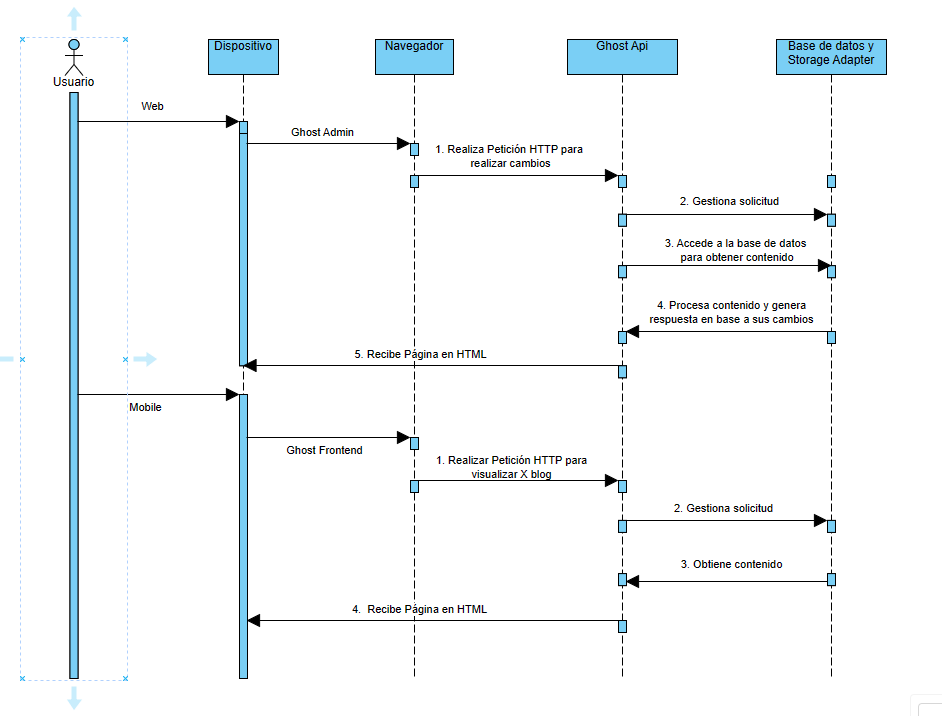
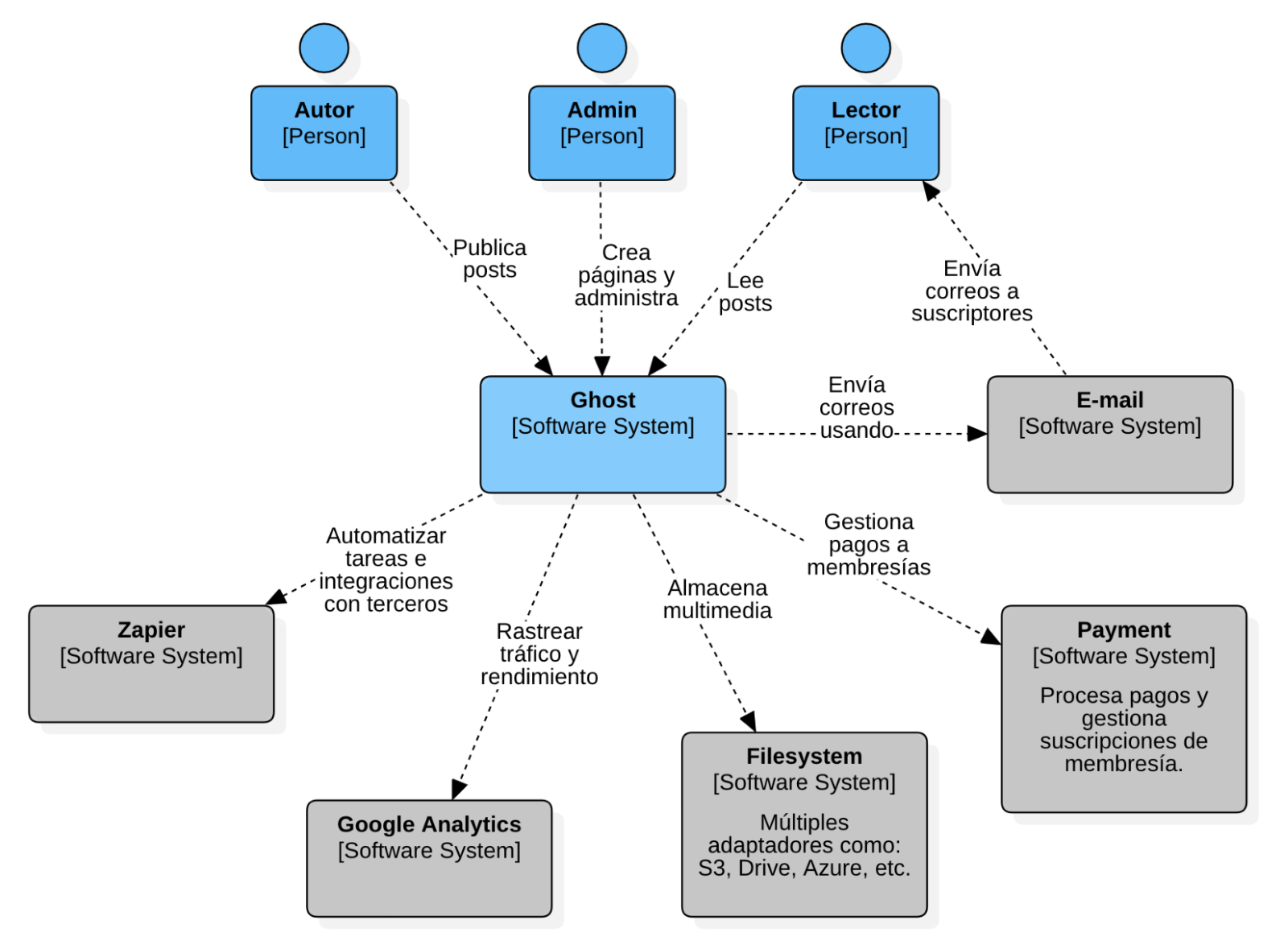
**Estrategia de Pruebas**

