



MAESTRÍA EN INGENIERÍA DE SOFTWARE-ONLINE

CURSO DE PRUEBAS AUTOMATIZADAS

**INTEGRANTES:
PATRICK MYKODA
YOSEF LOZADA
FREDY ALEJANDRO RUSSI FAJARDO
CHRISTIAN BORRÁS TORRES**

**ENTREGA SEMANA OCHO
ESTRATEGIA DE PRUEBAS**

2022

TABLA DE CONTENIDO

1. Aplicación bajo pruebas.....	3
1.1.Nombre Aplicación.....	3
1.2. Versión	3
1.3.Descripción	3
1.4.Funcionalidades Core.....	3
1.5.Diagrama de arquitectura... ..	3
1.6.Diagrama de contexto.....	4
1.7.Modelo de datos... ..	4
1.8.Modelo de GUI.....	4
2. Contexto de la estrategia de pruebas	4
2.1. Objetivos... ..	4
2.2.Duración de la iteración de pruebas.....	5
2.3.Presupuesto de pruebas... ..	5
2.3.1. Recursos humanos.....	5
2.3.2. Recursos computacionales... ..	6
2.3.3. Recursos económicos para la contratación de servicios/personal... ..	6
2.4.TNT (Técnicas, Niveles, Tipos) de pruebas	7
2.5 Distribución de esfuerzo... ..	8
.....	

1. Aplicación Bajo Pruebas

1.1. Nombre Aplicación: GHOST

1.2. Versión: 4.47.4

1.3. Descripción:

Crea tu propia plataforma en la web. Ghost es una aplicación poderosa 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

1.4. Funcionalidades Core:

Funcionalidad	Descripción
Editar el diseño del sitio	A través de esta funcionalidad el usuario puede cambiar los atributos de las páginas de forma tal que la letra, los colores, entre otros aspectos se puedan editar.
Ingresar nuevos miembros al sistema	La aplicación permite que nuevos miembros del blog se unan, al agregarlos a través del propio sistema, diligenciando sus datos de forma manual, o importando un archivo CSV que los cree de forma automática, generando una cantidad mayor de usuarios en la aplicación.
Crear y previsualizar el Post	Al momento de crear un Post, se permite visualizar cada uno de los atributos que el usuario administrador uso, ejemplo si agrego imágenes, un video, enlaces de descarga, etc..., con el fin de darle una mayor versatilidad. Además, se puede programar la publicación con una fecha y horas sugeridas.
Configuración General	La configuración del sitio en su meta data, a gusto del usuario administrador, para cambiar valores tales como la zona horaria, las redes sociales del sitio, el lenguaje utilizado en el miso, entre otros.
Crear y previsualizar Page	Esta funcionalidad permite al usuario crear una página (Page) y visualizar un ejemplo de la publicación prevista
Crear y asignar Tag	El usuario puede crear un Tag y asignarlo a un Post o a una página. Lo último se hace dentro de la ventana de edición de un Post/de una página.
Publicar Posts y Pages	Después de haber creado un Draft de su Post o de su página, esta funcionalidad le permite al usuario publicar el contenido para compartirlo con los suscriptores y visitantes de la página.

