

Estrategia de Pruebas

Presentación de la estrategia: [Video Estrategia Final Pruebas Ghost.mkv](#)

1. Aplicación Bajo Pruebas

1.1. Nombre Aplicación: Ghost

1.2. Versión: 4.42.0

1.3. Descripción: Aplicación para que los creadores de nuevos medios publiquen, compartan y hagan crecer un negocio en torno a su contenido. Viene con herramientas modernas para crear un sitio web, publicar contenido, enviar boletines y ofrecer suscripciones pagas a los miembros.

1.4. Funcionalidades Core: A continuación, se listan las principales funcionalidades de la aplicación en orden de prioridad definidas por nuestro equipo:

- Iniciar sesión como miembro del staff
- Crear una página
- Consultar página
- Crear nuevo post
- Consultar un post
- Modificar página
- Modificar post
- Modificar etiqueta
- Eliminar página
- Eliminar post
- Crear etiqueta
- Eliminar etiqueta
- Crear suscripción o membresía
- Registrarse como suscriptor
- Iniciar sesión como suscriptor
- Recuperar contraseña
- Crear nuevo miembro manualmente
- Importar miembros
- Invitar a una persona para ser miembro del staff
- Crear navegación
- Modificar navegación
- Eliminar navegación

1.5. Diagrama de Arquitectura:

[Diagrama Arquitectura](#)

1.6. Diagrama de Contexto:

[Diagrama Contexto](#)

1.7. Modelo de Datos:

[Modelo Datos](#)

1.8. Modelo de GUI:

[Modelo GUI](#)

2. Contexto de la estrategia de pruebas

2.1. Objetivos:

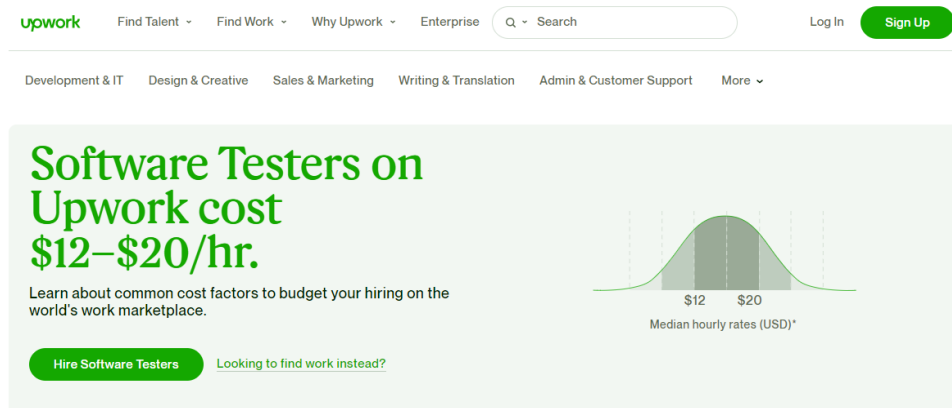
- Implementar pruebas exploratorias y de compatibilidad (usar diferentes browsers y diferentes tamaños de viewport) para redactar documentación inicial de la aplicación web Ghost durante dos semanas.
- Implementar pruebas de reconocimiento enfocadas a validar las funcionalidades Core definidas mediante la generación de una gran cantidad de datos aleatorios con el fin de identificar casos raros que no sean evidentes para un ser humano durante una semana
- Crear pruebas E2E con generación de datos para las 9 funcionalidades principales, con el propósito de validar la existencia o no de defectos funcionales del sistema durante un periodo de cuatro semanas.
- Buscar diferencias entre la versión anterior y actual de Ghost usando técnicas de regresión visual durante un periodo de una semana.

2.2. Duración de la iteración de pruebas: A partir del lunes 28 de marzo del 2022 y hasta el viernes 22 de mayo de 2022 (8 semanas).

