

Diplomado Full Stack Developer

Octubre 2024

Instructor: Nazir Rosas
Desarrollador de software

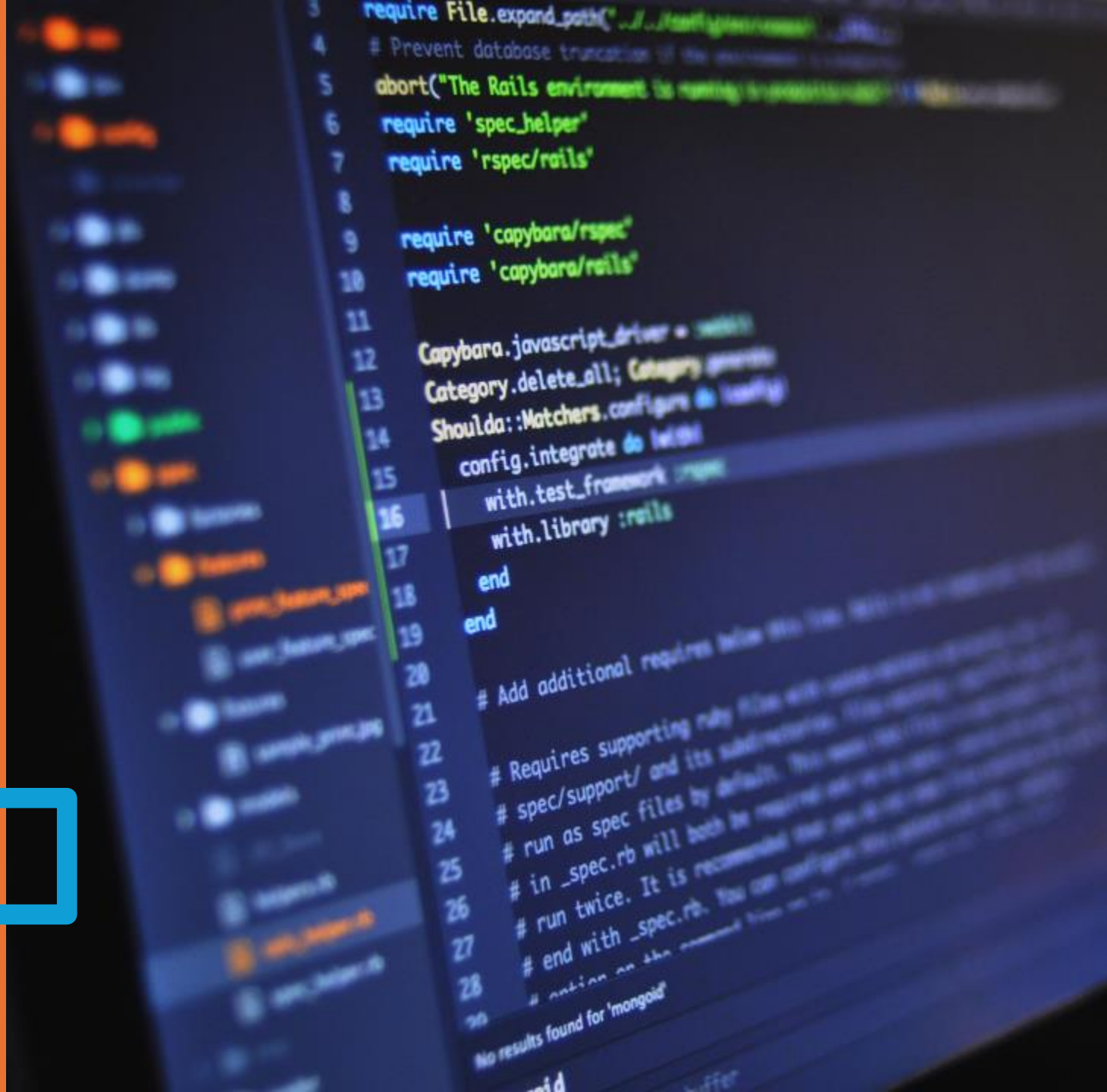


Ruta para ser Full Stack Developer





¡Bienvenidos!



```
3 require File.expand_path("../support", __FILE__)
4 # Prevent database truncation if the environment is production
5 abort("The Rails environment is running in production mode!")
6 require 'spec_helper'
7 require 'rspec/rails'
8
9 require 'capybara/rspec'
10 require 'capybara/rails'
11
12 Capybara.javascript_driver = :webkit
13 Category.delete_all; Category.create
14 Shoulda::Matchers.configure do |config|
15   config.integrate do |int|
16     with.test_framework :rspec
17     with.library :rails
18   end
19 end
20
21 # Add additional requires below this line. This is required for RSpec 3.0
22 # Requires supporting ruby files with static require for the classes and modules
23 # in spec/support/ and its subdirectories. Files matching *_spec.rb are ignored
24 # in _spec.rb will both be required when running _spec.rb and when running
25 # run twice. It is recommended that you do not use *_spec.rb files.
26 # end with _spec.rb. You can configure this behavior using RSpec::Core::Runner
27 # option on the command line.
28 #
29 # No results found for 'mongoid'
```

Sobre el diplomado



Perfil de Ingreso

- El Diplomado en Fullstack Developer está diseñado para aquellos interesados en adquirir habilidades integrales en el desarrollo de aplicaciones web, abarcando tanto el frontend como el backend. Este programa está orientado a personas con un interés marcado en la tecnología, la programación y el desarrollo de software, que desean obtener una formación completa para enfrentar los desafíos del desarrollo web moderno. A continuación, se describen las características ideales del perfil de ingreso:
- **1. Conocimientos Previos**
- **Fundamentos de Programación:** Se espera que los aspirantes tengan conocimientos básicos en algún lenguaje de programación, como JavaScript, Python, Java, o similares.
- **Desarrollo Web Básico:** Familiaridad con tecnologías web básicas, incluyendo HTML, CSS y JavaScript. Conocimientos sobre estructuras de datos y algoritmos son valorados.
- **Bases de Datos:** Comprensión elemental sobre bases de datos relacionales y/o no relacionales, con conocimiento básico en SQL o tecnologías similares.

Perfil de Ingreso

