

Estrategia de Pruebas

1. Aplicación Bajo Pruebas

1.1. Nombre Aplicación: GHOST

1.2. Versión: 5.69.0

1.3. Descripción:

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, crear blogs, enviar boletines y ofrecer suscripciones pagas a los miembros.

1.4. Funcionalidades Core:

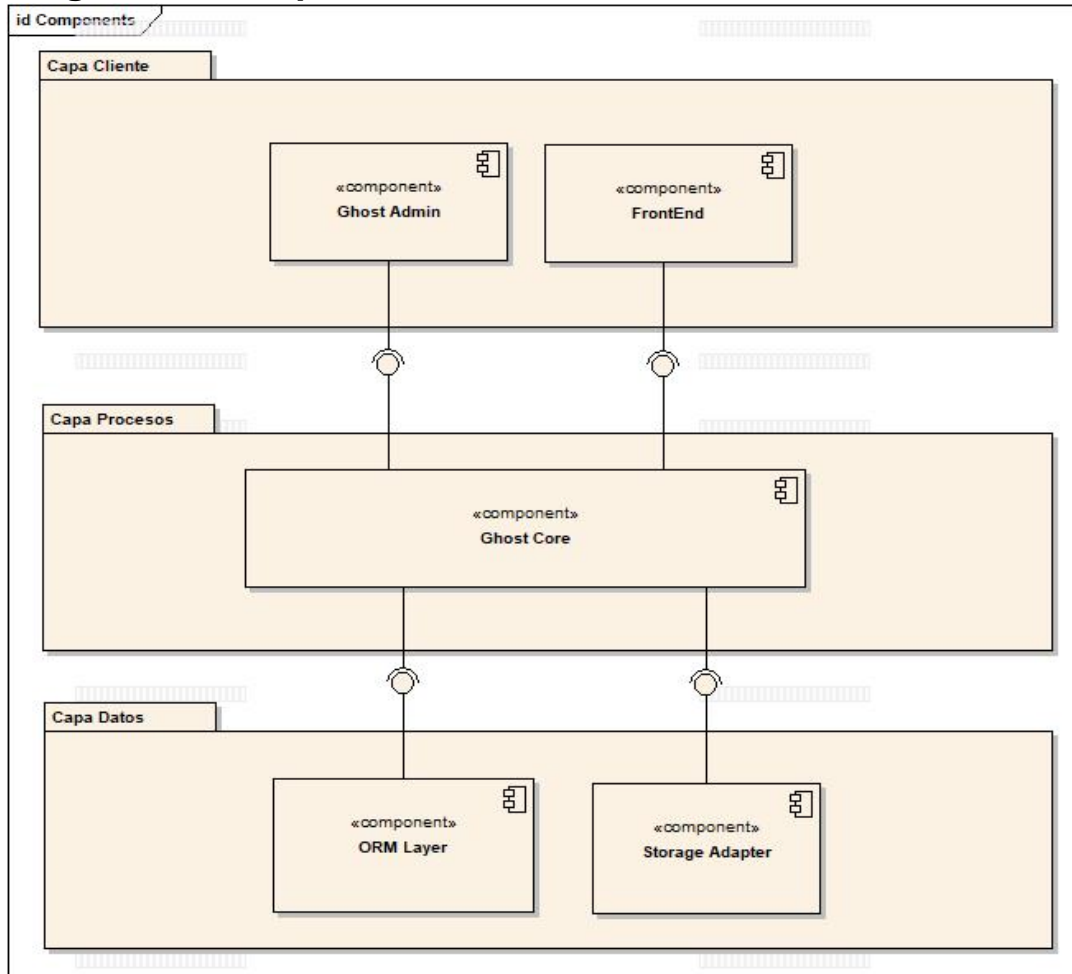
- **Publicación.-** Las publicaciones se crean dentro de Ghost-Admin utilizando el editor para determinar el contenido principal de su sitio. Dentro de ellos se encuentran todos los campos que puede esperar, como título, descripción, slug, metadatos, autores, etiquetas, etc.

Además, las publicaciones tienen campos de Inyección de código, lo que significa que puede registrar estilos adicionales, scripts u otro contenido para inyectar justo antes `</head>` o `</body>` en cualquier URL en particular donde lo desee.