2.3. Presupuesto de pruebas:

2.3.1. Recursos Humanos

Para la ejecución del plan de pruebas propuesto, contaremos con 4 testers senior por un tiempo de 256 horas en total. Estos ingenieros tienen una experiencia de tres años en la implementación pruebas con APIs de automatización, de reconocimiento, de regresión visual, manuales y validación de datos por lo que cuentan con los conocimientos necesarios para aplicar estas técnicas y aprovechar así los recursos de computación disponibles. Según la plataforma upwork, el costo de un ingeniero senior por hora es de aproximadamente 20 USD, por lo que se presupuesta un total de 5.120 USD para pagar los tester senior.



<https://www.upwork.com/hire/software-testers/cost/>

2.3.2. Recursos Computacionales

Para la ejecución del plan de pruebas automatizado, contaremos con una máquina virtual en AWS en un tiempo estimado de 200 horas. La máquina virtual AWS EC2 tiene un costo de 112 USD (<https://calculator.aws/#/createCalculator/EC2>) y cuenta con las siguientes especificaciones:

- Sistema operativo: Linux
- 8 núcleos de CPU - 64 bits

- Memoria RAM de 64 GB
- 1 instancia
- Utilización: 200 hora/mes

Cuatro portátiles en alquiler por el período de la estrategia de pruebas, con un costo de 30 USD. Esta estación de trabajo que se le va asignar a los testers seniors cuentan con las siguientes características:

- Core i7 8th/9th generación
- 16 GB de RAM
- 512 GB SSD

Recurso	Descripción	Esfuerzo (Horas)	Valor Hora (USD)	Subtotal (USD)
Humano	4 Tester Senior	256	20	\$ 5.120
Computacional	Maquina Virtual Amazon EC2	200	0,717	\$ 143
Computacional	4 Portatiles en Alquiler	5376	0,0416	\$ 224
Total				\$ 5.487

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

2.4.1. Técnicas

- **Record and Play:** los ingenieros senior crearan escenarios de prueba y entregaran scripts a partir de la técnica *Record and Play* a través de la herramienta Cypress.io, para luego poder replicar las pruebas. Esta técnica será implementada por dos ingenieros senior que se encargaran de realizar pruebas durante 16 horas de trabajo cada uno.
- **Monkey testing:** Se generan eventos sin intervención de humanos contra la interfaz gráfica, usando el recurso de computación adquirido para implementar la técnica *Smart Monkey aleatorio con heurísticas* mediante la herramienta Cypress.io, con el objetivo de descubrir problemas en el software y explorar aquellas funcionalidades que no pueden ser validadas fácilmente de manera manual por los ingenieros senior. Esta técnica será implementada por uno de los ingenieros senior que se encargará de la implementación inicial del Monkey y la dejará corriendo en su computador. recurso computacional disponible en EC2 AWS.
- **Uso de APIs:** El ingeniero senior ayudará a codificar los escenarios de prueba los cuales son ejecutados bajo demanda. Esta técnica se usa para la creación de pruebas E2E con validación de datos en los formularios de las funcionalidades a probar de Ghost.

2.4.2. Niveles

- De sistema (Objetivo a, b, c y d).

2.4.3. Tipos

- Funcionales (Objetivo a).
- Caja blanca (Objetivo b).
- Positivas (Objetivo c)
- Negativas (Objetivo d)
- Caja negra (Objetivo d)

Nivel	Tipo	Técnica	Objetivo
De sistema	Funcional	Record and Play	Implementar pruebas exploratorias y de compatibilidad (usar diferentes browsers y diferentes tamaños de viewport) para redactar documentación