- **2. Competencias Técnicas**
- **Lógica y Pensamiento Crítico:** Capacidad para resolver problemas de manera lógica y estructurada, aplicando técnicas de programación.
- **Interés en Tecnologías Web:** Curiosidad por aprender y trabajar con las tecnologías y herramientas más utilizadas en el desarrollo web moderno, como frameworks de JavaScript, servidores, y APIs.
- **3. Habilidades Personales**
- **Proactividad y Autodisciplina:** Motivación para aprender de forma autónoma, enfrentando desafíos técnicos con creatividad y persistencia.
- **Trabajo en Equipo:** Capacidad para colaborar efectivamente en entornos de trabajo en equipo, comunicando ideas de manera clara y contribuyendo al desarrollo colectivo de proyectos.
- **Adaptabilidad:** Disposición para adaptarse a nuevas tecnologías y metodologías de trabajo, en un entorno de constante evolución tecnológica.

Perfil de Ingreso

- **4. Requisitos Académicos**
- **Formación Académica:** Aunque no es obligatorio, se recomienda que los postulantes cuenten con un nivel académico mínimo de bachillerato concluido o formación técnica equivalente, preferiblemente en áreas relacionadas con la informática, sistemas computacionales, o similares.
- Este diplomado es ideal para quienes desean fortalecer sus competencias en desarrollo web y buscan integrarse o avanzar en la industria tecnológica, abordando tanto el diseño y la experiencia de usuario como la lógica de servidor y la gestión de bases de datos.

Perfil de Egreso

- Al finalizar el Diplomado en Fullstack Developer, los egresados estarán capacitados para desempeñarse de manera integral en el desarrollo de aplicaciones web, tanto en el frontend como en el backend. A continuación, se detallan las competencias y habilidades que habrán adquirido:
- **1. Habilidades Técnicas**
- **Desarrollo Frontend:** Capacidad para crear interfaces de usuario interactivas y responsivas utilizando HTML, CSS, y JavaScript, además de manejar frameworks modernos como React o Angular.
- **Desarrollo Backend:** Competencia en la implementación de la lógica de servidor, manejo de bases de datos, y creación de APIs utilizando tecnologías como Node.js, Python, Java, o similares.
- **Integración de Tecnologías:** Habilidad para conectar y manejar de manera eficiente la interacción entre el frontend y el backend, garantizando el funcionamiento fluido de las aplicaciones web.

