

Plataforma de E-Commerce

Bienvenido a la Plataforma de E-Commerce! Este proyecto está construido de la siguiente manera: - Angular, Javascript, Typescript, HTML y CSS para el frontend. - Django y Python para el backend. - La BD esta configurada para PostgreSQL en su versión 16.0. - Utiliza un server Nginx para la seguridad de la red y el uso de SSL. - Está contenedorizado utilizando Docker y Docker Compose.

Tabla de Contenidos

- [Instalación](#)
 - [Instalando Docker Compose](#)
 - [Windows](#)
 - [macOS](#)
 - [Linux](#)
 - [Ejecutar los Servicios](#)
 - [Estructura del Proyecto](#)
 - [Tecnologías usadas](#)
 - [Descripción del Proyecto](#)

Instalación

Instalando Docker Compose

Windows

1. **Instalar Docker Desktop:**
 - Descarga e instala Docker Desktop desde [Docker Hub](#).
 - Sigue las instrucciones de instalación y asegúrate de que Docker Desktop esté ejecutándose.
2. **Verificar la Instalación:**
 - Abre el Símbolo del sistema (cmd) o PowerShell.
 - Ejecuta el siguiente comando para verificar la instalación:
`docker-compose --version`

macOS

1. **Instalar Docker Desktop:**
 - Descarga e instala Docker Desktop desde [Docker Hub](#).
 - Sigue las instrucciones de instalación y asegúrate de que Docker Desktop esté ejecutándose.
2. **Verificar la Instalación:**
 - Abre la Terminal.
 - Ejecuta el siguiente comando para verificar la instalación:
`docker-compose --version`

Linux

1. **Instalar Docker:**
 - Sigue las instrucciones para instalar Docker desde [la documentación oficial de Docker](#).
2. **Instalar Docker Compose:**
 - Ejecuta los siguientes comandos para descargar e instalar Docker Compose:

```
sudo curl -L "https://github.com/docker/compose/releases/download/  
sudo chmod +x /usr/local/bin/docker-compose
```

3. Verificar la Instalación:

- Ejecuta el siguiente comando para verificar la instalación:

```
docker-compose --version
```

Ejecutar los Servicios

Una vez que Docker Compose esté instalado, sigue estos pasos para iniciar los servicios:

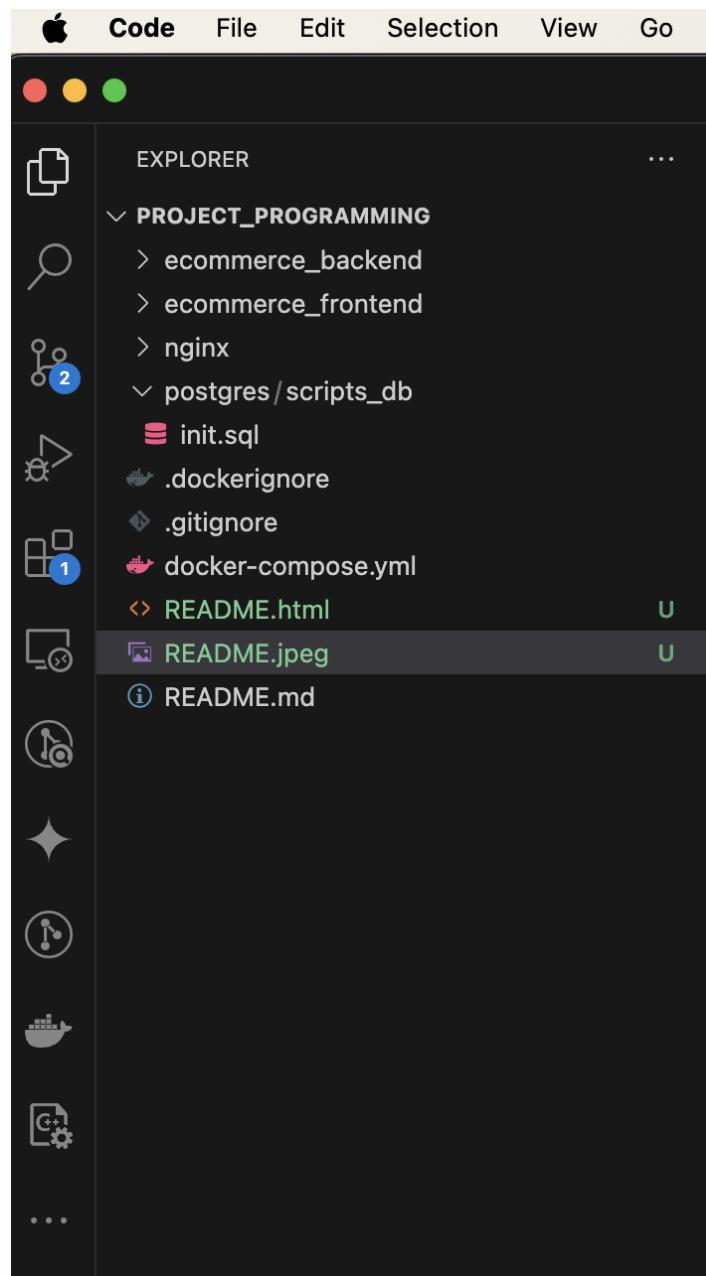
1. Clonar el Repositorio:

```
git clone https://github.com/rcglezreyes/project_programming.git  
cd project_programming
```

