



**Universidad Tecnológica de Salamanca**  
Organismo Público Descentralizado del Gobierno del Estado de Guanajuato

## **PLAN DE ADMINISTRACIÓN DE BASE DE DATOS**

**ANA Y MIA**

**PARA REVISIÓN DEL PROYECTO INTEGRADOR EN LA ASIGNATURA  
“ADMINISTRACIÓN DE BASE DE DATOS”, DE LA CARRERA DE INGENIERÍA  
EN GESTIÓN Y DESARROLLO DE SOFTWARE**

**PRESENTAN:**

**SAMARA NEFTALI RAMÍREZ GARCÍA**

**DIANA LIZBETH ROBLES GARCÍA**

**FRANCISCO JAVIER HERNANDEZ ARREDONDO**

**SALAMANCA, GTO., 8 DE FEBRERO DE 2024**



**Universidad Tecnológica de Salamanca**  
Organismo Público Descentralizado del Gobierno del Estado de Guanajuato

## **PLAN DE ADMINISTRACIÓN DE BASE DE DATOS**

**ANA Y MIA**

**PARA REVISIÓN DEL PROYECTO INTEGRADOR EN LA ASIGNATURA  
“ADMINISTRACIÓN DE BASE DE DATOS”, DE LA CARRERA DE INGENIERÍA  
EN GESTIÓN Y DESARROLLO DE SOFTWARE**

**ASIGNATURA:**

**ADMINISTRACIÓN DE BASE DE DATOS**

**PROFESOR:**

**JOSÉ GUSTAVO ALMANZA VÁZQUEZ.**



## Contenido

<b>1. INTRODUCCIÓN .....</b>	<b>1</b>
<b>1.1. Definición del problema .....</b>	<b>1</b>
<b>1.2. Justificación.....</b>	<b>3</b>
<b>2. OBJETIVOS.....</b>	<b>3</b>
<b>2.1. General.....</b>	<b>3</b>
<b>2.2. Específicos.....</b>	<b>4</b>
<b>3. MARCO TEÓRICO .....</b>	<b>4</b>
<b>3.1 ChatBot. ....</b>	<b>4</b>
<b>3.2 . Visual Studio Code.....</b>	<b>5</b>
<b>3.3 . Sistema Gestor de Base de Datos. ....</b>	<b>7</b>
<b>3.4 . Mongo DB .....</b>	<b>7</b>
<b>3.5 JavaScript .....</b>	<b>8</b>
<b>4. DESARROLLO.....</b>	<b>8</b>
<b>4.1 Conexión de la base de datos con el ChatBot Ana y Mia. ....</b>	<b>9</b>
<b>4.2 Procedimiento de copias de seguridad.....</b>	<b>14</b>
<b>4.3 Protocolo de restauración.....</b>	<b>16</b>
<b>4.4 Descripción y propósito de la automatización de tareas. ....</b>	<b>17</b>
<b>4.5 Métodos de importación y exportación de datos. ....</b>	<b>19</b>
<b>4.6 Lineamientos de seguridad en la base de datos.....</b>	<b>20</b>
<b>4.7 Reporte de rendimiento de una base de datos con los datos que arroja la herramienta de monitoreo. ....</b>	<b>23</b>
<b>5. CONCLUSIÓN.....</b>	<b>28</b>
<b>6. Bibliografía.....</b>	<b>29</b>

## Tabla de ilustraciones

Ilustración 1. ChatBot. Fuente: Google. ....	5
Ilustración 2 Visual Studio Code. Fuente: Google .....	6
Ilustración 3. Sistema Gestor de Base de Datos. Fuente: Google. ....	7
Ilustración 4. Mongo DB. Fuente: Google. ....	8
Ilustración 5. JavaScript. Fuente: Google.....	8
Ilustración 6. Creación de nuevo ChatBot. Fuente: Personal. ....	9
Ilustración 7. Creación de cuenta en Mongo DB. Fuente: Personal. ....	10
Ilustración 8. Creación de Base de Datos. Fuente: Personal. ....	10
Ilustración 9. Acceso de red. Fuente: Personal. ....	11
Ilustración 10. Código de Conexión entre Mongo DB con VSC. Fuente: Personal. .....	12
Ilustración 11. Código de conexión con Mongo DB. Fuente: Personal.....	12
Ilustración 12. Creación de "addKeyword" anteriormente intents. Fuente: Personal. .....	12
Ilustración 13. Inicialización del Servicio de NodeJS. Fuente: Personal. ....	13
Ilustración 14. Conexión con Base de Datos. Fuente: Personal.....	13
Ilustración 15. Conexión con WhatsApp. Fuente: Personal.....	14
Ilustración 16. Copia de seguridad en Mongo DB. Fuente: Google.....	15
Ilustración 17. Restauración desde la copia de seguridad. Fuente: Google.....	16
Ilustración 18. Restauración de Base de datos por el comando "mongodump". Fuente: Google.....	17
Ilustración 19. Restauración usando replicas en Mongo DB. Fuente: Google. ....	17
Ilustración 20. Importación de datos "mongoimport". Fuente: Google.....	19
Ilustración 21. Importación de datos "Mongo Shell". Fuente. Google.....	19
Ilustración 22. Exportación de Datos "mongoexport". Fuente: Google.....	20
Ilustración 23. Exportación de Datos "Mongo Shell". Fuente: Google.....	20
Ilustración 24. Rendimiento Mongo DB (Conexiones). Fuente: Mongo DB. ....	23
Ilustración 25.Rendimiento Mongo DB (Tamaño lógico). Fuente: Mongo DB. ....	24
Ilustración 26.Rendimiento Mongo DB(Red). Fuente: Mongo DB. ....	24
Ilustración 27.Rendimiento Mongo DB (Dominio). Fuente: Mongo DB.....	24

# 1. INTRODUCCIÓN

En el presente documento se presenta la integración de una base de datos para el proyecto integrador “Ana y Mia”. Esto con la finalidad de comenzar a almacenar algunos datos del ChatBot.

Además, se incluye el plan de administración de base de datos. Es importante contar con un plan de administración de base de datos ya que de esta manera podemos tener un mejor control, seguridad, accesibilidad y escalabilidad de la base de datos. El plan de administración en este caso cuenta con los siguientes puntos:

- Conexión de la base de datos con el ChatBot Ana y Mia.
- Procedimiento de copias de seguridad
- Protocolo de restauración
- Descripción y propósito de la automatización de tareas
- Métodos de importación y exportación de datos
- Lineamientos de seguridad en la base de datos
- Reporte de rendimiento de una base de datos con los datos que arroja la herramienta de monitoreo.

## 1.1. Definición del problema

Los trastornos alimenticios son enfermedades médicas graves que afectan tanto la salud física como la mental. Estas afecciones incluyen problemas en la forma de pensar sobre la comida, la alimentación, el peso y la figura, así como en los comportamientos alimentarios.

Es bien sabido que uno de los mayores afanes de los niños, según van acercándose a la juventud, es el de ser aceptados por sus iguales, tener éxito y conseguir un estatus que les permita tenerse por personas socialmente valiosas. Cada grupo o

cultura crea por ello un ideal al que dirigirse, cualificado de forma que sus valores consoliden y refuercen los valores del grupo. Un ideal que se configura estética y moralmente, que evoluciona paralelamente a la evolución de los poderes fácticos y que ejerce su influencia a partir de dos elementos: El deseo de éxito y la no cuestionabilidad. Ambos derivan del hecho de que sean justamente dichos poderes los que los sostengan.

Actualmente, sobre todo en las sociedades desarrolladas tecnológicamente, es el mercado quien marca esos estándares. Como en épocas anteriores, el objetivo es la influencia, la homogeneización y el control social, pero a diferencia de ellas, el impacto no radica en el ajuste del modelo propuesto sino en el alcance de los canales que usa para transmitirlo. Los denominados Medios de Comunicación de Masas, son actualmente quienes dan forma y contenido a la vida política, económica y social; crean opinión pública y son simultáneamente reflejo y modelo para quienes vivimos en ellas.

Los trastornos de alimentación se han subido a la red con un gran despliegue de páginas web, blog, foros y testimonios personales. Desde sus inicios, además de páginas informativas, y ofertas de tratamientos ha ido apareciendo una corriente en apoyo de dichos desórdenes que han hecho de esta enfermedad una “forma de vida” y que comparten con cualquier usuario lo que ellos mismos denominan su filosofía defendiendo una “mal entendida libertad de expresión”.

En los últimos años, los pacientes que acuden a consulta son cada vez más jóvenes. Entre otros motivos, esto se explica por los estándares de la belleza y de estética perfecta que rigen en la sociedad actual. Y en este punto adquieren un especial protagonismo las redes sociales. “Todos los chicos tienen acceso a su influencer, a la guapa, a la dieta milagro, a las fotos de cuerpos maravillosos... Y esa información fake está en las redes a la orden del día.

