

DIRECCIÓN ZONAL

Av.Centenario KM 4.500

FORMACIÓN PROFESIONAL

CFP/UCP/ESCUELA: SENATI

ID: 1513574 \_\_\_\_\_\_\_\_ BLOQUE: PIAD 503 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

CARRERA: INGENIERIA DE SOFTWARE CON IA

INSTRUCTOR: ELIO DAVID HOYOS BARBA

SEMESTRE: V DEL: 10-02-2025 AL: 31-06-2025

Informe de Implementación de Microservicios

Proyecto: Todo Barato S.A.C - API REST

1. Introducción

La empresa Todo Barato S.A.C ha solicitado la implementación de una API RESTful para la gestión de su inventario, autenticación de usuarios y operaciones CRUD sobre los datos de productos, marcas y clientes. Este informe detalla la arquitectura, microservicios utilizados, tecnología implementada y políticas de autenticación mediante JWT.

1. Base de Datos

Nombre del esquema: system-inventory

Tablas implementadas:

products

-name

-description

-expiration\_date

-quantity

-code

brand

-name

-description

client

-name

-document\_type

-document

-address

-cell\_phone

login

-email

-password (encriptada)

-returnSecureToken (booleano)

1. Tecnologías Utilizadas

|  |  |
| --- | --- |
| Componente | Tecnología |
| Autenticación | JWT (JSON Web Tokens) |
| Base de Datos | FireBase / RealTime DataBase |
| Seguridad | Bcrypt (para encriptar contraseñas) |
| Testing/API | Postman (Collection y Environment) |
| Control de acceso | Middleware personalizado con JWT |

1. Autenticación y Autorización

Para garantizar la seguridad en el acceso a los servicios del sistema, se ha implementado un mecanismo de autenticación y autorización utilizando JWT (JSON Web Tokens). Este sistema permite verificar la identidad del usuario mediante un token único generado al momento del inicio de sesión. Dicho token debe ser enviado en cada petición protegida para acceder a los recursos del sistema.

Se han definido dos tipos de usuarios dentro del sistema. El primer usuario es admin@todobarato.com, quien posee permisos de superusuario. Este usuario tiene acceso completo a todos los métodos disponibles en la API, incluyendo GET, POST, PUT, PATCH y DELETE, lo que le permite realizar operaciones de lectura, creación, modificación parcial o total, y eliminación de recursos.

El segundo usuario definido es editar@todobarato.com, quien representa un usuario sin autorización completa. Aunque este usuario puede autenticarse correctamente y obtener un token válido, cualquier intento de acceder a los endpoints protegidos de la API resultará en una respuesta con código de estado 401 Unauthorized, indicando que carece de los permisos necesarios para realizar dichas acciones. Este comportamiento permite mantener un control estricto sobre las operaciones críticas dentro del sistema y asegurar que solo los usuarios autorizados tengan acceso a funcionalidades sensibles.

Este enfoque basado en JWT permite una gestión flexible y segura de los permisos de los usuarios, manteniendo la integridad y confidencialidad de la información en Todo Barato S.A.C.

1. Microservicios Implementados

Cada recurso (producto, marca, cliente, login) está implementado como un microservicio REST:

Product Microservice

GET /products/id – Listar todos los productos

POST /products/id – Crear nuevo producto

PUT /products/:id – Reemplazar un producto existente

PATCH /products/:id – Modificar parcialmente un producto

DELETE /products/:id – Eliminar un producto

Brand Microservice

GET /brands/id – Listar marcas

POST /brands/id – Agregar nueva marca

PUT /brand/:id – Reemplazar una marca existente

PATCH /brand/:id – Modificar parcialmente una marca

DELETE /brand/:id – Eliminar una marca

Client Microservice

GET /clients/id – Listar clientes

POST /clients/id – Registrar nuevo cliente

LOGIN

POST {{Login}}/cliente?auth=idToken

POST {{crear }} la variable de para crear usuarios y el id de firebase

Middleware para validar token JWT en cada petición protegida

1. Postman

Se ha generado una Collection de Postman llamada "Todo Barato S.A.C" junto con un Environment llamado "Endpoint Todo Barato S.A.C", el cual incluye variables como:

-url

-crear

-login

Esto permite simular el flujo completo desde login hasta consumo de endpoints con token JWT incluido.

LINK

Crear:

https://identitytoolkit.googleapis.com/v1/accounts:signUp?key=[API\_KEY]

login:

https://identitytoolkit.googleapis.com/v1/accounts:signInWithPassword?key=[API\_KEY]

la api\_key tienes que obtenerlo de ti firebase

1. Manejo de Seguridad

Se usa bcrypt para el almacenamiento seguro de contraseñas.

Validación de email y token para asegurar accesos restringidos.

Middleware para verificación del JWT y roles autorizados.

Política de retorno de 401 Unauthorized para usuarios sin permisos.

1. Pruebas y Validación

Se probó acceso completo con admin@todobarato.com → acceso completo a todos los endpoints.

Se intentó acceso con editar@todobarato.com → acceso denegado (401) en todas las rutas protegidas.

Verificación de expiración y validación de tokens correcta.

1. Conclusión

La API REST para Todo Barato S.A.C está correctamente estructurada en microservicios modulares, seguros y escalables. La autenticación JWT y el control de acceso garantizan que solo usuarios autorizados puedan manipular los datos, cumpliendo con los requerimientos del proyecto.