**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS**

**Escuela de Posgrado**

**FACULTAD DE INGENIERIA DE SISTEMAS E INFORMATICA**

**Unidad de Posgrado**

**Sistema Web de Escaneo y Clasificación de Productos por fecha de vencimiento**

Dikxon Angel Luque Ticona

Ganymedes Rosales Reyes

Justo Pérez Soncco

Fecha: 10/08/2025

# Índice

1. **Introducción**
2. **Justificación**
3. **Objetivos**
4. **Marco Teórico**
5. **Arquitectura del Sistema**
6. **Diseño e Implementación**
7. **Arquitectura functional**
8. **Descripción de la arquitectura funcional y de sistemas**
9. **Arquitectura de componentes**
10. **Base de datos**
11. **Modelo de entrenamiento**
12. **Imágenes abiertas y cerradas**
13. **Manual de usuario**
14. **Conclusiones**
15. **Referencias**

# 1. Introducción

El presente documento describe en detalle el desarrollo de un sistema web integral para la detección de códigos de barras, lectura de fechas de vencimiento y clasificación de productos en estados 'Abierto' o 'Cerrado' usando técnicas de Visión por Computadora y Deep Learning. El proyecto fue desarrollado con FastAPI como backend, PostgreSQL como sistema de base de datos y un frontend basado en HTML5, CSS y JavaScript.

# 2. Justificación

En la gestión de inventarios y control de calidad de productos, es fundamental contar con herramientas que permitan identificar, registrar y analizar la información de los artículos de forma rápida y precisa. Este sistema surge como una solución a la necesidad de automatizar este proceso, reduciendo errores humanos y optimizando el tiempo de inspección.

# 3. Objetivos

Objetivo General:

Desarrollar una aplicación web que permita escanear, identificar y clasificar productos de forma automática utilizando tecnologías de visión artificial y deep learning.

Objetivos Específicos:

• Detectar códigos de barras en tiempo real desde la cámara o videos pregrabados.

• Extraer fechas de vencimiento mediante OCR.

• Clasificar el estado de los productos en 'Abierto' o 'Cerrado'.

• Almacenar los datos en PostgreSQL con SQLAlchemy.

• Visualizar resultados de forma ordenada y con indicadores de proximidad de vencimiento.

# 4. Marco Teórico

Este sistema combina múltiples áreas de la computación:

* Visión por Computadora: técnicas para procesar y analizar imágenes.
* Aprendizaje Profundo: redes neuronales para clasificación de imágenes (ResNet18).
* OCR: reconocimiento óptico de caracteres para leer fechas impresas.
* Bases de Datos: almacenamiento y consulta de registros.
* Desarrollo Web: integración de backend y frontend.

# 5. Arquitectura del Sistema

La arquitectura sigue un modelo cliente-servidor:

1. Frontend (HTML, CSS, JS) para interacción y visualización.
2. Backend (FastAPI) para procesamiento de imágenes y gestión de datos.
3. Base de datos PostgreSQL para persistencia.
4. Modelos IA en PyTorch para clasificación.

# 6. Diseño e Implementación

El sistema se estructura en módulos independientes:

* Módulo de escaneo de código de barras.
* Módulo OCR para fecha de vencimiento.
* Módulo de clasificación Abierto/Cerrado.
* API REST con endpoints específicos.
* Gestión de base de datos con SQLAlchemy.

**7. Arquitectura funcional**

Diagrama

El contenido generado por IA puede ser incorrecto.

**8. Descripción de la Arquitectura Funcional del Sistema**

Este diagrama representa la arquitectura funcional del sistema de fecha de vencimiento y detección de productos manipulados o abiertos en refrigeradores, utilizando visión por computadora. El flujo es completamente secuencial e interactivo, guiando al usuario paso a paso para asegurar capturas precisas.