1. **Aplicación Bajo Pruebas**
   1. **Nombre Aplicación: Ghost**
   2. **Versión: v5.82.2**
   3. **Descripción**: Ghost es una plataforma de blogging y publicación de contenido con características avanzadas como editoriales, membresías, recomendaciones, boletines informativos por correo electrónico y enfoque en la seguridad.
   4. **Funcionalidades Core**:
      1. Actualizar información básica de perfil: Permite modificar la información personal del perfil, como nombre, biografía, correo electrónico e imagen de perfil.
      2. Edición de título y descripción del sitio: Permite cambiar el título y la descripción del sitio web para mejorar la marca y la visibilidad en motores de búsqueda.
      3. Cambio de lenguaje de publicación: Permite seleccionar el idioma de publicación del contenido, afectando la interfaz de usuario y los textos predefinidos.
      4. Actualización de redes sociales: Permite agregar o modificar enlaces a perfiles de redes sociales, facilitando la conexión y el intercambio de contenido.
      5. Creación de newsletter: Permite configurar y gestionar newsletters, incluyendo la creación de listas de suscriptores y el diseño de plantillas de correo.
   5. **Diagrama de Arquitectura:**

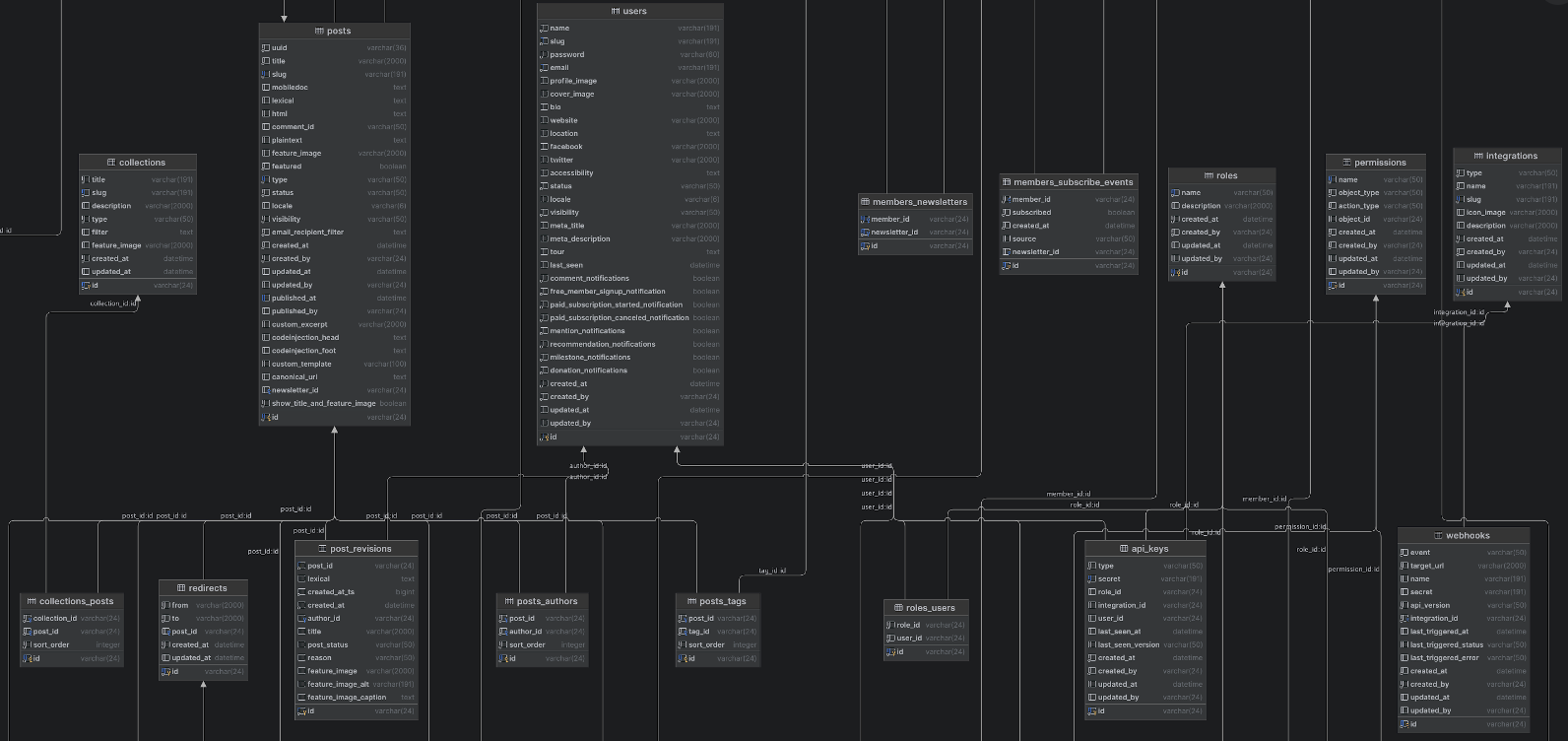


* 1. **Diagrama de Contexto:**



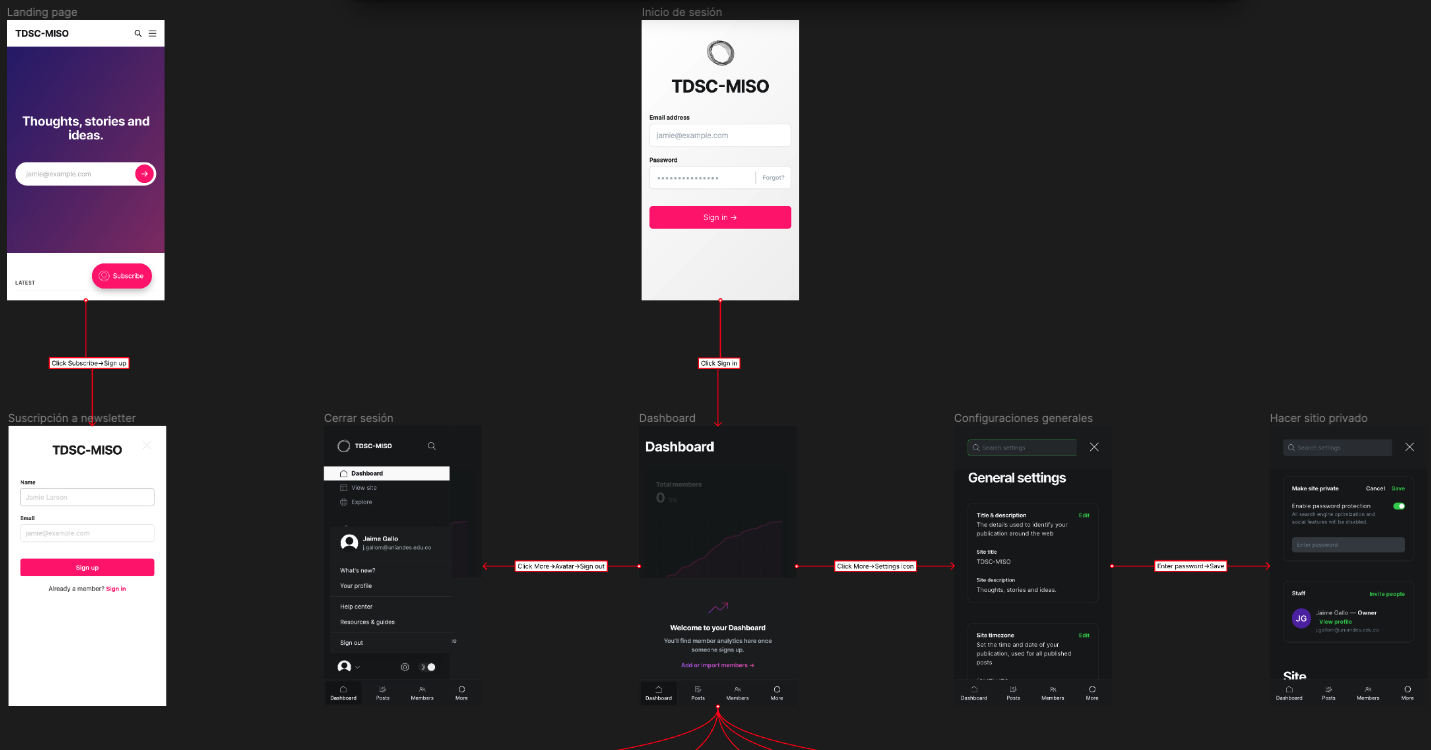
* 1. **Modelo de Datos:**

Se puede consultar el modelo completo en el siguiente enlace: <https://drive.google.com/file/d/15vwiidU40SIoZG0HZxuw8ItVPvaRYRuz/view?usp=sharing>



* 1. **Modelo de GUI:**

[**Ver modelo GUI completo en Figma**](https://www.figma.com/file/6FVhMm7mO4dfNkFMursJkQ/Pruebas-automatizadas---Modelo-GUI---Semana-3?type=design&node-id=0%3A1&mode=design&t=FlysVuApUkkG6YaJ-1)

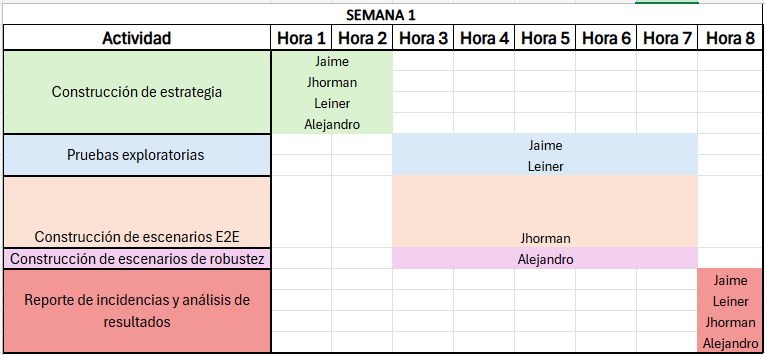


1. **Contexto de la estrategia de pruebas**
   1. **Objetivos:**
      1. **Semana 1:** Encontrar el mayor número de defectos en la configuración general del sitio.

