



USAC
TRICENTENARIA
Universidad de San Carlos de Guatemala
**INGENIERIA
CUNOC**

**HOLA KE HACE
MANUAL TÉCNICO
#3**



ÍNDICE

1. INTRODUCCIÓN	3
2. Aspectos técnicos	3
2.1. Herramientas utilizadas en el desarrollo	3
3. Diagramas	4
3.1. Diagrama entidad relación de base de datos	4
3.2. Diagrama de arquitectura de la aplicación	5
3.3 Diagrama de tablas	6
3.4 Diagrama de paquetes	7
3.5 Diagrama de despliegue	8
3.6 Diagrama de secuencia	8
3.7 Casos de uso	12
4. Historia de revisiones	30
5. Requerimientos mínimos del sistema	30
5.1. Requerimientos web	30
6. Versiones de herramientas utilizadas	30
7. Instalación de php en ubuntu 22.04	30
Instalar PHP	31
Verificar la versión:	31
8. Instalación de mysql en ubuntu 22.04	31
1. Actualizar el Sistema	31
2. Instalar MySQL	31
9. Configurar la Instalación de MySQL	31
4. Iniciar y Habilitar MySQL	31
5. Verificar la Instalación	32
6. Configurar el Acceso para el Usuario Root (Opcional)	32
10. Probar la Conexión con la Contraseña	32

1. INTRODUCCIÓN

El presente documento pretende mostrar de forma técnica los recursos e información relevante para el aplicativo

2. Aspectos técnicos

2.1. Herramientas utilizadas en el desarrollo

Tecnología	Descripción
Visual Studio Code	Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOS y Web. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.
GitHub	GitHub es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. GitHub aloja tu repositorio de código y te brinda herramientas muy útiles para el trabajo en equipo
HTML	HTML (HyperText Markup Language) es el lenguaje estándar utilizado para crear y estructurar contenido en la web. Es un lenguaje de marcado que utiliza etiquetas para definir los diferentes elementos de una página web, como texto, imágenes, enlaces, formularios, y otros componentes.
CSS	CSS (Cascading Style Sheets) es un lenguaje utilizado para describir la presentación visual de un documento escrito en HTML o XML. Mientras que HTML estructura el contenido de una página web, CSS controla su apariencia, permitiendo aplicar estilos a elementos como colores, fuentes, márgenes, alineaciones y otros aspectos visuales.
Mysql	MySQL es un sistema de gestión de bases de datos relacional (RDBMS) y es un software o servicio utilizado para crear y administrar bases de datos basadas en un modelo relacional.
MySQL Workbench	MySQL Workbench es una herramienta visual que proporciona una interfaz gráfica para trabajar con bases de datos MySQL. Es desarrollada por Oracle y está diseñada para facilitar la administración y diseño de bases de datos, así como la ejecución de consultas SQL.
Php	PHP (Hypertext Preprocessor) es un lenguaje de programación del lado del servidor diseñado para el desarrollo web. Es ampliamente utilizado para crear aplicaciones web dinámicas e interactivas. PHP se ejecuta en el servidor web y genera contenido que se envía al navegador del usuario, como HTML, imágenes y otros tipos de datos.
JS	"JS" se refiere a JavaScript, un lenguaje de programación utilizado en el desarrollo web para añadir interactividad a las páginas web. Con JavaScript, puedes manejar eventos en el lado del cliente, manipular el DOM (Document Object Model) para actualizar dinámicamente el contenido de la página, realizar validaciones en formularios, interactuar con APIs, entre otras cosas.
Bootstrap	Es un framework de desarrollo web de código abierto basado en HTML, CSS y JavaScript, diseñado para facilitar la creación de sitios web y aplicaciones web responsivas.

3.1. Diagrama entidad relación de base de datos

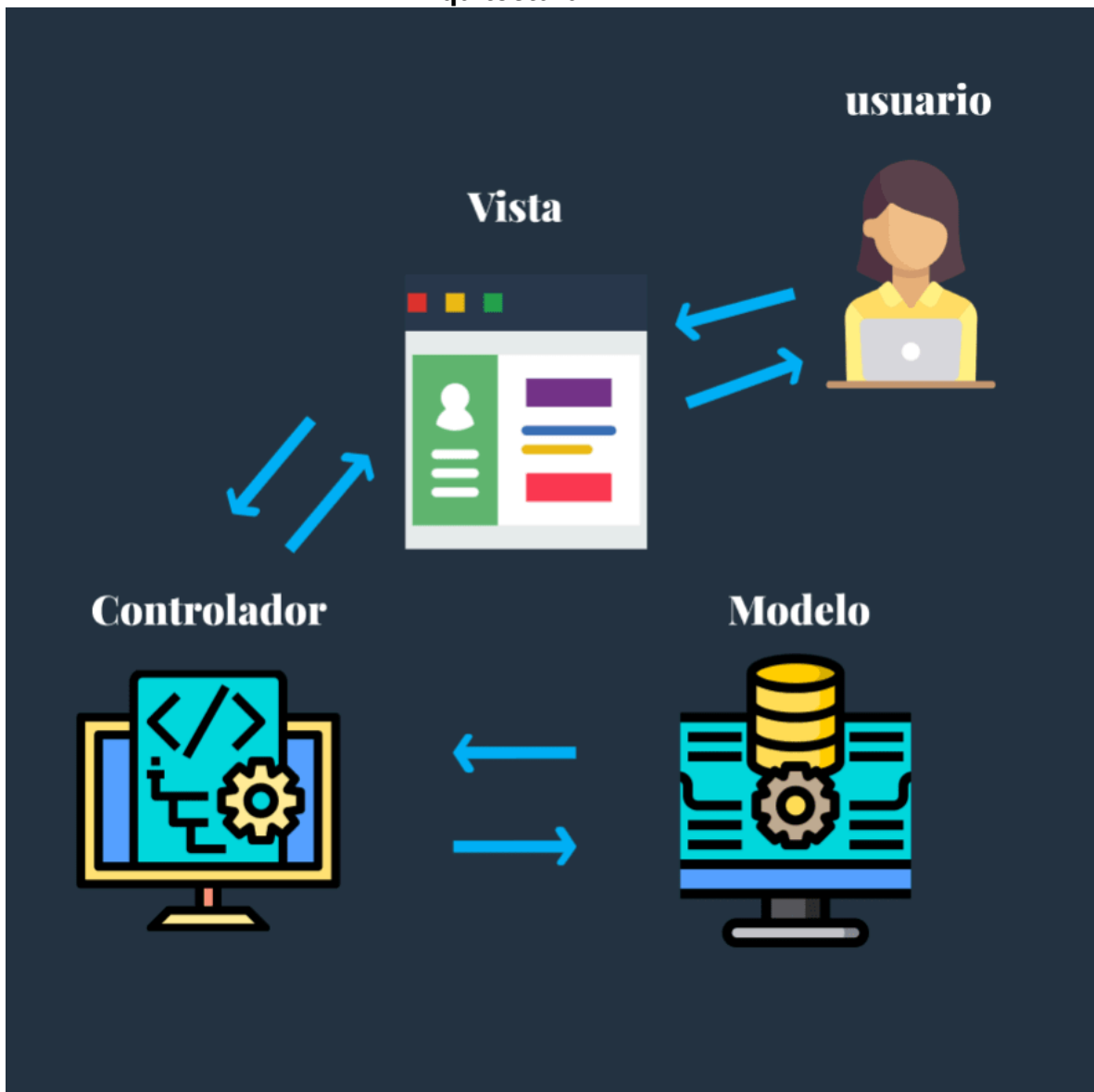
El Diagrama **Entidad-Relación (DER)** del sistema "Hola Qué Hace" está diseñado para registrar, gestionar y mantener la información clave relacionada con las publicaciones o eventos, los usuarios, y sus roles. Este diseño cubre todas las operaciones necesarias como guardar, buscar, modificar o eliminar datos mientras se alinea con las reglas de negocio definidas.



