



# FRONT-END JS

Trayecto - 2024

Duración: Cuatrimestral

Modalidad: Virtual

## Fundamentación:

El curso de Front-End está diseñado para proporcionar a los estudiantes una sólida base en el desarrollo web del lado del cliente, abarcando desde los conceptos básicos de HTML y CSS hasta la implementación de frameworks y la utilización de JavaScript para la creación de experiencias interactivas. En un mercado laboral cada vez más digitalizado, donde la demanda de habilidades en desarrollo web continúa en ascenso, este curso se presenta como una respuesta a la necesidad de formar profesionales capaces de diseñar, implementar y mantener interfaces de usuario modernas, accesibles y eficientes.

## Contenidos mínimos:

Los estudiantes aprenderán a:

- Desarrollar estructuras HTML semánticas y accesibles.
- Aplicar estilos CSS para crear diseños responsivos y atractivos.
- Utilizar frameworks como Bootstrap para acelerar el desarrollo.
- Implementar control de versiones con Git.
- Programar funcionalidades interactivas usando JavaScript.



- Consumir APIs REST y manejar la asincronía en JavaScript.
- Integrar todos los conocimientos en un proyecto final que combine HTML, CSS y JavaScript.

### Objetivo General:

Capacitar a los estudiantes para desarrollar interfaces web funcionales y estéticamente atractivas, aplicando buenas prácticas de accesibilidad y utilizando las herramientas y tecnologías más relevantes en el desarrollo frontend.

### Objetivos específicos:

Introducir a los estudiantes en los conceptos fundamentales de HTML y CSS.

Enseñar a los estudiantes cómo estructurar y diseñar páginas web semánticas y accesibles.

Capacitar en el uso de frameworks como Bootstrap y en la gestión de versiones con Git.

Proveer conocimientos básicos y avanzados de JavaScript, incluyendo la manipulación del DOM y el consumo de APIs.

Integrar todos los conocimientos adquiridos en un proyecto final que combine las tecnologías aprendidas.



## Resultados de aprendizaje:

Al finalizar el curso, los estudiantes serán capaces de:

- Crear páginas web estáticas utilizando HTML, aplicando una estructura semántica adecuada.
- Estilizar páginas web utilizando CSS, asegurando un diseño responsive y accesible.
- Desarrollar interfaces de usuario utilizando Bootstrap para una rápida implementación de componentes.
- Gestionar proyectos de desarrollo web utilizando Git para el control de versiones.
- Implementar lógica de negocio en el frontend utilizando JavaScript, incluyendo la manipulación del DOM y la integración con APIs REST.
- Desarrollar y presentar un proyecto final integrador que cumpla con los estándares modernos de accesibilidad y diseño web.

## Distribución de unidades por módulo:

Organiza el contenido en módulos manejables, facilitando la planificación y la entrega del curso de manera estructurada y coherente.

### Módulo 1 - HTML:

En este módulo, los estudiantes se introducirán en el mundo del desarrollo web comenzando con HTML, la estructura básica de cualquier sitio web. Se abordarán conceptos fundamentales como la creación de listas, rutas, inclusión de multimedia y tablas, además de formularios. El módulo concluirá con la publicación de una página estática en un servidor gratuito, utilizando herramientas como Netlify o GitHub Pages.

## Clase 1: Presentación y Conceptos Básicos de HTML

- Sobre Talento Tech
- Objetivo del curso y lineamientos de cursada
- Conceptos básicos sobre Full Stack
- Herramientas a utilizar
- Instalación del software
- Explicación del proyecto final
- Introducción a HTML: Etiquetas básicas y atributos
- Etiquetas semánticas más comunes: `<header>`, `<nav>`, `<main>`, `<footer>`, `<section>`, `<article>`, `<aside>`
- Cómo dividir una página HTML con las etiquetas semánticas
- Etiquetas básicas más comunes: `<h1>`, `<h2>`, `<p>`, `<b>`, `<strong>`, `<i>`
- Ejercicios

## Clase 2: HTML 2 - Listas, Rutas, Multimedia y Tablas

- Listas y enlaces
- Rutas absolutas y relativas
- Elementos en bloque y en línea
- Multimedia con HTML: imágenes, video, audio, iframes
- Accesibilidad en Multimedia (etiquetas `alt`, subtítulos en video)
- Tablas
- Ejercicios
- Cuestionario

## Clase 3: HTML 3 - Formularios y Subida al Servidor

- Formularios
- Formspree
- Herramienta de inspección
- Subida a un hosting gratuito (Netlify o GitHub Pages)
- Ejercicios

## Módulo 2 - CSS:

El segundo módulo se centra en la estilización de las páginas web utilizando CSS. Los estudiantes aprenderán desde las bases de CSS, explorando unidades de medida, colores, fondos, y tipografías, hasta temas más avanzados como el modelo de caja, posicionamiento, animaciones, y el uso de sistemas de diseño como Flexbox y Grid para crear layouts responsivos y estructurados.