Dos de los trastornos más comunes son la anorexia nerviosa (comúnmente conocida como anorexia) y bulimia nerviosa (comúnmente conocida como bulimia). Algunas personas muestran signos y síntomas de ambos trastornos alimentarios. Según un informe de la Agencia de Calidad de Internet, existen casi cuatro millones

de publicaciones en la Red con las etiquetas #ana (anorexia) y #mia (bulimia). Muchas víctimas de ambas enfermedades utilizan estos hashtags para compartir trucos, experiencias o remedios para adelgazar y hacer más accesible el contenido.

## **1.2. Justificación**

Presentar propuestas modernas, adaptadas a las tecnologías actuales hace llamar más la atención no solo de las personas jóvenes, sino también de las personas mayores. Brindar una opción de ayuda, asegurando que es un espacio seguro y privado donde se puedan desenvolver y decir todo lo que les incomoda. El hecho de ser una inteligencia artificial también llama la atención e incentiva a utilizarla; por la curiosidad de algo nuevo.

El ChatBot de “Ana y Mia”, necesita una base de datos con la finalidad de conservar y almacenar ciertos datos de los usuarios para llevar el control de cada usuario. De este modo es muy importante contar con un plan de administración de base de datos para tener un mejor manejo de esta.

## **2. OBJETIVOS**

### **2.1. General.**

Lograr contar con un plan de administración de base de datos que nos ayude con el control, seguridad y mejor gestión de esta.



## **2.2. Específicos.**

- Realizar la base de datos para el proyecto de Ana y Mia.
- Investigar acerca de la administración de base de datos para generar uno propio.

## **3. MARCO TEÓRICO**

### **3.1 ChatBot.**

Podemos definir un ChatBot como un asistente que se comunica con los usuarios a través de mensajes de texto. En muchas otras ocasiones, toma forma convirtiéndose en un compañero virtual que se integra en sitios web, aplicaciones... conversando y ayudando a los usuarios.

Se trata de una tecnología que permite al usuario mantener una conversación a través de un software que se integra en un determinado sistema de mensajería, como, por ejemplo: Facebook, Twitter, Telegram, WhatsApp, etc.

El sistema está programado para que interactúe con el cliente y le resuelva dudas, pero sin que haya una persona física contestando. Tienen la ventaja de que están disponibles siempre para resolver las dudas de los usuarios que quieran contactar contigo a cualquier hora del día.

Por lo tanto, estamos hablando de una herramienta que interactúa de manera automática con usuarios y potenciales clientes con el fin de guiarles hacia la acción deseada (conversión).

### **Antecedentes del ChatBot.**

El primer ChatBot de la historia fue ELIZA, un programa desarrollado en el MIT por Joseph Weizenbaum entre 1964 y 1966, capaz de procesar lenguaje natural. Su funcionamiento era aparentemente simple: identificaba palabras clave e interpelaba

al usuario sobre ellas. Lo más curioso de este primer bot conversacional es algo que no suele comentarse: fue creado para criticar y ridiculizar las técnicas psicoterapéuticas de Carl Rogers y mostrar la inutilidad y superficialidad de las conversaciones banales (un experimento crítico que, por cierto, fue reinterpretado en 2020 por Steve Worswick en una excéntrica cita romántica entre dos bots).

De hecho, Weizenbaum ha sido uno de los grandes críticos de la Inteligencia Artificial, afirmando que, llegado el momento, nunca debería permitirse la toma de decisiones por parte de una inteligencia artificial.



*Ilustración 1. ChatBot. Fuente: Google.*

### **3.2. Visual Studio Code.**

Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Es software libre y multiplataforma, está disponible para Windows, GNU/Linux y macOS. VS Code tiene una buena integración con Git, cuenta con soporte para depuración de código, y dispone de un sinnúmero de extensiones, que básicamente te da la posibilidad de escribir y ejecutar código en cualquier lenguaje de programación.

#### **Características de Visual Studio Code**

**Multiplataforma:** Es una característica importante en cualquier aplicación y más si trata de desarrollo. Visual Studio Code está disponible para Windows, GNU/Linux y macOS.

**IntelliSense:** Esta característica está relacionada con la edición de código, autocompletado y resaltado de sintaxis, lo que permite ser más ágil a la hora de escribir código. Como su nombre lo indica, proporciona sugerencias de código y

terminaciones inteligentes en base a los tipos de variables, funciones, etc. Con la ayuda de extensiones se puede personalizar y conseguir un IntelliSense más completo para cualquier lenguaje.

**Depuración:** Visual Studio Code incluye la función de depuración que ayuda a detectar errores en el código. De esta manera, nos evitamos tener que revisar línea por línea a puro ojo humano para encontrar errores. VS Code también es capaz de detectar pequeños errores de forma automática antes de ejecutar el código o la depuración como tal.

**Uso del control de versiones:** Visual Studio Code tiene compatibilidad con Git, por lo que puedes revisar diferencias o lo que conocemos con git diff, organizar archivos, realizar commits desde el editor, y hacer push y pull desde cualquier servicio de gestión de código fuente (SMC). Los demás SMC están disponible por medio de extensiones.

**Extensiones:** Hasta ahora, he mencionado varias veces el término extensiones porque es uno de los puntos fuertes. Visual Studio Code es un editor potente y en gran parte por las extensiones. Las extensiones nos permiten personalizar y agregar funcionalidad adicional de forma modular y aislada. Por ejemplo, para programar en diferentes lenguajes, agregar nuevos temas al editor, y conectar con otros servicios. Realmente las extensiones nos permiten tener una mejor experiencia, y lo más importante, no afectan en el rendimiento del editor, ya que se ejecutan en procesos independientes.



*Ilustración 2 Visual Studio Code. Fuente: Google*

### 3.3. Sistema Gestor de Base de Datos.

Un sistema gestor de base de datos o SGBD es un software que permite administrar una base de datos. Proporciona el método de organización necesario para el almacenamiento y recuperación flexible de grandes cantidades de datos.



*Ilustración 3. Sistema Gestor de Base de Datos. Fuente: Google.*

### 3.4. Mongo DB

MongoDB (del inglés humongous, "enorme") es un sistema de base de datos NoSQL, orientado a documentos y de código abierto.

En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

MongoDB es una base de datos adecuada para su uso en producción y con múltiples funcionalidades. Esta base de datos se utiliza mucho en la industria, contando con implementaciones en empresas como MTV Network, Craigslist, Foursquare.

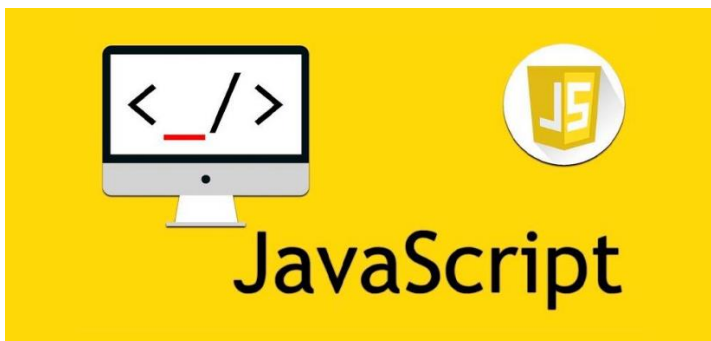
El código fuente está disponible para los sistemas operativos Windows, GNU/Linux, OS X y Solaris.



