

# Estrategia de Pruebas

## 1. Aplicación Bajo Pruebas

**1.1. Nombre Aplicación:** Ghost

**1.2. Versión:** 5.18.0

**1.3. Descripción:**

Ghost es una plataforma de publicación profesional de código abierto basada en una pila de tecnología moderna de Node.js, diseñada para equipos que necesitan potencia, flexibilidad y rendimiento.

Ghost es un CMS (por sus siglas en ingles Content Management System) en la nube, el cual te permite crear, organizar, publicar y eliminar contenidos de tu sitio web. Las CMS proveen 3 funciones principales:

- Creación de webs.
- Gestión y mantenimiento del sitio web.
- Administración de páginas web y del propio CMS.

Otras plataformas (en comparación con Ghost) abiertas son generalmente antiguas, lentas e infladas, mientras que otras plataformas cerradas no le otorgan absolutamente ningún control o propiedad de su contenido. Ghost ofrece lo mejor de ambos mundos y más.

**1.4. Enlace al video de Explicación:**

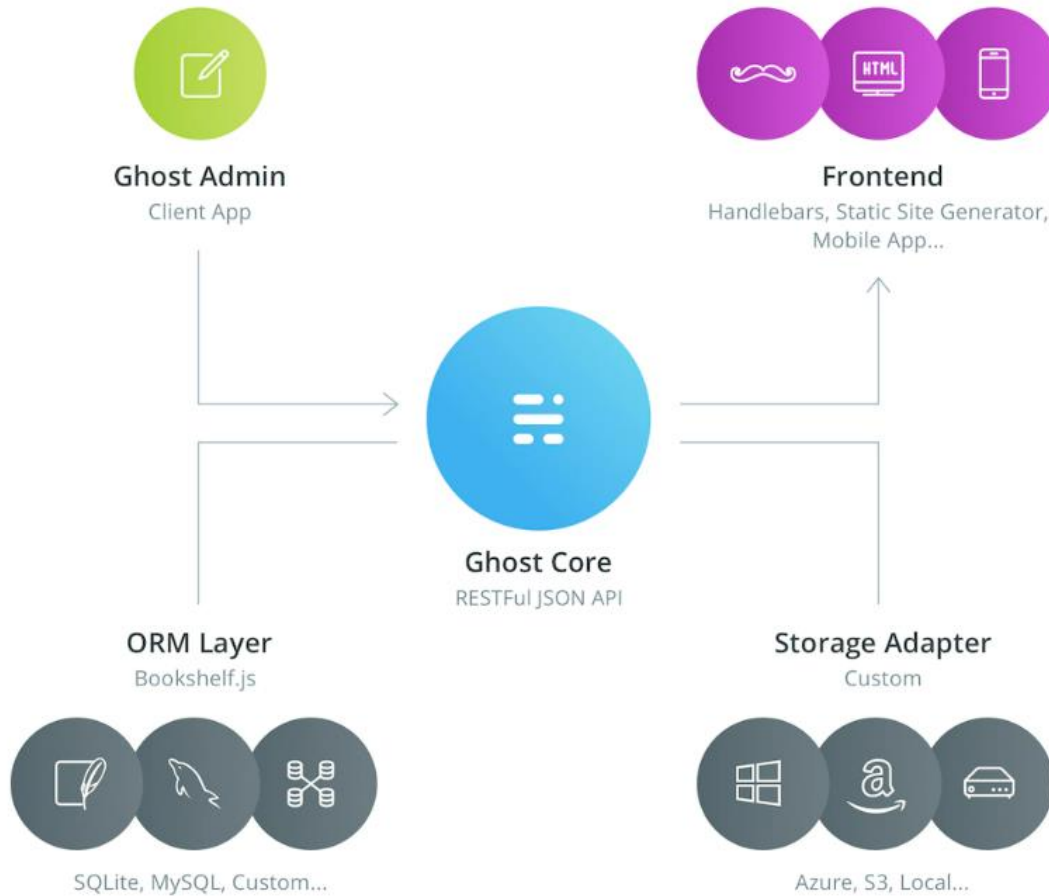
<https://youtu.be/XNEWRY1gQuw>

**1.5. Funcionalidades Core:**

Ghost provee una serie de funcionalidades entre las cuales encontramos:

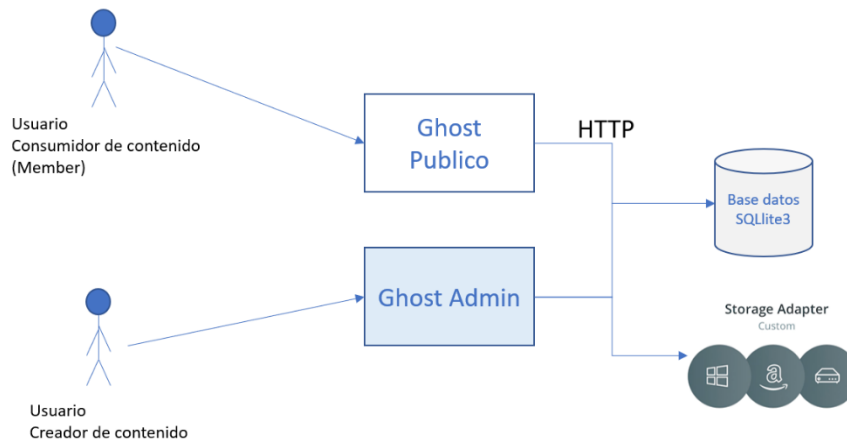
- Editar, crear y publicar un post.
- Borrar, despublicar y programar un post.
- Editar, crear, y publicar una página.
- Borrar, y despublicar una página.
- Editar, crear, y guardar un tag.
- Borrar, y despublicar un tag.
- Administrar el blog con distintos usuarios y roles específicos.
- Publicar en una fecha específica.

