



Benemérita Universidad Autónoma de Puebla

“Pensar bien, para vivir mejor”

Facultad Ciencias de la Computación

Programación distribuida y aplicada

Desarrollo de agente NTCIP 1205 Mysql

- Profesor: José Martín Orato Ramírez

- Alumnos:

Bruno Moreau García Sánchez

Luis Tomás Morales Vázquez

Antonio Ambrosio Jesús Noé

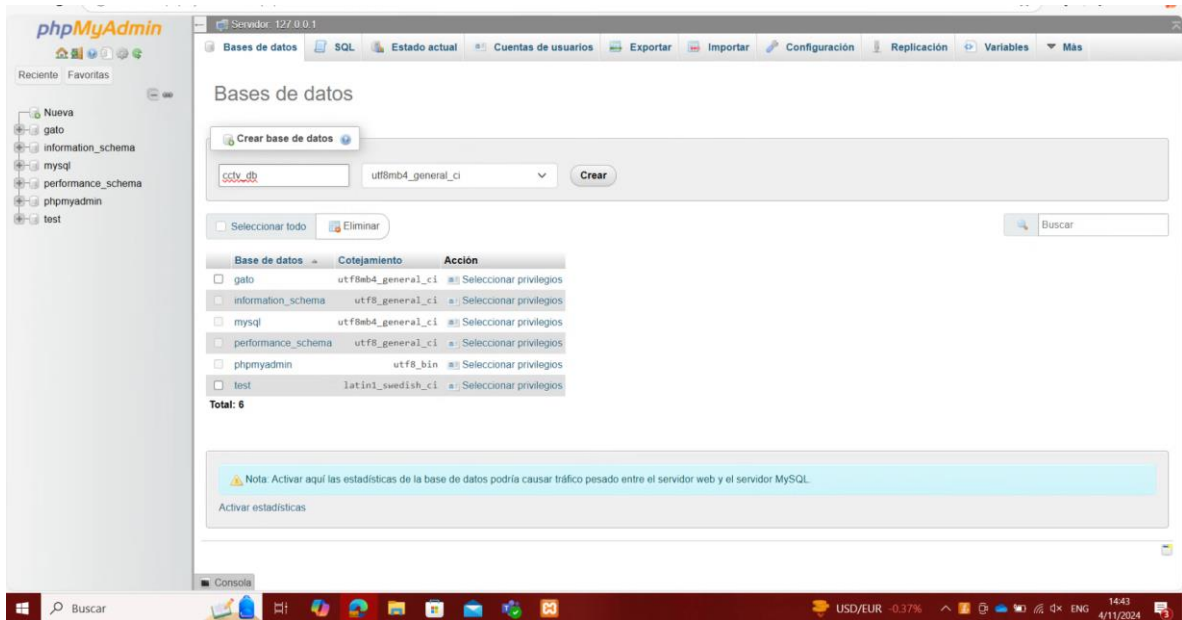
Noviembre, 2024

## **Introducción**

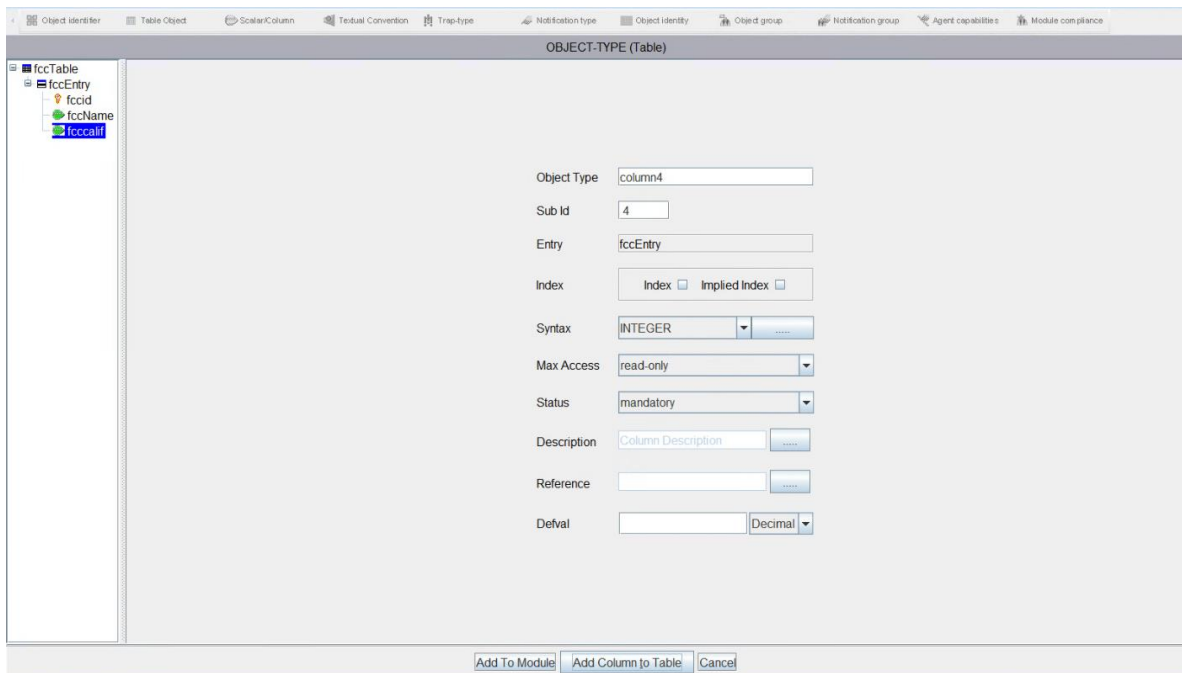
El desarrollo de un agente NTCIP 1205 utilizando una base de datos MySQL es una estrategia orientada a optimizar la gestión de sistemas de control de tráfico, proporcionando una solución adaptable y eficiente. Este proceso implica la creación de la base de datos y tablas, la definición de modos de almacenamiento y la compilación y empaquetado del agente. Además, se realizan pruebas exhaustivas como la ejecución en Eclipse, verificación con snmpwalk, y análisis de registros en herramientas como mTAIL, asegurando una correcta integración y funcionalidad del sistema.