*Ilustración 4. Mongo DB. Fuente: Google.*

### 3.5 JavaScript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.



*Ilustración 5. JavaScript. Fuente: Google.*

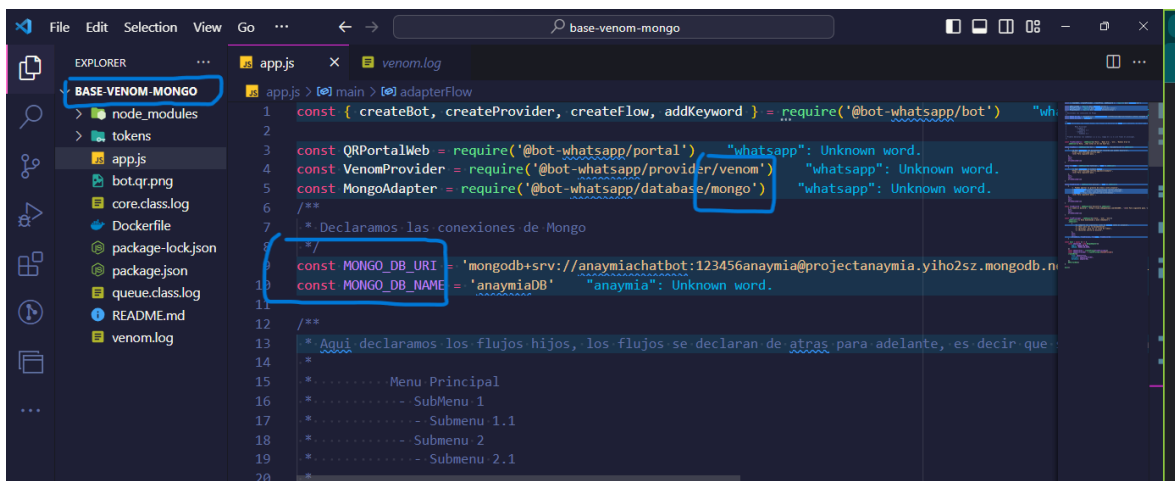
## 4. DESARROLLO

Para la creación de la base de datos del proyecto Ana y Mia se utilizó el sistema gestor de base de datos Mongo DB ya que no tiene un esquema, lo que proporciona más flexibilidad y le permite trabajar con datos estructurados, semiestructurados y no estructurados.

Por otra parte, para la conexión de la base de datos con el ChatBot, se tuvo que crear un ChatBot nuevo haciendo uso de JavaScript y también de librerías. Con lo cual se pudo lograr hacer la conexión con la base de datos además de que ahora es de forma gratuita y también se tiene una conexión con WhatsApp. Ya se llevaron a cabo las pruebas necesarias y el funcionamiento del ChatBot es muy bueno.

#### 4.1 Conexión de la base de datos con el ChatBot Ana y Mia.

4.1.1 Se volvió a crear un nuevo proyecto, con un nuevo proveedor para WhatsApp (No genera grandes cambios), se crea desde la configuración adaptándolo para MongoDB.



```
1 const { createBot, createProvider, createFlow, addKeyword } = require('@bot-whatsapp/bot')
2
3 const QRPortalWeb = require('@bot-whatsapp/portal')
4 const VenomProvider = require('@bot-whatsapp/provider/venom')
5 const MongoAdapter = require('@bot-whatsapp/database/mongo')
6
7 /**
8  * Declaramos las conexiones de Mongo
9  */
10 const MONGO_DB_URI = 'mongodb+srv://anaymiachatbot:123456anaymia@projectanaymia.yiho2sz.mongodb.net'
11 const MONGO_DB_NAME = 'anaymiaDB'
12
13 /**
14  * Aquí declaramos los flujos hijos, los flujos se declaran de atrás para adelante, es decir que
15  *
16  *
17  *
18  *
19  *
20  */
```

Ilustración 6. Creación de nuevo ChatBot. Fuente: Personal.

#### 4.1.2 Se creo una cuenta en MongoDB con la cuenta de correo de Ana & Mia.

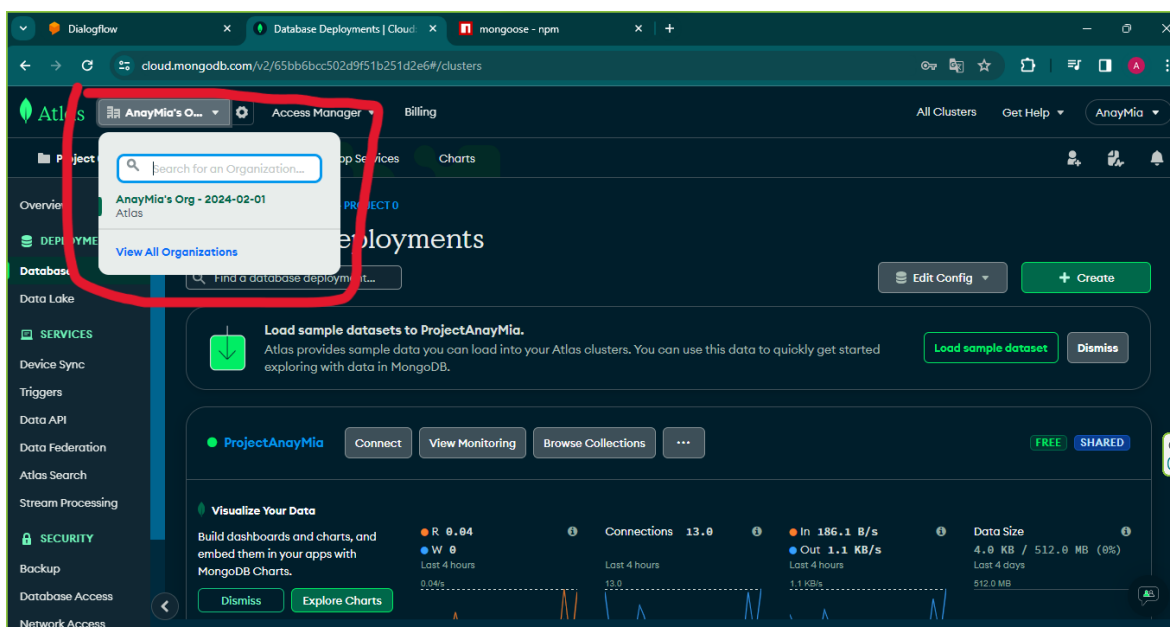


Ilustración 7. Creación de cuenta en Mongo DB. Fuente: Personal.

#### 4.1.3 Se creo un acceso a base de datos; un usuario con una contraseña, para que se pueda lograr la conexión con MongoDB.

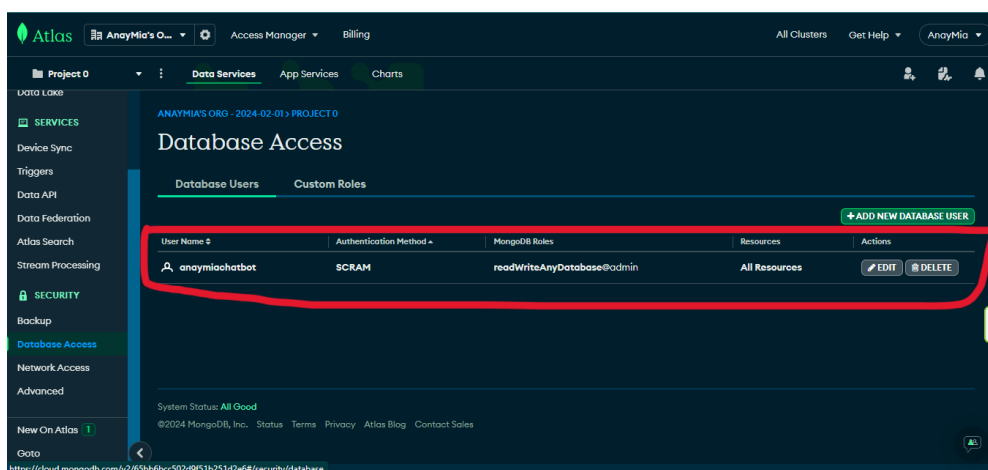
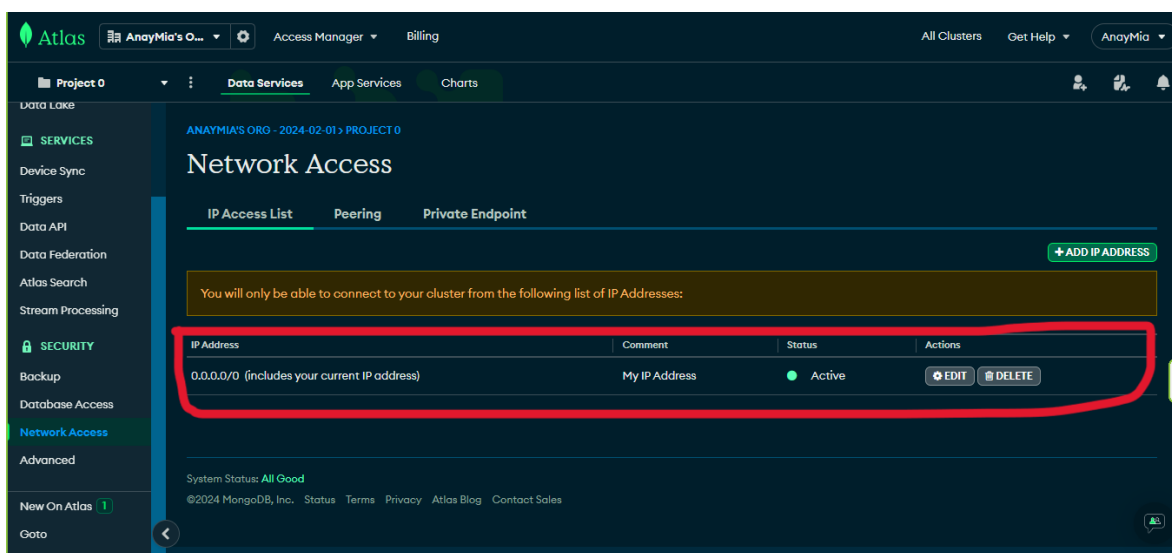


Ilustración 8. Creación de Base de Datos. Fuente: Personal.

