

COLEGIO ESPAÑOL PADRE ARRUPE
COLEGIO INTERNACIONAL



ACTIVIDAD

Avance: Sistema de detección de objetos o formas.
OutfitCombiner

ASIGNATURA

Diseño de Aplicaciones con IA

DOCENTE

Katherine Canales

ESTUDIANTE

Número	Nombre
2	Alexandra Elizabeth Alvarado Bautista

FECHA DE ENTREGA

17/08/2025

Contexto

La deficiencia de visión cromática (CVD o daltonismo) afecta la capacidad de distinguir ciertos tonos, especialmente en los pares rojo-verde (deuteranopia/protanopia) y azul-amarillo (tritanopia). En la vida diaria, esta limitación se vuelve crítica al elegir ropa: las fotos de prendas tienen fondos y luces variables, los balances de blanco cambian el tono aparente y las texturas/patrones añaden ruido visual. Además, la mayoría de reglas de “armonía de color” están escritas para quienes perciben bien el color, por lo que no ofrecen una guía clara y objetiva para quienes tienen CVD. En este contexto nace **OutfitCombiner**, una herramienta que no depende de “ver bien el color”, sino de **mediciones objetivas** para decir si dos prendas combinan y explicar por qué.

Problema

Personas con daltonismo reportan dos dificultades recurrentes al vestir: 1) **confundir tonos cercanos** (por ejemplo, azul marino vs. negro, beige vs. blanco), lo que lleva a conjuntos con **muy poco contraste**; y 2) **sobrecargar el conjunto** al combinar **dos patrones fuertes** o **complementarios muy saturados** sin un color neutro que los ancle. Esto provoca indecisión, dependencia de terceros y pérdida de tiempo. Las soluciones actuales (apps que nombran colores, blogs con “reglas”) no responden la pregunta concreta “¿estas dos prendas específicas combinan?” ni entregan una **explicación textual** que una persona con CVD pueda comprender sin ver el color.

Necesidad central: una evaluación binaria y comprensible—**Combinan / No combinan**—acompañada de una **explicación corta y accionable** (contraste, tono, patrón) que aumente la independencia y la confianza al vestir.

Objetivo y alcance

Objetivo: permitir que el usuario suba dos imágenes de prendas y reciba un veredicto con explicación y recomendación en pocos segundos.

Alcance del prototipo: procesar dos imágenes por sesión; detectar y cortar la prenda en la foto; extraer características interpretables (tono, contraste, saturación, patrón); y decidir con un clasificador entrenable por el propio usuario. Todo corre en Google Colab y guarda datos en Google Drive, sin APIs externas.

Qué implementamos en Colab (visión general)

OutfitCombiner se construyó como un cuaderno de Colab con celdas modulares:

1. **Gestión de archivos en Drive.** Se definen carpetas [/images](#), [/data](#) y [/models](#). En [/data/combina_dataset.csv](#) se registran automáticamente todas las predicciones y ejemplos etiquetados; en [/models/combina_model.joblib](#) se guarda el modelo entrenado.
2. **Lectura de imágenes y gestión de color.** Para evitar cambios de tono por perfiles ICC, se implementó una lectura **sRGB/ICC-safe**. Esto asegura que los colores analizados

correspondan a lo que el usuario espera.

3. **Extracción de paletas y métricas objetivas.** Para cada prenda se obtiene una **paleta dominante** y se calculan variables como **H/S/V**, **luminancia** (WCAG), **contraste**, **diferencia de tono** (ΔH , en grados), **diferencia perceptual** (ΔE , en Lab) y un **proxy de patrón** mediante **densidad de bordes** (Canny). Estas variables son **explicables** y mapean directamente a razones comprensibles (“poco contraste”, “dos patrones fuertes”, “tonos muy cercanos”).
4. **Clasificador interpretable y entrenable.** Con **scikit-learn** se entrena un modelo sencillo (LogisticRegression o RandomForest). El entrenamiento es **robusto**: valida el CSV, maneja datasets pequeños/desbalanceados y, si solo hay una clase, usa un **DummyClassifier** para no fallar.
5. **Predicción con explicación.** El cuaderno ofrece funciones para **subir dos imágenes**, preprocesarlas, **predecir** y **explicar** el resultado con lenguaje claro. Se imprimen métricas clave (contraste, ΔH , patrón) y una **recomendación** (“añade un neutro”, “sube el contraste”, “evita dos patrones fuertes a la vez”). Cada caso queda **logueado** en el CSV.
6. **Detección opcional de prenda.** Con **GroundingDINO** (PyTorch) se pueden recortar automáticamente las prendas en fotos reales. Esto mejora la calidad de la paleta de color al excluir fondos. Las detecciones se muestran con títulos en español (“**Camisa**” / “**Pantalón**”). El modelo de combinación funciona tanto con recortes como con imágenes completas.

