

# **CURSO DE NODE.JS**

Septiembre 2018

25 horas

# ACCESO AL CURSO

- [Contenido curso en GitHub](#)
- [Contenido vía web]

# PONENTE:

- Juan Daniel Burró Aláez (juandacorreo@gmail.com)
  - Profesor informática en CPIFP Los Enlaces
  - Actualmente en atribución funciones:
    - Desarrollo [nuevo portal Arasaac](#)
    - SPA con React.js
    - API REST con node.js y MongoDB

# TEMARIO DEL CURSO

# COMO EMPEZAR CON NODE.JS

- Instalación de Node
- Gestión de versiones de node
- Uso de linters
- Debug en Node
- Vistazo general a las características de ES6 y ES7.  
Qué puedo usar con Node

# REALIZAR PROYECTOS/PAQUETES CON NODE

- Uso del gestor de paquetes
- Instalación y configuración de paquetes del repositorio de npm
- Gestión de dependencias
- Creación de una librería
- Publicación en npm

# CREAR SCRIPTS DE SERVIDOR MEDIANTE NODE

- Aprender a manejar la naturaleza asíncrona de node: callbacks, promosas, async-await
- Uso de paquetes relacionados con el sistemas de ficheros
- Configuración de microservicios

# NODE.JS PARA WEB:

- Crear un servidor API REST mediante express
- Operaciones CRUD contra base de datos
- Operaciones CRUD autenticadas mediante tokens según OAUTH2
- Uso de Postman para comprobar API
- Uso de WebSockets



# **ENFOQUE DEL CURSO**

# CONFIGURACIÓN INICIAL

- Trabajamos con una máquina virtual parcialmente preparada:
  - Visual Code con extensiones
  - Docker
  - zsh
  - ...
- Está [documentada la instalación](#)
- Lo veremos por encima y se mencionará donde se utilice

# INTRODUCCIÓN A NODE.JS

- [Cómo empezar con node.js](#)
  - Instalación
  - Gestión de versiones
  - Características principales

# PROYECTOS

- Proyectos pequeños donde veremos los contenidos:
  - npm
  - Promises
  - Servidor web con node y arquitectura API REST
  - Autenticación con json web tokens
  - WebSockets

- Implementaremos en varios proyectos las siguientes tecnologías:
  - Gestión versiones mediante git
  - Test con enfoque TDD (Test Driven Development)
  - Dockerización de servidores adicionales (MongoDB)
  - Despliegue serverless

# **CONTENIDO DE LOS PROYECTOS**

# PROYECTO 1: APLICACIÓN ¡HOLA USUARIO!

- *Escribe un mensaje en un fichero:*

```
Hola usuario, tienes 17 años
```

- *Usuario es el username del equipo*
- *Edad es una variable*
- **Uso de módulos propios y built-in modules**

# PROYECTO 2: DOCUMENTACIÓN

## APUNTES

- *Generación de slides y pdf a partir de markdown, útil para hacer apuntes*
- **Uso de módulos de terceros**
- Configuración de proyecto y gestión de paquetes mediante **npm**



# PROYECTO 3: CREAR UNA LIBRERÍA

- *Crear una librería a partir de un json de datos*
- Configuración de **linters**
- Uso de **tests**
- **Publicación en el repositorio de npm**

# PROYECTO 4: LISTA DE TAREAS

- *Clásico ejercicio de todo list en fichero*
- Uso de json.

# PROYECTO 5: API REST

- *Servidor web con implementación de API REST*
- Uso básico librería express, middlewares y arquitectura MCV
  - Uso de MongoDB como base de datos
  - Uso de docker para virtualizar servicio de bbdd
- Uso de POSTMAN para tests de la api autenticada

# PROYECTO 6: AUTENTICACIÓN

- *Autenticación en servidor web*
- Uso de json web tokens
- Uso de POSTMAN para tests de la api autenticada

# PROYECTO 7: WEBSOCKETS

- Clásico ejercicio de todo list en fichero
- Uso de librería socket.io

# ¿EMPEZAMOS?

- Configuración inicial