**4.1.4 Se creo un acceso de red, para que se pueda encontrar la ruta de acceso hacia el clúster de Ana & Mia, se creó con un acceso libre para que no tuviese ningún problema.**



*Ilustración 9. Acceso de red. Fuente: Personal.*

**4.1.5 Se busca el código de conexión dentro de MongoDB, para que este se pueda comunicar con VSC Code.**

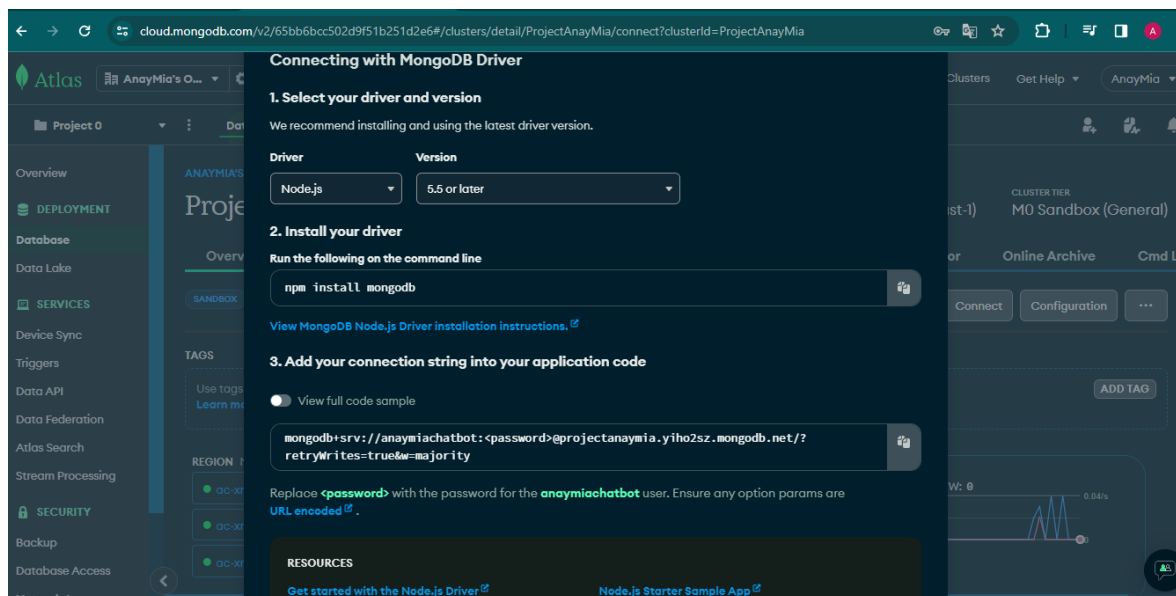
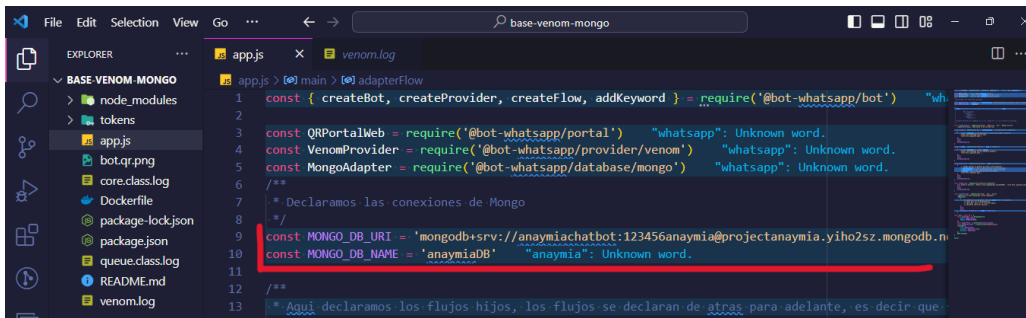




Ilustración 10. Código de Conexión entre Mongo DB con VSC. Fuente: Personal.

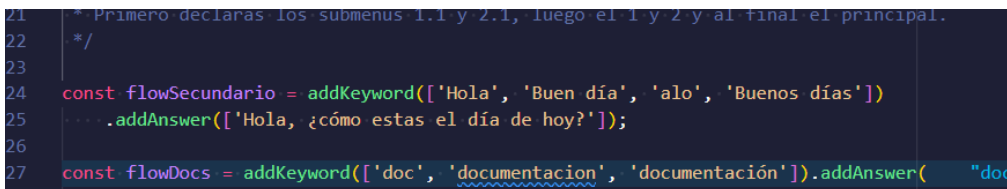
**4.1.6 Se reemplazan los datos en las variables del proyecto. Colocando la contraseña que se creó anteriormente en MongoDB, en el lugar que se indica dentro de la liga de conexión.**



```
1 const { createBot, createProvider, createFlow, addKeyword } = require('@bot-whatsapp/bot')
2
3 const QRPortalWeb = require('@bot-whatsapp/portal')
4 const VenomProvider = require('@bot-whatsapp/provider/venom')
5 const MongoAdapter = require('@bot-whatsapp/database/mongo')
6 /**
7  * Declaramos las conexiones de Mongo
8  */
9 const MONGO_DB_URI = 'mongodb+srv://anaymiachatbot:123456anaymia@projectanaymia.yiho2sz.mongodb.net'
10 const MONGO_DB_NAME = 'anaymiaDB'
11 /**
12  * Aquí declaramos los flujos hijos, los flujos se declaran de atrás para adelante, es decir que...
```

Ilustración 11. Código de conexión con Mongo DB. Fuente: Personal.

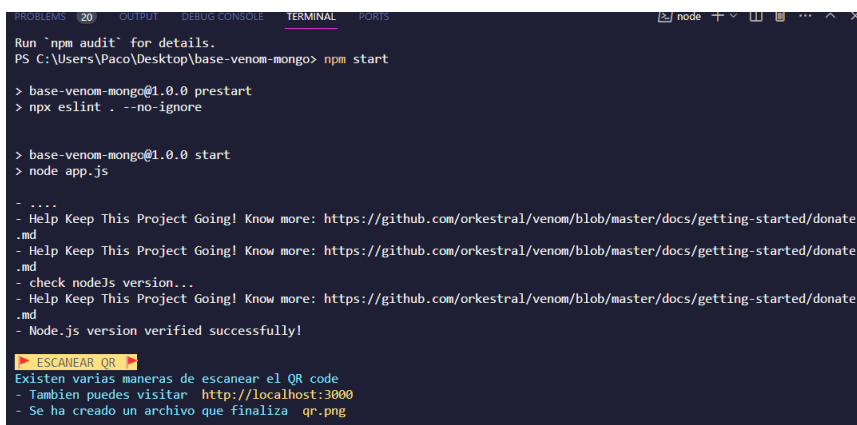
**4.1.7 Se crean los “addKeyword”, los cuales funcionan como los intent para que se pueda hacer la interacción con el usuario.**



```
21 * Primero declaras los submenús 1.1 y 2.1, luego el 1 y 2 y al final el principal.
22 */
23
24 const flowSecundario = addKeyword(['Hola', 'Buen día', 'alo', 'Buenos días'])
25   .addAnswer(['Hola, ¿cómo estas el día de hoy?']);
26
27 const flowDocs = addKeyword(['doc', 'documentacion', 'documentación']).addAnswer('doc')
```

Ilustración 12. Creación de "addKeyword" anteriormente intents. Fuente: Personal.