Perfil de Egreso

- **2. Gestión de Proyectos**
- **Desarrollo Ágil:** Conocimiento de metodologías ágiles como Scrum y Kanban para la gestión efectiva de proyectos, asegurando entregas de alta calidad en tiempos óptimos.
- **Control de Versiones:** Dominio de herramientas como Git para el control y seguimiento del desarrollo de software en colaboración con equipos de trabajo.
- **3. Habilidades Personales**
- **Resolución de Problemas:** Capacidad para analizar y solucionar problemas complejos de desarrollo web, aplicando lógica y creatividad.
- **Colaboración y Comunicación:** Habilidad para trabajar en equipo, comunicar ideas técnicas de manera clara y efectiva, y colaborar en la creación de soluciones de software.
- **Adaptación al Cambio:** Preparación para adaptarse rápidamente a nuevas tecnologías y tendencias en el desarrollo web, manteniéndose actualizado en un campo en constante evolución.

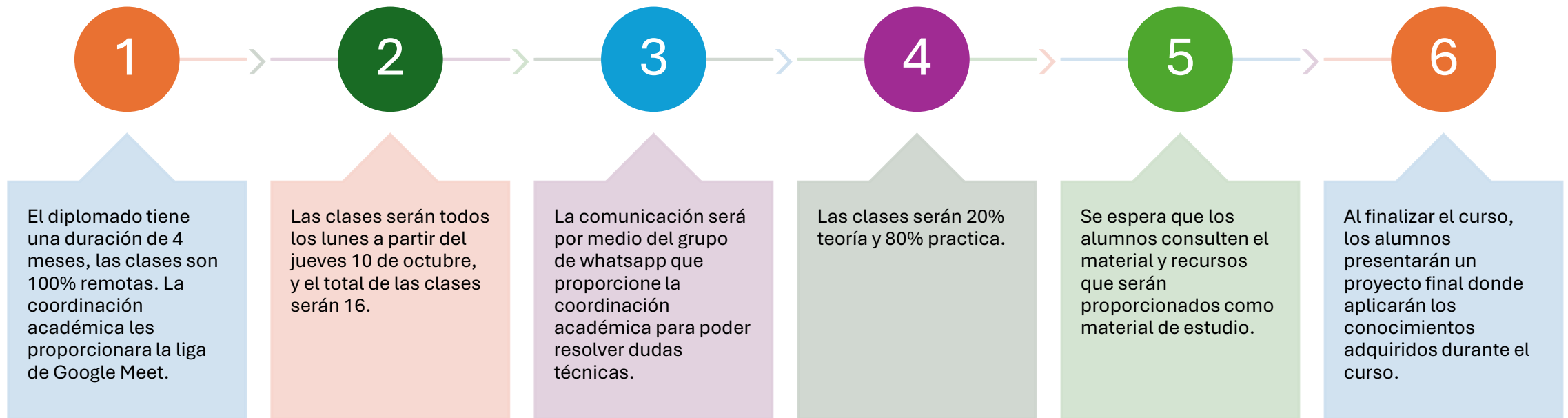
Perfil de Egreso

- **4. Aplicación Profesional**
- **Capacidad para Desarrollar Proyectos Web:** Los egresados podrán diseñar, desarrollar, y mantener aplicaciones web completas, desde la concepción de la idea hasta la implementación y optimización final.
- **Preparación para el Mercado Laboral:** Contarán con un portafolio de proyectos que demostrará sus habilidades y estará listo para enfrentar los desafíos del mundo laboral en roles como desarrollador fullstack, ingeniero de software, o desarrollador web.
- Los egresados del diplomado estarán preparados para integrarse con éxito en la industria tecnológica, aportando soluciones innovadoras y efectivas en el desarrollo de aplicaciones web modernas.

