

Plan de Pruebas – Formulario de registro de usuarios

1. Identificador del Plan de Pruebas

Código: PP-STUDENT REGISTRATION FORM-001

Versión: 1.0

Fecha: 16/01/2026

Elaborado por:

Jhoan Sebastian Hoyos Parada

2. Introducción

El presente documento define el plan de pruebas funcionales automatizadas para el formulario de registro de usuarios disponible en el sitio de demostración:

<https://demoqa.com/automation-practice-form>

El objetivo de este plan es validar el correcto funcionamiento del formulario, asegurando que los campos, validaciones y combinaciones de datos permitan un registro exitoso cuando se ingresan datos válidos y presenten errores adecuados cuando los datos son inválidos o incompletos.

Este plan se desarrolla como parte de la Kata técnica para el rol de QA Automatizador Junior, utilizando técnicas de prueba de caja negra y automatización con Playwright.

3. Objetivos de Prueba

- Validar que el formulario de registro permita el envío exitoso con datos válidos.
- Verificar las validaciones de campos obligatorios.
- Comprobar el comportamiento del sistema ante datos inválidos.
- Aplicar varias técnicas de prueba:
- Garantizar la estabilidad del formulario ante diferentes combinaciones de entrada.

4. Alcance

Incluye:

- Campos de texto (First Name, Last Name, Email, Mobile).
- Selección de género.
- Selección de hobbies.
- Login y manejo de sesión.
- Selección de estado y ciudad..
- Envío del formulario.
- Mensaje/modal de confirmación.

Excluye:

- Persistencia real de datos en base de datos
- Integración con sistemas externos.
- Reportes administrativos o métricas de usuario.
- Pruebas de rendimiento extremo o estrés.
- Seguridad avanzada (captcha, ataques, etc.).

5. Enfoque de Pruebas

Pruebas funcionales manuales: validación de interfaz, registro de datos y respuestas esperadas.

Pruebas automatizadas (Playwright): flujos críticos de registro de usuarios.

Tipos de pruebas aplicadas:

- *Smoke testing:* validación inicial de disponibilidad del sistema.
- *Pruebas de regresión:* repetición de pruebas tras correcciones.
- *Pruebas automatizadas end-to-end*

6. Técnicas de diseño de pruebas utilizadas

Partición de Equivalencia: *Consiste en dividir los datos de entrada en grupos (válidos e inválidos) que se espera que el sistema trate de la misma manera.*

- Email válido vs email inválido (**CF-001 / CF-003**)
- Mobile Number válido vs inválido (**CF-001 / CF-004**)
- Formulario con campos obligatorios completos vs incompletos (**CF-001 / CF-002**)

Justificación: Se seleccionaron valores representativos de cada grupo para reducir la cantidad de pruebas sin perder cobertura funcional.

Pruebas de Caja Negra: *Se valida el comportamiento del sistema sin considerar la lógica interna o el código fuente.*

- En **todos los casos de prueba**, ya que el formulario se evaluó únicamente desde la perspectiva del usuario final.

Justificación El objetivo fue validar la funcionalidad observable del formulario, independientemente de su implementación interna.

Pruebas Positivas (Happy Path): *Validan que el sistema funcione correctamente cuando se ingresan datos válidos.*

- Registro exitoso (**CF-001**)
- Registro con hobbies seleccionados (**CF-006**)
- Registro sin State y City (**CF-007**)
- Carga exitosa de archivo (**CF-008**)

Justificación: Permiten asegurar que el flujo principal del formulario funciona correctamente bajo condiciones esperadas.

7. Criterios de Aceptación

- El formulario debe enviarse correctamente cuando todos los campos obligatorios contienen datos válidos..
- El sistema debe impedir el envío si faltan campos obligatorios.
- El número de teléfono debe contener exactamente 10 dígitos.
- El email debe cumplir con un formato válido.
- Al enviar el formulario correctamente, debe mostrarse el modal de confirmación.

8. Criterios de Suspensión y Reanudación

Suspensión: si más del **30%** de los casos críticos fallan. Caída o indisponibilidad del sitio DemoQA

Reanudación: cuando los defectos críticos y bloqueantes hayan sido corregidos y validados en re-pruebas.

9. Entregables de Prueba

- Plan de pruebas (este documento).
- Casos de prueba documentados.
- Scripts automatizados en Playwright.
- Evidencias de ejecución (screenshots).
- Repositorio público en GitHub.