# Desarrollo

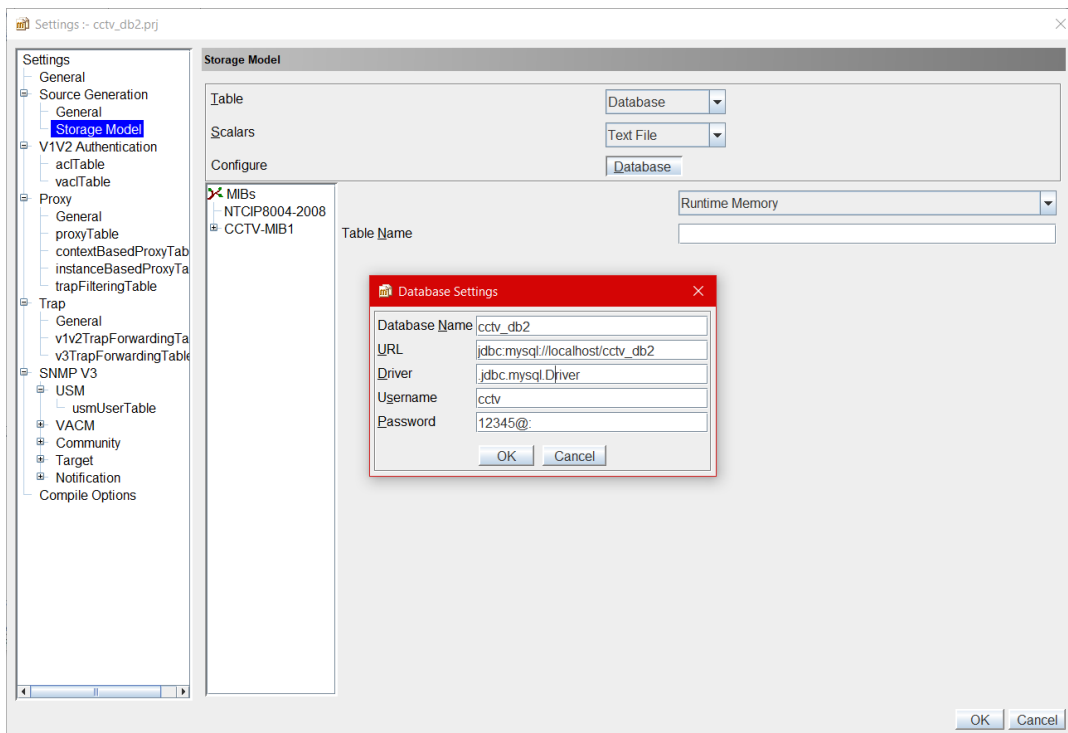
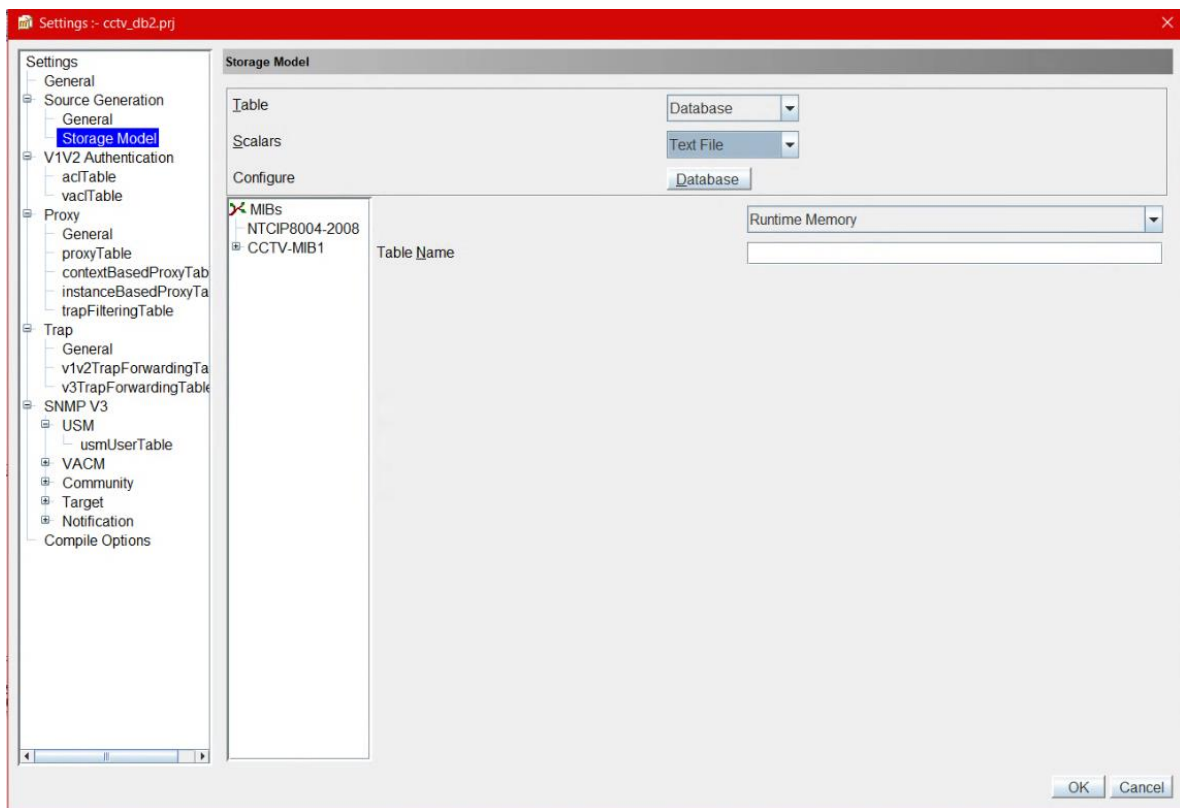
## Creación de la base de datos