Design Thinking de *OutfitCombiner*



Empatizar

Understand your user

Beneficios

Autonomía al vestir

Veredicto rápido

Explicación en texto (contraste, tono, patrón)

Acceso rápido e intuitivo, interfaz fácil de usar

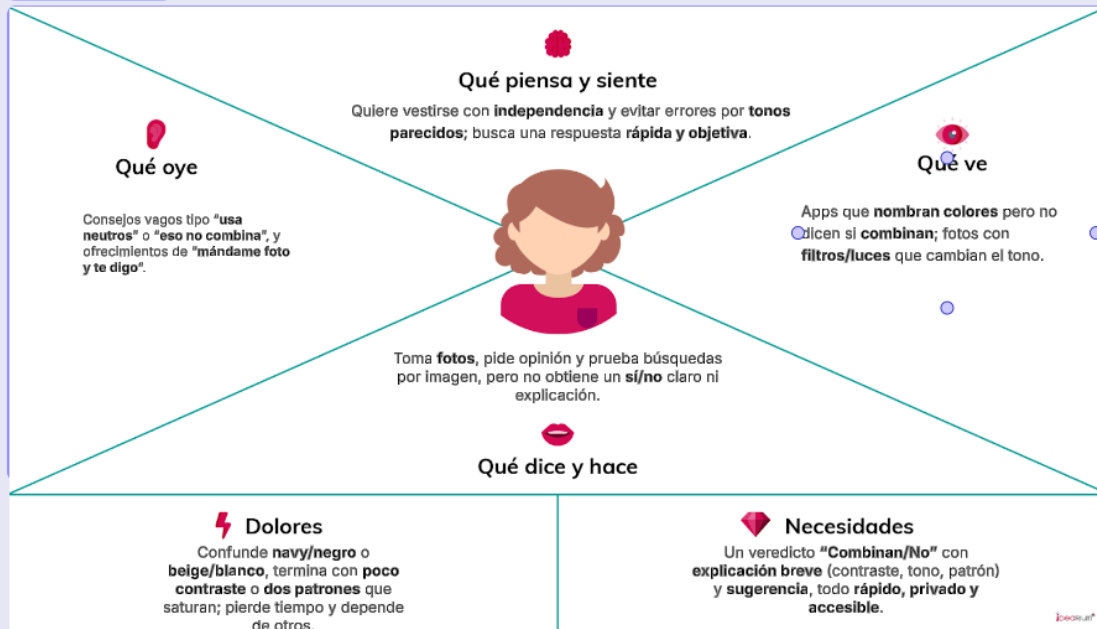
Sugerencias prácticas (añade neutro, sube contraste)

Funciona con mis fotos (detección/recorte evita fondo)

Privado y personalizable



Mapa de empatía





Define

Distill your research

Resumen de Investigación

Personas con **daltonismo** confunden tonos cercanos (p. ej., navy vs. negro, beige vs. blanco) y terminan con **poco contraste** o **choque de patrones**. Las apps que **nombran colores** no responden si **dos prendas combinan**, y las fotos con distintos **filtros/luces** alteran el tono. En la práctica dependen de terceros, lo que genera **ansiedad y pérdida de tiempo**. Prefieren una **respuesta objetiva y rápida** con **explicación textual** (contraste, tono, patrón), que funcione con **sus fotos** y respete su **privacidad**.



Declaración principal

Las personas con daltonismo necesitan una herramienta que, a partir de dos fotos, les diga si las prendas *combinan* o no, con una explicación breve y una sugerencia práctica, de forma rápida, accesible y privada.

