

## Instrucciones Instalación MicroServicios Authorization Server y Ms Products

1. Instalar postgresql 14+, bases de datos para authorization-server

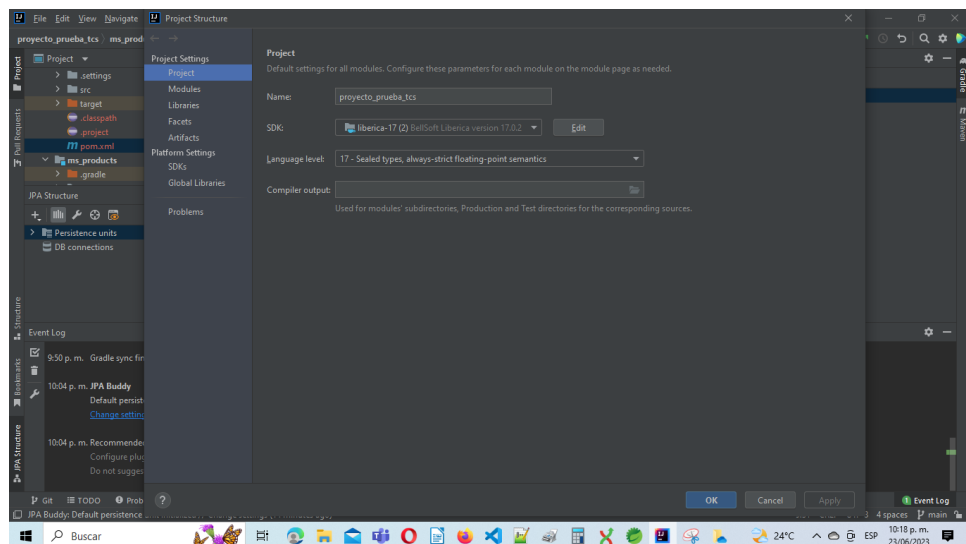
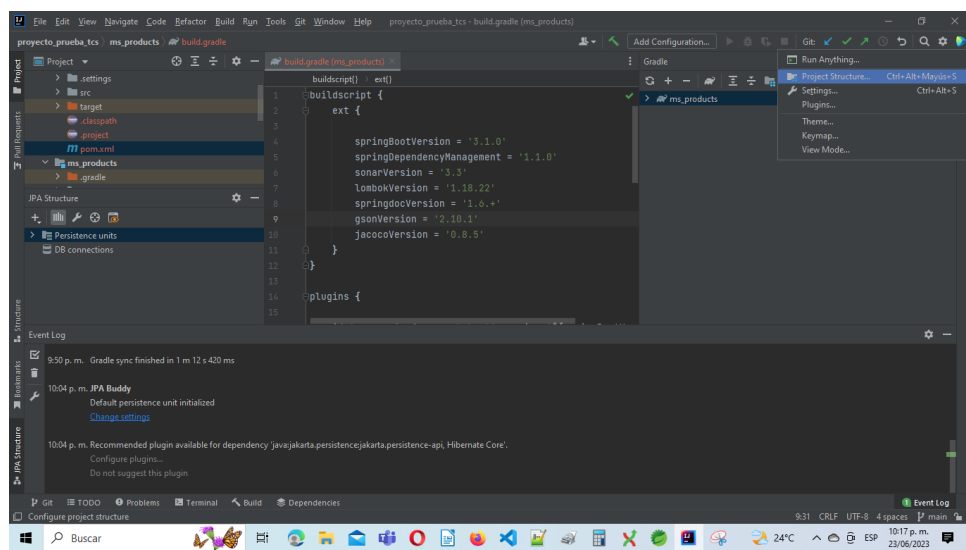
Usuario: postgres

