

Estrategia de Pruebas

Realizado Por:

Juan Andrés Romero – 202013449

Juan Sebastián Alegría – 202011282

Juan Pablo Correa – 201717404

Luisa Johanna Torres – 202314749

1. Aplicación Bajo Pruebas

1.1. Nombre Aplicación: Ghost

1.2. Versiones: 3.41.1 (principal) y 4.44.0 (para comparar con VRT)

1.3. Descripción:

Ghost es una plataforma de blogging de código abierto que se centra en la simplicidad, la velocidad y la eficiencia. Desde su creación en 2013, Ghost ha sido diseñada específicamente para bloggers y escritores profesionales que buscan una plataforma de blogging simple y fácil de usar.

A diferencia de otras plataformas de blogging, Ghost se caracteriza por su interfaz de usuario minimalista y su editor de contenido intuitivo que permite a los usuarios crear y publicar contenido de manera rápida y sencilla. Además, la plataforma ofrece una amplia variedad de temas personalizables, lo que permite a los usuarios crear un blog con una apariencia única y atractiva.

Otra ventaja de Ghost es su capacidad para integrarse fácilmente con herramientas y servicios populares, como Google Analytics, lo que ayuda a los usuarios a rastrear el rendimiento de su blog y optimizarlo para los motores de búsqueda. Además, Ghost es escalable y puede manejar un aumento en la cantidad de usuarios y la carga de trabajo.

Al ser una aplicación basada en Node.js y completamente de código abierto, cualquier persona puede contribuir al desarrollo de la plataforma. Ghost se distribuye bajo la Licencia MIT y puede ser descargado y utilizado de manera gratuita. También existe una versión comercial de Ghost, Ghost Pro, que ofrece características adicionales y soporte al usuario.

1.4. Funcionalidades Core:

- **Manejo de posts:** Esta funcionalidad permite al administrador de Ghost crear, editar y eliminar publicaciones de blog en su sitio web. Desde el panel de administración, se pueden redactar nuevos posts, agregar imágenes y otros medios, programar su publicación y actualizar o eliminar publicaciones existentes.
- **Manejo de tags:** Esta función permite al administrador de Ghost organizar y categorizar las publicaciones de blog utilizando tags. Se pueden crear, editar y eliminar tags para clasificar y ordenar los posts de manera eficiente y efectiva, lo que ayuda a los visitantes del sitio web a encontrar fácilmente el contenido relevante.
- **Manejo de páginas (pages):** Esta funcionalidad permite al administrador de Ghost crear, editar y eliminar páginas en su sitio web. Las páginas son diferentes a las publicaciones de blog ya que no están organizadas por fecha y se utilizan generalmente para crear contenido estático.

- Cambiar la navegación: Esta función permite al administrador de Ghost personalizar la navegación del sitio web, incluyendo la creación y eliminación de secciones de navegación y la modificación de la estructura y diseño de la navegación.
- Crear una integración personalizada: Ghost permite a los administradores crear integraciones personalizadas para conectar su sitio web con otros servicios y herramientas que utilizan. Estas integraciones pueden ser creadas y configuradas a través del panel de administración de Ghost.
- Cambiar la configuración de la página web: Esta funcionalidad permite al administrador de Ghost personalizar la configuración general de su sitio web, como el título del sitio, la descripción, la imagen de perfil y otros ajustes de configuración.
- Editar perfil de administrador: Ghost permite al administrador actualizar su información de perfil, incluyendo su nombre, correo electrónico y contraseña.
- Code Injection: Ghost ofrece una funcionalidad avanzada de inyección de código, lo que permite al administrador personalizar el código HTML, CSS y JavaScript de su sitio web. Esto puede ser utilizado para realizar cambios en la apariencia y funcionalidad del sitio, pero debe ser utilizado con precaución ya que los cambios pueden afectar la seguridad y estabilidad del sitio.
- Crear miembros e invitar usuarios: Ghost permite a los usuarios administrar sus cuentas de usuario y permisos a los usuarios de su blog. Además, los usuarios pueden manejar permisos específicos para cada rol, lo que les permite controlar el acceso a diferentes partes del sitio. También pueden invitar a otros usuarios para que se unan al sitio y establecer permisos para ellos.