### Clase 4: CSS 1 - Introducción a CSS

- Bases del CSS
- CSS externo, interno y en línea
- Selectores básicos (id, clase, etiqueta, universal)
- Especificidad, Herencia, Cascada y Orden de las reglas en CSS
- Ejercicios
- Cuestionario
- Ruta de Avance

### Clase 5: CSS 2 - Medidas, Colores, Fondos, Fuentes e Íconos

- Unidades de medida
- Colores CSS
- Fondos en CSS
- Fuentes y tipografías
- Estilos para textos y listas
- Íconos
- Ejercicios

### Clase 6: CSS 3 - Modelo de Caja, Posicionamiento y Flexbox

- Modelo de caja y propiedades
- Posicionamiento y visualización



- Selectores avanzados
- ¿Qué es Flexbox?
- Propiedades del contenedor Flex y los Flex items
- Ejercicios
- Cuestionarios

### Clase 7: CSS 4 - Grid y Media Queries

- ¿Qué es Grid?
- Implementación de Grid
- Maquetado con Flex y Grid
- Media Queries
- Ejercicios

### Clase 8: Bootstrap & Git

- ¿Qué es un framework?
- Componentes básicos de Bootstrap
- Cómo agregar un componente a nuestro proyecto
- Git: Descarga de Git
- Crear un repositorio externo (GitHub)
- Comandos básicos (Init, commit, push)
- Utilización de Git en todas las clases siguientes
- Ejercicios
- Cuestionario
- Ruta de Avance

## Módulo 3 - JavaScript:

En este módulo, los estudiantes aprenderán a añadir interactividad a las páginas web utilizando JavaScript. Se abordarán temas como la sintaxis básica, control de flujo mediante condicionales y ciclos, programación modular con funciones, manipulación del DOM, gestión

de eventos y manejo de la asincronía. Se culminará con la integración de JavaScript en el proyecto final.

### **Clase 9: JS 1 - Introducción a JavaScript**

- ¿Qué es y para qué se usa JavaScript?
- Conceptos generales. Sintaxis básica
- Variable: ¿qué es y cómo declararla? Tipos
- Asignación y cambio del valor
- Operadores aritméticos
- Conversión a entero y flotante
- **Ejercicio**

### **Clase 10: JS 2 - Condicionales y Ciclos**

- Diagrama de flujo
- Condicional: ¿Qué es?
- Operadores lógicos y de comparación: ¿Qué son y cuál es su uso en los condicionales?
- Bucles: ¿Qué son? Tipos y diferencias entre sí
- Cómo combinar operadores lógicos y ciclos
- **Ejercicio**
- **Cuestionario**

### **Clase 11: JS 3 - Programación Modular con Funciones**

- Funciones: ¿Qué son? Parámetros de entrada y de salida
- Scope global y local
- Programación modular vs. Funciones
- Ejercitación de funciones
- Parámetros.
- Funciones nativas.



- Ejercicio

### Clase 12: JS 5 - DOM y Eventos

- Manipulación del DOM
- Definición, alcance y su importancia para operar sobre elementos HTML
- Eventos en JS
- Eventos: ¿Qué son, para qué sirven y cuáles son los más comunes?
- Escuchar un evento sobre el DOM
- Ejercicios.
- Cuestionario.
- Ruta de Avance

### Clase 13: JS 6 - LocalStorage, SessionStorage y Carrito de Compras

- Introducción a LocalStorage y SessionStorage
- Diferencias entre LocalStorage y SessionStorage
- Implementación de un carrito de compras utilizando LocalStorage o SessionStorage
- Ejercicios.

### Clase 14: JS 7 - Asincronía

- Asincronía
- Consumo de API REST a través de `fetch`
- Procesamiento de los datos
- Incluir los datos consumidos y procesados por medio de `fetch` en nuestro proyecto
- Ejercicios.
- Cuestionario.

### Clase 15: API y Procesamiento de Datos

- Desarrollo de un proyecto integrador que combine HTML, CSS, y JavaScript





- Consumo de API REST
- Incorporación y procesamiento de los datos en nuestro HTML
- Incorporación de buenas prácticas de accesibilidad y SEO
- Presentación del proyecto final
- Ejercicios.

### **Clase 16: Revisión Final y Despedida**

- Presentación y revisión de los proyectos finales
- Feedback personalizado a cada proyecto
- Comparación con proyectos profesionales
- Despedida y recomendaciones para el futuro
- Ejercicios.
- Cuestionario.
- Ruta de Avance

### **Proyecto Final Integrador:**

Al finalizar el curso, cada estudiante habrá creado una aplicación web completa que integrará HTML, CSS y JavaScript, incluyendo la implementación de una API REST para el consumo de datos dinámicos. Además, se aplicarán buenas prácticas de accesibilidad y SEO para asegurar una experiencia de usuario optimizada y alineada con los estándares actuales de la web.



### Recursos Necesarios:

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

### Requisitos para la acreditación:

- 70% de asistencia.
- Realización de todos los cuestionarios obligatorios ubicados en el campus virtual.
- Aprobación del proyecto final integrador.