#### 4.1.8 Se inicia el servicio de NodeJS con “npm start”.



```
Run "npm audit" for details.
PS C:\Users\Paco\Desktop\base-venom-mongo> npm start

> base-venom-mongo@1.0.0 prestart
> npx eslint . --no-ignore

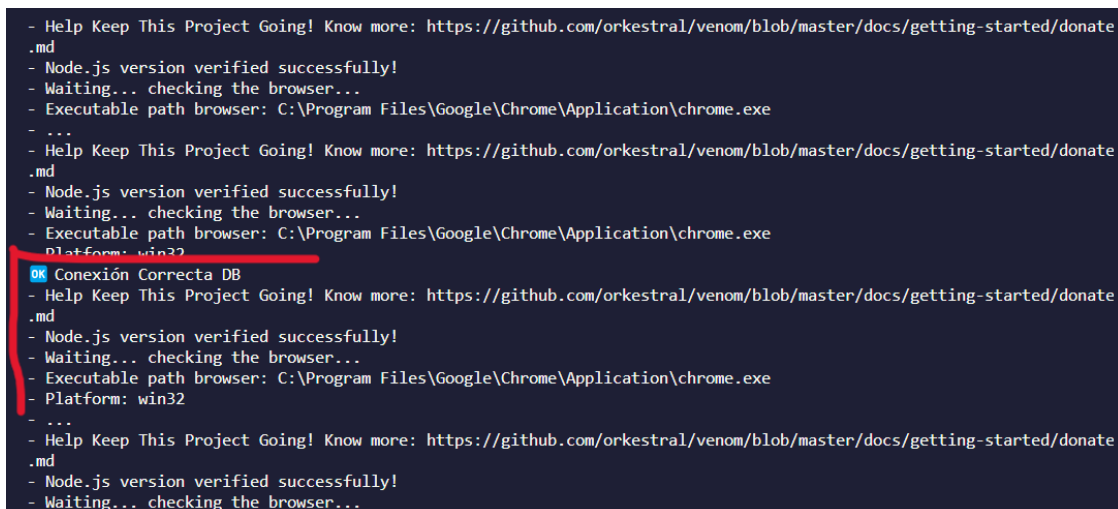
> base-venom-mongo@1.0.0 start
> node app.js

- ....
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- check nodeJs version...
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!

▶ ESCANEAR QR ▶
Existen varias maneras de escanear el QR code
- Tambien puedes visitar http://localhost:3000
- Se ha creado un archivo que finaliza qr.png
```

Ilustración 13. Inicialización del Servicio de NodeJS. Fuente: Personal.

#### 4.1.9 Visualizamos que la conexión con la base de datos fue exitosa.



```
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
- ...
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
- Platform: win32
OK Conexión Correcta DB
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
- Platform: win32
- ...
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
```

Ilustración 14. Conexión con Base de Datos. Fuente: Personal.

#### 4.1.10 El proceso siguiente es escanear el código QR, con un celular que ya cuente con WhatsApp, para que este pueda funcionar como servidor.

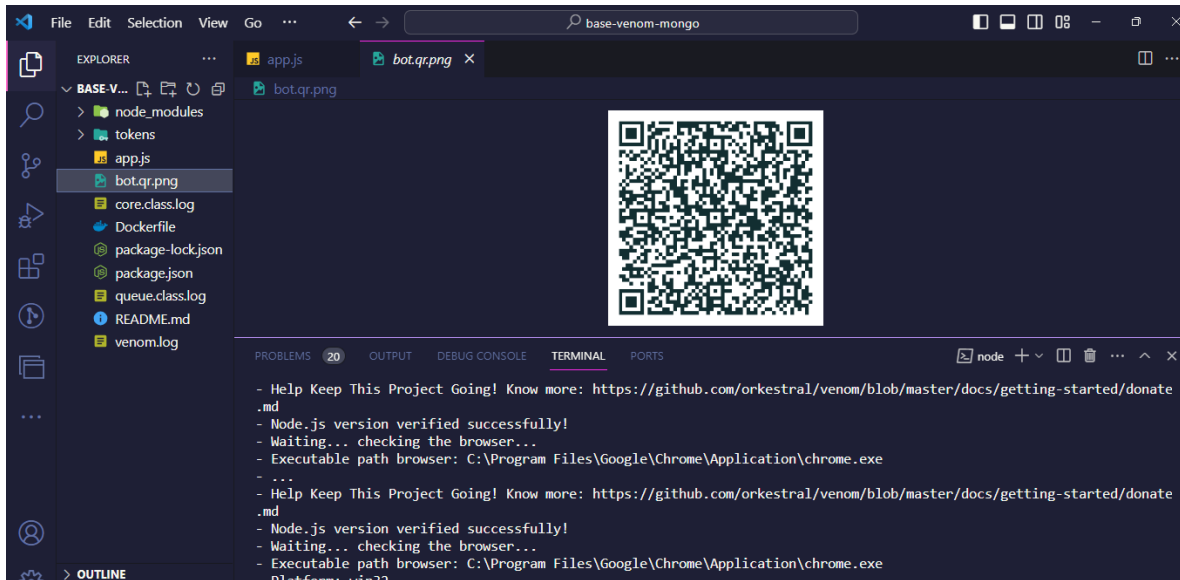


Ilustración 15. Conexión con WhatsApp. Fuente: Personal.

Una vez hecha la conexión del número que funcionará como proveedor, con otro celular se procesa y se realizan las pruebas de interacción. Esta interacción se estará registrando dentro de MongoDB.

## 4.2 Procedimiento de copias de seguridad.

En MongoDB, hay varios enfoques para realizar copias de seguridad de datos, y la elección del método adecuado depende de diversos factores, como los requisitos de recuperación, el tamaño de los datos, la infraestructura disponible y las políticas de retención.

Tipos comunes de copias de seguridad en MongoDB:

1. Copia de seguridad completa (Full Backup).

2. Copia de seguridad incremental (Incremental Backup).
3. Copia de seguridad en punto temporal (Point-in-Time Backup).
4. Copia de seguridad remota (Remote Backup).
5. Copia de seguridad en caliente (Hot Backup).

A continuación, se muestra el procedimiento de copias de seguridad usando la forma más común que es Copia de seguridad completa (Full Backup) esta se lleva a cabo haciendo uso del comando “mongodump” que lo da MongoDB:

#### 4.2.1 Conexión a la base de datos MongoDB:

Antes de realizar la copia de seguridad, debemos de tener una conexión con la base de datos en MongoDB que se desea respaldar.

#### 4.2.2 Ejecutar el comando mongodump:

Se debe utilizar el comando “mongodump” para realizar la copia de seguridad de la base de datos. Este comando nos ayudara a crear una copia de seguridad de todas las bases de datos que tengamos en MongoDB. Por ejemplo:

```
mongodump --out /ruta/del/directorio/de/respaldo
```

*Ilustración 16. Copia de seguridad en Mongo DB. Fuente: Google.*

4.2.3 Se reemplaza “/ruta/del/directorio/de/respaldocon” con la ubicación donde queremos guardar la copia de seguridad.

Verificar la copia de seguridad:

4.2.4 Después de que “mongodump” haya terminado de ejecutarse, debemos de checar que se haya creado la copia de seguridad donde la guardamos.

#### 4.2.5 Opcional: Restaurar desde la copia de seguridad.

Si se necesita restaurar la base de datos desde la copia de seguridad, se puede utilizar el comando “mongorestore” para poder realizarlo. Por ejemplo:

```
mongorestore /ruta/del/directorio/de/respaldo
```

*Ilustración 17. Restauración desde la copia de seguridad. Fuente: Google.*

### 4.3 Protocolo de restauración.

Los protocolos de restauración se refieren a los procedimientos y herramientas utilizadas para recuperar datos de una copia de seguridad en caso de pérdida de datos o corrupción del sistema.

A continuación, se muestran los pasos básicos para restaurar una base de datos en MongoDB:

- 4.3.1 Identificar la copia de seguridad: Para iniciar debemos de tener una copia de seguridad de su base de datos MongoDB, que fue lo que hizo en la sección anterior.
- 4.3.2 Detener el servidor MongoDB: Si el servidor está funcionando, se debe de parar el servicio para evitar tener algún problema durante el proceso de restauración.
- 4.3.3 Restaurar desde un archivo de volcado (dump): Si tenemos un archivo de copia de seguridad echo con el comando “mongodump”, podemos restaurar la base de datos utilizando el comando mongorestore. Ejemplo:

```
mongorestore --db <nombre_de_la_base_de_datos> <directorio_del_dump>
```

*Ilustración 18. Restauración de Base de datos por el comando "mongodump". Fuente: Google.*

4.3.4 Restaurar desde un punto de restauración (oplog): Por otra parte, si se está usando replicación y se tiene un conjunto de réplicas, se puede restaurar los datos utilizando el “oplog”. Se Puede utilizar “mongorestore” con la opción “--oplogReplay” para restaurar desde un punto de restauración.

```
mongorestore --oplogReplay <directorio_del_dump>
```

*Ilustración 19. Restauración usando replicas en Mongo DB. Fuente: Google.*

4.3.5 Verificar la restauración: Una vez completado el proceso de restauración, debemos de verificar que los datos se hayan restaurado correctamente haciendo consultas en la base de datos y comparando los resultados con la copia de seguridad original. Una vez que la restauración se haya completado, reinicia el servidor MongoDB para que los cambios surtan efecto.

