

# Temario

CLASE 01

**Presentación y  
Conceptos  
Básicos de  
HTML**



CLASE 02

**HTML 2 - Listas,  
Rutas,  
Multimedia y  
Tablas**



CLASE 03

**HTML 3 -  
Formularios y  
Subida al  
Servidor**



CLASE 04

**CSS 1 -  
Introducción  
a CSS**

CLASE 05

**CSS 2 - Medidas,  
Colores, Fondos,  
Fuentes e Íconos**



CLASE 06

**CSS 3 - Modelo  
de Caja,  
Posicionamiento  
y Flexbox**



CLASE 07

**CSS 4 - Grid y  
Media Queries**



CLASE 08

**Git & Github**

CLASE 09

**JS 1 - Introduc-  
ción a  
JavaScript**



CLASE 10

**JS 2 -  
Condicionales  
y Ciclos**



CLASE 11

**JS 3 -  
Programación  
Modular con  
Funciones**



CLASE 12

**JS 5 - DOM y  
Eventos**

CLASE 13

**JS 6 -  
LocalStorage,  
SessionStorage  
y Carrito de  
Compras**



CLASE 14

**JS 7 -  
Asincronía**



CLASE 15

**API y  
Procesamiento  
de Datos**



CLASE 16

**Revisión Final y  
Despedida**

# Distribución de Unidades por Módulo

## 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 01 >> 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 02 >> 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 03** >> **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 04** >> **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 05** >> **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 06** >> **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 07** >> **CSS 4 - Grid y Media Queries**

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

## **CLASE 08** >> **Git & Github**

- Git: Descarga de Git
- Crear un repositorio externo (GitHub)
- Comandos básicos (Init, commit, push)
- Utilización de Git en todas las clases siguientes
- Ejercicios
- Cuestionario
- Pre Entrega de Proyecto

## **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 09** >> **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.
- Proyecto Final

## 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.
- Entrega de Proyecto Final

### Proyecto Final Integrador:

Al finalizar el curso, cada estudiante habrá desarrollado un sitio web completo que integrará HTML, CSS y JavaScript, logrando una estructura sólida y un diseño visual atractivo. Este sitio incluirá la implementación de una **API REST** para consumir datos dinámicos en tiempo real, permitiendo que el contenido sea interactivo y se actualice en función de la información externa. Además, los estudiantes aplicarán **buenas prácticas de accesibilidad** para asegurar que el sitio sea utilizable para todas las personas, independientemente de sus capacidades, y **estrategias de SEO** (optimización para motores de búsqueda) para mejorar su visibilidad y posicionamiento en línea. De esta forma, el proyecto final estará alineado con los estándares modernos de desarrollo web, asegurando una **experiencia de usuario optimizada** y brindando a los estudiantes una base sólida para proyectos futuros en el mundo digital.