

«Talento Tech»

Back-End

Node JS

Clase 01



Clase N° 1 - Bienvenida y Herramientas

Temario:

1. **Bienvenida**
 - Presentación del curso.
 - Temario
 - Buenas prácticas de la cursada
2. **Herramientas y tecnologías utilizadas**
 - Presentación de las tecnologías clave en Node (Javascript, Node Js, Express Js, Firestore, JWT, Patrones de Arquitectura).
3. **Instalación y configuración de Visual Studio Code**
 - Introducción a Visual Studio Code como herramienta principal de desarrollo.
 - Paso a paso para la instalación y configuración básica de Visual Studio Code.
 - Plugins recomendados
4. **Instalación y configuración de Node Js y NPM**

Objetivos de la Clase

En esta clase abordaremos los siguientes objetivos:

- Recibir la bienvenida y presentación del curso.
- Conocer las herramientas a utilizar a lo largo de la cursada.
- Finalizar con el setup inicial y la instalación de las herramientas.
- Resolver dudas y consultas antes de comenzar el curso.



Presentación del curso

El mercado laboral actual es cada vez más exigente, buscando perfiles especializados en roles concretos pero que también tengan la versatilidad de desenvolverse en un ecosistema tecnológico más amplio. Con el constante surgimiento de nuevas herramientas y frameworks para el desarrollo web, es crucial que los desarrolladores frontend no sólo dominen la interfaz de usuario, sino que también comprendan conceptos relacionados a entornos de ejecución, servidores, compiladores, patrones de arquitectura, estrategias de renderizado y flujos de comunicación con el backend. Además, habilidades como trabajar con datos en soluciones serverless y gestionar almacenamiento en la nube se han convertido en un gran diferencial.

Por otro lado, los desarrolladores backend deben tener un sólido conocimiento de JavaScript y su ejecución en el lado del servidor, especialmente en un contexto donde los servicios web han experimentado un crecimiento exponencial en los últimos años.

En este escenario, Node.js se destaca como una de las tecnologías más demandadas en el desarrollo web moderno, reconocida por su rendimiento, escalabilidad y facilidad para construir aplicaciones del lado del servidor. En este curso, aprenderemos desde los fundamentos de Node.js hasta la creación de APIs REST completas, abordando buenas prácticas, integración con servidores de datos y la implementación de sistemas de autenticación modernos, equipándote con las habilidades necesarias para destacar en este competitivo mercado.



Nuevos conceptos.

Para llevar adelante nuestra misión, tendremos que conocer nuevos conceptos, herramientas y tecnologías, en particular sobre el desarrollo en el mundo backend pero también que nos permitirán reforzar y afianzar nuestros conocimientos en frontend y nos ayudarán a adquirir una visión global de cómo funciona el flujo de comunicación de la web.

A lo largo de 16 clases aprenderemos a fondo sobre los siguientes temas:

- **Fundamentos de Node.js y JavaScript Moderno:** veremos cómo funciona este entorno de ejecución y profundizaremos en el uso de JavaScript del lado backend.
- **Comunicación entre cliente y servidor:** conoceremos el lenguaje con el que se comunican los sistemas a través de la web.
- **Patrones de Arquitectura:** aprenderemos a desarrollar aplicaciones escalables y fáciles de mantener utilizando principios y patrones de desarrollo modernos.
- **APIs RESTful con Express.js:** aprenderemos a diseñar y desarrollar interfaces que conecten aplicaciones web con bases de datos y servicios externos.
- **Bases de datos con Firestore:** integraremos nuestra aplicación con bases de datos NoSQL para almacenar y consultar datos de manera eficiente.
- **Autenticación y Seguridad:** implementaremos autenticación segura con JSON Web Tokens (JWT) protegiendo nuestros proyectos de accesos no autorizados.

Herramientas y tecnologías utilizadas.



- **Node JS:** no es el único, pero sí el más famoso entorno de ejecución de Javascript del lado del servidor.



- **Express JS:** una herramienta capaz de complementar a Node JS a la perfección con el fin de crear soluciones de servidor web con javascript.



Cloud Firestore

- **Firestore:** uno de los servicios de gestión de datos en la nube más conocidos y utilizados. Aprenderemos sobre el concepto de tecnologías serverless, sus diferencias y similitudes con las bases de datos tradicionales y cómo implementarlas del lado del backend.