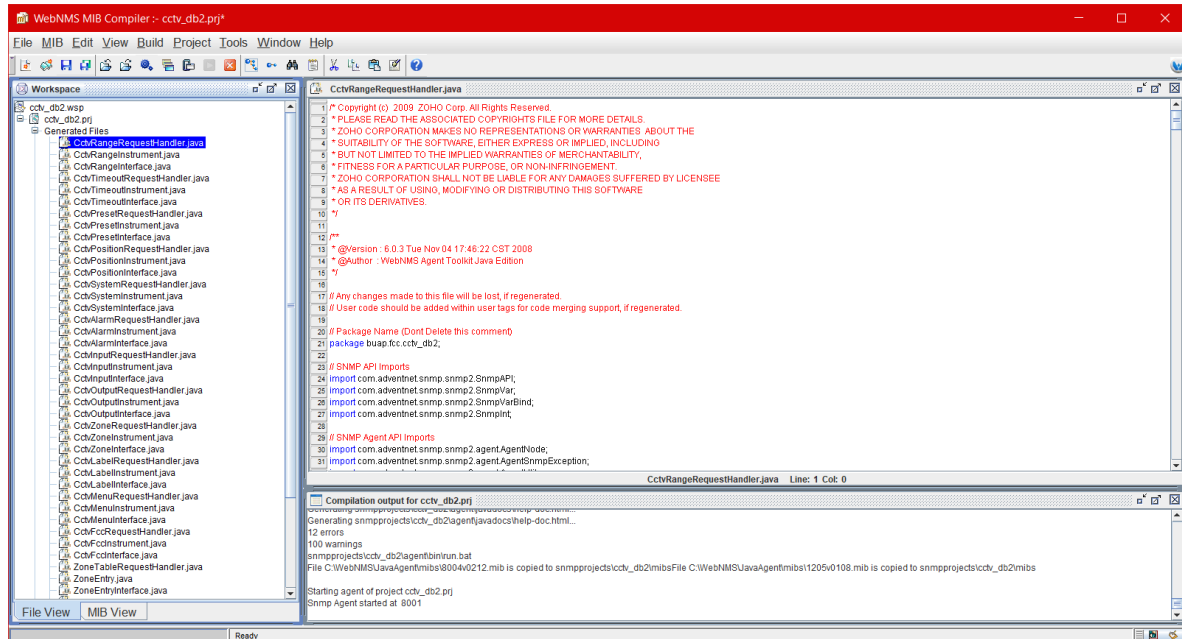
## Creación de la tabla



Definimos el modo de almacenamiento tipo data base y text file



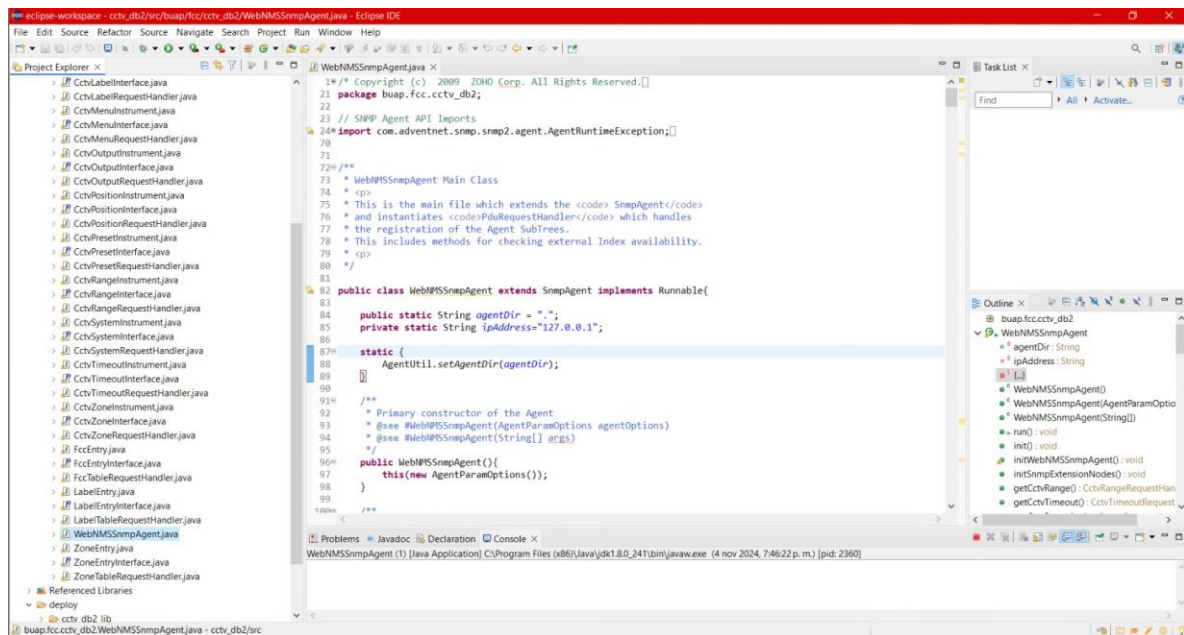
## Compilamos y empaquetamos el agente



## Comprobación snmpwalk

```
PS C:\Users\TommyBaskerville\eclipse-workspace> snmpwalk -c public -v 2c localhost:8001
SNMPv2-MIB::sysDescr.0 = STRING: mySysDescr
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::dod
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysContact.0 = STRING: mySysContact
SNMPv2-MIB::sysName.0 = STRING: mySysName
SNMPv2-MIB::sysLocation.0 = STRING: mySysLocation
SNMPv2-MIB::sysServices.0 = INTEGER: 0
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::snmpInPkts.0 = Counter32: 9
SNMPv2-MIB::snmpOutPkts.0 = Counter32: 9
SNMPv2-MIB::snmpInBadVersions.0 = Counter32: 0
SNMPv2-MIB::snmpInBadCommunityNames.0 = Counter32: 0
SNMPv2-MIB::snmpInBadCommunityUses.0 = Counter32: 0
SNMPv2-MIB::snmpInASNParseErrs.0 = Counter32: 0
SNMPv2-MIB::snmpInTooBigs.0 = Counter32: 0
SNMPv2-MIB::snmpInNoSuchNames.0 = Counter32: 0