Posibles declaraciones de punto de vista

- **Personas con daltonismo** necesitan una herramienta que, a partir de **dos fotos**, les dé un **veredicto claro (Combinan/No)** con **explicación corta** basada en **contraste, tono y patrón**, para decidir sin depender de otros.
- Requieren que funcione con **imágenes reales** (detección/recorte para evitar fondo) y que sea **rápida** (≤ 10 s) y **privada** (datos en su Drive).
- Valoran **sugerencias accionables** ("añade un neutro", "sube contraste") y **alternativas** cuando no combina.



Ideate

Distill your research

Ideas

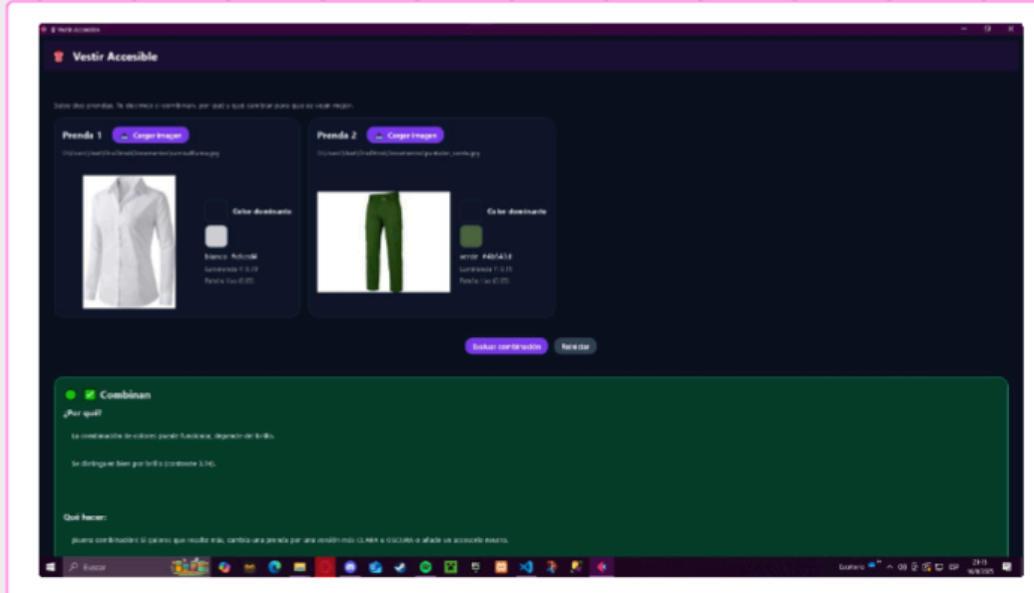
- **Evaluador objetivo con explicación y tip**
Dos fotos \rightarrow veredicto *Combinan/No*, **explicación breve** (contraste, $\Delta H/\Delta E$, patrón) y **recomendación** ("añade un neutro", "sube contraste").
- **Recorte de prenda para paleta limpia**
GroundingDINO para detectar y recortar la prenda; **fallback** con máscara HSV si no hay detección. Mejora el color medido.
- **Personalización y aprendizaje**
Guardar la decisión del usuario (sí/no) en el **CSV** y **reentrenar**; sliders de **tolerancia de tono** y **nivel de contraste** según gusto.
- **Sugerencias automáticas accesibles**
Si "no combinan", proponer **alternativas**: neutro compatible, bajar saturación, subir contraste; mostrar **paletas amigables para CVD**.
- **Interfaz clara y accesible**
Mensajes en **texto** e **iconos no cromáticos**, **alto contraste** y opción de **lectura por voz**; flujo completo en ≤ 10 s.