La justificación para escoger la estrategia de esta semana radica en que ya contamos con las funcionalidades principales de Ghost implementadas y validadas. Sin embargo, aún falta una funcionalidad crítica: la configuración general del sitio. Esta funcionalidad es esencial para asegurar que los administradores puedan personalizar y gestionar adecuadamente el sitio según sus necesidades.

Por lo tanto, hemos decidido enfocarnos en realizar pruebas exhaustivas sobre las opciones de configuración del sitio. Específicamente, llevaremos a cabo las siguientes pruebas:

* + **Pruebas Manuales exploratorias:** Para explorar de manera exhaustiva las diferentes opciones de configuración, identificar defectos ocultos y comprender mejor cómo los usuarios interactuarán con estas funcionalidades.
  + **Pruebas de extremo a extremo (E2E):** Para garantizar que las configuraciones realizadas en la sección de configuración del sitio se reflejen correctamente en todas las áreas del sistema, asegurando una integración completa y funcionalidad consistente en todo el producto.
  + **Pruebas de robustez:** Para evaluar cómo la configuración del sitio maneja situaciones límite y condiciones adversas, asegurando que el sistema siga siendo estable y funcional incluso bajo circunstancias poco comunes o extremas.

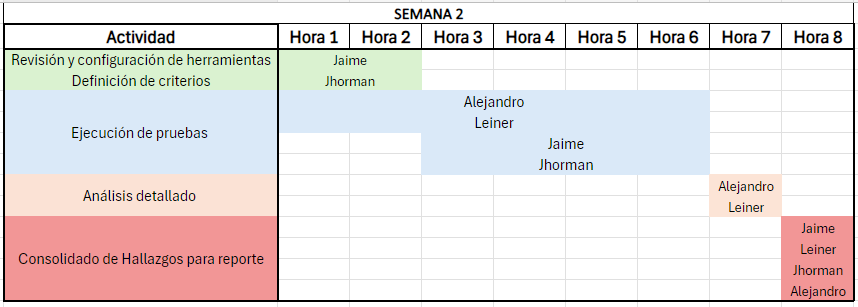


La realización de estas pruebas es fundamental para garantizar que la funcionalidad de configuración del sitio esté completamente operativa, sea intuitiva y libre de errores antes de su despliegue final. Esto no solo mejorará la calidad general del producto, sino que también aumentará la satisfacción y confianza de los usuarios en la plataforma.

* + 1. **Semana 2:** Realizar una auditoría completa de accesibilidad del sitio web utilizando herramientas como WAVE y AXE, y reportar al menos el 80% de los problemas identificados para garantizar que la sección de configuración del sitio sea accesible para usuarios con discapacidades.

El cumplimiento de normativas legales y estándares como las pautas WCAG (Web Content Accessibility Guidelines) son de alta importancia para evitar sanciones y asegurar el cumplimiento de accesibilidad. No solo es por reducir los riegos de sanción sino mostrar un cumplimiento del software frente a la inclusión y responsabilidad social. Además, debe considerarse que la accesibilidad mejora la experiencia de uso de todos los usuarios, lo que aumenta la satisfacción de los usuarios, mejorar conversiones y fidelidad de los clientes al usar la herramienta. En este caso por la capa de configuración específica del cada usuario.

Este tipo de pruebas, sobre accesibilidad, también permite expandirse en el mercado, considerando la población que vive con algún tipo de discapacidad y garantizar la accesibilidad de este grupo. Sobre todo, en una de sección del software donde se manejan convenciones generales y hasta colores.

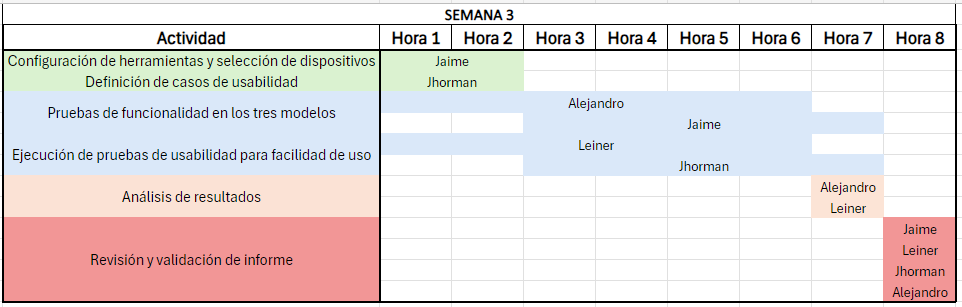


Teniendo en cuenta el anterior cronograma de actividades se espera obtener un reporte con, al menos, el 80% de los problemas de accesibilidad web sobre la zona de configuración del sitio. Con el análisis tener una propuesta parcial de solución que asegure el estándar de accesibilidad para esta zona del sitio.

