



# UNIVERSIDAD TECNOLOGICA DE NEZAHUALCOYOTL

Materia: Administración de Bases de Datos

Práctica: **Seguridad en MongoDB**

Integrantes:

Burgos Curiel Brandon

Efigenio Martinez Diego Axael

Mendoza Barrera Julio Jesus

Ramirez Baena Gerardo Daniel

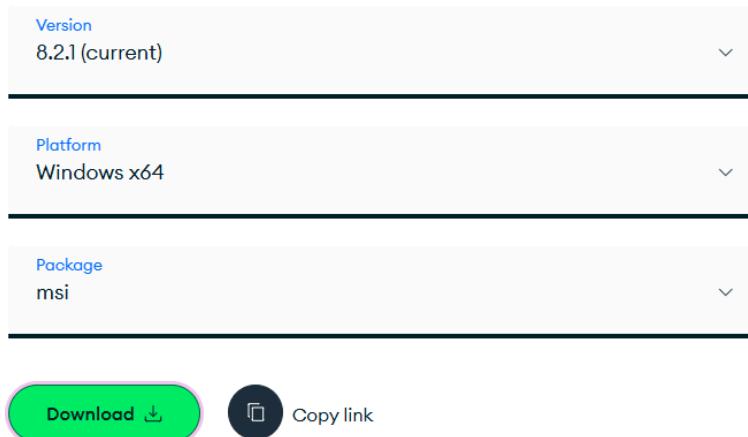
Grupo: ITIC-801M

Ciclo Escolar: Septiembre - Diciembre 2025

# Prerrequisitos (Instalación)

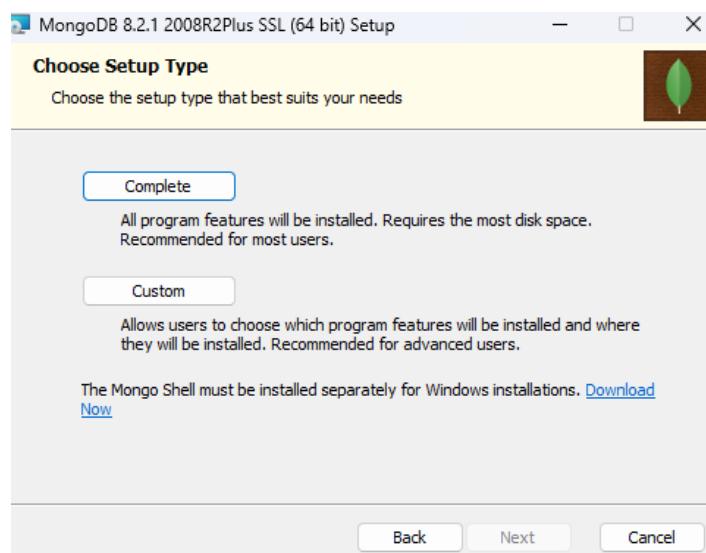
## Descargar MongoDB Community Server:

- Ve a la página oficial de [MongoDB Community Server Download](https://www.mongodb.com/try/download/community).  
(<https://www.mongodb.com/try/download/community>)
- Descarga el instalador `.msi` para Windows.

