

Contexto del Proyecto: Se está desarrollando una aplicación móvil y web para promover el turismo en Tarija, permitiendo a los turistas descubrir lugares poco convencionales ("lugares ocultos"), ver ubicaciones en mapas y compartir sus experiencias.

Tipo de Prueba: Prueba de Sistema (Caja Negra). Se evaluó el comportamiento externo de la aplicación (Frontend interactuando con la API REST) sin necesidad de inspeccionar el código fuente interno de los controladores o la base de datos. El objetivo es validar la experiencia final del turista.

Herramienta Utilizada: TestSprite (Agente de IA). A diferencia de las herramientas tradicionales donde se escribe código línea por línea, se utilizó TestSprite, un agente autónomo de Inteligencia Artificial. Esta herramienta explora la aplicación, genera los casos de prueba automáticamente basándose en lenguaje natural y se auto-repara si la interfaz gráfica cambia.

Escenario de Prueba (Flujo Crítico)

El objetivo fue validar el flujo completo de "**Publicación de Reseña y Calificación**" en un atractivo turístico, funcionalidad clave para generar confianza en la comunidad de usuarios.

Pasos de la Prueba Ejecutada por el Agente (TestSprite):

1. **Autenticación:** El agente inicia sesión como un usuario "Turista" registrado.
2. **Navegación:** Busca el atractivo turístico "Cascadas de Coimata" en el listado.
3. **Interacción:** Hace clic en el botón "Escribir Reseña".
4. **Entrada de Datos:** Selecciona 5 estrellas, escribe el comentario "Excelente lugar para visitar en familia" y adjunta una foto de prueba.
5. **Transacción:** Presiona el botón "Publicar".
6. **Validación (Resultado Esperado):**
 - El sistema muestra un mensaje (Toast): "¡Gracias por tu opinión!".
 - La nueva reseña aparece visible inmediatamente en el perfil del lugar.
 - **(Validación Negativa):** El agente intenta enviar una reseña vacía (sin estrellas o texto) y verifica que el botón "Publicar" esté deshabilitado o muestre error.

Conclusiones:

- **Evaluación End-to-End:** La prueba validó la integración completa: desde la interfaz de usuario (React/Flutter) hasta la persistencia en la base de datos y la respuesta del servidor.
- **Eficiencia con IA:** Al usar TestSprite, no fue necesario programar selectores CSS complejos (#id, .class) que suelen romperse. La IA entendió visualmente qué botones presionar.
- **Prueba de Regresión:** El agente asegura que las nuevas actualizaciones del mapa o del perfil de usuario no hayan roto la funcionalidad básica de dejar comentarios.

CONFIGURACIÓN EN TESTSPRITE (PROMPT & PLAN):

A diferencia de Cypress, en TestSprite se configura mediante **Instrucciones en Lenguaje Natural** y archivos de configuración YAML/JSON que el agente interpreta.