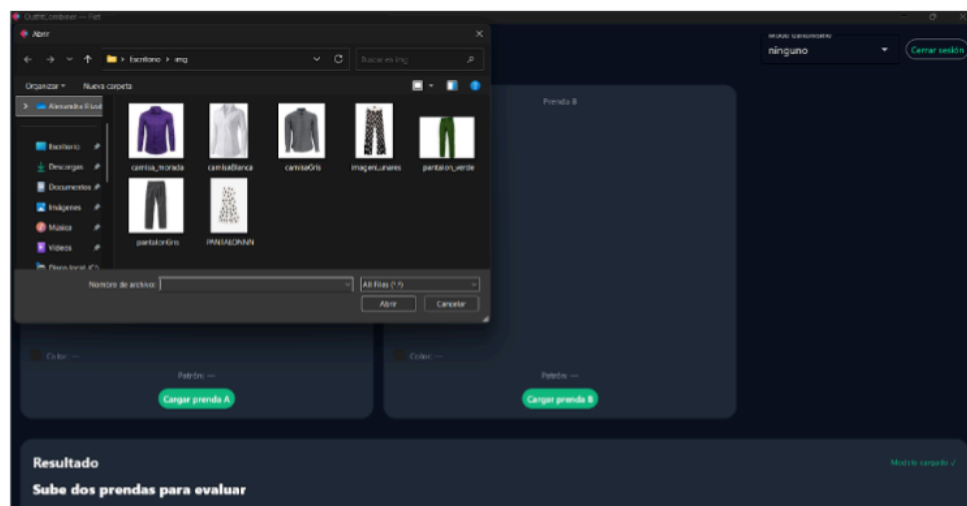
✨ Prototype

Flesh out your ideas from brainstorming

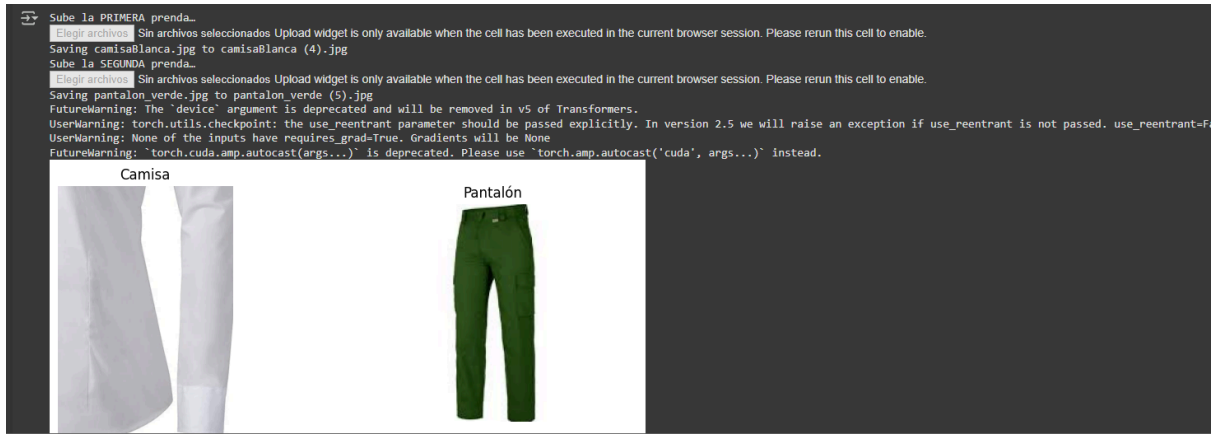
Prototype concept



Prototype concept



Implementación hecha, con resultados parciales



Flujo de la implementación

Propósito general: procesar dos imágenes de prendas para (i) detectar y recortar cada prenda, (ii) titular los recortes en español, (iii) extraer características de color/patrón, (iv) predecir si combinan, (v) explicar el veredicto y (vi) registrar todo en un CSV.

1) Ingesta de imágenes

Entrada: dos archivos de imagen (o rutas locales).

Procedimiento: carga vía `files.upload()` en Colab o lectura desde ruta; los archivos se guardan con nombre único (timestamp) en `IMAGES_DIR`.

Salida: `path1`, `path2` (rutas persistentes).

2) Detección y recorte de prendas

Entrada: `path1`, `path2`; *prompt* de clases (camisa, pantalón, falda, etc.).

Procedimiento: `detect_and_crop()` ejecuta GroundingDINO, selecciona una caja por *score* (“best”) o por área (“largest”) y recorta la región. La frase detectada se mapea a etiqueta en español (`Camisa`, `Pantalón`, ...).

Salida: para cada imagen: `crop` (BGR), `annotated` (RGB con cajas), `label_es` (etiqueta).

3) Rotulación visual

Entrada: `cropA`, `cropB`, `label_es` por prenda.