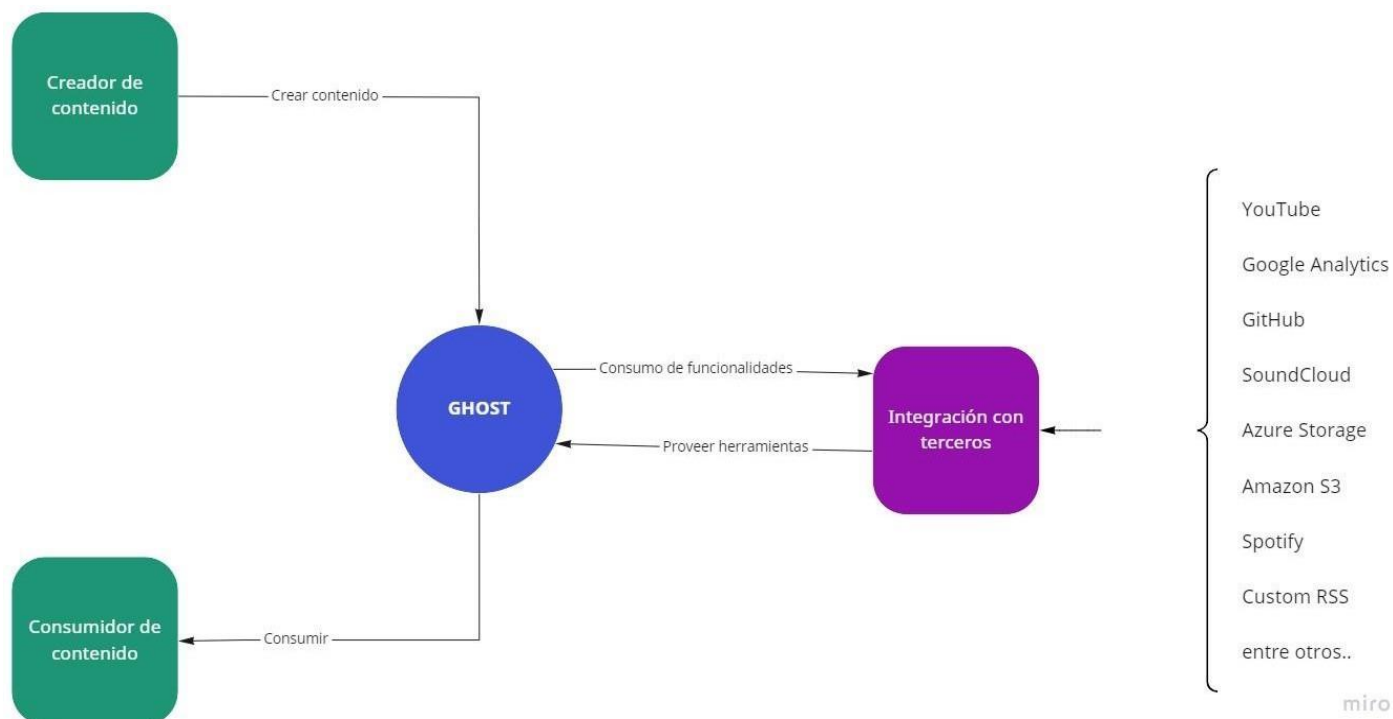
¹ Sacado el 23.04.2022 en la página: <https://ghost.org/>

1.5. Diagrama de Arquitectura:

Enlace para el Diagrama de Arquitectura, por favor entrar a través de la cuenta Uniandes para poder visualizar: [diagrama_architectura_GHOST_\(1\).pdf](#), o a través de Gmail en el siguiente enlace https://drive.google.com/file/d/11hg23rFXlo_jdFaI2Fsmc7bUp-IXqyNH/view?usp=sharing

1.6. Diagrama de Contexto:

Enlace del diagrama de contexto de la aplicación, se debe tener en cuenta que se observa solamente los elementos que componen la aplicación sin detallarlos.



1.7. Modelo de Datos:

El modelo de datos se puede consultar en el siguiente enlace https://res.cloudinary.com/dk5hasdld/image/upload/v1650828336/Modelo_datos_ghost.drawio_iwhpch.png, si no es funcional se recomienda entrar al siguiente enlace [Modelo_datos_ghost.drawio.png](#), debido al tamaño del archivo no se visualiza bien en OneDrive se debe descargar para detallarlo con mejor resolución.

1.8. Modelo de GUI:

[Modelo GUI Tercera entrega.png](#)

Por la cantidad de detalles en la gráfica se recomienda descargar el archivo para poder ver todo en la mejor resolución y más detallado.

