



Desarrollo web en Entorno Cliente

Tarea: Herramientas de Programación en Cliente

**2º Año de Grado Superior de Desarrollo de Aplicaciones Web
Curso 2024-2025**

Gabriel Callejón Sánchez

Índice

(Este PDF si es interectuable y se puede utilizar los hiperenlaces señalados con texto subrayado de color azul)

1. [React JS](#)
2. [Angular](#)
3. [Pregunta Tipo Test](#)

1. ReactJS

ReactJs es una librería de JavaScript de código abierto, que en la actualidad es mantenido por Facebook y la comunidad de Software libre y posee más de 1000 desarrolladores en el proyecto.

Su función es crear interfaces de usuario para así hacer más sencillo el desarrollo de aplicaciones, cuya implementación es muy sencilla:

```
1 <!DOCTYPE HTML>
2 <html lang = "en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 </head>
7 <body>
8   <h1>Página web con ReactJS implementado</h1>
9
10  <!--Librerías para implementar ReactJS-->
11  <script src="https://unpkg.com/react@18/umd/react.development.js" crossorigin></script>
12  <script src="https://unpkg.com/react-dom@18/umd/react-dom.development.js" crossorigin></script>
13 </body>
14 </html>
```

Aunque no notemos cambios en la página, podemos empezar a utilizar sus librerías, que nos da la oportunidad de poder desarrollar aplicaciones como [Buscador de Emojis](#) o [Calculadoras en IOs](#).

Una peculiaridad muy buena de React, es que se desarrolla en una página, es decir, que la aplicación solo necesita cargar una vez para luego actualizarse dinámicamente sin tener que recargar la página (Término conocido como SPA).

En cuanto a los mecanismos de integración, React es capaz de interactuar con otras librerías, frameworks o hasta con el DOM. Algún ejemplo puede ser:

React Router, que permite integrar rutas y gestionar la navegación entre diferentes componentes.

Context API, que puede usarse para compartir datos globales, como el tema de la aplicación o la información del usuario. Esto se debe a que olvida la jerarquía de los componentes reutilizables de otras librerías de la aplicación.

2. Angular

Angular es un Framework (Conjunto de herramientas e implementaciones que facilitan el desarrollo de aplicaciones) orientado a aplicaciones web, creado por [TypeScript](#). Este es de código abierto y lo mantiene Google.

Al igual que ReactJS, solo necesita cargarse una vez para luego actualizarse dinámicamente sin recargar la página. Angular ofrece un montón de herramientas para controlar la lógica de una app, gestionar su estado y la comunicación entre servidores, además de facilitar la creación de interfaces interactivas para el usuario.

Angular necesita de una versión estable de Node.js (IDE enfocado en el desarrollo de JavaScript). Algunos mecanismos de integración para Angular:

Observables y RxJS: Observables sirve para el manejo de eventos asíncronos y flujos de datos. Así se pueden integrar los datos de las APIs y manejar la comunicación en tiempo real.

Ruteo: Angular tiene un sistema de enrutamiento que permite la navegación entre diferentes vistas y componentes. Así, se pueden integrar de forma fluida varias secciones de una aplicación.

3. Pregunta Tipo test:

Son Mecanismos de integración de ReactJS:

a) **React Router**

b) **RxJS**

c) **Context API**

d) **SKIB**

Mantiene ReactJS:

a) **Facebook**

b) **IBM**

b) **Madero.net**

b) **Su propia comunidad activa**

Están relacionados con Angular:

a) **LinkedIn**

b) **TypeScript**

c) **Node.js**

d) **Archive.org**

No son funciones de Angular:

- a) Gestionar el DOM
- b) Gestionar la comunicación entre servidores
- c) Gestionar de forma sola una base de datos
- d) Facilitar la creación de interfaces interactivables para el usuario