Procedimiento: render de los dos recortes en una figura, con el **título superior** correspondiente (“Camisa”, “Pantalón”).

Salida: visualización de los recortes titulados; opcionalmente, figuras con las detecciones anotadas.

4) Extracción de características (features)

Entrada: `cropA`, `cropB`.

Procedimiento: `pair_features()` calcula, entre otras, **contraste de luminancia (WCAG)**, **luminancia** y **diferencia**, **HSV** (saturación/valor) de los colores principales, **distancia de tono (dH)**, **distancia en Lab (dE)**, **patrón** (densidad de bordes), **neutralidad** y **estadísticos de paleta A_*/B_***.

Salida: `feats` (diccionario de características) y `meta` (colores principales, etc.).

5) Predicción

Entrada: `feats` y `feature_cols` (orden esperado por el modelo).

Procedimiento: construcción del vector **X**; inferencia con `model` (`predict_proba` o equivalente).

Salida: `pred` (1=Combinan, 0=No combinan) y `proba` (confianza/score).

6) Explicación del resultado

Entrada: `feats`, `meta`.

Procedimiento: `explain_pair()` genera notas interpretables (p. ej., “colores casi complementarios”, “contraste alto”) y recomendaciones de mejora cuando aplique.

Salida: `notes` (porqué) y `recs` (cómo mejorar o recomendación).

7) Registro en CSV (trazabilidad)

Entrada: `feats`, `meta`, rutas de imagen, `pred`, `proba`, `notes`, `recs`, etiquetas detectadas.

Procedimiento: asegurar el esquema con `ensure_csv()/_ensure_columns_for_logging()` y anexar una fila con **todas** las columnas (features, rutas, colores principales, etiquetas ES, predicción, confianza,

explicación, [timestamp](#), [mode](#), [source](#)).

Salida: actualización de [DATASET_CSV](#) y confirmación en consola.

Evidencia observada en la captura

Panel superior:

- Dos imágenes lado a lado con títulos “Camisa” y “Pantalón” (crops correctos).

Panel inferior:

- **Detecciones A — Camisa y Detecciones B — Pantalón** con **cajas** resaltando cada prenda.
- En consola se imprime el **veredicto**: “**COMBINAN / NO COMBINAN** ” y la **confianza** (probabilidad).
- Mensaje final confirmando “**Guardado en dataset: .../combina_dataset.csv**”.

Resultados parciales (alcanzados)

- End-to-end funciona en el ejemplo: carga → detecta → recorta → titula → predice → explica → guarda.
- Etiquetas en ES asignadas correctamente (Camisa/Pantalón).
- Crops y anotaciones coherentes con la prenda objetivo.
- Registro en CSV con todas las columnas necesarias para auditoría y reentrenamiento.



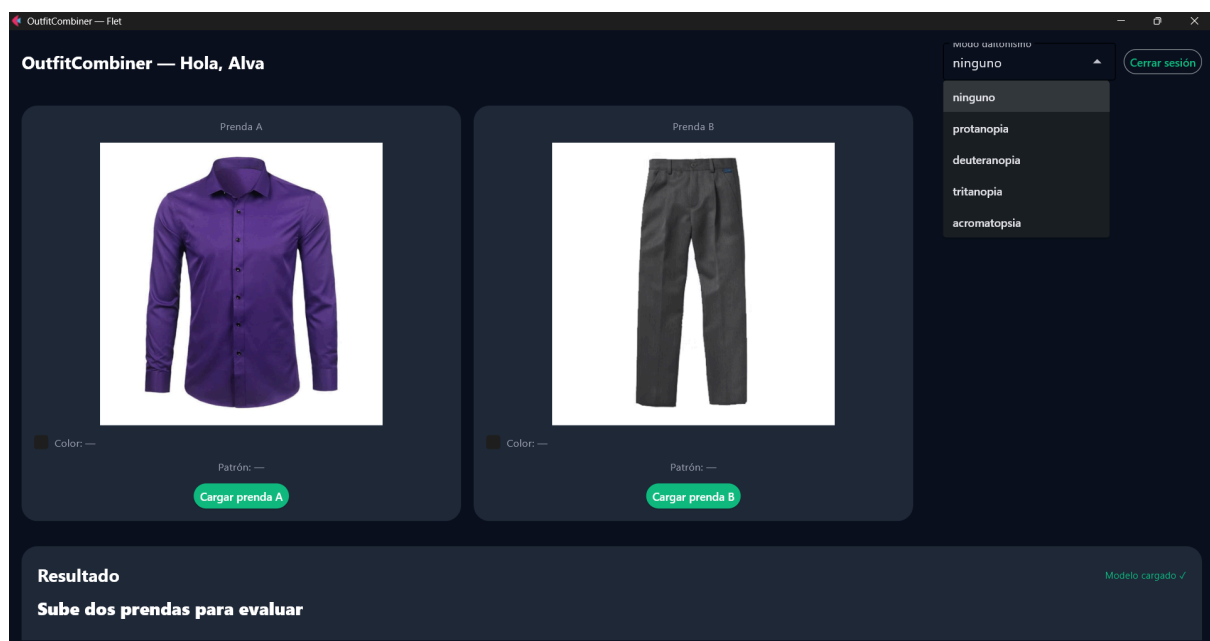
```
Resultado: NO COMBINAN ✖ | Confianza: 0.42
Por qué: Al menos una prenda es neutra (negro/blanco/gris/beige), lo que suele combinar con casi todo. Diferencia de tono ≈ 24°.
Cómo mejorarlo: La diferencia de brillo es muy baja; prueba un tono más claro/oscurο.
✓ Guardado en: /content/drive/MyDrive/OutfitCombiner/data/combina_dataset.csv
```