1.5. Diagrama de Arquitectura:

Enlace del diagrama de arquitectura: [clic aquí](#)

1.6. Diagrama de Contexto:

Enlace del diagrama de contexto: [clic aquí](#)

1.7. Modelo de Datos:

Enlace del modelo de datos: [clic aquí](#)

1.8. Modelo de GUI:

Enlace del modelo de GUI: [clic aquí](#)

2. Contexto de la estrategia de pruebas

2.1. Objetivos

- O1: Detectar defectos por medio de pruebas exploratorias manuales sobre las funcionalidades core de Ghost.
- O2: Detectar defectos por medio de pruebas automatizadas exploratorias y analizar los flujos de comportamiento más recurrentes con el fin de crear escenarios de pruebas robustos.
- O3: Tener documentación de las pruebas realizadas.
- O4: Promover en los equipos de desarrollo de la compañía la disciplina necesaria para la gestión de las pruebas automatizadas mediante la comparación de los resultados del antes y después de las estrategias propuestas.
- O5: Analizar los resultados obtenidos tras la ejecución de pruebas automatizadas para detectar y priorizar errores críticos.
- O6: Validar el funcionamiento de un proceso de inicio a fin para evitar que se presenten comportamientos inesperados.
- O7: Generar por medio de las estrategias de generación de datos a-priori, pseudo random y random, escenarios de pruebas que permitan verificar el comportamiento de Ghost ante situaciones inesperadas de datos de entrada.
- O8: Verificar las funcionalidades del sistema al realizar un análisis de regresión visual entre la versión 3.41.1 de ghost y la versión 4.44.0.
- O9: Registrar los defectos encontrados en un sistema de registro de incidencias con su nivel de prioridad.

2.2. Duración de la iteración de pruebas:

La iteración durará 1 sprint de 8 semanas, a continuación, se podrán encontrar sus fechas de inicio y final.

Rango de Fechas de la iteración:

- Inicio: jueves, 25 de mayo de 2023
- Fin: jueves, 20 de julio de 2023

2.3. Presupuesto de pruebas:

2.3.1. Recursos Humanos

Para esta estrategia contamos con 4 ingenieros automatizadores senior:

NOMBRE	EXPERIENCIA
Juan Andrés Romero	5 años de experiencia en el área de desarrollo con fuertes conocimientos en pruebas y experiencia en herramientas como: Cypress, Kraken, ResembleJS, Cucumber y Javascript
Juan Sebastián Alegría	5 años de experiencia en el área de desarrollo con fuertes conocimientos en pruebas y experiencia en herramientas como: Cypress, Kraken, ResembleJS, Cucumber y Javascript
Juan Pablo Correa	5 años de experiencia en el área de desarrollo con fuertes conocimientos en pruebas y experiencia en herramientas como: Cypress, Kraken, ResembleJS, Cucumber y Javascript

Luisa Johanna Torres	5 años de experiencia en el área de desarrollo con fuertes conocimientos en pruebas y experiencia en herramientas como: Cypress, Kraken, ResembleJS, Cucumber y Javascript
----------------------	--

La asignación de cada ingeniero automatizador será de 8 horas semanales, la estrategia propuesta tiene una duración de 8 semanas.

2.3.2. Recursos Computacionales

Se tendrá a disposición 4 computadores personales portátiles con las siguientes características:

- Marca: Lenovo.
- Procesador: Intel Core I7.
- Memoria RAM: 8GB.
- Disco: SSD 512 GB.
- Sistema Operativo: Windows 11 de 64 bits.
- Software: Los programas y servicios que se necesitan para realizar las pruebas como Cypress, ResembleJS y Kraken.

Adicionalmente, se cuenta con máquina de GCP para la ejecución de las pruebas automatizadas en la nube por medio de un pipeline ejecutado con GitHub Actions.