**1.6. Diagrama de Arquitectura:**



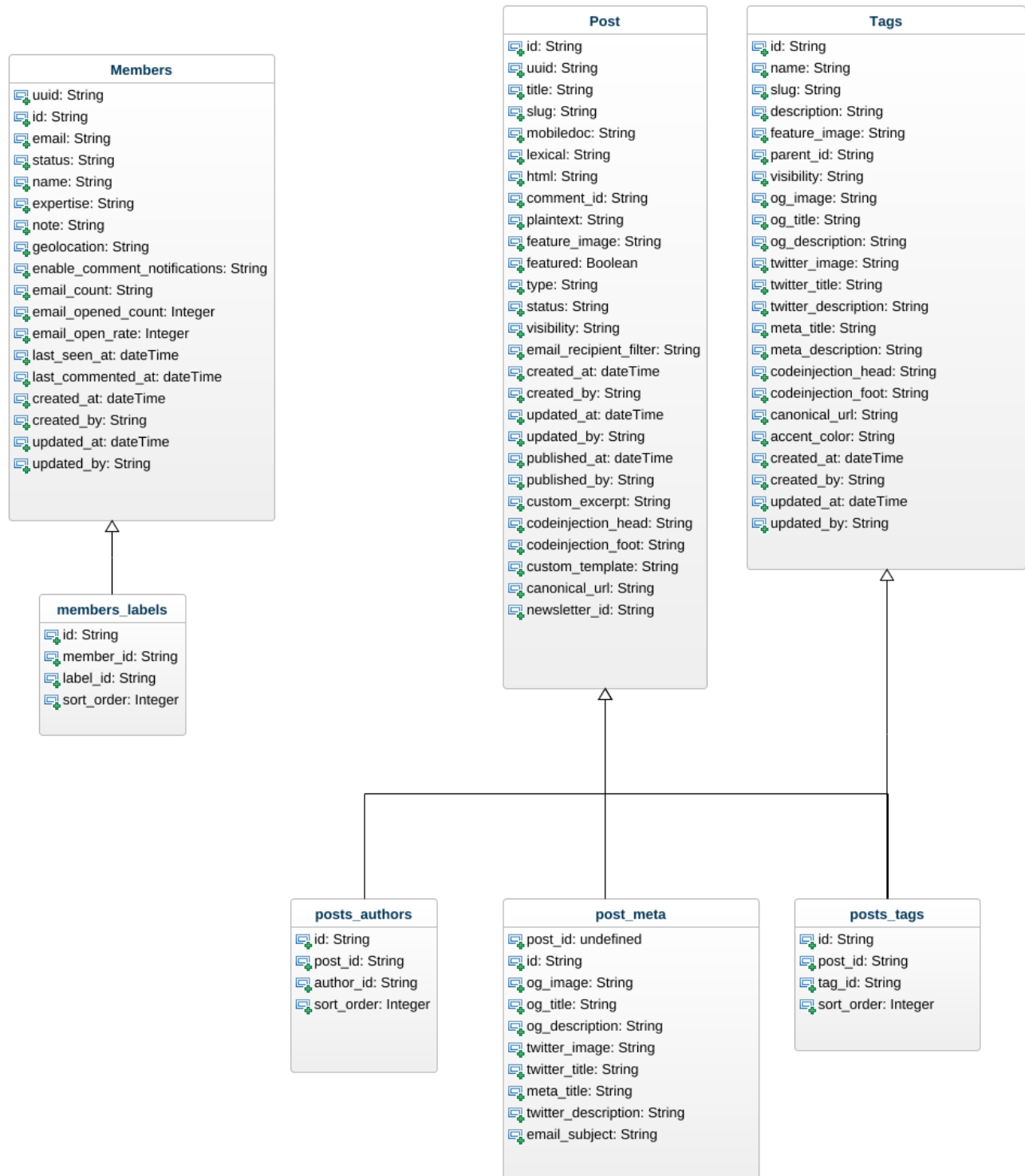
Si la imagen no se aprecia del todo una copia se encuentra dentro del zip de entrega o en el siguiente enlace:  
[https://1drv.ms/u/s!As\\_QMF2T0qpGgbpEXH1l\\_ksDyV2GcQ?e=llg8Vf](https://1drv.ms/u/s!As_QMF2T0qpGgbpEXH1l_ksDyV2GcQ?e=llg8Vf)

## 1.7. Diagrama de Contexto:



Si la imagen no se aprecia del todo una copia se encuentra dentro del zip de entrega o en el siguiente enlace:  
[https://1drv.ms/u/s!As\\_QMF2T0qpGgbpCd1lhx6P5wSTafg?e=fWpYof](https://1drv.ms/u/s!As_QMF2T0qpGgbpCd1lhx6P5wSTafg?e=fWpYof)

## 1.8. Modelo de Datos:



Si la imagen no se aprecia del todo una copia se encuentra dentro del zip de entrega o en el siguiente enlace:

[https://1drv.ms/u/s!As\\_QMF2T0qpGgbpBYUsPm1EPM9oiyA?e=ZucZ2N](https://1drv.ms/u/s!As_QMF2T0qpGgbpBYUsPm1EPM9oiyA?e=ZucZ2N)

Plantilla elaborada por

**THE SW DESIGN LAB**

## 1.9. Modelo de GUI:



Si la imagen no se aprecia del todo una copia se encuentra dentro del zip de entrega o en el siguiente enlace:

[https://1drv.ms/u/s!As\\_QMF2T0qpGgbpAaZqL\\_GglUdPoVw?e=yINb41](https://1drv.ms/u/s!As_QMF2T0qpGgbpAaZqL_GglUdPoVw?e=yINb41)

## 2. Contexto de la estrategia de pruebas

### 2.1. Objetivos:

La intención de este periodo de pruebas es el de empezar a tener un equipo estructurado de pruebas que no solo tenga un aumento de conocimientos sobre la ABP, sino que además empiece a desarrollar elementos de pruebas automatizadas que nos permita tener mejores resultados cada periodo subsecuente, es decir aumentar el catálogo de pruebas existente.