2. Ejecución del sistema:

Para levantar el sistema en ambiente docker solo tiene que ejecutar este comando desde la carpeta de la aplicación: sh docker-compose up -d --build Para detener el servicio: sh docker-compose down Finalmente accediendo a la URL: <https://127.0.0.1/> puede levantar la aplicación en ambiente docker

Estructura del proyecto



Estructura

Muestra las carpetas de las 4 imágenes del proyecto: backend, frontend, nginx y postgres En el caso de nginx, se deben generar los certificados si se correrá local. Estos son los comandos:
`openssl genpkey -algorithm RSA -out nginx/certificates/DNS/dns_key.pem`

```
openssl req -new -key nginx/certificates/DNS/dns_key.pem -x509 -days  
365 -out nginx/certificates/DNS/dns_cert.crt
```

```
openssl pkcs12 -export -out nginx/certificates/DNS/dns_cert.pfx -inkey  
nginx/certificates/DNS/dns_key.pem -in  
nginx/certificates/DNS/dns_cert.crt
```

```
openssl pkcs12 -in nginx/certificates/DNS/dns_cert.pfx -out  
nginx/certificates/DNS/dns_cert.pem -nodes -clcerts
```

```
openssl pkcs12 -in nginx/certificates/DNS/dns_cert.pfx -out  
nginx/certificates/DNS/dns_key.pem -nodes -nocerts
```

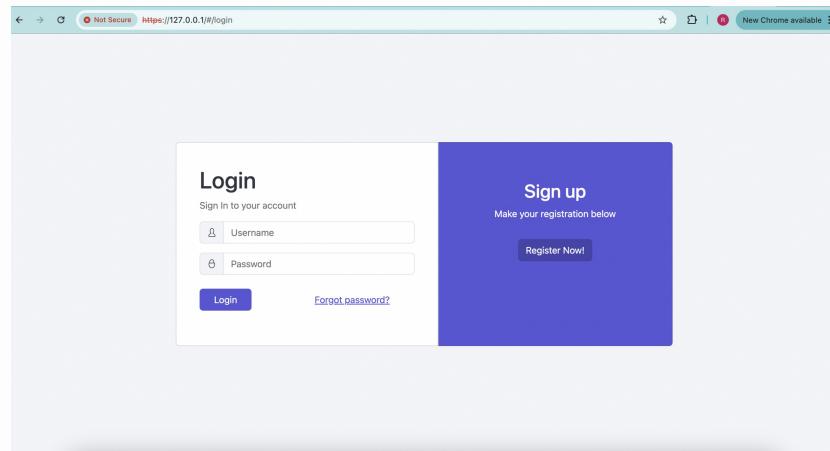
```
openssl pkcs12 -in nginx/certificates/DNS/dns_cert.pfx -out
```

```
nginx/certificates/DNS/dns_chain.pem -nodes -nokeys -clcerts
```

