

Estrategia de Pruebas #1

1. Aplicación Bajo Pruebas

- 1.1. **Nombre Aplicación:** Ghost
- 1.2. **Versiones:** 5.96.0 RC, 4.5 base
- 1.3. **Descripción:** Es la interfaz de administración de Ghost, una plataforma para la publicación y manejo de contenido. Permite crear, editar y programar publicaciones, gestionar usuarios y roles, integrar servicios externos y personalizar el sitio web. Tiene un editor de contenido, herramientas de analítica y una interfaz para facilitar la administración de contenido.

1.4. Funcionalidades Core:

- 1. Crear, editar y eliminar publicaciones: Permite a los administradores gestionar el contenido del sitio mediante la creación, edición y eliminación de publicaciones.
- 2. Agendar y desprogramar publicaciones: Permite programar publicaciones para que se publiquen en una fecha y hora definidas, así como cancelar la programación.
- 3. Crear, editar y eliminar páginas: Permite a los administradores gestionar el contenido del sitio mediante la creación, edición y eliminación de páginas.
- 4. Agendar y desprogramar páginas: Permite programar páginas para que se publiquen en una fecha y hora definidas, así como cancelar la programación.
- 5. Administrar usuarios y asignar roles: Facilita la gestión de usuarios y la asignación de roles como Administrador, Editor, Autor y Colaborador.
- 6. Personalizar el tema y la apariencia: Permite modificar el diseño y la apariencia general del sitio.
- 7. Integrar herramientas externas: Permite la integración con servicios como Google Analytics, Zapier y plataformas de correo.
- 8. Administrar etiquetas y categorías: Permite la (1) Crear, editar y eliminar publicaciones; y (2) Agendar y desprogramar publicaciones para organizar el contenido.
- 9. Gestionar suscriptores y boletines: Facilita la administración de la lista de suscriptores y el envío de boletines informativos.
- 10. Visualizar estadísticas: Muestra datos sobre visitas, vistas y suscriptores para monitorear el rendimiento del sitio.
- 11. Administrar configuraciones generales: Permite modificar configuraciones generales como el nombre, dominio y descripción del sitio.

1.1. Diagrama de Arquitectura:

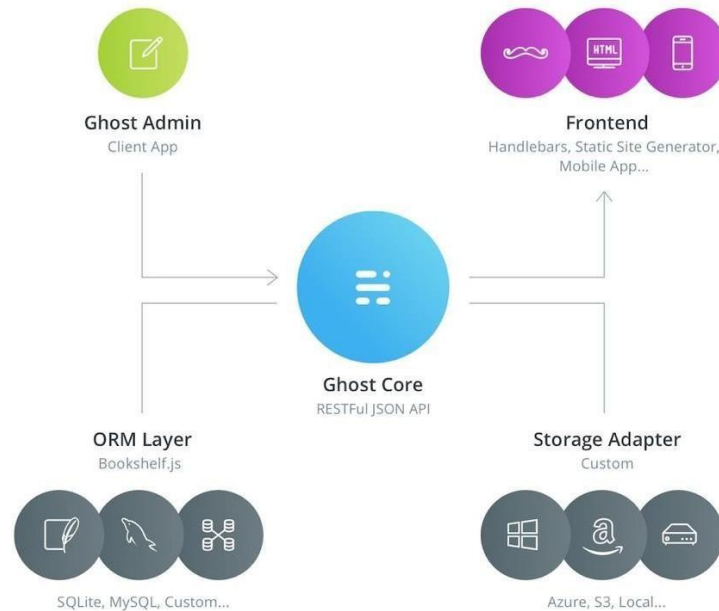


Figura 1. Diagrama arquitectura Ghost [3]