10. Roles y Responsabilidades

ROL	Responsabilidad Principal
Scrum Master Seguros Bolívar	Facilita el proceso
QA Tester – Jhoan Sebastian Hoyos	Ejecuta pruebas automáticas del formulario de registro de usuarios. Reporta defectos encontrados y valida su corrección durante las re-pruebas. Garantiza que los resultados sean coherentes con los criterios de aceptación definidos. Desarrolla y mantiene scripts de automatización en Playwright. Ejecuta pruebas de regresión y smoke tests automatizados.
Product Owner (Equipo evaluador)	Define requerimientos y criterios

11. Plan de Ambiente de Pruebas

Componente	Detalle Técnico
Servidor QA	Windows
Editor	Visual Studio Code
Frontend	interfaz DemoQA
Backend	TypeScript
Navegadores	Google Chrome, Mozilla Firefox
Herramientas	Playwright, GitHub, TypeScript.
Conectividad	Internet estable (mínimo 10 Mbps)

12. Gestión de Configuración

- Uso de **GitHub** para versionamiento de scripts Playwright.
- La gestión de configuración se realiza mediante un sistema de control de versiones Git, utilizando la rama “main” como rama principal y línea base del proyecto.
- Etiquetas por versión (v 1, 1.1, 2..., etc.).

13. Gestión de Defectos

Los defectos se registran en **dentro del repositorio (README o issues)**, con el flujo:

Descripción → Pasos para reproducir → Resultado esperado → Evidencia (screenshot).

Clasificación por **severidad**:

- Crítico (bloquea ejecución)
- Alto (afecta flujo principal)
- Medio (errores visuales o de datos)
- Bajo (cosméticos)

Se sugiere que cada defecto debe contener: ID, descripción, pasos para reproducir, evidencia y responsable asignado.

14. Métricas de Prueba estándar.

- % de casos ejecutados = $(\text{Ejecutados} / \text{Totales}) \times 100$
- % de casos exitosos = $(\text{Exitosos} / \text{Ejecutados}) \times 100$
- Tasa de defectos = $(\text{Defectos reportados} / \text{Casos ejecutados})$
- Promedio de tiempo de resolución = $(\text{Suma tiempos resolución} / \text{Total defectos})$
- Cobertura de pruebas = $(\text{Casos diseñados} / \text{Requerimientos totales}) \times 100$

15. Cronograma

Semana	Actividad
Dia 1	Análisis de requerimientos y diseño de casos.
Dia 2	Ejecución manual de pruebas funcionales.
Dia 3	Desarrollo y ejecución de automatización Playwright.
Dia 4	Entrega de informe final.

16. Análisis del formulario de registro – DemoQA

El formulario de registro del sitio **DemoQA (Student Registration Form)** es un formulario web interactivo diseñado para simular el proceso de registro de un usuario, permitiendo la captura de diferentes tipos de datos y la validación de múltiples comportamientos funcionales.

El formulario incluye **campos obligatorios y opcionales**, así como distintos tipos de controles de entrada, tales como campos de texto, botones de opción (radio buttons), casillas de verificación (checkboxes), selectores dinámicos y carga de archivos.

Campos relevantes del formulario

- **Campos obligatorios:**

- First Name
- Last Name
- Gender
- Mobile Number

- **Camposopcionales:**

- Email
- Date of Birth
- Subjects
- Hobbies
- Picture
- State y City

Algunos campos presentan **validaciones de formato**, como el correo electrónico y el número de teléfono móvil, el cual debe contener únicamente valores numéricos y una longitud específica.

Comportamiento funcional

El formulario permite el envío exitoso únicamente cuando los campos obligatorios son diligenciados correctamente. En caso contrario, el sistema impide el envío sin mostrar mensajes de error explícitos, lo que obliga a validar el comportamiento mediante la ausencia o presencia del modal de confirmación.

Al completar el formulario correctamente, el sistema muestra un **modal de confirmación** con el resumen de la información ingresada, el cual sirve como referencia principal para validar el resultado de la operación.

Consideraciones para pruebas

- Algunos elementos del formulario son **dinámicos** y requieren interacciones específicas (por ejemplo, clic sobre etiquetas o uso de teclado).
- El formulario presenta **comportamientos inconsistentes bajo ejecución concurrente o en diferentes navegadores**, lo que debe considerarse al diseñar y ejecutar pruebas automatizadas.
- El análisis se enfoca en validar la funcionalidad del formulario desde la perspectiva del usuario final.