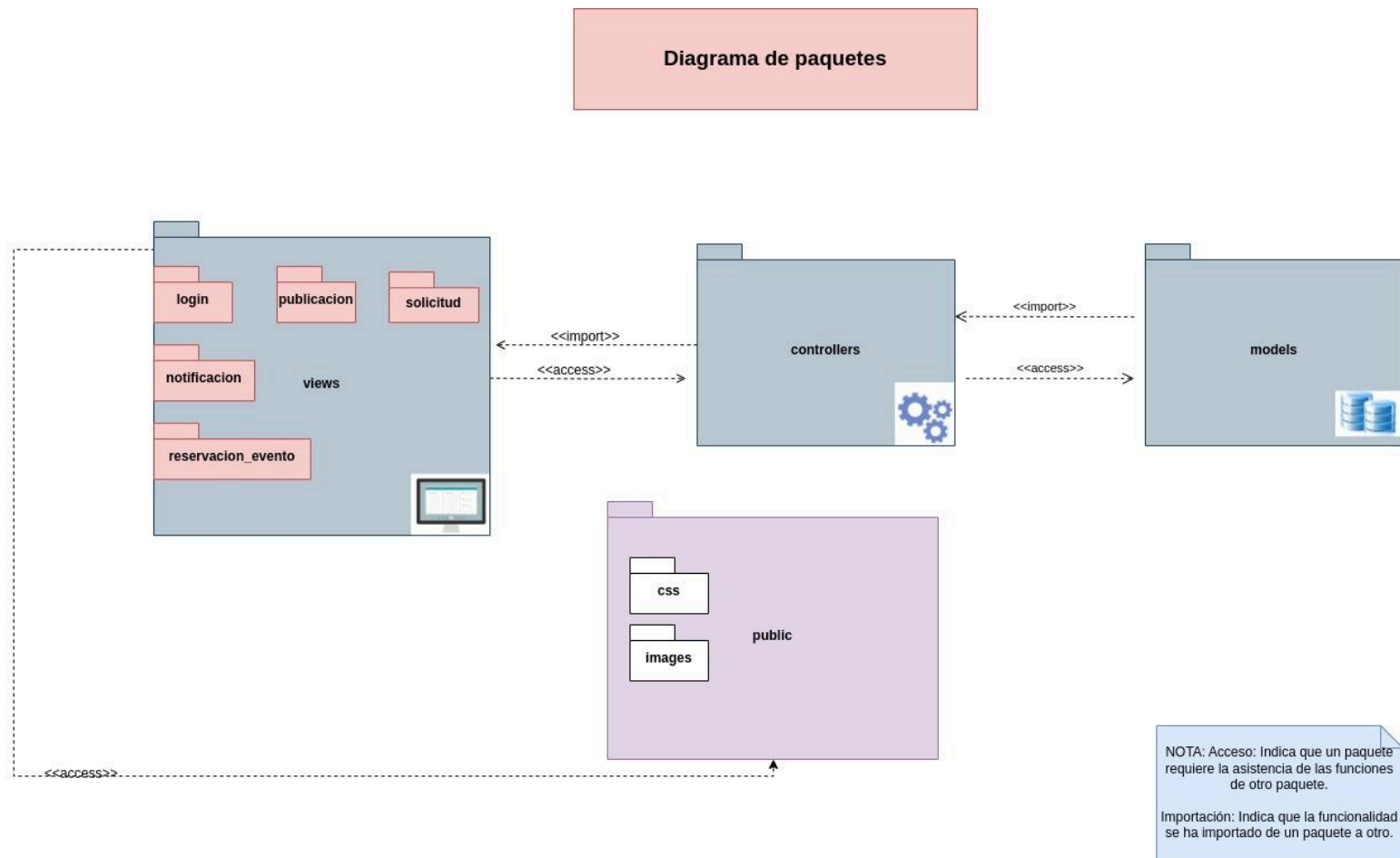
3.2. Diagrama de arquitectura de la aplicación

Diagrama que permite proporcionar una idea del diseño y sus componentes de la estructura del aplicativo.

