

LA CARRERA / PLAN DE ESTUDIOS

# Software Developer

Los contenidos de la carrera abarcan desde los fundamentos de la programación hasta tecnologías avanzadas como React.js y Next.js, preparando desarrolladores competentes. La formación en metodologías de trabajo y el enfoque intensivo en preparación para el mercado laboral, aseguran que los estudiantes adquieran habilidades prácticas y teóricas indispensables para su éxito profesional.



## Introducción a la programación

NIVEL 0

El nivel inicial de la carrera te dará una base sólida en los conceptos fundamentales de la computación y de JavaScript, uno de los lenguajes más utilizados del mercado.

### 4 CAPÍTULOS / 51 CLASES

---

#### ➤ INTRODUCCIÓN / 7 CLASES

- ▶ Hola dev
- ▶ La industria tech
- ▶ Plan de estudios
- ▶ El perfil
- ▶ Nivel 0
- ▶ Discord
- ▶ Sumate

#### ➤ HOLA MUNDO / 17 CLASES

- ▶ Primer objetivo
- ▶ ¿Qué hacen las computadoras?
- ▶ Software
- ▶ Tus compus

- ▶ ¿Qué es el código?
- ▶ ¿Qué es una app?
- ▶ Entradas y salidas
- ▶ JavaScript: El rey
- ▶ ¿Qué es un intérprete?
- ▶ Instalando Node.js
- ▶ ¿Por qué usamos comandos?
- ▶ La terminal
- ▶ Hola terminal
- ▶ Algunos comandos
- ▶ Carpeta "código"
- ▶ Visual Studio Code
- ▶ Hola mundo

## ➤ DATOS / 23 CLASES

- ▶ Objetivo del capítulo
- ▶ Primeros datos
- ▶ Tus primeros datos
- ▶ Comentarios
- ▶ Tipos de datos
- ▶ Sumando datos
- ▶ Typeof
- ▶ Nombre completo
- ▶ Primitivos
- ▶ Numeros y textos
- ▶ Usando tipos
- ▶ Es undefined?
- ▶ Preguntas
- ▶ Tipo dinámico

- ▶ Template
- ▶ Objetos
- ▶ Objeto terminator
- ▶ Objetos personas
- ▶ Modifica el objeto
- ▶ Arrays
- ▶ Mes actual
- ▶ Carrito de compras
- ▶ Yo, objeto

➤ MI PLAYLIST / 4 CLASES

- ▶ ¡Lo hiciste!
- ▶ Tu primer Gist
- ▶ Consigna
- ▶ Finalmente



## Programación en JavaScript

NIVEL 1

Nos iniciaremos en los conceptos fundamentales de la programación trabajando en equipo para resolver problemas de distinto tipo. Este nivel sienta las bases fundamentales respecto a la computación, la programación orientada a objetos y como pensar los problemas del software.

### 20 CAPÍTULOS / 160 CLASES

---

➤ INTRODUCCIÓN / 10 CLASES

- ▶ Nivel 1
- ▶ Ritmo, constancia y frustración
- ▶ Semana a semana
- ▶ Código desconocido
- ▶ Formateá tu código
- ▶ Encontrar respuestas

- ▶ Encuentra el error
- ▶ Pair programming
- ▶ Se programa en equipo
- ▶ Sumate

## ➤ LÓGICA BÁSICA / 26 CLASES

- ▶ Introducción
- ▶ Operadores
- ▶ Comparador de edades
- ▶ Chequeo de saldo
- ▶ Predicciones
- ▶ Calculadora de descuento
- ▶ Total del carrito
- ▶ Inputs
- ▶ ¿Qué es un argumento?
- ▶ Argumentos
- ▶ Hola yo
- ▶ Inspector de objetos
- ▶ Datos del mes
- ▶ Control de flujo
- ▶ ¿Qué es el control de flujo?
- ▶ If
- ▶ Buscando propiedades
- ▶ Calculadora de descuento
- ▶ Calculadora de envíos
- ▶ Truthy / Falsy
- ▶ ¿Truthy o Falsy?
- ▶ Switch
- ▶ Usando Switch

▶ Ternarios

▶ Casilla de mensajes

▶ Leyendo ternarios

## ➤ BUCLES / 25 CLASES

▶ ¿Para qué sirven los bucles?