- ✎ **Drafts.-** Guarda los post editados sin publicar.
- ✎ **Scheduled.-** Permite programar la publicación de los posts y a que privacidad tengan.
- ✎ **Membresías.-** Cualquier usuario que desee ofrecer una forma para que su audiencia apoye su trabajo puede usar la función Miembros para compartir contenido, crear una audiencia y generar ingresos de un negocio de membresía.
- ✎ **Boletines por correo electrónico.-** Los boletines por correo electrónico en Ghost se pueden programar y enviar a miembros gratuitos y de pago, o a un segmento de miembros gratuitos o de pago. Los boletines se entregan utilizando una hermosa plantilla HTML que está estandarizada para los clientes de correo electrónico más populares.

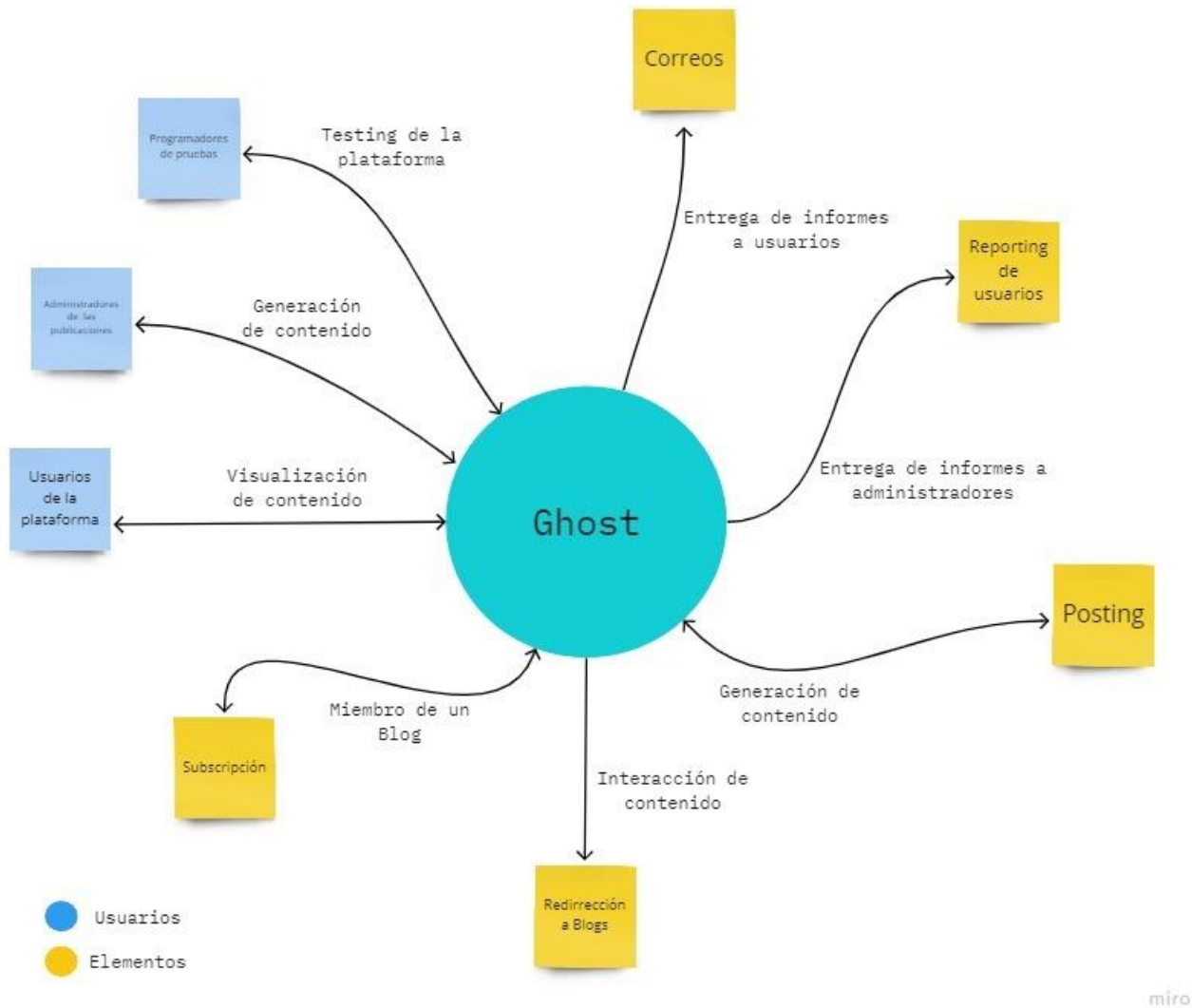
Los sitios Ghost tienen un solo boletín de forma predeterminada, pero se pueden crear y personalizar otros adicionales. Múltiples boletines le permiten personalizar el contenido para audiencias específicas y sus miembros pueden elegir qué contenido reciben.

