

Ejercicio Integral de Aptitud Técnica | Semi Senior QA Analyst

General:

- Deberás realizar los 3 desafíos mencionados en este documento:
 - [Desafío 1: Testing en Pushing IT](#)
 - [Desafío 2: Go Rest en Postman](#)
 - [Desafío 3: Aplicación Flutter](#)
 - Deberás entregar los documentos solicitados para cada ejercicio en una misma carpeta de Google Drive con los permisos de lectura públicos.
-

Desafío 1: Testing en Pushing IT

Objetivo

Evaluar tu capacidad para diseñar y ejecutar pruebas funcionales y exploratorias sobre la plataforma [Pushing IT](#).

Escenario

La empresa está evaluando la estabilidad y usabilidad del sitio **Pushing IT**, una plataforma que permite realizar pruebas en un entorno controlado. Queremos asegurarnos de que las funcionalidades principales operan correctamente en distintos escenarios.

Tareas

Diseño de casos de prueba

1. Acceder al sitio [Pushing IT](#) y explorar las funcionalidades principales.
2. Seleccionar una funcionalidad clave (por ejemplo, **Registro de usuario y Login**).
3. Diseñar al menos **5 casos de prueba** con los siguientes detalles:
 - **Título del Bug:** Claro y breve
 - **Navegador y SO**
 - **Prioridad**
 - **Pasos detallados para reproducir**
 - **Resultado esperado vs. Resultado obtenido**
 - **Evidencia visual**
 - **Recomendaciones:** (opcional) Mejora, propuesta o comentario adicional sobre la experiencia UX.

2 Ejecución de pruebas exploratorias

1. Sin seguir un guión estricto, interactuar con la web y buscar posibles errores en funcionalidades clave.
2. Documentar cualquier comportamiento inesperado, bug encontrado o posible mejora de UX.

3 Reporte de Errores

1. Si se encuentran errores, reportarlos en un documento estructurado con los siguientes datos:
 - **Título del bug:** Breve descripción del problema.
 - **Prioridad:** Alta/Media/Baja según impacto.
 - **Pasos para reproducir:** Detallar el flujo exacto.
 - **Evidencia:** Grabacion con <https://jam.dev/> dejar el link.

Entrega

En el formato que vos decidas, deberás entregar lo siguiente:

- ✓ Casos de prueba documentados
- ✓ Reporte de errores (si los encuentra)
- ✓ Conclusiones sobre la estabilidad del sistema

Criterios de evaluación

- ✓ Capacidad de análisis y exploración.
- ✓ Claridad en la documentación de pruebas.
- ✓ Rigor en la ejecución y reporte de errores.

Bonus

Si tienes conocimientos en automatización en Playwright puedes proponer un test automatizado. 

Selecciona una de las funcionalidades testeadas manualmente para recomendar al menos un caso automatizado. Envía el código en un repositorio o por zip.

Desafío 2: Go Rest en Postman

Objetivo

Evaluar la capacidad del analista QA para diseñar y ejecutar pruebas API utilizando Postman, validando los métodos principales (GET, POST, PUT/PATCH, DELETE) sobre la API pública [Go Rest](#)

Escenario

La empresa está evaluando la estabilidad y correcto funcionamiento de la API REST "[Go Rest](#)", verificando que los distintos endpoints funcionen adecuadamente ante diferentes tipos de solicitudes HTTP.

Tareas

En base a la API (Go Rest), debes crear y ejecutar **requests** distribuidas del siguiente modo:

1 3 Requests tipo POST:

- Crear 2 usuarios nuevos utilizando datos generados de forma aleatoria (random).
- Generar un request que no devuelva la respuesta esperada.

2 3 Requests tipo GET:

- 1 request para obtener todos los usuarios disponibles.
- 2 requests para obtener información específica de usuarios que creaste en el paso anterior utilizando su ID.
- Generar un request que no devuelva la respuesta esperada.

3 3 Requests tipo PATCH o PUT:

- Actualizar los datos de los 2 usuarios que generaste en la etapa previa (POST).
- Generar un request que no devuelva la respuesta esperada.

4 3 Requests tipo DELETE:

- Eliminar los 2 usuarios que generaste previamente con los requests POST.
- Generar uno que no devuelva la respuesta esperada.

Consideraciones técnicas:

- Deberás autenticarte en la API utilizando una cuenta de Gmail, GitHub o Microsoft.
- Al iniciar sesión obtendrás un TOKEN que deberás incluir en el HEADER de autorización de cada request como Bearer Token.
- Para cada request, incluí assertions relevantes como por ejemplo el código de estado esperado (ej. Status 200 o 201). Sé creativo respecto a qué cosas probarías. Podes enviar tus comentarios de la razón por la que elegiste las assertions.

