

FASE 1 — Instalación del Servidor MongoDB

Instalar MongoDB en el servidor Linux de la USS Horizon.

Procedimiento

1. Accede al servidor (terminal local o SSH).

2. Actualiza los paquetes del sistema:

```
sudo apt update && sudo apt upgrade -y
```

```
ruill@rui-server:~$ sudo apt update && sudo apt upgrade -y  
[sudo] password for ruill:
```

3. Añade la llave GPG del “repositorio de la Federación”:

```
curl -fsSL https://pgp.mongodb.com/server-7.0.asc | \  
sudo gpg -o /usr/share/keyrings/mongodb-server-7.0.gpg --  
dearmor
```

```
ruill@rui-server:~$ curl -fsSL https://pgp.mongodb.com/server-7.0.asc | \  
sudo gpg -o /usr/share/keyrings/mongodb-server-7.0.gpg --dearmor
```

4. Activa el repositorio estable:

```
echo "deb [arch=amd64,arm64 signed-by=/usr/share/keyrings/mongodb-server-7.0.gpg] \  
https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/7.0  
multiverse" | \  
sudo tee /etc/apt/sources.list.d/mongodb-org-7.0.list
```

```
ruill@rui-server:~$ echo "deb [arch=amd64,arm64 signed-by=/usr/share/keyrings/mongodb-server-7.0.gpg] \  
https://repo.mongodb.org/apt/ubuntu jammy/mongodb-org/7.0 multiverse" | \  
sudo tee /etc/apt/sources.list.d/mongodb-org-7.0.list  
deb [arch=amd64,arm64 signed-by=/usr/share/keyrings/mongodb-server-7.0.gpg] https://repo.mongodb.org/apt/  
ubuntu jammy/mongodb-org/7.0 multiverse
```

5. Instala MongoDB:

```
sudo apt update  
sudo apt install -y mongodb-org
```

```
ruill@rui-server:~$ sudo apt update  
sudo apt install -y mongodb-org
```

Validación

Si no ha habido errores, MongoDB está instalado en el servidor auxiliar del núcleo de datos.

FASE 2 — Arranque del Núcleo de Datos MongoDB

Objetivo

Levantar el servicio mongod y verificar su estado.

Procedimiento

1. Inicia el servicio:

```
sudo systemctl start mongod
```

2. Comprueba su estado operativo:

```
sudo systemctl status mongod
```

3. Actívalo para que arranque con el sistema de la nave:

```
sudo systemctl enable mongod
```

Validación

Si el servicio está activo, el núcleo de datos está disponible.

```
Docs: https://docs.mongodb.org/manual
ruill@rui-server:~$ sudo systemctl start mongod
ruill@rui-server:~$ sudo systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; disabled; preset: enabled)
   Active: active (running) since Thu 2026-01-22 08:28:55 UTC; 13s ago
     Docs: https://docs.mongodb.org/manual
    Main PID: 21761 (mongod)
      Memory: 72.8M (peak: 73.1M)
         CPU: 534ms
    CGroup: /system.slice/mongod.service
            └─21761 /usr/bin/mongod --config /etc/mongod.conf
```

```
ruill@rui-server:~$ sudo systemctl enable mongod
Created symlink /etc/systemd/system/multi-user.target.wants/mongod.service → /usr/lib/systemd/system/mongod.service.
```

FASE 3 — Acceso al Observador de Datos (mongosh)

Objetivo

Acceder a la consola interactiva de MongoDB.

Procedimiento

Entrada al núcleo:

```
mongosh
```

Resultado esperado:

```
test>
```

Ese prompt indica que estás dentro del sistema base.

```

ruil1@rui-server:~$ mongosh
Current Mongosh Log ID: 6971e364f657845cc98ce5af
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeout
MS=2000&appName=mongosh+2.6.0
Using MongoDB:      7.0.28
Using Mongosh:      2.6.0

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

-----
  The server generated these startup warnings when booting
  2026-01-22T08:28:55.482+00:00: Using the XFS filesystem is strongly recommended with the Wir
edTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
  2026-01-22T08:28:55.897+00:00: Access control is not enabled for the database. Read and writ
e access to data and configuration is unrestricted
-----
test> |

```

FASE 4 — Organización del Archivo Estelar

En la Flota Estelar, los registros se organizan por sectores. En MongoDB será así:

Servidor → Bases de datos → Colecciones → Documentos → Campos

FASE 5 — Creación de Base de Datos y Colecciones

Objetivo

Crear una base de datos para almacenar datos del Sector Andoria.

Procedimiento

1. Cambia a la base de datos andoria_sector:
use andoria_sector

```
test> use andoria_sector
switched to db andoria_sector
andoria_sector> show dbs
```

Ahora crearás una colección llamada planetas.

2. Inserta un documento básico:

```
db.planetas.insertOne({
  nombre: "Andoria",
  especie_principal: "Andoriano",
  clima: "Frío",
  alineacion: "Federación",
  tecnologia: "Warp"
})
```

```
andoria_sector> db.planetas.insertOne({
...   nombre: "Andoria",
...   especie_principal: "Andoriano",
...   clima: "Frío",
...   alineacion: "Federación",
...   tecnologia: "Warp"
... })
...
{
  acknowledged: true,
  insertedId: ObjectId('6971e3f0f657845cc98ce5b0')
}
andoria_sector> |
```

3. Comprueba el contenido:

```
db.planetas.find()
```

```
andoria_sector> db.planetas.find()
[
  {
    _id: ObjectId('6971e3f0f657845cc98ce5b0'),
    nombre: 'Andoria',
    especie_principal: 'Andoriano',
    clima: 'Frío',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  }
]
andoria_sector> |
```

FASE 6 — Manejo de Registros Científicos

Objetivo

Insertar, buscar, actualizar y eliminar registros.