1.5. Diagrama de Arquitectura:



- **GHOST ADMIN.-** Ghost-Admin es una interfaz de administración potente y opcional para editores. Facilita el flujo de trabajo editorial y ofrece un editor de alta calidad. Permite la gestión de contenido con acceso de lectura y escritura, brindando autenticación segura basada en roles.
- **FRONTEND.-** Ghost es independiente de cualquier interfaz en particular y su interfaz es opcional e intercambiable. Puede crear su propia interfaz completamente personalizada, especialmente útil si utiliza Ghost como servicio para aplicaciones móviles o nativas.
- **CORE.-** Ghost es una API RESTful JSON que permite la creación, administración y recuperación de contenido de publicaciones de manera sencilla. Divide la API en Contenido y Administración, cada una con sus métodos de autenticación y herramientas para diversos casos de uso.
- **ORM LAYER.-** Ghost utiliza la capa ORM Bookshelf.js por defecto y es compatible con una variedad de bases de datos, siendo SQLite3 la predeterminada en desarrollo y MySQL recomendada para producción. **STORAGE ADAPTER.**
- **Storage Adapter.-** Ghost usa almacenamiento local de manera predeterminada, pero los usuarios tienen la opción de emplear adaptadores de almacenamiento personalizados que permiten tener un sistema de archivos completamente externo. Además, existen numerosos adaptadores de almacenamiento prefabricados listos para ser utilizados en Ghost.

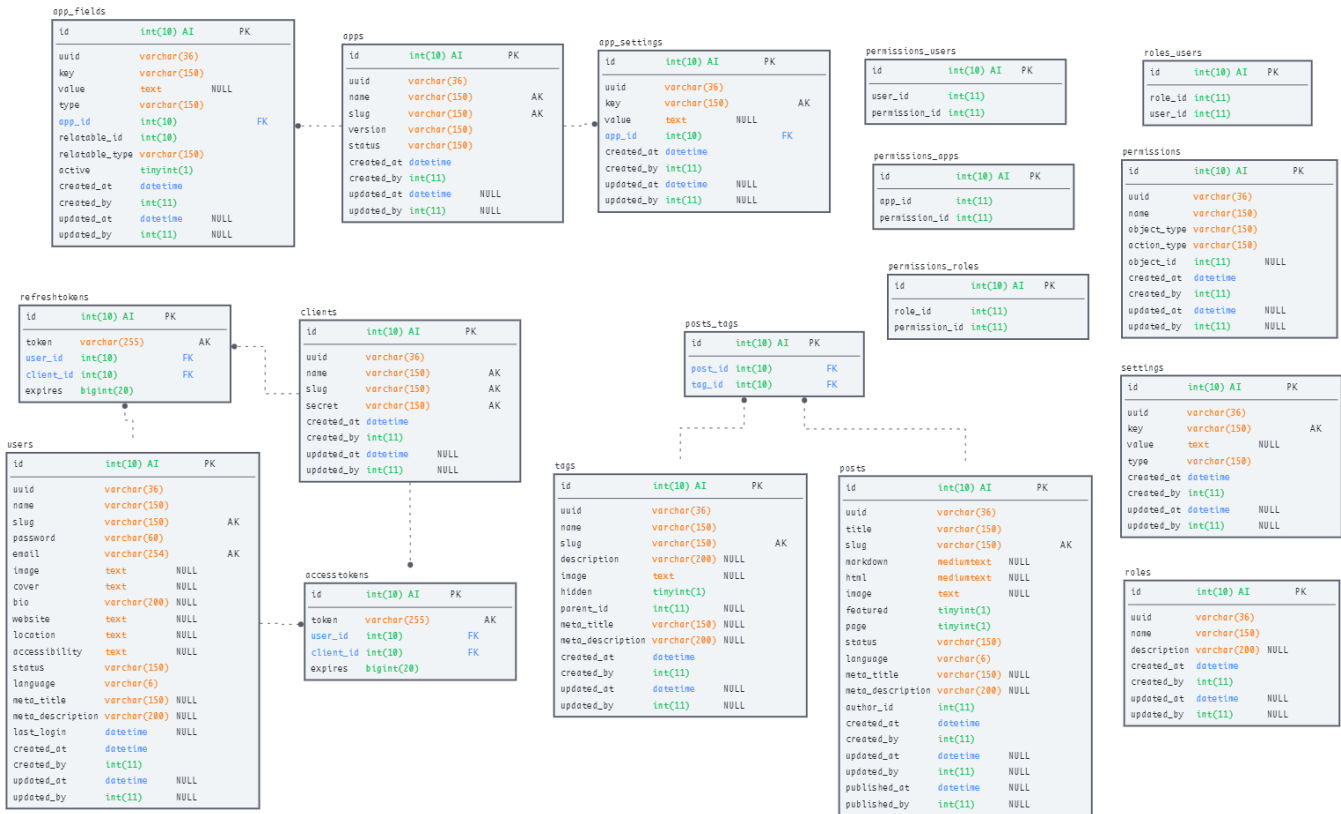
1.6. Diagrama de Contexto:



Obsérvela con mejor resolución aquí:

<https://drive.google.com/file/d/17qP8CrCwFYCjOB6ztcgNv2B3ahLGzIz/view?usp=sharing>

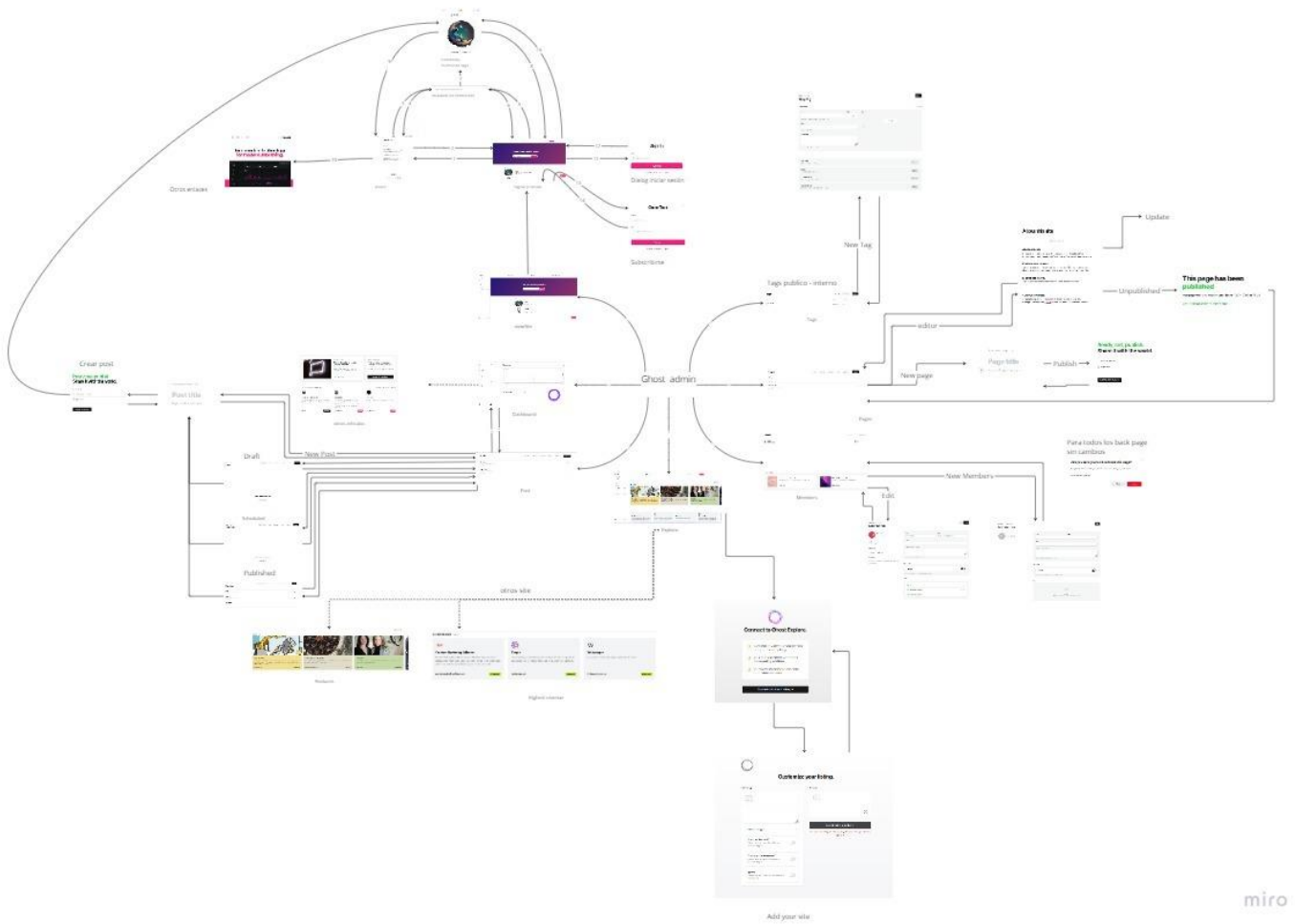
1.7. Modelo de Datos:



Fuente: <https://github.com/TryGhost/Ghost-Config/blob/master/ghost-versions/0.6.4/schema.sql>

Obsérvela con mejor resolución aquí: <https://ibb.co/XtrZjtk>

1.8. Modelo de GUI:



Obsérvela con mejor resolución aquí:

https://drive.google.com/file/d/1latEqXxOnG8CPAtQ5etrpddbg_2xN1Y_/view?usp=sharing

2. Contexto de la estrategia de pruebas

2.1. Objetivos:

- Comprobar el correcto funcionamiento de las funcionalidades.
- Detectar fallos o bugs en las funcionalidades implementadas.

2.2. Duración de la iteración de pruebas:

Para la iteración de pruebas se propone realizar una labor de un sprint de una semana (7 días) en los cuales se tendrán 50 horas por persona y 200 horas máquina para realizar las pruebas automatizadas las cuales se distribuyen entre todos los integrantes del equipo

- Fecha Inicio: 06/11/2023
- Fecha Fin: 12/11/2023

2.3. Presupuesto de pruebas:

2.3.1. Recursos Humanos

El tiempo de duración esta acotado a 1 sprint y el conjunto de testers a 3 automatizadores junior tienen una experiencia incipiente en la automatización de pruebas de software, con experiencia y conocimiento de la aplicación GHOST, por tanto, se dedicarán 7,14 horas diarias durante el sprint de una semana, equivalente a 7 días, con el siguiente detalle.

Equipo	Experiencia	Días	Horas
automatizadores junior 1	junior	7	7,14
automatizadores junior 2	Junior	7	7,14
automatizadores junior 3	junior	7	7,14

2.3.2. Recursos Computacionales

Maquina virtual para hacer las pruebas

- Fuente: <https://aws.amazon.com/es/ec2/pricing/on-demand/>
- Nombre de la Instancia: m6g.2xlarge
- Sistema Operativo: Linux
- VCPU: 8
- Memoria: 32 GiB
- Almacenamiento: Solo EBS
- Rendimiento Red: Hasta 10 gigabits
- Precio: \$ 0,308/Hora
- Utilización: Se instalará el CMS GHOST conjuntamente con Cypress y Playwright para las pruebas de componentes, integración y e2e, debido a que el equipo tiene conocimiento en estas herramientas.

Estación de trabajo para cada automatizador jr:

- Sistema Operativo: Linux
- Pantalla: 19"

- CPU: Core i7 9na gen
- Memoria: 16 GiB
- Almacenamiento: Ssd Nvme M 2 de 250 GB
- Unidades: 3
- Utilización: Utilizaremos las estaciones de trabajo existentes para escribir los scripts de prueba y conectarnos con los máquinas virtuales en Amazon AWS

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

- TRM= 1 dólar = \$4.108,50 Pesos
- 200 horas/máquina en Amazon AWS
- Fuente: <https://co.talent.com/salary?job=automatizador>
- Costo del salario de automatizador junior: \$ 1.034,44 mes / \$ 5,68 hora
- Costo por persona: \$ 5,68 * 50 horas = \$ 284,19 dólares
- Costo ingenieros: (\$ 284,19 *3 Ingenieros Junior) = \$ 852,57 dólares
- Fuente: <https://aws.amazon.com/es/ec2/pricing/on-demand/>
- Costo total Amazon AWS EC2: \$ 0,308 X 200 = \$ 61.6
- Costo total: \$ 852,57+ \$ 61.6 = \$914.17 dólares