Tomando en cuenta que en la actualidad no se cuenta ni con inventario de pruebas realizadas (ni manuales ni automatizadas) ni reportes de defectos, ni documentos o artefactos de diseños, que permitan al equipo interno de pruebas alivianar la curva de aprendizaje, se espera que con el desarrollo de este equipo de pruebas internos se puedan sentar bases solidad para futuros entregables.

Como segundo objetivo se espera la entrega del siguiente paquete de 65 pruebas:

- 45 pruebas Unitarias Automatizadas.
- 20 pruebas de GUI Automatizadas.

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

Estas pruebas serán ejecutadas en un periodo de 2 semanas, esto con la finalidad de alinearse con los estándares y buenas prácticas de los equipos agiles. También nos permitirá evaluar la estrategia después de un periodo relativamente corto y determinar los cambios necesarios.

### 2.3. Presupuesto de pruebas:

| Concepto                         | Costo Unitario | Cantidad    | Monto Total |
|----------------------------------|----------------|-------------|-------------|
| Hora de Trabajo Ingeniero Junior | \$USD 5        | 150 horas   | \$USD 750   |
| Hora maquina Amazon AWS          | \$USD 96       | 1 instancia | \$USD 96    |
| Hora de Trabajo Ingeniero Senior | \$USD 10       | 40 horas    | \$USD 400   |
| Renta de Equipo de Computo       | \$USD 50       | 3 equipos   | \$USD 150   |
| Total                            |                |             | \$USD 1396  |

Para 65 pruebas a ejecutar en esta estrategia, tendríamos un valor por prueba de \$USD21.48

#### 2.3.1. Recursos Humanos

En esta estrategia se consideran los siguientes equipos de trabajo:

- 3 ingenieros automatizadores Junior (50 horas/persona), los cuales deberán tener los conocimientos teóricos en Pruebas Automatizadas, así como conocimientos relevantes en el uso de AWS (proyectos escolares, tutoriales de al menos 20 horas).
- 4 ingenieros de Software Senior con dedicación limitada (10 horas/persona) que ya poseen experiencia con la ABP y pueden proveer guía y ayuda a la hora de ser necesario.

### 2.3.2. Recursos Computacionales

En esta estrategia se consideran los siguientes equipos tecnológico:

- 200 horas/máquina en Amazon AWS, estas horas maquina serán utilizadas para correr las pruebas automatizadas entregadas tanto por el equipo de ingenieros automatizadores junior.
- 3 computadores correspondientes a cada Ingeniero automatizador junior. Estos ordenadores son MacBook Air modelos 2020 en adelante.
- 4 computadores una correspondiente a cada Ingeniero Software Senior. Estos ordenadores son MacBook Air modelos 2020 en adelante.
- 1 dispositivos Móviles con sistema Operativo Android (Samsung Galaxy S20) que ya formaban parte del equipo de la empresa.
- 1 dispositivos Móviles con sistema Operativo iOS (iPhone 11) que ya formaban parte del equipo de la empresa.

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

En esta estrategia no se considerará el uso de Servicios o Personal Externo, ya que la intención principal es la formación de un equipo que entregará resultado de pruebas y a su vez se incrementará sus conocimientos en la ABP y creará valor a la empresa a lo largo del tiempo.