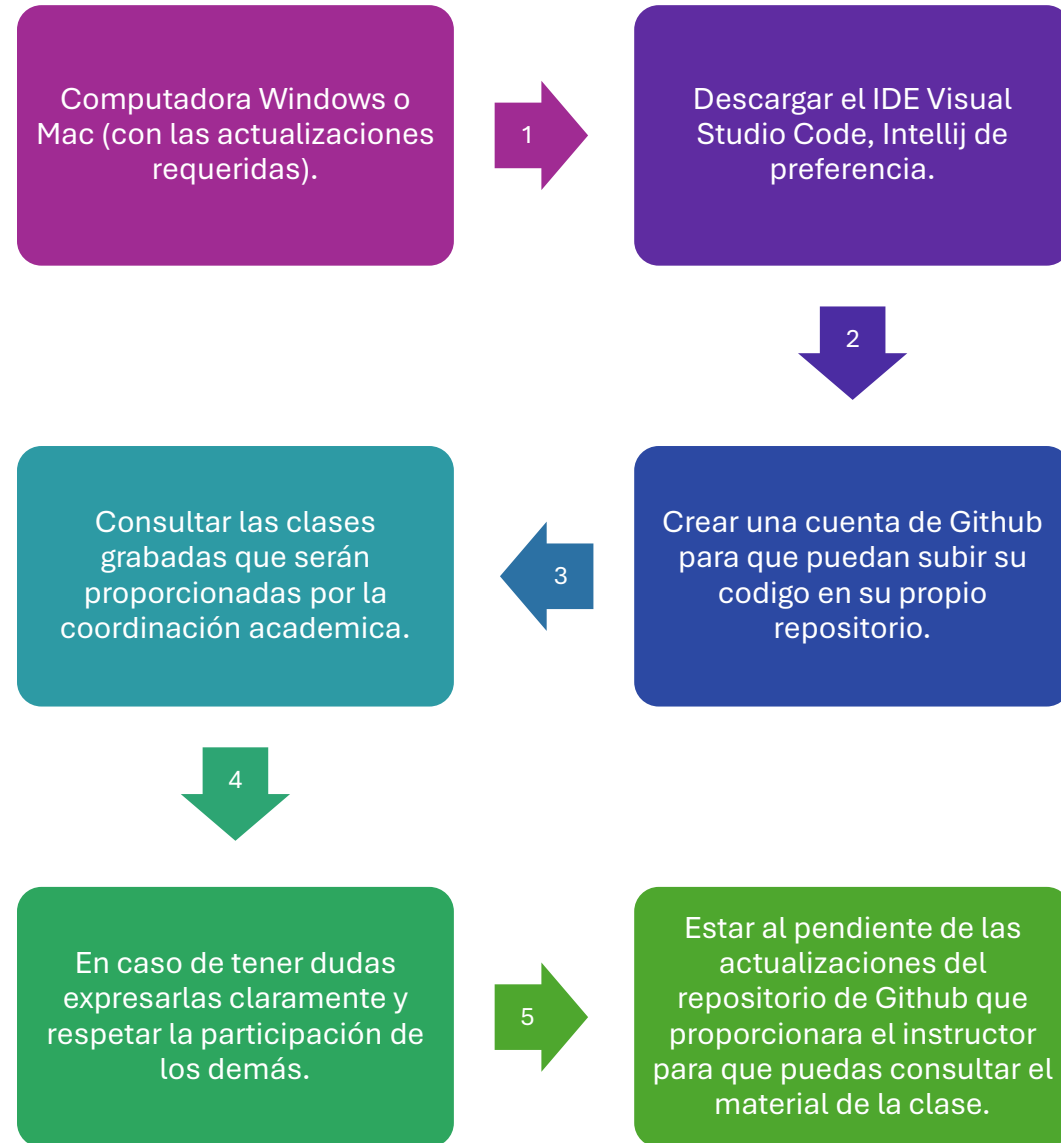
Duración



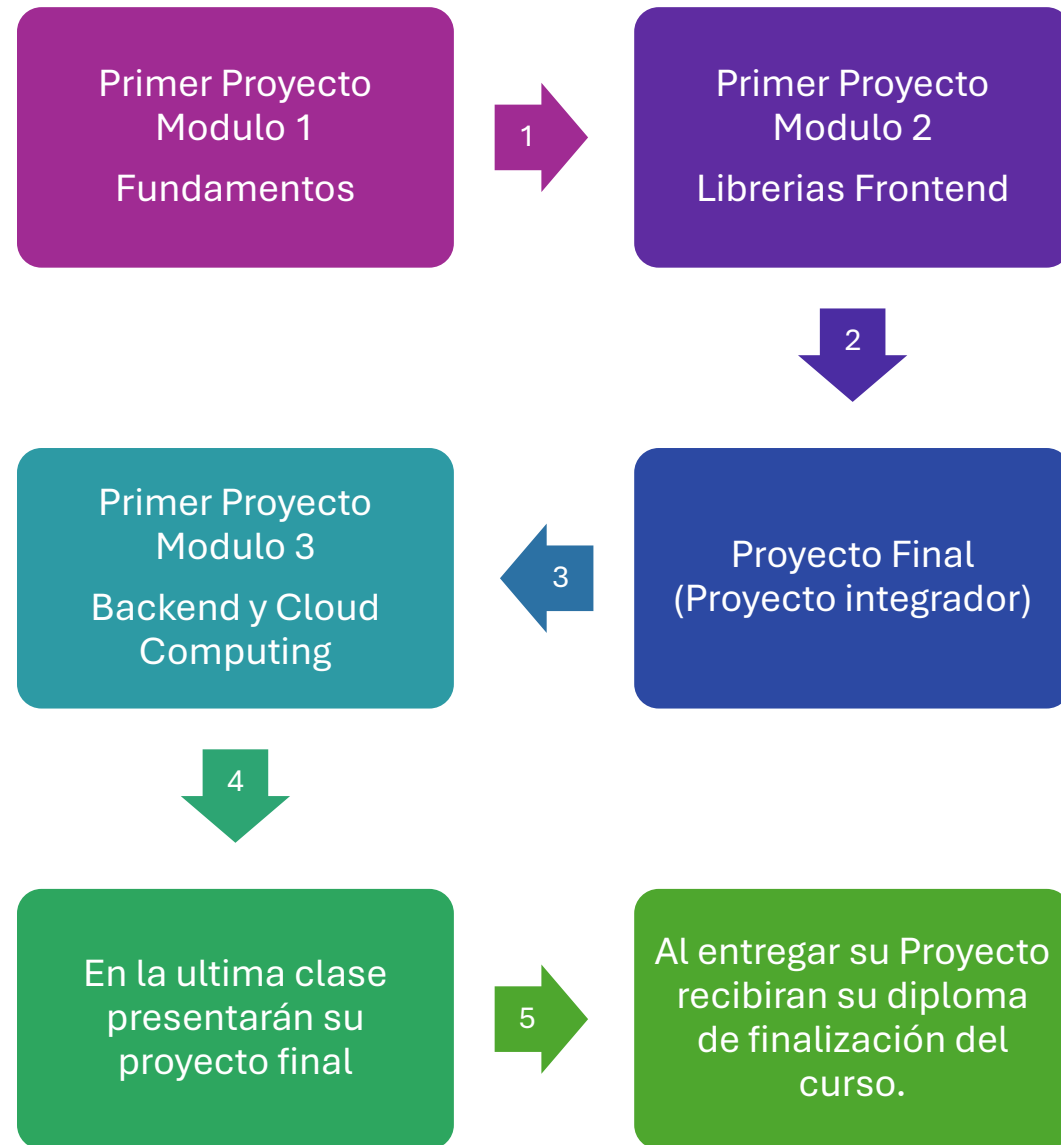
Clases



Herramientas para la clase



Evaluación por proyectos mensuales



Conoce a tu
instructor



Instructor Nazir Rosas

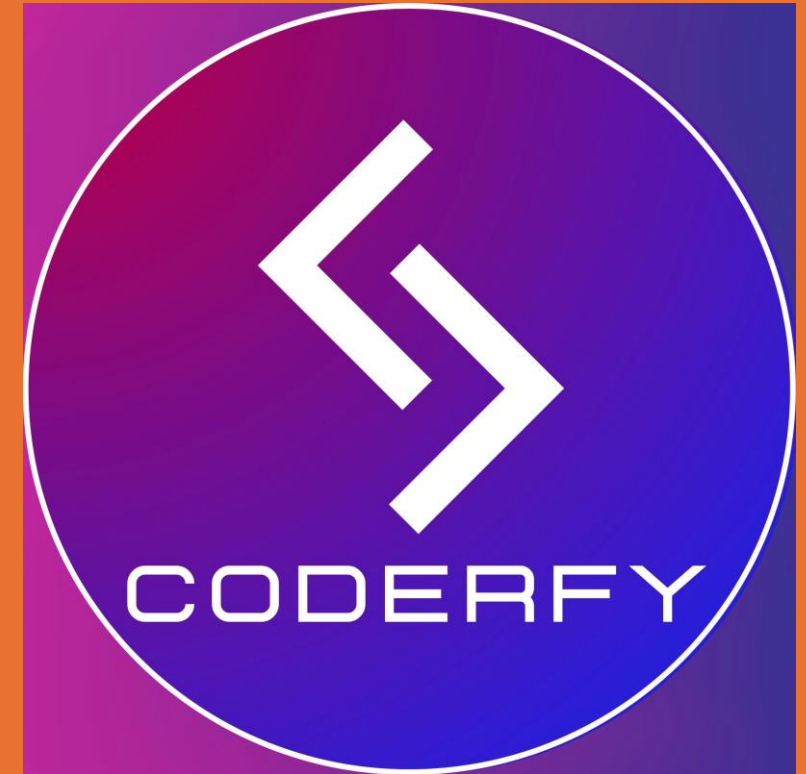
- Soy Desarrollador de software con 4 años de experiencia en la industria. A lo largo de mi carrera me he desempeñado como frontend developer, backend developer, fullstack developer, UX Engineer y consultor de tecnología.
- Trabaje como consultor en Deloitte Digital, en el área de Digital Customer, participando en diversos proyectos para clientes internacionales como Ford Estados Unidos, Coppel, Sky Mexico y centroamerica y Google.
- Cuando era estudiante tuve la oportunidad de realizar una pasantía en Microsoft Mexico.