2. Contexto de la estrategia de pruebas

2.1. Objetivos:

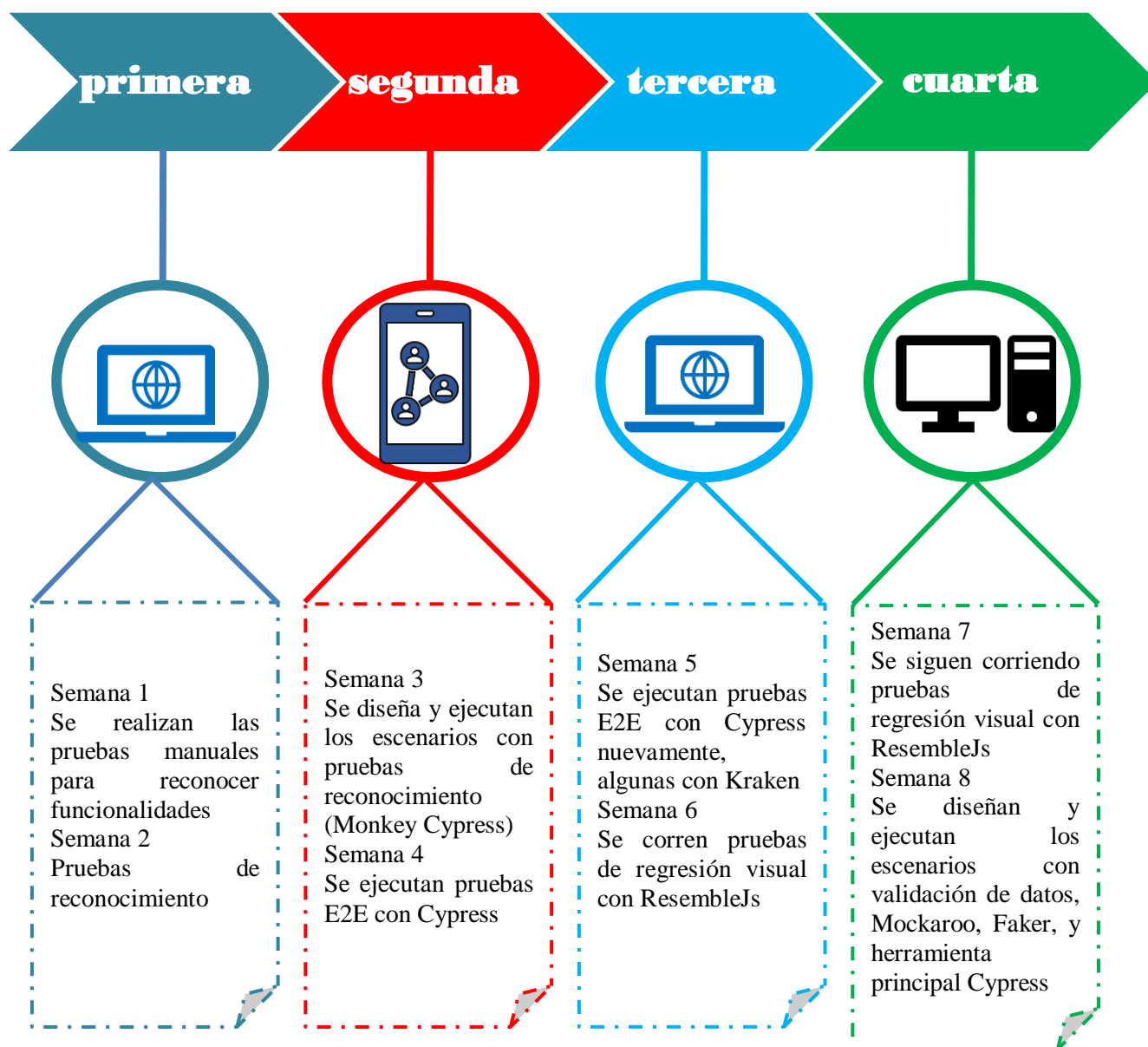
- Comprobar cada una de las funcionalidades principales
- Encontrar errores en la parte funcional de la aplicación usando valores válidos para observar si el sistema cumple con el comportamiento esperado