#### **4.4 Descripción y propósito de la automatización de tareas.**

La automatización de tareas en la administración de bases de datos es el proceso de utilizar herramientas, scripts o sistemas para ejecutar tareas relacionadas con la gestión, mantenimiento y operaciones de bases de datos de manera programada y sin intervención manual constante.

El propósito de la automatización de tareas es optimizar la eficiencia operativa, mejorar la seguridad y la disponibilidad de los datos, y permitir que los equipos de

administración de bases de datos se centren en tareas más estratégicas y de alto valor añadido.

A continuación, se muestran los propósitos clave de la automatización de tareas en la administración de bases de datos referente a cada tarea dentro de la administración de base de datos. Cabe mencionar que esto aplica para los gestores de base de datos SQL Y NO SQL.

- 4.4.1 **Copia de seguridad y recuperación:** Automatizar la programación y ejecución de copias de seguridad de bases de datos. Garantiza la disponibilidad y protección de los datos críticos y ayuda a minimizar el tiempo de inactividad en caso de desastres.
- 4.4.2 **Monitorización del rendimiento:** Automatizar la monitorización continua del rendimiento de la base de datos permite tomar medidas proactivas para optimizar el rendimiento y evitar problemas futuros.
- 4.4.3 **Gestión de la capacidad:** Automatizar la monitorización y gestión de la capacidad de la base de datos para garantizar que haya suficiente espacio de almacenamiento y recursos disponibles para satisfacer las demandas actuales y futuras de la aplicación.
- 4.4.4 **Parches y actualizaciones:** Automatizar la aplicación de parches de seguridad y actualizaciones de software para garantizar que la base de datos esté protegida contra vulnerabilidades.
- 4.4.5 **Aprovisionamiento y despliegue:** Sirve para agilizar el tiempo de entrega y garantizar la coherencia en la configuración y la seguridad de la base de datos.
- 4.4.6 **Gestión de usuarios y permisos:** Automatizar la gestión de usuarios, roles y permisos para garantizar que los usuarios tengan acceso adecuado a los datos según sus roles y responsabilidades, y para facilitar el cumplimiento de las normativas de seguridad y privacidad.

## 4.5 Métodos de importación y exportación de datos.

Para importar y exportar datos en una base de datos MongoDB, se pueden utilizar varias herramientas y métodos.

A continuación, se muestran los métodos más comunes de importación y exportación de datos para el sistema gestor de base de datos Mongo DB.

### 4.5.1 Importación de datos.

4.5.1.1 **mongoimport**: Esta es una herramienta de línea de comandos proporcionada por MongoDB para importar datos desde archivos JSON, CSV, TSV y BSON.

```
mongoimport --db nombre_basedatos --collection nombre_coleccion --file
```

*Ilustración 20. Importación de datos "mongoimport". Fuente: Google.*

4.5.1.2 **Mongo Shell**: Puede utilizar el Shell de MongoDB para insertar datos manualmente o ejecutar scripts de inserción.

```
db.nombre_coleccion.insertMany([ { documento1 }, { documento2 } ])
```

*Ilustración 21. Importación de datos "Mongo Shell". Fuente: Google.*

4.5.1.3 **MongoDB Compass**: Esta es una interfaz gráfica de usuario que también permite importar datos desde archivos JSON, CSV, TSV y BSON.



4.5.1.4 Herramientas de ETL (Extract, Transform, Load): Puedes usar herramientas como Apache Spark, Talend o Pentaho para transformar y cargar datos en MongoDB desde diversas fuentes.

## 4.5.2 Exportación de datos.

4.5.2.1 mongoexport: Es una herramienta de línea de comandos proporcionada por MongoDB para exportar datos desde una colección a archivos JSON, CSV o BSON. Por ejemplo:

```
mongoexport --db nombre_basedatos --collection nombre_coleccion --out
```

*Ilustración 22. Exportación de Datos "mongoexport". Fuente: Google.*

4.5.2.2 Mongo Shell: También puedes utilizar el Shell de MongoDB para exportar datos.

```
db.nombre_coleccion.find().forEach(function(doc)
```

*Ilustración 23. Exportación de Datos "Mongo Shell". Fuente: Google.*

## 4.6 Lineamientos de seguridad en la base de datos.

Para nuestro proyecto integrador es de suma importancia contar con la mejor seguridad para la base de datos ya que en ella se almacena la información de nuestros usuarios y como se trata de temas delicados nos preocupa mucho la información y que su seguridad sea la mejor para no perderla y también evitando

que quede en las manos equivocadas ya que puede afectar demasiado a nuestros usuarios.

El sistema gestor de base de datos MongoDB nos ofrece varias características y prácticas recomendadas para garantizar la seguridad de nuestra base de datos. A continuación, se muestran algunos de los lineamientos de seguridad.

#### **4.6.1 Autenticación y Autorización:**

Autenticación: Habilita la autenticación para exigir que los usuarios se autentiquen antes de acceder a la base de datos. Se puede configurar la autenticación mediante usuarios y roles.

Autorización: Se puede utilizar roles de usuario para controlar los privilegios de acceso a las bases de datos y colecciones.

#### **4.6.2 Configuración de Red:**

Firewalls: Se recomienda configurar firewalls para limitar el acceso a MongoDB solo a direcciones IP específicas o rangos de direcciones IP confiables.

Bind IP: MongoDB se puede configurar para que escuche en interfaces de red específicas y no en todas las interfaces disponibles.

#### **4.6.3 Encriptación:**

TLS/SSL: Habilitar el cifrado TLS/SSL para proteger la comunicación entre los clientes y el servidor MongoDB, así como entre los miembros de un conjunto de réplicas o un clúster de sharding.

#### **4.6.4 Auditoría de Seguridad:**

Auditoría: Al habilitar la auditoría nos ayuda para registrar eventos de seguridad relevantes, como autenticaciones exitosas o fallidas, y acciones de administración.

#### **4.6.5 Actualizaciones y Parches:**

Parches: Mantén tu instancia de MongoDB actualizada aplicando parches de seguridad y actualizaciones proporcionadas por MongoDB.

#### **4.6.6 Configuración de Contraseñas:**

Contraseñas Fuertes: Requiere que los usuarios elijan contraseñas fuertes y cambien las contraseñas periódicamente.

Hashing de Contraseñas: Almacena las contraseñas como hashes seguros utilizando algoritmos de hashing robustos, como bcrypt.

#### **4.6.7 Seguridad Física y de Acceso Físico:**

Seguridad Física: Protege los servidores MongoDB contra el acceso físico no autorizado y asegura que los servidores estén ubicados en lugares seguros.

Acceso Físico: Limita el acceso físico a los servidores y la infraestructura de red.

#### **4.6.8 Actualización de Configuraciones Predeterminadas:**

Configuraciones Predeterminadas: Cambia las configuraciones predeterminadas de MongoDB según las necesidades de seguridad específicas de tu entorno, como el puerto predeterminado, el tamaño del archivo de registro, etc.

#### **4.6.9 Monitoreo y Registro:**

Monitoreo: Implementa soluciones de monitoreo para detectar y responder rápidamente a posibles amenazas de seguridad o actividades sospechosas.

Registro de Eventos: Registra eventos de seguridad y actividad del sistema para análisis y seguimiento de posibles problemas de seguridad.

#### **4.6.10 Capacitación y Concienciación:**

Capacitación del Personal: Este es muy importante ya que la mayor parte de las vulnerabilidades en una base de datos son causadas por el personal ya que no se encuentran capacitadas o incluso porque no tienen noción de lo antes ya mencionado.

Conciencia sobre Seguridad: Fomenta una cultura de seguridad entre todos los usuarios y empleados que interactúan con la base de datos.

#### 4.7 Reporte de rendimiento de una base de datos con los datos que arroja la herramienta de monitoreo.

La herramienta de monitoreo se incluye en MongoDB, que se encuentra en la pestaña “View Monitoring”; los datos que arroja son las metricas de rendimiento de los componentes en un entorno determinado. A nivel clouster representa el número de servidores, la hora local del servidor, el nombre del host, la versión de Mongo, la instancia de Mongo que se ejecuta, el estado del servidor, entre otros.

A continuación, se muestran los diagramas con la información de rendimiento de la base de datos.

