

Programa de **Estudio** /

Especialización Front End



**Certified Tech
Developer**

The Ultimate Degree

from



Globant ▶

by

DIGITALHOUSE

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
# An Array of C  
attr_reader :ra  
# The control M  
attr_reader :c  
# An Experiment  
attr_reader :ex  
# An Array of O  
attr_reader :ob  
# An Array of O  
attr_reader :ob  
# An Array of O  
attr_reader :ob  
# An Array of O  
attr_reader :ob  
# Internal: Cre  
#  
# experiment  
# observations:  
# control:  
#  
def initialize(  
  @experiment  
  @observations
```

¿POR QUÉ **ESTUDIAR** **FRONT END?**

- > **Esta especialización te prepara para abordar los desafíos actuales y futuros del desarrollo front-end**, ofreciéndote un conjunto de habilidades altamente solicitadas en la industria. Te coloca en una posición competitiva para aprovechar oportunidades laborales y de crecimiento y contribuir de manera significativa al desarrollo de aplicaciones web de calidad.



>>>> Objetivos de aprendizaje

Eficiencia en el desarrollo

Las tecnologías como React y Redux permiten una creación de interfaces de usuario más eficiente y modular, lo que facilita la colaboración en equipos de desarrollo y agiliza el proceso de desarrollo.

Tipado seguro con TypeScript

TypeScript ofrece tipado estático que reduce errores en el código y mejora la calidad del software. Las empresas aprecian desarrolladores que pueden escribir código más seguro y menos propenso a errores.

Desarrollo de experiencias de usuario mejoradas

Las herramientas y patrones de diseño enseñados te permiten crear interfaces de usuario atractivas y funcionales que mejoran la experiencia del usuario, lo que es un aspecto esencial para cualquier empresa centrada en el cliente.

Experiencia en manejo de estados

Uno de los desafíos clave en el desarrollo front-end es la gestión efectiva de estados. La especialización te dota con experiencia en manejo de estados a través de Redux, lo que es altamente valorado en aplicaciones complejas.

Amplia aplicabilidad

Las habilidades adquiridas en esta especialización no se limitan solo al front-end. GraphQL y las prácticas de diseño también son relevantes en otros contextos, como el desarrollo back-end y la arquitectura de sistemas.

Requerimientos académicos

- > Conocimientos básicos de **React y su implementación con hooks**
- > Comprensión sólida de **HTML, JavaScript y CSS**
- > Obtención de datos de una **API**

¿A quién va dirigido?

- > Egresados de primer track de CTD.
- > Frontend/Backend/Full Stack Dev Jr.
- > Analista de QA Jr.
- > Analista de Bases de Datos Jr.
- > Analista de Infraestructura Jr.



*
Modalidad
On-demand
a tu ritmo.



> **Contenido 100% a demanda.**

Cursos a tu ritmo, cuando y donde quieras

> **Rutas de aprendizaje**

de 12 cursos.

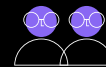
> Evaluaciones dentro
de Playground

> Una comunidad en discord
disponible 24/7



> Contenido hecho
por expertos/as

> Oportunidades de
aprendizaje colaborativo.



> **Asistencia académica** en canales de preguntas y ayuda por materia.



Ejes de Contenido

01

Cursos de especialización

- * React + Redux con Typescript
- * Patrones de diseño + testing
- * Next.js

02

Cursos complementarios

- * Bases de datos escalables y no relacionales
- * Infraestructura avanzada
- * Testing automatizado

03

Cursos de Contenidos soft

- * Gestión de productos digitales
- * Gestión de proyectos escalables
- * Data analytics
- * Problem solving & project management
- * Time management

04

Desafío profesional



React + Redux con TypeScript

- Fortalecer el uso de Hooks básicos y avanzados.
- Comprender las ventajas de usar elementos de tipado para anticipar y reducir errores en el código e implementarlos en un componente de React.
- Aprender herramientas para la obtención y manipulación de información/datos provenientes de un servidor API.
