

Tecnologías para Pruebas Unitarias en JavaScript

Las pruebas unitarias son una parte fundamental del desarrollo de software, ya que permiten verificar el correcto funcionamiento de los componentes individuales de una aplicación. En JavaScript, existen diversas tecnologías para realizar pruebas unitarias, pero Jest se destaca por su popularidad, facilidad de uso y amplia adopción en la comunidad.

1. Instalación de Jest

- Crear un nuevo proyecto (si no existe):

```
mkdir
```

```
mi-proyecto
```

```
cd
```

```
mi-proyecto
```

```
npm init -y
```

- Instalar Jest como una dependencia de desarrollo:

```
npm install --save-dev jest
```

2. Configuración Básica

- Modificar package.json y agregar el script 'test': 'jest'.

3. Estructura

de Archivos

```
mi-proyecto/
```

```
  ■■■ src/
```

```
  ■ ■■■ funciones.js
```

```
  ■■■ tests/
```

```
    ■■■ funciones.test.js
```

4. Creación de Funciones para Probar

- Implementar funciones como suma, multiplicación y dividir en funciones.js.

5. Escribir Pruebas Unitarias con Jest

- Definir tests con expect() en funciones.test.js.

-

6. Ejecutar las Pruebas

- Usar npm test para ejecutar las pruebas.

7. Uso de Mocks y Simulaciones

- Simulación de peticiones con jest.mock().

8. Verificación de Cobertura de Código

- Ejecutar npm run test:coverage para revisar cobertura.

9. Pruebas de Funciones Asíncronas

- Uso de async/await en Jest para probar código asíncrono.

10. Pruebas de Integración con el DOM

- Simulación de manipulación del DOM en Jest.