* + 1. **Semana 3:** Ejecutar pruebas exhaustivas de usabilidad y funcionalidad en dispositivos móviles utilizando BrowserStack para identificar posibles errores que aseguren que el sitio web sea completamente funcional y fácil de usar en al menos tres modelos diferentes de smartphones y tablets, cubriendo tanto sistemas operativos iOS como Android.

La creciente demanda de utilización de diferentes dispositivos móviles para acceder a internet hace relevante que los sitios web sean funcionales y adaptables para smartphones y tablets. Realizar pruebas en diferentes modelos y sistemas operativos garantiza que el sitio ofrezca una experiencia de usuario consistente y de calidad independientemente del dispositivo de uso. Esto mejora la satisfacción del usuario, retención y fidelidad del cliente.

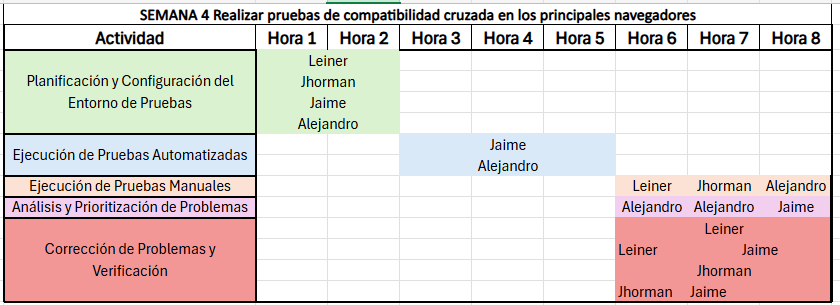
Esta diversidad de dispositivos y versiones de sistemas operativos en el mercado hace de suma importancia realizar verificaciones para que el sitio web funcione de manera correcta. El uso de herramientas como BrowserStack, permite la simulación de diferentes dispositivos y sistemas sin la disposición física de estos. Esto permite identificar y corregir problemas de compatibilidad y usabilidad de manera eficiente.



Al final de la entrega se espera una comprobación de la funcionalidad sobre el sitio web y su facilidad de uso en al menos tres modelos diferentes. Identificación de problemas de usabilidad y funcionalidad en IOS y Android.

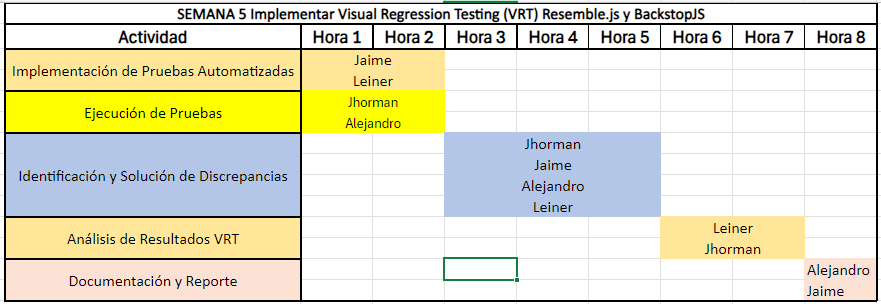
* + 1. **Semana 4:** Realizar pruebas de compatibilidad cruzada en los principales navegadores (Chrome, Firefox, Safari, Edge) y corregir los problemas identificados para asegurar una experiencia de usuario consistente en al menos el 95% de los navegadores utilizados por los visitantes del sitio.

La realización de pruebas de compatibilidad cruzada en los principales navegadores y la subsiguiente corrección de problemas identificados en la semana 4 son pasos cruciales para garantizar una experiencia de usuario coherente en una amplia gama de plataformas. Al abordar la compatibilidad con Chrome, Firefox, Safari y Edge, se busca asegurar que el sitio funcione sin contratiempos para al menos el 95% de los visitantes, optimizando así su accesibilidad y usabilidad. Esto no solo contribuye a la satisfacción del usuario, sino que también fortalece la credibilidad del sitio al demostrar su compromiso con la calidad y la atención a los detalles en todos los aspectos de su funcionamiento.



* + 1. **Semana 5:** Implementar Visual Regression Testing (VRT) para la versión 5.82.11 de Ghost utilizando herramientas como Resemble.js y BackstopJS, y solucionar cualquier discrepancia visual significativa identificada para garantizar la integridad visual del sitio tras actualizaciones o cambios.

La implementación de Visual Regression Testing (VRT) es crucial para mantener la calidad visual y la consistencia de la apariencia del sitio web a medida que se introducen actualizaciones y cambios en el código. Utilizar herramientas como Resemble.js y BackstopJS permite detectar de manera automática y eficiente cualquier discrepancia visual significativa, minimizando el riesgo de errores humanos y mejorando la precisión en la identificación de problemas.



**Consistencia Visual:**

* + **Impacto de los Cambios:** Cada actualización del código puede alterar la apariencia del sitio. Las herramientas de VRT identifican estos cambios rápidamente, asegurando que la experiencia visual del usuario no se vea comprometida.
  + **Experiencia de Usuario:** Una experiencia consistente y sin interrupciones es vital para la satisfacción del usuario. Cambios inesperados en la apariencia pueden generar desconfianza y frustración.