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

| Nivel   | Tipo                                  | Técnica      | Objetivo  |
|---------|---------------------------------------|--------------|---|
| Sistema | Funcional<br>Caja Negra<br>Positivas  | Manual       | Son las pruebas exploratorias de los Ingenieros Junior, la intención es que se familiaricen con la ABP y se ubiquen en las vistas y elementos de esta.<br>Se espera un reporte manual con pasos y resultados.   |
| Sistema | Funcional<br>Caja Negra<br>Positivas  | Automatizada | En estas pruebas se espera la automatización de las pruebas exploratorias que presentaron defectos. En esta sección se espera que los Ingenieros Seniors aporten con sus conocimientos previos de la ABP.<br>Se espera el Script de automatización a ser corrido en los servidores AWS.   |
| Unidad  | Funcional<br>Caja Blanca<br>Positivas | Automatizada | Estas pruebas tienen la intención de empezar la validación desde el nivel más alto de granularidad para detectar errores desde esta etapa, se quiere que las pruebas sean positivas validando con los inputs esperados las salidas de funciones.<br>Se espera un script que corra sobre el repositorio del código y se ejecute de manera automática cada nuevo Pull Request |

|        |                                       |              |   |
|--------|---------------------------------------|--------------|---|
| Unidad | Funcional<br>Caja Blanca<br>Negativas | Automatizada | Estas pruebas tienen la intención de empezar la validación desde el nivel más alto de granularidad para detectar errores desde esta etapa, se quiere que las pruebas sean negativas utilizando inputs diferentes a los esperados y validar el manejo de los errores.<br>Se espera un script que corra sobre el repositorio del código y se ejecute de manera automática cada nuevo Pull Request |
|--------|---------------------------------------|--------------|---|

## 2.5. Distribución de Esfuerzo

En esta estimación se espera poder dividir el tiempo (de 2 semanas) de la siguiente forma:

1. Pruebas exploratorias manuales 20 horas, ya que los ingenieros Junior necesitan tiempo para conocer la ABP.
2. Pruebas de GUI Automatizadas 40 horas, ya que podrán automatizar ya sea los errores encontrados por los Ingenieros Seniors en iteraciones pasadas o errores encontrados durante la etapa anterior.
3. Prueba Unitarias Automatizadas 90 horas, ya que así se podrán encontrar aún más errores diseccionando el código de la ABP y se empezara la construcción de catalogo formal de pruebas.

La intención es que a medida que las iteraciones pasen se puedan agregar más pruebas Unitarias y de Cobertura de código y se vaya empujando al equipo a llegar a la pirámide de automatización Ideal.



## 3. Análisis de la Propuestas

### 3.1 Ventajas

- En esta estrategia se busca que el equipo contratado obtenga mayores conocimientos a lo largo del tiempo, haciendo que su entrega de valor a la empresa sea mayor a medida que pasa el tiempo.
- Una gran ventaja es que una vez el equipo interno realiza pruebas exploratorias, se puede enfocar en desarrollar y generar conocimiento en el entendimiento del código para así asegurar las pruebas unitarias, lo cual minimizara los errores en la aplicación, lo que con el tiempo los puede llevar a tener un mayor foco en pruebas automatizadas GUI de sistema, integración y aceptación, dado que ya tienen un control del Código con la automatización de pruebas unitarias levantadas en etapas tempranas de este proceso.
- En esta estrategia se espera un mayor número de pruebas unitarias entregadas, haciendo mayor referencia a la pirámide de automatización esperada.
- A largo plazo se espera que el equipo haya entregado suficientes pruebas unitarias para poder enfocarse en pruebas End to End.

### 3.2 Desventajas

- Los resultados iniciales son limitados ya que se espera que al principio se entrene al equipo y tengan mayor conocimiento respecto a la ABP
- la Curva de aprendizaje es larga ya que los ingenieros que se están contratando no tienen conocimientos anteriores en pruebas automatizadas.
- Necesidad de Mantenimiento de pruebas unitarias por parte del equipo de desarrollo.
- se debe hacer una inversión inicial alta en generar una curva de aprendizaje en los ingenieros internos que generen los resultados esperados en corto tiempo, esperando que a largo plazo este valor agregado a la empresa sea mayor.