A) Inserciones múltiples

Añade varios planetas explorados:

```
db.planetas.insertMany([
  { nombre: "Vulcano", especie_principal: "Vulcano",
    alineacion: "Federación", tecnologia: "Warp" },
  { nombre: "Qo'noS", especie_principal: "Klingon",
    alineacion: "Imperio Klingon", tecnologia: "Warp" },
  { nombre: "Romulus", especie_principal: "Romulano",
    alineacion: "Imperio Estelar", tecnologia: "Warp" }
])
```

```

andoria_sector> db.planetas.insertMany([
... { nombre: "Vulcano", especie_principal: "Vulcano", alineacion: "Federación", tecnologia: "Warp" },
... { nombre: "Qo'noS", especie_principal: "Klingon", alineacion: "Imperio Klingon", tecnologia: "Warp" },
... { nombre: "Romulus", especie_principal: "Romulano", alineacion: "Imperio Estelar", tecnologia: "Warp" }
... ])
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6971e46ff657845cc98ce5b1'),
    '1': ObjectId('6971e46ff657845cc98ce5b2'),
    '2': ObjectId('6971e46ff657845cc98ce5b3')
  }
}

```

B) Consultas

Lista los planetas:

```
db.planetas.find()
```

```

andoria_sector> db.planetas.find()
[
  {
    _id: ObjectId('6971e3f0f657845cc98ce5b0'),
    nombre: 'Andoria',
    especie_principal: 'Andoriano',
    clima: 'Frío',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b1'),
    nombre: 'Vulcano',
    especie_principal: 'Vulcano',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b2'),
    nombre: "Qo'noS",
    especie_principal: 'Klingon',
    alineacion: 'Imperio Klingon',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b3'),
    nombre: 'Romulus',
    especie_principal: 'Romulano',
    alineacion: 'Imperio Estelar',
    tecnologia: 'Warp'
  }
]

```

Filtra por alineación:

```
db.planetas.find({ alineacion: "Federación" })
```



```

andoria_sector> db.planetas.find({ alineacion: "Federación" })
[
  {
    _id: ObjectId('6971e3f0f657845cc98ce5b0'),
    nombre: 'Andoria',
    especie_principal: 'Andoriano',
    clima: 'Frío',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b1'),
    nombre: 'Vulcano',
    especie_principal: 'Vulcano',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  }
]
andoria_sector> |

```

C) Actualizaciones

El Consejo Andoriano ha cambiado la clasificación de Andoria:

```

db.planetas.updateOne(
  { nombre: "Andoria" },
  { $set: { clima: "Ártico" } }
)

```

```

andoria_sector> db.planetas.updateOne(
...   { nombre: "Andoria" },
...   { $set: { clima: "Ártico" } }
... )
...
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}

```

D) Eliminaciones

Tras un incidente diplomático, se decide borrar temporalmente un registro:

```
db.planetas.deleteOne({ nombre: "Romulus" })
```

```
andoria_sector> db.planetas.deleteOne({ nombre: "Romulus" })  
{ acknowledged: true, deletedCount: 1 }  
andoria_sector> |
```

Comprueba que desapareció:

```
db.planetas.find()
```

```

andoria_sector> db.planetas.find()
[
  {
    _id: ObjectId('6971e3f0f657845cc98ce5b0'),
    nombre: 'Andoria',
    especie_principal: 'Andoriano',
    clima: 'Ártico',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b1'),
    nombre: 'Vulcano',
    especie_principal: 'Vulcano',
    alineacion: 'Federación',
    tecnologia: 'Warp'
  },
  {
    _id: ObjectId('6971e46ff657845cc98ce5b2'),
    nombre: "Qo'noS",
    especie_principal: 'Klingon',
    alineacion: 'Imperio Klingon',
    tecnologia: 'Warp'
  }
]
andoria_sector> |

```

FASE 7 — Gestión de Colecciones y Base de Datos

Ver colecciones existentes

show collections

```

andoria_sector> show collections
planetas

```

Eliminar una colección entera

Ejemplo:

```
db.planetas.drop()
```

```
andoria_sector> db.planetas.drop()
true
andoria_sector> show collections
```

Eliminar una base de datos

Primero asegúrate de estar en ella:

```
use andoria_sector
```

Luego:

```
db.dropDatabase()
```

```
andoria_sector> db.dropDatabase()
{ ok: 1, dropped: 'andoria_sector' }
```

FASE 8 — Exportación e Importación (Bitácora Estelar)

Para generar bitácoras externas:

Exportar colección:

```
mongoexport --db andoria_sector --collection planetas --out
planetas.json
```

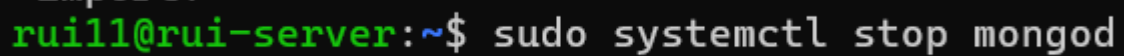
Importar colección:

```
mongoimport --db andoria_sector --collection planetas --file planetas.json
```

FASE 9 — Cierre del Núcleo

Puedes detener el servicio si la nave entra en modo de sigilo:

```
sudo systemctl stop mongod
```

A terminal window screenshot with a black background. The prompt is 'rui11@rui-server:~\$' in green. The command 'sudo systemctl stop mongod' is entered in white text.

```
rui11@rui-server:~$ sudo systemctl stop mongod
```