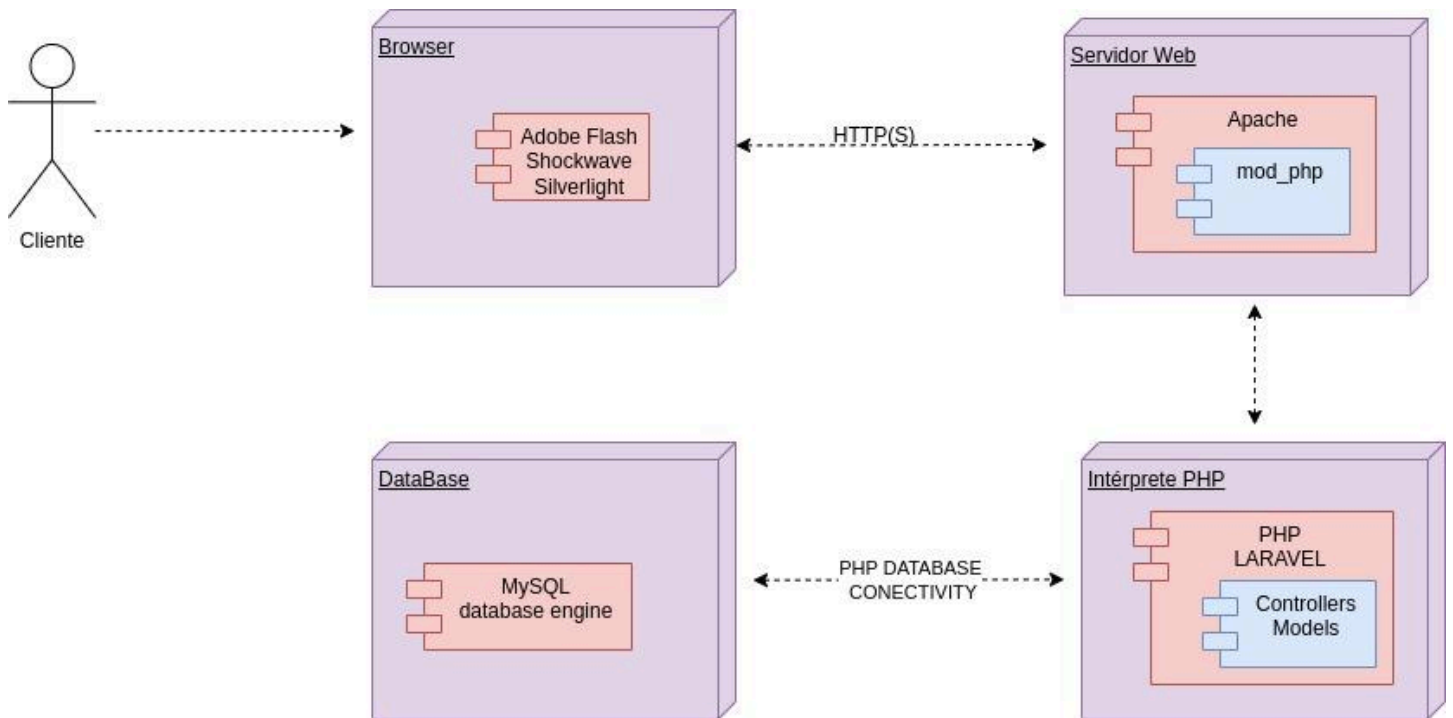
Arquitectura MVC



3.4 Diagrama de paquetes



3.5 Diagrama de despliegue



3.6 Diagrama de secuencia

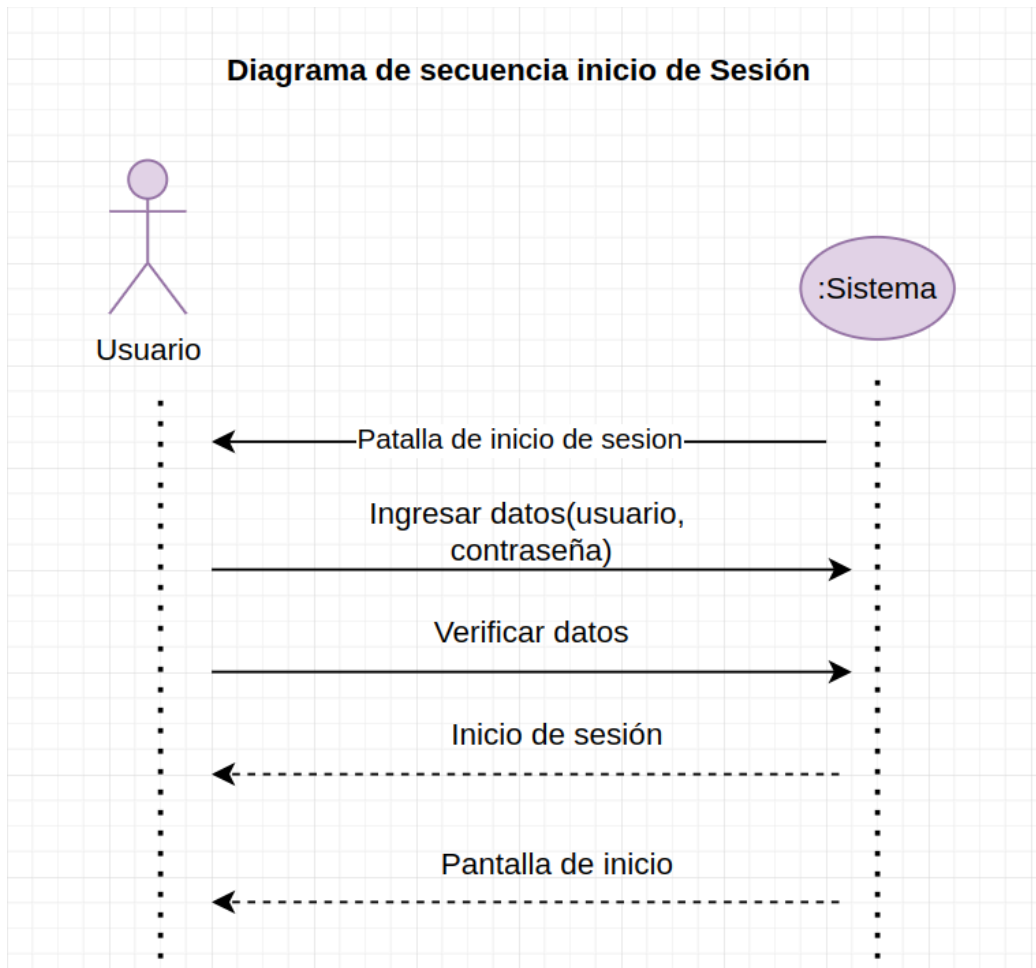


Diagrama de secuencia creación usuario

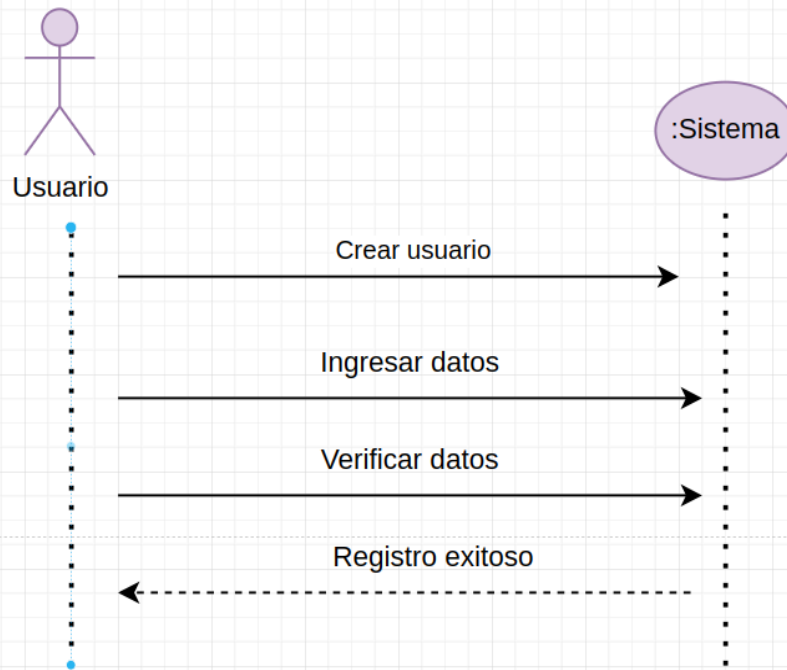