Password: admin

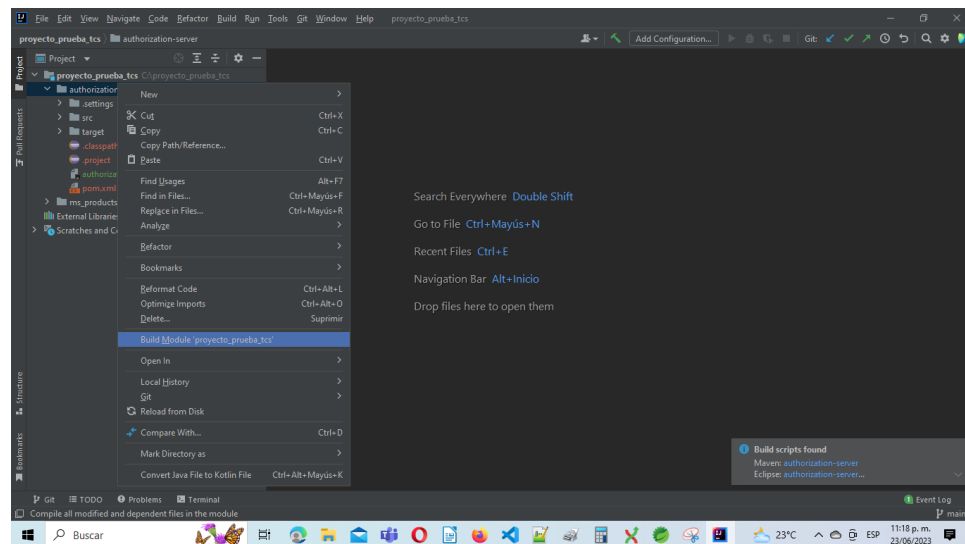
Puerto: 7777

DataBase: authorization-server

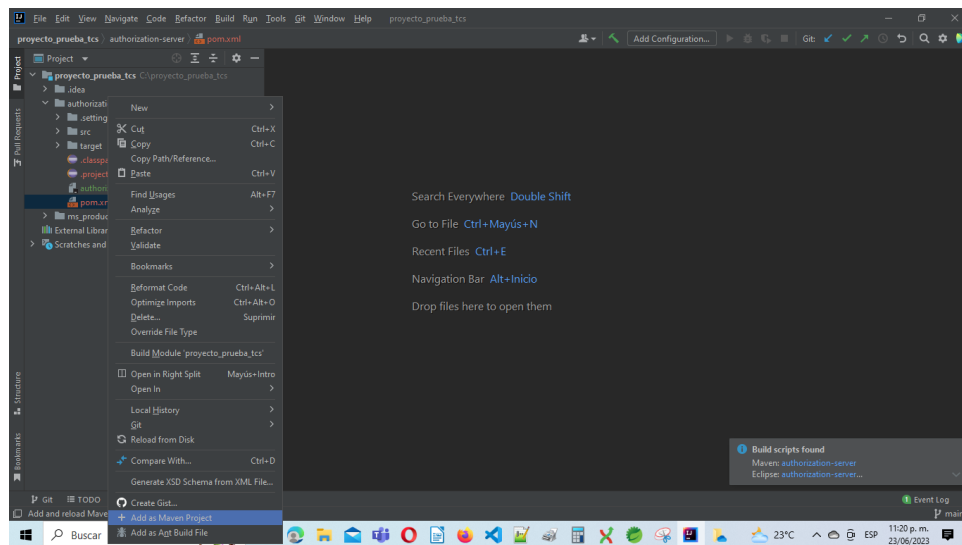
2. Abrir IntelliJIde: Abrir la carpeta C:\proyecto\_prueba\_tcs
3. Configurar SDK



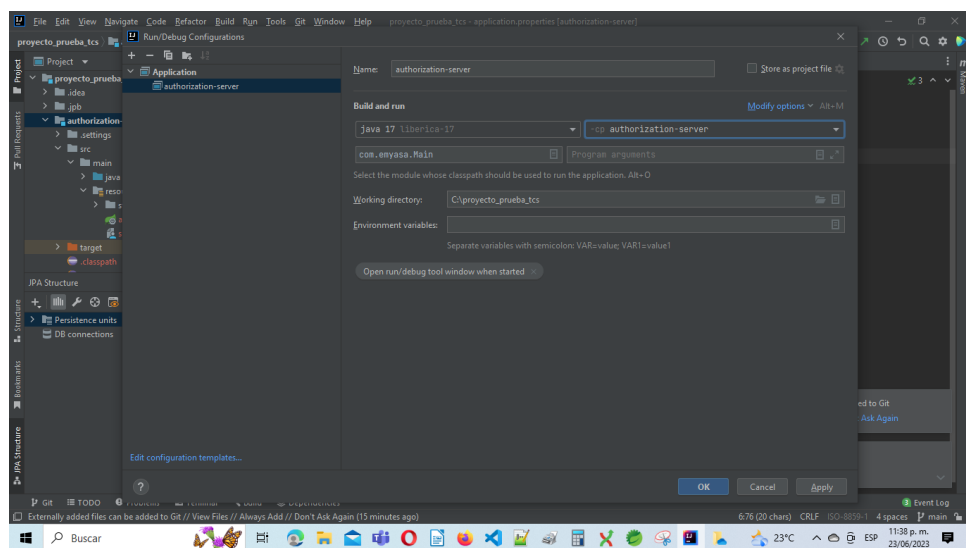
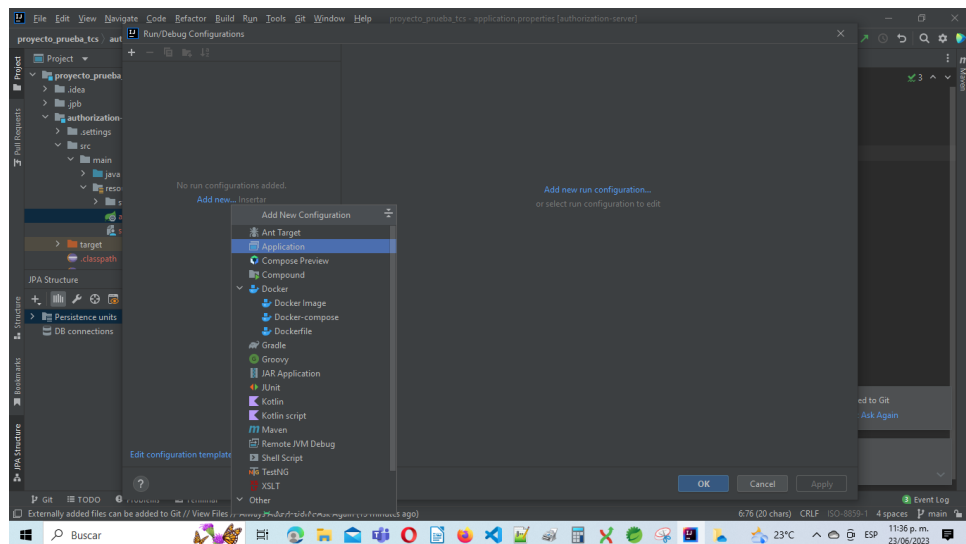
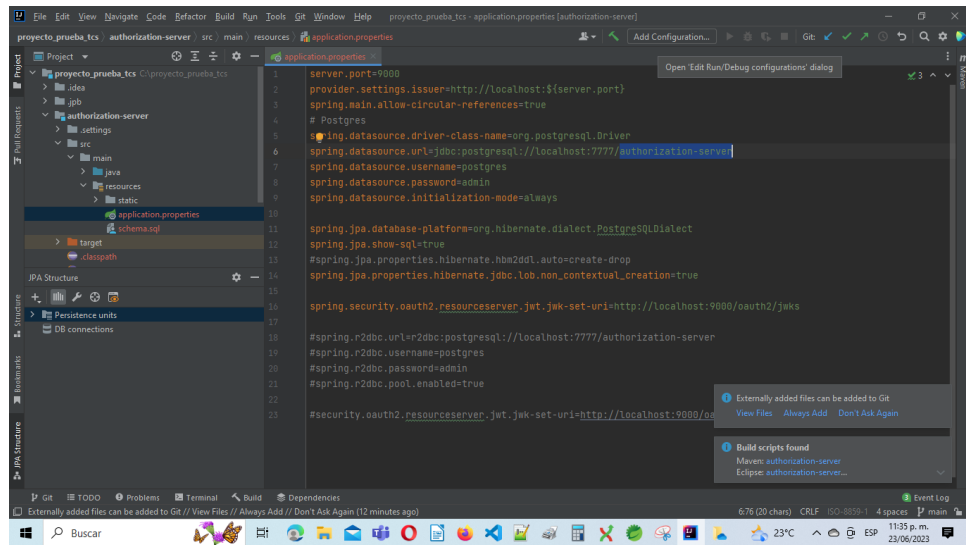
#### 4. Generar Module authorization-server