SNMPv2-MIB::snmpInBadValues.0 = Counter32: 0
SNMPv2-MIB::snmpInReadOnlys.0 = Counter32: 0
SNMPv2-MIB::snmpInGenErrs.0 = Counter32: 0
SNMPv2-MIB::snmpInTotalReqVars.0 = Counter32: 19
SNMPv2-MIB::snmpInTotalSetVars.0 = Counter32: 0
SNMPv2-MIB::snmpInGetRequests.0 = Counter32: 0
SNMPv2-MIB::snmpInGetNexts.0 = Counter32: 23
SNMPv2-MIB::snmpInSetRequests.0 = Counter32: 0
SNMPv2-MIB::snmpInGetResponses.0 = Counter32: 0
SNMPv2-MIB::snmpInTraps.0 = Counter32: 0
SNMPv2-MIB::snmpOutTooBigs.0 = Counter32: 0
SNMPv2-MIB::snmpOutNoSuchNames.0 = Counter32: 0
SNMPv2-MIB::snmpOutBadValues.0 = Counter32: 0
SNMPv2-MIB::snmpOutGenErrs.0 = Counter32: 0
SNMPv2-MIB::snmpOutGetRequests.0 = Counter32: 0
SNMPv2-MIB::snmpOutGetNexts.0 = Counter32: 0
SNMPv2-MIB::snmpOutSetRequests.0 = Counter32: 0
SNMPv2-MIB::snmpOutGetResponses.0 = Counter32: 33
SNMPv2-MIB::snmpOutTraps.0 = Counter32: 1
SNMPv2-MIB::snmpEnableAuthenTraps.0 = INTEGER: enabled(1)
SNMPv2-MIB::snmpSilentDrops.0 = Counter32: 0
SNMPv2-MIB::snmpProxyDrops.0 = Counter32: 0
PS C:\Users\TommyBaskerville\eclipse-workspace>
```