[←](#) Edit ghost-2 instance

Machine configuration

✓ General purpose

Compute optimized

Memory optimized

GPUs

Machine types for common workloads, optimized for cost and flexibility

Series

E2

CPU platform selection based on availability

Machine type

Choose a machine type with preset amounts of vCPUs and memory that suit most workloads. Or, you can create a custom machine for your workload's particular needs. [Learn more](#)

PRESET

CUSTOM

e2-micro (2 vCPU, 1 GB memory)

vCPU

0.25-2 vCPU (1 shared core)

Memory

1 GB

ADVANCED CONFIGURATIONS

Pricing summary

Monthly estimate

\$6.11

That's about \$0.01 hourly

Pay for what you use: no upfront costs and per second billing

Item	Monthly estimate
2 vCPU + 1 GB memory	\$6.11
Total	\$6.11

Esta instancia en la nube ayuda a los ingenieros a utilizar el mismo ambiente de ejecución de pruebas y agilizar su flujo de trabajo, al proveer un sistema bajo pruebas en una URL pública. Además, se garantiza la calidad de las pruebas al ejecutar un proceso de CI/CD que evita la existencia de problemas, como por ejemplo los flaky tests. Al mismo tiempo, al final de este workflow se publican reportes de interés utilizando GitHub pages, como lo son los resultados de las pruebas de VRT.

code-automators / ghost-e2e Public

← Cypress Test Automation

✓ Week 7 - Data Generation and Input Validation #158

Re-run all jobs

Summary

Jobs

- ✓ get-date
- ✓ cypress-data-testing
- ✓ cypress (v3.41.1)
- ✓ cypress (v4.44.0)
- ✓ vrt-report

Run details

- Usage
- Workflow file

Triggered via pull request 5 days ago

LuisaTorresMon synchronize #38 development

Status: Success

Total duration: 13m 23s

Artifacts: 3

cypress.yml

on: pull_request

```

graph LR
    A[get-date 3s] --> B[cypress-data-testing 2m 57s]
    B --> C[Matrix: cypress  
cypress (v3.41.1) 7m 50s  
cypress (v4.44.0) 4m 54s]
    C --> D[vrt-report 2m 8s]
  
```

2.3.3. Recursos Económicos para la contratación de servicios/personal:

Dentro de los recursos económicos adicionales, somos conscientes de que el uso de GitHub Actions para ejecutar los pipelines dentro del repositorio privado (como es el caso de la empresa TDSC) no es siempre gratis. Más específicamente, la documentación de GitHub Actions indica que “el uso de GitHub Actions es gratis para los runners estándar dentro de repositorios públicos, pero para los repositorios privados, cada cuenta de GitHub recibe una cierta cantidad de minutos y almacenamiento gratis, siempre controlado por los límites establecidos”, referencia: <https://docs.github.com/en/billing/managing-billing-for-github-actions/about-billing-for-github-actions>

Adicionalmente como fue mencionado en la sección anterior, contamos con una máquina desplegada en GCP, que cuenta con las siguientes características, para la cual destinamos un total de \$6.11 USD mensuales más los costos Github Actions:

Pricing summary

Monthly estimate

\$6.11

That's about \$0.01 hourly

Pay for what you use: no upfront costs and per second billing

Item	Monthly estimate
2 vCPU + 1 GB memory	\$6.11
Total	\$6.11

2.4. TNT (Técnicas, Niveles y Tipos) de pruebas:

NIVEL	TIPO	TÉCNICA	OBJETIVO
Sistema	Funcional	Pruebas manuales	O1
Sistema	Funcional – Caja negra	Pruebas de reconocimiento con Monkey Cypress y RIPuppet	O2
Sistema	Funcional – Caja blanca	Pruebas con APIs de automatización	O4
Sistema	Funcional – Caja blanca	Pruebas de E2E	O6
Sistema	Funcional – Caja blanca	Pruebas de regresión visual VRT	O8
Sistema	Funcional – Caja blanca	Pruebas con escenarios de validación de datos	O7

2.5. Distribución de Esfuerzo

La siguiente distribución de esfuerzo está asociada a la pirámide de automatización, debido a que hay un mayor énfasis en el desarrollo de pruebas unitarias y automatizadas sobre las pruebas manuales.