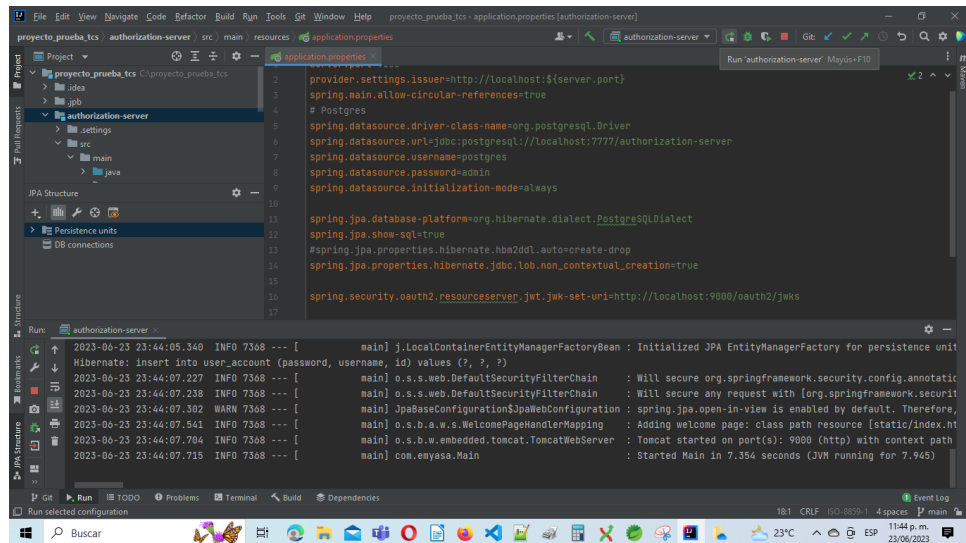
#### 5. Add as Maven Project authorization-server



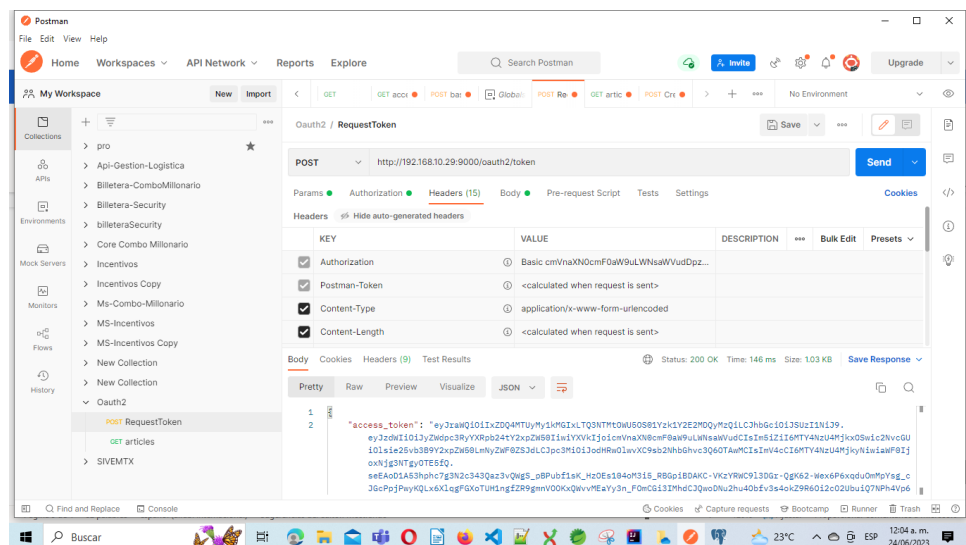
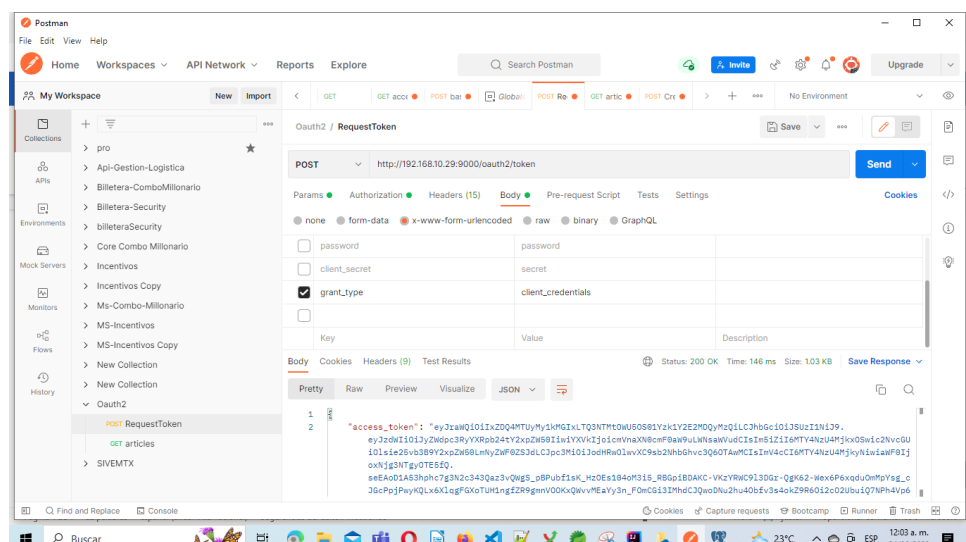
## 6. Add Configuration