17. Casos de Prueba Funcionales

CF-001	Registro exitoso con datos válidos	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al sitio web disponible.	
Precondición	Acceso al sitio https://demoga.com/automation-practice-form	
Descripción	Validar que el formulario permita el registro exitoso cuando se ingresan datos válidos en todos los campos obligatorios.	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio.
	1	Completar todos los campos obligatorios con datos válidos
	2	Seleccionar género, hobbies, estado y ciudad
	3	Presionar el botón Submit
Postcondición	Se muestra el modal de confirmación con la información ingresada.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	El formulario se envía correctamente y no se presentan mensajes de error.	

CF-002	Envío del formulario con campos obligatorios vacíos	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario cargado	
Descripción	Verificar que el sistema impida el envío si no se completan los campos obligatorios	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	No completar los campos obligatorios.
	2	Presionar Submit
	3	
Postcondición	El formulario no se envía y se resaltan los campos obligatorios.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	No se muestra el modal de confirmación.	

CF-003	Validación de formato de correo electrónico inválido	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario disponible	
Descripción	Validar que el sistema no permita el registro con un email inválido	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Ingresar un email sin formato válido
	2	Completar los demás campos obligatorios
	3	Presionar Submit
Postcondición	El formulario no se envía y el email es marcado como inválido.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	No se genera confirmación de registro.	

CF-004	Validación de número de teléfono con menos de 10 dígitos	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario cargado	
Descripción	Verificar que el campo teléfono valide la longitud mínima requerida	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Ingresar un número con menos de 10 dígitos
	2	Completar los demás campos
	3	Presionar Submit
Postcondición	El formulario no se envía.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	El campo teléfono no permite completar el flujo.	

CF-005	Registro con selección de hobbies	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario disponible	
Descripción	Verificar que el formulario se envíe correctamente con selección de hobbies	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Seleccionar uno o más hobbies
	2	Completar los demás campos y enviar
	3	Presionar Submit
Postcondición	El registro se completa exitosamente.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	Los hobbies seleccionados aparecen en el modal de confirmación.	

CF-006	Registro sin selección de género	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario cargado	
Descripción	Validar que el sistema requiera la selección de género	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Completar campos obligatorios sin seleccionar género
	2	Presionar Submit
	3	
Postcondición	El formulario no se envía.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	El sistema impide continuar sin género.	

CF-007	Registro sin selección de State y City	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Formulario cargado	
Descripción	Validar que State y City no sean obligatorios	
Secuencia normal	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Completar todos los campos obligatorios
	2	No seleccionar State ni City
	3	Presionar Submit
Postcondición	El formulario se envía correctamente.	
Excepciones	PASO	ACCIÓN
	N/A	
Comentarios	El formulario no bloquea el envío por estos campos.	

CF-008	Carga de archivo en el campo Picture	
Versión	1.0 (16/01/2026)	
Dependencias	Acceso al formulario	
Precondición	Ingresar a "New user"	
Descripción	PASO	ACCIÓN
	0	Ingresar al sitio
	1	Completar los campos obligatorios
	2	Seleccionar un archivo válido en el campo Picture
	3	Presionar Submit
Postcondición	El formulario se envía correctamente y el nombre del archivo cargado se muestra en el modal de confirmación.	
Excepciones	PASO	ACCIÓN
	2	Seleccionar un archivo no válido o cancelar la selección
Comentarios	El formulario no presenta errores y permite continuar sin archivo	

18. Riesgos y Contingencias

Riesgo	Impacto	Probabilidad	Plan de Mitigación
Caída del entorno QA o sitio DemoQA inactivo	Alto	Media	Ejecutar pruebas en ambiente espejo o reprogramar ejecución.
Retraso en ejecución automatizada	Medio	Media	Priorización de smoke tests para validación mínima.

19. Plan de Comunicación

- Reporte a los correos sugeridos:** Proyecto de la Kata finalizada.
- Reunión de sustentación:** seguimiento con QA Lead y equipo de selección.
- Canales:** correo corporativo.

20. Lecciones Aprendidas

- La estabilidad del sistema bajo prueba impacta directamente la automatización
- La estabilidad del sistema bajo prueba impacta directamente la automatización
- La ejecución paralela no siempre es la mejor opción
- La documentación es tan importante como la automatización

21. Conclusión

Esta Kata permitió validar un formulario web mediante pruebas funcionales y automatizadas, aplicando buenas prácticas de QA y tomando decisiones técnicas basadas en el comportamiento real del sistema bajo prueba. La automatización desarrollada garantiza resultados confiables, evidencias claras y una base sólida para el rol de QA Automatizador Junior.