

Documentación de API

Microservicio ISO/IEC 25010

Proyecto: Microservicio para Evaluación de Calidad de Software

Versión: 1.0.0

Fecha: Diciembre 2024

Autor: Estudiante Universidad Mariano Gálvez

Información General

Esta documentación describe los endpoints disponibles en el microservicio desarrollado con Spring Boot para la evaluación de calidad según la norma ISO/IEC 25010. La API sigue los principios REST y utiliza JSON como formato de intercambio de datos.

URL Base

```
http://localhost:8080/api
```

Autenticación

La API actualmente no implementa autenticación. Todos los endpoints son públicos.

Formatos de Respuesta

- **Éxito:** Código HTTP 2xx con cuerpo JSON
- **Error Cliente:** Código HTTP 4xx con cuerpo JSON de error
- **Error Servidor:** Código HTTP 5xx con cuerpo JSON de error

Formato de Error

```
{
  "timestamp": "2024-12-15T14:30:45.123Z",
  "status": 400,
  "error": "Bad Request",
  "message": "Descripción del error",
  "path": "/api/endpoint",
  "details": ["Error específico 1", "Error específico 2"]
}
```

Endpoints de Usuarios

Listar Todos los Usuarios

Endpoint: GET /usuarios

Descripción: Retorna una lista con todos los usuarios registrados en el sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
[
  {
    "id": 1,
    "nombre": "Juan Carlos",
    "apellido": "García López",
    "email": "juan.garcia@email.com",
    "telefono": "50212345678",
    "activo": true,
    "fechaCreacion": "2024-12-15T10:30:45.123Z",
    "fechaActualizacion": null,
    "nombreCompleto": "Juan Carlos García López",
    "totalPedidos": 2
  },
  {
    "id": 2,
    "nombre": "María Elena",
    "apellido": "Rodríguez Pérez",
    "email": "maria.rodriguez@email.com",
    "telefono": "50298765432",
    "activo": true,
    "fechaCreacion": "2024-12-15T11:20:15.456Z",
    "fechaActualizacion": null,
    "nombreCompleto": "María Elena Rodríguez Pérez",
    "totalPedidos": 1
  }
]
```

Obtener Usuario por ID

Endpoint: GET /usuarios/{id}

Descripción: Retorna un usuario específico basado en su ID.

Parámetros: - {id} - ID único del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
{
  "id": 1,
  "nombre": "Juan Carlos",
  "apellido": "García López",
  "email": "juan.garcia@email.com",
  "telefono": "50212345678",
  "activo": true,
  "fechaCreacion": "2024-12-15T10:30:45.123Z",
  "fechaActualizacion": null,
  "nombreCompleto": "Juan Carlos García López",
  "totalPedidos": 2
}
```

Respuesta de Error: - **Código:** 404 Not Found - **Contenido:**

```
{
  "timestamp": "2024-12-15T14:30:45.123Z",
  "status": 404,
  "error": "Not Found",
  "message": "Usuario no encontrado con ID: 999",
  "path": "/api/usuarios/999"
}
```

Crear Usuario

Endpoint: POST /usuarios

Descripción: Crea un nuevo usuario en el sistema con los datos proporcionados.

Cuerpo de la Solicitud:

```
{
  "nombre": "Pedro",
  "apellido": "González",
  "email": "pedro.gonzalez@test.com",
  "telefono": "50212345678"
}
```

Respuesta Exitosa: - Código: 201 Created - Contenido:

```
{
  "id": 11,
  "nombre": "Pedro",
  "apellido": "González",
  "email": "pedro.gonzalez@test.com",
  "telefono": "50212345678",
  "activo": true,
  "fechaCreacion": "2024-12-15T15:45:30.789Z",
  "fechaActualizacion": null,
  "nombreCompleto": "Pedro González",
  "totalPedidos": 0
}
```

Respuesta de Error: - Código: 400 Bad Request - Contenido:

```
{
  "timestamp": "2024-12-15T15:45:30.789Z",
  "status": 400,
  "error": "Bad Request",
  "message": "Error de validación en los datos enviados",
  "path": "/api/usuarios",
  "details": ["email: El formato del email no es válido"]
}
```

Actualizar Usuario

Endpoint: PUT /usuarios/{id}

Descripción: Actualiza los datos de un usuario existente.