##### 4.7.1 Conexiones.

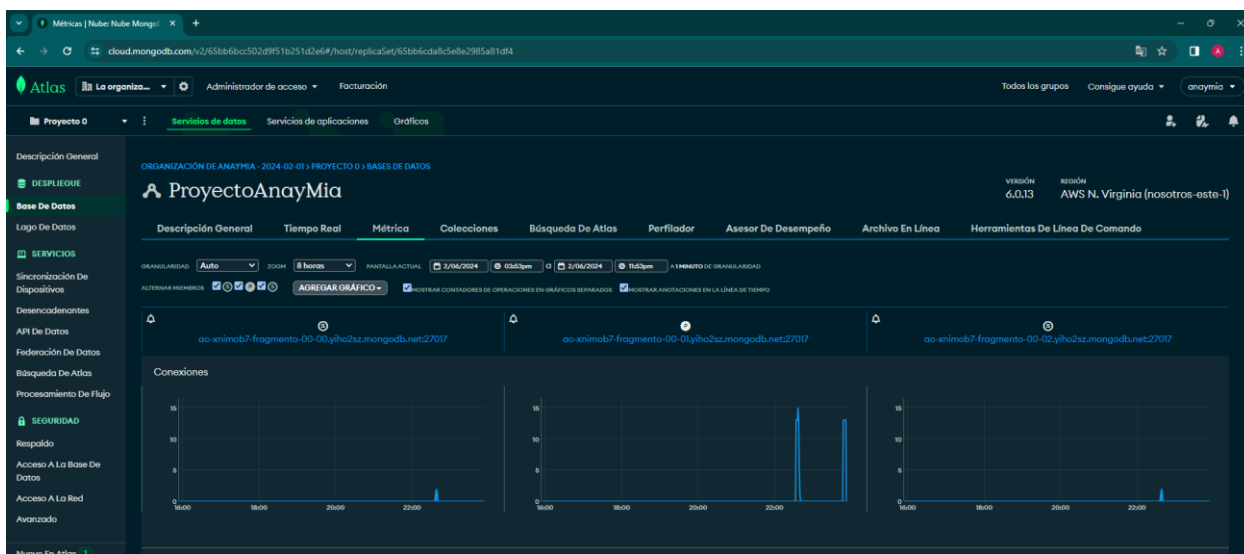


Ilustración 24. Rendimiento Mongo DB (Conexiones). Fuente: Mongo DB.

### 4.7.2 Tamaño lógico.

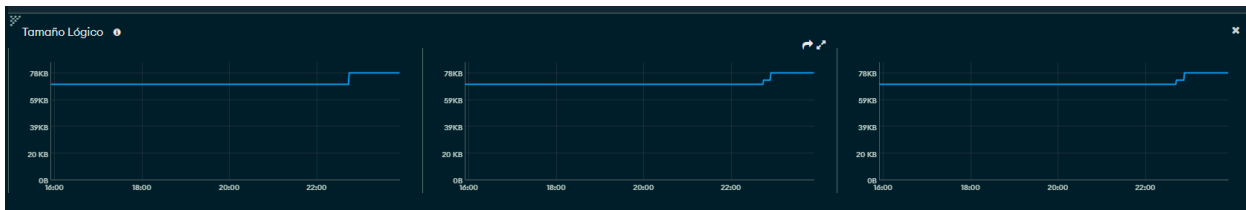


Ilustración 25.Rendimiento Mongo DB (Tamaño lógico). Fuente: Mongo DB.

### 4.7.3 Red.

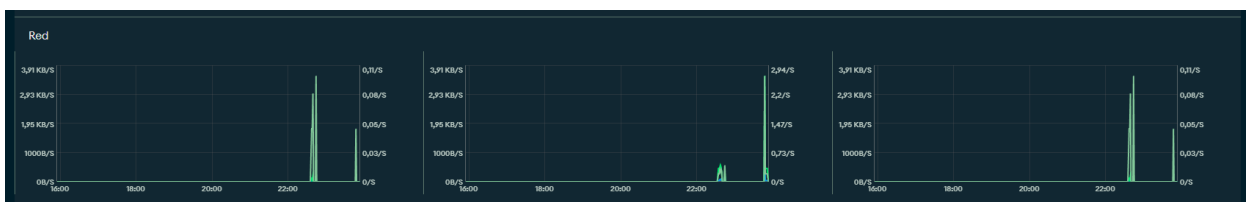


Ilustración 26.Rendimiento Mongo DB(Red). Fuente: Mongo DB.

### 4.7.4 Dominio.

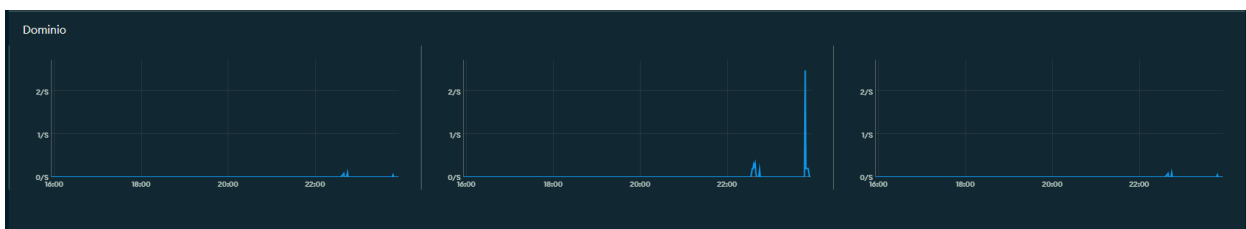
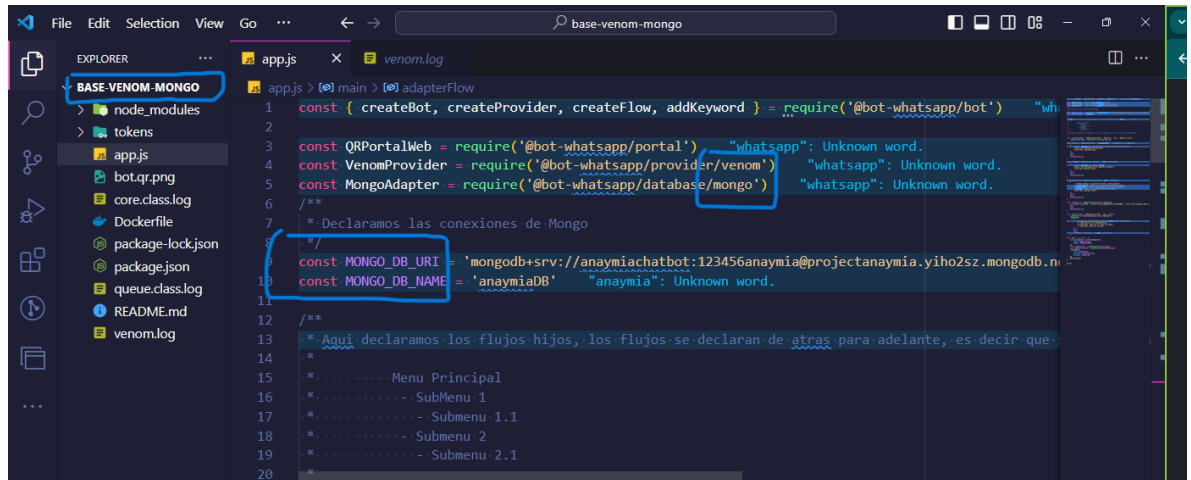


Ilustración 27.Rendimiento Mongo DB (Dominio). Fuente: Mongo DB.

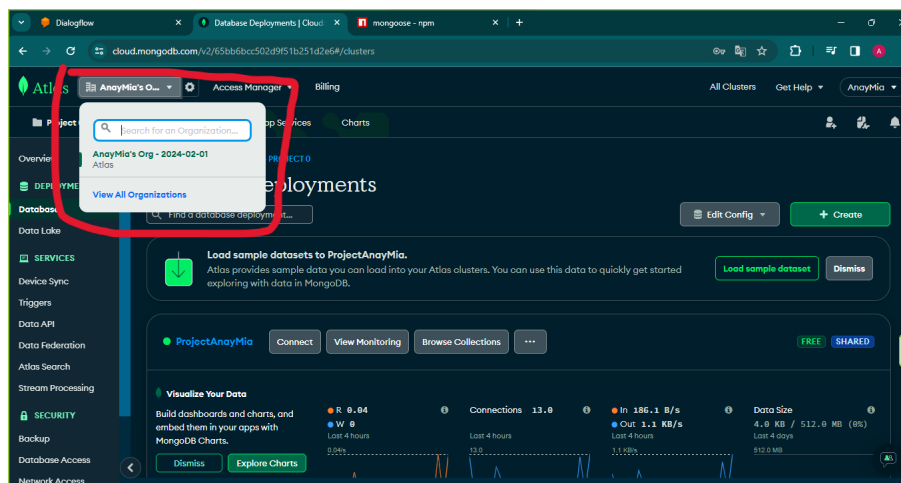
## 5. Aportaciones individuales

Mis aportaciones incluyen la creación de un código realizado en “Visual Studio Code”, con apoyo de la librería “bot-whatsapp”, la cual fue el complemento principal para realizar la adaptación con la aplicación de WhatsApp.