Diagrama de secuencia crear publicación

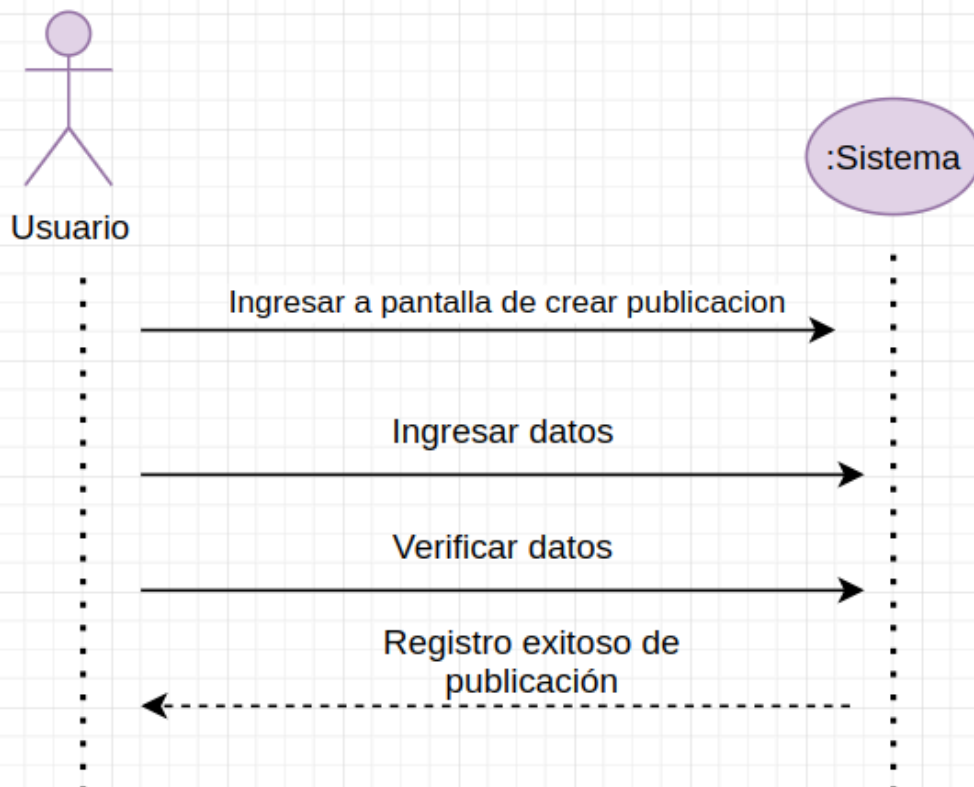


Diagrama de secuencia de asistir a un evento

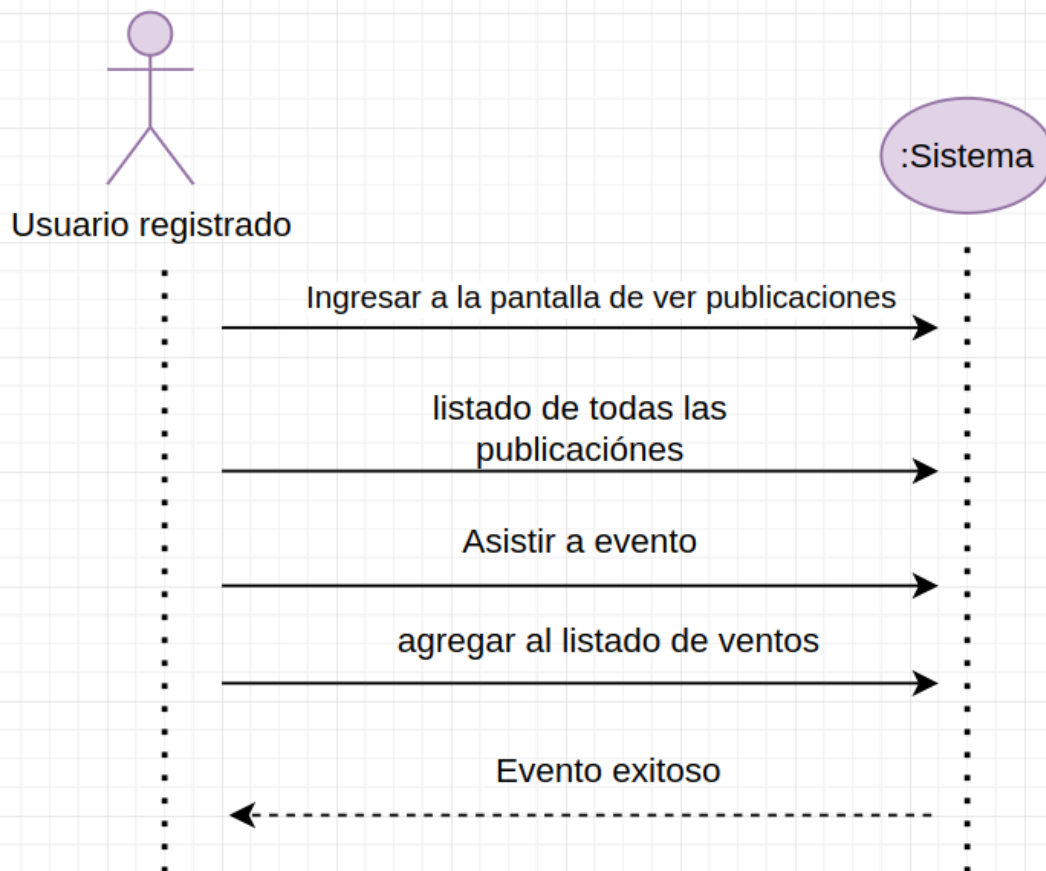
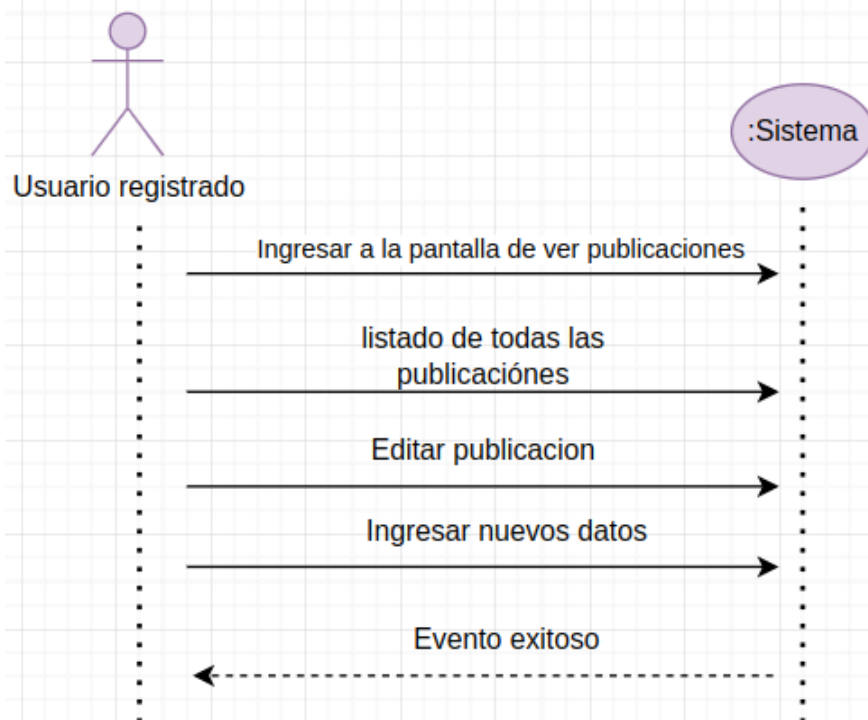
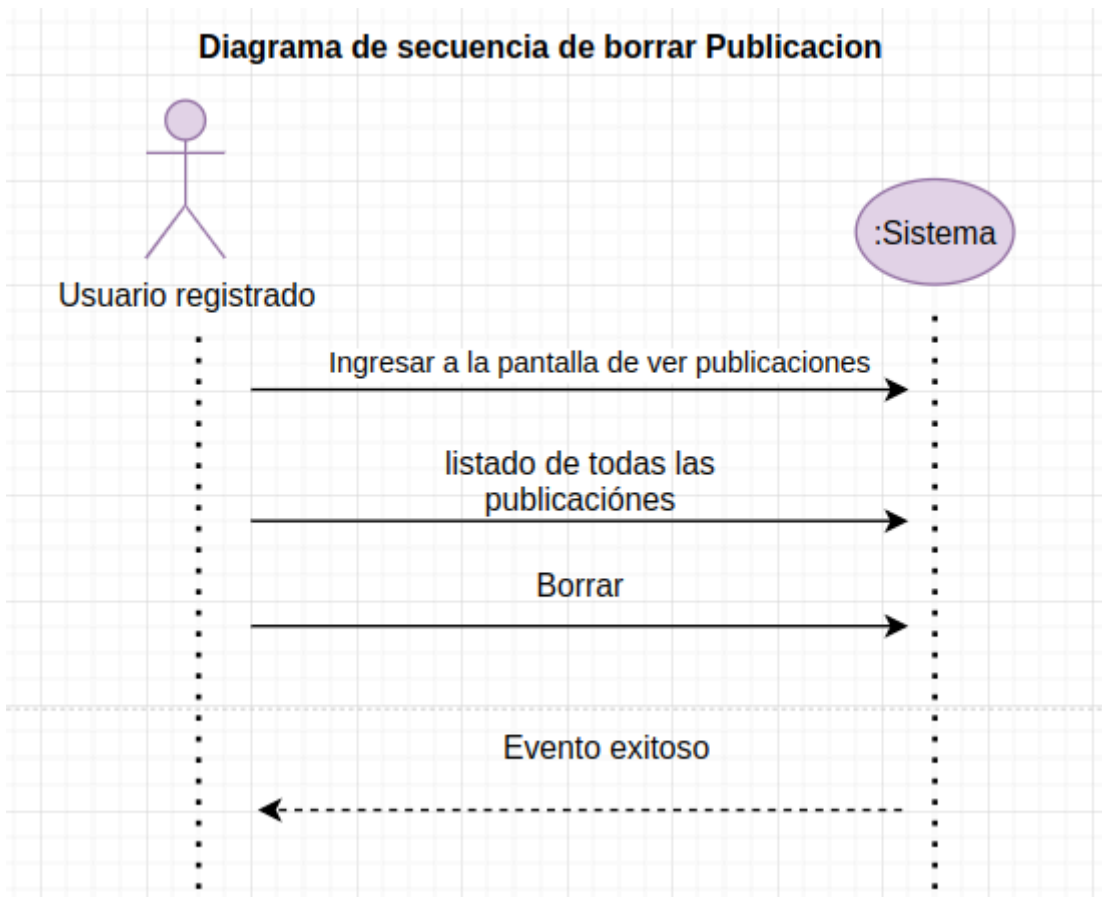