## La ejecución del programa en eclipse

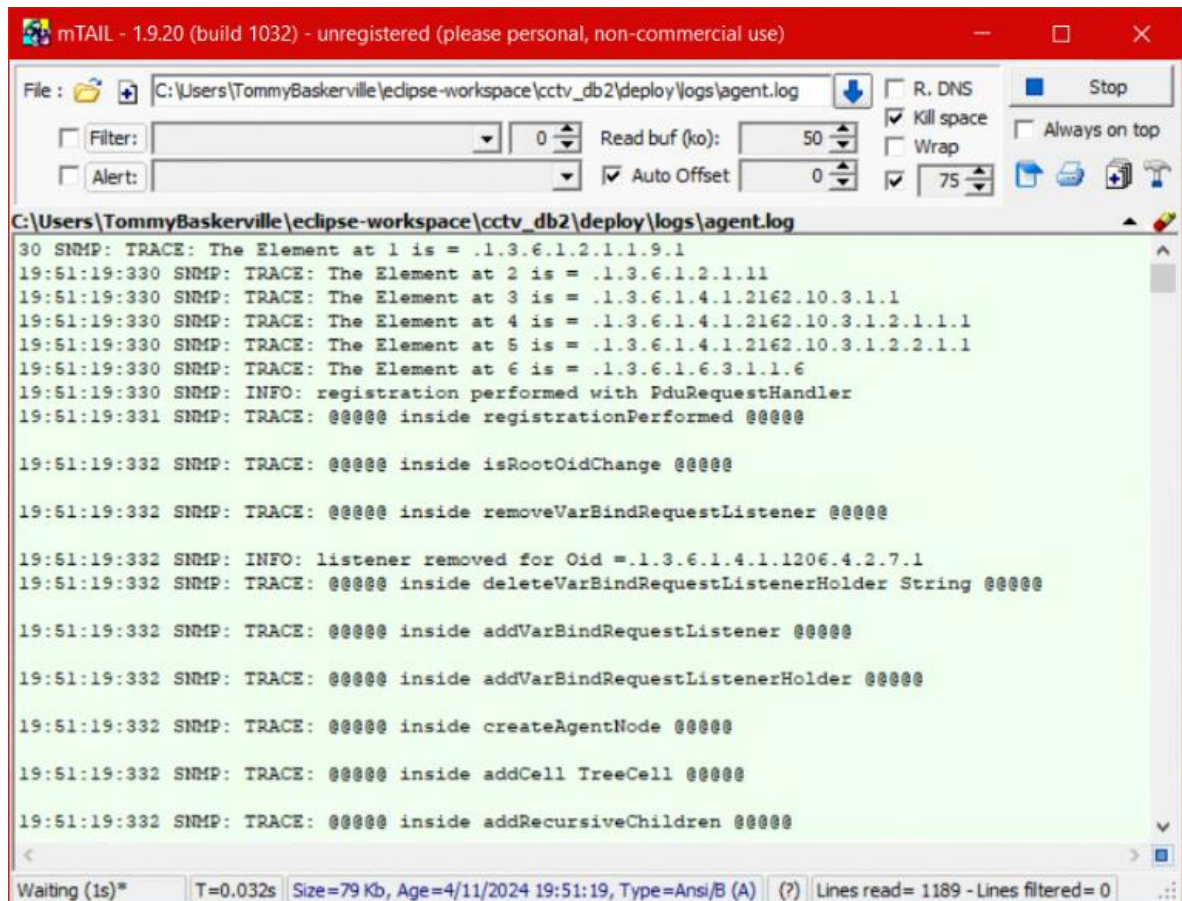




## Ejecución del programa mediante terminal

```
PS C:\Users\TommyBaskerville\eclipse-workspace\cctv_db2\deploy> java -jar cctv_db2.jar
```