Parámetros: - {id} - ID único del usuario (Path Parameter)

Cuerpo de la Solicitud:

```
{
  "nombre": "Pedro Luis",
  "apellido": "González Ramírez",
  "email": "pedro.gonzalez@test.com",
  "telefono": "50212345678"
}
```

Respuesta Exitosa: - Código: 200 OK - Contenido:

```
{
  "id": 11,
  "nombre": "Pedro Luis",
  "apellido": "González Ramírez",
  "email": "pedro.gonzalez@test.com",
  "telefono": "50212345678",
  "activo": true,
  "fechaCreacion": "2024-12-15T15:45:30.789Z",
  "fechaActualizacion": "2024-12-15T16:20:15.456Z",
  "nombreCompleto": "Pedro Luis González Ramírez",
  "totalPedidos": 0
}
```

Eliminar Usuario

Endpoint: DELETE /usuarios/{id}

Descripción: Elimina un usuario del sistema de forma permanente.

Parámetros: - {id} - ID único del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 204 No Content

Buscar Usuario por Email

Endpoint: GET /usuarios/email/{email}

Descripción: Busca un usuario específico por su dirección de email.

Parámetros: - {email} - Email del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto usuario (igual que en GET /usuarios/{id})

Buscar Usuarios por Nombre

Endpoint: GET /usuarios/buscar?nombre={nombre}

Descripción: Busca usuarios que contengan el texto especificado en su nombre.

Parámetros: - nombre - Texto a buscar en el nombre (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de usuarios (igual que en GET /usuarios)

Obtener Usuarios Activos

Endpoint: GET /usuarios/activos

Descripción: Retorna todos los usuarios que están activos en el sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de usuarios activos

Obtener Usuarios Inactivos

Endpoint: GET /usuarios/inactivos

Descripción: Retorna todos los usuarios que están inactivos en el sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de usuarios inactivos

Activar Usuario

Endpoint: PATCH /usuarios/{id}/activar

Descripción: Activa un usuario que estaba previamente inactivo.

Parámetros: - {id} - ID único del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto usuario actualizado

Desactivar Usuario

Endpoint: PATCH /usuarios/{id}/desactivar

Descripción: Desactiva un usuario sin eliminarlo del sistema.

Parámetros: - {id} - ID único del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto usuario actualizado

Búsqueda Libre de Usuarios

Endpoint: GET /usuarios/buscar-texto?texto={texto}

Descripción: Busca usuarios por texto libre en nombre, apellido o email.

Parámetros: - texto - Texto a buscar (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de usuarios que coinciden

Obtener Estadísticas de Usuarios

Endpoint: GET /usuarios/estadisticas

Descripción: Retorna estadísticas básicas sobre los usuarios del sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
{
  "total": 10,
  "activos": 9,
  "inactivos": 1,
  "porcentajeActivos": 90.0
}
```

Endpoints de Productos

Listar Todos los Productos

Endpoint: GET /productos

Descripción: Retorna una lista con todos los productos del catálogo.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
[
  {
    "id": 1,
    "nombre": "Smartphone Samsung Galaxy A54",
    "descripcion": "Teléfono inteligente con pantalla AMOLED de 6.4 pulgadas...",
    "precio": 2499.99,
    "stock": 25,
    "categoria": "Electrónicos",
    "marca": "Samsung",
    "activo": true,
    "fechaCreacion": "2024-12-15T10:30:45.123Z",
    "fechaActualizacion": null
  },
  {
    "id": 2,
    "nombre": "Laptop Dell Inspiron 15",
    "descripcion": "Laptop con procesador Intel Core i5, 8GB RAM...",
    "precio": 4299.99,
    "stock": 15,
    "categoria": "Electrónicos",
    "marca": "Dell",
    "activo": true,
    "fechaCreacion": "2024-12-15T10:35:20.456Z",
    "fechaActualizacion": null
  }
]
```

Obtener Producto por ID

Endpoint: GET /productos/{id}

Descripción: Retorna un producto específico basado en su ID.

