

Máster Frontend

Temario

Master Frontend Lemoncode decimoctava edición, presentación y temario

Lemoncode
c/ Don Cristian 5
Málaga
www.lemoncode.net



Lemoncode

Somos un equipo de desarrolladores de dilatada experiencia, establecidos como grupo en 2010. Estamos especializados en tecnologías web, nodejs y .NET.

Tenemos experiencia en el desarrollo de productos completos desde su gestación a su implantación, lo que incluye la planificación, la arquitectura, el abordaje de problemas complejos y la definición de una metodología de trabajo. Somos ágiles y nos centramos tanto en proporcionar valor como en obtener *feedback* temprano.

Aprovechando nuestra experiencia, ofrecemos una formación especializada orientada a situaciones reales, con un enfoque eminentemente práctico. Nuestros docentes son expertos en la materia que imparten.

Objetivos de este máster

El objetivo de este máster es formar a desarrolladores especializados en el área de Front-End:

- Que sepan implementar un *layout responsive*, que se visualice bien en diferentes tipos de dispositivos y navegadores.
- Qué conozcan los desafíos de un desarrollo Front-End y cómo tratarlos.
- Qué conozcan lo que ofrece el lenguaje, tanto ES6 como TypeScript.
- Que conozcan los *frameworks* más populares del mercado con sus pros y contras, y tengan criterio para elegir uno u otro dependiendo del desarrollo que tengan que realizar.
- Que puedan asegurar un buen nivel de calidad en sus desarrollos mediante la implementación de pruebas unitarias en todos los aspectos de un proyecto Front.
- Que conozcan cómo integrarse con *backends* (lado servidor).
- Que tengan en cuenta aspectos tan importantes como el de la seguridad, o el rendimiento de los sitios y aplicaciones que desarrollen.
- Que sepan como desplegar sus proyectos en la nube de manera automatizada.

Temario

UX

- User Experience.
- Diseño para desarrolladores.
- Figma.
- Patrones de diseño

Layout

- Conceptos básicos HTML + CSS.



- Maquetación responsiva.
- SASS
- Concepto de diseño web para desarrolladores.
- Animaciones CSS.
- De Figma a CSS

Base Lenguaje:

- Javascript ES5.
- Javascript ES6 + propuestas principales ES7.
- Typescript.
- Programación funcional con JavaScript.
- Lodash.

Herramientas de Bundling

- Conceptos herramientas de Bundling.
- Webpack.
- Vite.
- Librerías con Vite.

Frameworks actuales:

- React:
 - Introducción a React.
 - Componentes.
 - Hooks.
 - Navegación.
 - Acceso a datos.
 - Arquitectura:
 - CSS in JS.
 - NextJS.
- Angular:
 - Conceptos básicos.
 - Arquitectura.
 - Mostrando datos.
 - Servicios.
 - Acceso a datos.
 - Navegación.
 - RxJs.
 - Arquitectura
- Vuejs:
 - Conceptos Vue.
 - Vue-cli
 - Computed Properties, Dynamic Attributes.
 - VueRouter
 - Composition API
 - Async setup.
 - Slots.



- Introducción a Svelte + Svelte Kit.
- Introducción a SolidJS.
- Introducción a Lit.
- Introducción a Astro.
- Introducción a D3js
- Introducción a Web Components.
- Principios Solid.
- Accesibilidad.
- MicroFront Ends
- Pasarelas de pago.

Pruebas unitarias

- Conceptos.
- Librerías (Jest / Vitest).
- Mocking.
- React Testing Library.
- Cobertura de código.
- Integración continua.
- Unit Testing con Angular.
- Introducción a e2e testing:
 - Cypress.
 - Playwright.

Integración con API's REST

- Conceptos API REST.
- Librerías de acceso a datos:
 - Fetch.
 - Axios.
- Manejo de seguridad:
 - Concepto de sesión.
 - Cookie / Header.
 - JWT
- Leyendo datos de fuentes en otros dominios (CORS).
- Introducción a GraphQL.
- Introducción a Socket.io
- OAUTH 2.

Despliegue en la nube

- Soluciones en la nube, conceptos.
- Docker.
- Azure.
- Amazon.
- Render.
- Github Actions + Turbo Repo + Changeset

Movilidad

- PWA
- React Native



Áreas de conocimiento

Frontend

Figma, React, Nextjs, Angular, Svelte, Vuejs, Flexbox, CSS Grid, Material Design, Jest, Vitest, Webpack, Vite, HTML5, CSS, JavaScript (ES6/ES7), TypeScript, d3js, CSS in JS, GraphQL, Blazor, Web components, socket.io.

Metodología

Agile, desarrollo en *sprints*.

Revisiones de código, centradas en la calidad y la facilidad de mantenimiento.

Branch-per-feature, utilizando como repositorio distribuido Git.

Infraestructura

Docker

Cloud:

- Microsoft Azure
- Amazon AWS
- Render

Repositorios:

- Github

Movilidad

React Native

PWA