1.2. Diagrama de Contexto:

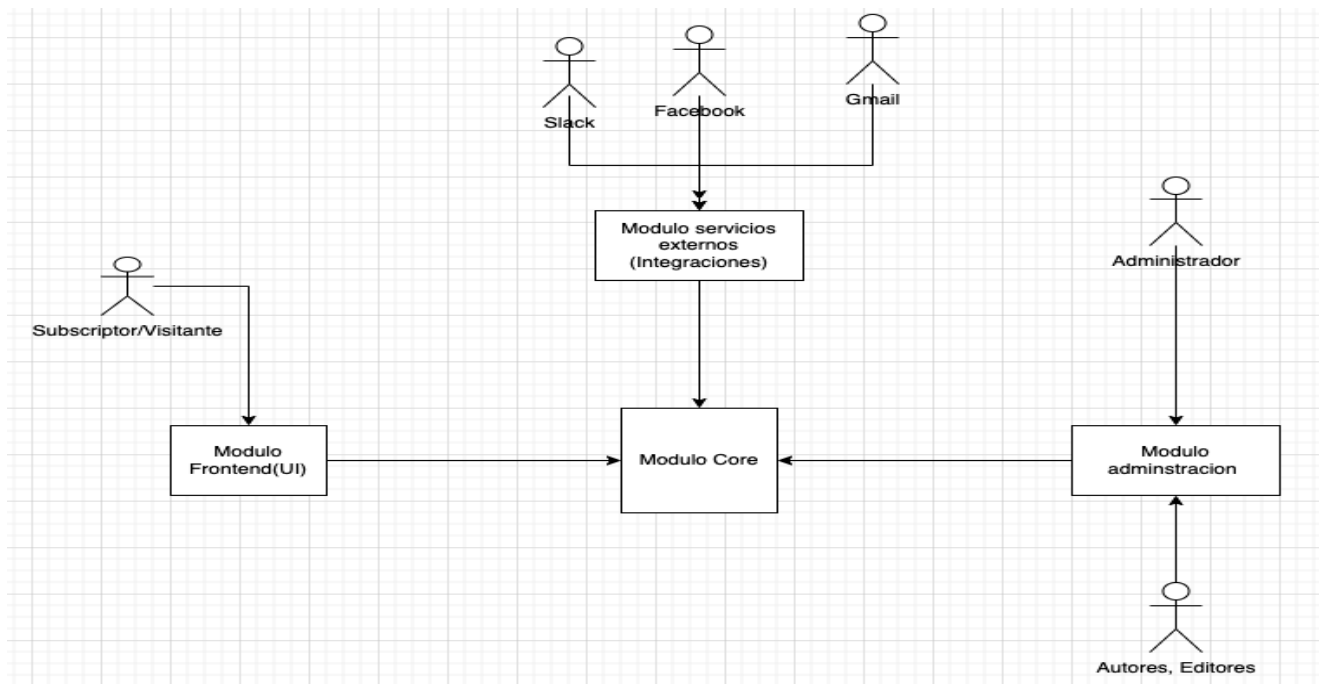


Figura 2. Diagrama contexto aplicación Ghost [4]