- Encontrar errores en la parte funcional de la aplicación usando valores inválidos
- Comprobar la funcionalidad de la interacción de dos o más componentes ejecutándose en la aplicación
- Comprobar que el flujo de la aplicación funcione correctamente de principio a fin
- Cubrir la mayor cantidad de código posible.

2.2. Duración de la iteración de pruebas

De acuerdo a lo solicitado por la empresa, el tiempo en el que se desarrollaran las pruebas es de ocho semanas, teniendo en cuenta esto, se contara con sprints con una duración de dos semanas, para un total de cuatro sprints, con una dedicación total de 256 horas entre todos los colaboradores (integrantes del grupo, cuatro ingenieros testers senior).

El siguiente gráfico muestra las pruebas que se realizarán en los cuatro sprints:



2.3. Presupuesto de pruebas:

2.3.1. Recursos Humanos

- Los integrantes del equipo de prueba serán cuatro testers senior (64 horas/persona), dado que se reparte el trabajo por persona en un tiempo determinado de ocho (8) horas/hombre por semana.

Fecha: 23.05.2022 - 17.07.2022

Horarios:

Patrick Mykoda: lunes a jueves, de 8:00pm a 10:00pm

Christian Borrás Torres: lunes a jueves, de 8:00pm a 10:00pm

Fredy Alejandro Russi Fajardo: lunes a jueves, 7:00pm a 9:00pm

Yosef Israel Lozada Gaviria: lunes a jueves, de 7:00pm a 9:00pm

No se hace uso de recurso alguno para contratar a personal adicional, no se presentó presupuesto extra monetario por parte de la empresa, por lo tanto, el equipo designado será quien llevará a cabo las pruebas por el resto de la estrategia.

2.3.2. Recursos Computacionales

- Asus AOI, Windows 10 Home Single Sistema operativo de 64 bits; procesador basado en x64: RAM 12,0 GB; NodeJSv16.14.2; 1TB;
- HP, Windows 10 Home, Sistema operativo de 64 bits; procesador basado en x64, RAM 8,0 GB, 1TB, NodeJSv16.14.2
- Lenovo-PC, Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz, 4,00 GB, Sistema operativo de 64 bits, procesador basado en x64
- ASUS VIVOBOK, Sistema operativo de 64 bits, procesador basado en x64
- Samsung Galaxy A5 (2017) SM-A520F, Android 8.0.0, 3GB RAM, 32GB memoria interna
- Samsung Galaxy A2 (2021) SM-A127M/DS, Android 11, 4GB RAM, 64GB memoria interna

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

Todos los integrantes del equipo cuentan con la experiencia requerida para el uso de las herramientas mencionadas en esta estrategia. Por lo tanto, no se requiere de la contratación de recursos externos; a continuación, se detalla el salario devengado por cada uno de los integrantes del equipo con fines informativos, ya que al pertenecer a la compañía estos no representa un costo que afecte al presupuesto.

Recursos solicitados	Costo
1. Un ingeniero automatizador senior costo mensual	\$5'025.200 * 2
2. Un ingeniero automatizador senior costo mensual	\$5'025.200 * 2
3. Un ingeniero automatizador senior costo mensual	\$5'025.200 * 2
4. Un ingeniero automatizador senior costo mensual	\$5'025.200 * 2
Total	\$40'201.600

* Se calculó el costo de los testers de acuerdo a la página talent.com, tener en cuenta que los testers son seniors y por lo tanto se consideran profesionales más experimentados, quienes según esta talent.com perciben hasta \$60.302.400 al año, por lo que se procedió a dividir este valor en 12, para obtener el salario mensual, con valor de \$5.025.200

<https://co.talent.com/salary?job=tester#:~:text=El%20salario%20tester%20promedio%20en,a%C3%B1o%20o%20%2416.615%20por%20hora.>

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

Los números corresponden al orden cronológico utilizado durante la iteración.

