

# OCTAVIO QUINTERO

## FULL STACK DEVELOPER



Actualmente, me estoy desarrollando como programador **Full Stack**, donde implemento mi capacidad para crear **aplicaciones** completas, desde la conceptualización hasta la ejecución. Encontrar soluciones a posibles problemas tanto en el **frontend** como en el **backend** me ha permitido comprender el ciclo de vida de una app, convirtiéndome en un desarrollador **versátil**.

- **Linkedin** – <https://www.linkedin.com/in/octavio-quintero/>
- **GitHub** – <https://github.com/octa-quintero>
- **Portafolio** - <https://octavioquintero.vercel.app/>
- **Tel** - +54 3585106603
- **Nacionalidad** - Argentina, *Cordoba, Rio Cuarto*

---

### Formación

#### ❖ **SoyHenry BootCamp 800 hs** ( Concluido diciembre **2022** )

FullStack REST APIs - Object Oriented Programming (OOP) – Teamwork - PostgreSQL - SQL - JavaScript - Node.js - React.js - Sequelize - Express - Redux - SCSS - HTML5 - CSS – Git

#### ❖ **CoderHouse** ( Concluido noviembre **2022** )

Desarrollo y estructura Web - SCSS - HTML5 - CSS - GitHub – Bootstrap

#### ❖ **Elementos Desarrollo FullStack Fundacion CarlosSlim 453hs Diplomatura** (Concluido Abril **2024** )

Git – HTML5 - SCSS - React.js - Bootstrap y Materialize - Node.js - Express.js - NPM – PostgreSQL – Nginx – Jest – Canva – Postman – PostgreSQL – SQL – Express – Redux – ReduxToolkit – JavaScript - Jira

#### ❖ **Desarrollador de React.JS Fundacion Carlos Slim 96 hs** (Concluido enero **2024**)

Aptitudes - Desarrollo y estructura Web - SCSS - HTML5 - CSS - GitHub – Bootstrap

#### ❖ **Desarrollador de Node.JS Fundacion Carlos Slim 76hs** (Concluido febrero **2024**)

Desarrollo de Servidores - NPM - Express.js - Persistencia de Datos - Autenticación y Autorización - Testing – Jest

#### ❖ **Administrador de servidores Carlos Slim 100hs** (Concluido febrero **2024**)

Sistemas Operativos de Servidores - Gestión de Redes - PostgreSQL - Nginx – PowerShell - Bash - Testing – Jest - Gestión de Usuarios y Permisos

#### ❖ **Desarrollador Front-end Carlos Slim 35hs** (Concluido febrero **2024**)

Responsive Design - NPM - ECMAScript 6- API Rest - Visual Studio Code- jQuery – Materialize – Git – GitHub – Yarn

#### ❖ **Diseñador de experiencia de usuario (UX) 32hs** (Concluido marzo **2024**)

Git - GitHub- Pruebas de usabilidad - Jira - Autenticación y Autorización – Testing

#### ❖ **Desarrollo de sitios Web Responsive 62hs** (Concluido marzo **2024**)

Media queries – CSS - SASS- Inspectores de navegador - BrowserStack

#### ❖ **Control de versiones 17hs** (Concluido marzo **2024**)

Git – GitHub

#### ❖ **Desarrollo de sitios Web Responsive 62hs** ( Concluido marzo **2024** )

Media queries – CSS - SASS- Inspectores de navegador - BrowserStack  
Figma - Diagramas de flujo - Postman.

#### ❖ **TechWordl 13 hs** (Concluido noviembre 2023)

Gestión de aplicaciones con Docker en diferentes entornos.

#### ❖ **Ingles Tecnico Lingo Insider** (En desarrollo actualmente)

## - Talent Place (Backend Developer)

### ❖ Desarrollo

El objetivo principal fue crear una plataforma para empresas, startups y desarrolladores IT juniors. Esta enfocada en ofrecer una solución facilitando la interacción tanto para los juniors como para las empresas, proporcionando un acceso sencillo a los nuevos talentos.

