LIBRERÍA PILLOW PYTHON

PRESENTADO POR

KEVIN CAMILO DELGADO R

CENTRO DE BIOTECNOLOGÍA AGROPECUARIA (CBA) SENA

ANÁLISIS Y DESARROLLO DE SOFTWARE

2025

ESTE PROYECTO FUE DESARROLLADO COMO PARTE DEL PROCESO FORMATIVO DEL PROGRAMA DE TECNÓLOGO EN ANÁLISIS Y DESARROLLO DE SOFTWARE DEL CENTRO DE BIOTECNOLOGÍA AGROPECUARIA (CBA) DEL SENA.

APLICACIÓN DE LOS NIVELES CMMI EN EL PROGRAMA DE DESARROLLO DE NUMEROS FACTORIALES

ES UN PROGRAMA QUE FACILITA BUSCAR EL FACTORIALA DE LOS NUMEROS, SIENDO UNA TAREA MAS ASENCILLA PARA EL USUARIO, YA QUE MI PROGRAMAA CUENTA CON LA OPTIMIZACION Y LOS NIVELES DE CMMI PARA QUE SEA MAS UTIL Y MAS SENCILLA DE USAR

PROYECTO PRESENTADO POR:  
KEVIN CAMILO DELGADO R.

TUTOR:  
I

SENA – CENTRO DE BIOTECNOLOGÍA AGROPECUARIA  
MOSQUERA, CUNDINAMARCA  
2025

Contenido

[Resumen 4](#_Toc200916841)

[1. Funcionalidad ¿Para qué sirve? 5](#_Toc200916842)

[2. Sintaxis básica: Ejemplo de uso. 6](#_Toc200916843)

[3. Aplicaciones reales 7](#_Toc200916844)

[Ejemplo de código: 8](#_Toc200916845)

# Resumen

Este documento muestra la introducción de usar la librería **PILLOW**, esta librería nos ayuda con las imágenes, editándolas con filtros, ajustando pixeles, hasta incluso cambia su resolución y su peso. Esta es muy importante para nosotros los desarrolladores web.  
Ya que facilita a la hora de que un usuario quiera subir una imagen, también una miniatura.  
Esto librería no se demora nada en subir las imágenes

# 1. Funcionalidad ¿Para qué sirve?

Pillow es una potente biblioteca de procesamiento de imágenes que se utiliza para realizar una amplia gama de operaciones en imágenes, como cambiar su tamaño, recortar, rotar y mucho más. Es una bifurcación de Python Imaging Library (PIL), que es una biblioteca de referencia de Python para trabajar con imágenes.

Pillow ofrece una amplia gama de opciones para trabajar con imágenes y es compatible con muchos formatos populares como JPEG, PNG, GIF e incluso formatos menos comunes como TIFF y WebP. Para usar Pillow, primero debemos instalarlo en nuestro PC local mediante PIP.

# 2. Sintaxis básica: Ejemplo de uso.

*rom* PIL *import* Image

imagen = Image.open("boca.jpg")

imagen.show()

nueva = imagen.resize((200, 200))

nueva.save("nueva\_imagen.jpg")

# 3. Aplicaciones reales

**1. Edición de imágenes en proyectos web**

**2. Automatización de tareas de imágenes (como redimensionar muchas imágenes a la vez)**

**3. Filtros y efectos en edición básica**

**4. Generación de miniaturas (thumbnails)**

**5. Creación de memes o imágenes personalizadas automáticamente**

**6. Edición de imágenes en aplicaciones gráficas y videojuegos**

## 

## Ejemplo de código:

*from* PIL *import* Image

# *Cargar la imagen original*

imagen = Image.open("ejemplo.jpg")

# *Convertir la imagen a blanco y negro*

blanco\_negro = imagen.convert("L")

# *Guardar la imagen modificada*

blanco\_negro.save("blanco\_y\_negro.jpg")

# *Mostrar el resultado*

blanco\_negro.show()