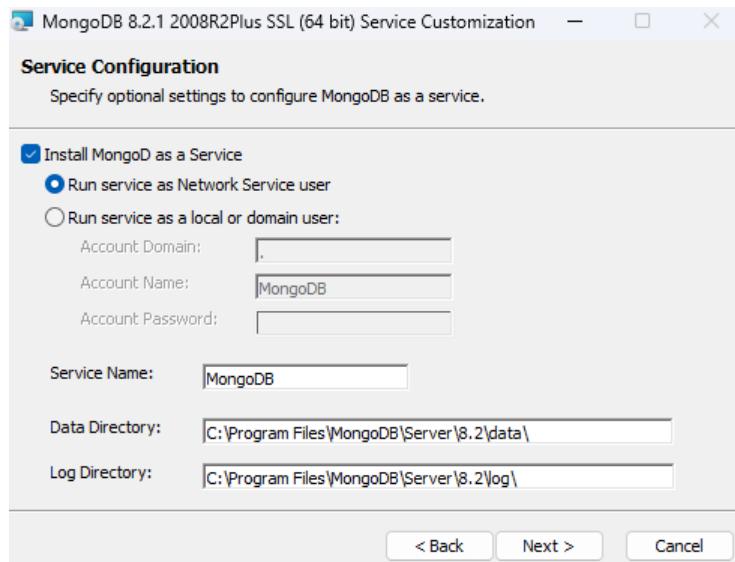


## Instalar MongoDB:

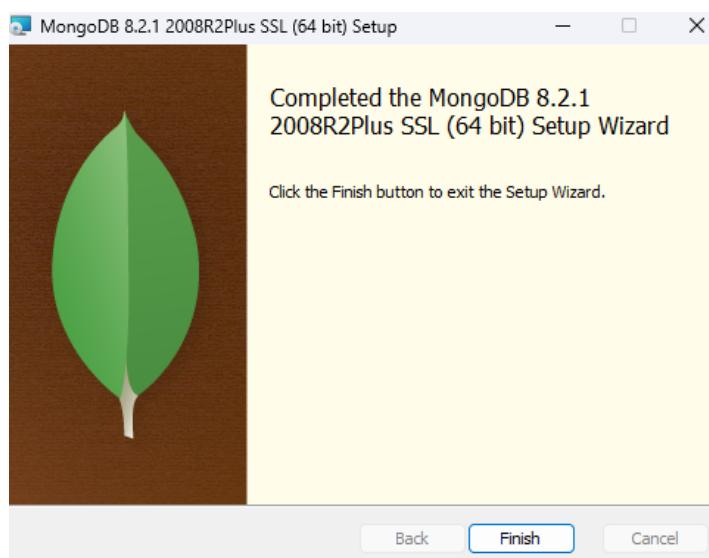
- Ejecuta el instalador.
- Elige la instalación "**Completa**" (**Complete**).



- ¡MUY IMPORTANTE! En la pantalla "Service Configuration", asegúrate de que la opción **"Install MongoDB as a Service"** esté **activada**. Esto es fundamental.
  - Deja el nombre del servicio como "MongoDB".
  - Toma nota del "Data Directory" y "Log Directory".



- En la siguiente pantalla, te ofrecerá instalar **"MongoDB Compass"** (la herramienta gráfica). Recomendamos **instalarlo**. Aunque la práctica es por consola, Compass es excelente para *verificar* visualmente que tus usuarios y datos se crearon correctamente.
- Finaliza la instalación.



En este punto, el **motor** de la base de datos ya está instalado y corriendo como un servicio en tu PC, pero aún no tienes la **consola** para hablar con él.

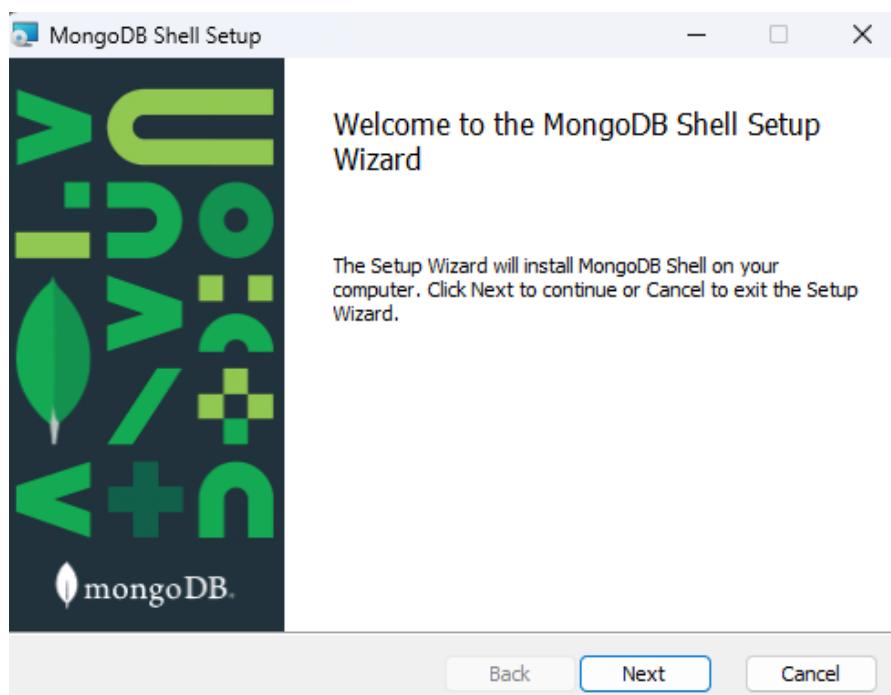
### Instalar el Shell (mongosh)

#### 1. Descargar:

- Ve a [MongoDB Shell Download](#).
- Descarga el instalador **.msi** para Windows (suele ser un archivo **zip** que contiene el **msi**).