Diagrama de secuencia de editar Publicacion





4. Requerimientos mínimos del sistema

4.1. Requerimientos web

- Sistema Operativo: Linux Ubuntu 22.04.4 LTS
- Procesador: Intel Core Celeron
- Memoria RAM: 8GB
- Disco Duro: 256GB

3.7 Casos de uso

CASO DE USO	
Número	CU001
Caso de Uso:	Crear publicación o evento
Actores:	Principal: Publicador de Anuncios Secundario: Administrador
Personal involucrado e intereses	Personal Involucrado: Publicar información de eventos. -Administrador: Validar publicaciones. Intereses: Asegurar que las publicaciones sean relevantes y cumplan con las reglas del sistema.
Precondiciones	<ul style="list-style-type: none"> • Debe existir al menos una publicación pendiente de validación.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • La publicación es visible si es aprobada. • Se notifica al publicador.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. Acceso al sistema. 2. Crear Publicación. 4. Validación del sistema. 5. Publicación automática o revisión por el administrador
Extensiones (flujos alternos)	Escenario 1: Si la publicación tiene reportes, el administrador revisa los reportes antes de tomar una decisión.
Requisitos especiales	El sistema debe cumplir con las normativas de protección de datos personales. Interfaz de usuario fácil de usar y eficiente.
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript. Variaciones de Datos: Formatos de fecha, hora, categorías
Frecuencia	Este caso de uso puede ocurrir varias veces al día, dependiendo de la demanda de publicaciones.

Temas abiertos	Automatización de moderación.
Descripción:	Este caso describe cómo un publicador crea una publicación, que es revisada o publicada automáticamente cuando el administrador lo acepta.
Tipo:	<p>Operativo: Publicación o evento es una operación rutinaria realizada por los usuarios de tipo publicador.</p> <p>Funcional: describe una funcionalidad clave que el sistema debe implementar para permitir que los usuarios hagan puedan asistir a los eventos.</p>

CASO DE USO	
Número	CU002
Caso de Uso:	Validar Publicación
Actores:	<p>Principal: Administrador</p> <p>Secundario: Publicador de Anuncios</p>
Personal involucrado e intereses	<p>Personal Involucrado</p> <p>Administrador: Asegurarse de que los eventos publicados cumplan con las normas y sean relevantes.</p> <p>Publicador de Anuncios: Desea que sus eventos sean validados y publicados eficientemente</p> <p>Intereses: Asegurar que las publicaciones sean relevantes y cumplan con las reglas del sistema.</p>
Precondiciones	<ul style="list-style-type: none"> • El administrador debe estar registrado en el sistema. • Deben existir publicaciones pendientes de validación.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • La publicación es aprobada y visible en el panel principal. • Si es rechazada, el sistema notifica

	al publicador con el motivo.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El administrador inicia sesión en el sistema. 2. Selecciona la opción "Publicaciones Pendientes". 3. El sistema muestra la lista de publicaciones enviadas por los publicadores. 4. El administrador revisa la información de cada publicación. 5. El administrador decide: Aprobar: El sistema publica la publicación en el panel principal. Notifica al publicador la aprobación. Rechazar: El sistema elimina la publicación de la cola pendiente. Notifica al publicador indicando el motivo del rechazo.
Extensiones (flujos alternos)	<p>Escenario 1: No hay publicaciones pendientes</p> <ul style="list-style-type: none"> • El sistema muestra un mensaje al administrador indicando que no hay publicaciones pendientes.
Requisitos especiales	El sistema debe cumplir con las normativas de protección de datos personales. Interfaz de usuario fácil de usar y eficiente.
Lista de tecnología y variaciones de datos	<p>Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.</p> <p>Variaciones de Datos: Formatos de fecha, hora, categorías</p>
Frecuencia	Este caso de uso puede ocurrir varias veces al día, dependiendo de la demanda de publicaciones.
Temas abiertos	Automatización de moderación.
Descripción:	Este caso describe cómo el administrador acepta o rechaza publicaciones.