1.3. Modelo de Datos:

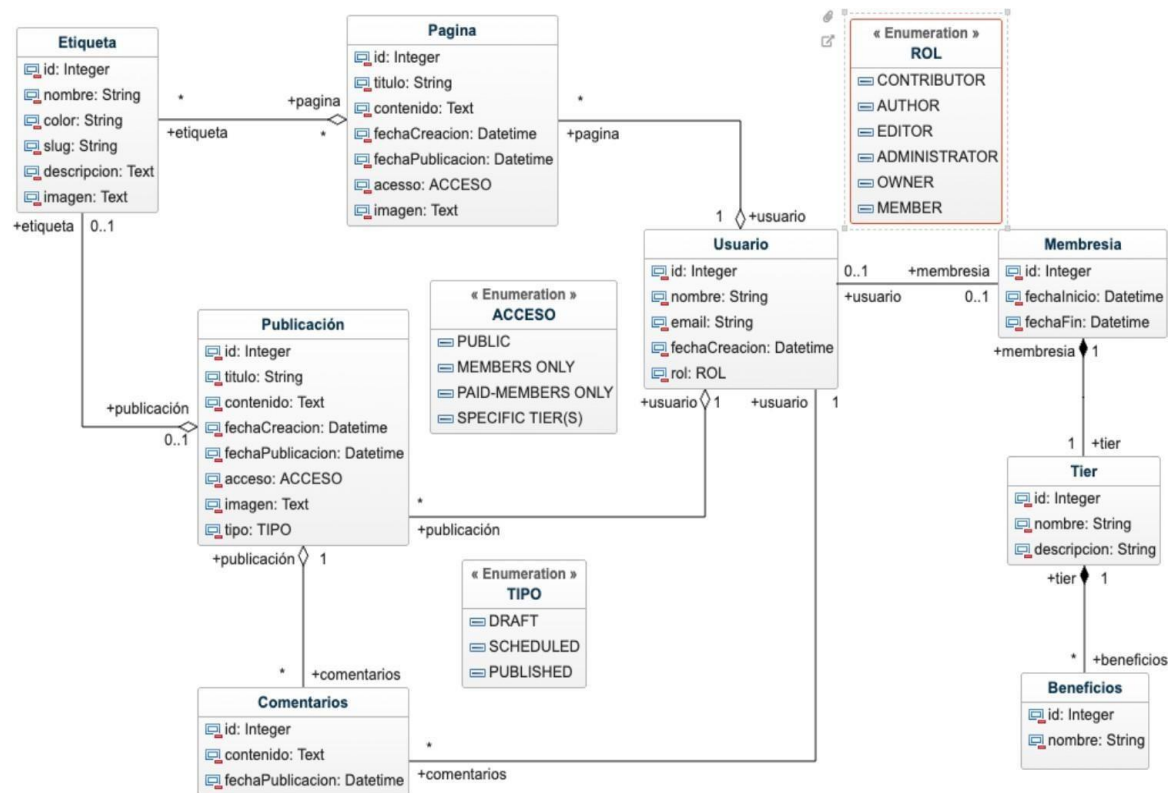


Figura 3. Modelo de datos

1.4. Modelo de GUI:

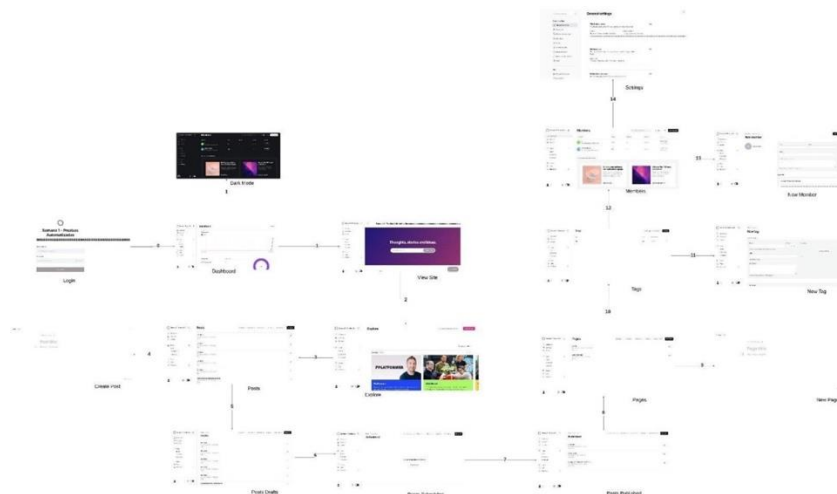


Figura 4. Modelo GUI [5]

2. Contexto de la estrategia de pruebas

2.1. Objetivos:

1. Realizar pruebas de reconocimiento con herramientas como Monkeys y Rippers para evaluar posibles casos no considerados en los flujos principales de las funcionalidades y analizando al menos 2500 eventos aleatorios por funcionalidad.
2. Ejecutar pruebas de regresión visual para validar las funcionalidades Ghost, usando las herramientas Resemble y Backstop para comparar al menos 20 pantallas clave entre las versiones RC Ghost 5.96 y 4.5, y registrar incidencias encontradas en el sistema de incidentes [9], además de almacenar y generar un reporte de evidencias visual entre la versión base y rc.
3. Realizar pruebas de extremo a extremo con Kraken y Cypress para los flujos completos de las funcionalidades. Ejecutar al menos 200 pruebas, logrando una cobertura del 95% de los casos identificados.
4. Realizar pruebas manuales de exploración en el sistema, documentarlas en el inventario de pruebas exploratorias y registrar cualquier error encontrado en el sistema de incidencias para que sean solucionados, logrando verificar al menos el 90% de las funcionalidades críticas.
5. Diseñar y ejecutar pruebas de validación de datos utilizando escenarios a-priori, aleatorios y pseudo-aleatorios para verificar las validaciones de entrada en las funcionalidades del sistema, logrando completar al menos el 95% de los casos de prueba, utilizando faker-js para generar datos aleatorios, Mockaroo con su API para datos pseudo-aleatorios reproducibles y fixtures de Cypress para los escenarios a-priori.

