

# Estrategia de Pruebas

Link video estrategias: [https://uniandes-my.sharepoint.com/:v:/g/personal/d\\_acostao\\_uniandes\\_edu\\_co/ETB2KUUcdMZEunmioVhr89sByKX9uQB\\_D8tRBvao3vCQEw?e=uUVcgp](https://uniandes-my.sharepoint.com/:v:/g/personal/d_acostao_uniandes_edu_co/ETB2KUUcdMZEunmioVhr89sByKX9uQB_D8tRBvao3vCQEw?e=uUVcgp)

## 1. Aplicación Bajo Pruebas

**1.1. Nombre Aplicación:** Ghost

**1.2. Versión:** 3.42.0

**1.3. Descripción:** Ghost es una aplicación que permite generar, gestionar y publicar contenido (CMS) para la realización de sitios web y blogs. Adicionalmente cuenta con un módulo de administración que permite visualizar diferentes KPI's en forma de dashboard para monitorear la evolución del sitio, administrar usuarios, crear contenido programado, entre otras funcionalidades.

**1.4. Funcionalidades Core:**

- Gestión de contenido web (Diseño, visualización y personalización de sitios web)
- Gestión de usuarios (Roles y permisos según el rol del usuario)
- Gestión de perfiles
- Gestión de Contenido (Escritura y edición de posts)
- Creación de cuentas
- Visualización del historial de posts (incluye los borradores que son aquellos que tienen progreso pero no han sido publicados)

**1.5. Diagrama de Arquitectura:**

[Diagrama de Arquitectura.](#)

**1.6. Diagrama de Contexto:**

[Diagrama de Contexto.](#)

**1.7. Modelo de Datos:**

[Modelo de Datos.pdf](#)

**1.8. Modelo de GUI:**

[Modelo de GUI](#)

## 2. Contexto de la estrategia de pruebas

**2.1. Objetivos:**

El objetivo principal de la presente estrategia de pruebas es poder detectar bugs en la versión 3.42.0 de Ghost, para ello el enfoque principal se encuentra orientado hacia el aspecto funcional de la aplicación, a través de la validación de las funcionalidades seleccionadas, utilizando escenarios definidos.

**2.2. Duración de la iteración de pruebas:**

La duración total de las pruebas será de 8 semanas. En donde durante las primeras 5 semanas se contará con 4 ingenieros tester senior dedicados 8 horas semanales y durante las últimas 3 semanas, se contará con un solo ingeniero tester senior dedicado 12 horas semanales.

Teniendo en cuenta lo anterior, el esfuerzo dedicado suma en total 196 horas de dedicación para llevar a cabo la iteración.

y se dividirá en dos fases:

- Fase 1: Durante las primeras 5 semanas, se cuenta con 4 ingenieros testers senior, dedicados a llevar a cabo las tareas de:
  - Definir la estrategia de pruebas.
  - Elegir las herramientas a utilizar
  - Implementar el código de las pruebas automatizadas.
  - Plantear los escenarios para las pruebas de extremo a extremo.
  - Describir los escenarios en lenguaje Gherkin.
  - Implementar el código que ejecuta las pruebas e2e.
- Fase 2: Durante las 3 últimas semanas, se cuenta con 1 solo ingeniero tester el cual se dedicará a realizar las siguientes tareas:
  - Evaluar una nueva versión de Ghost.
  - Ajustar el código para la ejecución de los escenarios definidos para la versión anterior.
  - Realizar pruebas VRT.
  - Implementar escenarios de validación de datos
    - Pool de datos a-priori
    - Pool de datos (psedo) aleatorio.
    - Escenario aleatorio.

## 2.3. Presupuesto de pruebas:

### 2.3.1. Recursos Humanos

Teniendo en cuenta el salario medio para un automatizador de pruebas senior el cual es de **\$26.099**, se plantea la siguiente tabla para saber el valor total destinado para la realización la iteración:

Personal	Valor hora	Horas/ Sem	Semanas	Total
Ing Automatizador senior x4	\$104.386	8	5	\$4'175.840
Ing Automatizador senior x1	\$26.099	12	3	\$939.564
<b>Total</b>				<b>\$5'115.404</b>

Tomado de: Salario para Automatizador senior en Colombia 2023  
<https://co.talent.com/salary?job=automatizador+senior>

### 2.3.2. Recursos Computacionales

Dado que los ingenieros trabajan en modalidad freelance, cada uno de ellos cuenta con sus propios equipos de cómputo:

Recursos computacionales	
Equipo personal x2	MacBook Pro 14 inch – M1 Pro 16GB Unified Memory
Equipo personal x1	MSI PS42-8RC – Intel® Core™ i7-8550U CPU @ 1.80GHz, 16Gb RAM, SO Arch Linux kernel 6.2.8-arch1-1.
Equipo personal x1	Asus - AMD Ryzen 5 2500U with Radeon Vega Mobile G, 16Gb RAM, SO Ubuntu 20.04.6 LTS.

Finalmente, para las últimas 3 semanas se contó únicamente con la siguiente máquina:

Recursos computacionales	
Equipo personal x1	MSI PS42-8RC – Intel® Core™ i7-8550U CPU @ 1.80GHz, 16Gb RAM, SO Arch Linux kernel 6.2.8-arch1-1.

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

Para el desarrollo de esta propuesta no se cuenta con recursos monetarios para contratar servicios externos.

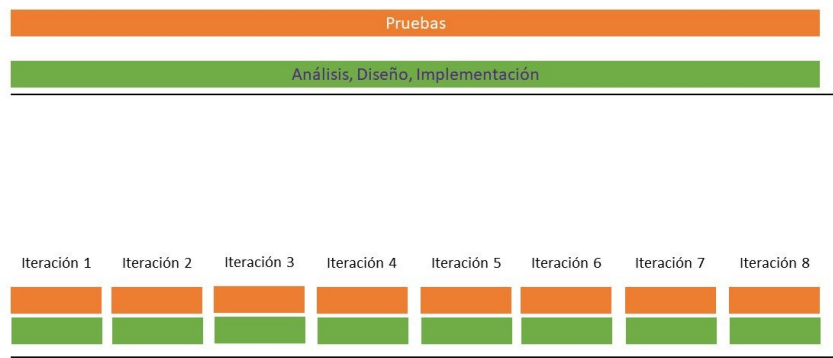
## 2.4. TNT (Técnicas, Niveles y Tipos) de pruebas:

La estrategia definida está compuesta por diferentes técnicas en cada una de las categorías de niveles de pruebas. Cubriendo así el espectro de pruebas bajo los cuales una aplicación debería ser sometida para garantizar su calidad.

Nivel	Tipo	Técnica	Objetivo
Unidad	Caja blanca	API de automatización	Identificar posibles fallos.
Integración	Funcionales	Record & Replay	Automatizar casos de prueba de diferentes funcionalidades.
Sistema	Caja Negra	Monkey Testing	Pruebas explotatorias del funcionamiento de la aplicación.
Aceptación	Positiva	Generación datos de prueba	Implementar casos de prueba en distintos ambientes.

## 2.5. Distribución de Esfuerzo

### Distribución de pruebas continuas



La intención de esta estrategia de pruebas es permitir al equipo de pruebas trabajar con retroalimentación continua durante las diferentes etapas del desarrollo del producto. Esto permite detectar fallas lo más pronto

posible y, junto a esto, solucionarlas de igual forma ayudando a mitigar los fallos que se envían al producto cuando se va a entregar.