**Infografía Interactiva: Tipos de Pruebas de Software**

1. **Prueba Unitarias (Unit Testing)**
   * **Descripción:** Verifican el correcto funcionamiento de componentes individuales o funciones.
   * **Cuándo aplicarlas:** Cuando estás desarrollando una función específica, por ejemplo, asegurarte que una función de sumar números regrese el resultado correcto.
   * **Ejemplo:** Aplicar pruebas unitarias a una función que calcula descuentos en un carrito de compras.
   * **Imagen:** Icono de una lupa sobre una pieza de código.
   * **Interactividad sugerida:** Al hacer clic en la imagen, se despliega el ejemplo detallado de código.
2. **Pruebas de Integración (Integration Testing)**
   * **Descripción:** Validan la interacción entre múltiples componentes o módulos de un sistema.
   * **Cuándo aplicarlas:** Al conectar dos sistemas o servicios, como verificar que el frontend se comunique correctamente con el backend.
   * **Ejemplo:** Probar la integración entre un sistema de autenticación y el sistema de pagos.
   * **Imagen:** Dos piezas de rompecabezas uniéndose.
   * **Interactividad sugerida:** Un botón que al hacer clic muestra la explicación de cómo se integran los módulos.
3. **Pruebas Funcionales (Functional Testing)**
   * **Descripción:** Se asegura de que el sistema funcione según los requisitos establecidos.
   * **Cuándo aplicarlas:** Después de terminar el desarrollo de una funcionalidad completa, como cuando terminas la parte de registro de usuarios.
   * **Ejemplo:** Probar que el formulario de registro en una aplicación permita crear usuarios con todos los campos obligatorios completos.
   * **Imagen:** Una lista de verificación.
   * **Interactividad sugerida:** Al hacer clic, mostrar cómo se implementan estos requisitos en un sistema real.
4. **Pruebas de Rendimiento (Performance Testing)**
   * **Descripción:** Evalúan el rendimiento de un sistema bajo carga, estrés, y volumen.
   * **Cuándo aplicarlas:** Cuando necesitas saber cómo se comportará el sistema en producción con una alta cantidad de usuarios, por ejemplo, un sitio de e-commerce en Black Friday.
   * **Ejemplo:** Probar la carga de una API para asegurar que soporte 1000 peticiones por segundo.
   * **Imagen:** Un velocímetro.
   * **Interactividad sugerida:** Clic en el velocímetro para ver una simulación de la carga en tiempo real.
5. **Pruebas de Seguridad (Security Testing)**
   * **Descripción:** Se encargan de encontrar vulnerabilidades o riesgos de seguridad en el software.
   * **Cuándo aplicarlas:** Siempre que trabajes con datos sensibles, como sistemas de pagos o información personal de usuarios.
   * **Ejemplo:** Probar si una aplicación web es vulnerable a ataques de SQL Injection.
   * **Imagen:** Un candado cerrado.
   * **Interactividad sugerida:** Al hacer clic en el candado, se muestra una explicación interactiva sobre cómo prevenir estos ataques.
6. **Pruebas de Usabilidad (Usability Testing)**
   * **Descripción:** Evalúan qué tan fácil es para los usuarios interactuar con el sistema.
   * **Cuándo aplicarlas:** Durante el desarrollo de interfaces de usuario o después de haber terminado el diseño de la UX/UI.
   * **Ejemplo:** Probar cuántos clics le toma a un usuario novato completar un formulario.
   * **Imagen:** Una mano navegando en una pantalla táctil.
   * **Interactividad sugerida:** Al hacer clic, se muestra un video corto simulando la experiencia de un usuario con una mala interfaz.