

# Reporte pruebas GHOST

## Monkey-Cypress

**Aplicación bajo pruebas:** Ghost.

**Versión:** 4.41.3

**Sistema operativo:** Windows 11.

**Navegador:** Google Chrome.

## Procedimiento:

Para la ejecución de las pruebas se utiliza la herramienta monkey-cypress, la cual ejecuta pruebas aleatorias en dos versiones; un módulo aleatorio completo y uno un poco más inteligente. Para el desarrollo de las pruebas sobre GHOST se pretende abordar la mayor cobertura posible utilizando los dos métodos.

## Descripción de resultados:

Con el fin de garantizar la mayor cobertura se realizan pruebas sobre los módulos de administración y sobre la vista pública. Sin embargo, todas las pantallas de administración tienen protocolos que validan la autenticación del usuario lo cual hace que si el usuario no está autenticado lo redirija a la página de login y debido a que monkey-cypress no tiene forma de suministrar usuario y contraseña, todas los intentos de prueba no pasaron de la página de login. Debido a esto se presenta a continuación la descripción de los resultados de las páginas de login y la página de contenidos.

### Página de Login con monkey-config.json

Las pruebas con esta configuración pasan de forma exitosa, tal como se muestra en la siguiente imagen la cual muestra el resultado obtenido en consola al finalizar las pruebas:

(Results)

Tests:	1
Passing:	1
Failing:	0

---

(Run Finished)

Spec		Tests	Passing	Failing	Pending	Skipped
✓ monkey.js	00:27	1	1	-	-	-
✓ All specs passed!	00:27	1	1	-	-	-

Además, la prueba arrojó un archivo HTML el cual contiene el detalle de los eventos ejecutados y un archivo mp4 con el video de las pruebas. Los archivos mencionados se pueden encontrar en la siguiente ruta:

- [HTML](#)
- [Video](#)

### Página de Login con Smart-monkey-config.json

Las pruebas con esta configuración no se completaron, tal como se muestra en la siguiente imagen la cual muestra el resultado obtenido en consola al finalizar las pruebas:

```

    }
  ]
}
Finished logging
[2022-05-01T00:26:56.725Z] console.log called
                          Arguments:
                          [
                            {
                              "type": "object",
                              "subtype": "node",
                              "className": "HTMLDivElement",
                              "description": "div.gh-flow",
                              "1) visits App prueba and survives smarter monkeys

```

Esta prueba a diferencia de la primera no produjo ningún resultado, ya que la prueba quedó estática por más de dos horas en el mensaje de la consola mostrado en la imagen anterior.

### Página de Blog con monkey-config.json

Las pruebas con esta configuración pasan de forma exitosa, tal como se muestra en la siguiente imagen la cual muestra el resultado obtenido en consola al finalizar las pruebas:

```
(Results)

Tests:      1
Passing:    1
Failing:    0

=====

(Run Finished)

Spec                                Tests  Passing  Failing  Pending  Skipped
✓ monkey.js                        00:27    1         1        -        -        -
✓ All specs passed!                00:27    1         1        -        -        -
```

Además, la prueba arroja un archivo HTML el cual contiene el detalle de los eventos ejecutados y un archivo mp4 con el video de las pruebas. Los archivos mencionados se pueden encontrar en la siguiente ruta:

- [HTML](#)
- [Video](#)

### Página de Blog con Smart-monkey-config.json

Las pruebas con esta configuración fallaron, tal como se muestra en la siguiente imagen la cual muestra el resultado obtenido en consola al finalizar las pruebas:

```
    }
  ]
}
]
Finished logging
[2022-05-01T00:26:56.725Z] console.log called
Arguments:
[
  {
    "type": "object",
    "subtype": "node",
    "className": "HTMLDivElement",
    "description": "div.gh-flow",
    1) visits App prueba and survives smarter monkeys
  ]
}
```

Esta prueba a diferencia de la primera no produjo ningún resultado, ya que la prueba quedó estática por más de dos horas en el mensaje de la consola mostrado en la imagen anterior.

## Conclusiones

Las pruebas con la herramienta “no smart” arrojan resultados rápidos, ya que el tiempo de reproducción no fueron mayores a 3 minutos. Sin embargo, se puede ver por ejemplo en el video de pruebas de la página de login que las pruebas duran la mayoría de tiempo en tiempos muertos en los que no se ve actividad en el navegador. También se puede apreciar que en el video de pruebas de la página del blog hay una interacción entre el monkey y el navegador, principalmente con el botón de “subscribirse”. Infortunadamente las pruebas con la herramienta “smart” no dieron resultados positivos o negativos para este análisis debido al bug de la herramienta que se mencionó anteriormente. Luego de apreciar estos resultados, se puede decir que a pesar de que las pruebas con monkey test son una herramienta útil para detectar fallos “raros”, es necesario aplicar métodos de pruebas adicionales que permitan dar un alcance y cobertura mayor que los resultados obtenidos e.