEXPRESS	PC COMPONENTES
PRECIO TOTAL	✓	✓	✗	✗	✗
PRECIO DE CADA PRODUCTO	✓	✓	✓	✗	✗
UNIDADES MODIFICABLES	✓	✗	✗	✗	✗
PREVISUALIZACIÓN	✓	✓	✓	✗	✗
BOTÓN ELIMINAR PRODUCTO	✓	✓	✓	✗	✗
ICONO CON NÚMERO DE PRODUCTOS	✓	✓	✓	✓	✓
BOTÓN PARA IR A LA CESTA	✓	✓	✓	✓	✓



Descripción

Este componente es bastante diferente según el sitio web. Encontramos que el de Amazon cumple todas las características, por lo que será la principal fuente de inspiración para el futuro componente. La información se presenta de manera mucho más clara y directa al usuario.

4. Página de visualización completa de la cesta de compra

La página de visualización completa de la cesta de compra es donde los usuarios pueden ver todos los artículos que han agregado a su carrito de compras en detalle. Esta página muestra una lista completa de los productos seleccionados, junto con sus imágenes, nombres, precios individuales y la opción de modificar las cantidades o eliminar elementos. Además, también puede mostrar el costo total de la compra, opciones de envío, códigos de descuento y otros detalles relevantes relacionados con la compra.

<p>AMAZON</p>	<div data-bbox="464 824 1559 1182"> <h3>Cesta</h3> <p>Anular la selección de todos los artículos Precio</p> <hr/> <div style="display: flex; justify-content: space-between;"> <div style="display: flex; align-items: center;"> <input checked="" type="checkbox"/>  <div style="margin-left: 10px;"> <p>Fire TV Stick con mando por voz Alexa (incluye controles del TV), dispositivo de streaming HD</p> <p>El n.º 1 más vendido en Electrónica</p> <p>En stock</p> <p> prime</p> <p> Climate Pledge Friendly</p> <p><input type="checkbox"/> Es un regalo Más información</p> <p>Nombre de estilo: Fire TV Stick</p> <p>Cant.: 1 <input type="button" value="Eliminar"/> <input type="button" value="Guardar para más tarde"/> <input type="button" value="Ver otros productos como este"/></p> </div> <div style="text-align: right;"> <p>22,99 €</p> </div> </div> <hr/> <p style="text-align: right;">Subtotal (1 producto): 22,99 €</p> </div> </div>						
<p>EBAY</p>	<div data-bbox="635 1272 1375 1505"> <p>Vendedor <u>summerstore92</u></p> <div style="display: flex; align-items: center;">  <div style="margin-left: 10px;"> <p>Xiaomi Redmi Note 11 Pro 6GB 128GB Smartphone 4G 5000 mAh 67W AMOLED Graphite Gray</p> <p>Nuevo</p> </div> <div style="margin-left: 20px;"> <p>Cantidad <input type="text" value="1"/></p> <p>Envío desde el extranjero con seguimiento</p> <p><input type="button" value="Guardar para más tarde"/> <input type="button" value="Eliminar"/></p> </div> <div style="margin-left: 20px; text-align: right;"> <p>259,00 EUR</p> <p>Envío gratis</p> </div> </div> </div>						
<p>EL CORTE INGLES</p>	<div data-bbox="657 1585 1385 1832"> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 60%;">PRODUCTO</th> <th style="width: 40%;">CANTIDAD</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">  </td> <td style="text-align: center;"> <p>Samsung Galaxy A53 5G móvil libre</p> <p>Color: Negro Capacidad: 128 GB + 6 GB RAM</p> <p><input type="button" value="ENVIÓ A DOMICILIO EN 48H"/></p> <p><input type="checkbox"/> Envolver gratis para regalo</p> </td> </tr> <tr> <td colspan="2" style="text-align: center;"> <p>- 1 +</p> </td> </tr> </tbody> </table> </div>	PRODUCTO	CANTIDAD		<p>Samsung Galaxy A53 5G móvil libre</p> <p>Color: Negro Capacidad: 128 GB + 6 GB RAM</p> <p><input type="button" value="ENVIÓ A DOMICILIO EN 48H"/></p> <p><input type="checkbox"/> Envolver gratis para regalo</p>	<p>- 1 +</p>	
PRODUCTO	CANTIDAD						
	<p>Samsung Galaxy A53 5G móvil libre</p> <p>Color: Negro Capacidad: 128 GB + 6 GB RAM</p> <p><input type="button" value="ENVIÓ A DOMICILIO EN 48H"/></p> <p><input type="checkbox"/> Envolver gratis para regalo</p>						
<p>- 1 +</p>							

<p>ALIEXPRESS</p>	
<p>PC COMPONENTES</p>	
<p>GOOGLE SHOPPING</p>	<p>NO TIENE</p>

CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLÉS	ALIEXPRESS	PC COMPONENTES
PRECIO TOTAL	✓	✗	✗	✗	✗
CANTIDAD	✓	✓	✓	✓	✗
SELECTOR DE PRODUCTOS	✓	✗	✗	✓	✓
BOTÓN PARA ELIMINAR UN PRODUCTO	✓	✓	✗	✓	✓
LINK A PRODUCTO ORIGINAL	✓	✓	✓	✓	✓

MARCAR COMO REGALO	✓	✗	✓	✗	✗
SERVICIOS AÑADIDOS	✗	✗	✗	✗	✓
BOTÓN VACIAR CESTA	✗	✗	✗	✗	✓

Descripción

En este caso optaría por una estética entre PC Componentes y Amazon, ya que juntas ofrecen gran cantidad de información y funcionalidad necesaria para el último paso antes de proceder a la compra.

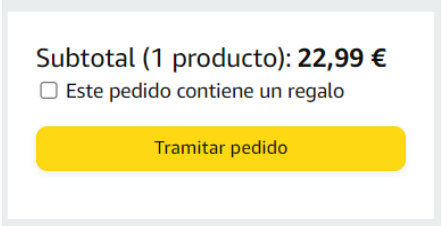



- Precio total
- Cantidad
- Eliminar un producto
- Link a producto original
- Marcar como regalo
- Servicios añadidos
- Botón para vaciar la cesta.

Estas características deberán verse reflejadas en el componente final.

5. Resumen de pago

El resumen de pago es un componente que permite al usuario ver rápidamente el coste total de la compra, con algunos detalles como el envío o distintos descuentos.

Además, le permite acceder rápidamente al pago y completar el pedido.

