**BLOAGG**

**Plataforma Web con API REST y FRONTEND**

Autores:

MARIO EMMANUEL RIOS BENITES I202200351

Cibertec - Escuela de Computación e Informática

2025

INDICE

[1. RESUMEN 3](#_Toc207504746)

[2. INTRODUCCIÓN 3](#_Toc207504747)

[3. DIAGNÓSTICO 3](#_Toc207504748)

[4. OBJETIVOS 4](#_Toc207504749)

[5. JUSTIFICACIÓN 4](#_Toc207504750)

[6. DEFINICIÓN Y ALCANCE 4](#_Toc207504751)

[7. PRODUCTOS Y ENTREGABLES 5](#_Toc207504752)

[8. CONCLUSIONES 5](#_Toc207504753)

[9. RECOMENDACIONES 5](#_Toc207504754)

[10. GLOSARIO 5](#_Toc207504755)

[11. BIBLIOGRAFÍA 6](#_Toc207504756)

# RESUMEN

El proyecto BlogApp es una API REST desarrollada con Spring Boot que permite la gestión de usuarios, publicaciones y comentarios. Su funcionalidad incluye la creación, consulta, listado y eliminación de estos elementos, proporcionando endpoints estructurados y documentados con Swagger.

La API utiliza Spring Boot, MySQL y Hibernate (JPA) para garantizar una arquitectura modular y escalable. Su diseño RESTful facilita la integración con cualquier frontend o aplicación externa que requiera gestionar contenido de blogs de manera eficiente y segura.

# INTRODUCCIÓN

El proyecto BlogApp es una API REST desarrollada con Spring Boot que permite la gestión de usuarios, publicaciones y comentarios. Su funcionalidad incluye la creación, consulta, listado y eliminación de estos elementos, proporcionando endpoints estructurados y documentados con Swagger.

La API utiliza Spring Boot, MySQL y Hibernate (JPA) para garantizar una arquitectura modular y escalable. Su diseño RESTful facilita la integración con cualquier frontend o aplicación externa que requiera gestionar contenido de blogs de manera eficiente y segura.

# DIAGNÓSTICO

Para identificar oportunidades de mejora, se realizó un análisis SEPTE considerando las siguientes variables:

**Social:**

* La creación/consumo de contenido corto y la interacción (comentarios/reacciones) son expectativas básicas de los usuarios.
* Accesibilidad y desempeño en móviles son críticos para adopción.

**Económico:**

* + Preferencia por stack open-source para minimizar costos.
  + Escalabilidad elástica en la nube evitará sobredimensionar recursos en etapas tempranas.

**Político-legal:**

* + Cumplimiento de políticas de privacidad y protección de datos personales; manejo seguro de contraseñas y archivos.
  + Términos de uso y moderación de contenido para evitar responsabilidad por publicaciones de terceros.

**Tecnológico:**

* + Stack actual (Spring Boot + Angular + MySQL) es estándar, con amplio soporte y ecosistema.
  + Recomendable introducir JWT, CI/CD, monitorización (Prometheus/Grafana o equivalente) y almacenamiento de archivos en S3/Blob.

**Ecológico:**

* + Uso eficiente de recursos (containers, auto-scaling) y ambientes compartidos reduce huella energética.
  + Evitar archivos innecesariamente pesados y políticas de retención/limpiez

# OBJETIVOS

**Objetivo Principal**:

Lograr una plataforma web de blogs eficiente en términos de funcionalidad, usabilidad y diseño, que permita a los usuarios crear, editar y gestionar contenido de manera efectiva.

**Objetivos Específicos**:

* + **MEJORAR LA FUNCIONALIDAD DE REGISTRO Y AUTENTICACIÓN DE USUARIOS:** El objetivo es implementar un sistema seguro para crear cuentas y acceder a la plataforma, con un tiempo de registro menor a 3 segundos y una tasa de éxito en autenticación del 90%. Se utilizará Spring Security y se realizarán pruebas de funcionalidad. La implementación debe completarse en 2 meses.
  + **OPTIMIZAR LA INTERFAZ DE CREACIÓN Y EDICIÓN DE BLOGS**:   
    El objetivo es mejorar la creación y edición de publicaciones, garantizando accesibilidad en dispositivos móviles y de escritorio. Se medirá la satisfacción del usuario, buscando que el 90% pueda crear y editar blogs sin ayuda. Se aplicará diseño responsivo y pruebas de usabilidad. La optimización debe completarse en 3 meses.
  + **INCORPORAR UNA SECCIÓN DE COMENTARIOS INTERACTIVOS EN LAS PUBLICACIONES:** El objetivo es integrar una sección de comentarios en cada publicación para fomentar la interacción entre usuarios. Se medirá el número de comentarios y la tasa de interacción, con la meta de alcanzar 100 comentarios diarios en publicaciones populares. Se usará Spring Boot y Thymeleaf para crear el sistema, y la implementación debe completarse en 4 meses.

# JUSTIFICACIÓN

El desarrollo de BlogApp se justifica en la necesidad de contar con una plataforma web moderna que permita la creación, gestión e interacción de contenidos digitales. En un entorno donde la comunicación y la difusión de información son fundamentales, disponer de un sistema escalable y seguro facilita la participación de usuarios mediante publicaciones y comentarios.

