

DEV.F.:

Clase 4

Implementación de Rutas Privadas

Comenzamos en 10 min

6:40pm (hora CDMX)



DEV.F.:

Implementación de Rutas Privadas

Rutas Privadas en MERN



| Introducción

La autenticación y autorización son dos procesos clave que garantizan que los usuarios adecuados accedan a la información correcta. En una aplicación **MERN** (MongoDB, Express, React y Node.js), esto suele lograrse mediante JSON Web Tokens (JWT), un estándar seguro para la transmisión de información.



| ¿Qué es un JSON Web Token (JWT)?

Un JWT es un token codificado en formato JSON que se utiliza para transferir información de manera segura entre el cliente y el servidor. El token consta de tres partes:

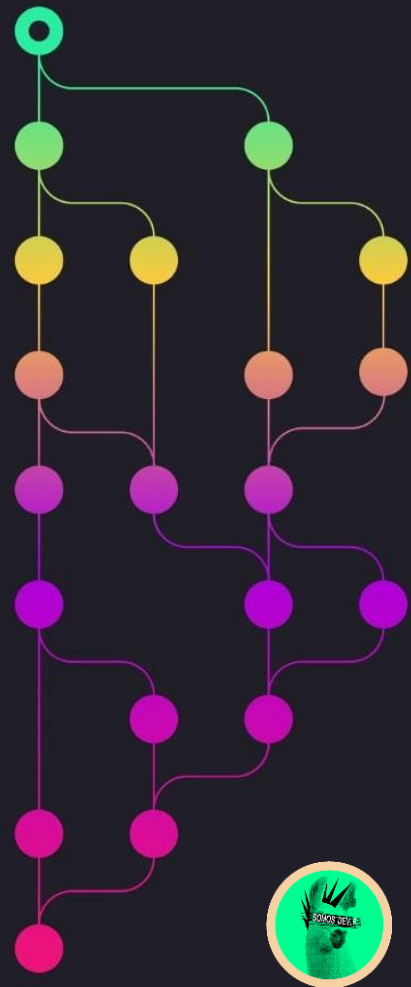
Header:

Contiene el tipo de token (JWT) y el algoritmo de cifrado.

Payload:

Contiene la información del usuario y las reclamaciones (claims).

Signature: Garantiza la integridad del token.

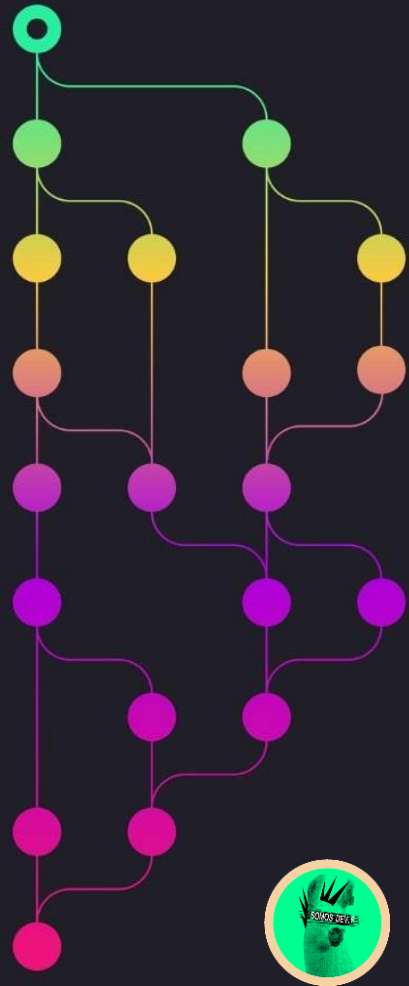


| Ejemplo de JWT decodificado

```
// Ejemplo de JWT decodificado|
{
  "header": {
    "alg": "HS256",
    "typ": "JWT"
  },

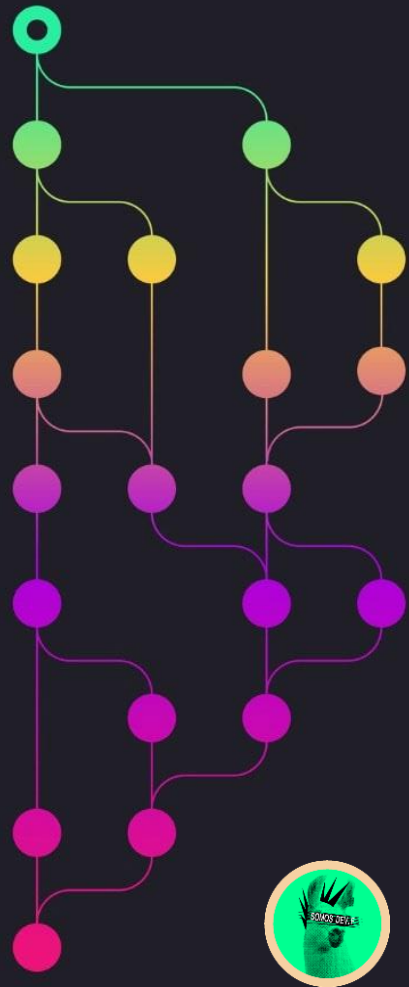
  "payload": {
    "id": "12345",
    "email": "usuario@example.com",
    "role": "admin"
  },

  "signature": "firma_generada"
}
```



| ¿Cómo funciona JWT en MERN?

- El usuario inicia sesión y proporciona credenciales válidas.
- El servidor genera un JWT firmado y lo envía al cliente.
- El cliente almacena el JWT (normalmente en localStorage o cookies).
- En cada solicitud posterior, el cliente envía el JWT para autenticarse.
- El servidor valida el token antes de otorgar acceso.



Práctica

Implementar rutas privadas en MERN



We Can Do It!



CONST : DEV.F

GENIO O HACKER?

DEV.F FUNCTION ()

404

DEV.F = EDUCATION

SOMOS DEV.F

DEV.F