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

Nivel	Tipo	Tecnica	Objetivo
componente	funcionales	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
componente	caja negra	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
componente	caja negra	manual	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
componente	positivas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
componente	negativas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
componente	negativas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
integracion	funcionales	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
integracion	caja negra	manual	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
integracion	caja negra	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
integracion	positivas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
integracion	negativas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
e2e	funcionales	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
e2e	caja negra	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.

e2e	positivas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	funcionales	manual	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	caja negra	manual	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	positivas	manual	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	funcionales	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	caja negra	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	positivas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.
exploratorias	negativas	automatizada	- Comprobar el correcto funcionamiento de las funcionalidades. - Detectar fallos o bugs en las funcionalidades implementadas.

(*) Pruebas de Unidad: Se van a obviar debido a que estamos probando un producto terminado, y las pruebas unitarias ya se realizaron en la fase de desarrollo del software.

(*) Se deciden realizar las pruebas de forma manual por la complejidad de implementar su automatización, esto debido a la poca experiencia de los automatizadores junior.

(*) La distribución de pruebas utilizada es la de pirámide de automatización esto debido a la cantidad de pruebas utilizadas en cada nivel y para obtener un mayor rendimiento de los recursos que tenemos a nuestra disposición (automatizadores junior y equipos en Amazon AWS):

exploratorias	3
e2e	2
integración	4
componente	11

(*) Las pruebas exploratorias se realizan de forma manual y automática mediante la utilización de monkeys y rippers, pondremos una mayor dedicación a este último dado que es una técnica más sofisticada con la cual podremos obtener mejores resultados.

2.5. Distribución de Esfuerzo

Listado de Pruebas:

ID	Descripcion de la prueba	Tipo	Tecnica
PC1	suscribirse al sitio	componente	automatizada
PC2	crear post	componente	automatizada
PC3	publicar post	componente	automatizada
PC4	cronogramar publicacion de post	componente	automatizada
PC5	listar posts	componente	automatizada
PC6	listar pages	componente	automatizada
PC7	crear page	componente	automatizada
PC8	login de usuario administrador	componente	automatizada
PC9	modificar titulo del sitio	componente	automatizada

Plantilla elaborada por

THE SW DESIGN LAB

PC10	modificar tema del sitio	componente	automatizada
PC11	modificar barra de navegacion	componente	automatizada
PI1	Logear y suscribirse	integracion	automatizada
PI2	crear y publicar post	integracion	automatizada
PI3	crear y cronogramar publicacion post	integracion	automatizada
PI4	Crear usuario y loguearse	integracion	automatizada
PE2E1	crear usuario admin, crear post, publicar post, logout	e2e	automatizada
PE2E2	crear usuario admin, cambiar tema del sitio, logout	e2e	automatizada
PEX1	Creacion y publicacion de posts.	exploratoria	manual
PEX2	exploracion con ripper	exploratoria	automatizada
PEX3	exploracion con monkey	exploratoria	automatizada

Distribución de esfuerzo por persona y máquina.

Equipo	30 Oct.	31 Oct.	1 Nov.	2 Nov.	3 Nov.	4 Nov.	5 Nov.
J1	CM+PE2E1	PC3	PE2E2	PI2	PC6	PC9	VR
J2	PC1+PC2	PEX2	PC4	PI3	PC7	PC10	VR
J3	PI1 + PEX1	PEX3	PC5	PI4	PC8	PC11	VR
Maquina 1	CM+EAP	EAP	EAP	EAP	EAP	EAP	N/A
Maquina 2	CM+EAP	EAP	EAP	EAP	EAP	EAP	N/A
Maquina 3	CM+EAP	EAP	EAP	EAP	EAP	EAP	N/A

* Convenciones:

J#	Automatizador Junior #
CM	Configuracion Maquina
EAP	Ejecucion Pruebas Automatizadas
VR	Validar Resultados

VIDEO: <https://www.youtube.com/watch?v=P0cFu-4VtGI>

