

# Testing y publicación para productos de IA, apps y videojuegos

Una guía operativa para convertir una build en una prueba acotada, obtener evidencia reproducible y decidir si está lista para llegar a usuarios reales.

## PRINCIPIO

No puntúes por abrir el producto. Puntúa por producir feedback que permita decidir o actuar.

### CÓMO USARLA

#### Una pasada, un rastro de evidencia

- ☐ Define primero una build, un objetivo y el riesgo que quieres reducir.
- ☐ Marca solo lo comprobado. Añade versión, dispositivo y evidencia a cada hallazgo.
- ☐ Separa bloqueos de publicación, problemas importantes y mejoras opcionales.
- ☐ Repite las pruebas críticas después de cada cambio de modelo, prompt, herramienta o build.

#### Definición de la misión

- ☐ Producto y build identificados con versión, fecha y canal de acceso.
- ☐ Objetivo de la sesión expresado como una pregunta observable.
- ☐ Perfil de usuario, dispositivo, idioma y experiencia relevante definidos.
- ☐ Duración, tareas, datos permitidos y criterio de finalización acordados.
- ☐ Riesgos y acciones prohibidas explicados antes de empezar.

01

# Base común: lo que toda build debe superar

Prueba primero el recorrido completo y sus fronteras. Un resultado útil describe dónde, cómo y con qué impacto se rompe la experiencia.

## Primera experiencia y flujo principal

- ☐ La promesa inicial se entiende sin ayuda externa.
- ☐ Registro, permisos y onboarding piden solo lo necesario.
- ☐ La primera tarea valiosa puede completarse de principio a fin.
- ☐ Estados vacíos, carga, error y éxito explican qué ocurre.
- ☐ Volver atrás, cancelar y recuperar una sesión no destruye trabajo.

## Comportamiento funcional y regresión

- ☐ Entradas válidas, inválidas, incompletas y extremas producen respuestas controladas.
- ☐ Los datos guardados reaparecen correctamente tras cerrar y volver a abrir.
- ☐ Red, proveedor o servicio lento no congela ni duplica acciones.
- ☐ Mensajes, enlaces, exportaciones, pagos y borrados se confirman cuando corresponde.
- ☐ Una corrección no reabre un fallo ya cerrado en el flujo principal.

## Compatibilidad, accesibilidad y localización

- ☐ Móvil, escritorio, orientación, zoom y tamaños de texto mantienen el contenido legible.
- ☐ Teclado, foco, lector de pantalla y contraste permiten completar tareas críticas.
- ☐ Idiomas, fechas, números, moneda y textos largos no rompen la interfaz.
- ☐ Hardware mínimo y recomendado se comportan dentro de límites declarados.

02

# Productos de IA: comportamiento, límites y control

Una interfaz que responde no demuestra fiabilidad. Evalúa el sistema con casos normales, difíciles, incompletos y hostiles.

## Conjunto de evaluación y utilidad

- ☐ Existe un conjunto versionado de casos con entrada, contexto y criterio de éxito.
- ☐ Se mide utilidad, exactitud, completitud, claridad y reconocimiento de incertidumbre.
- ☐ Los fallos reales se convierten en regresiones permanentes sin conservar datos sensibles.
- ☐ Evaluadores automáticos están calibrados contra muestras revisadas por personas.
- ☐ Cambios de modelo, prompt, contexto o proveedor se comparan antes de publicar.

## Seguridad, datos y herramientas

- ☐ Prompt injection directa e indirecta no altera instrucciones privilegiadas.
- ☐ Usuarios, sesiones y organizaciones permanecen separados.
- ☐ Cada herramienta usa el mínimo permiso y valida todos sus argumentos.
- ☐ Publicar, borrar, pagar o enviar requiere confirmación proporcional al riesgo.
- ☐ Logs y trazas ocultan secretos, tokens y datos personales.

## Fallo, fallback y escalado humano

- ☐ Timeouts y reintentos tienen límites y no repiten acciones.
- ☐ Fallback entre modelos o rutas deterministas conserva un resultado seguro.
- ☐ Respuestas mal formadas y límites de cuota generan un mensaje honesto.
- ☐ Latencia y coste se observan por tarea y proveedor.
- ☐ El escalado humano incluye contexto suficiente y evita exponer información innecesaria.

03

# Videojuegos: experiencia, rendimiento y repetición

Además de encontrar bugs, observa si el jugador entiende, disfruta y puede progresar. Registra plataforma, hardware, mando, idioma y experiencia con el género.

## Primera partida y controles

- ☐ Los primeros minutos explican objetivo, movimiento, interacción y peligro.
- ☐ Tutoriales, controles, cámara, sensibilidad y mandos se comportan como se espera.
- ☐ Menús, pausa, guardado, carga y salida protegen el progreso.
- ☐ Texto, subtítulos, audio y señales visuales cubren necesidades diversas.

## Balance, progresión y repetición

- ☐ Dificultad, economía, recompensas y progresión forman una curva comprensible.
- ☐ No hay bloqueos de progreso, estados imposibles ni estrategias dominantes evidentes.
- ☐ La repetición y el multijugador revelan fatiga, latencia, reconexión y motivos para volver.

## Rendimiento y plataforma

- ☐ FPS, tiempos de carga, memoria, batería y temperatura se miden en hardware objetivo.
- ☐ Cuelgues, cierres y artefactos incluyen escena, acción y frecuencia.
- ☐ Resolución, pantallas y canal de instalación no ocultan UI ni impiden actualizar la build.

04

# Evidencia reproducible y decisión de publicación

Un informe debe permitir a otra persona reproducir el resultado sin adivinar. La evidencia explica el contexto; no sustituye una descripción precisa.

## Informe mínimo de bug o hallazgo

- ☐ Título observable: qué falla y dónde.
- ☐ Build, entorno, dispositivo, cuenta de prueba e idioma.
- ☐ Precondiciones y pasos mínimos numerados.
- ☐ Resultado esperado frente a resultado real.
- ☐ Frecuencia, severidad e impacto para el usuario.
- ☐ Captura, vídeo o log redactado cuando aporta contexto y existe consentimiento.

## Puerta de publicación

- ☐ No quedan bloqueos críticos ni riesgos sin responsable y fecha.
- ☐ Privacidad, permisos, retención y eliminación de datos coinciden con el comportamiento real.
- ☐ Ficha, capturas, compatibilidad, soporte y notas de versión describen la build correcta.
- ☐ Instalación, actualización, rollback, monitorización y canal de incidencias están ensayados.
- ☐ La decisión queda registrada: publicar, publicar con condiciones o detener.

## Detente y escala

No uses datos personales reales, credenciales propias ni acciones irreversibles. Detén la sesión si accedes a información ajena, permisos inesperados o un riesgo que excede el alcance acordado.

Límites. Esta checklist mejora la evidencia y la preparación; no garantiza la aprobación de Google Play, App Store, Steam ni otra plataforma. Revisa siempre las reglas actuales del canal de publicación.

Convierte esta checklist en misiones concretas con el generador privado de Beta Lab: [maxbonas.com](https://maxbonas.com)