<p>AMAZON</p>	 <p>Subtotal (1 producto): 22,99 € <input type="checkbox"/> Este pedido contiene un regalo Tramitar pedido</p>
<p>EBAY</p>	 <p>Ir a Pago y envío Artículo (1) 259,00 EUR Envío ⓘ Gratis Total 259,00 EUR</p>
<p>EL CORTE INGLES</p>	 <p>ENTREGA PRECIO <input checked="" type="radio"/> Envío a domicilio en 48h 349,90€ <input type="radio"/> Entrega en el día Antes 649,90€ <input type="radio"/> Click&Car Ahora 100€ (22%) <input type="radio"/> Recogido en tienda CUPONES Introduce aquí tu cupón... APLICAR Subtotal: 349,90€ Gastos de envío estimados: Gratis TOTAL 349,90€ <input type="checkbox"/> Recibir el ticket regalo de la compra por email (así podrás enviárselo a quien quieras) TRAMITAR COMPRA</p>
<p>ALIEXPRESS</p>	 <p>Resumen Subtotal € 0,01 Total a pagar € 0,01 IVA incluido ⓘ Pagar (1)</p>

PC COMPONENTES	<p>Resumen</p> <p>Subtotal artículos 459€</p> <hr style="border-top: 1px dashed #ccc;"/> <p>Total (Impuestos incluidos) 459€</p> <p style="text-align: center;">Guardar y continuar</p>
-----------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------


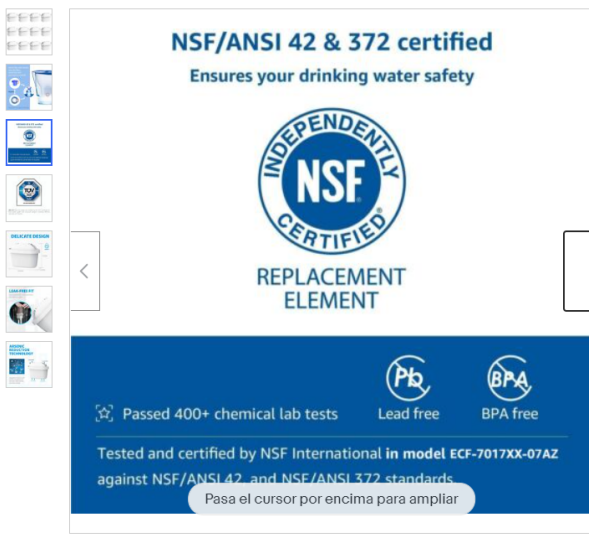
CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLES	ALIEXPRESS	PC COMPONENTES
TOTAL A PAGAR	✓	✓	✓	✓	✓
TOTAL DE ARTÍCULOS	✗	✓	✓	✓	✓
BOTÓN PARA PROCEDER AL PAGO	✓	✓	✓	✓	✓
DESCOMPOSICIÓN DE PRECIOS	✗	✗	✓	✗	✗

Descripción

Este componente tendrá todas las características antes mencionadas excepto la descomposición de precios.

6. Detalles del producto

Los detalles del producto consta de una colección de imágenes y medios visuales relacionados con un producto específico que se muestra en la página del producto en un sitio web de comercio electrónico. La galería de producto permite a los usuarios ver diferentes vistas del producto, detalles en primer plano, variaciones de color o estilo, e incluso videos o animaciones que muestran el producto en uso o en acción.

<p>AMAZON</p>	 <p>The image shows an Amazon product detail for the Echo Dot. It features a main image of the black Echo Dot on a wooden surface. Above the product, the text reads 'Nuevo echo dot' and 'Nuestro Echo Dot con mejor calidad de sonido hasta la fecha'. Below the product, there are six icons representing features: 'Sonido potente', 'Temporizadores', 'Información', 'Controles de privacidad', 'Climate Pledge Friendly', and 'Sensor de temperatura'. A vertical gallery of smaller images is visible on the left side of the main image.</p>
<p>EBAY</p>	 <p>The image shows an eBay product detail for a water filter. The main text reads 'NSF/ANSI 42 & 372 certified' and 'Ensures your drinking water safety'. Below this is a large circular logo that says 'INDEPENDENTLY CERTIFIED NSF REPLACEMENT ELEMENT'. At the bottom, there are icons for 'Lead free' and 'BPA free', along with the text 'Passed 400+ chemical lab tests'. A blue banner at the bottom contains the text 'Tested and certified by NSF International in model ECF-7017XX-07AZ against NSF/ANSI 42 and NSF/ANSI 372 standards' and a button that says 'Pasa el cursor por encima para ampliar'. A vertical gallery of smaller images is visible on the left side of the main image.</p>

EL CORTE INGLES

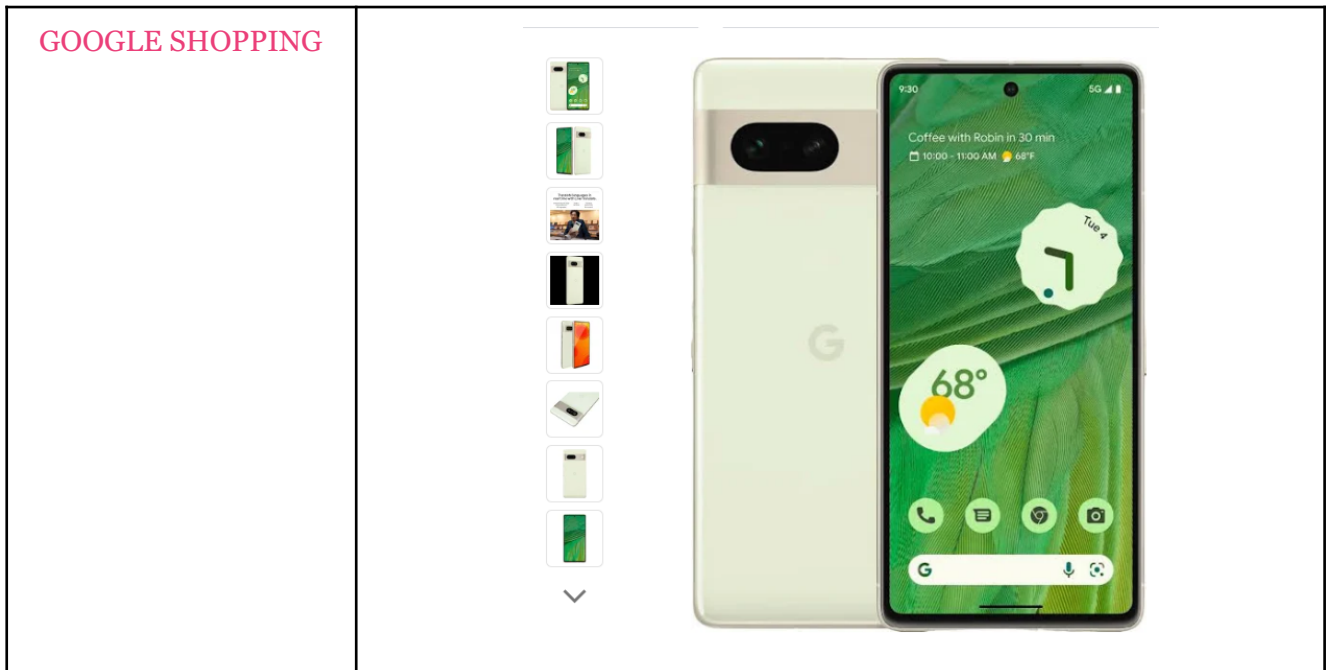


ALIEXPRESS



PC COMPONENTES





CARACTERÍSTICAS	AMAZON	EBAY	EL CORTE INGLÉS	ALIEXPRESS	PC COMPONENTES	GOOGLE SHOPPING
IMAGEN CENTRAL	✓	✓	✓	✓	✓	✓
IMAGENES LATERALES	✓	✓	✓	✓	✗	✓
ZOOM	✓	✓	✓	✓	✓	✓
CARRUSEL	✗	✓	✗	✗	✗	✗
VIDEO	✓	✓	✓	✓	✗	✓

Descripción

Se ha optado por una visualización por imágenes y similar a amazon, añadiendo zoom a la imagen al pasar el cursor por encima.