#### 2. Instalar:

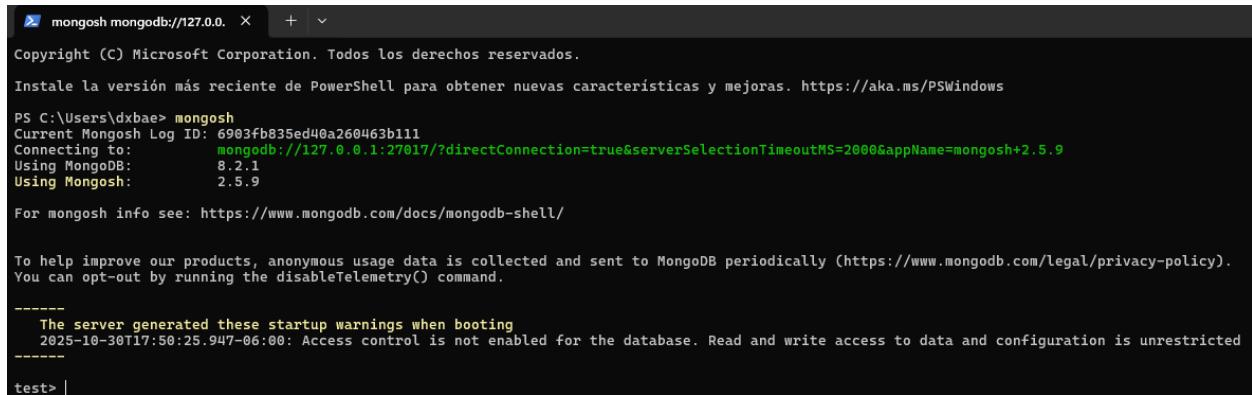
- Ejecuta el instalador **.msi** que descargaste.



- Esta instalación es muy simple. Solo da clic en "Siguiente" a todo. Este instalador se encargará de poner **mongosh.exe** en la ubicación correcta y, lo más importante, de agregarlo al **PATH** de tu sistema

## Verificar la Instalación:

- **Cierra** todas las ventanas de terminal (CMD/PowerShell) que tengas abiertas.
- Abre una **NUEVA** terminal.
- Escribe **mongosh**.
- Si te conectas y ves un cursor **test>** significa que:
  - (A) El Servidor está corriendo.
  - (B) El Shell se instaló correctamente y lo encontró en el PATH.



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\dxbae> mongosh
Current Mongosh Log ID: 6903fb835ed40a260463b111
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
Using MongoDB:      8.2.1
Using Mongosh:      2.5.9

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

-----
The server generated these startup warnings when booting
2025-10-30T17:50:25.947-06:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----
test> |
```

- Escribe **exit** para salir y prepararte para la práctica.

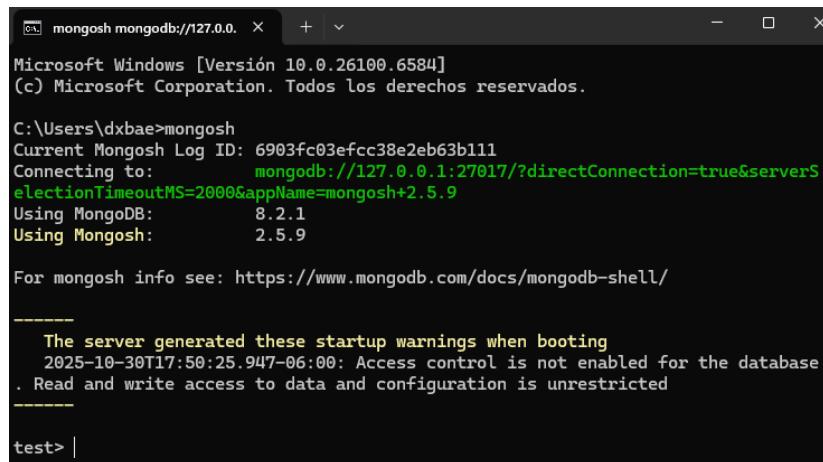
# Administrar Usuarios y Permisos (Autenticación y Autorización)

Por defecto, MongoDB se instala **sin seguridad**. Tu primer paso es crear un "super usuario" antes de "encender" la seguridad.

## Crear el Usuario Administrador (Root)

1. Abre una terminal (CMD) y conéctate a tu base de datos (que aún no tiene seguridad):

Comando: **mongosh**