▶ While

▶ Detené el while

▶ La cafetería más cercana

▶ Teléfonos dentro de presupuesto

▶ Monto total

▶ ¿Para qué me alcanza?

▶ Usuarios con alquileres

▶ For Of

▶ ¿Cuántos pares hay?

▶ Solo windows

▶ Videos pendientes

▶ Cafeterías cercanas

▶ Usuarios con alquileres

▶ For In

▶ Formulario de envío

▶ Requisitos

▶ Candidatos

▶ Lista de candidatos

▶ Usuarios por video

▶ For

▶ Mazo de cartas

▶ Dividir mazo

▶ Promedios de temperatura

▶ Filtrando propiedades

## ➤ FUNCIONES / 13 CLASES

▶ Introducción

▶ ¿Qué hace esta función?

▶ Partes

▶ Todo en funciones

▶ En resumen

▶ Organizando mi app

▶ Scope

▶ Solo pares

▶ Métodos

▶ Arrow functions

▶ filterByLength (filtrarPorLogitud)

▶ Transformá esta función a una arrow function

▶ Transformando textos

## ➤ ALGORITMOS / 6 CLASES

▶ Introducción

▶ Eficiencia

▶ Búsqueda lineal

▶ Búsqueda binaria

▶ Conclusión

▶ Ordenar el array

## ➤ MÓDULOS / 3 CLASES

▶ Introducción

▶ Ejemplo

▶ Modularizando

## ➤ NODE.JS API / 4 CLASES

▶ ¿Qué es una API?

▶ La API de Node.js

▶ 6 Ejemplos

▶ Leyendo argumentos

## ➤ TESTING / 8 CLASES

▶ Introducción

▶ Test unitario

▶ Prueba y error 🚀

▶ TDD

▶ Tests de integración

▶ Test automáticos

▶ En conclusión

▶ Tu primer test

## ➤ PERSISTENCIA / 2 CLASES

▶ Datos permanentes

▶ Leyendo data del disco

## ➤ DESAFÍO: BUSCADOR DE PELÍCULAS / 3 CLASES

▶ Calculadora

▶ Consigna

▶ The real life

## ➤ DISEÑO DE SOFTWARE / 3 CLASES

▶ Escalar

▶ MVC

▶ Un controller

## ➤ NPM / 5 CLASES

▶ ¿Qué es NPM?

▶ Navega npmjs.com

- ▶ Dependencias
- ▶ Nuestra primera dependencia
- ▶ PNPM

## ▶ TYPESCRIPT / 13 CLASES

- ▶ Introducción a Typescript
- ▶ Chequea el sitio de Typescript
- ▶ Instalando typescript
- ▶ Instalando typescript
- ▶ tsx
- ▶ tsc - Typescript Compiler
- ▶ Compilando Typescript
- ▶ Archivos .ts y .js
- ▶ Interfaces
- ▶ Usando interfaces
- ▶ tsconfig.json
- ▶ Probando el tsconfig
- ▶ Import / Export

## ▶ PROGRAMACIÓN ORIENTADA A OBJETOS / 12 CLASES

- ▶ POO
- ▶ Una clase
- ▶ Clase "Libro"
- ▶ Instancias
- ▶ Métodos
- ▶ Constructor
- ▶ Clase Banda
- ▶ Clases con clases
- ▶ Pisos y deptos
- ▶ Herencia



- ▶ Clase: ProductoAlimenticio

- ▶ Clase: ProductoCongelado

## ➤ GIT / 4 CLASES

- ▶ Introducción a Git

- ▶ Un repo

- ▶ Mi primer repo

- ▶ Algunos comandos

## ➤ GITHUB / 11 CLASES

- ▶ ¿Qué es Github?