### ❖ Tecnologías Aplicadas

Mi participación fue en el equipo de **Backend**, donde desarrollé una **API RESTful** usando **Node.js**, **Express** y **TypeScript**. La elección de **TypeScript** nos permitió escribir código más limpio y tener un mejor control de tipos, mejorando así su calidad y facilidad de mantenimiento. La arquitectura de la API se basó en **microservicios**, por lo que permitió una mayor escalabilidad y flexibilidad en el desarrollo. Usamos **TypeORM** para interactuar con la base de datos y ayudar a crear un sistema de gestión para esquemas de bases de datos de entidades. Para la DB, optamos por **PostgreSQL** para manejar las consultas en la base de datos. Usamos **Docker** para el despliegue, lo que nos permitió empaquetar nuestras aplicaciones y dependencias en contenedores aislados.

## - Pixel Gaming (Fullstack Developer)

### ❖ Desarrollo

Pixel Gaming es un proyecto que desarrollé pensando en el **usuario** gamer. Con una **interfaz intuitiva** para que los usuarios amantes de los videojuegos puedan navegar, buscar, catalogar y administrar fácilmente sus colecciones de una manera simple. Agregue perfiles para los usuarios para que puedan tener su propia biblioteca de juegos personalizada y también compartir reseñas y calificaciones. También agregue un método de pago para que pueda elegir entre diferentes membresías que permitan el acceso a diversas categorías.

### ❖ Tecnologías Aplicadas

La aplicación fue realizada **JavaScript**, **React**, **Node.js** y **Express**. También, usa una base creada con **PostgreSQL** y **Sequelize** para la gestión de datos de usuarios y elementos del catálogo. Para el diseño se utilizó NPM, SCSS y Material UI. La interfaz es totalmente responsiva, permitiendo a los usuarios buscar juegos, crear cuentas y gestionar elementos desde cualquier dispositivo. La aplicación también integra tecnologías como **Docker**, **Docker-Compose**, **Redux**, **ReduxToolkit**, **Context API** para mantener el estado del estado global, **JWT** para la autenticación, **Nodemailer** para el envío de correos, **Webpack** para la minimización de recursos.

## - Portfolio Project (FrontEnd Developer)

### ❖ Desarrollo:

Con un enfoque en la **simplicidad**, este portafolio presenta una interfaz minimalista que permite a los visitantes acceder de manera sencilla a secciones como experiencia, proyectos e información general, garantizando una experiencia intuitiva.

### ❖ Tecnologías Aplicadas

En mi aplicación de portafolio, optimicé la organización del proyecto utilizando **React.js**, **JavaScript** y un preprocesador **SCSS**. Además, integré tecnologías como **Material-UI** para facilitar la construcción de componentes de interfaz de usuario estilizados y reutilizables.

Implementé una **estructura modular** que simplifica el acceso al código y mejora la comprensión, permitiendo una gestión eficiente de componentes, estilos y funcionalidades. Esto asegura una experiencia de desarrollo más fluida y una mayor claridad en el proyecto en general.

## - Country APP (Fullstack Developer)

### ❖ Desarrollo

Country APP es una aplicación diseñada para los amantes de los viajes y la exploración. Su objetivo principal es proporcionar a los usuarios una plataforma donde puedan **buscar información** sobre países, descubrir actividades turísticas de cada destino y compartir sus propias recomendaciones de actividades.

### ❖ Tecnologías Aplicadas

Para crear la aplicación, utilicé un conjunto de tecnologías, incluyendo **React**, **JavaScript**, **Node.js**, y **Express**. En el backend, diseñé y desarrollé una arquitectura basada en microservicios, lo que permitió una mayor modularidad, escalabilidad y facilidad de mantenimiento en la aplicación. Cada microservicio se encargó de una función específica, lo que facilitó su desarrollo.