

MAYO 7, 2024

# DESPEGAR.COM – EXITO.COM Test Plan

Web interactions and Services

Johan Cifuentes

QA Automation Engineer

## Historial del documento

versión	fecha	autor	descripción
V1	7/05/2024	Johan Cifuentes	Creación documento

## Objetivo:

Este plan de prueba tiene la finalidad de asegurar la funcionalidad de la página web de <https://www.despegar.com.co/> como el apartado de vuelos, hoteles y alojamiento. También tiene la finalidad de probar un flujo completo de compra de la página <https://www.exito.com/> Además, se testearán los servicios de API que ofrece la pagina <https://developer.marvel.com/documentation/apireults>

## Descripción página web despegar:

- Vuelos
  - Reservar vuelos
  - Cotizar vuelos
  - Mirar vuelos

## Descripción página web éxito.com:

- Compra de producto
  - Seleccionar aleatoriamente varios productos de interés
  - Llenar formularios
  - Generar orden de compra

## Descripción de las APIs:

- Leer (GET) servicio
  - Leer comics
  - Leer personajes
  - Leer creadores
  - Leer eventos
  - Leer series
  - Leer historias

## Alcance:

Alcance para el plan de pruebas de despegar.com

### Página web despegar:

- En alcance:
  - Reserva de vuelos flujo completo
- Fuera de alcance:
  - Todo lo que no está en alcance
  - Pruebas no funcionales

### Página web éxito:

- En alcance:
  - Agregar varios productos al carrito
  - Llenar formularios de confirmación
  - Generar orden de compra
- Fuera de alcance:
  - Todo lo que no está en alcance
  - Pruebas no funcionales

### Servicios API:

- En alcance:
  - Consulta de todos los personajes GET
  - Consulta de comics por identificador GET
- Fuera de alcance:
  - Todo lo que no está en alcance
  - Pruebas no funcionales

## Entorno de pruebas:

- Navegador Chrome
- Navegador Microsoft Edge
- Sistema operativo Windows 10 Pro

## Datos de prueba:

- Datos ingresados manualmente a través de los escenarios Gherkin
- Datos autogenerados utilizando librerías externas como Faker

## Estrategia de pruebas:

### Página web:

- Paso 1: Creación de casos de prueba en lenguaje natural usando Gherkin utilizando pruebas de caja negra y partición de equivalencia
- Paso 2: Creación de los casos de prueba utilizando Java con Selenium y herramientas de automatización
- Paso 3: Automatizar los casos de prueba

### Servicios API:

- Paso 1: Creación de casos de prueba en lenguaje natural usando Gherkin utilizando pruebas de caja negra y partición de equivalencia

- Paso 2: Creación de los casos de prueba utilizando Java con Rest Assured y herramientas de automatización
- Paso 3: Automatizar los casos de prueba

## Criterios de aceptación:

- Automatizar más del 80% de los casos de prueba.
- Pasar exitosamente más del 80% de los casos de prueba automatizados.
- Documentar más del 90% de los errores encontrados en el sistema y abrir el reporte correspondiente, incluyendo información detallada sobre el error.
- Completar la automatización dentro del plazo establecido, que será especificado en días, semanas o meses según lo acordado previamente.

## Herramientas de la prueba:

- Java (lenguaje de programación)
- Selenium (dependencia del del desarrollo)
- Rest Assured (dependencia del desarrollo)
- Postman
- IntelliJ Idea

## Calendario de pruebas

Nro.	Prueba	Fecha	Total, días
1	Creación plan de pruebas	7/05/2024	2
2	Diseño casos de prueba	7/05/2024	2
3	Automatización casos de prueba	7/05/2024	2
4	Reporte de pruebas	7/05/2024	2

## Entregables de la prueba

Los siguientes productos de trabajo serán entregados

Entregable	Descripción
Plan de pruebas	Detalles del objetivo de la prueba, el alcance, la estrategia y los entregables de la prueba
Diseños de caso de prueba	Diseños de caso de prueba realizados en archivo .feature que describe la secuencia de cada prueba
Matriz de riesgo	Matriz que resalta posibles riesgos y como mitigarlos en la fase del proyecto y producto
Automatización de las pruebas	Proyecto en Java con las pruebas automatizadas