



«Talento Tech»

Back - End Node Js

PLAN DE ESTUDIOS 2025

MODALIDAD VIRUTAL

1 Clase virtual sincrónica semanal de 2 hs (obligatoria).
1 Clase de consulta de 1 h (opcional).

ACREDITACIÓN

- 70% de asistencia a clases virtuales semanales.
- Aprobación de cuestionarios parciales de contenidos.
- Entrega y aprobación de Proyecto Final Integrador.

DURACIÓN

1 cuatrimestre de 16 clases.

CERTIFICACIÓN

Microcredencial oficial del Ministerio de Educación de la Ciudad de Buenos Aires.

Fundamentación

El curso busca introducir a los estudiantes al desarrollo web backend, brindándoles herramientas para manejar entornos de servidor mediante Node.js y Express.js. Estas competencias permiten complementar los conocimientos frontend de los participantes, sentando las bases para perfiles fullstack altamente demandados en el mercado laboral. Se enfoca en el desarrollo de aplicaciones escalables y mantenibles, esenciales para la creación de ecosistemas y soluciones web modernas.

Contenidos mínimos

- ▶ Conocer a profundidad la comunicación entre cliente y servidor.
- ▶ Entender el protocolo HTTP, sus métodos y códigos de respuesta.
- ▶ Crear servidores web multipropósito mediante el uso de Node JS y Express JS.
- ▶ Desarrollar aplicaciones con arquitectura API Rest.
- ▶ Profundizar sus conocimientos en el lenguaje de programación Javascript del lado del servidor.
- ▶ Trabajar con patrones de arquitectura e implementar soluciones escalables y mantenibles.

Contenidos mínimos



Atender y responder solicitudes de servidor entregando datos a demanda en formato JSON.



Configurar y consumir un esquema de modelo de datos en la nube basado en el servicio Firestore del proveedor Firebase.



Conocer sobre estándares de seguridad y autenticación.

Objetivo General

Capacitar a los estudiantes en la creación de aplicaciones web de servidor utilizando la arquitectura API REST, profundizando sus conocimientos en JavaScript del lado del servidor y en tecnologías relacionadas.

Objetivos específicos:

- Desarrollar aplicaciones web del tipo API REST.
- Dominar patrones de arquitectura y comprender las diferencias entre cliente y servidor.
- Utilizar Node.js y Express.js para crear aplicaciones backend.
- Comprender el uso y gestión de paquetes, módulos nativos y de terceros.
- Implementar soluciones con servicios dinámicos en la nube, como Firestore, y diferenciarlas de bases de datos tradicionales.
- Implementar y desarrollar capas de autenticación.
- Integrar conocimientos adquiridos en un proyecto final que combine las tecnologías y herramientas aprendidas.

Resultados de aprendizaje

Al finalizar este curso, los estudiantes serán capaces de:

- Construir servidores web multipropósito con Node.js y Express.js.
- Implementar aplicaciones basadas en la arquitectura API REST.
- Comprender el protocolo HTTP, incluyendo métodos y códigos de respuesta.
- Utilizar herramientas como POSTMAN para probar y desarrollar servidores.
- Configurar y conectar aplicaciones con servicios de almacenamiento en la nube como Firestore.
- Diseñar aplicaciones backend escalables y mantenibles aplicando conceptos de arquitectura.
- Desarrollar una capa de autenticación estándar con JWT y middlewares.
- Manejar errores en el servidor y procesar datos en formato JSON.
- Utilizar JavaScript eficientemente en entornos backend con Node.js.

Temario



Temario

CLASE 01 >> Bienvenida y Herramientas

- Sobre Talento Tech
- Presentación del curso y su dinámica
- Recomendaciones de buenas prácticas para aprovechar al máximo la cursada.
- Introducción a los proyectos y ejercicios semanales que permitirán poner en práctica los conocimientos adquiridos.
- Revisión de las herramientas fundamentales que utilizaremos en el curso
- Configuración del entorno de desarrollo para facilitar una experiencia fluida
- Ejercicios

CLASE 02 >> Introducción a Node Js

- Fundamentos:
 - ¿Qué es Node Js? Sus características principales y cómo permite ejecutar JavaScript fuera del navegador
- Diferencias con Javascript en el browser:
 - Comparación entre el entorno del navegador y el entorno de ejecución de Node Js (Browser vs Runtime Environment)
- Funcionamiento del core de Node Js:
 - Modelo de single-threading y la diferencia con modelos multi-threading
- Cuestionario en Campus Virtual

CLASE 03 >> Funciones, Arrays y Methods

- Funciones:
 - Tradicionales vs arrow functions
 - HOF: Higher order functions (funciones como argumentos)
- Arrays:
 - Declaración
 - Métodos de arrays más utilizados
- Template Literals:
 - Uso de backticks para strings dinámicos y multilínea.
- Ejercicios

Temario

CLASE 04 >> Objetos y operadores avanzados

- Objetos:
- Literales, funcionales y de clase
- Javascript moderno:
- Operadores avanzados, destructuring y spread operator
- Ejercicios
- Cuestionario en Campus Virtual
- Ruta de Avance

