

## HINTS

### RESUMEN DE CONCEPTOS Y DEPENDENCIAS

1. **NodeJS:** Es un entorno de ejecución para desarrollar con JavaScript del lado del servidor.
2. **ExpressJS:** Es un framework que se encuentra sobre nodejs que permite trabajar con el protocolo http y tener sistemas Web de rutas.
3. **MongoDB:** Es una base de datos NoSQL que gestiona documentos json binarios (bson), en lugar de utilizar el sistema clásico de tablas y relaciones, las consultas son más rápidas que un sistema tradicional. Tiene colecciones de documentos y estos documentos son objetos json (bson) y su almacenamiento es de manera binaria que aumenta el rendimiento.
4. **Nodemon:** Utilidad para monitoreo y reiniciar el servidor automáticamente ante cualquier cambio realizado en la IDE.
5. **body-parser:** Permite convertir los datos que nos llegan en las peticiones al servidor en objetos JSON.
6. **Bcrypt-nodejs:** Librería para encriptar contraseñas con el método bcrypt.
7. **Connect-multiparty:** Librería para subir ficheros.
8. **Mongoose:** Es un modelo de programación que permite mapear las estructuras de una base de datos relacional (ORM) que nos provee métodos y funcionalidades para trabajar mejor con mongoDB.
9. **Jwt-simple:** Librería para la gestión de tokens (gestiona la autenticación y cifrado de tokens).
10. **Moment:** Librería para trabajar con fechas.
11. **Moongoose-pagination:** Librería para hacer paginados.
12. **Middlewares:** Es un bloque de código que se ejecuta entre la petición que hace el usuario (request) hasta que la petición llega al servidor.
13. **Modelos:** Representan una entidad de nuestra base de datos, más concretamente un único registro o recurso. Son una abstracción que vamos a utilizar para realizar operaciones en la base de datos. (Están definidos por un esquema).
14. **Esquemas:** Es la estructura de una colección compuesta por atributos.

### REINICIAR LAS APLICACIONES AUTOMÁTICAMENTE CON NODEMON

En Node.js, debe reiniciar el proceso para que los cambios surtan efecto. Eso añade un otro paso a su flujo de trabajo para que los cambios surtan efecto. Puede eliminar este paso adicional usando nodemon para reiniciar el proceso automáticamente.

Nodemon es una utilidad de interfaz de línea de comandos (CLI) desarrollada por @rem que envuelve su aplicación Node, vigila el sistema de archivos y reinicia automáticamente el proceso.

Por lo tanto es un wrapper de node, para reiniciar nuestro API Server cada vez que detecte modificaciones.

Para instalar nodemon en el proyecto realizamos:

```
1 npm i -D nodemon
```

Cada vez que ejecutemos **npm start** ejecutaremos nodemon en vez de **node**. Se tendrá que cambiar el script en el fichero package.json:

```
1 "scripts": {  
2   "start": "nodemon app/index.js"  
3 }
```

## EXTENSIONES RECOMENDADAS PARA VISUAL STUDIO CODE

1. npm
2. npm Intellisense
3. ESLint
4. DotENV
5. Code Spell Checker
6. REST Client
7. Prettier
8. Document This
9. Path Intellisense

## TEST DE LA API

1. [Mocha](#) como test framework
2. [supertest](#) para hacer las peticiones http create