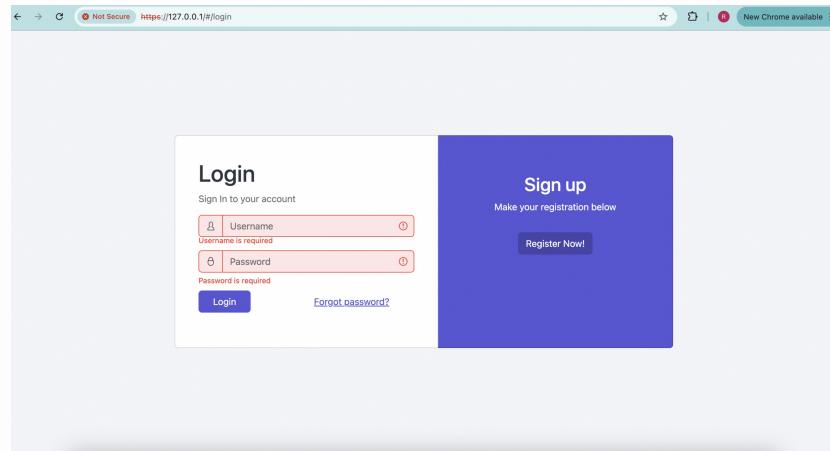
Tecnologías usadas

Lenguajes de Programación: HTML5, Javascript y Python
Frameworks: Angular y Django
Estilos: Material UI y CSS
Otras librerías: Sweetalert (Swal)

Descripción del proyecto



Esta es nuestra página de inicio, cuenta con el login, y el acceso mediante https protocol.



Se muestra un ejemplo de la validacion del login usando las librerias de validación de Material UI.

The screenshot shows a registration form titled "Register" with the sub-instruction "Create your account". The form consists of several input fields: First Name, Last Name, Username, Email, Password, Repeat password, Address, Phone, City, Zip Code, and Select country. Each field has a small icon next to it indicating its purpose. At the bottom are two buttons: "Create Account" and "Cancel".

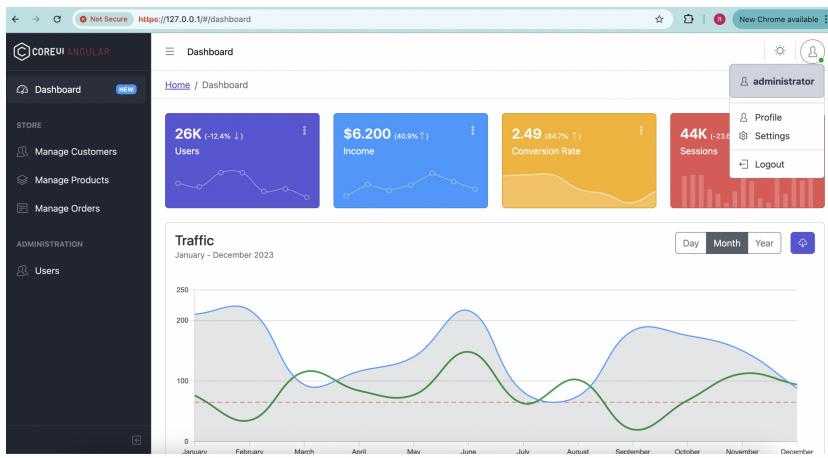
Se muestra el formulario desarrollado en Angular con Material UI para el registro

This screenshot shows the same registration form as above, but with all fields highlighted in red. Each field has a red border and a red error message below it. The errors correspond to the required validation rule: "First Name is required", "Last Name is required", "Username is required", "Email is required", "Password is required", "Repeat password is required", "Address is required", "Phone is required", "City is required", and "Zip Code is required". The "Select country" field also has a red border.

Se muestra un ejemplo de las validaciones de tipo required

This screenshot shows the registration form again, but with more complex validation errors. The "Email" field for "rcglezreyes@gmail.com" shows two error messages: "Email already exists in Users" and "Email already exists in Customers". The "Zip Code" field contains the value "sdfds" and has an error message: "Zip Code must be a positive integer". The "Password" and "Repeat password" fields both have the error message "Passwords do not match". The "Select country" dropdown is set to "United States".

Algunos ejemplos de validacion personalizadas: En el caso del email, ahí se observa como se realiza una subscripción a un endpoint de nuestro backend que devuelve la existencia o no del email en las tablas Customer y User. En el caso del zip code, se customizó una validación para que solo aceptara valores numéricos. En el caso de los campos passwords, se personalizó una validación para cuando ambos no coincidieran.



El admin del proyecto tendrá acceso a: Manage Customers, Manage Products, Manage Orders y User.

