



«Talento Tech»

# React 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

React es fundamental para los desarrolladores Frontend hoy en día debido a su amplia adopción en la industria y su capacidad para crear interfaces modernas y dinámicas. Su ecosistema maduro, con herramientas como Hooks y Context API, permite un desarrollo eficiente y escalable. Además, su flexibilidad para integrarse con otras tecnologías y su enfoque en la experiencia del usuario lo convierten en una opción clave para aplicaciones web de cualquier tamaño. Aprender React no solo mejora las habilidades técnicas, sino que también aumenta la empleabilidad, ya que muchas empresas lo demandan para sus proyectos.

## Contenidos mínimos

- ▶ **Introducción a React y configuración del proyecto:** Instalación con Vite y Node.js. JSX y creación de componentes funcionales.
- ▶ **Gestión del Estado:** useState para manejar el estado local. Manejo de eventos (clics, formularios).
- ▶ **Ciclo de vida con useEffect:** Realizar peticiones a APIs y manejar efectos secundarios.
- ▶ **React Router:** Configuración de rutas, navegación y rutas protegidas.

# Contenidos mínimos

- ▶ **Context API:** Crear y consumir contexto para el manejo del estado global (ej. carrito, autenticación).
- ▶ **CRUD Básico de Productos:** Crear, leer, actualizar y eliminar productos. Formularios con validación y manejo de errores.
- ▶ **Autenticación de Usuarios:** Formulario de login, validación y rutas protegidas.
- ▶ **Estilización:** Uso de librerías de UI (Bootstrap, styled-components).

## Objetivo General

El curso tiene como objetivo proporcionar a los estudiantes los conocimientos y habilidades necesarios para desarrollar aplicaciones web interactivas utilizando ReactJS. Al finalizar el curso, los estudiantes serán capaces de crear un proyecto funcional, como un e-commerce, que integre la gestión de estado, autenticación de usuarios, CRUD de productos, y rutas protegidas.

## Objetivos específicos:

- Familiarizar a los estudiantes con los fundamentos de ReactJS y su ecosistema.
- Desarrollar habilidades para manejar el estado dentro de componentes y globalmente con Context API.
- Capacitar a los estudiantes para crear y gestionar rutas dinámicas y protegidas.
- Enseñar la integración de formularios con validación y la implementación de un sistema de autenticación básico.
- Guiar en la construcción de un proyecto final de e-commerce utilizando conceptos como CRUD y manejo de estados complejos.
- Introducir buenas prácticas de estilización y diseño para crear aplicaciones web visualmente atractivas y funcionales.

# Resultados de aprendizaje

Al finalizar este curso, los estudiantes serán capaces de:

1. Crear una aplicación ReactJS desde cero utilizando Vite.
2. Implementar la gestión de estado utilizando useState y Context API.
3. Desarrollar rutas dinámicas y protegidas para una navegación fluida y segura.
4. Construir formularios con validación y manejar la autenticación de usuarios.
5. Desarrollar un CRUD básico para gestionar productos dentro de un proyecto de e-commerce.
6. Estilizar una aplicación utilizando librerías como Bootstrap y styled-components.
7. Desplegar la aplicación final en una plataforma de hosting.

# Temario

CLASE 01

**Introducción a React**



CLASE 02

**JSX y Componentes**



CLASE 03

**Estructura de Componentes y Navegación Básica**



CLASE 04

**Introducción a useState**

CLASE 05

**Uso de useEffect y Peticiones a APIs**



CLASE 06

**React Router - Rutas Estáticas y Dinámicas**



CLASE 07

**Rutas Dinámicas y Protegidas**



CLASE 08

**Introducción a Context API**

CLASE 09

**Autenticación de Usuarios**



CLASE 10

**Creación de Productos - Formulario y Validación**



CLASE 11

**Leer, Actualizar y Eliminar Productos**



CLASE 12

**Implementación del CRUD Completo**

CLASE 13

**Estilización con Bootstrap o styled-components**



CLASE 14

**Diseño Responsivo y UX**



CLASE 15

**Despliegue de la Aplicación**



CLASE 16

**Revisión Final y Presentación del Proyecto**

# Temario

## CLASE 01 >> Introducción a React

- Instalación con Vite y Node.js.
- Estructura básica del proyecto React.
- Creación de componentes funcionales.

## CLASE 02 >> JSX y Componentes

- Explicación de JSX.
- Creación de componentes más complejos.
- Propiedades de los componentes (props).

## CLASE 03 >> Estructura de Componentes y Navegación Básica

- Creación de la estructura básica de la aplicación.
- Primeros componentes reutilizables (Header, Footer).
- Visualización de los componentes en el navegador.