Entrega

Para este ejercicio deberás entregar:

✓ La URL pública del workspace o colección Postman con tus pruebas documentadas o Adjunta el archivo JSON correspondiente a la colección exportada desde Postman.

Criterios de evaluación

- ✓ Habilidades de generación de colecciones.
 - ✓ Capacidad de generar pruebas negativas.
 - ✓ Elección de elementos a testear.
-

Desafío 3: Aplicación Flutter

Objetivo

Evaluar tu capacidad para diseñar y ejecutar pruebas funcionales y exploratorias en aplicaciones móviles desarrolladas en Flutter, considerando distintos dispositivos y escenarios de uso, asegurando estabilidad, usabilidad y compatibilidad.

Escenario

La empresa ha desarrollado una aplicación móvil híbrida en Flutter que muestra las capacidades de los widget. Queremos garantizar la calidad en diversos dispositivos y condiciones de uso reales.

App para pruebas

Demo de Syncfusion: La aplicación oficial de demostración de **Syncfusion** ofrece diversas funcionalidades para realizar tus pruebas y está disponible para muchas plataformas. Puedes descargarla desde:

- **Repositorio:** [This repository contains the Syncfusion Flutter UI widgets examples and the guide to use them.](#)
- **Web:** [Flutter Demos | Syncfusion®](#)
- **PlayStore:** <https://apps.apple.com/us/app/syncfusion-flutter-ui-widgets/id1475231341>
- **APK (Android):** [Syncfusion Flutter UI Widgets - Apps on Google Play](#)

Tareas a Realizar

1 Diseño y Documentación de casos de prueba para Mobile:

Selecciona **2 funcionalidades clave** (por ejemplo: Autenticación y Visualización de información en tiempo real).

- Para cada funcionalidad crea al menos 4 casos de prueba detallados, incluyendo:
 - Nombre descriptivo del caso
 - Dispositivo objetivo
 - Precondiciones específicas
 - Pasos precisos a ejecutar
 - Resultado esperado claramente definido
 - Severidad en caso de fallo

2 Pruebas Exploratorias Mobile:

- Realiza una sesión de pruebas exploratorias en al menos dos dispositivos diferentes.
- Enfócate en encontrar bugs de UI, funcionales y problemas específicos del sistema operativo (rendimiento, accesibilidad, adaptación gráfica).
- Documenta hallazgos con evidencia visual (capturas o videos), identificando claramente el dispositivo y la versión del SO.

3 Gestión y reporte de errores Mobile:

En caso de detectar errores, documenta utilizando el siguiente formato estructurado:

- **Título del Bug**

- **Dispositivo y SO**
- **Prioridad**
- **Pasos detallados para reproducir**
- **Resultado esperado vs. Resultado obtenido**
- **Evidencia visual**
- **Recomendaciones:** (opcional) Mejora, propuesta o comentario adicional sobre la experiencia UX.

4 Compatibilidad y Testing Cross-Device:

- Propone una estrategia concisa para realizar pruebas cross-device, garantizando el correcto funcionamiento de la aplicación en diferentes plataformas.
- Sugiere herramientas para la ejecución de estas pruebas y **justifica tu elección en función de tu experiencia previa, explicando detalladamente por qué consideras que son la mejor opción.**
- Además, describe brevemente cómo validarías las funcionalidades críticas, teniendo en cuenta la fragmentación de versiones de los sistemas operativos y las distintas resoluciones de pantalla. **Justifica tu enfoque con base en tu experiencia, argumentando por qué crees que es la mejor alternativa.**

Bonus

Si logras clonar el proyecto y enviar evidencia que usaste un emulador para las pruebas 🚀

Entrega

En el formato que vos decidas, deberás entregar lo siguiente:

- ✓ Casos de prueba mobile claramente documentados.
- ✓ Resultados de pruebas exploratorias realizadas sobre diferentes dispositivos.
- ✓ Reporte estructurado y detallado de errores encontrados.
- ✓ Justificación de Herramientas utilizadas.
- ✓ Estrategia propuesta para asegurar compatibilidad cross-device.

Criterios de Evaluación

- ✓ Claridad y organización en la documentación.
 - ✓ Profundidad del análisis exploratorio y capacidad para detectar errores relevantes.
 - ✓ Calidad técnica en la definición y documentación de casos específicos para entornos móviles.
 - ✓ Aplicación de experiencia previa en la justificación de la elección de herramientas y validación de funcionalidades críticas
-