- Profundizar en las distintas herramientas de manejo de estados globales de una aplicación en React, identificando las diferencias existentes entre cada una de ellas.
- Conocer diferentes tipos de manejadores de paquetes para un proyecto.

Hooks | JS(x) Docs | Prop-types | ReactQuery
Typescript | Redux

Patrones de diseño + testing

- Identificar diferentes patrones de diseño para el desarrollo de aplicaciones web -reconociendo las fortalezas y debilidades de cada uno- y construir criterios para su aplicación en proyectos diversos.
- Revisar conceptos relacionados al uso de buenas prácticas al momento de escribir código y su aplicación en el desarrollo de proyectos futuros.
- Entender la importancia de la accesibilidad en nuestra aplicación, siguiendo los principios estandarizados de WCAG.
- Introducir GraphQL como lenguaje y Apollo como herramienta alternativa al uso de API REST, comprendiendo ventajas y desventajas de uso.
- Fortalecer las habilidades en el uso de estilos para componentes. Introducir Styled components, identificando ventajas y desventajas de uso.
- Abarcar conceptos sobre el testing con foco en React, introduciendo nuevas herramientas para tal fin. Fomentar la metodología TDD (Test Driven Development) como práctica recomendada al momento de escribir código.

Patrones avanzados | SOLID | CSS in JS |
Jest, Mock | React-Testing Library |
GraphQL - Apollo

Next.js

- Reforzar habilidades de programación, explorando nuevas herramientas y recursos que permitan mejorar la calidad de desarrollos y aplicaciones.
- Introducir Next.js como herramienta para crear aplicaciones con React, comprendiendo sus posibilidades y las ventajas de su uso.
- Avanzar en la creación de formularios simples y avanzados por medio de hooks de React. Se presenta la librería React Hook Form y se introduce la administración de validaciones customizadas y su testeo lógico
- Fortalecer habilidades en el diseño de la UI y el uso de estilos para componentes. Introducir la librería Material UI, identificando ventajas y desventajas de uso.
- Abarcar conceptos sobre el testing con foco en Javascript y React, y las posibilidades que nos brindan para el testing End to End en aplicaciones web.

NextJS | SSSR vs SSG vs SPA |
Data Fetching | Serverless |
React Hook Form | Testing con Cypress

Bases de datos escalables y NoSQL

- > Desarrollar el pensamiento analítico en la gestión de datos y comprender la trayectoria entre el requerimiento y su implementación física.

Configurar una base de datos relacional y no relacional para que obtenga su mayor performance en grandes volúmenes de datos.
- > Acceder a la información con el lenguaje propio de la herramienta, de manera tal que los resultados obtenidos sean confiables y seguros.
- > Lograr la optimización del uso de los datos, de forma que su volumen no sea un impedimento en los tiempos de proceso.
- > Reconocer la forma correcta del tratamiento de la información, a fin de no entorpecer el acceso de otros usuarios en forma simultánea.
- > Introducirse en el mundo de Big Data, Data Warehouse, sus conceptos y aplicaciones.

Procedimientos | Funciones | Cursores |
Transacciones | Triggers MongoDB

Infraestructura avanzada

- > Comprender el enfoque de Microservicios en el rol de las aplicaciones modernas.
- > Conocer los fundamentos de la orquestación de contenedores a través de la visualización de servicios desplegados y funcionales.
- > Conocer los elementos que forman parte de Kubernetes y como realizamos despliegues de nuestros servicios.
- > Adentrarnos en la práctica de GitOps.
Microservicios | Kubernetes |
GitOps | ArgoCD

Testing automatizado

- > Conocer e implementar procesos de calidad dentro de un proceso de desarrollo de software.
- > Reconocer casos de pruebas automatizables.
- > Implementar un proyecto de automatización con Selenium y JAVA
- > Implementar un proyecto de automatización con RestAssured y JAVA.