3. Estudio de mercado

En el mundo del desarrollo web, las librerías de componentes son herramientas indispensables para acelerar el proceso de desarrollo y mejorar la calidad de las aplicaciones. En este análisis de mercado, nos centraremos en comparar las distintas librerías de componentes disponibles para Angular, uno de los frameworks de JavaScript más populares, con esta nueva librería.

3.1. Análisis de competidores

Se han recopilado algunas de las librerías de componentes más utilizadas en Angular.

Angular Material

Angular Material [5] es una librería de UI que proporciona componentes basados en Material Design para Angular. Se utilizan sus componentes mediante la importación de módulos específicos y la inclusión de etiquetas HTML personalizadas.

PrimeNG

PrimeNG [6] es una librería de UI de código abierto que ofrece una amplia variedad de componentes para Angular.

NFX Bootstrap

NGX Bootstrap [7] es una librería de UI que proporciona componentes basados en Bootstrap 4 para Angular.

Clarity

Clarity [8] es una librería de UI de VMware que proporciona componentes de diseño limpio y moderno para Angular.

Materialize

Materialize [9] es una librería de UI que proporciona componentes basados en Material Design para Angular.



Se ha comprobado que componentes existen ya en las librerías más populares de Angular.

PRODUCTOS	Angular Material	PrimeNG	NGX Bootstrap	Clarity	Materialize
COMPONENTES					
Tarjeta de producto	✗	✗	✗	✗	✗
Barra de búsqueda	✗	✗	✗	✗	✗
Previsualización de la cesta de compra	✗	✗	✗	✗	✗
Página de visualización completa de la cesta de compra	✗	✗	✗	✗	✗
Resumen de pago	✗	✗	✗	✗	✗
Detalles del producto	✗	✗	✗	✗	✗

He podido comprobar que las librerías de componentes más populares actualmente en el mercado se centran en ser lo más flexibles posible para poder ser utilizadas en el desarrollo de cualquier tipo de web.

Esto hace que no sean librerías especializadas, sino genéricas, siendo menos eficientes para abordar un problema concreto.

Además, encontré una librería llamada **AngularEcommerce**. Lamentablemente, esta librería no funciona correctamente. En NPM no la podemos encontrar y si lo descargamos directamente de su repositorio, aparece con errores de sintaxis.

3.2. Análisis DAFO

El análisis DAFO [10] es una herramienta de análisis estratégico que se utiliza para evaluar la situación de una entidad, ya sea una empresa, producto o proyecto. Sus siglas significan Fortalezas, Oportunidades, Debilidades y Amenazas.

El análisis DAFO se utiliza para:

- **Evaluar la situación:** Permite identificar y comprender los factores internos y externos que afectan a la entidad, tanto positivos como negativos.
- **Identificar fortalezas:** Ayuda a identificar los aspectos internos positivos que brindan ventajas competitivas y diferenciación.
- **Detectar oportunidades:** Permite identificar factores externos favorables que se pueden aprovechar para el beneficio de la entidad.
- **Analizar debilidades:** Ayuda a identificar los aspectos internos negativos que limitan el desempeño y la competitividad.
- **Evaluar amenazas:** Permite identificar factores externos desfavorables que representan riesgos potenciales.

FORTALEZAS	DEBILIDADES
<ul style="list-style-type: none">● Es un proyecto creado completamente desde cero● Librería muy específica para resolver un problema concreto● Instalación ligera● Compatible con otras librerías de componentes	<ul style="list-style-type: none">● Falta de conocimiento del framework.● Librería poco flexible debido a su especificidad.● Bajos recursos● Necesidad de productos extra para darse a conocer (web, ejemplos, documentación...)
OPORTUNIDADES	AMENAZAS
<ul style="list-style-type: none">● No existen librerías especializadas● Al ser una librería gratuita puede ser utilizada por cualquier desarrollador	<ul style="list-style-type: none">● Existen librerías mucho más populares abarcando gran parte del mercado● Dificultad de destacar debido al SEO



4. Diseño de componentes en Figma

Los componentes han sido diseñados en Figma [11], para utilizarlos como base y prototipo para su posterior implementación.

Han sido creados siguiendo las directrices de lo que se considera un buen diseño UI/UX [12,13], como podrían ser:

- Botones claros con una llamada a la acción.
- Información resumida e importante a la vista del usuario
- Clara diferenciación utilizando diferentes formas y colores.
- Uniformidad en las formas de los componentes.

Se puede acceder al documento de Figma en el siguiente [enlace](#).

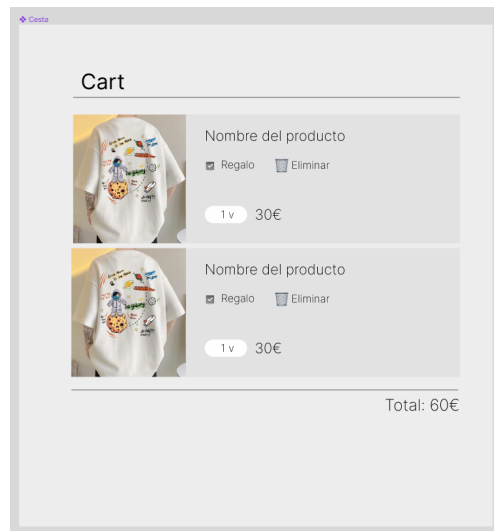
Aquí podemos observar una tabla en la que se muestran los diferentes diseños para cada componente.

Componente	Diseño
Galería de imágenes	
Pago	

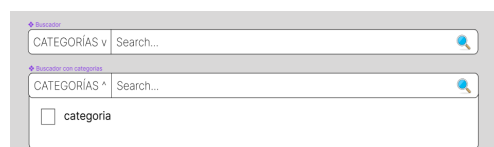
Vista previa del carrito de compra



Carrito de compra



Barra de búsqueda



Tarjeta de producto

Product card



CAMISETA
20€



5. Desarrollo de ngEcommerce

Para el desarrollo de la librería se ha utilizado Angular. Éste es un framework de desarrollo web de código abierto que permite crear aplicaciones web de una sola página (SPA) [14]. Angular.js facilita el desarrollo de aplicaciones web dinámicas y escalables, utilizando HTML, CSS y JavaScript. Ofrece numerosas herramientas y características, como la inyección de dependencias, la vinculación de datos y el enrutamiento, lo que lo convierte en una opción popular para el desarrollo de aplicaciones web complejas.