- ▶ Configurando Github

- ▶ Repo privado

- ▶ Repos remotos

- ▶ Un repo remoto

- ▶ Clona, modifica, commitea y pushea

- ▶ Forks y Pull Requests

- ▶ Proponé tus cambios

- ▶ Extendiendo clases

- ▶ Open source

- ▶ Forkear y clonar

## ➤ TEST RUNNERS / 2 CLASES

- ▶ Ava

- ▶ Testeando clases

## ➤ MVC / 2 CLASES

- ▶ MVC orientado a objetos

- ▶ Desafío

## ➤ ASYNC / 5 CLASES

- ▶ Sync y Async

- ▶ Callbacks
- ▶ Promesas
- ▶ Más promesas
- ▶ Métodos async

➤ **BUSCADOR DE PELÍCULAS ORIENTADO A OBJETOS / 3 CLASES**

- ▶ MVC
- ▶ Async / Await

- ▶ Consigna



## **Desarrollo web fullstack**

NIVEL 2

Aprenderás que es la web y como crear aplicaciones dinámicas y funcionales que vivan en internet. Crearemos un sistema basado en frontend y backend utilizando lo aprendido en el Nivel 1. Gracias a las prácticas y ejercicios, desarrollarás habilidades valiosas para el mercado laboral web.

### **16 CAPÍTULOS / 116 CLASES**

---

➤ **PLANNING / 1 CLASES**

- ▶ Introducción

➤ **LA WEB / 8 CLASES**

- ▶ Redes: TCP/IP
- ▶ Mi primer servidor
- ▶ El navegador
- ▶ URLs
- ▶ Protocolo HTTP
- ▶ HTML
- ▶ HTTPS
- ▶ Debuggeando la web

➤ **HTML / 3 CLASES**

- ▶ Introducción

▶ Tags / Etiquetas

▶ Mi servidor web

## ➤ CSS / 25 CLASES

▶ Un selector

▶ Un tunel

▶ Primeros estilos

▶ Modelo de cajas

▶ Maquetando

▶ Live server

▶ CSS: Flex

▶ Usando flexbox

▶ Diferencias entre navegadores: Can i use

▶ CSS: Grids

▶ Maquetar la grilla

▶ Mobile first

▶ Media queries

▶ Maquetando responsive

▶ Unidades de medida

▶ Maquetando con unidades de medida

▶ Precedencia y especificidad

▶ Selectores seguros

▶ Practicando selectores

▶ Pseudo-clases

▶ Pseudo-elementos

▶ Usando selectores

▶ BEM

▶ BEM en acción

▶ Maquetando con BEM

## ➤ MAQUETADO / 3 CLASES

- ⌕ Maquetando un form
- ⌕ Maquetando un layout
- ⌕ Consigna

## ➤ JAVASCRIPT WEB / 16 CLASES

- ⌕ Intro
- ⌕ Los grandes objetos
- ⌕ Rutas
- ⌕ `<script />`
- ⌕ `async / defer`
- ⌕ Comenzando con JS en el navegador
- ⌕ DOM
- ⌕ DOM: Buscando elementos
- ⌕ DOM: Creando elementos
- ⌕ Manipulando el DOM
- ⌕ DOM: Eventos
- ⌕ DOM: Bubbling
- ⌕ Moviendo el cuadrito
- ⌕ DOM: Forms
- ⌕ DOM: Evento Submit
- ⌕ Enviando forms

## ➤ APIS / 6 CLASES

- ⌕ ¿Qué es una API?
- ⌕ Fetch
- ⌕ Fetch: Métodos
- ⌕ Usando APIs
- ⌕ CMS
- ⌕ Administrando contenidos

➤ LIBRERÍAS WEB / 4 CLASES

- ⌕ Librerías: la solución para no reinventar la rueda
- ⌕ Librerías, APIs y frameworks
- ⌕ Librerías de CSS
- ⌕ Maquetando con Bulma

➤ COMPONENTES / 2 CLASES

- ⌕ Componentizando
- ⌕ Maquetando con componentes

➤ WEB & COMPONENTS / 1 CLASES

- ⌕ Consigna

➤ BUNDLERS / 6 CLASES

- ⌕ Módulos y TypeScript en la web
- ⌕ Modificando la webapp
- ⌕ Vite
- ⌕ Migra tu proyecto a Vite
- ⌕ Parcel
- ⌕ Migra tu proyecto a Parcel