#	Image	Name	Price	Stock	Available	Category	Actions
1		Apple Newton Max	\$705.48	4	YES	Electronics	
2		Beef Vit	\$29.99	35	YES	Food	
3		Best Alarm System	\$99.14	10	YES	Electronics	
4		Best T-Shirt	\$10.79	107	YES	Clothing	

Módulo de Manage Products List

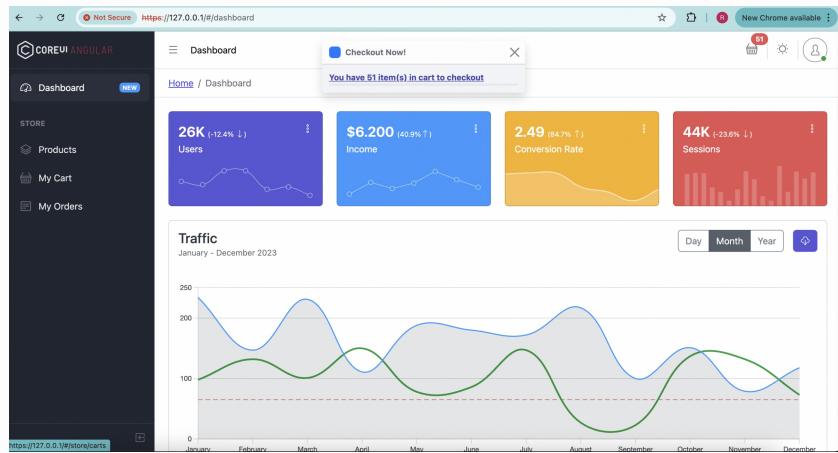
The screenshot shows the 'Add Product' form in the CoreUI Angular store. It includes fields for 'Name' (Apple Newton Max), 'Price' (\$705.48), 'Stock' (4), 'Category' (Select category), 'Description' (Apple Newton Max), and an 'Image' section with a file input field ('Choose File') and a preview area showing a smartphone image. There are 'Save' and 'Cancel' buttons at the bottom.

Módulo de Manage Products Add

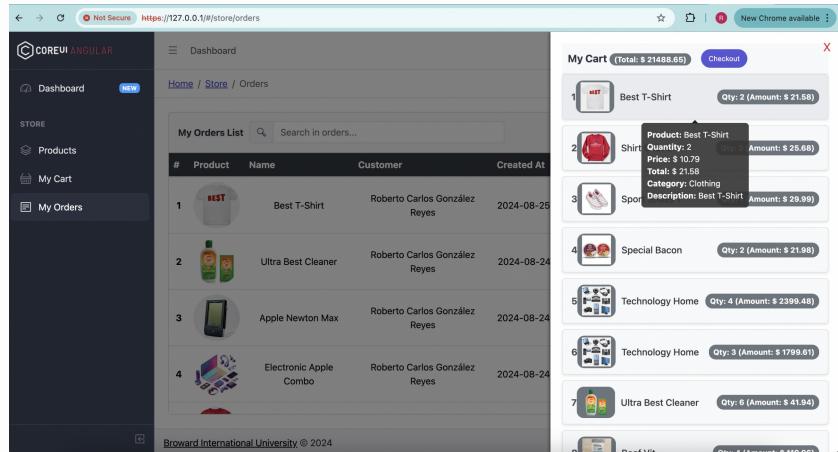
The screenshot shows the 'Edit Product' form for the 'Apple Newton Max' item. It displays the current values: Name (Apple Newton Max), Price (\$705.48), Stock (4), Category (Select category), Description (Apple Newton Max), and an 'Image' section with a file input field ('Choose File') and a preview area showing a smartphone image. There are 'Update' and 'Cancel' buttons at the bottom.

Módulo de Manage Products Edit

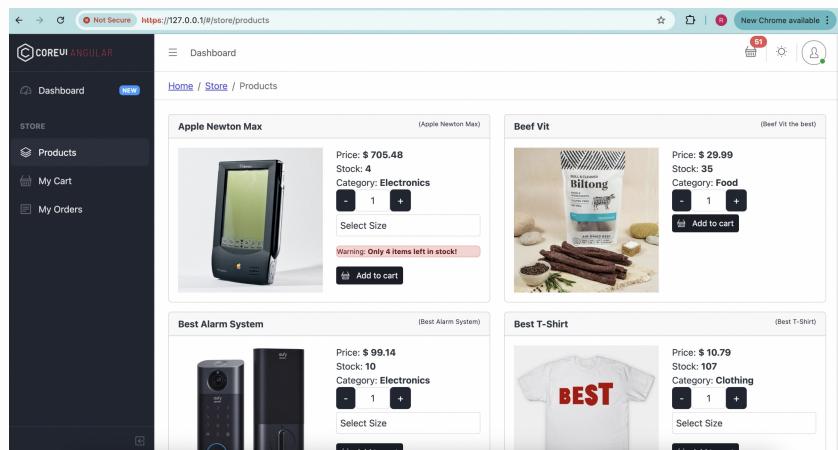
Mostramos un ejemplo del CRUD de Manage Products.