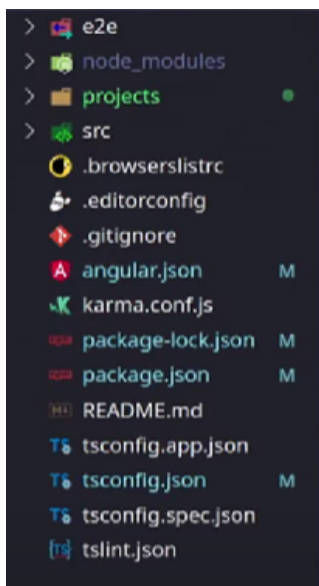
Por otro lado, como entorno de desarrollo, he elegido Visual Studio Code. Éste es un entorno de desarrollo integrado (IDE) gratuito y de código abierto. Es muy popular entre los desarrolladores web debido a su facilidad de uso y su capacidad para integrarse con herramientas populares como Git y TypeScript.

1. Creación y arquitectura de la librería

El primer paso del desarrollo es crear el proyecto y la estructura de carpetas. Para ello Angular nos ofrece el siguiente comando:

```
ng generate library ngEcommerce [15]
```

Este comando nos creará la siguiente estructura:



En ella podemos encontrar multiples archivos de configuración en la parte inferior. Cabría destacar el **package.json**, ya que en él podremos definir muchos datos del desarrollo que serán de utilidad al publicar el paquete. Además, se pueden ver dependencias necesarias para su desarrollo.

```
{
  "name": "ng-ecommerce",
  "version": "0.0.1",
  "description": "This is an Angular component library that will help you creating new e-commerce websites",
  "author": {
    "name": "Adrian Rocafull",
    "email": "adrianrocafull@gmail.com",
    "url": "https://adrianrocafull.onrender.com"
  },
  "peerDependencies": {
    "@angular/common": "^15.0.0",
    "@angular/core": "^15.0.0"
  },
  "dependencies": {
    "tslib": "^2.3.0"
  }
}
```

Otro apartado que llama la atención es la carpeta **projects**. Dicha carpeta almacenará toda la lógica y el código necesario para crear los componentes de la librería.

```

└─ projects \ng-ecommerce
  └─ src
    └─ lib
      > components
      > interfaces
      TS ng-ecommerce.module.ts
      TS ng-ecommerce.service.ts
      TS public-api.ts

```

Podemos encontrar el archivo **public-api.ts**. Aquí decidiremos que exportar para que pueda ser utilizado en uno o más proyectos de Angular. En nuestro caso, exportamos tanto los componentes como los módulos.

```

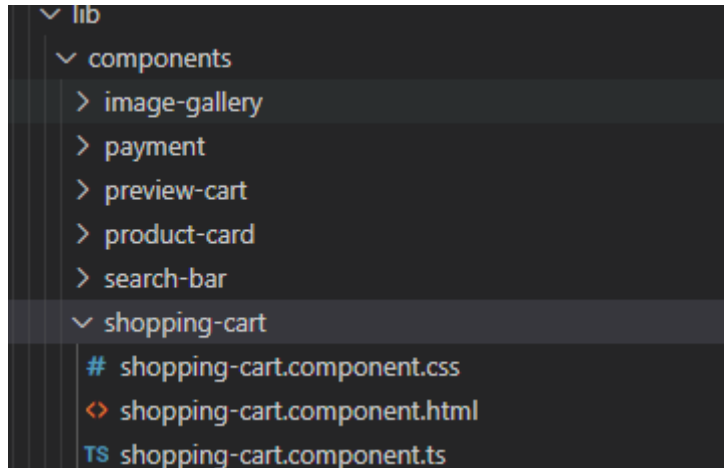
/*
 * Public API Surface of ng-ecommerce
 */

export * from './lib/ng-ecommerce.service';
export * from './lib/ng-ecommerce.module';
export * from './lib/components/image-gallery/image-gallery.component';
export * from './lib/components/search-bar/search-bar.component';
export * from './lib/components/payment/payment.component';
export * from './lib/components/preview-cart/preview-cart.component';
export * from './lib/components/shopping-cart/shopping-cart.component';
export * from './lib/components/product-card/product-card.component';

```

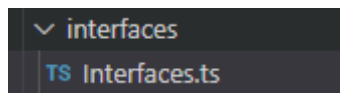
Una vez accedemos a la carpeta **lib** encontramos **components** y **interfaces**.

El directorio **components** está compuesto por carpetas que almacenan cada uno de los componentes.



Encontramos que están formados por un archivo CSS, otro HTML y un último TypeScript. Estos conforman tanto la lógica como la presentación de cada componente.

También podemos ver la carpeta **interfaces**. Esta contiene un único archivo de TypeScript en el que se especifican las interfaces de cada componente.



1. Estructura del componente

Angular tiende a modularizar y dividir todo lo máximo posible. Esto incluye los archivos CSS, HTML y TS. El archivo TS es el que se ocupa de unificarlo todo mediante distintos atributos en el decorador `@Component`.

```
@Component({
  selector: 'ecom-shopping-cart',
  templateUrl: './shopping-cart.component.html',
  styleUrls: ['./shopping-cart.component.css']
})
```

Aquí tenemos en orden el selector HTML, el documento HTML del componente y los estilos de CSS. El archivo TS tendrá gran parte de la lógica, mientras que el HTML y CSS se ocupan de la presentación, aunque siendo dinámicos gracias a las directivas de Angular. Este es un ejemplo de implementación:



```

<div class="contenedor-preview-cesta">
  <h1>Cart</h1>
  <hr>
  <div *ngFor="let product of setData.products; let i=index" class="contenedor-producto-cesta">
    <img [src]="product.image" alt="image-product">
    <div class="contenedor-datos-producto-cesta">
      <a [routerLink]="product.link">{{product.productName}}</a>
      <div class="seccion-prod-cesta">
        <div class="present">
          <input name="present" type="checkbox">
          <label for="present">Present</label>
        </div>
        <button class="boton-eliminar" (click)="eliminarProducto(i)">Delete</button>
      </div>
      <div class="seccion-prod-cesta">
        <div class="selector-cantidad">
          <button (click)="menosCantidad(i)">-</button>
          <p>{{cantidad[i]]}</p>
          <button (click)="masCantidad(i)">+</button>
        </div>
        <p class="precio-articulo">{{product.price}}</p>
      </div>
    </div>
  </div>
  <div>
    <button class="proceed-to-payment" (click)="sendData()">Proceed to payment</button>
  </div>

```

2. Interfaces

Una vez se requiera utilizar la librería, el usuario necesitará enviar ciertos datos para crear los componentes personalizados. ¿Pero, cómo saber qué enviar? Para ello se han añadido las **interfaces** [16]. Éstas contienen las directrices necesarias para la correcta creación del componente.