Para la mayoría de las actividades, los cuatro ingenieros automatizadores trabajarán en conjunto distribuyéndose los diferentes escenarios para abarcar una mayor cantidad de pruebas posibles y poder encontrar el mayor número de defectos en la aplicación. Con esto en mente y teniendo en cuenta que cada tester senior dispone de 8 horas por semana, se presenta a continuación la distribución de esfuerzo planeada para la estrategia de pruebas.

SEMANA	ID	ACTIVIDADES
1	1	Realizar pruebas manuales de exploración para detectar posibles errores iniciales y reportarlos en el sistema de registro de incidencias.
	2	Definir y distribuir los escenarios para implementar de acuerdo con lo encontrado en las pruebas manuales.
2	3	Configurar el ambiente para realizar pruebas con Monkey Cypress y RIPuppet.
	4	Ejecutar las pruebas de reconocimiento con Monkey Cypress y RIPuppet.
	5	Ejecutar varias iteraciones de pruebas con diferentes números de eventos y semillas.
	6	Reportar las incidencias encontradas en las pruebas de reconocimiento.
3	7	Aprovisionar y configurar ambientes en GCP.
	8	Diseñar e implementar un pipeline para la ejecución automática de las pruebas a futuro en la nube.
4	9	Instalar y configurar las herramientas (Cypress y Kraken) para las pruebas E2E.

	10	Generar scripts de los escenarios para las pruebas E2E para la versión 3.41.1 de Ghost.
	11	Ejecutar los escenarios de las pruebas E2E para la versión 3.41.1 de Ghost.
5	12	Implementar la función de tomar registros en forma de pantallazo para cada una de las pruebas E2E.
	13	Revisar y reportar los resultados de la ejecución de las pruebas E2E de la versión 3.41.1 para la versión 4.44.0.
	14	Diseñar y modificar scripts de los escenarios para las pruebas E2E para la versión 4.44.0 de Ghost.
	15	Ejecutar los escenarios de las pruebas E2E para la versión 4.44.0 de Ghost.
6	16	Configurar e implementar un script de reportes VRT con la herramienta (ResembleJS).
	17	Modificar el pipeline para realizar la ejecución, análisis y publicación del reporte automática en una página web.
	18	Ejecutar los escenarios de las pruebas de regresión visual.
	19	Reportar posibles incidencias encontradas con el script de VRT entre las versiones 3.41.1 y 4.44.0 de Ghost.
7	20	Definir escenarios de generación de datos que usen las técnicas de datapool a-priori, pseudo random y/o random con la ayuda de Faker y/o Mockaroo.
	21	Implementar las técnicas de generación de datos en los escenarios E2E previamente generados.
	22	Ejecutar y reportar posibles incidencias encontradas con las pruebas de generación de datos.
8	23	Realizar el análisis de los resultados de las diferentes pruebas realizadas a la aplicación de Ghost.
	24	Presentar informe final.

ID ACTIVIDAD	RECURSOS HUMANOS UTILIZADOS	RECURSOS COMPUTACIONALES UTILIZADOS (Trabajo humano + ejecución)	HORAS TRABAJO POR PERSONA
1	4	20 horas computadores personales	5
2	4		2
3	4	4 horas computadores personales	1
4	4	4 horas computadores personales	1
5	4	4 horas computadores personales	1
6	4		1
7	1	1 hora máquina GCP	1
8	4	20 horas computadores personales	5

9	4	8 horas computadores personales	2
10	4	20 horas computadores personales	5
11	4	8 horas máquina GCP	1
12	4	8 horas computadores personales	2
13	4		2
14	4	12 horas computadores personales	3
15	4	8 horas computadores personales	1
16	4	16 horas computadores personales	4
17	4	1 hora computador personal	1
18	4	2 horas máquina GCP	1
19	4		1
20	4		2
21	4	20 horas computadores personales	5
22	4	4 horas máquina GCP	1
23	4		4
24	4		4

- Diagrama de Gantt:

	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8
16								
17								
18								
19								
20								
21								
22								
23								
24								