- Estudie la Ingenieria en Sistemas Computacionales en la Universidad Latinoamericana, además de estudiar un diplomado en Diseño y Desarrollo de Apps dictado por la Universidad Anáhuac.



Instructor Nazir Rosas



Fundamentos de Front End



Ruta desarrollador Full Stack

**Fundamentos**




HTML CSS JavaScript




Estos son los pilares de la web. **Estructurar** contenido (HTML), darle **estilo** (CSS) y agregar **interactividad** (JavaScript)



Git GitHub

Herramientas indispensables para el control de versiones y colaboración en proyectos







**Front End**



React Vue.js Angular


Librerías y frameworks para construir interfaces de usuario más complejas y eficientes.





Back End



Node.js Python Java Ruby SQL Mongo DB

Lenguajes de programación y frameworks para el funcionamiento lógico del Desarrollo




**Frameworks Back End**



Django Spring Laravel ASP.NET Core

Herramientas fundamentales para agilizar el desarrollo de aplicaciones web y simplificar muchas tareas comunes.

Cloud Computing



AWS Google Cloud Platform Azure

Utiliza plataformas para desplegar aplicaciones en la nube.

Conceptos basicos

Desarrollador Frontend

Un desarrollador frontend se especializa en la creación de la parte visual y la experiencia de usuario de un sitio web o aplicación. Su trabajo se centra en lo que los usuarios ven y con lo que interactúan directamente. Utiliza tecnologías como HTML, CSS y JavaScript para diseñar y desarrollar interfaces atractivas y funcionales. Los desarrolladores frontend se encargan de garantizar que los sitios web sean responsivos (que se vean y funcionen bien en cualquier dispositivo) y accesibles, además de optimizar el rendimiento visual y la interacción del usuario.