El código se realizó con el entorno de NodeJs, y utilizando MongoDB, como nuestra base de datos; esto porque se manejarán grandes cantidades de información, y hacer uso de una base de datos No-SQL es la mejor opción.

Como parte de mis aportaciones, fue la creación y configuración de una cuenta en MongoDB, para posteriormente hacer la conexión del entorno local de ChatBot, con la base de datos MongoDB.



Atlas | AnayMia's Org | Access Manager | Billing | All Clusters | Get Help | AnayMia

Project 0 | Data Services | App Services | Charts

ANAYMIA'S ORG - 2024-02-01 > PROJECT 0

## Database Access

Database Users | Custom Roles

+ ADD NEW DATABASE USER

User Name	Authentication Method	MongoDB Roles	Resources	Actions
anaymiachatbot	SCRAM	readWriteAnyDatabase@admin	All Resources	EDIT DELETE

System Status: All Good

©2024 MongoDB, Inc. | Status | Terms | Privacy | Atlas Blog | Contact Sales

https://cloud.mongodb.com/v2/69bb6cc502d9f51b251d2e6#/security/database

Atlas | AnayMia's Org | Access Manager | Billing | All Clusters | Get Help | AnayMia

Project 0 | Data Services | App Services | Charts

ANAYMIA'S ORG - 2024-02-01 > PROJECT 0

## Network Access

IP Access List | Peering | Private Endpoint

+ ADD IP ADDRESS

You will only be able to connect to your cluster from the following list of IP Addresses:

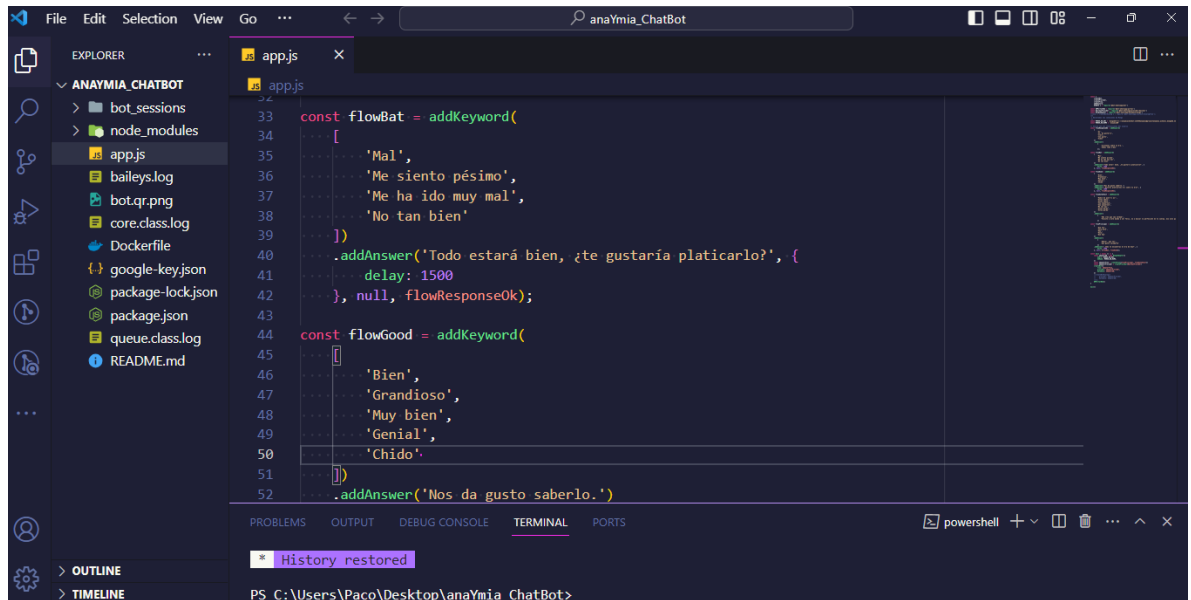
IP Address	Comment	Status	Actions
0.0.0.0/0 (includes your current IP address)	My IP Address	Active	EDIT DELETE

System Status: All Good

©2024 MongoDB, Inc. | Status | Terms | Privacy | Atlas Blog | Contact Sales

```
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
- ...
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
Platform: win32
OK Conexión Correcta DB
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
- Executable path browser: C:\Program Files\Google\Chrome\Application\chrome.exe
Platform: win32
- ...
- Help Keep This Project Going! Know more: https://github.com/orkestral/venom/blob/master/docs/getting-started/donate
.md
- Node.js version verified successfully!
- Waiting... checking the browser...
```

Ya que se hizo el cambio de la ubicación de almacenamiento de la información para la conversación que realizará el ChatBot. Se están creando los flujos de conversación para pueda funcionar de la forma similar a la de DialogFlow.



```
File Edit Selection View Go ... anaYmia_ChatBot
EXPLORER
  ANAYMIA_CHATBOT
    bot_sessions
    node_modules
    app.js
    baileys.log
    bot.qr.png
    core.class.log
    Dockerfile
    google-key.json
    package-lock.json
    package.json
    queue.class.log
    README.md
  OUTLINE
  TIMELINE

app.js
33 const flowBat = addKeyword(
34   [
35     'Mal',
36     'Me siento pésimo',
37     'Me ha ido muy mal',
38     'No tan bien'
39   ])
40 .addAnswer('Todo estará bien, ¿te gustaría platicarlo?', {
41   delay: 1500
42 }, null, flowResponseOk);
43
44 const flowGood = addKeyword(
45   [
46     'Bien',
47     'Grandioso',
48     'Muy bien',
49     'Genial',
50     'Chido'
51   ])
52 .addAnswer('Nos da gusto saberlo.')
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

History restored

PS C:\Users\Paco\Desktop\anaYmia\_ChatBot>



## 6. CONCLUSIÓN

En esta unidad pudimos comprender la importancia de la administración de base de datos.

Consideramos que la administración de la base de datos es una tarea muy importante, pero a la vez es algo complicada ya que de suma importancia ya que se garantiza la integridad, disponibilidad y seguridad de los datos.

Por otra parte, las tareas de administración de base de datos son diversas por lo cual generalmente se consideran varias personas para la administración ya que es algo complicado cumplir con las tareas para cada tarea es por ello por lo que se considera una persona que se especialice en cierta tarea.

Para el proyecto integrador Ana y Mia se ha utilizado el sistema gestor de base de datos Mongo DB ya que nos permite mayor flexibilidad en cuestión de la estructura de los datos y es muy ágil para las consultas, además de que se adapta a nuestras necesidades. Además de que la forma de generar las copias de seguridad es muy sencilla y nos permite y el entorno de desarrollo es muy efectivo para nuestras necesidades.

También es necesario mencionar que la conexión de la base de datos con el ChatBot está funcionando de manera correcta al igual que su integración con Whats App tomando en cuenta que anteriormente solo se podía hacer ciertas consultas por día ahora podemos hacer las que gustemos. Por lo tanto, el ChatBot está mejorando de la mejor manera.

## 7. Bibliografía

- Team, D. (2023, 30 octubre). *MongoDB: todo sobre la base de datos NoSQL orientada a documentos*. Formación en ciencia de datos | DataScientest.com. <https://datascientest.com/es/mongodb-todo-sobre-la-base-de-datos-nosql-orientada-a-documentos>
- Fernández, A. F. (2019, 17 septiembre). Aplicando seguridad a MongoDB. *Antonio Fernández - Desarrollador de software | Freelance*. <https://antoniofernandez.com/seguridad-mongodb-produccion/>
- Francisco Gutiérrez. (2021, 20 marzo). *Exportación e importación de datos en Mongo DB* [Vídeo]. YouTube. <https://www.youtube.com/watch?v=dovW6S06kik>
- T3rr0rZ0n, D. (2022, 27 julio). MongoDB, Como exportar e importar base de datos. \_ void Null. <https://voidnull.es/mongodb-como-exportar-e-importar-base-de-datos/>
- Toli. (2021, 7 abril). *Cómo hacer una copia de seguridad, restablecer y migrar una base de datos de MongoDB en Ubuntu 20.04*. DigitalOcean. <https://www.digitalocean.com/community/tutorials/how-to-back-up-restore-and-migrate-a-mongodb-database-on-ubuntu-20-04-es>
- Fazt. (2017, 27 noviembre). Backups y Restauración en Mongoddb. *Backups y Restauración en Mongoddb*. <https://blog.faztweb.com/2017/11/backups-y-restauracion-en-mongoddb.html>
- MongoDB. (s. f.). *MongoDB: la plataforma de datos para aplicaciones*. <https://www.mongodb.com/es>