- > Definición de procesos de calidad en los distintos momentos de desarrollo tales como plan de pruebas, plan para la liberación de nuevas versiones, mantenimiento de la prueba, procesos de revisión, almacenamiento de pruebas, métricas y reportes.
- > Principios para implementar testing en dispositivos móviles.
Page Object Model | Selenium con Java |
Suites | RestAssured | Jenkins |
Master Test Plan

Gestión de productos digitales

- Conocer de qué manera se crea, produce y gestiona un producto digital con una perspectiva del desarrollo de software.
- Utilizar los conocimientos adquiridos en los talleres de Metodologías del Trabajo, Design Thinking y UX/UI para concebir un producto digital desde la idea hasta el lanzamiento.
- Tener contacto con storytellings que remitan a casos reales para aplicarlos en situaciones de trabajo del campo profesional.

Product Discovery | Agile Inception |
Lean Startup | MVP | Planificación Ágil |
Product Backlog

Gestión de proyectos escalables

- Profundizar el conocimiento de la gestión de productos digitales, con un abordaje de escalonamiento de las metodologías ágiles en la organización.
- Conocer las buenas prácticas del mercado y analizar modelos de las mismas para su aplicación en el campo profesional.
- Observar la importancia de la gestión del conocimiento en entornos empresariales para conservar el conocimiento organizacional.

Research | Business Case | Jira |
Nueva Versión Producto |
Gestión del Conocimiento

Data Analytics

- Conocer la importancia de las estructuras de datos para el análisis y la toma de decisiones en un producto digital.
- Aplicar las etapas de un proceso analítico: manipulación, modelado y visualización.
- Manejar herramientas necesarias para el análisis de datos.

Indicadores KPI | OKRs | SMART |
Manipulación Datos | Modelado Datos |
Visualización Datos

03 Cursos soft skills

Programa de Estudio /

Problem solving & project management

- > Investigar los diferentes tipos de problemas.
- > Construir un proceso de resolución más efectivo.
- > Identificar el estilo de decisión y su correlación con la aversión al riesgo y la incertidumbre.
- > Conocer y aplicar una estructura metodológica y secuencial para gestionar proyectos, aumentando las posibilidades de éxito en términos de tiempo, forma y calidad.
- > Generación de un plan de desarrollo individual.

Time Management

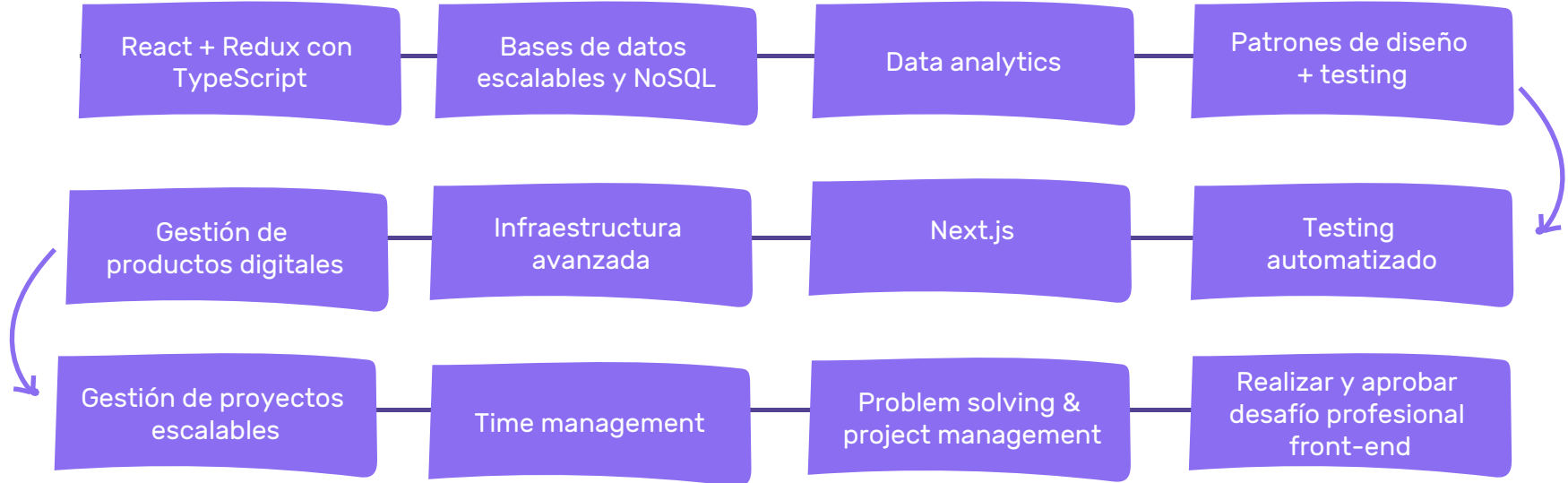
- > Redefinición del concepto de tiempo
- > Adquirir herramientas puntuales de gestión del tiempo
- > Reconocer los diferentes estilos de gestión

04

Desafío profesional

Ruta de aprendizaje sugerida

A continuación te dejamos la ruta de aprendizaje ideal. Recuerda que puedes cursar a tu propio ritmo y con tus propios horarios.



Criterios de **aprobación** y **certificación**

De **especialización**

- React + Redux con TypeScript
- Patrones de diseño + testing
- Next.js
- Bases de datos escalables y NoSQL
- Infraestructura avanzada
- Testing automatizado
- Gestión de productos digitales
- Gestión de proyectos escalables
- Data analytics
- Problem solving & project management
- Time management
- Realizar y aprobar desafío profesional front-end

Individual por contenido

- Haber recorrido todo el contenido en nuestra plataforma de aprendizaje
- Haber aprobado los cuestionarios de cada módulo de aprendizaje
- Haber aprobado el cuestionario final del curso
[La aprobación de los cuestionarios antes mencionados es del 75% de las preguntas]

Individual para desafío profesional

- Desarrollar y entregar vía repositorio un proyecto completo con los requerimientos solicitados en el mismo (sea front end o back end)
- Resultar aprobado post corrección individual en primera o segunda instancia



¿Todo listo para empezar esta **aventura tecnológica**?

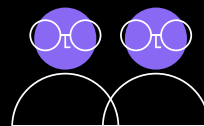
Comienza ahora y accede a una experiencia exclusiva



Expertos en línea



Aprendizaje sin
horarios y reglas



Una comunidad
disponible 24/7 para ti



Acceso a los contenidos
durante 12 meses.