

Documentación Carrito de compras backend API REST

Login

POST /auth/login

Ejemplo de request:

```
{
  "email": "userhash@example.com",
  "password": "123456"
}
```

Respuesta:

```
{
  "message": "Login exitoso",
  "token": "<JWT_TOKEN>",
  "userId": 6
}
```

Registro de Usuario

POST /auth/register

Ejemplo:

```
{
  "email": "usuario_nuevo@correo.com",
  "password": "123456",
  "rol": "usuario"
}
```

Respuesta:

```
{
  "message": "Usuario registrado"
}
```

Productos

Obtener todos los productos

GET `/api/productos`

Requiere token JWT

Respuesta:

```
[
  {
    "id": 1,
    "nombre": "celular",
    "precio": 1000
  }
]
```

Obtener producto por ID

GET `/api/productos/:id`

Respuesta:

```
{
  "id": 1,
  "nombre": "celular",
  "precio": 1000
}
```

Crear producto (sólo admin)

POST `/api/productos`

Ejemplo:

```
{
  "nombre": "tablet",
  "precio": 3000
}
```

Respuesta:

```
{
  "id": 2
}
```

Carrito

Agregar producto al carrito

POST `/api/carrito`

Requiere token JWT

Ejemplo:

```
{
  "producto_id": 200,
  "cantidad": 2
}
```

Respuesta:

```
{
  "message": "Producto agregado al carrito"
}
```

Listar productos del carrito

GET `/api/carrito`

Respuesta:

```
[
  {
    "id": 1,
    "nombre": "Smartwatch Deportivo GPS",
    "precio": 259,
    "cantidad": 2
  }
]
```

Eliminar producto del carrito por ID

DELETE `/api/carrito/:id`

Respuesta:

```
{
  "message": "Producto eliminado del carrito"
}
```

Vaciar carrito completo

DELETE /api/carrito

Respuesta:

```
{
  "message": "Carrito vaciado"
}
```

 **Compra**

POST /api/compra

Requiere token JWT

Respuesta:

```
{
  "message": "Compra finalizada con éxito",
  "total": 1528,
  "itemsComprados": 2,
  "detalles": [
    {
      "id": 6,
      "nombre": "Drone Plegable con Camara 4K",
      "precio": 979,
      "cantidad": 1
    },
    {
      "id": 7,
      "nombre": "Robot Aspirador con Mapeo",
      "precio": 549,
      "cantidad": 1
    }
  ]
}
```



Historial Compras

GET /api/compra

Requiere token JWT

Respuesta:

```
{
  "compras": [
    {
      "id": 5,
      "fecha": "2025-07-17 01:22:37",
      "total": 1499,
      "detalles": [
        {
          "producto_id": 207,
          "nombre": "Laptop Ultra Performance",
          "cantidad": 1,
          "precio_unitario": 1499
        }
      ]
    },
    {
      "id": 4,
      "fecha": "2025-07-14 18:25:06",
      "total": 129,
      "detalles": [
        {
          "producto_id": 250,
          "nombre": "Mochila Trekking 60 Litros",
          "cantidad": 1,
          "precio_unitario": 129
        }
      ]
    },
    {
      "id": 3,
      "fecha": "2025-07-14 17:58:07",
      "total": 668.99,
      "detalles": [
        {
          "producto_id": 208,
          "nombre": "Auriculares Gaming Inalambricos",
          "cantidad": 2,
          "precio_unitario": 279.5
        },
        {
          "producto_id": 210,
          "nombre": "Teclado Mecanico Retroiluminado",

```

```

        "cantidad": 1,
        "precio_unitario": 109.99
      }
    ],
  },
  {
    "id": 2,
    "fecha": "2025-07-14 17:46:34",
    "total": 729,
    "detalles": [
      {
        "producto_id": 209,
        "nombre": "Monitor Curvo 34 Pulgadas",
        "cantidad": 1,
        "precio_unitario": 729
      }
    ]
  }
]
}

```

Notas:

- Todas las rutas que modifican datos requieren token válido en `Authorization: Bearer <TOKEN>`
- Los usuarios con `rol: "admin"` pueden acceder a endpoints de creación, actualización y borrado de productos, usuarios y logs.
- Se puede usar el token guardado en `localStorage.getItem('authToken')` desde el frontend para hacer peticiones.