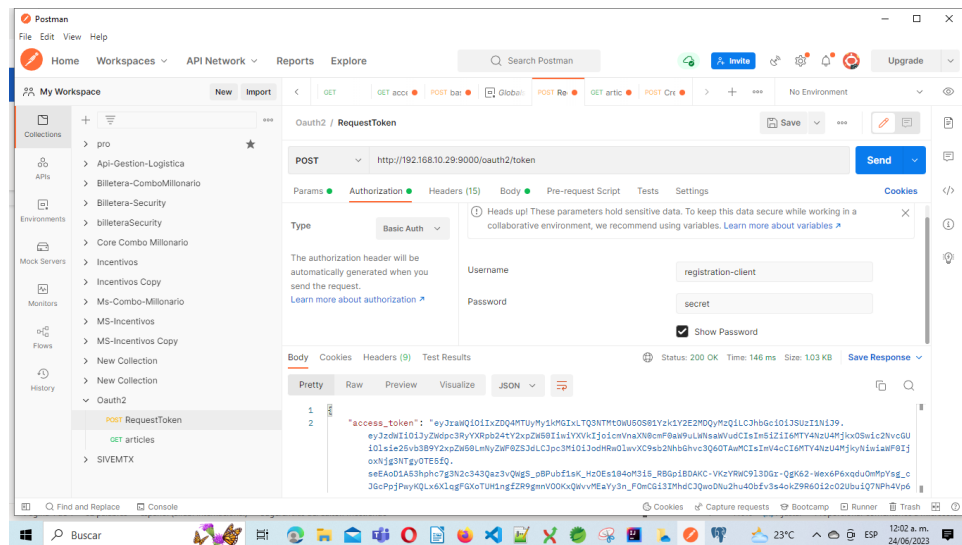


## 7. Run authorization-server



## 8. Pruebas con Postman Solicitud Token





## 9. Instalar postgresql 14+, bases de datos para ms-products

Usuario: postgres

Password: admin

Puerto: 7777

DataBase: db\_products

Generar Script en Consola de psql

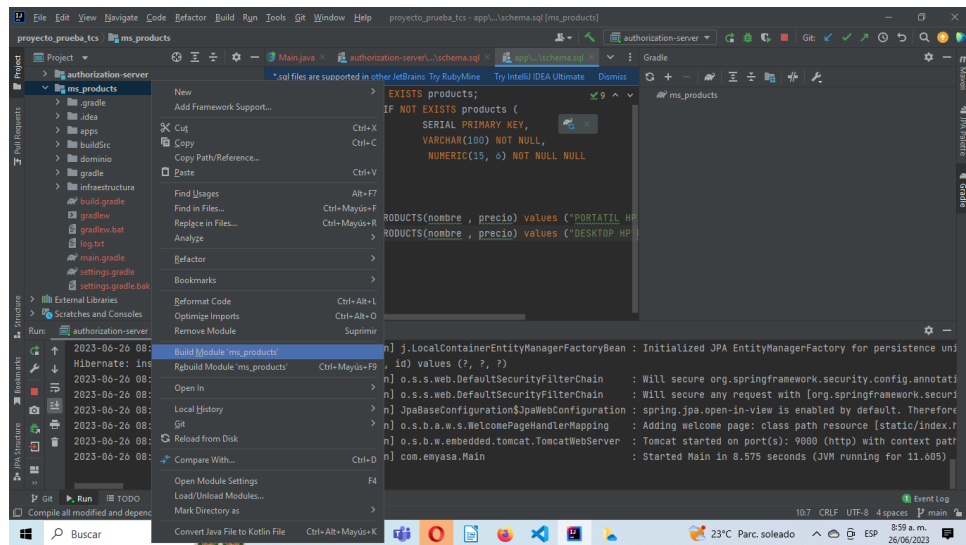
- Psql -U postgres -p 7777 pide contraseña = admin
- Conectar base de datos \connect db\_products
- Pegar script

```

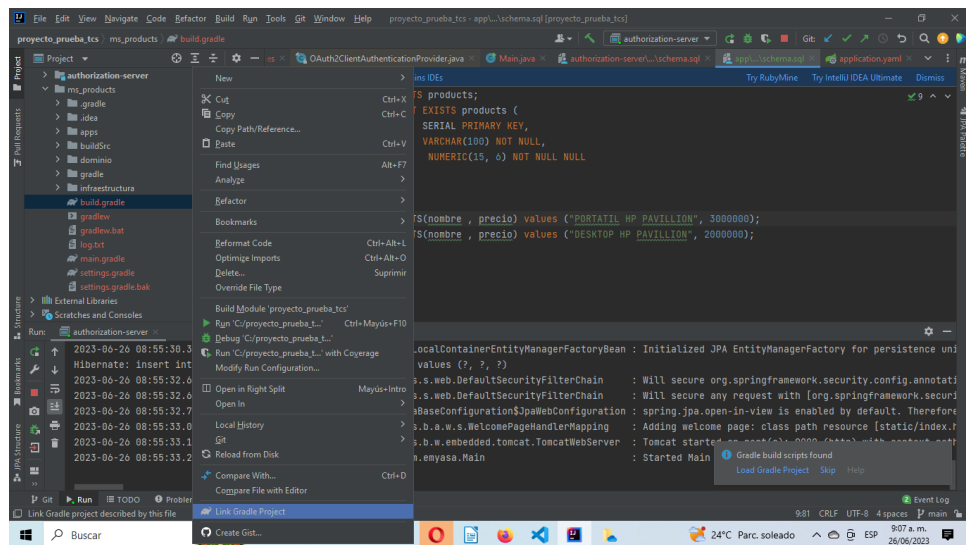
DROP TABLE IF EXISTS products;
CREATE TABLE IF NOT EXISTS products (
    id                BIGSERIAL PRIMARY KEY,
    nombre            VARCHAR(100) NOT NULL,
    precio            NUMERIC(15, 6) NOT NULL

);
INSERT INTO PRODUCTS(nombre , precio) values ('PORTATIL HP
PAVILLION',          3000000);
INSERT INTO PRODUCTS(nombre , precio) values ('DESKTOP HP PAVILLION',
2000000);
  