El sistema clasifica la combinación como **NO COMBINAN** (confianza 0.42) porque, aunque al menos una prenda es **neutra** —lo que en general facilita combinar—, la **diferencia de tono** entre ambas es **pequeña** ($\approx 24^\circ$, colores análogos) y, sobre todo, la **diferencia de brillo/luminancia** es **muy baja**; ese doble parecido en tono y brillo genera un conjunto **plano**, sin separación visual suficiente. En otras palabras, falta **contraste**: los dos elementos quedan demasiado cercanos en “color” y “claridad”, por lo que el modelo penaliza a la pareja.

Cambios y mejoras recientes

Selector de “Modo de daltonismo” (vista de usuario).

Se añadió un *dropdown* en la pantalla de usuario con las opciones **ninguno**, **protanopia**, **deuteranopia**, **tritanopia** y **acromatopsia**; este dato **no modifica** la lógica del modelo ni el pipeline de features, pero **se usa para personalizar las recomendaciones** y **se registra en el CSV** como **cvd_mode** para trazabilidad y futuros análisis. El flujo es transparente: el usuario elige su modo una vez, se guarda junto a cada predicción y luego el motor de sugerencias del catálogo lo pondera para priorizar ítems acordes a su condición.



Recomendaciones del catálogo integradas al resultado (si “No combinan”). Vista Administrador

Cuando el veredicto es **NO COMBINAN**, ahora se muestra un bloque de “**Sugerencias del catálogo**” con hasta **3 ítems** ordenados por un **score** que combina cuatro señales objetivas: diferencia de tono (premia **casi complementarios**, ΔH alrededor de 120°), **contraste de luminancia** (WCAG) para mejorar legibilidad visual, **bonificación por neutros** (blanco/negro/gris/beige) porque estabilizan la combinación, y **bonificación por CVD** si el ítem fue marcado en la base como recomendado para el modo de daltonismo elegido. En términos simples: el sistema propone alternativas que elevan contraste y armonía sin tocar la predicción base.

The screenshot shows the 'Vista Admin' interface. At the top right is a 'Cerrar sesión' button. On the left is a sidebar with 'Catálogo' and 'Actividad' links. The main area has a form with fields for 'Nombre prenda', 'Color HEX (#10b981)', a checkbox for 'Es neutro (blanco/negro/gris/beige)', a dropdown for 'Recomendado para' (set to 'ninguno'), and a 'Descripción' field. Below the form is a green button 'Agregar al catálogo'. Underneath is a table with 7 columns: ID, Nombre, Color, Neutro, Recomendado, Descripción, and Acciones. The table lists 7 items with their respective colors, neutrality status, recommendations, and descriptions.