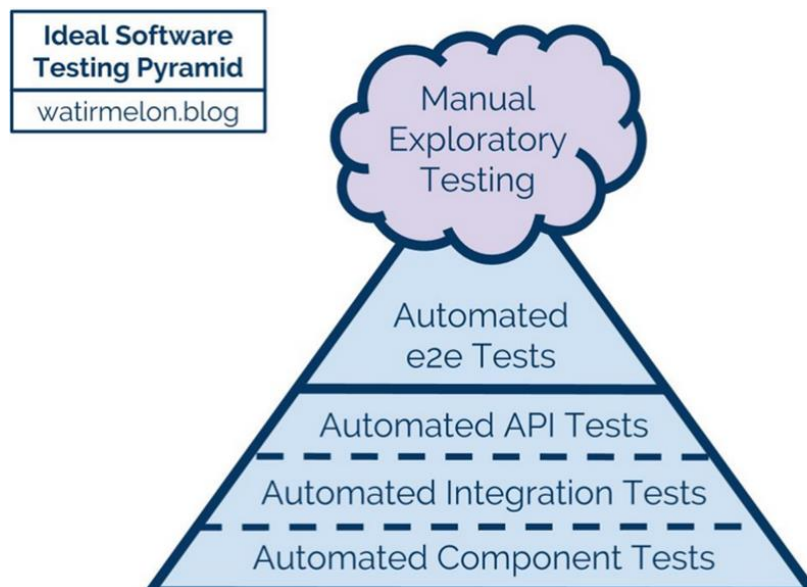
			inicial de la aplicación web Ghost durante dos semanas.
De sistema	Funcional Negativa Positiva	APIs de Automatización	Crear pruebas E2E con generación de datos para las 9 funcionalidades principales, con el propósito de validar la existencia o no de defectos funcionales del sistema durante un periodo de cuatro semanas.
De sistema	Funcional	APIs de Automatización	Buscar diferencias entre la versión anterior y actual de Ghost usando técnicas de regresión visual durante un periodo de una semana.
De sistema	Funcional Caja negra	Monkey testing	Implementar pruebas de reconocimiento enfocadas a validar las funcionalidades Core definidas mediante la generación de una gran cantidad de datos aleatorios con el fin de identificar casos raros que no sean evidentes para un ser humano durante una semana.

2.5. Distribución de Esfuerzo

Item	Horas	Recurso	Objetivo	Descripción
Pruebas exploratorias	16 horas/hombre	Ingenieros senior I, II, III y IV.	a	Pruebas que realizaran los ingenieros seniors a través de la exploración manual de la plataforma Ghost y el uso de herramientas record and play, con el objetivo principal de validar que la aplicación Web responde de manera funcional y así poder crear una documentación inicial.
Pruebas E2E con generación de datos	32 horas/hombre	Ingenieros senior I, II, III y IV.	c	Teniendo en cuenta la buena disponibilidad de personal y de horas hombre, se asignan cuatro ingenieros senior que se encargaran de realizar los escenarios de prueba E2E y ejecutarlos para todas las funcionalidades Core definidas, dedicando 28 horas por ingeniero senior.
Regresión visual	8 horas/hombre	Ingenieros senior I, II, III y IV. I	d	Se propone realizar pruebas con regresión visual para comprobar las funcionalidades de la versión más actual de la aplicación al

				compararla visualmente con su versión anterior.
Pruebas de reconocimiento	50 horas/máquina 8 horas/hombre	Máquina EC2 AWS Ingenieros senior I, II, III y IV.	b	Se propone implementar 8 horas de de cada ingeniero automatizador senior para configurar las pruebas de reconocimiento y ejecutarlas sobre Ghost con gran cantidad de datos aleatorios de manera automatizada. Además, las pruebas de reconocimiento planteadas por los ingenieros con monkeys y rippers, serán ejecutadas por la máquina virtual que correrá durante unas 200 horas. En este punto, el trabajo de los ingenieros será principalmente configurar las pruebas, realizar ajustes y evaluar las incidencias detectadas para documentarlas y escalarlas al equipo de desarrollo.

Según el plan propuesto, la estrategia apunta a una pirámide de automatización, con el presupuesto horas/hombre que se tiene se decidió hacer un mayor uso de las pruebas automatizadas con APIs, de reconocimiento, y un porcentaje menor se destinó a pruebas manuales exploratorias, con el fin de darle el enfoque a nuestra estrategia en el tema de pruebas automatizadas. Con respecto a pruebas unitarias no se asignó porcentaje a estas prueba de este tipo debido a que no se conoce muy a nivel de código la aplicación.



El objetivo es cubrir como mínimo el 90 % de las funcionalidades especificadas, para cuales para cada una de ellas se deberán definir como mínimo la estrategia, los criterios y recursos invertidos para cada una de ellas. La prioridad de las pruebas se estimará de acuerdo a su nivel de importancia.