Nombre de las interfaces y parámetros:

1. Product
 - **Class:** permite añadir una o varias clases al componente.
 - **ProductName:** Nombre del producto.
 - **Image:** Nombre del producto.
 - **Price:** precio del producto
 - **Link:** link del producto para ver en detalle.
 - **Amount:** cantidad de unidades del producto.
2. Cart
 - **Class:** permite añadir una o varias clases al componente.
 - **Products:** Lista de productos que han sido añadidos al carrito.
3. SearchBar
 - **Class:** permite añadir una o varias clases al componente.
 - **Categories:** Lista de categorías de búsqueda.
4. Payment
 - **Class:** permite añadir una o varias clases al componente.
 - **Price:** precio total de la compra
 - **Link:** link para proceder a la compra.
 - **Amount:** cantidad de productos.
5. PreviewCart
 - **Class:** permite añadir una o varias clases al componente.
 - **Total:** precio total de los productos del carrito.
 - **Link:** link para proceder a la compra.
 - **Products:** Lista de productos de la cesta
6. ImageGallery
 - **Class:** permite añadir una o varias clases al componente.
 - **Images:** Lista de imágenes de un producto.

3. Selectores HTML

Un selector es una de las propiedades del objeto que usamos junto con la configuración del componente.

Se utiliza un selector para identificar cada componente de manera única en el árbol de componentes, y también define cómo se representa el componente actual en el DOM HTML [17].

Estos han sido el selector seleccionado para cada componente, teniendo una primera parte que hace referencia a la librería (ecom), y una segunda parte representativa

Componente	Selector HTML
Galería de imágenes	<ecom-image-gallery>
Pago	<ecom-payment>
Vista previa del carrito de compra	<ecom-preview-cart>
Carrito de compra	<ecom-shopping-cart>
Barra de búsqueda	<ecom-search-bar>
Tarjeta de producto	<ecom-product-card>



4. SetData y GetData

Una vez tenemos la estructura del componente, falta saber cómo se comunica con la aplicación de Angular que lo utilizará.

Para ello se ha diseñado un cómodo sistema de comunicación en dos direcciones dividido en dos partes: GetData y SetData. Estas nuevas directivas [18] nos proporcionarán la funcionalidad necesaria.

- SetData: Define la información inicial del componente
- GetData: Recibe la información actual del componente.

Angular nos ofrece dos decoradores esenciales para su implementación: @Input y @Output. Utilizando el input crearemos el SetData y con el output, el GetData.

Componente	Interfaz SetData	GetData
Galería de imágenes	ImageGallery	-
Pago	Payment	-
Vista previa del carrito de compra	PreviewCart	PreviewCart
Carrito de compra	Cart	Cart
Barra de búsqueda	SearchBar	<ul style="list-style-type: none"> • searchText: Texto introducido en la barra de búsqueda • category: Categoría seleccionada
Tarjeta de producto	Product	Product

5. Flujo de trabajo con GetData y SetData

El flujo de trabajo es el estudio de las características operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo...

Para este ejemplo de uso utilizaremos el componente **Tarjeta de producto**.

```
@Input()
setData!:Product;
@Output()
getData = new EventEmitter<Product>();

emitData(){
  this.getData.emit(this.setData);
}
```

Como puede observarse, tenemos un @Input() setData de tipo Product y un emisor de eventos @Output() getData que devuelve un elemento de tipo Product.

Primero, desde el punto de vista de la aplicación cliente, deberá crear diversos productos utilizando la interfaz **Products**.

```
productCards:Array<Product>=[]
ngOnInit(){
  this.productsService.getProducts().subscribe((products) => {
    this.parseDataProducts(products);
  });
}
```

En este caso se crea un array de productos obtenidos mediante una llamada a una API externa. Una vez tiene la lista de productos, itera sobre ella creando una tarjeta de producto para cada uno.

```
<section class="shop-container">
  <ecom-product-card *ngFor="let product of productCards" [setData]="product" (getData)="getDataProduct($event)"></ecom-product-card>
</section>
```

Para ello, establece los datos de cada producto mediante la directiva [setData]. Una vez hecho esto, la librería digiere la información y la transforma en código HTML.




```

<div class="card-container" [class]="setData.class">
  <a [href]="setData.link" class="product-link"></a>
  <img [src]="setData.image" class="product-image" alt="">
  <div class="bottom-card-container">
    <div class="product-data-container">
      <h3>{{setData.productName}}</h3>
      <p>{{setData.price}}</p>
    </div>
    <div class="add-cart-container">
      <button class="add-to-cart" type="button" (click)="emitData()"> </button>
    </div>
  </div>
</div>

```

Si se desean emitir los datos, el usuario deberá hacer click en el botón *añadir al carrito*. Esto activará el evento `getData()`, que será capturado desde la aplicación cliente.

```

<section class="shop-container">
  <ecom-product-card *ngFor="let product of productCards" [setData]="product" (getData)="getDataProduct($event)"></ecom-product-card>
</section>

```

6. Publicación en NPM

Una vez terminada la librería, se publicará en NPM [4]. NPM es un servicio de Node.js que gestiona los distintos paquetes utilizados para el desarrollo de software.

No es imprescindible que se suba a esta plataforma, ya que los desarrolladores podían acceder a la librería mediante GitHub, GitLab u otras plataformas de repositorios de código. Sin embargo, npm ofrece muchas ventajas como la comodidad, facilidad y fiabilidad de su software y paquetes.

1. Crear perfil en npm
2. Añadir usuario de npm utilizando **npm adduser**
3. Compilar la librería utilizando **npm build**
4. Entrar en la carpeta **dist** generada
5. Subir la librería utilizando **npm publish**

Podemos encontrar la librería subida [aquí](#).

Para instalarla en nuestro proyecto, utilizaremos el comando **npm i ng-e-commerce-components**

7. Construcción de una web utilizando ngEcommerce

Este apartado explicará cómo preparar tu proyecto de Angular para utilizar la librería, además de un ejemplo de uso en una web sencilla para comprobar su funcionamiento.

1. Instalación de la librería

Para empezar, ejecutaremos el siguiente comando para instalar la librería [4]:

```
npm i ng-ecommerce-components
```

Esto hará que se añada a los paquetes de node y obtendremos acceso a ella.

Una vez lo tenemos, importamos el módulo dentro del app.module.ts.

```
imports: [  
  BrowserModule,  
  AppRoutingModule,  
  NgEcommerceModule,  
  CommonModule,  
  HttpClientModule  
],
```

Ahora ya estamos listos para utilizarla.