➤ ARQUITECTURA WEB / 15 CLASES

- ⌕ Web Apps vs páginas tradicionales
- ⌕ Identificando SPAs
- ⌕ Rutas / Escenas
- ⌕ UI Kits
- ⌕ Custom elements
- ⌕ Creando componentes
- ⌕ Pages & components
- ⌕ Router & Pages
- ⌕ Router

- ▶ State
- ▶ Stateless / Statefull
- ▶ Todo list
- ▶ Wizard
- ▶ Web storage
- ▶ Post its

## ➤ PIEDRA, PAPEL O TIJERA / 5 CLASES

- ▶ Repaso
- ▶ Deploy a Github Pages
- ▶ Router Base Path
- ▶ Tips
- ▶ Consigna

## ➤ BACKEND / 13 CLASES

- ▶ Rutas
- ▶ Verbos
- ▶ Primera API
- ▶ Bases de datos
- ▶ Firebase
- ▶ Firestore
- ▶ Collections & documents
- ▶ Guardando y escribiendo datos
- ▶ Firebase: Realtime database
- ▶ Estructurando las databases
- ▶ Chat
- ▶ Rooms
- ▶ Chatrooms

## ➤ DEPLOY / 7 CLASES

- ▶ PAAS: Plataform as a Service

- ▶ CI: Continuous integration
- ▶ Env vars: Variables de entorno
- ▶ Statics server
- ▶ Deployar rooms
- ▶ Diseño de API con Postman
- ▶ Mi API con Postman

➤ PPT ONLINE / 1 CLASES

- ▶ Consigna



## Desarrollo fullstack avanzado

NIVEL 3

Desarrollarás aplicaciones web escalables y mantenibles utilizando las herramientas estandar. Estudiaremos las prácticas más habituales de la industria del desarrollo web y finalmente, analizaremos el mercado laboral del software y aprenderás cómo se trabaja en las grandes empresas.

### 23 CAPÍTULOS / 143 CLASES

---

➤ PLANNING / 3 CLASES

- ▶ Nivel 3
- ▶ OKRs
- ▶ Tu gran objetivo

➤ BASES DE DATOS RELACIONALES / 14 CLASES

- ▶ Bases de datos relacionales
- ▶ Postgres
- ▶ ORM - Sequelize
- ▶ Tablas y Modelos
- ▶ Mi primer modelo
- ▶ Modelado

- ▶ Modelemos datos
- ▶ Sistemas de autenticación
- ▶ Signup: Registro
- ▶ Signin: Inicio de sesión
- ▶ Authorization
- ▶ Auth
- ▶ Relaciones en Sequelize
- ▶ Product & Users

## ➤ ALGOLIA / 4 CLASES

- ▶ Geodata
- ▶ Algolia
- ▶ Integrando Algolia
- ▶ Geo búsquedas

## ➤ MVC / 4 CLASES

- ▶ Introducción
- ▶ Funciones ❤️
- ▶ Buenos controllers
- ▶ Mi perfil

## ➤ PET-FINDER / 1 CLASES

- ▶ Consigna

## ➤ INTRODUCCIÓN A REACT / 22 CLASES

- ▶ Hola React
- ▶ Componentes
- ▶ react-dom
- ▶ JSX
- ▶ Mi primer componente en React
- ▶ TypeScript ❤️ React



- ▶ Webpack
- ▶ Proyecto base
- ▶ Reactivo
- ▶ Reutilizable
- ▶ El state
- ▶ Lifecycle
- ▶ Clases vs. funciones
- ▶ Un buscador
- ▶ Functional components
- ▶ Login
- ▶ Hooks
- ▶ Hooks: Ciclo de vida
- ▶ Custom Hooks
- ▶ Buscador con hooks
- ▶ Creando custom hooks
- ▶ react-hook-form

