### Pruebas de UX/UI

1. **Evaluación de la Experiencia del Usuario (UX):**
   * Facilidad de navegación y coherencia en el diseño.
   * Claridad y utilidad de los mensajes de error.
   * Comprobación de los flujos de usuario para realizar tareas críticas (por ejemplo, encontrar una camiseta, personalizarla y comprarla).
   * Accesibilidad para usuarios con discapacidades.
   * Adaptabilidad a diferentes tamaños de pantalla y dispositivos.
   * Tiempos de carga y rendimiento en diferentes condiciones de red.
2. **Pruebas de la Interfaz de Usuario (UI):**
   * Alineación y consistencia visual con el diseño gráfico.
   * Funcionalidad de todos los elementos interactivos (botones, enlaces, formularios).
   * Validación de formularios y gestión de errores de entrada.
   * Pruebas de los estados de los elementos UI (hover, activo, desactivado).
   * Resolución y adaptabilidad de imágenes y otros medios.
   * Coherencia y legibilidad de la tipografía.

### Pruebas de API

1. **Funcionalidad:**
   * Verificación de que todos los endpoints funcionan según lo esperado.
   * Pruebas de los métodos de la API (GET, POST, PUT, DELETE) para asegurar que manejan correctamente las solicitudes y respuestas.
   * Validación de los códigos de respuesta para diferentes escenarios.
2. **Seguridad:**
   * Implementación de autenticación y autorización.
   * Pruebas de inyección SQL y XSS para asegurar la protección de los datos.
   * Verificación de la encriptación de datos sensibles.
3. **Rendimiento:**
   * Pruebas de carga para evaluar cómo maneja el sistema un alto número de solicitudes.
   * Pruebas de estrés para determinar los límites de la infraestructura de la aplicación.
   * Análisis del tiempo de respuesta de la API bajo diferentes condiciones.
4. **Interoperabilidad:**
   * Comprobación de la integración con otros servicios (por ejemplo, pasarelas de pago, servicios de envío).
   * Validación del formato de los datos de respuesta (generalmente JSON o XML) para asegurar que se ajustan a las especificaciones.
5. **Versionado:**
   * Asegurar que las actualizaciones de la API no rompan la funcionalidad existente.
   * Pruebas de compatibilidad hacia atrás con versiones anteriores de la API.

Este listado ofrece una visión amplia de los aspectos cruciales que se deben cubrir en las pruebas de UX/UI y API para una tienda virtual de camisetas personalizadas. La implementación detallada de estas pruebas dependerá de las especificaciones técnicas y de diseño de tu proyecto.

User

para cada uno de los puntos agrega los tipos de pruebas funcionales y no funcionales que consideras que deberían realizarse

Para cada uno de los elementos mencionados anteriormente, voy a detallar los tipos de pruebas funcionales y no funcionales que se deben considerar para garantizar una cobertura de pruebas exhaustiva en un plan de pruebas para una tienda virtual de camisetas personalizadas:

### Pruebas de UX/UI

#### Evaluación de la Experiencia del Usuario (UX):

* **Funcionales:**
  + Pruebas de escenarios de usuario para validar flujos completos de tareas específicas.
  + Pruebas de campo de formularios para verificar la captura y validación correctas de datos.
* **No Funcionales:**
  + Pruebas de usabilidad para evaluar la facilidad de uso y la satisfacción del usuario.
  + Pruebas de accesibilidad para garantizar la conformidad con estándares como WCAG.

#### Pruebas de la Interfaz de Usuario (UI):

* **Funcionales:**
  + Pruebas de componentes individuales (botones, enlaces, campos de texto) para validar su funcionamiento.
  + Validación de formularios y gestión de errores.
* **No Funcionales:**
  + Pruebas de compatibilidad en diferentes navegadores y dispositivos.
  + Pruebas de rendimiento para evaluar tiempos de carga de la página.

### Pruebas de API

#### Funcionalidad:

* **Funcionales:**
  + Pruebas de contrato para asegurar que la API cumple con su especificación documentada.
  + Pruebas de integración para validar las interacciones entre diferentes partes del sistema y servicios externos.
* **No Funcionales:**
  + No aplica directamente, ya que la mayoría de las pruebas de funcionalidad son inherentemente funcionales.

#### Seguridad:

* **Funcionales:**
  + Pruebas de penetración para identificar vulnerabilidades explotables en la aplicación.
* **No Funcionales:**
  + Análisis de seguridad para evaluar la conformidad con las mejores prácticas y estándares de seguridad.

#### Rendimiento:

* **Funcionales:**
  + No aplica directamente, ya que las pruebas de rendimiento son generalmente no funcionales.
* **No Funcionales:**
  + Pruebas de carga para verificar el comportamiento bajo condiciones de uso elevado.
  + Pruebas de estrés para determinar los puntos de falla bajo cargas extremas.

#### Interoperabilidad:

* **Funcionales:**
  + Pruebas de integración para asegurar que la API funciona correctamente con otros sistemas y servicios.
* **No Funcionales:**
  + Pruebas de compatibilidad para verificar la interacción con diferentes versiones de sistemas y servicios externos.

#### Versionado:

* **Funcionales:**
  + Pruebas de regresión para asegurar que nuevas versiones no introduzcan errores en funcionalidades existentes.
* **No Funcionales:**
  + Pruebas de mantenibilidad para evaluar la facilidad con la que se pueden aplicar cambios y actualizaciones a la API.

Esta división muestra cómo se pueden aplicar pruebas tanto funcionales como no funcionales en diferentes áreas de enfoque para un plan de pruebas exhaustivo, garantizando así que tanto la funcionalidad como la calidad del software cumplen con los estándares y expectativas.