### Contribución al Proyecto Final:

Creación de la estructura inicial de la aplicación y de los primeros componentes reutilizables.

## CLASE 04 >> Introducción a useState

- Gestión del estado local con useState.
- Manejo de eventos (clics, formularios).
- Primer ejercicio práctico: agregar productos al carrito.

# Temario

## CLASE 05 >> Uso de useEffect y Peticiones a APIs

- Introducción a useEffect para efectos secundarios.
- Realización de peticiones a APIs (cargar productos).
- Manejo de estado de carga y errores.

### Contribución al Proyecto Final:

Implementación del carrito de compras básico y carga de productos.

## CLASE 06 >> React Router - Rutas Estáticas y Dinámicas

- Instalación y configuración de React Router.
- Creación de rutas estáticas (inicio, lista de productos).
- Uso de [Link](#) para navegar entre componentes.

## CLASE 07 >> Rutas Dinámicas y Protegidas

- Creación de rutas dinámicas (detalle de productos).
- Implementación de rutas protegidas (carrito, administración de productos).
- Redirección de usuarios no autenticados.

### Contribución al Proyecto Final:

Desarrollo de la navegación entre la lista de productos, detalle de productos y el carrito de compras.

Implementación de rutas protegidas para el carrito y la página de administración de productos.



# Temario

## CLASE 08 >> Introducción a Context API

- Creación de Context API para manejo de estado global.
- Uso de `useContext` para compartir datos entre componentes.
- Implementación del estado global para el carrito.

## CLASE 09 >> Autenticación de Usuarios

- Implementación de formulario de login.
- Manejo de autenticación con tokens (simulada).
- Protección de rutas usando Context API para la autenticación.

### Contribución al Proyecto Final:

Implementación del formulario de login y la gestión del estado de autenticación con Context API.

Protección de rutas que solo los usuarios autenticados puedan acceder (carrito y administración).

## CLASE 10 >> Creación de Productos - Formulario y Validación

- Creación de formulario para agregar productos.
- Validación de datos de formularios.
- Implementación de la funcionalidad para agregar productos.

## CLASE 11 >> Leer, Actualizar y Eliminar Productos

- Implementación de la funcionalidad para leer productos desde el estado o API.
- Actualización y eliminación de productos.
- Validación de los formularios de edición.

# Temario

## CLASE 12 >> Implementación del CRUD Completo

- Conectar el formulario de agregar/editar productos con el estado global.
- Validación de formularios y manejo de errores.

### Contribución al Proyecto Final:

Desarrollo de la funcionalidad de CRUD para administrar productos en la tienda.

Implementación de una interfaz administrativa para agregar y editar productos.

## CLASE 13 >> Estilización con Bootstrap o styled-components

- Introducción a Bootstrap o styled-components para estilizar componentes.
- Creación de un diseño básico y responsive.
- Aplicación de estilos en componentes (botones, formularios, productos).

## CLASE 14 >> Diseño Responsivo y UX

- Diseño mobile-first y adaptabilidad de la aplicación.
- Mejores prácticas de diseño UI/UX.
- Revisión de la aplicación para mejorar la experiencia de usuario.

### Contribución al Proyecto Final:

Aplicación de estilos y diseño visual a la tienda online y formularios de autenticación.

Mejora en la experiencia de usuario (UX) y diseño adaptativo.

# Temario

## CLASE 15 >> Despliegue de la Aplicación

- Despliegue de la aplicación en plataformas como Vercel o Netlify.
- Pruebas de funcionamiento en el entorno de producción.
- Revisión de los pasos de despliegue y configuración.

## CLASE 16 >> Revisión Final y Presentación del Proyecto

- Revisión del código y la funcionalidad.
- Solución de posibles errores y optimización.
- Presentación final del proyecto de e-commerce.
- Evaluación de los resultados y feedback del instructor.

### Contribución al Proyecto Final:

Despliegue de la aplicación final.

Revisión de la integración de todas las funcionalidades desarrolladas (CRUD, autenticación, carrito, rutas protegidas).

## Proyecto Final Integrador:

El objetivo de este proyecto final es que los estudiantes apliquen todos los conceptos aprendidos durante el curso de ReactJS en la creación de una aplicación de e-commerce básica. El proyecto debe integrar funcionalidades clave como autenticación de usuarios, manejo de productos mediante un CRUD (Crear, Leer, Actualizar, Eliminar), y un carrito de compras

# 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.
- Aprobación del proyecto final integrador.

## Bibliografía de Referencia

### **Awesome React:**

<https://github.com/brillout/awesome-react-components>

### **Frontend Masters React Course:**

<https://frontendmasters.com/learn/react/>

### **React Docs:**

<https://react.dev/>



**Buenos Aires**  
*aprende*

Agencia de Habilidades para el Futuro