Parámetros: - {id} - ID único del producto (Path Parameter)

Respuesta Exitosa: - Código: 200 OK - Contenido:

```
{
  "id": 1,
  "nombre": "Smartphone Samsung Galaxy A54",
  "descripcion": "Teléfono inteligente con pantalla AMOLED de 6.4 pulgadas...",
  "precio": 2499.99,
  "stock": 25,
  "categoria": "Electrónicos",
  "marca": "Samsung",
  "activo": true,
  "fechaCreacion": "2024-12-15T10:30:45.123Z",
  "fechaActualizacion": null
}
```


Crear Producto

Endpoint: POST /productos

Descripción: Crea un nuevo producto en el catálogo con los datos proporcionados.

Cuerpo de la Solicitud:

```
{
  "nombre": "Smartphone Test Pro",
  "descripcion": "Teléfono de prueba con características avanzadas",
  "precio": 1599.99,
  "stock": 50,
  "categoria": "Electrónicos",
  "marca": "TestBrand"
}
```

Respuesta Exitosa: - **Código:** 201 Created - **Contenido:** Objeto producto creado

Actualizar Producto

Endpoint: PUT /productos/{id}

Descripción: Actualiza los datos de un producto existente.

Parámetros: - {id} - ID único del producto (Path Parameter)

Cuerpo de la Solicitud:

```
{
  "nombre": "Smartphone Test Pro Plus",
  "descripcion": "Versión mejorada del teléfono de prueba",
  "precio": 1799.99,
  "stock": 45,
  "categoria": "Electrónicos",
  "marca": "TestBrand"
}
```

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto producto actualizado

Eliminar Producto

Endpoint: DELETE /productos/{id}

Descripción: Elimina un producto del catálogo de forma permanente.

Parámetros: - `{id}` - ID único del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 204 No Content

Buscar Productos por Nombre

Endpoint: `GET /productos/buscar?nombre={nombre}`

Descripción: Busca productos que contengan el texto especificado en su nombre.

Parámetros: - `nombre` - Texto a buscar en el nombre (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos que coinciden

Buscar Productos por Categoría

Endpoint: `GET /productos/categoria/{categoria}`

Descripción: Retorna todos los productos de una categoría específica.

Parámetros: - `{categoria}` - Categoría del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos de la categoría

Buscar Productos por Marca

Endpoint: `GET /productos/marca/{marca}`

Descripción: Retorna todos los productos de una marca específica.

Parámetros: - `{marca}` - Marca del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos de la marca

Obtener Productos Activos

Endpoint: `GET /productos/activos`

Descripción: Retorna todos los productos que están activos en el catálogo.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos activos

Obtener Productos con Stock

Endpoint: GET /productos/con-stock

Descripción: Retorna todos los productos que tienen stock disponible.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos con stock > 0

Obtener Productos sin Stock

Endpoint: GET /productos/sin-stock

Descripción: Retorna todos los productos que no tienen stock disponible.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos con stock = 0

Buscar Productos por Rango de Precios

Endpoint: GET /productos/precio?precioMinimo={min}&precioMaximo={max}

Descripción: Busca productos dentro de un rango de precios específico.

Parámetros: - precioMinimo - Precio mínimo (Query Parameter) - precioMaximo - Precio máximo (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos en el rango de precios

Búsqueda Libre de Productos

Endpoint: GET /productos/buscar-texto?texto={texto}

Descripción: Busca productos por texto libre en nombre, descripción, categoría o marca.

Parámetros: - texto - Texto a buscar (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de productos que coinciden