El customer del proyecto tendrá acceso a: Products (con otro formato), My Cart y My Orders. Se mostrará un Toast en el top-center de la página principal indicándole al customer que tiene artículos pendientes en su carrito (cart), con un link directo al checkout.



Aquí el customer podrá ver su lista de artículos en su carrito, así como el detalle de cada uno al hacer mouse over (tooltip) y el monto total de esa selección. También tendrá un link a checkout directamente.



Aquí se observa la vista de productos para los clientes, en este caso el cliente podrá seleccionar la cantidad (no más de lo que hay en stock), podrá seleccionar el size (en caso que el producto lo tenga) y le mostrará un warning para aquellos productos que tienen baja disponibilidad. En el botón Add to Cart, podrá añadirlo directamente a su carrito.

Dashboard

Home / Store / Products

My Carts List

Unselect	#	Product	Name	Created At	Quantity	Price	Amount	Actions
<input checked="" type="checkbox"/>	1		Best T-Shirt	2024-08-25 00:34	2	\$ 10.79	\$ 21.58	
<input checked="" type="checkbox"/>	2		Shirt Coat 2	2024-08-25 00:33	2	\$ 12.84	\$ 25.68	
<input checked="" type="checkbox"/>	3		Sport Shoes 1	2024-08-24 01:26	1	\$ 29.99	\$ 29.99	
<input checked="" type="checkbox"/>	4		Special Bacon	2024-08-22 23:32	2	\$ 10.99	\$ 21.98	

Total Amount: \$ 21488.65

Broward International University © 2024

Powered by Master Computer Sciences & Software Engineering

Módulo de Cart

Dashboard

Home / Store / Products

My Carts List

Unselect	#	Product	Name	Created At	Quantity	Price	Amount	Actions
<input checked="" type="checkbox"/>	1		Best T-Shirt	2024-08-25 00:34	2	\$ 10.79	\$ 21.58	
<input checked="" type="checkbox"/>	2		Shirt Coat 2	2024-08-25 00:33	2	\$ 12.84	\$ 25.68	
<input checked="" type="checkbox"/>	3		Sport Shoes 1	2024-08-24 01:26	1	\$ 29.99	\$ 29.99	
<input checked="" type="checkbox"/>	4		Special Bacon	2024-08-22 23:32	2	\$ 10.99	\$ 21.98	

Total Amount: \$ 21488.65

Are you sure?
You are about to checkout selected cart(s) for \$21488.65!

Yes, checkout! No, keep it

Broward International University © 2024

Powered by Master Computer Sciences & Software Engineering

Módulo de Cart 2

Aquí se observa la vista del carrito del customer, aquí en la primera columna puede seleccionar o deseleccionar todos, o uno por uno. El monto real se va reflejando dinámicamente en la parte superior. Y al dar click en checkout, los productos seleccionados automáticamente se convierten en orders.

Dashboard

Home / Store / Orders

My Orders List

#	Product	Name	Customer	Created At	Amount	Delivery At
1		Best T-Shirt	Roberto Carlos González Reyes	2024-08-25 12:58	\$ 32.37	6215 Monroe Place, West New York, 07093
2		Ultra Best Cleaner	Roberto Carlos González Reyes	2024-08-24 18:31	\$ 20.97	6215 Monroe Place, West New York, 07093
3		Apple Newton Max	Roberto Carlos González Reyes	2024-08-24 18:30	\$ 1410.96	6215 Monroe Place, West New York, 07093
4		Electronic Apple Combo	Roberto Carlos González Reyes	2024-08-24 18:28	\$ 3292.68	6215 Monroe Place, West New York, 07093

Total Paid: \$ 10033.94

Broward International University © 2024

Powered by Master Computer Sciences & Software Engineering

Aquí se observan las órdenes del customer y el monto total gastado en los productos que ha comprado.