2.2. Duración de la iteración de pruebas:

Para este presupuesto se propone una duración de ocho semanas y 4 sprint de dos semanas cada uno, que comienza el lunes 25 de noviembre de 2024 y terminará el viernes 17 de enero de 2025.

Sprint 1:

25 de noviembre 2024 – 6 de diciembre 2024

Sprint 2:

9 de diciembre 2024 – 20 de diciembre 2024

Sprint 3:

23 de diciembre 2024 – 3 de enero 2025

Sprint 4:

6 de enero 2025 – 17 de enero 2025

2.3. Presupuesto de pruebas:

El presupuesto contempla la participación de cuatro ingenieros QA senior, cada uno con un salario mensual promedio de \$5.250.000 COP[1][2], basado en datos actuales del mercado. Teniendo en cuenta que en Colombia el promedio de horas laborales mensuales es de 160 horas, el costo por hora se estima en \$32.812,5 COP/hora.

Cada ingeniero trabajará un total de 64 horas a lo largo de las ocho semanas, distribuidas en cuatro sprint de dos semanas cada uno, con una dedicación semanal de ocho horas por ingeniero. De esta forma, el costo por ingeniero se calcula como:

Costo por ingeniero = $32.812,5 \text{ COP/hora} \times 64 \text{ horas} = 2.100.000 \text{ COP}$. Costo por ingeniero = $32.812,5 \text{ COP/hora} \times 64 \text{ horas} = 2.100.000 \text{ COP}$.

El costo total del equipo completo, considerando los cuatro ingenieros, será:

Costo total del equipo = $2.100.000 \text{ COP/ingeniero} \times 4 = 8.400.000 \text{ COP}$. Costo total del equipo = $2.100.000 \text{ COP/ingeniero} \times 4 = 8.400.000 \text{ COP}$.

Por lo tanto, el costo total del equipo de cuatro ingenieros QA senior para las ocho semanas es de \$8.400.000 COP.

2.3.1 Recursos Humanos:

Equipo Senior QA: consiste en cuatro ingenieros con un perfil senior, con cinco años de experiencia en la industria deben tener experiencia de pruebas exploratorias, documentación y registro de inventario de pruebas exploratorias, además de experiencia en pruebas de extremo a extremo, pruebas vrt (visual regression testing), además de Pruebas de Escenarios de Validación de Datos, además de conocimiento del patrón de pirámide de automatización.

De igual forma, debe tener experiencia en el manejo de herramientas como: Cypress, Kraken, ResembleJS y Backstop.

2.3.2 Recursos Computacionales

Para los recursos computacionales, se implementará la política empresarial BYOD ("Bring Your

Own Device"), en la que cada ingeniero utilizará sus propios equipos para ejecutar las tareas asignadas en cada sprint. Los equipos deberán cumplir con las siguientes especificaciones mínimas:

MacBook Pro M1 o equivalente.

16 GB de RAM.

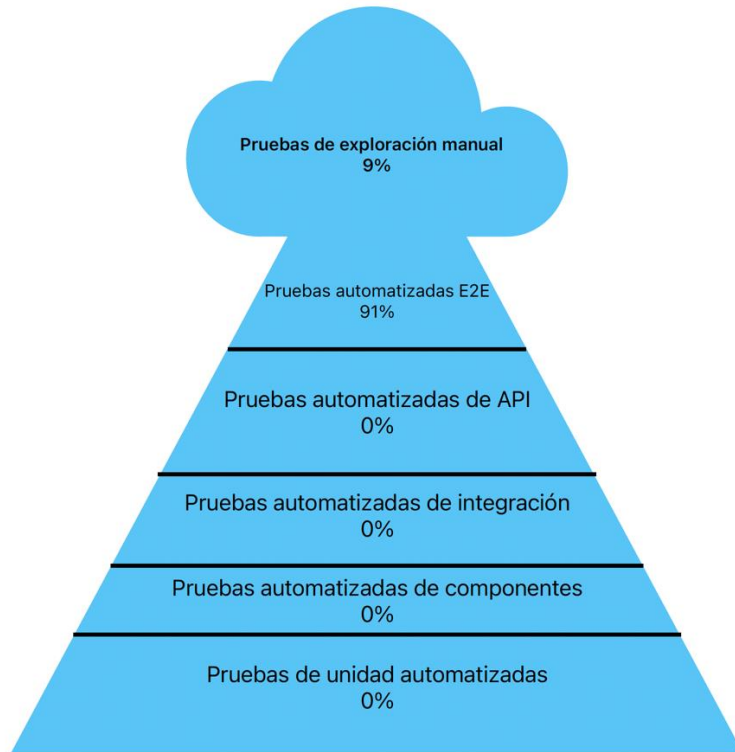
Estas características aseguran que los dispositivos sean adecuados para las pruebas requeridas a lo largo de las 8 semanas, optimizando los costos al no incluir adquisiciones de hardware en el presupuesto.