## Comprobación del log en mTAIL



The screenshot shows the mTAIL application window, version 1.9.20 (build 1032), which is unregistered for personal use. The window title is "mTAIL - 1.9.20 (build 1032) - unregistered (please personal, non-commercial use)". The file being monitored is "C:\Users\TommyBaskerville\eclipse-workspace\cctv\_db2\deploy\logs\agent.log". The interface includes a "File" menu, a "Filter" dropdown, a "Read buf (ko)" spinner set to 50, and checkboxes for "R. DNS", "Kill space", "Wrap", and "Always on top". The log content is displayed in a text area, showing various SNMP trace and info messages. The status bar at the bottom indicates "Waiting (1s)\*", "T=0.032s", "Size=79 Kb, Age=4/11/2024 19:51:19, Type=Ansi/B (A)", and "Lines read= 1189 - Lines filtered= 0".

```
mTAIL - 1.9.20 (build 1032) - unregistered (please personal, non-commercial use)

File : C:\Users\TommyBaskerville\eclipse-workspace\cctv_db2\deploy\logs\agent.log
Filter: 0 Read buf (ko): 50
Alert: Auto Offset 0 75

C:\Users\TommyBaskerville\eclipse-workspace\cctv_db2\deploy\logs\agent.log

30 SNMP: TRACE: The Element at 1 is = .1.3.6.1.2.1.1.9.1
19:51:19:330 SNMP: TRACE: The Element at 2 is = .1.3.6.1.2.1.11