2. Desplegar un componente

Para este caso de uso vamos a utilizar el componente de SearchBar. Primero importamos la interfaz SearchBar y crearemos un objeto utilizandola.

```
searchBarData:SearchBar={  
  class: " ",  
  categories: ["Ropa", "Baterías", "Cocina", "Muebles", "Oficina", "Viaje", "Productos para mascotas"],  
}
```

Este objeto tendrá todos los datos necesarios para crear el objeto. Además, opcionalmente podrás crear un método que capture los eventos de dicho objeto.

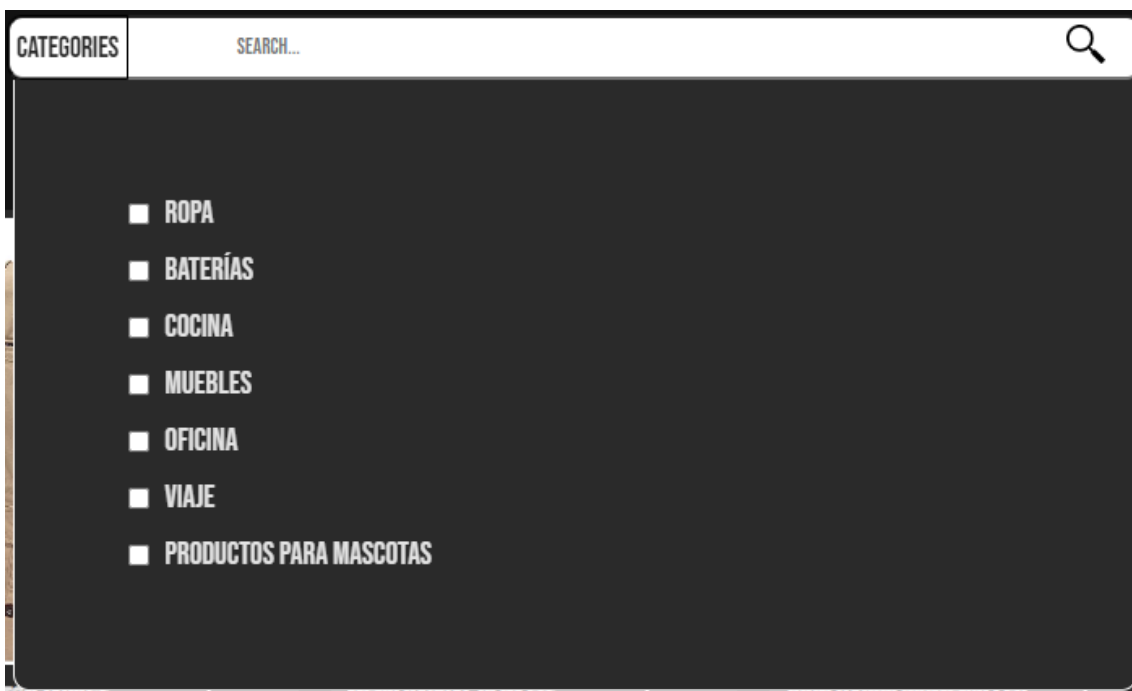


```
getDataSearchBar(data:any){
  // ...
}
```

Ahora, solo nos queda pasar los datos a través de las directivas del componente.

```
<ecom-search-bar [setData]="searchBarData" class="search-bar" (getData)="getDataSearchBar($event)">
```

Esto nos creará un componente que se verá de la siguiente manera:




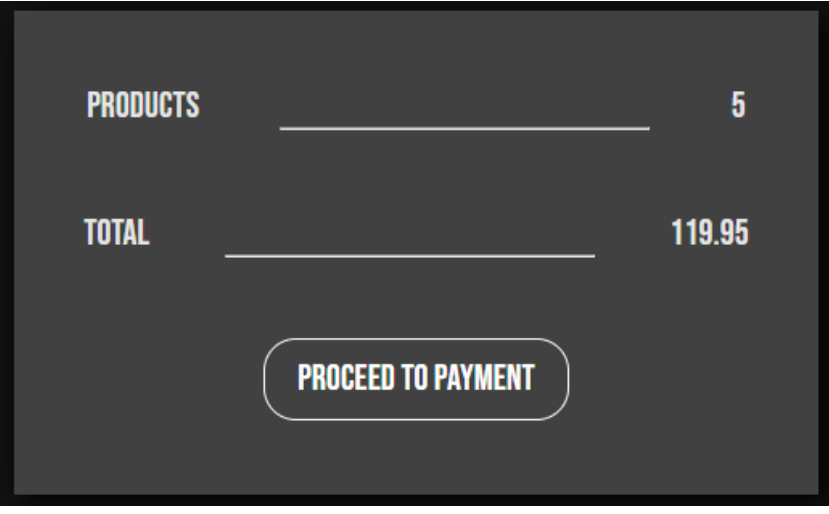
3. Sitio web a desarrollar

Se ha creado un sitio web de ventas online en el que se podrán comprar todo tipo de productos: ropa, productos tecnológicos, bisutería...

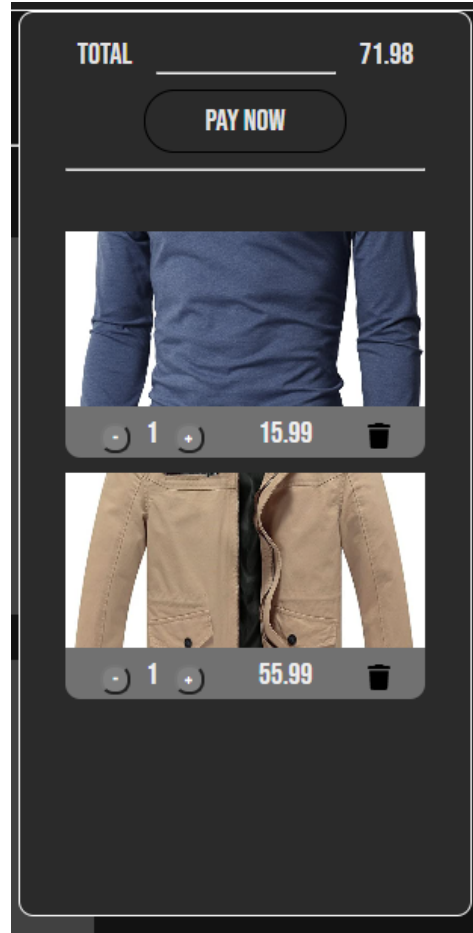
Esto demostrará la versatilidad de la librería y la capacidad de polivalencia dentro del nicho de los e-commerce.