1. Cámara Web: El sistema inicia con la activación de la cámara web integrada, que sirve como fuente de entrada visual.
2. Solicitud de Enfoque sobre el Producto: El sistema solicita al usuario que enfoque el producto individualmente para asegurar una captura clara y centrada.
3. Lectura del Código de Barras: Utilizando la librería DBR (Dynamsoft Barcode Reader), se identifica el código de barras del producto. Esta lectura permite registrar el tipo de producto.
4. Almacenamiento del Código en Base de Datos: El código extraído se guarda en la base de datos, generando el primer registro asociado al producto.
5. Detección de Fecha de Vencimiento (Optical Character Recognition - OCR): El sistema solicita enfocar la zona donde se encuentra la fecha de vencimiento. Con EasyOCR, se detecta y extrae el texto correspondiente.
6. Almacenamiento de la Fecha en Base de Datos: La fecha leída se almacena junto al código de barras del producto, enriqueciendo el registro.
7. Solicitud de Enfoque sobre el Envase: Luego, se solicita al usuario enfocar el producto completo o su zona crítica (tapa, sello, anilla) para determinar su estado físico.
8. Clasificación Visual con YOLOv8: Se utiliza un modelo YOLOv8 previamente ajustado mediante transfer learning para clasificar el estado del producto como “abierto” o “cerrado”.
9. Almacenamiento del Estado + Visualización Final: El resultado de la clasificación se almacena en la base de datos. Finalmente, se presenta una visualización consolidada con toda la información recolectada: código de producto, fecha de vencimiento y estado físico.

Este enfoque modular y guiado asegura un control preciso sobre cada etapa, y permite validar la integridad de los productos en entornos de refrigeración en tiempo real.

**9. Arquitectura de componentes**

Diagrama

Descripción generada automáticamente

# 10. Base de Datos

Fue utilizado Base de datos PostgreSQL.

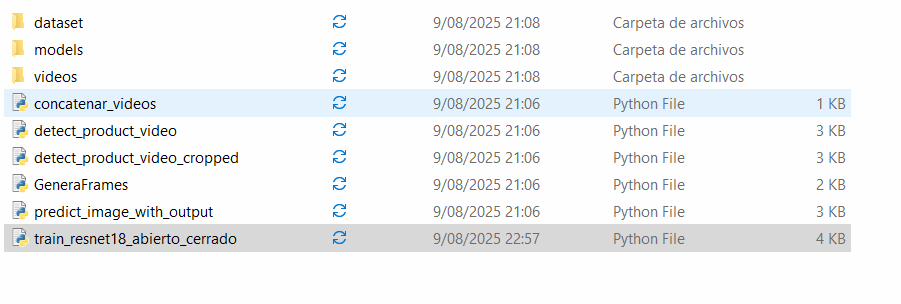
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

# Modelos para ENTRENAMIENTOS.

**Entrenamiento:** El entrenamiento del modelo se realizará mediante la técnica “Transfer learning” de una red neuronal pre entrenada basada en la arquitectura YOLOv8 (You Only Look Once), utilizando el entorno de ejecución con GPU de **Google Colab PRO (**Plataforma **CLOUD,** que soporta entrenamiento a demanda, utilizado para procesos pesados**)**, esto nos permite sesiones extendidas y una mayor prioridad de recursos.

Este proyecto utilizará la técnica **transfer-learning**, lo que implica que partimos de un modelo pre entrenado (yolov8n.pt o yolov8s.pt) y entrenamos principalmente las últimas capas responsables de la detección de clases nuevas de productos ("abierto" o "cerrado"). Este enfoque permite adaptar eficientemente el modelo a nuestro dominio específico sin necesidad de reentrenar todas las capas desde cero, lo cual es ideal para pruebas de concepto con datasets (conjunto de imágenes, donde hay imágenes normales y anómalos) limitados y clases similares a las del entrenamiento original.