Obtener Todas las Categorías

Endpoint: GET /productos/categorias

Descripción: Retorna una lista con todas las categorías de productos disponibles.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
[  
  "Electrónicos",  
  "Hogar y Jardín",  
  "Deportes y Fitness",  
  "Libros y Educación"  
]
```

Obtener Todas las Marcas

Endpoint: GET /productos/marcas

Descripción: Retorna una lista con todas las marcas de productos disponibles.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de strings con nombres de marcas

Actualizar Stock de Producto

Endpoint: PATCH /productos/{id}/stock?stock={stock}

Descripción: Actualiza la cantidad de stock disponible de un producto.

Parámetros: - {id} - ID único del producto (Path Parameter) - stock - Nuevo stock (Query Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto producto actualizado

Activar Producto

Endpoint: `PATCH /productos/{id}/activar`

Descripción: Activa un producto que estaba previamente inactivo.

Parámetros: - `{id}` - ID único del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto producto actualizado

Desactivar Producto

Endpoint: `PATCH /productos/{id}/desactivar`

Descripción: Desactiva un producto sin eliminarlo del catálogo.

Parámetros: - `{id}` - ID único del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto producto actualizado

Endpoints de Pedidos

Listar Todos los Pedidos

Endpoint: `GET /pedidos`

Descripción: Retorna una lista con todos los pedidos del sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
[
  {
    "id": 1,
    "usuario": {
      "id": 1,
      "nombre": "Juan Carlos",
      "apellido": "García López",
      "email": "juan.garcia@email.com"
    },
    "producto": {
      "id": 1,
      "nombre": "Smartphone Samsung Galaxy A54",
      "precio": 2499.99
    },
    "cantidad": 2,
    "precioUnitario": 2499.99,
    "total": 4999.98,
    "estado": "PENDIENTE",
    "observaciones": "Pedido para regalo de cumpleaños",
    "fechaPedido": "2024-12-15T10:40:15.789Z",
    "fechaEntrega": null,
    "fechaActualizacion": null
  },
  {
    "id": 2,
    "usuario": {
      "id": 2,
      "nombre": "María Elena",
      "apellido": "Rodríguez Pérez",
      "email": "maria.rodriguez@email.com"
    },
    "producto": {
      "id": 5,
      "nombre": "Smart TV LG 55 pulgadas",
      "precio": 5999.99
    },
    "cantidad": 1,
    "precioUnitario": 5999.99,
    "total": 5999.99,
    "estado": "PENDIENTE",
    "observaciones": "Entrega preferiblemente en horario matutino",
    "fechaPedido": "2024-12-15T11:20:30.456Z",
    "fechaEntrega": null,
    "fechaActualizacion": null
  }
]
```

Obtener Pedido por ID

Endpoint: GET /pedidos/{id}

Descripción: Retorna un pedido específico basado en su ID.

Parámetros: - {id} - ID único del pedido (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto pedido detallado

Crear Pedido

Endpoint: POST /pedidos

Descripción: Crea un nuevo pedido con los datos proporcionados.

Cuerpo de la Solicitud:

```
{
  "usuarioId": 1,
  "productoId": 1,
  "cantidad": 3,
  "observaciones": "Pedido de prueba para validación de stock"
}
```

Respuesta Exitosa: - **Código:** 201 Created - **Contenido:** Objeto pedido creado

Respuesta de Error: - **Código:** 400 Bad Request - **Contenido:**

```
{
  "timestamp": "2024-12-15T15:45:30.789Z",
  "status": 400,
  "error": "Bad Request",
  "message": "Stock insuficiente. Stock disponible: 2",
  "path": "/api/pedidos"
}
```

Actualizar Pedido

Endpoint: PUT /pedidos/{id}

Descripción: Actualiza los datos de un pedido existente.

Parámetros: - {id} - ID único del pedido (Path Parameter)

Cuerpo de la Solicitud:

```
{
  "usuarioId": 1,
  "productoId": 2,
  "cantidad": 1,
  "observaciones": "Pedido actualizado"
}
```

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto pedido actualizado

Eliminar Pedido

Endpoint: DELETE /pedidos/{id}

Descripción: Elimina un pedido del sistema y restaura el stock.

Parámetros: - {id} - ID único del pedido (Path Parameter)

Respuesta Exitosa: - **Código:** 204 No Content

Obtener Pedidos por Usuario

Endpoint: GET /pedidos/usuario/{usuarioId}

Descripción: Retorna todos los pedidos de un usuario específico.

Parámetros: - {usuarioId} - ID del usuario (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de pedidos del usuario

Obtener Pedidos por Producto

Endpoint: GET /pedidos/producto/{productoId}

Descripción: Retorna todos los pedidos de un producto específico.

Parámetros: - {productoId} - ID del producto (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de pedidos del producto

Obtener Pedidos por Estado

Endpoint: GET /pedidos/estado/{estado}

Descripción: Retorna todos los pedidos con un estado específico.

Parámetros: - {estado} - Estado del pedido (Path Parameter) - Valores válidos: PENDIENTE, CONFIRMADO, EN_PROCESO, ENVIADO, ENTREGADO, CANCELADO

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Array de pedidos con el estado especificado

Cambiar Estado de Pedido

Endpoint: PATCH /pedidos/{id}/estado?estado={estado}

Descripción: Actualiza el estado de un pedido específico.