```

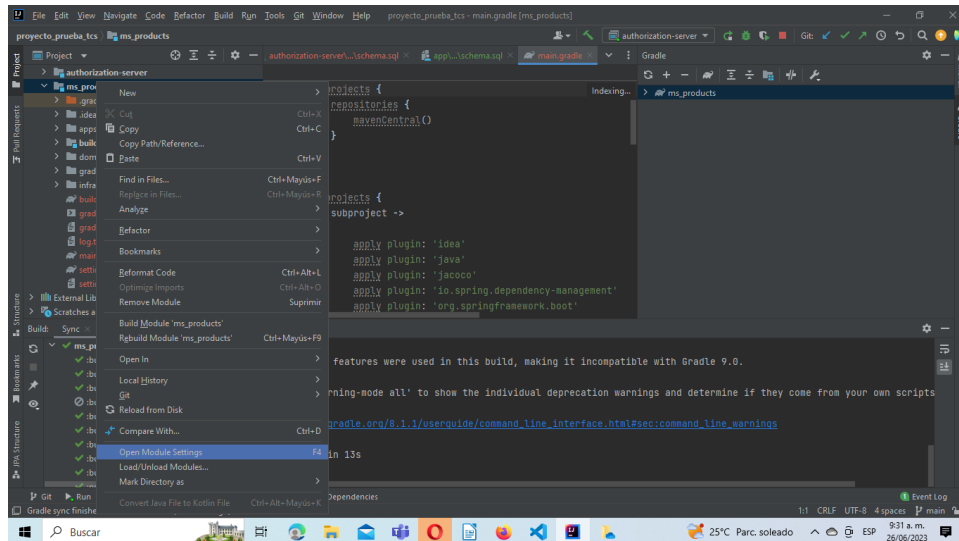
## 10. Generar Module ms-products



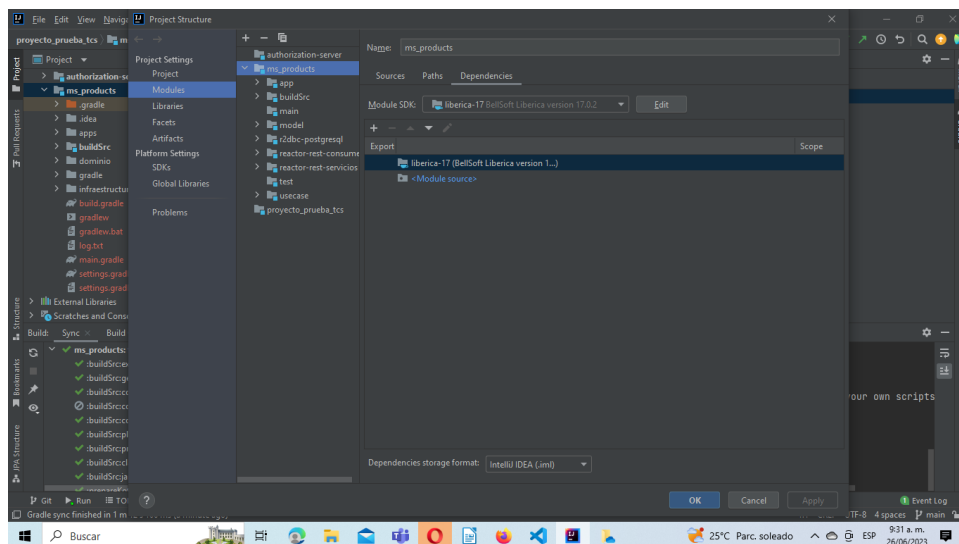
## 11. Generar Link Gradle Project ms-products