```
mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
Current Mongosh Log ID: 6903fc03efcc38e2eb63b111
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
Using MongoDB:      8.2.1
Using Mongosh:       2.5.9

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/
-----
The server generated these startup warnings when booting
2025-10-30T17:50:25.947-06:00: Access control is not enabled for the database
. Read and write access to data and configuration is unrestricted
-----
test> |
```

2. Una vez dentro (**test>**), cámbiate a la base de datos **admin**. Esta base es especial y guarda todos los usuarios y roles.

Comando: **use admin**

```
test> use admin
switched to db admin
admin>
```

3. Ahora, crea tu usuario administrador. Este usuario tendrá el rol de `root` (control total). `passwordPrompt()` es una forma segura de escribir tu contraseña.

Comando: `db.createUser({ user: "adminGeneral", pwd: passwordPrompt(), roles: [ { role: "root", db: "admin" } ] })`

4. La consola te pedirá la contraseña. Escríbela (ej. `adminGeneral`) y presiona Enter.
5. Si todo sale bien, verás un mensaje de éxito.

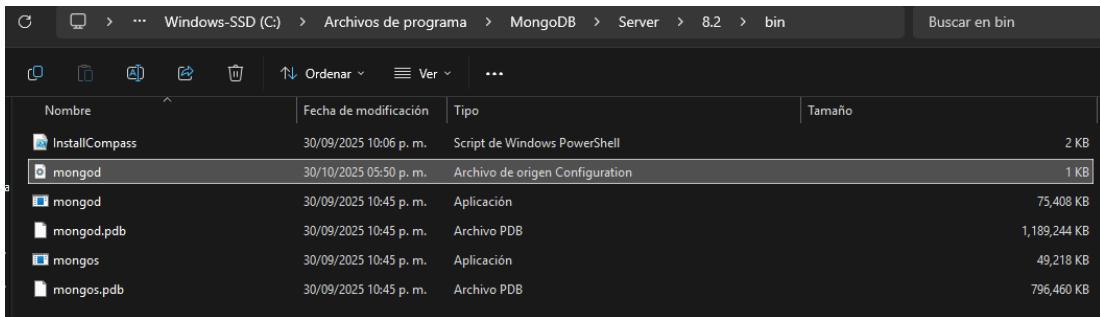
```
admin> db.createUser({ user: "adminGeneral", pwd: passwordPrompt(), roles: [ { r
role: "root", db: "admin" } ] })
Enter password
*****{ ok: 1 }
admin> |
```

6. Escribe `exit` para salir de `mongosh`.

## Activar la Seguridad (Forzar Autenticación)

Ahora le diremos a MongoDB que, de ahora en adelante, *siempre* pida usuario y contraseña.

1. Busca el archivo de configuración de MongoDB. Basado en tu captura, está en:  
`C:\Program Files\MongoDB\Server\8.2\bin\mongod.cfg`



2. Abre este archivo con un editor de texto (como VS Code o Notepad) **en modo Administrador** (clic derecho, "Ejecutar como administrador").

A screenshot of a VS Code editor window. The title bar says 'mongod.cfg C:\Program Files\MongoDB\Server\8.2\bin'. The code in the editor is:

```
1  # mongod.conf
2
3  # for documentation of all options, see:
4  #   http://docs.mongodb.org/manual/reference/configuration-options/
5
6  # Where and how to store data.
7  storage:
8    dbPath: C:\Program Files\MongoDB\Server\8.2\data
9
10 # where to write logging data.
11 systemLog:
12   destination: file
13   logAppend: true
14   path: C:\Program Files\MongoDB\Server\8.2\log\mongod.log
15
16 # network interfaces
17 net:
18   port: 27017
19   bindIp: 127.0.0.1
20 #processManagement:
21
22 #security:
23
24 #operationProfiling:
```

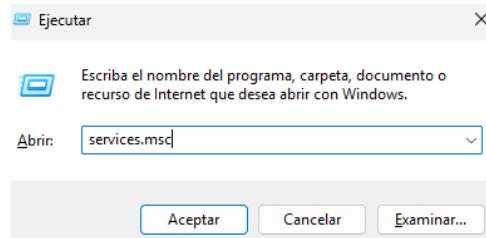
The line 'path: C:\Program Files\MongoDB\Server\8.2\log\mongod.log' is highlighted in blue.

3. Busca la sección `#security`:. Quítale el `#` (descoméntala) y añade la línea `authorization: "enabled"` justo debajo, respetando la sangría (no uses tabulador, hazlo con espacios).

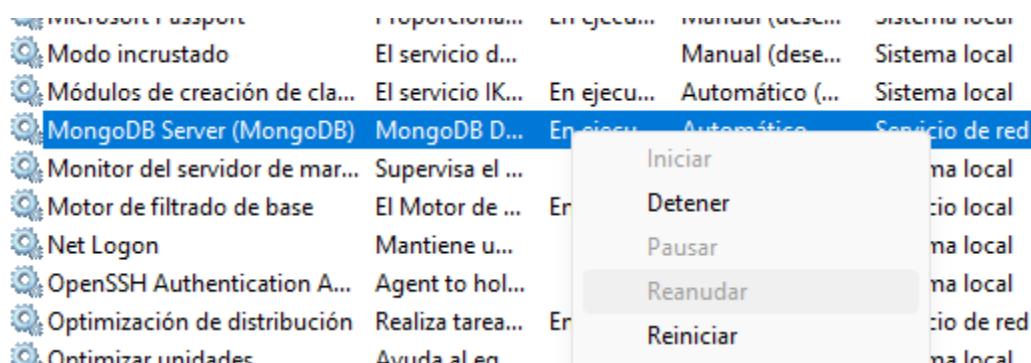
Deberia verse asi:

```
10  # where to write logging data.
11  systemLog:
12    destination: file
13    logAppend: true
14    path: C:\Program Files\MongoDB\Server\8.2\log\mongod.log
15
16  security:
17    authorization: "enabled"
18
19  # network interfaces
20  net:
21    port: 27017
22    bindIp: 127.0.0.1
```

4. Guarda el archivo.
5. **Reinicia el servicio de MongoDB** para que lea la nueva configuración.
  - Presiona **Win + R**, escribe `services.msc` y presiona Enter.



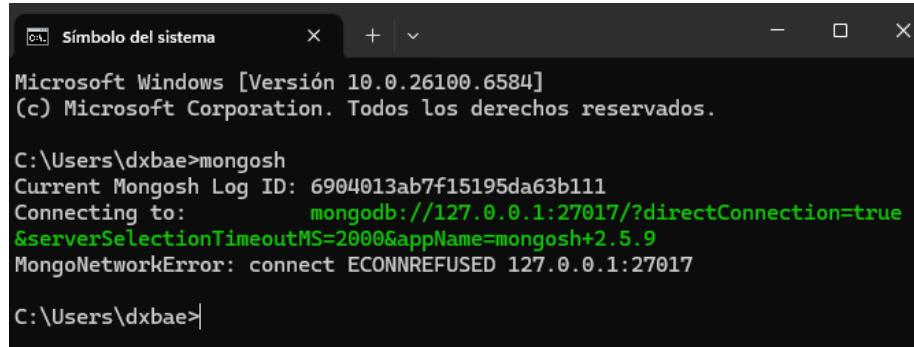
- Busca en la lista el servicio llamado "**MongoDB Server (MongoDB)**".
- Dale clic derecho y selecciona "**Reiniciar**" (**Restart**).



## Verificar que la Seguridad está Activada

1. Abre una nueva terminal. Intenta conectarte como antes:

Comando: **mongosh**



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.26100.6584]
(c) Microsoft Corporation. Todos los derechos reservados.

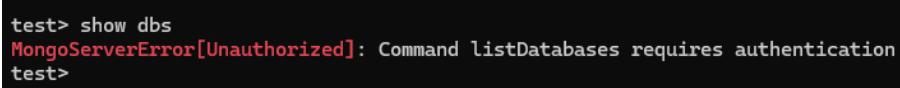
C:\Users\dxbae>mongosh
Current Mongosh Log ID: 6904013ab7f15195da63b111
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true
&serverSelectionTimeoutMS=2000&appName=mongosh+2.5.9
MongoNetworkError: connect ECONNREFUSED 127.0.0.1:27017

C:\Users\dxbae>
```

2. Ahora intenta ver las bases de datos:

Comando: **show dbs**

Recibirás un error de autenticación. ¡Perfecto! El "portero" ya está activo.