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

Nivel	Tipo	Técnica	Objetivo
Sistema	Funcional	Pruebas E2E, Api automatización (Kraken)	3
Sistema	Funcional	Pruebas E2E, API automatización (Cypress)	3
Sistema	Funcional	Pruebas de reconocimiento (GUI Ripping)	1
Sistema	Funcional	Pruebas de reconocimiento (Monkey Testing)	1
Sistema	Funcional	Pruebas de regresión visual, API automatización (ResembleJS)	2
Sistema	Funcional	Pruebas de regresión visual, API automatización (BackstopJS)	2
Sistema	Funcional	Pruebas exploratorias manuales	4
Sistema	Positivo	Escenarios de validación de datos (Aleatorio - Faker)	5
Sistema	Positivo	Escenarios de validación de datos (A-priori - Mookaro)	5
Sistema	Positivo	Escenarios de validación de datos (Pseudo-aleatorio – Api Mookaro)	5
Sistema	Negativo	Escenarios de validación de datos (Aleatorio - Faker)	5
Sistema	Negativo	Escenarios de validación de datos (A-priori - Mookaro)	5
Sistema	Negativo	Escenarios de validación de datos (Pseudo-aleatorio – Api Mookaro)	5

2.5 Distribución de Esfuerzo

A continuación, se presenta el cronograma de trabajo de ocho semanas.



El 9% de las pruebas son manuales.

Mientras que el 91% de las pruebas son cubiertas por pruebas automatizadas E2E y su 100% se divide en:

- Pruebas E2E – 40% (Kraken 20% y Cypress 20%)
- Pruebas de reconocimiento – 10% (Rippers 5% y Mokey testing 5%)
- Pruebas de regresión visual – 20% (Resemble 10% - Backstop 10%)
- Escenarios de validación de datos positivas (aleatorios) – 5%
- Escenarios de validación de datos positivas (a-priori) – 5%
- Escenarios de validación de datos positivas (pseudo-aleatorio) – 5%
- Escenarios de validación de datos negativas (aleatorios) – 5%
- Escenarios de validación de datos negativas (a-priori) – 5%
- Escenarios de validación de datos negativas (pseudo-aleatorio) – 5%

Bibliografías

1. Cooltesters, "Salarios en software testing 2022 Colombia." Disponible en:
<https://www.cooltesters.com/blog/salarios-en-software-testing-2022-colombia>
2. Computrabajo, "Salarios para Tester QA en Colombia." Disponible en:
<https://co.computrabajo.com/salarios/tester-qa>
3. Ghost Foundation, "Ghost Documentation – Architecture Overview." Disponible en: <https://ghost.org/docs/architecture/>
4. Draw.io, "Diagrama de contexto." Disponible en: <https://drive.google.com/file/d/1Q-ofhqIvnn91Cqoopb8ulOw9FOdHGMHI/view?usp=sharing>
5. Miro, "Modelo GUI." Disponible en:
https://miro.com/app/board/uXjVLT4c5bQ=?share_link_id=475695657223
6. Comparador Precios instancias EC2. Disponible en : <https://instances.vantage.sh/>
7. EC2 spot instances <https://aws.amazon.com/ec2/spot/>
8. Procesador <https://www.intel.la/content/www/xl/es/products/sku/238777/intel-core-i5-processor-14401e-24m-cache-up-to-4-70-ghz/specifications.html>