19:51:19:330 SNMP: TRACE: The Element at 3 is = .1.3.6.1.4.1.2162.10.3.1.1
19:51:19:330 SNMP: TRACE: The Element at 4 is = .1.3.6.1.4.1.2162.10.3.1.2.1.1
19:51:19:330 SNMP: TRACE: The Element at 5 is = .1.3.6.1.4.1.2162.10.3.1.2.2.1.1
19:51:19:330 SNMP: TRACE: The Element at 6 is = .1.3.6.1.6.3.1.1.6
19:51:19:330 SNMP: INFO: registration performed with PduRequestHandler
19:51:19:331 SNMP: TRACE: @@@@ inside registrationPerformed @@@@

19:51:19:332 SNMP: TRACE: @@@@ inside isRootOidChange @@@@

19:51:19:332 SNMP: TRACE: @@@@ inside removeVarBindRequestListener @@@@

19:51:19:332 SNMP: INFO: listener removed for Oid =.1.3.6.1.4.1.1206.4.2.7.1
19:51:19:332 SNMP: TRACE: @@@@ inside deleteVarBindRequestListenerHolder String @@@@

19:51:19:332 SNMP: TRACE: @@@@ inside addVarBindRequestListener @@@@

19:51:19:332 SNMP: TRACE: @@@@ inside addVarBindRequestListenerHolder @@@@

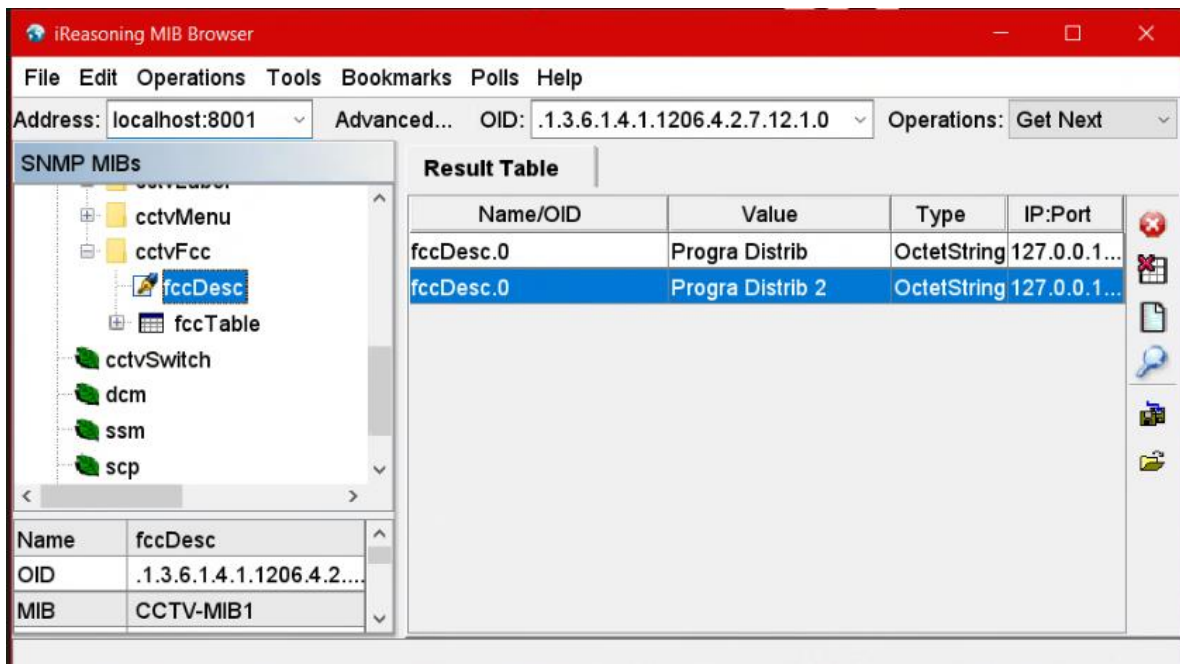
19:51:19:332 SNMP: TRACE: @@@@ inside createAgentNode @@@@

19:51:19:332 SNMP: TRACE: @@@@ inside addCell TreeCell @@@@

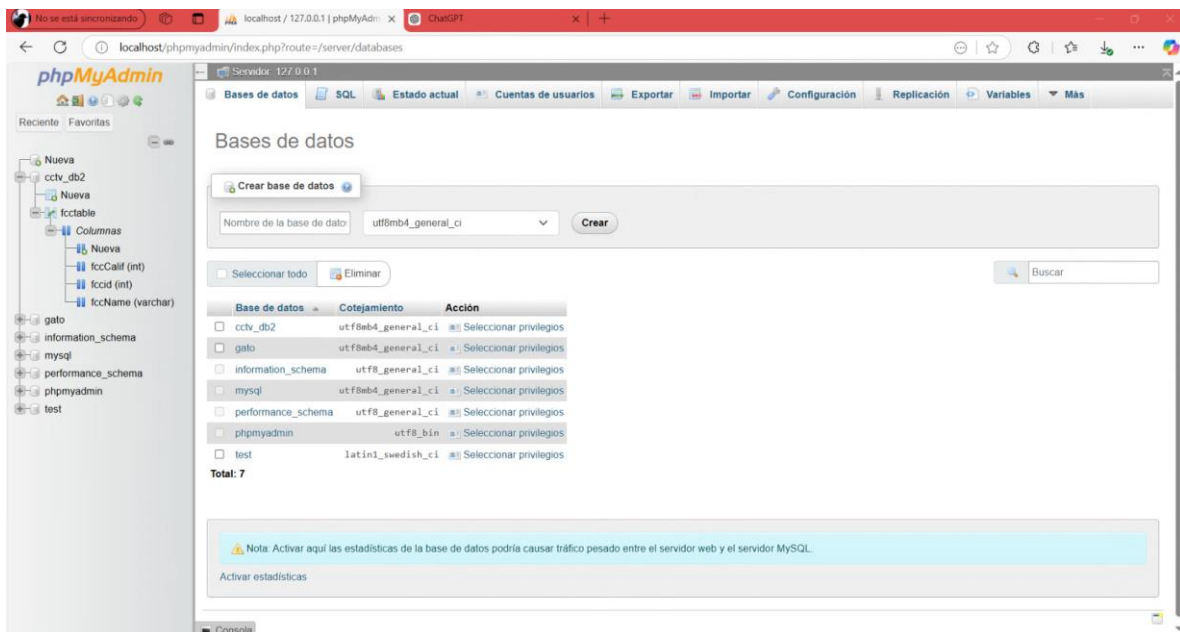
19:51:19:332 SNMP: TRACE: @@@@ inside addRecursiveChildren @@@@

Waiting (1s)* T=0.032s Size=79 Kb, Age=4/11/2024 19:51:19, Type=Ansi/B (A) (?) Lines read= 1189 - Lines filtered= 0
```

