

Guía Completa para un Sistema de Envío de Newsletters con Node.js y MySQL

Esta guía detalla los pasos para configurar y ejecutar un sistema de envío de newsletters utilizando Node.js, MJML, Nodemailer y MySQL.

1. Configuración del Entorno y MySQL

Clave SSH y Conexión

- **chmod 400 /ruta/a/tu/archivo/mi-key-pair.pem** Cambia los permisos de tu clave SSH para que solo tú puedas leerla, lo cual es esencial para la seguridad.
- **ssh -i ./Descargas/llave.pem ubuntu@""IP""** Inicia una conexión segura (SSH) a un servidor remoto, usando tu archivo de clave (llave.pem) para autenticarte.

Gestor de Versiones de Node.js (NVM)

- **curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash** Descarga e instala NVM, una herramienta para gestionar múltiples versiones de Node.js en tu sistema.
- **exit** Cierra tu sesión actual; a menudo necesario para que NVM se cargue correctamente después de la instalación.
- **nvm install -lts** Instala la versión LTS (Soporte a Largo Plazo) de Node.js, recomendada por su estabilidad.
- **nvm use -lts** Activa la versión LTS de Node.js instalada.
- **node -v** Muestra la versión actual de Node.js.
- **npm -v** Muestra la versión actual de npm (gestor de paquetes de Node.js).

Instalación y Gestión de MySQL

- **sudo apt update** Actualiza la lista de paquetes disponibles.
- **sudo apt install mysql-client-core-8.0** Instala las utilidades del cliente MySQL, para conectarte a un servidor.
- **sudo apt upgrade -y** Actualiza todos los paquetes instalados a su última versión, confirmando automáticamente.
- **sudo apt autoremove --purge 'mysql*' 'mariadb*' -y** Elimina completamente paquetes antiguos de MySQL o MariaDB y sus archivos de configuración.
- **sudo apt autoclean** Borra archivos de paquetes descargados que ya no son necesarios.
- **sudo apt clean** Limpia el caché local de paquetes descargados.
- **sudo rm -rf /etc/mysql /var/lib/mysql /var/log/mysql /var/run/mysqld** Elimina forzosamente directorios de MySQL (configuración, datos, logs, etc.). ¡Úsalo con extrema precaución para una limpieza total!

- **sudo rm -rf /etc/mysql/conf.d** Elimina específicamente el directorio de configuraciones adicionales de MySQL.
 - **sudo apt dist-upgrade -y** Realiza una actualización de distribución, manejando cambios de dependencias más complejos.
 - **sudo apt install mysql-server -y** Instala el servidor MySQL completo.
 - **sudo systemctl status mysql** Verifica el estado del servicio MySQL (si está funcionando).
 - **sudo mysql_secure_installation** Ejecuta un script para asegurar tu instalación de MySQL (establecer contraseña root, etc.).
 - **sudo mysql -u root -p** Inicia sesión en MySQL como el usuario root (pedirá contraseña).
 - **mysql -u newsletter_user -p newsletter_db** Inicia sesión en la base de datos newsletter_db como newsletter_user (pedirá contraseña).
-

2. Configuración de la Base de Datos MySQL

Una vez que hayas iniciado sesión en MySQL, puedes pegar las siguientes sentencias SQL:

SQL

```
CREATE TABLE clients (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL UNIQUE,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);  
  
INSERT INTO clients (name, email) VALUES  
    ('roberto', 'robertomoramoren03@gmail.com'),  
    ('alvaro', 'alvaromurillopuchalt06@gmail.com');  
  
EXIT; -- para salir
```

3. Configuración de Variables de Entorno

Crea un archivo `.env` para almacenar tus credenciales:

- **nano .env**

Pega el siguiente contenido, asegurándote de reemplazar los valores con tu información real:

```
DB_HOST=localhost
DB_USER=newsletter_user
DB_PASSWORD=Tu_Contraseña_Super_Segura
DB_DATABASE=newsletter_db

EMAIL_SERVICE=gmail
EMAIL_USER=robertomoramoreno3@gmail.com
EMAIL_PASSWORD=La contraseña que tengamos en el correo para aplicaciones.
```

4. Configuración del Proyecto Node.js

Inicialización y Dependencias

- **mkdir newsletter-sender** Crea un nuevo directorio (carpeta) llamado `newsletter-sender`. Este será el espacio de trabajo para tu proyecto.
- **cd newsletter-sender/** Cambia el directorio actual a la carpeta `newsletter-sender` que acabas de crear. A partir de ahora, todos los comandos se ejecutarán dentro de esta carpeta.
- **npm init -y** Inicializa un nuevo proyecto de Node.js en la carpeta actual. El `-y` responde automáticamente "sí" a todas las preguntas, creando un archivo `package.json` por defecto. Este archivo es crucial para gestionar las dependencias de tu proyecto.
- **npm install mjml mysql2 nodemailer dotenv** Instala las dependencias necesarias para tu proyecto:
 - **mjml**: Una librería para construir correos electrónicos responsivos de forma sencilla.
 - **mysql2**: Un cliente para interactuar con bases de datos MySQL.
 - **nodemailer**: Una librería para enviar correos electrónicos desde Node.js.
 - **dotenv**: Para cargar variables de entorno desde un archivo `.env`, lo que es útil para gestionar credenciales y configuraciones sensibles.

Plantilla de Correo MJML

- **nano mjml-template.mjml** Aquí es donde escribirás el código de tu plantilla de correo electrónico usando la sintaxis de MJML.

Y aquí se inserta el código de la plantilla MJML.

Script Principal `index.js`

- **nano index.js** Aquí escribirás el código principal de tu aplicación Node.js.

Dentro pondremos un código que hará las funciones de: **enviar correos electrónicos personalizados a clientes**. Obtener los datos de los clientes de una **base de datos MySQL**, compilar una **plantilla de correo MJML** a HTML, personalizar el HTML para cada cliente y luego enviar los correos utilizando **Nodemailer**, todo mientras gestiona variables de entorno para la información sensible.

5. Ejecución del Sistema de Envío

- **node index.js** Ejecutará el archivo `index.js`, iniciando el proceso de envío de newsletters.