



## Estructura del Nombre del Archivo de Prueba

El formato que Playwright busca es: [nombre-de-la-prueba].[sufijo].[extensión]

### 1. test-titulo (Nombre descriptivo)

- **Propósito:** Es la parte descriptiva que tú eliges. Debe indicar de qué trata el archivo o el grupo de pruebas que contiene.
- **Ejemplo:** En nuestro caso, test-titulo sugiere que las pruebas en este archivo están enfocadas en validar el título de la página web.

### 2. .spec o .test (Sufijo del archivo)

- **Propósito:** Este es el componente **más crítico**. Playwright (y otras herramientas como Jest, Cypress o Mocha) lo utiliza como una **marca o patrón** para **identificar** qué archivos dentro de tu proyecto son archivos de pruebas.
- **Alternativas:** Puedes usar .spec (de *specification*, especificación) o .test. Playwright está configurado por defecto para buscar cualquiera de los dos.
- **Mecanismo:** Cuando ejecutas `npx playwright test`, el *runner* (el programa que ejecuta las pruebas) escanea tu proyecto buscando cualquier archivo que termine con `*.spec.js` (o `.ts`) o `*.test.js` (o `.ts`).

### 3. .js o .ts (Extensión)

- **Propósito:** Indica el lenguaje de programación en el que está escrito el código.
  - **.js:** Indica que el código está escrito en **JavaScript** puro (como en nuestro ejemplo).
  - **.ts:** Indica que el código está escrito en **TypeScript** (el formato recomendado para proyectos Playwright grandes, debido a las ventajas del tipado estático).
- 

### Conclusión: El "Por Qué"

El nombre test-titulo.spec.js es importante porque la parte **.spec.js** le dice al ejecutor de pruebas de Playwright: **"Este archivo contiene definiciones de pruebas que deben ser ejecutadas."**

Si no incluyeras este sufijo (por ejemplo, si lo llamaras simplemente test-titulo.js), el *framework* ignoraría el archivo y las pruebas que contiene no se ejecutarían.