

INFORMACIÓN DE DESPLIEGUE DE LA SOLUCIÓN

Para desplegar la solución que permite administrar Entidades y Empleados mediante los métodos CRUD y un sistema de autenticación y autorización, se deben seguir los siguientes pasos:

1. Clonar el repositorio de la solución desde la plataforma de control de versiones Git (por ejemplo, GitHub).
2. Asegurarse de tener instalado Node.js y MongoDB en el equipo donde se va a desplegar la solución.
3. Abrir una terminal y navegar hasta el directorio donde se encuentra el repositorio clonado.
4. Ejecutar el comando "npm install" para instalar las dependencias del backend de la solución.
5. Navegar hasta la carpeta "client-react" y ejecutar el comando "npm install" para instalar las dependencias del frontend de la solución.
6. Crear un archivo ".env" en la raíz del proyecto y configurar las variables de entorno necesarias para la conexión a la base de datos MongoDB y la generación de tokens JWT. Un ejemplo de archivo ".env" sería:

bashCopy code

```
MONGO_URI=mongodb://localhost:27017/mydatabase JWT_SECRET=mysecretkey
```

7. Ejecutar el comando "npm run dev" para iniciar el servidor de desarrollo del backend y el frontend.
8. Abrir un navegador web y acceder a la dirección "<http://localhost:3000>" para probar la aplicación.

Es importante tener en cuenta que estos son solo los pasos básicos para desplegar la solución en un entorno de desarrollo. En un entorno de producción, se deben tomar medidas adicionales para garantizar la seguridad y el rendimiento de la aplicación, como la configuración de un servidor web, la implementación de medidas de seguridad adicionales y la optimización del rendimiento. Además, se debe asegurar de contar con una estrategia adecuada de copias de seguridad y recuperación ante posibles fallas en la aplicación o en el hardware donde se encuentra alojada.

DESPLIEGUE EN PRODUCCIÓN

Para desplegar en producción usando PM2 para el backend y Apache para el cliente de React, se pueden seguir los siguientes pasos:

1. Crear una instancia de servidor en la nube: En primer lugar, se debe crear una instancia de servidor en la nube que pueda ejecutar tanto PM2 como Apache. Puede ser una instancia de Amazon Web Services (AWS), Google Cloud Platform (GCP) o cualquier otro proveedor de servicios en la nube.

2. Instalar y configurar Apache: Una vez que se tenga una instancia de servidor en la nube, se debe instalar y configurar Apache para poder servir el cliente de React. Esto puede hacerse siguiendo los pasos específicos de cada sistema operativo o proveedor de servicios en la nube.
3. Compilar el código de React: Antes de que se pueda servir el cliente de React con Apache, es necesario compilar el código. Para ello, se puede utilizar el comando "npm run build" dentro de la carpeta del proyecto de React.
4. Configurar Apache para servir el cliente de React: Después de compilar el código de React, se deben configurar las rutas en Apache para servir el contenido estático del cliente. Esto se puede hacer utilizando la configuración de Apache y apuntando a la carpeta "build" dentro del proyecto de React.
5. Instalar y configurar PM2: Para el backend, se debe instalar PM2 en la instancia de servidor en la nube y configurarlo para que se inicie automáticamente cuando se reinicie el servidor. PM2 también se puede configurar para que monitoree el estado de la aplicación y reinicie automáticamente si hay algún fallo.
6. Ejecutar la aplicación con PM2: Una vez que se haya instalado y configurado PM2, se debe ejecutar la aplicación Node.js con PM2. Esto se puede hacer utilizando el comando "pm2 start <archivo_de_inicio>" dentro de la carpeta del proyecto Node.js.
7. Configurar Apache para redirigir al servidor de Node.js: Por último, se debe configurar Apache para redirigir las solicitudes a la aplicación Node.js que se está ejecutando en PM2. Esto se puede hacer utilizando la configuración de proxy de Apache y apuntando a la dirección IP y puerto en el que se está ejecutando la aplicación Node.js.

Con estos pasos, se debería poder desplegar la solución completa en producción, utilizando PM2 para el backend y Apache para el cliente de React. Es importante tener en cuenta que la configuración específica de cada sistema operativo y proveedor de servicios en la nube puede variar, por lo que se deben consultar las guías de cada uno para obtener más detalles.

DOMINIO DE INTERNET

Para asociar el dominio del proyecto con la instancia de servidor en la nube, se deben seguir los siguientes pasos:

1. Comprar un dominio: En primer lugar, se debe comprar un dominio en un proveedor de dominios. Hay muchos proveedores de dominios disponibles, como Namecheap, GoDaddy, Google Domains, entre otros.
2. Configurar la zona DNS: Después de comprar el dominio, se debe configurar la zona DNS en el proveedor de dominios. Esto implicará agregar registros DNS para apuntar el dominio a la dirección IP pública de la instancia de servidor en la nube. Los registros DNS que se deben agregar son un registro A para el dominio principal y cualquier subdominio que se quiera utilizar, como "www". Por ejemplo, si la dirección IP pública de la instancia de servidor es "123.45.67.89", se debe agregar un registro A para "example.com" que apunte a "123.45.67.89" y un registro A para "www.example.com" que también apunte a "123.45.67.89".

3. Configurar el servidor web: Después de configurar la zona DNS, se debe configurar el servidor web para responder a las solicitudes del dominio. En el caso de Apache, esto se puede hacer utilizando la configuración de VirtualHost. Se debe crear un VirtualHost para cada dominio o subdominio que se quiera utilizar y configurarlos para servir el contenido correspondiente.
4. Habilitar HTTPS: Por último, se debe habilitar HTTPS para asegurar la conexión entre el navegador y el servidor. Para esto, se debe obtener un certificado SSL válido para el dominio y configurar el servidor web para utilizarlo. Se puede obtener un certificado SSL gratuito de Let's Encrypt o comprar uno de un proveedor de certificados SSL.

Con estos pasos, se debería poder asociar el dominio del proyecto con la instancia de servidor en la nube y asegurar la conexión con HTTPS. Es importante tener en cuenta que la configuración específica de cada servidor web puede variar, por lo que se deben consultar las guías correspondientes para obtener más detalles.