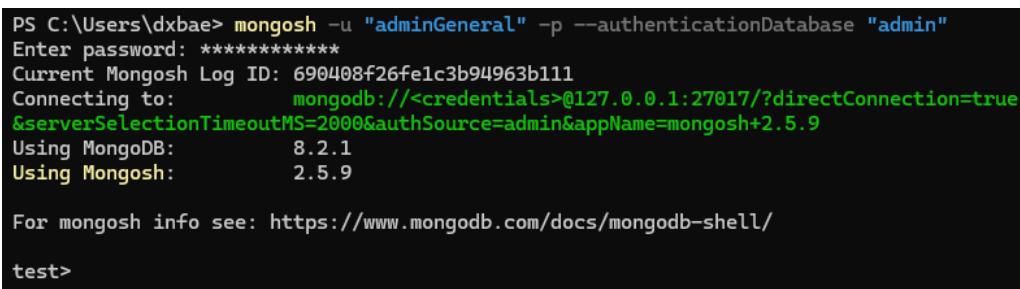


```
test> show dbs
MongoServerError[Unauthorized]: Command listDatabases requires authentication
test>
```

3. Sal (**exit**) y conéctate ahora como el administrador que creaste:

Comando: **mongosh -u "adminGeneral" -p --authenticationDatabase "admin"**

Te pedirá la contraseña (**adminGeneral**). Escríbela.



```
PS C:\Users\dxbae> mongosh -u "adminGeneral" -p --authenticationDatabase "admin"
Enter password: *****
Current Mongosh Log ID: 690408f26fe1c3b94963b111
Connecting to:      mongodb://<credentials>@127.0.0.1:27017/?directConnection=true
&serverSelectionTimeoutMS=2000&authSource=admin&appName=mongosh+2.5.9
Using MongoDB:      8.2.1
Using Mongosh:      2.5.9

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/
test>
```

4. Ahora, si escribes `show dbs`, verás la lista de bases de datos.

```
test> show dbs
admin    100.00 KiB
config   12.00 KiB
local    40.00 KiB
```

## Crear un Usuario Limitado

Ahora, creamos un usuario que *solo* pueda leer y escribir en una base de datos específica (ej. "projetoApp").

1. Aún conectado como `adminGeneral`, crea una base de datos nueva simplemente usándola:

Comando: `use proyectoApp`

```
test> use proyectoApp
switched to db proyectoApp
```

2. Ahora, crea el usuario "limitado". Nota que el rol es `readWrite` y el `db` es `projetoApp`.

Comando: `db.createUser({ user: "devApp", pwd: passwordPrompt(), roles: [ { role: "readWrite", db: "projetoApp" } ] })`

Ponle una contraseña (ej. `claveApp456`).

```
projetoApp> Comando: db.createUser({ user: "devApp", pwd: passwordPrompt(), roles: [
{ role: "readWrite", db: "projetoApp" } ] })
Enter password
*****{ ok: 1 }
projetoApp>
```

3. Sal de la consola (`exit`).

## Probar el Usuario Limitado

1. Vamos a conectarnos como el nuevo usuario "limitado":

Comando: **mongosh -u "devApp" -p --authenticationDatabase "projetoApp"**

- Usa la contraseña **claveApp456**
- Posterior a eso usa el comando: **use projetoApp**

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. https://aka.ms/PSWindows

PS C:\Users\dxbae> mongosh -u "devApp" -p --authenticationDatabase "projetoApp"
Enter password: *****
Current Mongosh Log ID: 69040dfe6a506ef99a63b111
Connecting to:      mongodb://<credentials>@127.0.0.1:27017/?directConnection=true
$serverSelectionTimeoutMS=2000&authSource=projetoApp&appName=mongosh+2.5.9
Using MongoDB:     8.2.1
Using Mongosh:     2.5.9

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

test> use projetoApp
switched to db projetoApp
```

2. Intenta escribir un dato (esto **debería funcionar**):

Comando: **db.clientes.insertOne({ nombre: "Juan", edad: 30 })**

```
projetoApp> db.clientes.insertOne({ nombre: "Juan", edad: 30 })
{
  acknowledged: true,
  insertedId: ObjectId('69040dd196b829f1d363b113')
}
projetoApp>
```

3. Ahora, intenta hacer algo de administrador, como ver los usuarios (esto **debería fallar**):

Comando 1: **use admin**

Comando 2: **show users**