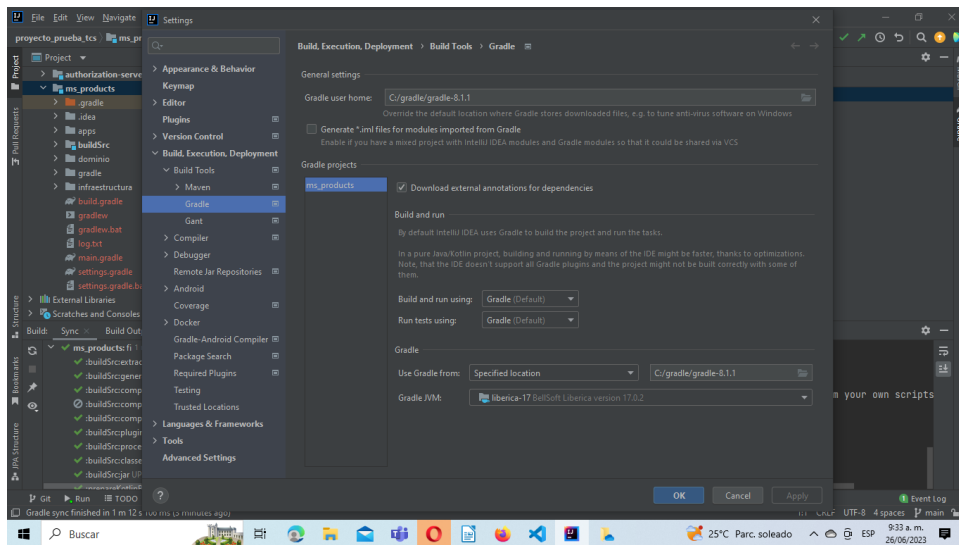


## 12. Open Module Settings



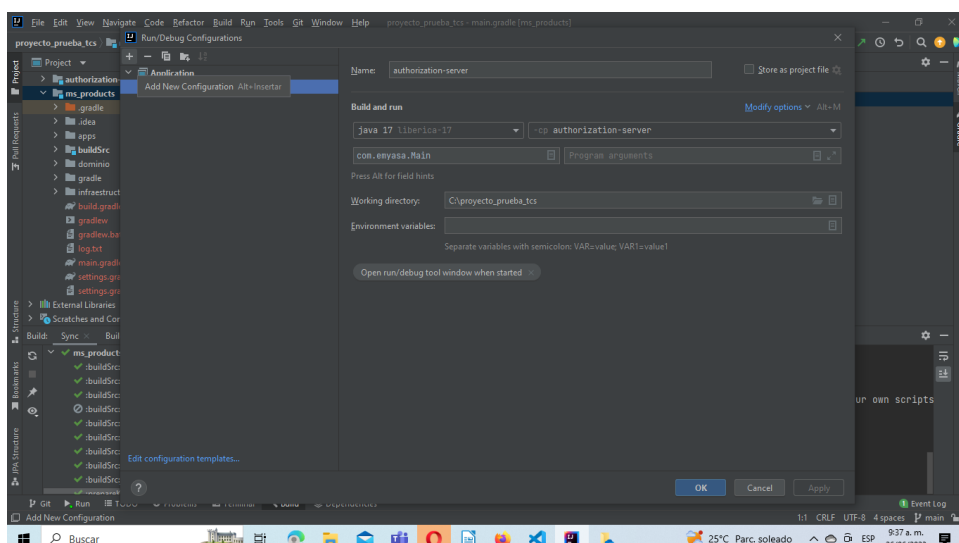
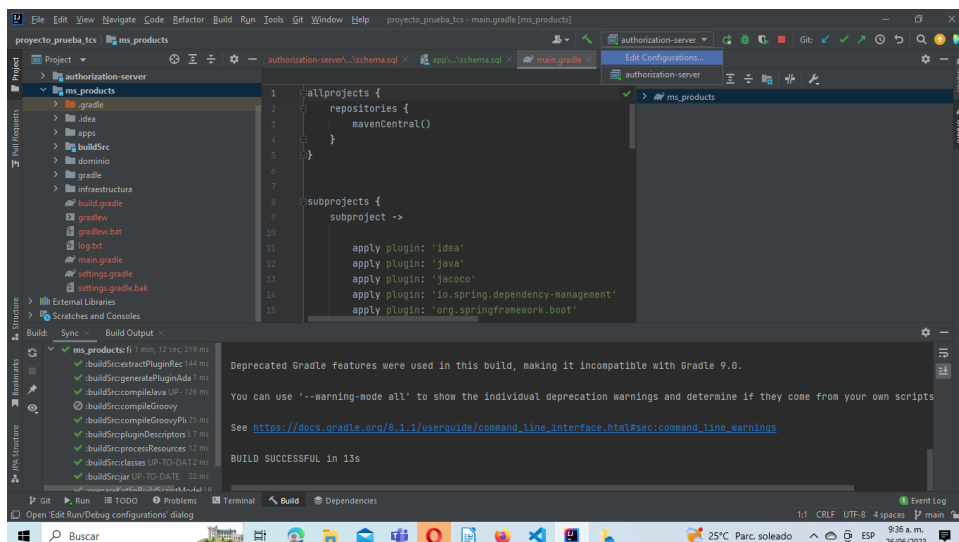
## 13. Gradle Settings