Desarrollador Backend

El desarrollador backend se enfoca en la lógica del servidor y la gestión de bases de datos, que son fundamentales para el funcionamiento de una aplicación web, aunque no sean visibles para el usuario final. Trabaja con lenguajes de programación del lado del servidor como Python, Java, Ruby, PHP, o Node.js, y maneja bases de datos como MySQL, PostgreSQL, o MongoDB. Los desarrolladores backend se encargan de crear, mantener y asegurar que los servidores, bases de datos, y APIs funcionen de manera eficiente, asegurando que la aplicación pueda procesar solicitudes y enviar respuestas correctamente.

Conceptos basicos

Desarrollador Fullstack

Un desarrollador fullstack tiene conocimientos tanto de frontend como de backend, lo que le permite trabajar en todas las partes de una aplicación web. Este profesional puede diseñar interfaces de usuario y, al mismo tiempo, gestionar la lógica del servidor y las bases de datos. Los desarrolladores fullstack son versátiles y pueden encargarse de proyectos de desarrollo web completos, desde la conceptualización hasta la implementación, manejando todas las capas tecnológicas necesarias para el funcionamiento de la aplicación. Su habilidad para entender tanto el lado del cliente como el del servidor les permite colaborar de manera efectiva con otros desarrolladores y optimizar la interacción entre frontend y backend.

Lenguajes de programación

Un lenguaje de programación es un conjunto de instrucciones y reglas que permiten a los desarrolladores comunicarse con una computadora para crear software, aplicaciones y sistemas. Estos lenguajes proporcionan una forma estructurada de dar instrucciones a la máquina, permitiendo realizar tareas, manipular datos, y automatizar procesos. Los lenguajes de programación varían en su sintaxis y funcionalidades, adaptándose a diferentes necesidades y tipos de desarrollo.

Lenguajes de Programación más Usados por Desarrolladores Fullstack

Los desarrolladores fullstack suelen utilizar una combinación de lenguajes que les permite trabajar tanto en el frontend como en el backend de las aplicaciones web. Los lenguajes más comunes incluyen:

- **JavaScript:** Esencial para el desarrollo frontend y, con Node.js, también se utiliza en el backend. Es versátil y ampliamente utilizado en el desarrollo web.
- **Python:** Popular en el backend por su simplicidad y eficiencia. Se usa en frameworks como Django y Flask para crear aplicaciones web robustas.
- **Java:** Utilizado en el backend, especialmente en aplicaciones empresariales. Es conocido por su portabilidad y seguridad.
- **Ruby:** Utilizado en el backend con el framework Ruby on Rails, que permite un desarrollo rápido y eficiente de aplicaciones web.
- **PHP:** Común en el backend, especialmente en el desarrollo de sitios web dinámicos y aplicaciones web, como WordPress.

Herramientas de un desarrollador

Los desarrolladores fullstack utilizan una variedad de herramientas en su trabajo diario para gestionar todo el ciclo de vida del desarrollo de software. Algunas de las herramientas más comunes incluyen:

- **Control de Versiones (Git):** Git es esencial para el control de versiones, permitiendo a los desarrolladores rastrear cambios en el código, colaborar con otros desarrolladores, y gestionar diferentes versiones de un proyecto a través de plataformas como GitHub o GitLab.
- **Entornos de Desarrollo Integrado (IDE):** Herramientas como Visual Studio Code, IntelliJ IDEA, o Sublime Text son utilizadas para escribir, editar, y depurar código. Estos IDEs ofrecen funcionalidades avanzadas que mejoran la productividad y la calidad del código.
- **Frameworks y Librerías:** Para el frontend, frameworks como React, Angular o Vue.js son comunes, mientras que en el backend, se utilizan frameworks como Express.js (Node.js), Django (Python), o Spring Boot (Java). Estas herramientas facilitan la creación de aplicaciones web eficientes y escalables.
- **Gestión de Bases de Datos:** Herramientas como MySQL Workbench, pgAdmin, o MongoDB Compass son usadas para diseñar, gestionar, y consultar bases de datos. Permiten a los desarrolladores interactuar fácilmente con bases de datos relacionales y no relacionales.