**Eficiencia y Precisión:**

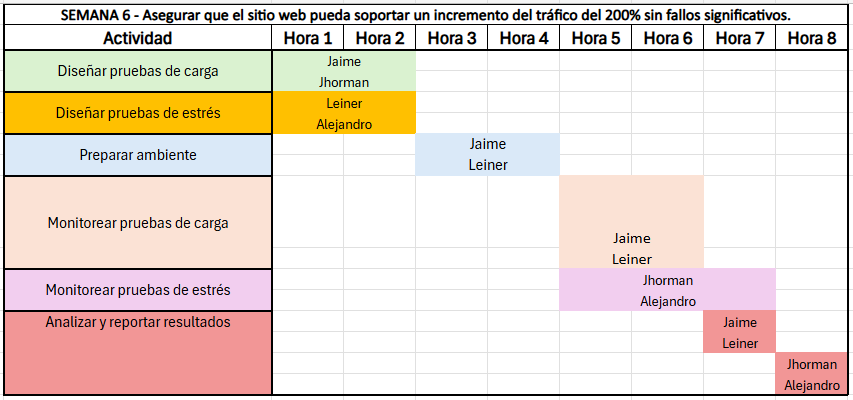
* + **Automatización:** Herramientas como Resemble.js y BackstopJS automatizan las pruebas visuales, comparando capturas de pantalla antes y después de los cambios, ahorrando tiempo y aumentando la precisión.
  + **Corrección Rápida de Errores:** La rápida identificación de discrepancias visuales permite a los desarrolladores corregir problemas de inmediato, manteniendo la alta calidad visual en cada actualización.

**Seguridad en el Despliegue:**

* + **Confianza en las Actualizaciones:** Implementar VRT proporciona seguridad adicional en los despliegues, permitiendo a los desarrolladores lanzar nuevas versiones con mayor confianza.
  + **Mantenimiento de la Integridad Visual:** Ghost requiere que cada actualización mantenga la integridad visual del sitio. Las pruebas de regresión visual aseguran que las mejoras y nuevas funcionalidades no comprometan la apariencia establecida.

Al final de la semana 5, se espera entregar la configuración completa de las herramientas Resemble.js y BackstopJS, una lista documentada de escenarios de prueba visual, y los scripts automatizados necesarios. Además, se incluirán los resultados de la ejecución inicial de las pruebas, un informe detallado de cualquier discrepancia visual encontrada, y un registro de las correcciones realizadas para garantizar la integridad visual del sitio tras actualizaciones o cambios.

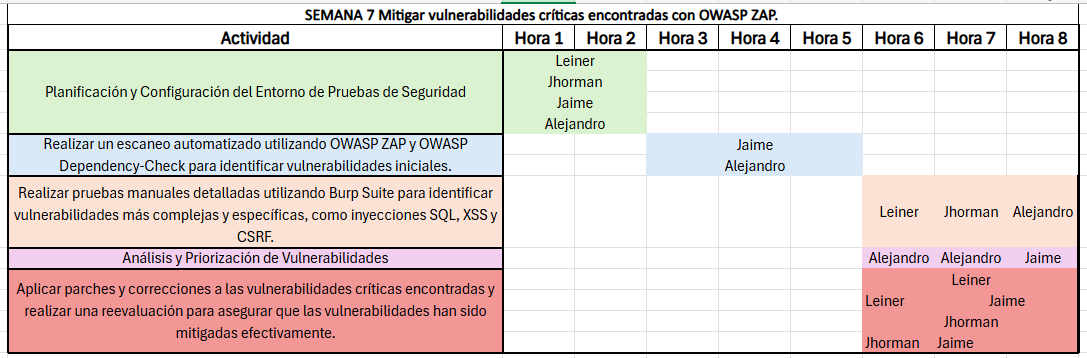
* + 1. **Semana 6:** Diseñar y ejecutar pruebas de estrés y de carga, para evaluar cómo el sistema maneja situaciones extremas y condiciones adversas, asegurando que el sitio web pueda soportar un incremento del tráfico del 200% sin fallos significativos.



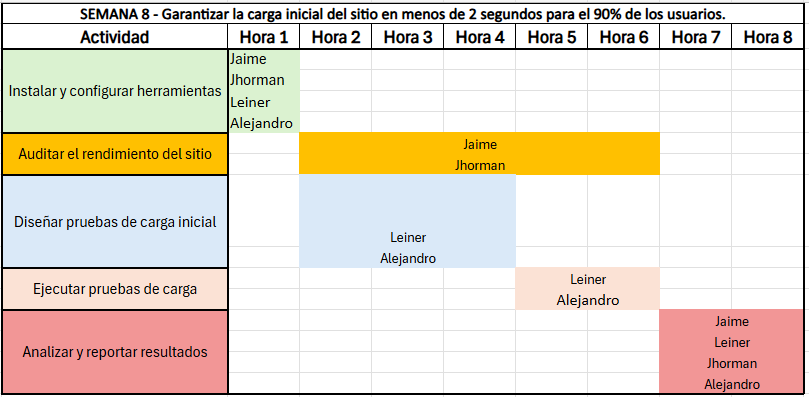
Este objetivo es crucial para garantizar que el sitio web mantenga su rendimiento y estabilidad bajo condiciones de tráfico extremo, evitando fallos significativos que puedan afectar la experiencia del usuario y la reputación del sitio, especialmente durante picos inesperados de visitas.

