EXAMEN DE CASOS DE PRUEBA - QA (Nivel

Intermedio)

Duración: 30 minutos

Recursos permitidos: Documentación y evidencias de prueba.

Entrega: subir el documento final y evidencias al repositorio GitHub asignado.

# Instrucciones:

El examen evalúa tu capacidad para diseñar, documentar y evidenciar casos de prueba sobre flujos funcionales, alternos y de error. Basado en escenarios de plataformas como Saga Ripley, YouTube y Facebook. Deberás redactar los casos de prueba con pasos, datos, resultados esperados y adjuntar evidencia (capturas, reportes, logs, etc.). El trabajo final debe subirse a un repositorio GitHub, incluyendo la carpeta de evidencias.

# Preguntas

1. [Saga Ripley] Diseña un caso de prueba de REGRESIÓN para verificar que, tras una actualización del módulo de pagos, el flujo de compra con cupón de descuento sigue funcionando correctamente. Incluye: flujo feliz, flujo alterno (cupón expirado) y flujo de error (fallo en pasarela).

**Objetivo:** Verificar que, después de una actualización del módulo de pagos, el flujo de compra con cupón de descuento sigue funcionando correctamente, incluyendo flujo feliz, alterno (cupón expirado) y error (fallo en pasarela).

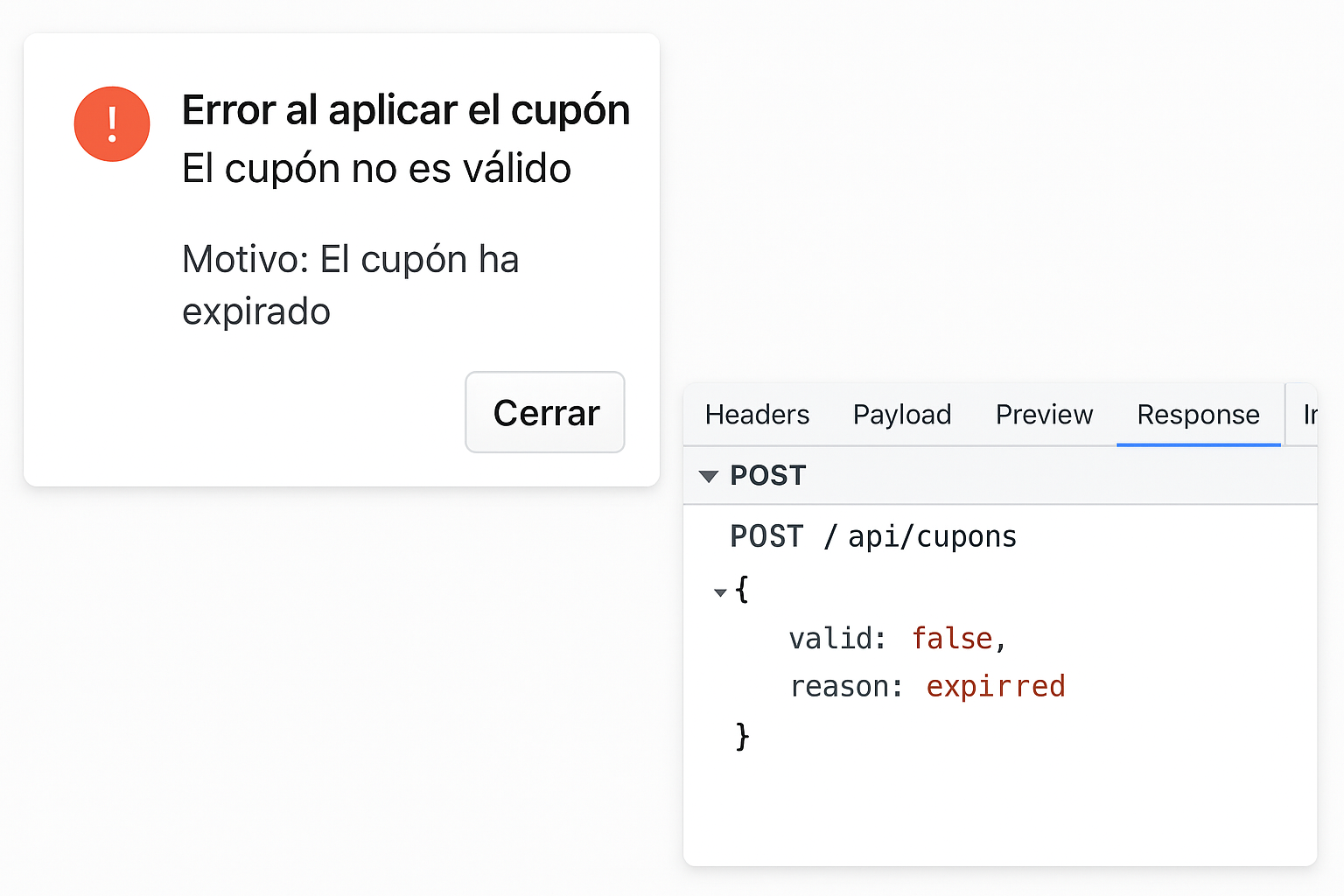
### 1.1 Flujo feliz — Cupón válido aplicado y pago exitoso