	Nivel	Tipo	Técnica	Objetivo
1	Pruebas Sistema	Funcional Positivo Negativo Caja negra	Pruebas manuales de manera exploratoria	Encontrar errores en la parte funcional de la aplicación usando valores válidos e inválidos para observar si el sistema cumple con el comportamiento esperado
2	Pruebas Sistema	Funcional Positivo Caja negra	Pruebas de Reconocimiento Herramientas: - Monkey-Cypress	Comprobar que el flujo de la aplicación funcione correctamente de principio a fin
3	Pruebas Sistema	Funcional Positivo Caja gris	E2E Herramientas: - API de Automatización Cypress - Librería Faker	Comprobar el flujo de navegación correspondiente a cada una de las funcionalidades principales.
4	Pruebas Sistema	Funcional Negativo Caja gris	E2E Herramientas: - API de Automatización Cypress - Librería Faker	Encontrar errores en el flujo de navegación de las funcionalidades principales ingresando valores inválidos.

5	Pruebas Integración	Caja negra	Regresión Herramientas: - ResembleJS	Verificar los cambios a nivel visual entre dos versiones de la aplicación
6	Pruebas Sistema	Funcional Positivo y Negativo Caja gris	Escenarios de validación de datos Herramientas: - API de Automatización - Cypress - Schemas de Mockaroo - Librería Faker	Verificar por si los componentes con entradas de texto o datos cumplen con los rangos frontera establecidos, así como con valores e inválidos.

2.5. Distribución de Esfuerzo*

Semana	Tarea	Responsables	Horas/Persona
1	Diseñar y ejecutar escenarios de pruebas manuales	Ingeniero Automatizador de Pruebas Senior (4)	8
2	Diseñar y ejecutar escenarios de pruebas de reconocimiento (Monkey Cypress)	Ingeniero Automatizador de Pruebas Senior (4)	8
3	Diseñar y ejecutar escenarios de prueba de reconocimiento (Monkey Cypress)	Ingeniero Automatizador de Pruebas Senior (4)	8
4	Diseñar y ejecutar escenarios de pruebas de extremo a extremo (Cypress)	Ingeniero Automatizador de Pruebas Senior (4)	8
5	Diseñar y ejecutar escenarios de pruebas de extremo a extremo (Cypress)	Ingeniero Automatizador de Pruebas Senior (4)	8
6	Diseñar y ejecutar escenarios de pruebas de regresión visual (v4.10.0 vs v4.47.4)	Ingeniero Automatizador de Pruebas Senior (4)	8
7	Diseñar y ejecutar escenarios de pruebas de regresión visual (v4.10.0 vs v 4.47.4)	Ingeniero Automatizador de Pruebas Senior (4)	8

8	Diseñar y ejecutar escenarios de validación de datos	Ingeniero Automatizador de Pruebas Senior (4)	8
---	--	---	---

*Considerando la división de trabajo y tareas del cuadro anterior se puede evidenciar que el patrón escogido para el desarrollo de esta iteración de pruebas es el de pirámide, dado que la proporción de pruebas de integración, aceptación y componente es mayor que las pruebas automatizadas de GUI y las exploratorias.

Video Justificación

Enlace de YouTube, contenido de la justificación de la estrategia de pruebas: <https://youtu.be/kiXp4smj91E>

De igual forma se adjunta enlace de OneDrive: https://uniandes-my.sharepoint.com/:v:/g/personal/y_lozada_uniandes_edu_co/EUQyiq-XM3IPhq-Jj2oScUIBFgn9-rkSYbe5zRGxDPwykg?e=1mKYaG