Tipo:	<p>Operativo: Validar publicación es una operación rutinaria realizada por el administrador para aceptar o rechazar publicaciones.</p> <p>Funcional: describe una funcionalidad clave que el sistema debe implementar para permitir que los usuarios puedan asistir a los eventos.</p>
--------------	--

CASO DE USO	
Número	CU003
Caso de Uso:	Reportar Publicación
Actores:	<p>Principal: Usuario Registrado</p> <p>Secundario: Administrador</p>
Personal involucrado e intereses	<p>Personal Involucrado</p> <p>Usuario Registrado: Reportar publicaciones inadecuadas o irrelevantes.</p> <p>Administrador: Validar los reportes y tomar decisiones basadas en ellos.</p> <p>Intereses: Asegurar que las publicaciones sean relevantes y cumplan con las reglas del sistema.</p>
Precondiciones	<ul style="list-style-type: none"> • El usuario debe estar registrado en el sistema. • Debe existir una publicación visible en el panel principal.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • La publicación reportada queda marcada como "reportada". • El sistema notifica al administrador para revisión.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El usuario accede al sistema. 2. Navega por las publicaciones en el panel principal. 3. Selecciona la publicación que desea reportar. 4. El sistema presenta un formulario para indicar el motivo del reporte. 5. El usuario completa y envía el

	<p>formulario.</p> <p>6.El sistema marca la publicación como "reportada" y notifica al administrador.</p>
Extensiones (flujos alternos)	<p>Escenario 1: Publicación ya reportada</p> <ul style="list-style-type: none"> • El sistema informa al usuario que la publicación ya ha sido reportada y está bajo revisión.
Requisitos especiales	<p>El sistema debe cumplir con las normativas de protección de datos personales.</p> <p>Interfaz de usuario fácil de usar y eficiente.</p>
Lista de tecnología y variaciones de datos	<p>Tecnología: Sistema web de publicaciones, base de datos MySQL,PHP, HTML, CSS, JavaScript.</p> <p>Variaciones de Datos: validación de publicación y reporte realizado</p>
Frecuencia	<p>Este caso de uso puede ocurrir varias veces al día, dependiendo de la demanda de publicaciones rechazadas.</p>
Temas abiertos	<ul style="list-style-type: none"> • Automatización del proceso de moderación para publicaciones repetitivas o que cumplan criterios preestablecidos. • Integración de inteligencia artificial para identificar publicaciones inadecuadas.
Descripción:	<p>Este caso describe cómo los usuarios pueden reportar publicaciones que consideren inapropiadas o irrelevantes y cómo el administrador gestiona los reportes.</p>
Tipo:	<p>Operativo:Permitir que los usuarios reporten publicaciones con problemas de manera eficiente.</p> <p>Funcional:Describir cómo el sistema debe gestionar y registrar los reportes enviados por los usuarios.</p>

CASO DE USO	
Número	CU004
Caso de Uso:	Deseo Asistir
Actores:	Principal: Usuario Registrado
Personal involucrado e intereses	<p>Usuario Registrado: Guardar eventos a los que planea asistir y recibir notificaciones de ellos.</p> <p>Intereses: Asistir a un evento</p>
Precondiciones	<ul style="list-style-type: none"> • El usuario debe estar registrado en el sistema. • El sistema activa un conteo regresivo basado en la fecha del evento.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • El evento se guarda en la lista de eventos del usuario. • El sistema activa un conteo regresivo basado en la fecha del evento.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El usuario accede al sistema.. 2.Explora las publicaciones en el panel principal. 3.Selecciona un evento y hace clic en "Asistir". 4. El sistema guarda el evento en la lista personal del usuario. 5.Activa el conteo regresivo para el evento. 6.El sistema notifica al usuario que el evento ha sido agregado exitosamente.
Extensiones (flujos alternos)	<p>Escenario 1: El evento ya está en la lista del usuario</p> <ul style="list-style-type: none"> • El sistema muestra un mensaje indicando que el evento ya fue

	agregado previamente.
Requisitos especiales	El sistema debe tener un contador preciso de tiempo para los eventos. La interfaz debe mostrar una lista clara y accesible de eventos seleccionados.
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.
Frecuencia	Este caso puede ocurrir varias veces al día según la interacción de los usuarios con los eventos.
Temas abiertos	Notificaciones automatizadas: Configuración para notificar al usuario antes del evento. Acceso rápido: Mejorar la interfaz para que los usuarios accedan rápidamente a su lista de eventos.
Descripción:	Este caso describe cómo los usuarios registrados seleccionan eventos de interés, los guardan en su lista y reciben un conteo regresivo hacia el inicio del evento.
Tipo:	Operativo: Permitir que los usuarios guarden eventos y reciban notificaciones fácilmente. Funcional: Describir cómo el sistema muestra los eventos seleccionados y gestiona el conteo regresivo.

CASO DE USO	
Número	CU005
Caso de Uso:	Crear Usuarios Administradores
Actores:	Principal: Administrador

	Secundario: Sistema
Personal involucrado e intereses	<p>Administrador: Desea agregar nuevos administradores al sistema para distribuir la carga de trabajo.</p> <p>Sistema: Garantiza que sólo un administrador tenga permisos para crear nuevos administradores.</p>
Precondiciones	<ul style="list-style-type: none"> • El actor debe tener el rol de Administrador. • Debe estar autenticado en el sistema.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • El nuevo administrador se registra correctamente y puede acceder al sistema con sus credenciales.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El administrador inicia sesión en el sistema. 2. Selecciona la opción "Crear Usuario Administrador". 3. Llena el formulario con los datos requeridos: nombre, correo electrónico y contraseña. 4. Envía el formulario. 5. El sistema valida los datos ingresados: <ul style="list-style-type: none"> • Si son válidos, registra al nuevo administrador. • Si hay errores, solicita corregirlos. 6. El sistema notifica al administrador que el registro fue exitoso.
Extensiones (flujos alternos)	<p>Escenario 1: Acceso denegado:</p> <ul style="list-style-type: none"> • Si un usuario no administrador intenta acceder a esta funcionalidad, el sistema muestra un mensaje de error.

Requisitos especiales	El sistema debe cifrar las contraseñas antes de almacenarlas.
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript. Variaciones de Datos: Correo electrónico único, contraseña segura.
Frecuencia	Ocurre cuando un administrador desea agregar nuevos administradores.
Temas abiertos	Implementar niveles adicionales de validación para asegurar que solo administradores autorizados puedan crear nuevos administradores.
Descripción:	Este caso describe cómo los administradores crean nuevas cuentas para otros administradores.
Tipo:	Operativo: Permite gestionar usuarios con permisos elevados. Funcional: Garantiza la exclusividad de la funcionalidad para administradores.

CASO DE USO	
Número	CU006
Caso de Uso:	Login
Actores:	Principal: Usuario (Cualquier Rol: Administrador, Publicador, Usuario Registrado)
Personal involucrado e intereses	Usuario Registrado: Desea acceder a sus funcionalidades específicas según su rol. Intereses: Asistir a un evento, administrar, publicar
Precondiciones	<ul style="list-style-type: none"> • El usuario debe estar registrado en el sistema. • .Debe recordar sus credenciales.

Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> El usuario accede al sistema con los permisos correspondientes a su rol.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> El usuario accede a la página de login. Introduce su nombre de usuario y contraseña. Hace clic en "Iniciar Sesión". El sistema valida las credenciales: <ul style="list-style-type: none"> Si son correctas, permite el acceso al sistema según el rol del usuario. Si son incorrectas, muestra un mensaje de error.
Extensiones (flujos alternos)	<p>Escenario 1: Credenciales incorrectas:</p> <ul style="list-style-type: none"> Si el usuario ingresa un nombre de usuario o contraseña incorrecta, el sistema muestra un mensaje indicando el error y permite volver a intentarlo.
Requisitos especiales	Debe mostrar la interfaz con las acciones que puede hacer cada rol
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.
Frecuencia	Este caso puede ocurrir varias veces al día según la interacción de los usuarios con los eventos.
Temas abiertos	<p>Recuperación de acceso: Implementación de un sistema seguro para restablecer contraseñas.</p> <p>Autenticación avanzada: Opciones para autenticación de dos factores (2FA).</p>
Descripción:	Este caso describe cómo los usuarios

	acceden al sistema introduciendo sus credenciales. En caso de error, el sistema muestra un mensaje de error.
Tipo:	Operativo: Permitir el acceso de usuarios según su rol de manera segura. Funcional: Describir cómo se valida el acceso de los usuarios y cómo se gestionan los errores de autenticación.