POSTMAN

- **POSTMAN:** con esta herramienta podremos probar nuestros servicios de una manera ágil y gráfica.



- **NPM:** el gestor de paquetes de Node JS más conocido, con esta herramienta tendremos acceso a código de terceros que nos facilitará el desarrollo.



- **JWT:** los json web tokens nos ayudarán a manejar la capa de seguridad de nuestras aplicaciones y evitar accesos no autorizados.



- **API Rest:** aprenderemos sobre patrones de arquitectura y porque este en particular es tan importante para Node JS.



- **Visual Studio Code:** instalaremos el IDE o entorno de desarrollo más famoso y también el más preparado para trabajar con todas estas tecnologías.



- **Javascript:** profundizaremos mucho más en el uso, implementación y sintaxis del lenguaje de programación más utilizado en el mundo web.

Instalación y Configuración de Visual Studio Code

Visual Studio Code (VS Code) es un editor de texto ligero y potente desarrollado por Microsoft, diseñado específicamente para programadores. Es una herramienta gratuita, de código abierto, y compatible con Windows, macOS y Linux, que se ha convertido en una de las favoritas en la comunidad de desarrollo gracias a su versatilidad y facilidad de uso.



Visual Studio Code

1. Descarga e Instalación

- Visitá el sitio oficial de Visual Studio Code: <https://code.visualstudio.com/>
- Descargá la versión correspondiente a tu sistema operativo (Windows, macOS, Linux).
- Seguí las instrucciones de instalación en pantalla. El proceso es sencillo y no requiere configuración especial.



2. Configuración Básica

- **Abrir Visual Studio Code:** Una vez instalado, abrí el editor.
- **Configurar Preferencias:**
 - Ve a **File > Preferences > Settings**.
 - Desde aquí, podés personalizar el tema, fuentes, y otros aspectos visuales.
 - Puedes instalar plugins ajustar la estética visual y herramientas que sean de ayuda en el desarrollo.

Ejercicio Práctico:

Instalación y Configuración de Node JS y NPM



Ambas tecnologías se instalan de forma conjunta mediante la descarga y ejecución del instalador de Node JS.

1. Descarga e Instalación

- Visitá el sitio oficial de Node Js: <https://nodejs.org>
- Descargá la versión correspondiente a tu sistema operativo (Windows, macOS, Linux).
- Seguí las instrucciones de instalación en pantalla.



2. Verificación de la instalación:

- Abrir la terminal e ingresar el comando: `node -v` el cual nos debería arrojar la versión de Node JS recientemente instalada.

```
→ node -v
v20.17.0
```

- Luego ingresamos el comando `npm -v` donde si todo se ha instalado correctamente, debería indicar la versión actual de NPM

```
→ npm -v
10.8.2
```

Materiales y Recursos Adicionales:

Documentación oficial de Node.js: [Node.js Docs](#)

Guía oficial de Express.js: [Express.js Docs](#)

Introducción a Firestore: [Firebase Firestore](#)

Descarga e instalación de Visual Studio Code: [VS Code](#)

Preguntas para Reflexionar:

1. ¿Por qué es importante contar con un entorno de desarrollo bien configurado antes de comenzar a programar?
2. ¿Qué ventajas ofrece Node.js frente a otros entornos de ejecución en términos de rendimiento y escalabilidad?
3. ¿Cómo crees que las herramientas presentadas facilitarán el desarrollo de aplicaciones web durante este curso?



Próximos Pasos:

Introducción a Node JS: Aprenderemos sobre este entorno de desarrollo de Javascript, sus características principales y su funcionamiento.

Funciones y Arrays: Conoceremos más a fondo para que sirven las funciones, que formas existen para su declaración y el concepto de “**funciones de orden superior**”. Además, aprenderemos sobre los Arrays y los métodos disponibles para trabajar con ellos.

Clases y Objetos: Introducción a la Programación Orientada a Objetos (POO). Aprenderemos qué son las clases, cómo crear objetos y cómo trabajar con métodos y propiedades.



Buenos Aires
aprende
Agencia de Políticas para el Futuro

BA Buenos
Aires
Ciudad