4. Implementación de los componentes

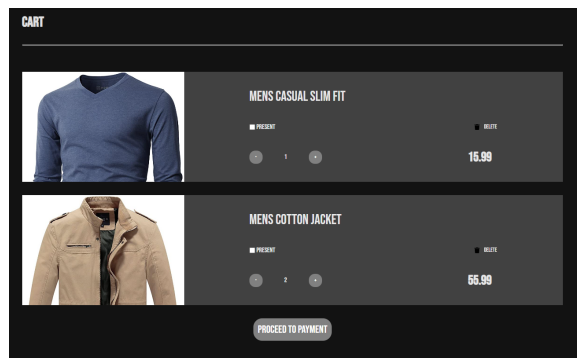
Los componentes han sido implementados siguiendo el diseño realizado en figma, aunque adaptándose al diseño de la web.

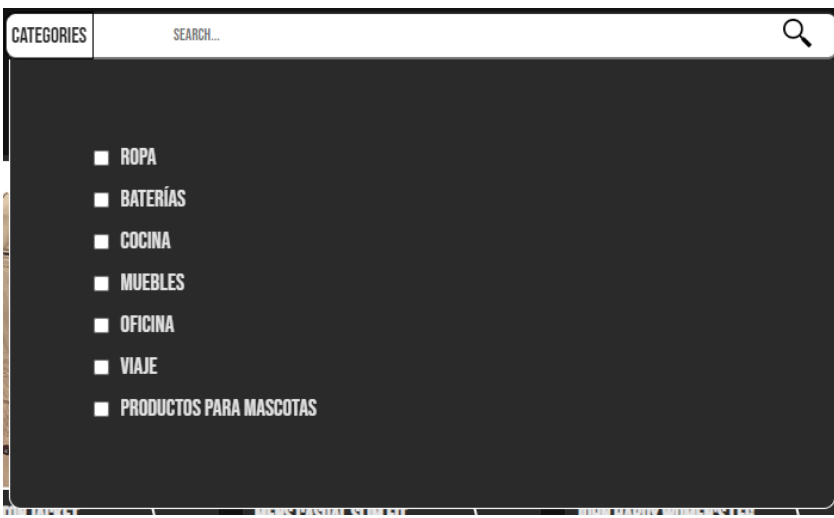

Componente	Implementación				
Galería de imágenes					
Pago	 <table border="1"><tbody><tr><td>PRODUCTS</td><td>5</td></tr><tr><td>TOTAL</td><td>119.95</td></tr></tbody></table> <p>PROCEED TO PAYMENT</p>	PRODUCTS	5	TOTAL	119.95
PRODUCTS	5				
TOTAL	119.95				

Vista previa del carrito de compra



Carrito de compra



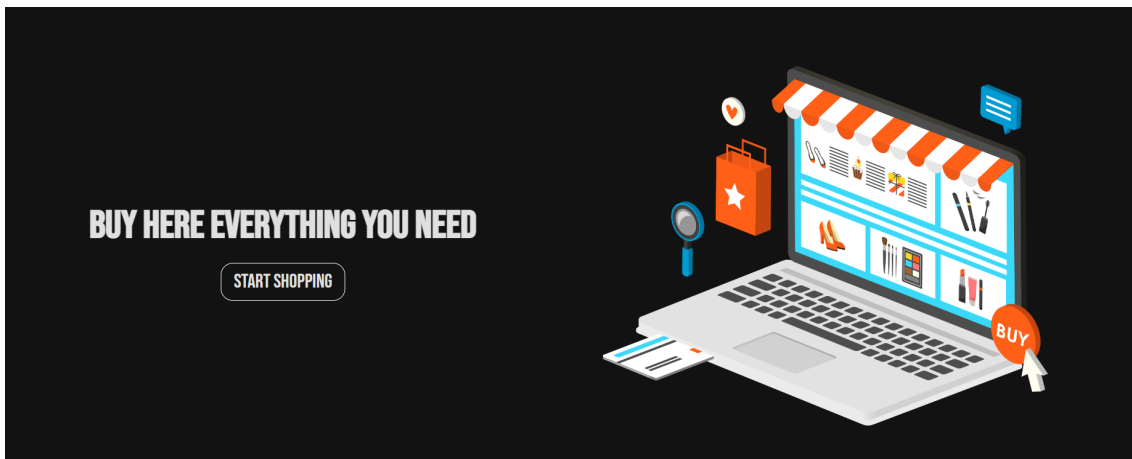
<p>Barra de búsqueda</p>	
<p>Tarjeta de producto</p>	

5. Funcionalidad añadida

Para la creación del sitio web, se ha necesitado implementar otras funcionalidades y componentes.

- **Página de inicio**

La página de inicio está diseñada específicamente para captar la atención de los visitantes y llevarlos a realizar una acción deseada, convirtiéndolos en leads o clientes potenciales, invitándolos a clicar en el botón para ir a la sección de compras de nuestra web.



- Barra de navegación

Componente que nos permitirá navegar en la página por sus diferentes apartados.

Consta de diferentes animaciones y estados para indicar en qué página se encuentra el usuario.



- Servicios

Un servicio es una clase, comúnmente decorada con el decorador Injector de Angular, mismo que indica que este Servicio puede inyectar otras dependencias de la aplicación, ya sean otros servicios como el de Http para hacer consultas AJAX.

La clase productsService ha sido utilizada para hacer llamadas a una API externa llamada fakestoreapi. Esta tiene múltiples opciones, entre las que se encuentran obtener datos sobre productos como ropa, electrónica y bisutería.

```
export class productsService {
  private apiUrl= "https://fakestoreapi.com/products/";

  constructor(private http:HttpClient){}

  getProducts() :Observable<any[]>{
    return this.http.get<any[]>(this.apiUrl);
  }
  getProductWithID(id:string) :Observable<any[]>{
    return this.http.get<any[]>(this.apiUrl + id);
  }
}
```

Además, se ha creado un segundo servicio para gestionar los datos comunes a toda la aplicación, como el precio total del carrito, añadir productos u obtener el número de productos añadido.

```
constructor(private sanitizer:DomSanitizer){ ...
}

get dataProductCart(){ ...
}

get cartProducts(){ ...
}

get totalPrice(){ ...
}

get numberOfProducts(){ ...
}

addProduct(product:Product){ ...
}

setDataProductCart(data:PreviewCart){ ...
}

sanitize(url:string){ ...
}
```

- **Persistencia de datos**

A la hora de gestionar los datos en un ecommerce, requieren ser almacenados para que si el usuario cierra la página o navega en nuestra web, no los pierda. Por ello se ha optado por almacenarlos utilizando LocalStorage [3].

El LocalStorage permite almacenar datos en el navegador web. Y que estos persistan y estén disponibles durante la navegación en la aplicación web, hasta que esta información sea borrada del navegador.

productData	{ "class": "", "link": "payment", "total": "119..."
cartProducts	[{ "image": "https://fakestoreapi.com/im..."