## Probando funcionamiento en iReasoning

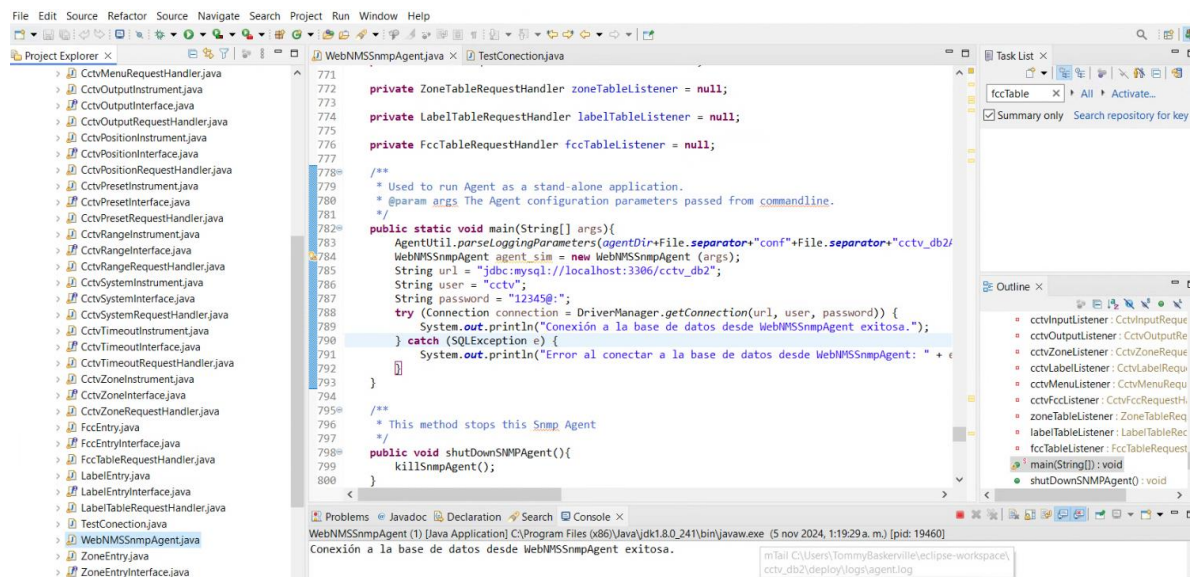
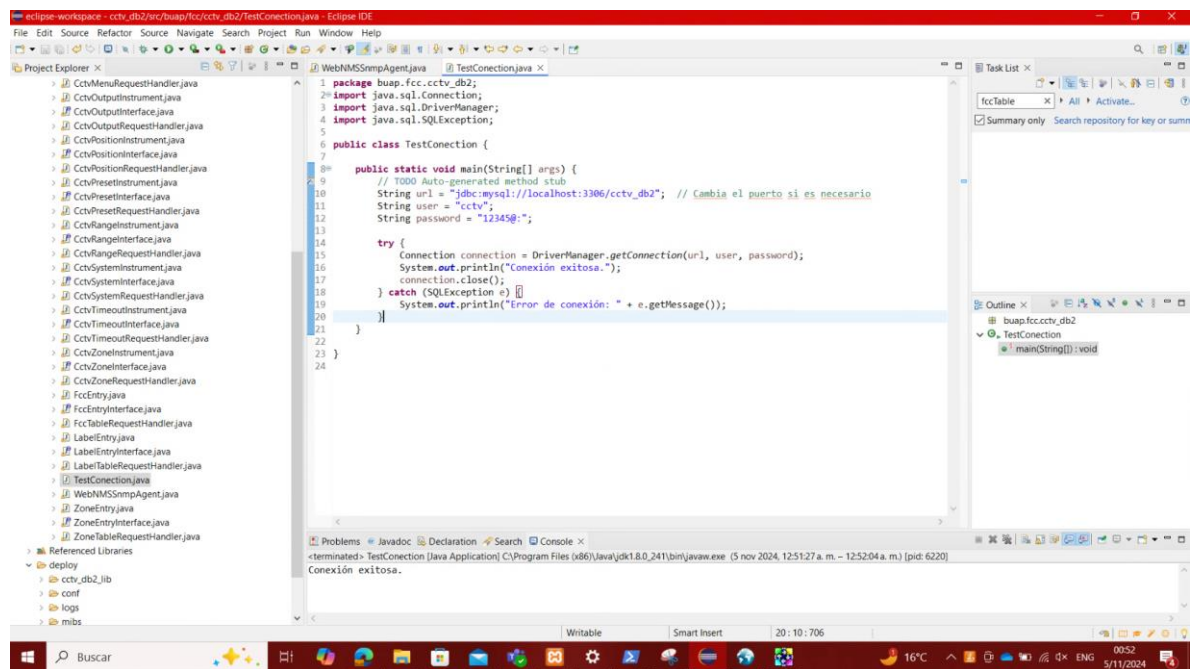


Nos aseguramos de haber creado la tabla y el usuario en la base de datos para tener conexión mediante el código





## Comprobamos la conexión a la base de datos de mysql después de agregar el jar



## **Conclusión**

La implementación de un agente NTCIP 1205 sobre MySQL demuestra ser un enfoque eficaz para la gestión de dispositivos de control de tráfico, proporcionando un manejo de datos fiable y adaptable a las necesidades del entorno. Las verificaciones y pruebas realizadas, como la comprobación de conexión y la ejecución en diversas plataformas, garantizan que la solución esté lista para un despliegue robusto y capaz de integrarse con otros sistemas y herramientas de monitoreo, asegurando así la operatividad y eficiencia del sistema.