# Nodejs y Mysql, Aplicación Completa (Login, Registro, CRUD, ES6+ y Más )

Aprende a crear una aplicación web Backend y Frontend, utilizando Nodejs, Mysql, Express y otras tecnologías de Javascript. En este ejemplo crearemos una aplicacion de Enlaces, que sera un CRUD de nodejs, más un Login y Registro utilizando Mysql, ademas de proteger nuestras rutas del servidor y estilizar nuestra aplicación con CSS, Bootstrap4, y Handlebars.

INDICE DEL TUTORIAL

1. Backend, Creando el servidor: [04:08](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=248s)

2. Mysql, Conexión: [31:55](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=1915s)

3. Views (Template Engine): [53:28](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=3208s)

4. Guardar datos en Mysql: [01:10:26](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=4226s)

5. Obteniendo Datos desde Mysql: [01:17:35](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=4655s)

6. Eliminando Datos de Mysql: [01:33:47](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=5627s)

7. Actualizar Datos de Mysql: [01:40:58](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=6058s)

8. Mensajes de connect-flash y navegación: [01:53:27](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=6807s)

9. Registro de Usuarios en Mysql: [02:11:28](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=7888s)

10. Login de Usuarios en Mysql: [02:50:56](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=10256s)

11. Protegiendo Rutas del servidor: [03:12:12](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=11532s)

12. Datos Privados: [03:31:05](https://www.youtube.com/watch?v=qJ5R9WTW0_E&t=12665s)

Creamos la carpeta notes-app

Creamos el package.json

npm init –yes

instalamos todas las dependencias:

npm i express express-handlebars express-session mysql express-mysql-session morgan bcryptjs

passport passport-local timeago.js connect-flash

express: framework de node, escribir el server

express-handlebars: es la integracion del motor de plantillas handlebars en express, otras opciones son jade, pug, ejs, jinja, etc

*En general lo que se hace es una de estas dos opciones:*

*1. El Backend envia data "cruda" al Frontend (x ej, un JSON), y el Frontend tiene los templates y los renderiza dinamicamente, usando la data que el Backend envio.*

*2. El Frontend envia data "cruda" al Backend y el Backend tiene los templates y los renderiza dinamicamente, usando la data que el Frontend envio. Como resultado, el Backend envia el HTML renderizado al Frontend.*

*Lo normal hoy en dia es la primer opcion. La 2da opcion seria lo que se hacia antes en un sitio php/asp/etc "clasico".*

*Entiendo que la opcion handlebars seria la 2*

express-session: crear sesiones dentro del server, para auntenticar el usuario y que se recuerde esa autenticacion, almacenamos sus datos temporalmente en una sesion

mysql: modulo de npm para conectarnos a la bd.

express-mysql-session: este modulo almacenara las sesiones en la bd, en lugar del servidor, esto es ideal para cuando la app este en produccion

morgan: permite crear logs o mensajes de que es lo que las aplicaciones cliente estan pidiendo al server

bcryptjs: para cifrar la contraseña, convierte un string en un hash, en nuestro caso la contraseña

passport: modulo para autenticar y manejar el proceso de login del usuario, podemos usar facebook y google

passport-local: complemento de passport para autenticar a los usuarios con nuestra propia bd

timeago.js: convierte los timestamps o fechas de la base de datos en un formato de: 2 minutes ago, 2 hours ago, etc

connect-flash: para enviar mensajes entre multiples vistas, mensajes al usuario

express-validator: validar los datos que nos envia el cliente desde la app cliente

instalamos nodemon:

npm i nodemon –D

-D es una dependencia de desarrollo

reinicia automaticamente el server, esta escuchando los cambios en caliente

creamos carpeta:

\src

creamos las subcarpetas dentro de src

cd src

mkdir lib public routes views

index.js: archivo para arrancar la app

database.js: conexión a mysql

keys.js: para almacenar claves, puertos, etc ,serian como variables de entorno

INSTALAR

MySQL Installer 8.0.15

Windows (x86, 32-bit),

MSI Installer 8.0.15

324.3M

(mysql-installer-community-8.0.15.0.msi)