```
projetoApp> use admin
switched to db admin
admin> show users
MongoServerError[Unauthorized]: not authorized on admin to execute command { userInfo: 1, lsid: { id: UUID("c18532ad-e421-42e9-9fb9-27d03702061c") }, $db: "admin" }
```

# Configuración de Acceso Remoto

Por defecto, MongoDB solo acepta conexiones de la **misma máquina** (`127.0.0.1`). Vamos a cambiar esto para que acepte conexiones desde otras máquinas en la red.

## Editar el Archivo de Configuración

1. Vuelve a abrir tu archivo `mongod.cfg` como **Administrador**: `C:\Program Files\MongoDB\Server\8.2\bin\mongod.cfg`
2. Busca la sección `net`:. Por defecto, se ve así:

```
19  # network interfaces
20  net:
21    port: 27017
22    bindIp: 127.0.0.1
23    #processManagement:r
```

3. Cambia `127.0.0.1` (solo local) por `0.0.0.0` (todas las interfaces de red).

```
19  # network interfaces
20  net:
21    port: 27017
22    bindIp: 0.0.0.0
23    #processManagement:r
```

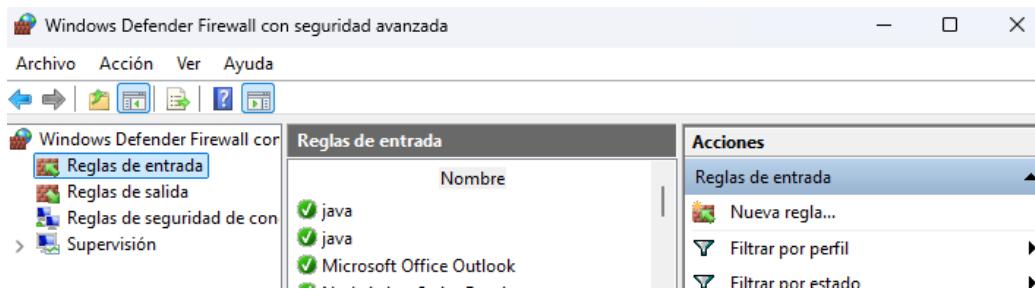
**Advertencia:** En producción, NUNCA usarías `0.0.0.0`. Pondrías la IP específica del servidor de tu aplicación. Esto es solo para la práctica.

4. Guarda el archivo y **Reinicia el servicio** de MongoDB (`services.msc` -> "MongoDB Server" -> "Reiniciar").

# Abrir el Puerto en el Firewall de Windows

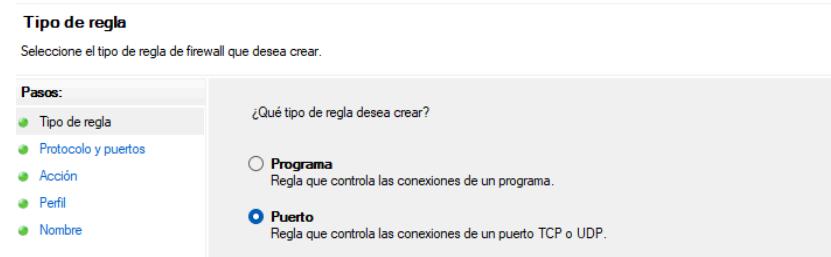
El servicio ya acepta conexiones remotas, pero el **Firewall de Windows** las bloqueará.