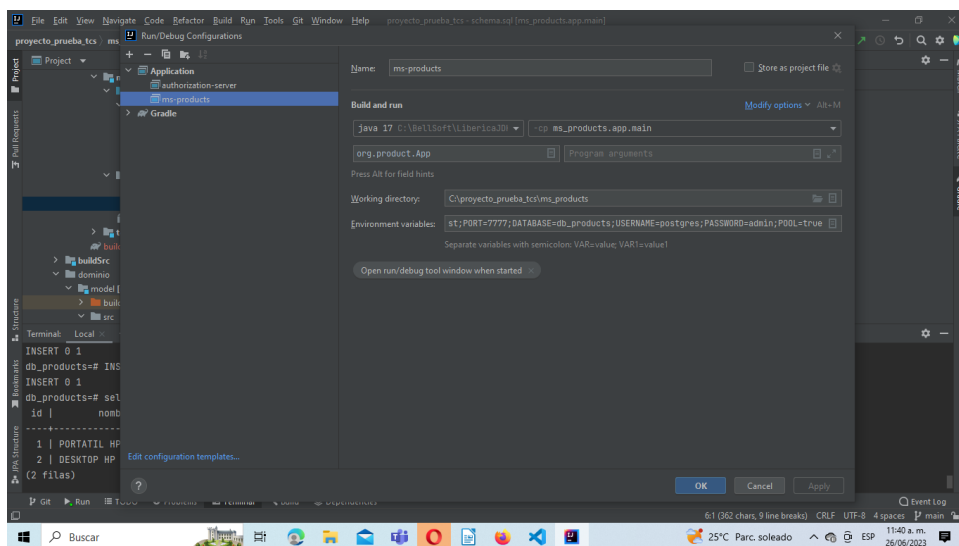
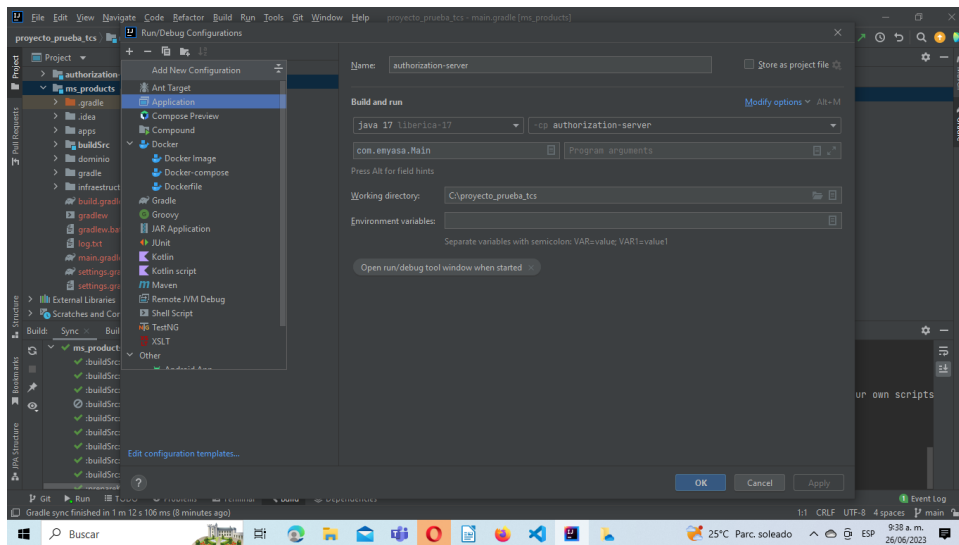


## 14. Edit Configurations

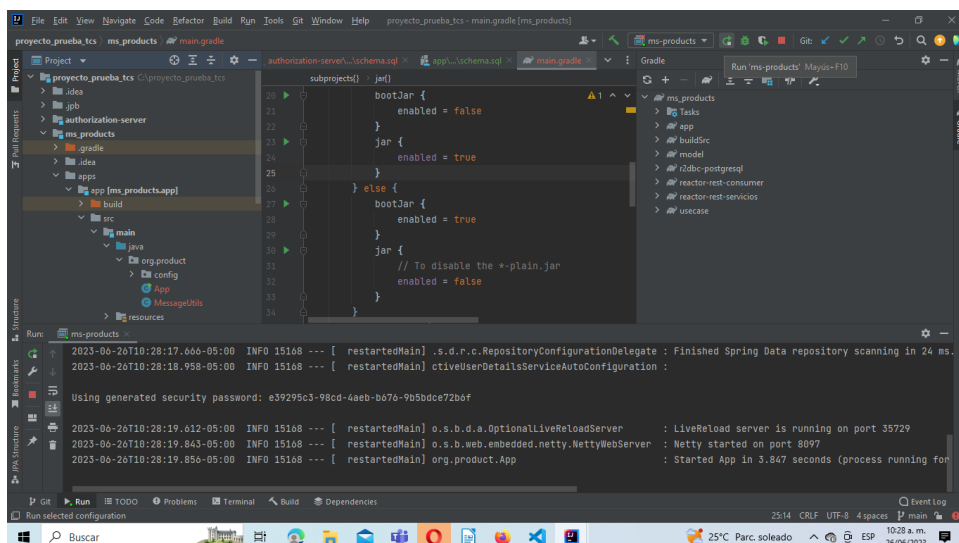
En Environment variables colocar = PORT\_MS=8097;HOST=localhost;PORT=7777;  
DATABASE= db\_products;USERNAME=postgres;PASSWORD=admin;POOL=true