* **Precondiciones:**
  + Usuario registrado con cuenta activa: user: qa.regresion+sr@correo.com / pass: \*\*\*\*\*\*
  + Producto en stock: SKU: SR-{{SKU\_VÁLIDO}} (por ejemplo, zapatillas)
  + Cupón válido: RIPLEY10 (10% de descuento, vigente al {{fecha}})
  + Métodos de pago configurados: tarjeta VISA válida de prueba (BIN válido)
* **Datos de prueba:**
  + Cantidad: 1
  + Dirección: Av. QA 123, Piura
  + Tarjeta: 4111 1111 1111 1111, venc: 12/28, CVV: 123, titular: QA Tester
* **Pasos:**
  + Iniciar sesión.
  + Buscar producto por SKU y abrir PDP (Product Detail Page).
  + Añadir al carrito.
  + Ir al carrito y aplicar cupón RIPLEY10.
  + Verificar descuento aplicado en subtotal.
  + Continuar a checkout → completar envío → seleccionar VISA.
  + Confirmar el pago.
* **Resultado esperado:**
  + El precio con descuento se refleja correctamente (10% sobre subtotal elegible).
  + El pedido se genera con estado **“Pago Aprobado”** y número de orden ORD-{{id}}.
  + Se muestra página de confirmación con resumen: producto, cupón aplicado, total final, dirección y método de pago.
* **Evidencias:**

### 

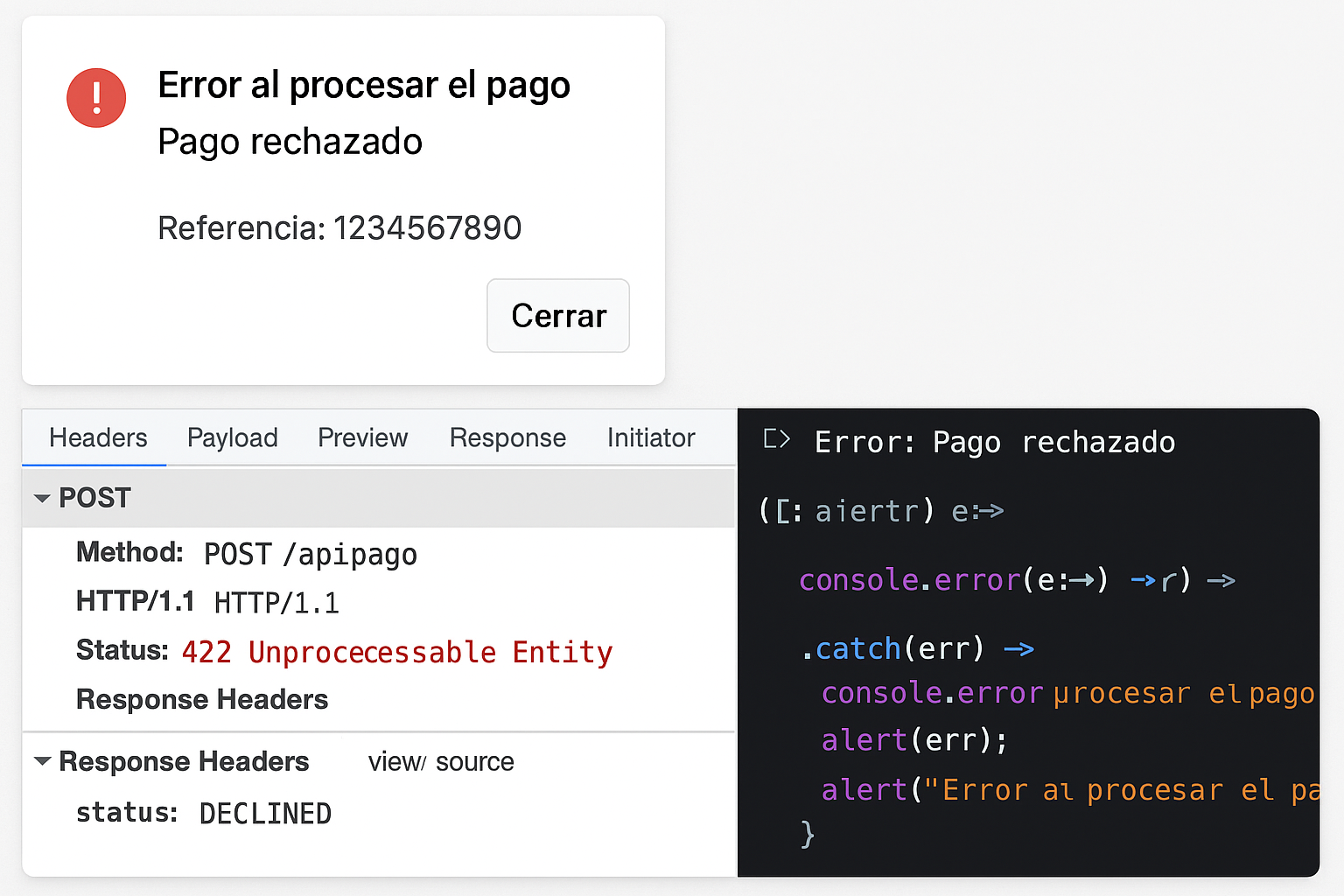
### 1.2 Flujo alterno — Cupón **expirado**