Funcionalidades	Tipo	Niveles	Implementación
Iniciar sesión como miembro del staff	Funcional	Sistema	Automatizada
Crear una página	Funcional	Sistema	Automatizada/manual

Consultar página	Funcional	Sistema	Automatizada/manu al
Crear nuevo post	Funcional	Sistema	Automatizada/manu al
Consultar un post	Funcional	Sistema	Automatizada/manu al
Modificar página	Funcional	Sistema	Automatizada/manu al
Modificar post	Funcional	Sistema	Automatizada/manu al
Eliminar página	Funcional	Sistema	Automatizada/manu al
Eliminar post	Funcional	Sistema	Automatizada/manu al
Crear etiqueta	Funcional	Sistema	Automatizada/manu al
Crear suscripción o membresía	Funcional	Sistema	Automatizada/manu al
Registrarse como suscriptor	Funcional	Sistema	Automatizada/manu al
iniciar sesión como suscriptor	Funcional	Sistema	Manual
Recuperar contraseña	Funcional	Sistema	Automatizada/manu al
Crear nuevo miembro manualmente	Funcional	Sistema	Automatizada/manu al
Importar miembros	Funcional	Sistema	Manual
Invitar a una persona para ser miembro del staff	Funcional	Sistema	Automatizada /manual

De acuerdo con la estrategia de pruebas propuesta, la distribución de esfuerzo que se plantea es de tipo pirámide dado que se dedica mayor cantidad de esfuerzo a las pruebas automatizadas entre horas hombre y horas máquina. Esta distribución de la cantidad de pruebas obedece a más del 60% a pruebas automatizadas y el restante a pruebas manuales, dado que a pesar que contamos con una buena cantidad de recurso humano, este se enfocó a realizar pruebas y labores que puedan aportar a la documentación, grabación de escenarios y generación de scripts, mientras que por otro lado se aprovecha el recurso computacional que complementa la estrategia de pruebas mediante pruebas automatizadas de reconocimiento que pueden encontrar casos raros de pruebas que normalmente un humano no puede y, además, estas pruebas pueden correr las 24 horas del día mientras los ingenieros descansan. De esta manera, se plantea una distribución de esfuerzo que pueda aprovechar lo mejor de las pruebas manuales y las automatizadas en el periodo estipulado para ejecutar la estrategia.

2.5.1. Cronograma de trabajo

En la siguiente tabla se puede observar el cronograma de trabajo con una disposición de 8 horas por semana para los ingenieros Senior, teniendo una dedicación de 8 semanas en total por las tareas implementadas. Por otro lado, para el recurso computacional se dedican 100 horas por semana durante un periodo de 2 semanas para completar un total de 200 horas.

ESTRATEGIA DE PRUEBAS PARA APLICACIÓN GHOST									
CRONOGRAMA		Semanas							
FASES O ETAPAS	RESPONSABLES	1	2	3	4	5	6	7	8
Pruebas exploratorias	Ingeniero Automatizador I								
	Ingeniero Automatizador II								
	Ingeniero Automatizador III								
	Ingeniero Automatizador IV								
Pruebas de reconocimiento	Ingeniero Automatizador I								
	Ingeniero Automatizador II								
	Ingeniero Automatizador III								
	Ingeniero Automatizador IV								
	EC2 AWS								
Pruebas E2E con generación de datos	Ingeniero Automatizador I								
	Ingeniero Automatizador II								
	Ingeniero Automatizador III								
	Ingeniero Automatizador IV								
Revisión visual	Ingeniero Automatizador I								
	Ingeniero Automatizador II								
	Ingeniero Automatizador III								
	Ingeniero Automatizador IV								

	Ingeniero Automatizador I (8 horas/semana)
	Ingeniero Automatizador II (8 horas/semana)
	Ingeniero Automatizador III (8 horas/semana)
	Ingeniero Automatizador IV (8 horas/semana)
	EC2 AWS (200 horas/semana)