## ➤ ARQUITECTURA EN APLICACIONES REACT / 8 CLASES

- ▶ Router
- ▶ Arquitectura
- ▶ Buscador en páginas
- ▶ One state to rule them all
- ▶ Contexto
- ▶ State manager
- ▶ Nuevo paradigma
- ▶ Un state lleno de hooks

## ➤ CSS IN JS / 2 CLASES

- ▶ Introducción
- ▶ UI components

## ➤ PETFINDER EN REACT / 3 CLASES

- ⌕ Deploy
- ⌕ Consideraciones
- ⌕ Consigna

## ➤ VERCEL / 2 CLASES

- ⌕ ¿Qué es Vercel?
- ⌕ Serverless API

## ➤ BACKEND II / 9 CLASES

- ⌕ Listas
- ⌕ Integrando varias bases de datos
- ⌕ Búsquedas con Algolia
- ⌕ Buscador de productos
- ⌕ Passwordless
- ⌕ Middlewares
- ⌕ Passwordless auth
- ⌕ BFF
- ⌕ GraphQL

## ➤ PAGOS ONLINE / 5 CLASES

- ⌕ Introducción
- ⌕ ¡Cuidado!
- ⌕ Link de pago
- ⌕ Confirmación de pago
- ⌕ Vaquitapp

## ➤ E-COMMERCE: BACKEND / 5 CLASES

- ⌕ Tests
- ⌕ Arquitectura
- ⌕ Validaciones

- ⌕ Problemas de CORS en Next.js

- ⌕ Consigna

## ➤ NEXT.JS / 4 CLASES

- ⌕ Pages

- ⌕ Server Side

- ⌕ SSR, SSG e ISG

- ⌕ Server Side Rendering

## ➤ SWR / 2 CLASES

- ⌕ SWR

- ⌕ Data fetching

## ➤ ARQUITECTURA / 8 CLASES

- ⌕ Capas

- ⌕ Path

- ⌕ Pages

- ⌕ Components

- ⌕ UI

- ⌕ Critical CSS

- ⌕ SVG

- ⌕ Algunos UI Components

## ➤ WPO & SEO / 5 CLASES

- ⌕ Performance

- ⌕ Web Vitals

- ⌕ Lighthouse

- ⌕ SEO

- ⌕ Mejoras

## ➤ TIPS AND TRICKS / 1 CLASES

- ⌕ Recursos

## ➤ E-COMMERCE: FRONTEND / 2 CLASES

- ⌚ Resolviendo problemas de CORS en Next.js
- ⌚ Consigna

## ➤ SCRUM / 9 CLASES

- ⌚ Agilidad
- ⌚ Scrum
- ⌚ User stories
- ⌚ Agilidad y proyectos
- ⌚ Lean Startup
- ⌚ MVP
- ⌚ Problemas y MVPs
- ⌚ Un brief
- ⌚ Mi primer brief

## ➤ CORRIENDO UN SPRINT / 8 CLASES

- ⌚ Introducción
- ⌚ Elegir el problema
- ⌚ Armá tu equipo
- ⌚ Prepará tu brief
- ⌚ Planning
- ⌚ Dailies
- ⌚ Retro
- ⌚ Entregable

## ➤ MUNDO LABORAL / 20 CLASES

- ⌚ Empresas & Procesos de selección
- ⌚ Categorías de empresas
- ⌚ Modalidades
- ⌚ Etapas de una búsqueda

- ▶ Desafíos
- ▶ Perfil profesional
- ▶ CV
- ▶ LinkedIn
- ▶ 1. Perfil de LinkedIn
- ▶ Búsquedas y Entrevistas
- ▶ Entrevistas
- ▶ Tips & tricks
- ▶ Programas de referidos
- ▶ Análisis e introspección
- ▶ Freelance
- ▶ ¿Dónde se buscan los trabajos freelance?
- ▶ Typeland
- ▶ Flujo de trabajo freelance
- ▶ Mi primer trabajo
- ▶ Retro

▶ EN BUSCA DE MI PRIMER TRABAJO / 2 CLASES

- ▶ Introducción

- ▶ Consigna