Herramientas de un desarrollador

- **Sistemas de Gestión de Proyectos:** Plataformas como Jira, Trello, o Asana se utilizan para la gestión ágil de proyectos, permitiendo a los equipos planificar, seguir el progreso, y colaborar de manera eficiente.
- **Servicios en la Nube (AWS, Azure, Google Cloud):** Los desarrolladores fullstack a menudo utilizan servicios en la nube para desplegar y gestionar aplicaciones, aprovechar bases de datos escalables, y usar recursos de computación bajo demanda.
- **Contenedores y Orquestación (Docker, Kubernetes):** Herramientas como Docker y Kubernetes son usadas para empaquetar aplicaciones y gestionar su despliegue en entornos de producción, asegurando que se ejecuten de manera consistente en diferentes entornos.
- Estas herramientas permiten a los desarrolladores fullstack manejar todas las partes de una aplicación web, desde la escritura de código hasta la gestión de proyectos y el despliegue en producción.



Introduccion a JavaScript



¿Qué es JavaScript?

JavaScript es un lenguaje de programación de alto nivel y basado en texto que se utiliza principalmente para crear y controlar el contenido dinámico en los sitios web. Es el lenguaje que permite agregar interactividad a las páginas web, como formularios que validan datos, animaciones, menús desplegables, y mucho más. JavaScript se ejecuta en el navegador del usuario (frontend) y, con la introducción de tecnologías como Node.js, también puede ser utilizado en el lado del servidor (backend). Es uno de los lenguajes más populares y versátiles en el desarrollo web, siendo esencial para la creación de aplicaciones modernas y dinámicas.

¿Qué es JavaScript?

JavaScript es un lenguaje de programación de alto nivel y basado en texto que se utiliza principalmente para crear y controlar el contenido dinámico en los sitios web. Es el lenguaje que permite agregar interactividad a las páginas web, como formularios que validan datos, animaciones, menús desplegables, y mucho más. JavaScript se ejecuta en el navegador del usuario (frontend) y, con la introducción de tecnologías como Node.js, también puede ser utilizado en el lado del servidor (backend). Es uno de los lenguajes más populares y versátiles en el desarrollo web, siendo esencial para la creación de aplicaciones modernas y dinámicas.

Sintaxis

La sintaxis de JavaScript define las reglas y la estructura que deben seguirse para escribir código válido en este lenguaje. A continuación, se describen algunos de los elementos clave de la sintaxis de JavaScript:

1. Variables

Las variables se utilizan para almacenar datos que pueden ser utilizados y manipulados en el código. En JavaScript, se pueden declarar usando var, let, o const:

```
let nombre = "Juan"; // Variable que puede cambiar  
const edad = 25;     // Variable constante, no puede cambiar
```

Sintaxis

2. Tipos de Datos

JavaScript maneja varios tipos de datos, como números, cadenas de texto, booleanos, y más:

```
let numero = 10;           // Número
let texto = "Hola, Mundo"; // Cadena de texto
let esVerdad = true;       // Booleano (verdadero o falso)
```

Sintaxis

3. Operadores

Los operadores permiten realizar operaciones matemáticas, lógicas, y de asignación:

```
let suma = 5 + 3;      // Suma (8)
let resta = 10 - 4;    // Resta (6)
let esIgual = (5 == 5); // Comparación (true)
```

Sintaxis

4. Funciones

Las funciones son bloques de código que realizan una tarea específica y pueden ser reutilizados:

```
function saludar() {  
    console.log("¡Hola!");  
}  
  
saludar(); // Llama a la función e imprime "¡Hola!" en la consola
```

Sintaxis

5. Condicionales

Las estructuras condicionales permiten ejecutar diferentes bloques de código según ciertas condiciones:

```
let edad = 18;

if (edad >= 18) {
    console.log("Eres mayor de edad");
} else {
    console.log("Eres menor de edad");
}
```

Sintaxis

6. Bucles

Los bucles se utilizan para repetir un bloque de código varias veces:

```
for (let i = 0; i < 5; i++) {  
    console.log(i); // Imprime los números del 0 al 4  
}
```

7. Objetos

En JavaScript, los objetos son estructuras que permiten almacenar datos en pares clave-valor:

```
let persona = {  
    nombre: "Juan",  
    edad: 25,  
    saludar: function() {  
        console.log("Hola, " + this.nombre);  
    }  
};  
  
persona.saludar(); // Llama a la función e imprime "Hola, Juan"
```