CASO DE USO	
Número	CU007
Caso de Uso:	Login
Actores:	Principal: Usuario (Cualquier Rol: Administrador, Publicador, Usuario Registrado)
Personal involucrado e intereses	Usuario Registrado: Desea acceder a sus funcionalidades específicas según su rol. Intereses: Asistir a un evento, administrar, publicar
Precondiciones	<ul style="list-style-type: none"> • El usuario debe estar registrado en el sistema. • .Debe recordar sus credenciales.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • El usuario accede al sistema con los permisos correspondientes a su rol.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El usuario accede a la página de login. 2. Introduce su nombre de usuario y contraseña. 3. Hace clic en "Iniciar Sesión". 4. El sistema valida las credenciales: <ul style="list-style-type: none"> ○ Si son correctas, permite el acceso al sistema según el rol del usuario. ○ Si son incorrectas, muestra

	un mensaje de error.
Extensiones (flujos alternos)	Escenario 1: Credenciales incorrectas: <ul style="list-style-type: none"> Si el usuario ingresa un nombre de usuario o contraseña incorrecta, el sistema muestra un mensaje indicando el error y permite volver a intentarlo.
Requisitos especiales	Debe mostrar la interfaz con las acciones que puede hacer cada rol
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.
Frecuencia	Este caso puede ocurrir varias veces al día según la interacción de los usuarios con los eventos.
Temas abiertos	Recuperación de acceso: Implementación de un sistema seguro para restablecer contraseñas. Autenticación avanzada: Opciones para autenticación de dos factores (2FA).
Descripción:	Este caso describe cómo los usuarios acceden al sistema introduciendo sus credenciales. En caso de error, el sistema muestra un mensaje de error.
Tipo:	Operativo: Permitir el acceso de usuarios según su rol de manera segura. Funcional: Describir cómo se valida el acceso de los usuarios y cómo se gestionan los errores de autenticación.

CASO DE USO	
Número	CU008

Caso de Uso:	Login
Actores:	Principal: Usuario (Cualquier Rol: Administrador, Publicador, Usuario Registrado)
Personal involucrado e intereses	<p>Usuario Registrado: Desea acceder a sus funcionalidades específicas según su rol.</p> <p>Intereses: Asistir a un evento, administrar, publicar</p>
Precondiciones	<ul style="list-style-type: none"> • El usuario debe estar registrado en el sistema. • .Debe recordar sus credenciales.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • El usuario accede al sistema con los permisos correspondientes a su rol.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El usuario accede a la página de login. 2. Introduce su nombre de usuario y contraseña. 3. Hace clic en "Iniciar Sesión". 4. El sistema valida las credenciales: <ul style="list-style-type: none"> ○ Si son correctas, permite el acceso al sistema según el rol del usuario. ○ Si son incorrectas, muestra un mensaje de error.
Extensiones (flujos alternos)	<p>Escenario 1: Credenciales incorrectas:</p> <ul style="list-style-type: none"> • Si el usuario ingresa un nombre de usuario o contraseña incorrecta, el sistema muestra un mensaje indicando el error y permite volver a intentarlo.
Requisitos especiales	Debe mostrar la interfaz con las acciones que puede hacer cada rol

Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.
Frecuencia	Este caso puede ocurrir varias veces al día según la interacción de los usuarios con los eventos.
Temas abiertos	<p>Recuperación de acceso: Implementación de un sistema seguro para restablecer contraseñas.</p> <p>Autenticación avanzada: Opciones para autenticación de dos factores (2FA).</p>
Descripción:	Este caso describe cómo los usuarios acceden al sistema introduciendo sus credenciales. En caso de error, el sistema muestra un mensaje de error.
Tipo:	<p>Operativo: Permitir el acceso de usuarios según su rol de manera segura.</p> <p>Funcional: Describir cómo se valida el acceso de los usuarios y cómo se gestionan los errores de autenticación.</p>

CASO DE USO	
Número	CU008
Caso de Uso:	Borrar Publicación (Lógico)
Actores:	<p>Principal: Administrador</p> <p>Secundario: Publicador de Anuncios</p>
Personal involucrado e intereses	<p>Administrador: Desea eliminar publicaciones que no cumplan con las políticas del sistema sin afectar la integridad de los datos.</p> <p>Publicador de Anuncios: Quiere evitar que sus publicaciones sean eliminadas permanentemente, permitiendo su</p>

	recuperación tras una revisión.
Precondiciones	<ul style="list-style-type: none"> • El administrador debe estar autenticado en el sistema • La publicación debe existir y estar visible en el panel principal.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • La publicación se marca como "inactiva" en el sistema. • La publicación deja de ser visible en el panel principal. • Los datos de la publicación permanecen en la base de datos para futuras referencias o auditorías.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1.El administrador inicia sesión en el sistema. 2.Accede al panel de administración de publicaciones. 3. Selecciona la publicación que desea eliminar. 4. Hace clic en la opción "Eliminar". 5.El sistema marca la publicación como "inactiva" (estado = 0) en la base de datos. 6.La publicación ya no es visible para los usuarios en el panel principal.
Extensiones (flujos alternos)	<p>Escenario 1:</p> <p>Publicación inexistente:</p> <ul style="list-style-type: none"> • Si la publicación seleccionada no existe, el sistema muestra un mensaje de error indicando que no se puede eliminar.
Requisitos especiales	El sistema debe garantizar que las publicaciones eliminadas lógicamente no sean accesibles ni visibles para los

	usuarios normales.
Lista de tecnología y variaciones de datos	Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript. Variaciones de Datos: Campo "estado" de la publicación (activo = 1, inactivo = 0).
Frecuencia	Este caso puede ocurrir ocasionalmente, según las necesidades de moderación del administrador.
Temas abiertos	Recuperación de publicaciones: Implementar una funcionalidad de restauración para publicaciones eliminadas lógicamente.
Descripción:	Este caso describe cómo un administrador puede eliminar una publicación del panel principal de forma lógica, manteniendo los datos en el sistema para futuras auditorías o revisiones.
Tipo:	Operativo: Garantiza la moderación de publicaciones sin eliminarlas Funcional: Describe el proceso de marcación como "inactiva" para mantener la integridad de los datos.