Además, el proyecto integra tecnologías actuales como Spring Boot, MySQL y Angular, lo que garantiza la aplicación de buenas prácticas de desarrollo, fomentando en los estudiantes habilidades en la construcción de aplicaciones empresariales completas (full stack). Así, la solución no solo resuelve una necesidad práctica de gestión de blogs, sino que también contribuye a la formación académica y profesional en entornos tecnológicos demandados por el mercado laboral.

# DEFINICIÓN Y ALCANCE

**Definición:**

BlogApp es una plataforma web con arquitectura cliente-servidor, que expone un API REST en el backend y un frontend desarrollado en Angular. El sistema permite gestionar usuarios, publicaciones y comentarios, integrando seguridad mediante Spring Security y almacenamiento en una base de datos relacional MySQL.

**Alcance:**

* + Registro, autenticación y administración básica de usuarios.
  + Creación, edición, listado y eliminación de publicaciones.
  + Gestión de archivos adjuntos (imágenes) en las publicaciones.
  + Interacción social mediante comentarios.
  + Interfaz web accesible desde navegadores modernos.
  + Documentación de los endpoints vía Swagger.

El alcance inicial se centra en funcionalidades esenciales (CRUD de usuarios, publicaciones y comentarios), dejando abierta la posibilidad de futuras ampliaciones como notificaciones, categorías y métricas de interacción.

# PRODUCTOS Y ENTREGABLES

## USUARIO – CONTROLLER

## LOGIN Y REGISTRO DE USUARIO

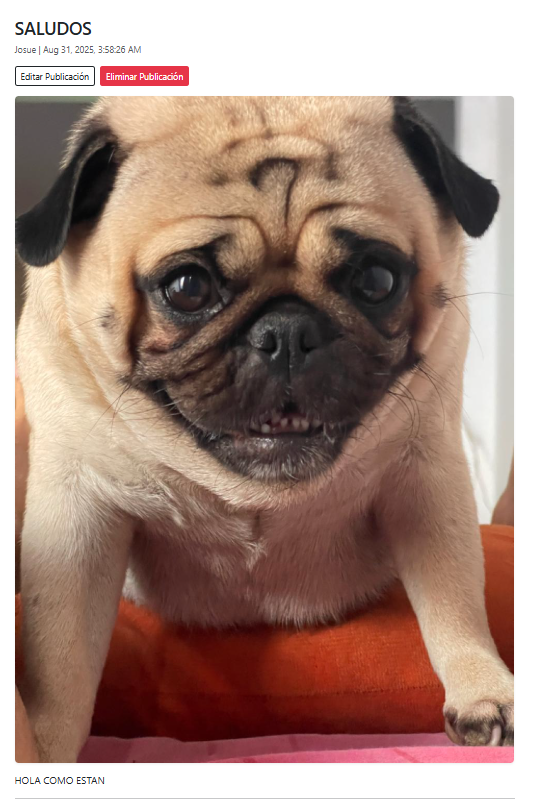
Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

## PUBLICACIÓN – CONTROLLER

## PUBLICACIONES

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

# CONCLUSIONES

* Se logró implementar una solución web completa que combina backend, frontend y base de datos, permitiendo a los usuarios interactuar mediante publicaciones y comentarios.
* La arquitectura basada en Spring Boot y Angular garantizó modularidad, escalabilidad y buenas prácticas de desarrollo.
* La integración de Spring Security y BCrypt proporcionó un nivel adecuado de seguridad en la autenticación de usuarios.
* El proyecto permitió aplicar los conocimientos adquiridos en programación, bases de datos y desarrollo web, consolidando la experiencia práctica en un entorno realista.

# RECOMENDACIONES

* Ampliar la funcionalidad con el uso de JWT en lugar de Basic Auth para mejorar la seguridad y escalabilidad del sistema.
* Implementar pruebas unitarias y de integración automatizadas para garantizar la calidad del software.
* Migrar el despliegue a un servicio en la nube (Heroku, AWS o Azure) para asegurar disponibilidad y acceso remoto.
* Incluir métricas de rendimiento y monitoreo de logs para facilitar el mantenimiento.
* Extender el sistema con nuevas funcionalidades como categorías de publicaciones, “likes” y panel de administración para moderación.

# GLOSARIO

* **Spring Boot:** Framework de Java para crear aplicaciones web rápidas y escalables.
* **Angular:** Framework frontend basado en TypeScript para aplicaciones SPA.
* **JWT (JSON Web Token):** Estándar para autenticación y autorización segura.
* **BCrypt:** Algoritmo de cifrado para contraseñas.
* **CRUD:** Operaciones básicas de Crear, Leer, Actualizar y Eliminar.

# BIBLIOGRAFÍA

* drawSQL. (2025). *Herramienta de modelado de bases de datos relacionales*. <https://drawsql.app>
* MDN Web Docs. (2025). *AJAX, JSON y JavaScript modernos*. <https://developer.mozilla.org>
* [Bizagi - Líder en Automatización Inteligente de Procesos](https://www.bizagi.com/es)
* Spring Boot Documentation. (2025). <https://spring.io/projects/spring-boot>
* Angular Docs. (2025). <https://angular.dev>
* MDN Web Docs. (2025). REST APIs y JSON. <https://developer.mozilla.org>