8. Conclusiones

En conclusión, el desarrollo de la librería de componentes de Angular para E-commerce ha sido un éxito, ya que se han logrado todos los objetivos planteados inicialmente. La librería proporciona una amplia gama de componentes personalizables y reutilizables que permiten a los desarrolladores crear aplicaciones de comercio electrónico de alta calidad de manera más eficiente y efectiva.

La librería ha sido diseñada y desarrollada teniendo en cuenta las mejores prácticas y estándares de Angular, lo que la hace fácilmente integrable con proyectos existentes y ofrece una gran flexibilidad para adaptarse a diferentes necesidades.

También se ha conseguido aprender más acerca del desarrollo con Angular y de cómo funcionan sus librerías.

Además, se ha proporcionado una documentación detallada y ejemplos de uso para ayudar a los desarrolladores a comprender y utilizar la librería de manera efectiva. Crear un ecommerce desde cero puede ser muy complejo y, gracias a esta librería, ha sido un proceso mucho más rápido.

En resumen, la librería de componentes de Angular para E-commerce representa una valiosa contribución al ecosistema de desarrollo de aplicaciones de comercio electrónico en Angular, y esperamos que sea de gran ayuda para los desarrolladores en sus proyectos futuros.

9. Trabajos futuros

A pesar de haber logrado los objetivos planteados, se ha tenido que prescindir de algunas funcionalidades que no han podido entrar en esta primera versión.

- Crear componentes responsive para todos los dispositivos
- Añadir más componentes
- Añadir un modo día/noche dependiendo de la clase del componente
- Crear un archivo readme para la página de NPM
- Crear una web o un tutorial en GitHub para desarrolladores
- Hacer pruebas unitarias y de integración



Bibliografía

1. Crecimiento y estadísticas del eCommerce (2023) | StackScale.
<https://www.stackscale.com/es/blog/crecimiento-estadisticas-ecommerce/>
Acceso: Febrero 2023
2. Angular. The web development framework for building the future.
<https://angular.io/>. Acceso: Mayo 2023
3. Como utilizar el LocalStorage en Angular - Itelisoft
<https://itelisoft.com/como-utilizar-el-localstorage-en-angula/>
Acceso: Abril 2023
4. NPM
<https://www.npmjs.com/>
Acceso: Mayo 2023
5. Angular Material UI component library
<https://material.angular.io/>
Acceso: Febrero 2023
6. PrimeNG - Angular UI Component Library
<https://primeng.org/>
Acceso: Febrero 2023
7. [NFX] How to bootstrap vSRX VNF
https://supportportal.juniper.net/s/article/NFX-How-to-bootstrap-vSRX-VNF-with-a-custom-default-configuration?language=en_US
Acceso: Febrero 2023
8. Clarity Design System
<https://clarity.design/>
Acceso: Febrero 2023
9. Documentation - Materialize
<https://materializecss.com/>
Acceso: Febrero 2023
10. Qué es un Análisis DAFO: usos, ventajas y ejemplos
<https://www.iebschool.com/blog/analisis-dafo-creacion-empresas/#:~:text=El%20an%C3%A1lisis%20DAFO%2C%20tambi%C3%A9n%20conocido,empresariales%20y%20de%20marketing%20digital.>
Acceso: Mayo 2023
11. Figma
<https://www.figma.com/>

- Acceso: Mayo 2023
12. ¿Qué es el UX/UI Design? - NEOLAND
<https://www.neoland.es/que-es-el-ux-ui-design/>
Acceso: Marzo 2023
 13. UI y UX: ¿qué son y cómo se distinguen en el diseño web?
<https://rockcontent.com/es/blog/ui-ux/>
Acceso: Marzo 2023
 14. ¿Qué es una SPA? (Angular) – Programa en Línea
<https://www.programaenlinea.net/una-spa-angular/>
Acceso: Marzo 2023
 15. Angular - Creating libraries
<https://angular.io/guide/creating-libraries>
Acceso: Marzo 2023

 16. Usar clases e Interfaces en los servicios Angular
<https://desarrolloweb.com/articulos/clases-interfaces-servicios-angular.html>
Acceso: Marzo 2023
 17. ¿Qué es una etiqueta HTML y cuáles son las más importantes?
<https://www.iebschool.com/blog/que-es-etiqueta-html-analitica-usabilidad/>
Acceso: Marzo 2023
 18. Directivas en Angular
<https://medium.com/notasdeangular/directivas-en-angular-efb8a8cf78eo>
Acceso: Febrero 2023



Anexo

OBJETIVOS DE DESARROLLO SOSTENIBLE

El 25 de septiembre de 2015, los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible. Cada objetivo tiene metas específicas que deben alcanzarse en los próximos 15 años.

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza.			X	
ODS 2. Hambre cero.			X	
ODS 3. Salud y bienestar.			X	
ODS 4. Educación de calidad.			X	
ODS 5. Igualdad de género.			X	
ODS 6. Agua limpia y saneamiento.			X	
ODS 7. Energía asequible y no contaminante.			X	
ODS 8. Trabajo decente y crecimiento económico.	X			

ODS 9. Industria, innovación e infraestructuras.	X			
ODS 10. Reducción de las desigualdades.		X		
ODS 11. Ciudades y comunidades sostenibles.		X		
ODS 12. Producción y consumo responsables.		X		
ODS 13. Acción por el clima.			X	
ODS 14. Vida submarina.			X	
ODS 15. Vida de ecosistemas terrestres.			X	
ODS 16. Paz, justicia e instituciones sólidas.			X	
ODS 17. Alianzas para lograr objetivos.			X	

De los anteriores objetivos de desarrollo sostenibles mencionados, el proyecto relacionado está relacionado con:

1. **ODS 8 - Trabajo Decente y Crecimiento Económico:** Al proporcionar una herramienta eficiente para la construcción de sitios web de comercio electrónico, la librería de componentes de Angular puede contribuir al crecimiento económico y promover oportunidades de empleo en el sector tecnológico.
2. **ODS 9 - Industria, Innovación e Infraestructura:** Una librería de componentes de Angular puede fomentar la innovación en el desarrollo de sitios web de comercio electrónico, mejorando la eficiencia y la calidad de las soluciones tecnológicas.
3. **ODS 10 - Reducción de las desigualdades:** La librería de componentes de Angular puede ayudar a reducir las brechas tecnológicas al proporcionar herramientas accesibles y fáciles de usar para la construcción de sitios web de comercio electrónico, permitiendo que más personas y empresas puedan participar en el comercio en línea.
4. **ODS 11 - Ciudades y Comunidades Sostenibles:** La librería de componentes de Angular puede contribuir a la construcción de sitios web de comercio electrónico que sean accesibles, intuitivos y responsivos, mejorando así la experiencia de usuario y promoviendo entornos digitales sostenibles.
5. **ODS 12 - Producción y Consumo Responsables:** La librería de componentes de Angular puede promover la construcción de sitios web de comercio electrónico más sostenibles, facilitando la implementación de prácticas de diseño responsables y optimización de recursos.