Interfaz de usuario gráfica, Aplicación

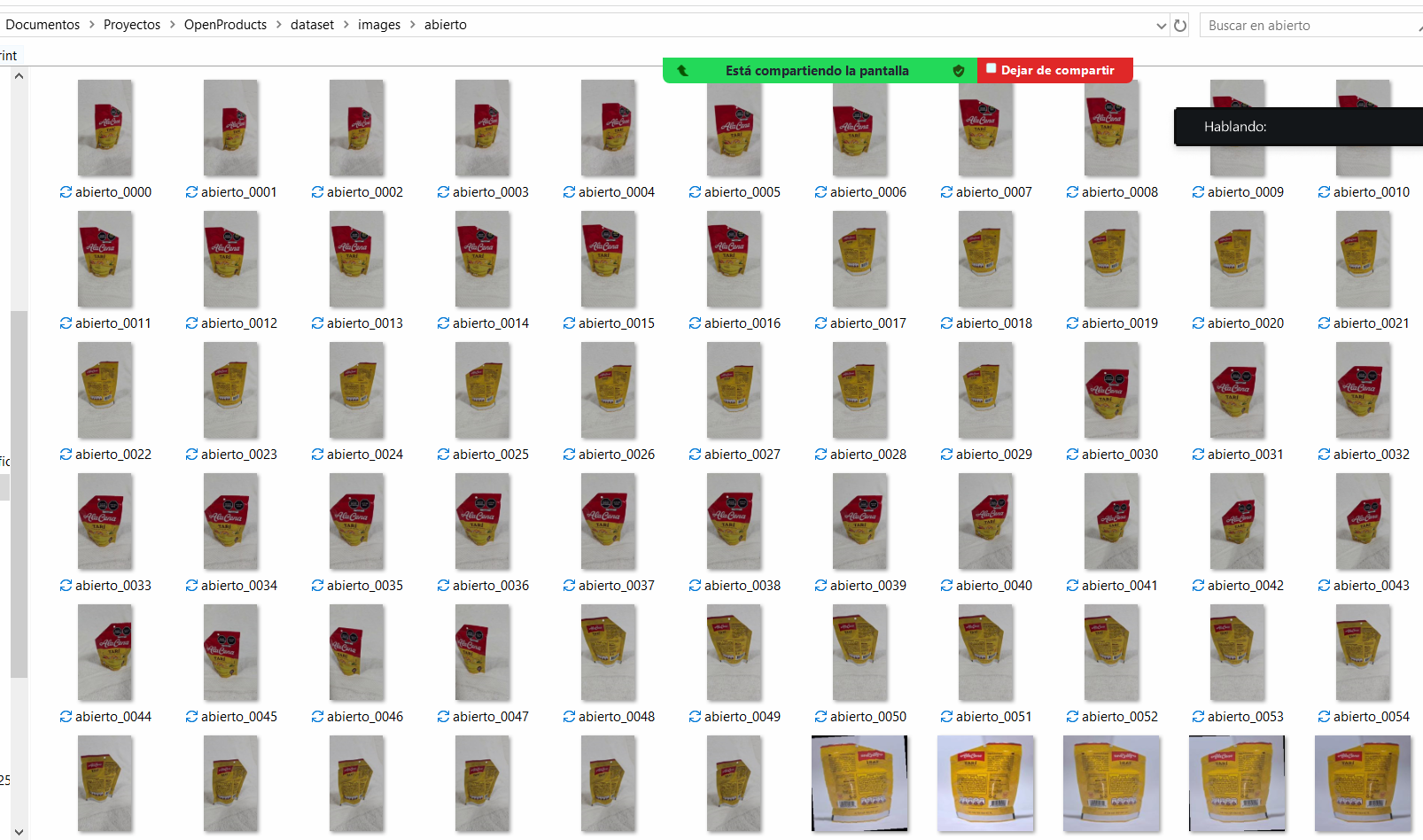
Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**12. Imagenes abiertas y Cerradas**

**Imagen abierta**



**Imágenes cerradas**

Imagen que contiene objeto

Descripción generada automáticamente

**13. Manual de Usuario**

Pantalla WEB e interfaces

Captura de pantalla de un celular con la imagen de una persona

Descripción generada automáticamente

**Registro**

El usuario debe preparar el ambiente ejecutando el siguiente comando DOS

venv\Scripts\activate

Luego, debe ejectuar en DOS lo siguiente:

uvicorn app:app --reload --host 0.0.0.0 --port 8000

Por último el usuario debe ejecutar en un navegador lo siguiente:

<http://localhost:8000/>

luego se mostrará la siguiente pantalla:



1. Dar click en la opción START. Para activar las opciones de activas lectura código de Barra

Interfaz de usuario gráfica

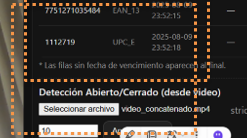
Descripción generada automáticamente

1. Debemos mostrar en pantalla el código de barra del producto. Para que sea capturado.

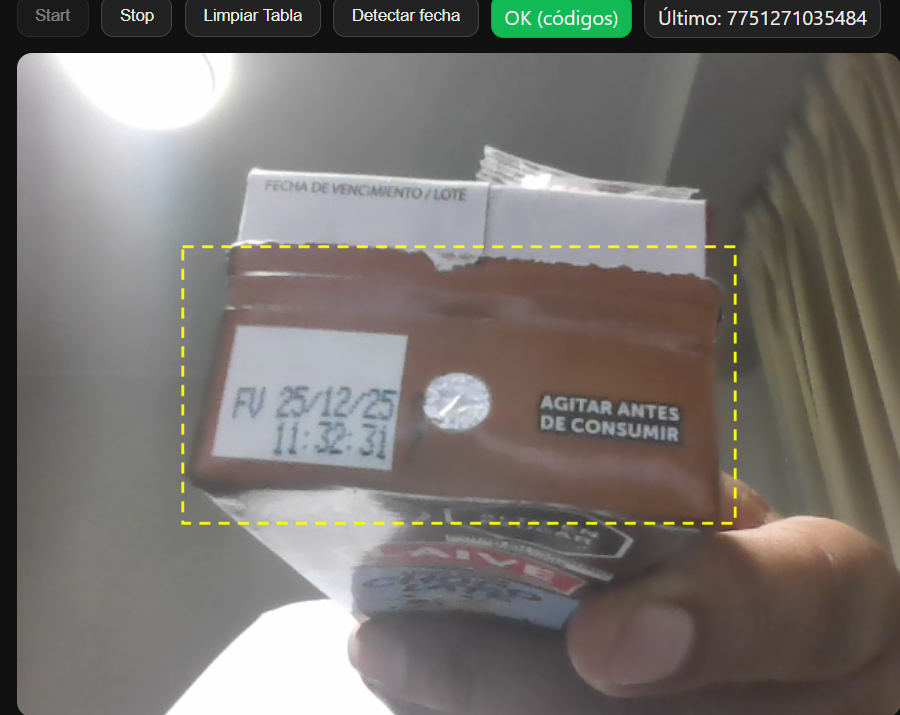
Pantalla de un celular en la mano

Descripción generada automáticamente

1. El código de barra detectado será mostrado al lado derecho de la pantalla

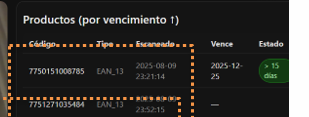


1. Luego capturaremos la fecha de vencimiento.



1. Una vez captura la fecha de vencimiento será registrado en la pantalla lado derecho.

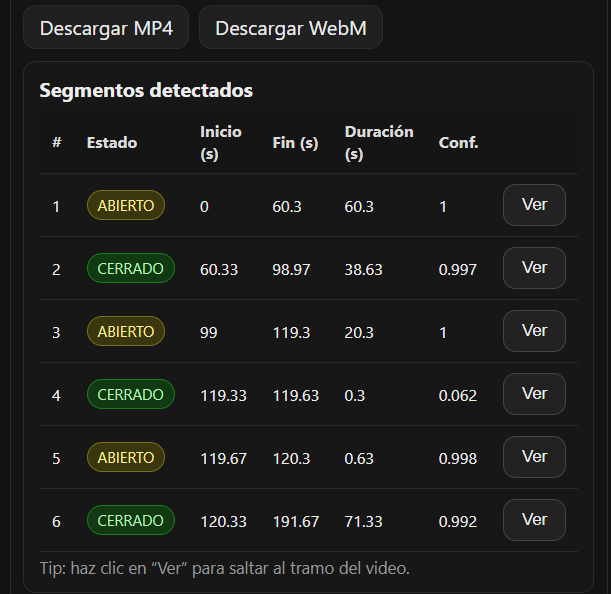
En el código se encuentra alarma por colores: Rojo menor igual a 5 dias. Amarrillo de 6 a 10 días, Verde mayor a 10 dias.



1. Puedo detectar si un producto esta abierto. Nos muestra mensaje lado derecho



1. Podemos observar la lista de productos registrados



**14. Conclusiones**

Logramos cumplir con el desarrollo de los procesos de 1.) Detectar códigos de barras en tiempo real desde la cámara o videos pregrabados y 2.) la captura de fecha de vencimiento de productos. Para nuestro ejercicio se trabajo con 4 productos.

Se implemento regla de negocio al establecer alarmas de vencimiento del producto: Color rojo menor o igual a 5 días, color amarillo de 6 a 10 días inclusive y color verde mayor a 10 días.

Utilizamos BD PostgreSQL para registrar los productos y los archivos de movimiento o de registro.

Una buena oportunidad de aprendizaje para trabajar con producto en movimiento y la creación de entrenamiento para conocer si producto esta abierto o cerrado.

**15. Referencias bibliográficas**

1. "Expiry Date Recognition Using Deep Neural Networks", Autores: Vlad Florea, Traian Rebedea Publicado en: International Journal of User-System Interaction, 2020
2. "Robust Deep-Learning Based Refrigerator Food Recognition", Autores: Xiaoyan Dai. Publicado en: Frontiers in Artificial Intelligence, 2024.
3. "AI-Based Food Spoilage Detection", Autores: Ms. Mulla Aarshiya Yousuf, Prof. Ankush Dhamal

Publicado en: International Journal of Progressive Research in Engineering Management and Science (IJPREMS), 2025