* + 1. **Semana 7:** Realizar una evaluación de seguridad completa utilizando las directrices y herramientas de OWASP, como OWASP ZAP, para identificar y mitigar las principales vulnerabilidades de seguridad (por ejemplo, inyecciones, XSS, CSRF), y aplicar parches a todas las vulnerabilidades críticas encontradas.

Este objetivo representa un paso fundamental para garantizar la integridad y protección del sitio web. Al identificar y mitigar las principales vulnerabilidades de seguridad, como inyecciones, XSS y CSRF, se fortalece significativamente la defensa del sistema contra potenciales ataques maliciosos. La aplicación de parches a todas las vulnerabilidades críticas descubiertas no solo protege los datos confidenciales de los usuarios y la integridad del sitio, sino que también salvaguarda la reputación de la empresa al demostrar su compromiso con la seguridad cibernética y la privacidad de los usuarios.



* + 1. **Semana 8:** Ejecutar pruebas de rendimiento utilizando herramientas como Google Lighthouse y JMeter para medir y optimizar los tiempos de carga del sitio web, con el objetivo de reducir el tiempo de carga total a menos de 2 segundos para el 90% de los usuarios.



Optimizar los tiempos de carga del sitio web es esencial para mejorar la experiencia del usuario, reducir la tasa de abandono y aumentar la satisfacción general. Un tiempo de carga rápido también contribuye positivamente al SEO, aumentando la visibilidad y el tráfico del sitio.

* 1. **Duración de la iteración de** **pruebas :** 
     1. **Recursos Humanos:** 4 Ingenieros Senior
     2. **Horas disponibles:** 8 Horas semanales por ingeniero
     3. **Total, de horas semanales:** 4 ingenieros \* 8 / Ingeniero = 32 horas
     4. **Fechas:** Inicio de estrategia: 20 de mayo, Fecha final de estrategia: 12 de julio
  2. **Presupuesto de pruebas**
     1. **Recursos Humanos**

**Ingenieros de Automatización Senior**

* + **Experiencia previa:** Los ingenieros senior de automatización de software con más de 5 años de experiencia han trabajado en diversas industrias, liderando equipos de QA y gestionando proyectos complejos. Han implementado estrategias de automatización que mejoraron la calidad del software y redujeron los tiempos de prueba, demostrando habilidad para resolver problemas y optimizar procesos. Sus conocimientos en metodologías ágiles y DevOps ha sido clave para garantizar la excelencia en la entrega de productos.
  + **Descripción de cargo:** Los ingenieros de automatización seniors se encargarán de diseñar y ejecutar casos de prueba automatización para las siguientes funcionalidades de Ghost:
    - Actualizar información básica de perfil
    - Edición de título y descripción del sitio
    - Cambio de lenguaje de publicación
    - Actualización de redes sociales
    - Creación de newsletter
  + **Tiempo disponible:** Estarán disponibles durante 8 horas semanales cada uno.
  + **Estudio:**

El estudio sirve para mostrar un acercamiento promedio a los costos reales dentro de la contratación, lo que permitirá realizar un estimado del costo total para la estrategia. Estos datos fueron un promedio obtenido de: <https://co.talent.com/salary?job=software+test+engineer%2Fa>

* + Para ver el estudio completo de mercado por favor ingresar al siguiente documento [Presupuesto.xlsx](https://docs.google.com/spreadsheets/d/1_jW6RFRw9K3twjY3WSCVA7Us3Odqgbmy/edit#gid=1489414248)
    1. **Recursos Computacionales**

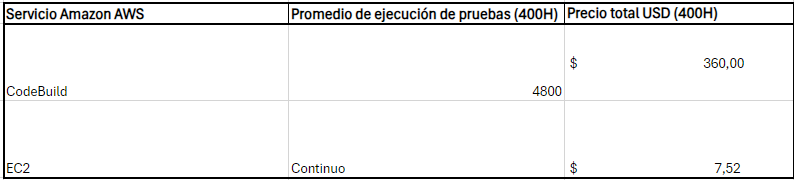
En primer lugar, los ingenieros dispondrán de un equipo asignado a cada uno con las siguientes especificaciones

* Procesador: Intel Core i7 de 8 núcleos a 3.6 GHz
* Memoria RAM: 16 GB DDR4
* Almacenamiento: SSD de 256 GB
* Sistema Operativo: Ubuntu 20.04 LTS

Adicionalmente se utilizará una máquina AWS para tener el despliegue de la aplicación con las siguientes especificaciones

* CPU: 8 Núcleos
* Memoria: 16GiB
* SO: Linux

El servicio cloud para este caso es el EC2 con los siguientes costos:



Se planea un servicio de uso continuo para la ejecución de las pruebas establecidas. Para conocer el detalle del estimado revisar la estimación en AWS [https://calculator.aws/#/estimate?id=a4390a8fa81add624f90e5c45732d25c7fb042f7](https://calculator.aws/%22%20/l%20%22/estimate?id=a4390a8fa81add624f90e5c45732d25c7fb042f7)

