

Benemérita Universidad Autónoma de Puebla



Facultad de Ciencias de la Computación
Ingeniería en Tecnologías de la Información
Modelos de Desarrollo Web

Verano 2025

PhD Alfredo García Suárez

Guadalupe Tovar Cruz – 202163318

**Actividad 0 – Levantamiento de requerimientos y
análisis funcional**

API para la gestión de productos de un supermercado

La creación de esta **API** tiene como propósito mantener un enlace con la base de datos de un supermercado para poder gestionar su inventario. Esta **API** permitirá consultar, agregar, actualizar y eliminar productos.

Planteamiento (elementos que integran la API)

Por medio de los siguientes métodos se pueden realizar las modificaciones en el inventario.

Con **GET** podemos obtener la lista de productos o una descripción de este. **POST** para agregar un nuevo producto. **PUT** para actualizar un producto existente y **DELETE** para eliminar un producto.

Se integrará una autenticación por medio del uso de un token para que solo los empleados que estén autorizados puedan modificar el inventario.

Requerimientos

Requerimientos funcionales:

- Listar los productos disponibles.
- Consultar productos en específico.
- Agregar productos nuevos al inventario.
- Modificar información de un producto, ya sea su nombre, descripción, precio, etc.
- Eliminar productos del sistema.
- Autenticar usuarios
- Desconectar a un usuario después de un periodo de inactividad.

Requerimientos no funcionales:

- Usar tokens para garantizar la autenticación
- La API debe estar disponible en todo momento.

Análisis funcional

GET tiene como función consultar todos los productos o alguno en específico.

POST tiene como función agregar un nuevo producto al inventario.

PUT actualiza la información de un producto existente.

DELETE elimina un producto del inventario.

Documentación

1. GET `/productos`

Descripción: Obtiene la lista de productos.

Parámetros:

- *categoria* (opcional): filtra un producto por categoría.
- *disponible* (opcional): filtra un producto por su disponibilidad.

Respuesta:

```
1  [
2    {
3      "id": 1,
4      "nombre": "Leche entera",
5      "precio": 25.50,
6      "categoria": "Lácteos",
7      "cantidad": 40,
8      "disponible": true
9    }
10 ]
```

2. POST `/productos`

Descripción: Agrega un nuevo producto.

Cuerpo:

```
1  {
2    "nombre": "Pan integral",
3    "precio": 18.00,
4    "categoria": "Panadería",
5    "cantidad": 100,
6    "disponible": true
7  }
```

Respuesta:

```
1  {
2    "mensaje": "Producto creado exitosamente",
3    "id": 2
4  }
```

3. PUT /productos/{id}

Descripción: Actualiza los datos de un producto.

Cuerpo:

```
1  ✓ {  
2    |  "precio": 20.00,  
3    |  "cantidad": 80  
4    |  }
```

Respuesta:

```
1  {  
2    |  "mensaje": "Producto actualizado"  
3    |  }
```

4. DELETE /productos/{id}

Descripción: Elimina un producto.

Respuesta:

```
{  
  "mensaje": "Producto eliminado"  
}
```