CASO DE USO	
Número	CU009
Caso de Uso:	Editar Publicación
Actores:	Principal: Administrador Secundario: Publicador de Anuncios
Personal involucrado e intereses	Publicador de Anuncios: Desea modificar una publicación previamente creada para corregir errores o actualizar información. Administrador: Quiere asegurarse de que

	las publicaciones editadas cumplan con las políticas del sistema, si es necesario.
Precondiciones	<ul style="list-style-type: none"> • El publicador debe estar autenticado en el sistema. • La publicación debe estar activa en el sistema. • El publicador debe ser el dueño de la publicación.
Garantías de éxito (Postcondiciones)	<ul style="list-style-type: none"> • La publicación es actualizada correctamente con los nuevos datos.
Escenario principal de éxito (flujo básico)	<ol style="list-style-type: none"> 1. El publicador inicia sesión en el sistema. 2. Accede al panel de publicaciones y selecciona la publicación que desea editar. 3. Hace clic en la opción "Editar". 4. El sistema muestra un formulario con los datos actuales de la publicación. 5. El publicador actualiza los campos necesarios (e.g., nombre, descripción). Confirma los cambios. 6. El sistema valida la nueva información ingresada. El sistema actualiza la base de datos con los nuevos datos:
Extensiones (flujos alternos)	<ol style="list-style-type: none"> 1. Datos inválidos: Si los nuevos datos ingresados no cumplen con las validaciones (e.g., fecha en formato incorrecto), el sistema muestra un mensaje indicando el error y solicita corregirlo. 2. Publicación inexistente: Si la publicación no existe o no pertenece al publicador, el sistema muestra un mensaje de error.

	<p>3. Error en el servidor:</p> <p>Si ocurre un problema técnico al guardar los cambios, el sistema muestra un mensaje de error temporal y sugiere intentar más tarde.</p>
Requisitos especiales	El sistema debe registrar un historial de cambios realizados para auditorías.
Lista de tecnología y variaciones de datos	<p>Tecnología: Sistema web de publicaciones, base de datos MySQL, PHP, HTML, CSS, JavaScript.</p> <p>Variaciones de Datos: Campo "estado" de la publicación (activo = 1, inactivo = 0).</p>
Frecuencia	Este caso puede ocurrir ocasionalmente, dependiendo de la necesidad de actualizar publicaciones.
Temas abiertos	<p>Recuperación de publicaciones: Implementar una funcionalidad de restauración para publicaciones eliminadas lógicamente.</p>
Descripción:	Este caso describe cómo un administrador puede eliminar una publicación del panel principal de forma lógica, manteniendo los datos en el sistema para futuras auditorías o revisiones.
Tipo:	<p>Operativo: Garantiza la moderación de publicaciones sin eliminarlas</p> <p>Funcional: Describe el proceso de marcación como "inactiva" para mantener la integridad de los datos.</p>

4. Historia de revisiones

Versión	Fecha	Descripción	Autor
Borrador inicio	18 de octubre de 2024	Implementación de publicaciones y demás requerimientos	Melanni Tzul
Borrador Final	19 5 Noviembre de 2024	Implementación de roles, de publicaciones, asistir a un evento y reportar eventos etc...	Melanni Tzul

5. Requerimientos mínimos del sistema

5.1. Requerimientos web

- Sistema Operativo: Linux, Windows y macOS
- Procesador: Intel Core Celeron
- Memoria RAM: 8GB
- Disco Duro: 250GB

6. Versiones de herramientas utilizadas



```
melanni@MelanniTzul: ~  
melanni@MelanniTzul:~$ php -v  
PHP Warning:  Module "mbstring" is already loaded in Unknown on line 0  
PHP 8.3.11 (cli) (built: Aug 30 2024 09:27:49) (NTS)  
Copyright (c) The PHP Group  
Zend Engine v4.3.11, Copyright (c) Zend Technologies  
with Zend OPcache v8.3.11, Copyright (c), by Zend Technologies  
melanni@MelanniTzul:~$ mysql --version  
mysql Ver 8.0.39-0ubuntu0.22.04.1 for Linux on x86_64 ((Ubuntu))  
melanni@MelanniTzul:~$
```

7. Instalación de php en ubuntu 22.04

Abrir terminal: Primero, actualiza los paquetes de tu sistema para asegurarte de que estás instalando la versión más reciente disponible:

Comandos:

```
sudo apt update
```

`sudo apt upgrade`

Instalar PHP

En Ubuntu 22.04 ya se incluye una versión de PHP en sus repositorios predeterminados.

comando: `sudo apt install php`

Verificar la versión:

`php -v`

8. Instalación de mysql en ubuntu 22.04

1. Actualizar el Sistema

Primero, asegúrate de que tu sistema esté actualizado:

`sudo apt update`

`sudo apt upgrade`

2. Instalar MySQL

Para instalar MySQL, ejecuta el siguiente comando:

`sudo apt install mysql-server`

9. Configurar la Instalación de MySQL

Después de instalar MySQL, puedes ejecutar el script de seguridad para realizar algunas configuraciones básicas, como establecer una contraseña para el usuario root, eliminar usuarios anónimos y deshabilitar el acceso remoto para el usuario root. Ejecuta el siguiente comando para iniciar el script de configuración:

`sudo mysql_secure_installation`

4. Iniciar y Habilitar MySQL

Asegúrate de que el servicio de MySQL esté en funcionamiento:

`sudo systemctl start mysql`

También puedes habilitar MySQL para que se inicie automáticamente al arrancar el sistema:

```
sudo systemctl enable mysql
```

5. Verificar la Instalación

Para verificar que MySQL está funcionando, puedes iniciar sesión en el servidor MySQL:

```
sudo mysql
```

Esto te llevará a la consola de MySQL. Para salir de la consola de MySQL, escribe:

```
exit
```

6. Configurar el Acceso para el Usuario Root (Opcional)

Si deseas configurar la autenticación basada en contraseña en lugar de **auth_socket**, sigue estos pasos:

1. Accede a MySQL como root:

```
bash
```

```
Copiar código
```

```
sudo mysql
```

Cambia el método de autenticación del usuario root a **mysql_native_password** y establece una contraseña:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password  
BY 'your_password';
```

```
FLUSH PRIVILEGES;
```

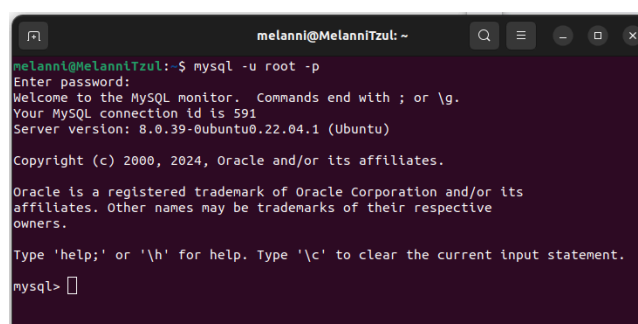
Sal de MySQL:

```
exit
```

10. Probar la Conexión con la Contraseña

Intenta conectarte a MySQL utilizando la contraseña que acabas de configurar:

```
mysql -u root -p
```

A screenshot of a terminal window with a dark background. The prompt is 'melanni@MelanniTzul: ~'. The user has entered 'mysql -u root -p'. The terminal shows the MySQL login process: 'Enter password:', 'Welcome to the MySQL monitor. Commands end with ; or \g.', 'Your MySQL connection id is 591', 'Server version: 8.0.39-0ubuntu0.22.04.1 (Ubuntu)', 'Copyright (c) 2000, 2024, Oracle and/or its affiliates.', 'Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.', and 'Type \'help;\' or \'h\' for help. Type \'c\' to clear the current input statement.' The prompt 'mysql>' is visible at the bottom.

Manual Técnico

©Copyright CUNOC 2024 #3