CLASE 05 >> Módulos y gestores de paquetes

- Gestores de paquetes:
 - ¿Qué son? ¿Para qué sirven?
- NPM - Node Package Manager:
 - Instalación de paquetes, creación de scripts y gestión de dependencias.
- Inicio de un proyecto nuevo con Node Js y NPM
- Módulos Nativos:
 - Introducción a módulos esenciales de Node Js como path, fs y process
- Módulos de terceros:
 - Descargando dependencias externas.
- Dirname:
 - Gestión de rutas absolutas y acceso a archivos del servidor
- Ejercicios

CLASE 06 >> Asincronismo

- Fundamentos de asincronismo
- Manejo de promesas:
 - then, catch & finally - async & await
- Uso de **Fetch** para consumir datos externos
- Ejercicios

Temario

CLASE 07 >> Servidores Web

- Funcionamiento básico de servidores y cómo gestionan las solicitudes de los clientes
- Relación cliente/servidor (interacción entre los navegadores y los servidores)
- Protocolo HTTP
- URI: URL + URN, composición de las rutas que navegan la web
- Consignas de la Pre-entrega del Proyecto Final
- Cuestionario en Campus Virtual

CLASE 08 >> Patrones de Arquitectura

- Métodos HTTP
- Códigos HTTP
- ¿Qué son los patrones de arquitectura?:
 - Repaso por los patrones más comunes.
- MVC vs API Rest:
 - Diferencias y similitudes.
- Pre-Entrega de Proyecto Final
- Cuestionario en Campus Virtual

CLASE 09 >> Creando un Servidor Web

- Servidor con Node nativo:
 - Construcción de un servidor básico utilizando únicamente Node JS
- Express JS:
 - Instalación, configuración y uso de Express Generator para crear una estructura inicial
- Desarrollo de un servidor web con acceso desde la carpeta pública
- Middlewares:
 - Introducción a los middlewares y su propósito
- Ejercicios

Temario

CLASE 10 >> **Modelado de una API Rest**

- Estructura de archivos y carpetas
- Introducción a las capas de controladores, modelos, rutas y servicios
- División de responsabilidades
- Manejando solicitudes y respuestas desde el servidor
- Ejercicios
- Cuestionario en Campus Virtual

CLASE 11 >> **Request & Response: entendiendo la comunicación entre aplicaciones**

- Rutas en detalle
- CORS: solicitudes entre dominios
- Error Handle (404)
- Path y query params
- POSTMAN
- Ejercicios

CLASE 12 >> **Capa lógica: controlando la respuesta de nuestra aplicación**

- Controladores y Servicios
- Manejando la lógica de nuestra aplicación
- Ejercicios
- Cuestionario en Campus Virtual
- Ruta de Avance

Temario

CLASE 13 >> **Modelo de datos y trabajo con JSON**

- Modelos
- Formato JSON, métodos parse y stringify
- Acceso a los datos mediante archivos
- Ejercicios

CLASE 14 >> **Datos en la nube: configurando y accediendo a datos desde un servidor externo**

- Firebase/Firestore,
- Creando nuestro servicio de almacenamiento de datos
- Consumiendo Firestore desde nuestra API
- Devolviendo los datos mediante nuestra arquitectura RestFul
- Ejercicios
- Cuestionario en Campus Virtual

CLASE 15 >> **Autenticación y Autorización**

- Introducción a la Autenticación:
 - Token JWT (conceptos básicos y uso)
 - Uso de middlewares para proteger rutas
- Buenas prácticas de seguridad
- Consignas del Proyecto Final
- Ejercicios

CLASE 16 >> **Fin de la cursada: proyecto final y despedida**

- Subida a producción
- Despedida y próximos pasos
- Entrega del Proyecto Final
- Ejercicios

Proyecto Final Integrador:

Al finalizar el curso cada estudiante habrá desarrollado una API Rest como servicio web backend con Node Js y Express Js para una tienda en línea, proyecto que les habrá permitido implementar los conocimientos adquiridos sobre la relación cliente/servidor, el protocolo de comunicación HTTP, sus métodos y códigos de respuesta, como capturar y devolver solicitudes a través de endpoints que apunten a rutas predefinidas, leyendo parámetros y entendiendo cada parte que compone dichas solicitudes.

Además, sabrán separar las distintas capas de una aplicación a través del concepto de división de responsabilidades, estructurando la capa lógica de la aplicación dentro de los controladores y los servicios, utilizando módulos y librerías internas y externas como así también, habrán aprendido a modelar el acceso a los datos para consumirlos localmente desde archivos en formato JSON o desde servidores en la nube en servicios como Firestore, entendiendo la diferencia con servidores de bases de datos tradicionales.

Finalmente, habrán configurado una capa de autenticación para proveer al sistema de seguridad bajo los estándares del mercado.

Con esta experiencia, los estudiantes podrán afianzar y complementar sus conocimientos como desarrolladores frontend y sentar las bases para explorar el mundo backend en profundidad, con la idea de alcanzar un perfil fullstack.



Recursos Necesarios

- ▶ Computadora con conexión a Internet
- ▶ Cámara y micrófono
- ▶ Espacio en memoria para instalación de herramientas (ej. Visual Studio Code)

Requisitos para recibir la acreditación

- 70% de asistencia a clases semanales de dos horas.
- Realización de cuestionarios obligatorios presentes en campus.
- Entrega y aprobación del proyecto final integrador.



Buenos Aires
aprende
Agencia de Habilidades para el Futuro

BA Buenos
Aires
Ciudad