1. Abre "Firewall de Windows Defender con seguridad avanzada" (búscalos en el menú Inicio).
2. Clic en "**Reglas de entrada**" (izquierda).
3. Clic en "**Nueva regla...**" (derecha).

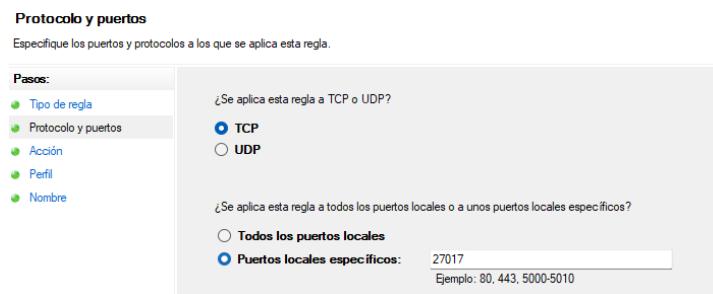


4. Se abre un asistente:

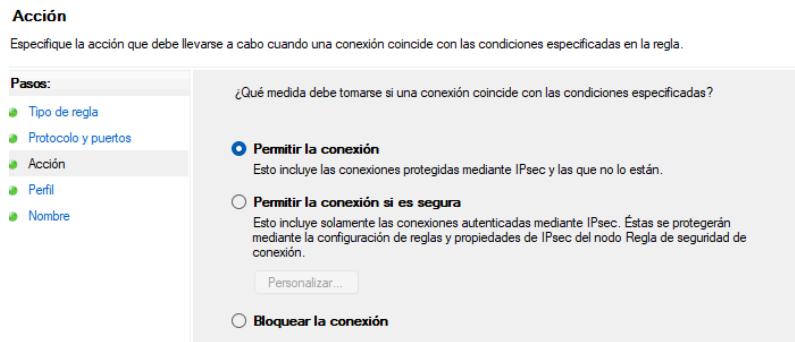
- **Tipo de regla:** Elige "**Puerto**". Siguiente.



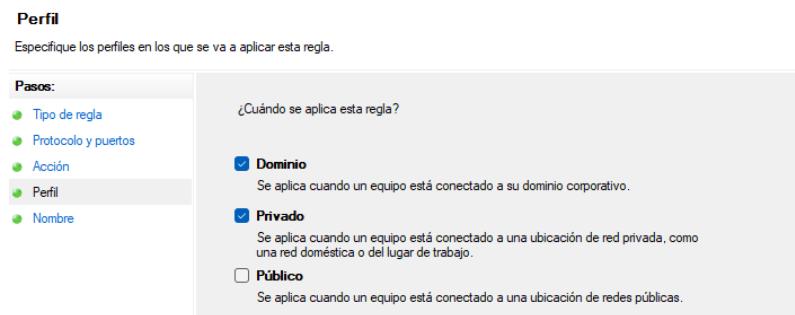
- **Protocolo y puertos:** Elige "**TCP**" y en "Puertos locales específicos", escribe **27017**. Siguiente.



- **Acción:** Elige "Permitir la conexión". Siguiente.

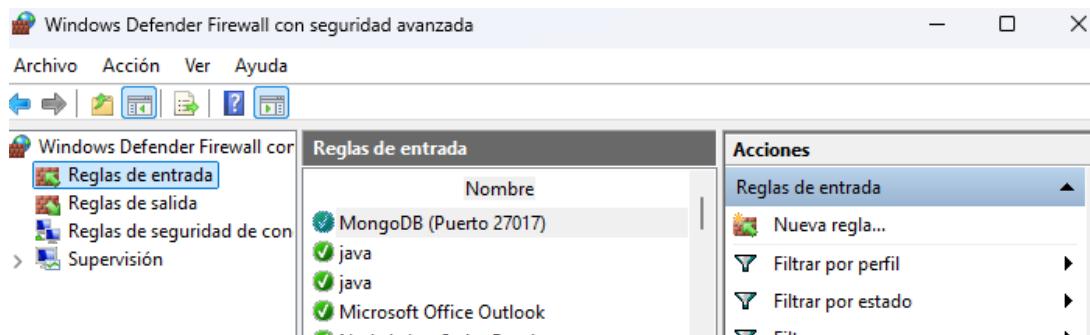


- **Perfil:** Deja marcadas "Dominio" y "Privado". (Desmarca "Público" por seguridad). Siguiente.



- **Nombre:** Ponle un nombre descriptivo, como MongoDB (Puerto 27017).

5. Haz clic en "Finalizar".



# Prueba de Conexión Remota

En tu máquina (donde está el servidor), abre **cmd** y escribe **ipconfig**. Busca tu dirección "IPv4". Será algo como **192.168.56.1**.

(Idealmente) Pídele a un compañero en la **misma red** que abra su **mongosh** y se conecte a tu IP usando la cadena de conexión completa:

```
mongosh "mongodb://devApp:claveApp456@[TU_IP_AQUI]:27017/proyectoApp"
```

(Reemplaza [TU\_IP\_AQUI] con tu dirección IPv4, ej. 192.168.56.1).

Si tu compañero logra conectarse e insertar un dato, ¡la práctica ha sido un éxito!