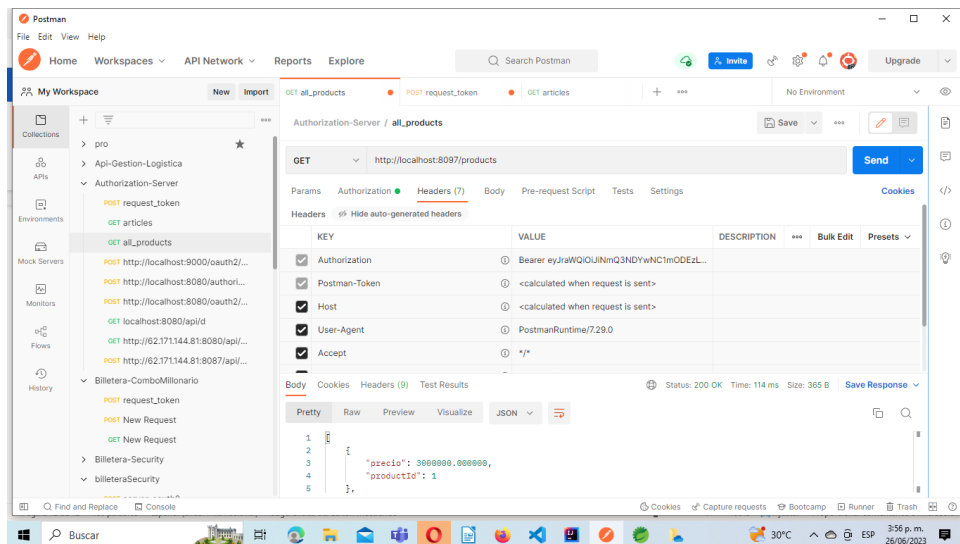
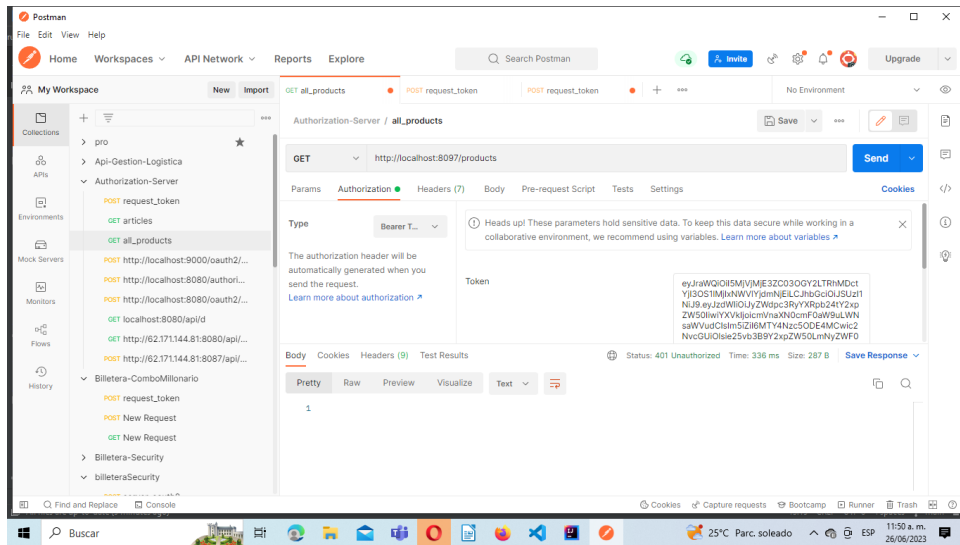




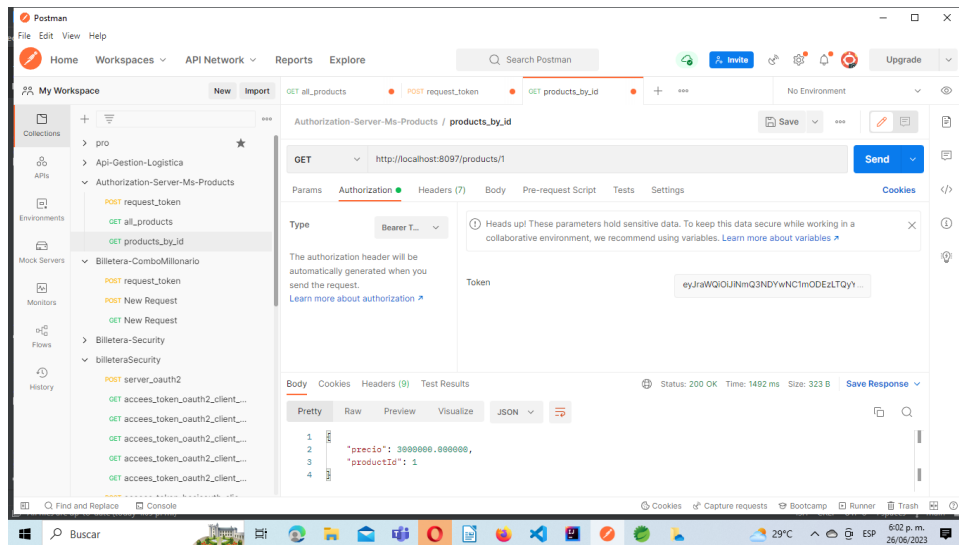
## 15. Run ms-products



### 16. Prueba findAllProducts get /products con postman



### 17. Prueba findProductById get /products/{id} con postman



## 18. Prueba create post /products con postman

Body

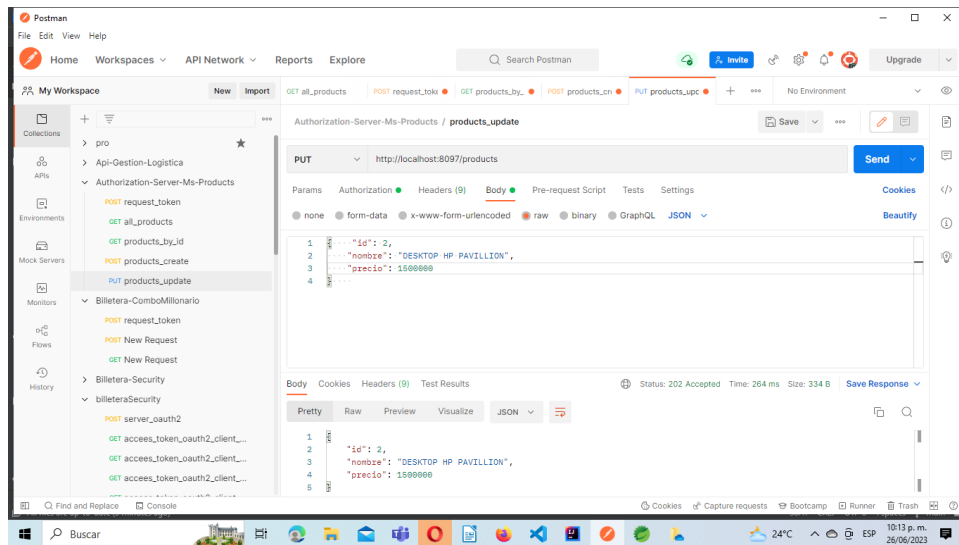
```
{
  "nombre": "Torre Gamer",
  "precio": 3000000
}
```



## 19. Prueba update put /products con postman

Body

```
{ "id": 2,
  "nombre": "DESKTOP HP PAVILLION",
  "precio": 1500000
}
```



## 20. Prueba delete delete /products/{id} con postman