Parámetros: - {id} - ID único del pedido (Path Parameter) - estado - Nuevo estado (Query Parameter) - Valores válidos: PENDIENTE, CONFIRMADO, EN_PROCESO, ENVIADO, ENTREGADO, CANCELADO

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto pedido actualizado

Cancelar Pedido

Endpoint: PATCH /pedidos/{id}/cancelar

Descripción: Cancela un pedido y restaura el stock del producto.

Parámetros: - {id} - ID único del pedido (Path Parameter)

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:** Objeto pedido cancelado

Respuesta de Error: - **Código:** 400 Bad Request - **Contenido:**

```
{
  "timestamp": "2024-12-15T16:30:45.123Z",
  "status": 400,
  "error": "Bad Request",
  "message": "El pedido no puede ser cancelado. Estado actual: ENTREGADO",
  "path": "/api/pedidos/5/cancelar"
}
```

Obtener Estadísticas de Pedidos

Endpoint: GET /pedidos/estadisticas

Descripción: Retorna estadísticas generales sobre los pedidos del sistema.

Parámetros: Ninguno

Respuesta Exitosa: - **Código:** 200 OK - **Contenido:**

```
{
  "total": 16,
  "pendientes": 3,
  "confirmados": 3,
  "entregados": 4,
  "cancelados": 2,
  "ventasTotal": 26699.85
}
```

Modelos de Datos

Usuario

```
{
  "id": 1,
  "nombre": "Juan Carlos",
  "apellido": "García López",
  "email": "juan.garcia@email.com",
  "telefono": "50212345678",
  "activo": true,
  "fechaCreacion": "2024-12-15T10:30:45.123Z",
  "fechaActualizacion": null,
  "nombreCompleto": "Juan Carlos García López",
  "totalPedidos": 2
}
```

Producto

```
{
  "id": 1,
  "nombre": "Smartphone Samsung Galaxy A54",
  "descripcion": "Teléfono inteligente con pantalla AMOLED de 6.4 pulgadas...",
  "precio": 2499.99,
  "stock": 25,
  "categoria": "Electrónicos",
  "marca": "Samsung",
  "activo": true,
  "fechaCreacion": "2024-12-15T10:30:45.123Z",
  "fechaActualizacion": null
}
```

Pedido

```
{
  "id": 1,
  "usuario": {
    "id": 1,
    "nombre": "Juan Carlos",
    "apellido": "García López",
    "email": "juan.garcia@email.com"
  },
  "producto": {
    "id": 1,
    "nombre": "Smartphone Samsung Galaxy A54",
    "precio": 2499.99
  },
  "cantidad": 2,
  "precioUnitario": 2499.99,
  "total": 4999.98,
  "estado": "PENDIENTE",
  "observaciones": "Pedido para regalo de cumpleaños",
  "fechaPedido": "2024-12-15T10:40:15.789Z",
  "fechaEntrega": null,
  "fechaActualizacion": null
}
```

Estados de Pedido

- **PENDIENTE:** Pedido recién creado, pendiente de confirmación
- **CONFIRMADO:** Pedido confirmado, listo para procesamiento
- **EN_PROCESO:** Pedido en proceso de preparación
- **ENVIADO:** Pedido enviado al cliente
- **ENTREGADO:** Pedido entregado exitosamente
- **CANCELADO:** Pedido cancelado

Códigos de Estado HTTP

- **200 OK:** Operación exitosa
- **201 Created:** Recurso creado exitosamente
- **204 No Content:** Operación exitosa sin contenido de respuesta
- **400 Bad Request:** Error en los datos enviados
- **404 Not Found:** Recurso no encontrado

- **409 Conflict:** Conflicto con el estado actual del recurso
 - **500 Internal Server Error:** Error interno del servidor
-

Herramientas de Desarrollo

Swagger UI

La documentación interactiva de la API está disponible en:

```
http://localhost:8080/api/swagger-ui.html
```

H2 Console

La consola de la base de datos H2 está disponible en:

```
http://localhost:8080/api/h2-console
```

Credenciales: - **JDBC URL:** jdbc:h2:mem:testdb - **Username:** sa - **Password:** password

Documento generado para: Universidad Mariano Gálvez

Curso: Aseguramiento de la Calidad de Software

Proyecto: Microservicio ISO/IEC 25010

Fecha: Diciembre 2024