* **Precondiciones:**
  + Cupón expirado: RIPLEY10-OLD (fecha fin < hoy).
* **Pasos:**
  + Repetir pasos 1–3 del flujo feliz.
  + En carrito, aplicar RIPLEY10-OLD.
* **Resultado esperado:**
  + El sistema muestra mensaje claro: **“Cupón expirado o no vigente”**.
  + No se modifica el subtotal ni se aplica descuento.
  + El usuario puede **eliminar el cupón** y continuar con compra **sin cupón**.
* **Evidencias:**



### 1.3 Flujo de **error** — Falla en pasarela de pago

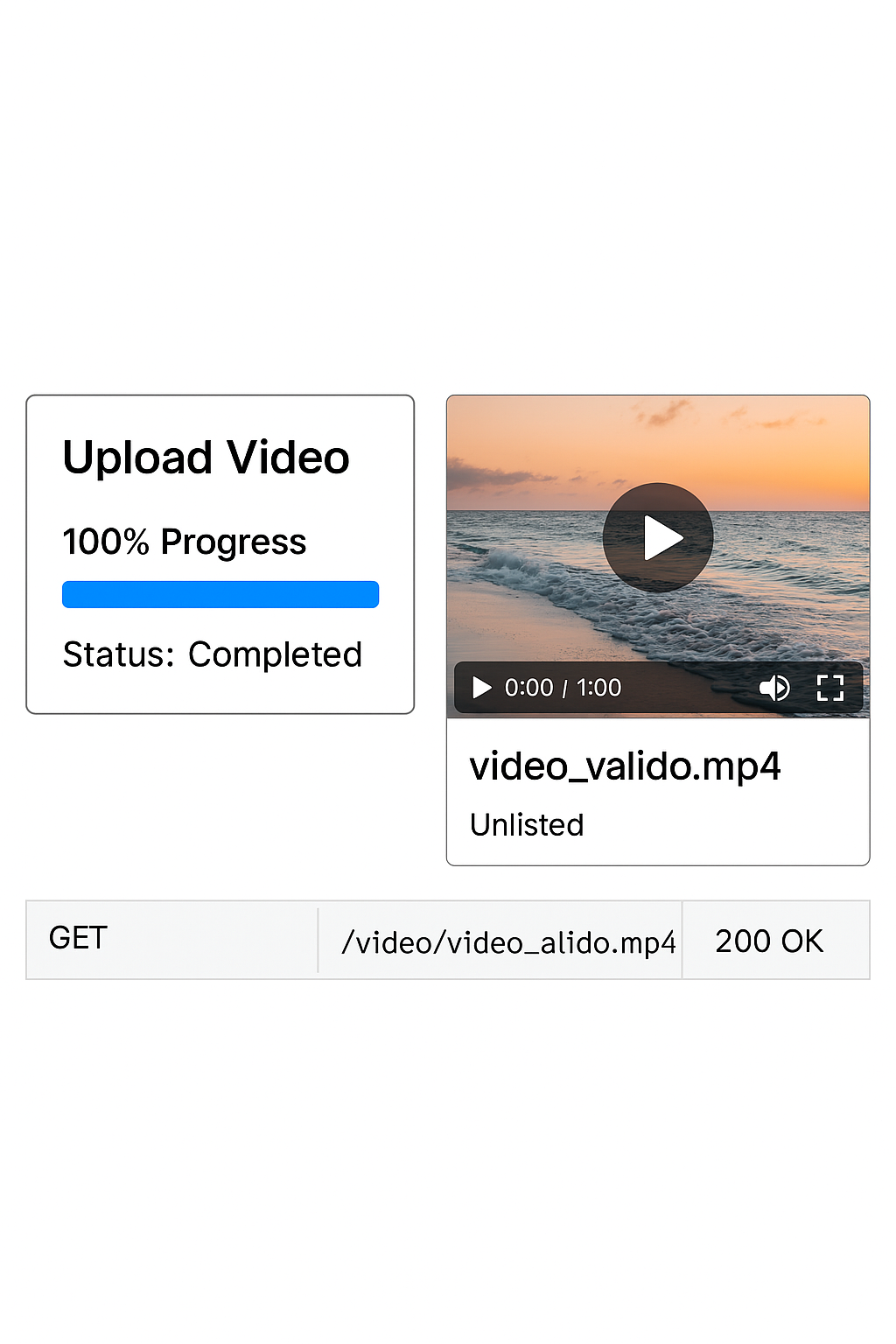
* **Precondiciones:**
  + Pasarela simulada en modo fallo (p. ej., devuelve HTTP 502 o DECLINED).
* **Pasos:**
  + Repetir flujo feliz hasta paso 6.
  + Confirmar pago.
* **Resultado esperado:**
  + Se muestra mensaje **no técnico y claro**: “No pudimos procesar tu pago. Intenta nuevamente o usa otro método.”
  + **No** se genera orden final (sin ORD-…) o se queda en estado **“Pago Rechazado/Pendiente”** sin confirmar.
  + Se preserva el carrito para reintentar.
* **Evidencias:**



1. [YouTube] Crea un conjunto de tres casos de prueba para el proceso de subida de video: (a) Flujo feliz: video válido subido correctamente. (b) Flujo alterno: error al subir por conexión inestable. (c) Flujo de error: archivo no compatible. Incluye resultados esperados y evidencias.

### 2.1 Flujo feliz — Video válido subido correctamente

* **Precondiciones:**
  + Cuenta con permisos para subir.
  + Archivo válido: video\_valido.mp4 (H.264/AAC), duración 00:01:00, tamaño < 1 GB.
* **Pasos:**
  + Click en **Crear** → **Subir video**.
  + Seleccionar video\_valido.mp4.
  + Completar título, descripción, miniatura, etiquetas y visibilidad (“No listado”).
  + Esperar procesamiento → guardar.
* **Resultado esperado:**
  + Progreso sube a 100% sin errores.
  + Estado de procesamiento: **Completado** (SD/HD disponibles).
  + URL final accesible (status 200, reproduce).
* **Evidencias:**