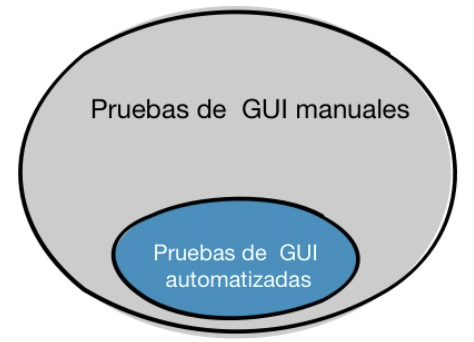
* + 1. **Recursos Económicos para la contratación de servicios/personal:**

**NO APLICA**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Nivel** | **Tipo** | **Técnica** | **Objetivo** |
| Sistema | Funcional y No funcional | Manual exploratorio, pruebas de extremo a extremo (E2E), pruebas de robustez | Encontrar el mayor número de defectos en la configuración general del sitio, asegurando que esté operativa, intuitiva y libre de errores antes del despliegue final. |
| Sistema | Funcional Positiva | Pruebas automatizadas usando herramientas y manuales para asegurar precisión e indenficación | Semana 2: Realizar una auditoría completa de accesibilidad del sitio web utilizando herramientas como WAVE y AXE, y reportar al menos el 80% de los problemas identificados para garantizar que la sección de configuración del sitio sea accesible para usuarios con discapacidades. |
| Sistema | Funcional Postiva | Manual exploratorio | Semana 3: Ejecutar pruebas exhaustivas de usabilidad y funcionalidad en dispositivos móviles utilizando BrowserStack para identificar posibles errores que aseguren que el sitio web sea completamente funcional y fácil de usar en al menos tres modelos diferentes de smartphones y tablets, cubriendo tanto sistemas operativos iOS como Android. |
| Sistema | Funcional positiva | Pruebas automatizadas haciendo uso de herramientas como BrowserStack, CrossBrowserTesting y Selenium | Semana 4: Corregir problemas de compatibilidad en navegadores principales. Es fundamental utilizar herramientas que permitan identificar inconsistencias y errores en la renderización y funcionalidad del sitio web. Durante esta semana, el equipo se enfocará en garantizar que el sitio web ofrezca una experiencia de usuario uniforme y sin contratiempos en los navegadores más utilizados, incluyendo Chrome, Firefox, Safari y Edge. Esto implicará la implementación de un enfoque meticuloso y sistemático que abarque tanto pruebas automatizadas como manuales. |
| Integración | No funcional positiva | Automatización con Resemble.js y BackstopJS | Semana 5: Implementar Visual Regression Testing (VRT) para la versión 5.82.11 de Ghost usando Resemble.js y BackstopJS, y corregir discrepancias visuales para garantizar la integridad visual del sitio tras actualizaciones o cambios. |
| Integración | No funcional positiva | API de automatización: estrés y carga | Semana 6: asegurar que el sitio web pueda soportar un incremento del tráfico del 200% sin fallos significativos. |
| Sistema | No funcional negativa | Pruebas automatizadas y manuales con herramientas y enfoques para evaluación de seguridad (OWASP ZAP (Zed Attack Proxy, OWASP Dependency-Check, Burp Suite) | Semana 7:  Mitigar vulnerabilidades críticas encontradas con OWASP ZAP.  Para realizar una evaluación de seguridad completa utilizando las directrices y herramientas de OWASP, es esencial emplear una combinación de herramientas automatizadas y manuales para identificar y mitigar las principales vulnerabilidades de seguridad. |
| Integración | No funcional positiva | Manual y API de automatización de rendimiento | Semana 8: Garantizar la carga inicial del sitio en menos de 2 segundos para el 90% de los usuarios. |

* 1. **Distribución de Esfuerzo**



La estrategia de distribución de esfuerzo se ve representada como la imagen anterior, se llevan a cabo pruebas exploratorias manuales en nuevas funcionalidades y por las siguientes semanas también se adicionan pruebas manuales como por ejemplo de accesibilidad.

La planeación distribuye los esfuerzos entre pruebas GUI manuales y automatizadas durante ocho semanas: la primera semana se dedica a identificar y resolver defectos en la configuración del sitio a través de pruebas manuales; la segunda se enfoca en una auditoría de accesibilidad utilizando herramientas automatizadas como WAVE y AXE, corrigiendo el 80% de los problemas encontrados; la tercera semana realiza pruebas manuales de usabilidad y funcionalidad en dispositivos móviles con BrowserStack; la cuarta combina pruebas automatizadas y manuales para corregir problemas de compatibilidad en los principales navegadores (Chrome, Firefox, Safari y Edge); la quinta implementa pruebas automatizadas de regresión visual con Resemble.js y BackstopJS para asegurar la integridad visual del sitio; la sexta se centra en pruebas de escalabilidad automatizadas para soportar un incremento del tráfico del 200%; la séptima semana emplea herramientas automatizadas y manuales de OWASP ZAP para mitigar vulnerabilidades de seguridad; y la octava optimiza manualmente la velocidad de carga inicial para que sea inferior a 2 segundos para el 90% de los usuarios.