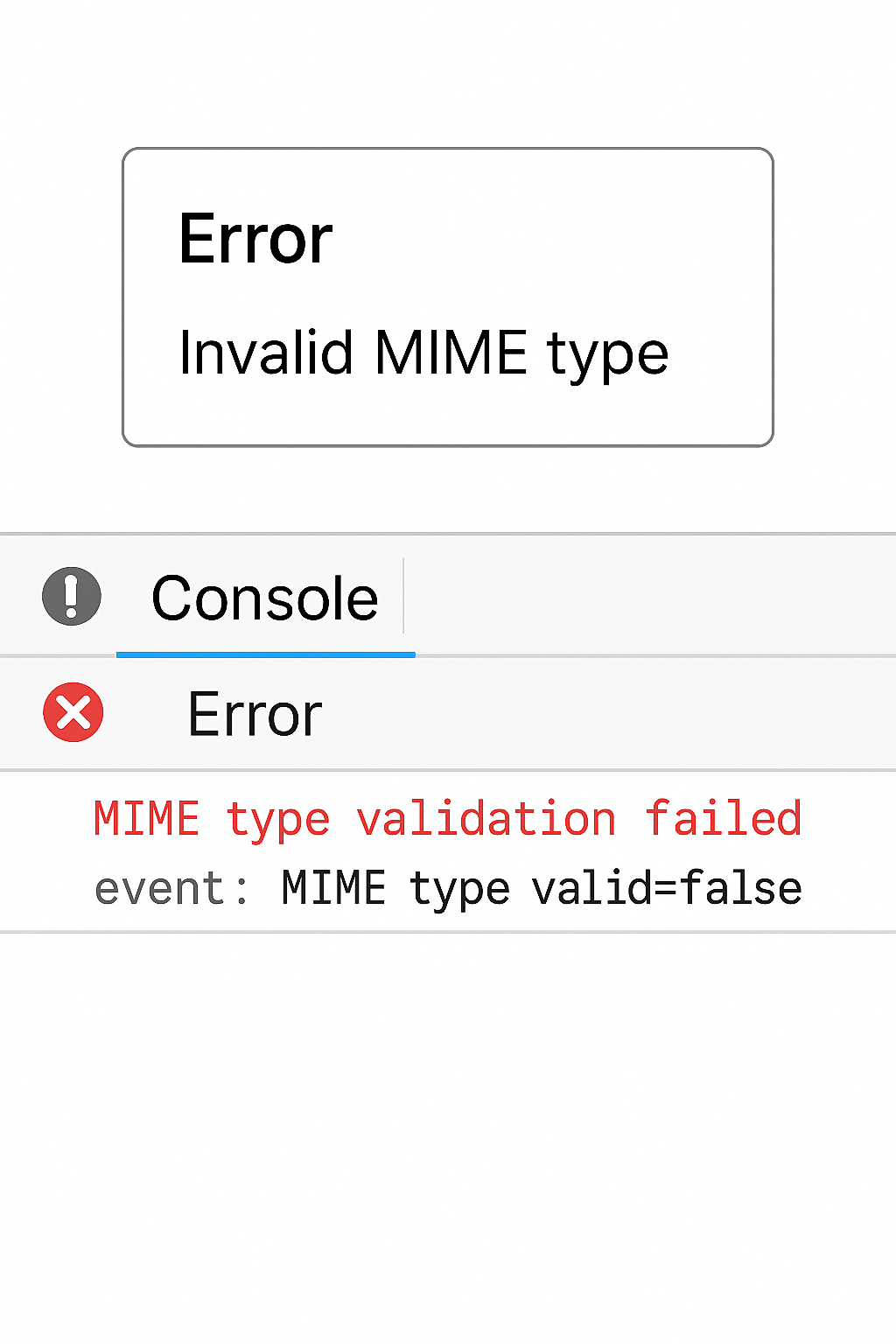
### 2.2 Flujo alterno — Error por **conexión inestable**

* **Precondiciones:**
  + Simular limitación/red intermitente (p. ej., throttling o desconexiones).
* **Pasos:**
  + Iniciar subida de video\_valido.mp4.
  + Interrumpir conexión por 30–60 s y reactivar.
* **Resultado esperado:**
  + La UI muestra **reintentos** o mensaje de conexión.
  + Al restablecer, **retoma** desde el último chunk (reanudación) o permite **reintentar** sin perder metadatos.
* **Evidencias:**

### 

### 2.3 Flujo de error — Archivo **no compatible**

* **Precondiciones:**
  + Archivo con formato no soportado: video\_no\_valido.xyz.
* **Pasos:**
  + Intentar subir video\_no\_valido.xyz.
* **Resultado esperado:**
  + Mensaje claro: “Formato no compatible. Formatos admitidos: MP4, MOV, AVI, …”
  + Subida cancelada, botón para **descartar** o **subir otro archivo**.
* **Evidencias:**



1. [Facebook] Diseña un caso de prueba exploratorio para verificar la funcionalidad de comentarios con texto largo, emojis y menciones. Incluye posibles comportamientos esperados y pasos exploratorios.

**Carta de misión:** Evaluar la robustez del cuadro de comentarios en posts: límites de caracteres, manejo de Unicode (emojis), autocompletado de menciones y visualización en diferentes vistas (feed, modal, móvil).

* **Áreas a cubrir (charlas exploratorias):**
  1. **Longitud:** publicar comentarios de 1, 140, 500, 8,000 y > límite (esperar bloqueo/contador).
  2. **Emojis/Unicode:** insertar emojis simples (😊), combinados (🇵🇪), tonos de piel, ZWJ (👨‍👩‍👧‍👦), y texto RTL/LTR mixto.
  3. **Menciones:** @Nombre Apellido con autocompletar; verificar resaltado, enlace, notificación (si accesible).
  4. **Edición/Borrado:** editar comentario largo con emojis; borrar y confirmar.
  5. **Pegado/Teclado:** pegar texto muy largo desde portapapeles; atajos (Enter = enviar, Shift+Enter = nueva línea).
  6. **Accesibilidad/Responsivo:** navegación con teclado, lectores de pantalla (roles ARIA), vista móvil.
* **Pasos exploratorios sugeridos:**
  1. Abrir un post público de prueba → escribir comentario con 7,500+ caracteres incluyendo emojis mixtos y menciones a 2 contactos de prueba.
  2. Verificar contador (si existe) y conducta al sobrepasar límite (deshabilitar botón, mensaje).
  3. Publicar, refrescar, revisar render (saltos de línea, emojis, menciones en azul).
  4. Editar el comentario, agregar más emojis y otra mención; guardar.
  5. Eliminar comentario; confirmar se oculta o elimina según permisos.
* **Comportamientos esperados:**
  1. El input bloquea o advierte al exceder límite.
  2. Emojis se renderizan correctamente .
  3. Menciones se convierten en enlaces clicables.
  4. Edición mantiene formato; borrado requiere confirmación.
  5. Sin desbordes visuales en móvil/desktop, sin cortes de palabras raros.
* **Evidencias:**
  1. .

1. Reporta un BUG, una INCIDENCIA y un ERROR encontrados durante tus pruebas. Usa formato estándar (título, descripción, pasos para reproducir, resultado actual, resultado esperado, severidad y prioridad).

### 4.1 BUG

* **Título:** Cupón válido no aplica descuento en carrito.
* **Descripción:** Tras la actualización del módulo de pagos, al aplicar el cupón RIPLEY10 el carrito no refleja el 10% de descuento aunque la API de cupones responde valid=true.
* **Pasos para reproducir:**
  1. Iniciar sesión con qa.regresion+sr@correo.com.
  2. Agregar SKU SR-{{SKU\_VÁLIDO}} al carrito.
  3. Aplicar cupón RIPLEY10.
* **Resultado actual:** Subtotal y total no cambian; mensaje “Cupón aplicado” aparece sin impacto en montos.
* **Resultado esperado:** Descuento 10% aplicado al subtotal elegible y total actualizado.
* **Evidencia:**
* **Severidad:** Alta (afecta precios/ingresos). **Prioridad:** Alta (post-release).

### 4.2 INCIDENCIA

* **Título:** Subida intermitente por pérdida de conectividad.
* **Descripción:** Usuarios reportan subidas fallidas en ventana de 10 minutos por intermitencia ISP.
* **Impacto:** Retrasos y reintentos múltiples; no hay pérdida permanente de datos.
* **Evidencia:**
* **Severidad:** Media. **Prioridad:** Media (monitoreo y mitigación).

### 4.3 ERROR

* **Título:** Mensaje de validación no localizado al exceder límite de caracteres
* **Descripción:** Al exceder límite, el UI muestra “String length exceeded” en inglés.
* **Pasos:** Escribir >{{LÍMITE}} caracteres en comentario y presionar Enter.
* **Resultado actual:** Mensaje en inglés.
* **Resultado esperado:** Mensaje localizado al español: “Superaste el límite de caracteres permitido.”
* **Evidencia:**
* **Severidad:** Baja. **Prioridad:** Baja (UI/UX y localización).

1. Explica cómo subirías tus casos y evidencias a un repositorio de GitHub. Incluye pasos, estructura de carpetas y convenciones de nombres recomendadas.

**Rúbrica de Evaluación**

|  |  |  |
| --- | --- | --- |
| **Criterio** | **Descripción** | **Puntaje Máximo** |
| Estructura de casos de prueCbasos | bien redactados con pasos, datos y resultados espera | dos. 20 |
| Cobertura de flujos | Incluye flujo feliz, alterno y de error en cada escenario. | 20 |
| Reporte de defectosIdentifica | correctamente bug, incidencia y error con formato ade | cuado.15 |
| Evidencias Inclu | ye capturas, reportes o logs que respalden la ejecució | n. 15 |
| Uso de GitHub Reposito | rio ordenado, con estructura clara y documentación RE | ADME1. 5 |
| Presentación general | Claridad, ortografía, formato y orden en el documento. | 15 |
| TOTAL |  | 100 |