ID	Nombre	Color	Neutro	Recomendado	Descripción	Acciones
19	Marrón chocolate	#5D4037	No	ninguno	Base cálida; combina con crema/beige y mostaza.	
18	Azul petróleo	#0B1328	No	acromatopsia	Muy baja luminancia; gran contraste con blancos/amarillos.	
17	Amarillo brillante	#FFD600	No	acromatopsia	Muy alta luminancia; contrasta con azules/negros.	
16	Rosa intenso	#EC407A	No	tritanopia	Acento vivo; combina con gris/negro.	
15	Verde esmeralda	#2E7D32	No	tritanopia	Verde oscuro; buen contraste con crema/blanco.	
14	Rojo ladrillo	#C62828	No	tritanopia	Rojo profundo; contrasta con gris/beige.	
13	Morado medio	#7E57C2	No	deuteranopia	Morado legible junto a grises y blancos.	

“puntos” tengan **tamaño razonable** (entre **2% y 12%** del lado menor) y aparezcan con **densidad suficiente** y en **texturas globales bajas** (si hay demasiados bordes, se descarta la hipótesis de lunares). Además, se **controlan líneas** (H/V) y **anisotropía** para no confundir **rayas/pliegues** con puntos. El efecto práctico es que prendas lisas o con pequeñas arrugas ya **no se clasifican erróneamente** como “lunares”.

Bloque de resultado con sugerencias y “Modelo cargado ✓”.

Se mejoró la **experiencia de usuario**: el panel de resultado ahora muestra, de forma compacta, el veredicto con confianza, la explicación breve y, si aplica, las **sugerencias del catálogo**; además, se mantiene un “**Modelo cargado ✓**” para indicar estado del clasificador. Todo se diseñó para **no obstaculizar** el flujo (cargar prendas → evaluar → actuar) y para que el usuario tenga **acciones inmediatas** cuando la respuesta es negativa. Todo esto, si al usuario le sale el resultado de que sus prendas no combinan, habrá un bloque de sugerencias para el, para indicarle según su condición con que lo puede acompañar.

Qué es y de dónde salen los datos

La vista **Actividad** muestra un historial de predicciones recientes tomadas del archivo [data/combina_dataset.csv](#).

Para no saturar la UI, se cargan **solo las últimas 300 filas** (`df.tail(300)`) y se muestran **en orden cronológico inverso** (la más reciente arriba). Cada fila en la tabla corresponde a **una evaluación** que hizo cualquier usuario desde la app.

OutfitCombiner — Flat

Vista Admin Cerrar sesión

Registros: 29 Aciertos: 86.2% Conf. media: 0.82

Prenda A	Color A	Patrón A	Prenda B	Color B	Patrón B	Resultado	Conf.	ΔHue°	Acciones
Camisa (camisaBlanca.jpg)	gris (#B9888E)	liso	Camisa (PANTALONNN.jpg)	gris (#DFDD09)	lunares	✓	0.90	150°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	estampado	✓	0.89	104°	Eliminar
Prenda A (camisaBlanca.jpg)	gris (#D7D6DC)	liso	Prenda B (pantalonGris.avif)	gris (#424244)	lunares	✓	0.67	10°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	estampado	✓	0.89	104°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	lunares	✓	0.89	104°	Eliminar
Camisa (camisaBlanca.jpg)	gris (#B9888E)	liso	Pantalón (pantalon_verde.jpg)	verde (#254315)	lunares	✓	0.97	150°	Eliminar
Prenda A (camisa_morada.jpg)	índigo (#422A69)	liso	Prenda B (pantalonGris.avif)	gris (#424244)	lunares	✗	0.37	22°	Eliminar
Camisa (camisa_morada.jpg)	índigo (#422A69)	liso	Camisa (PANTALONNN.jpg)	gris (#DFDD09)	lunares	✓	0.88	138°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	lunares	✓	0.89	104°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	lunares	Vestido (imagenLunares.jpg)	negro (#1C191B)	liso	✓	0.89	104°	Eliminar

Explicación de las columnas:

1. Fecha

- **Qué muestra:** fecha y hora de la predicción.
- **Origen:** columna `timestamp` (segundos UNIX) del CSV.
- **Tipo:** texto.

2. Usuario

- **Qué muestra:** nombre de usuario que ejecutó la evaluación.
- **Origen:** `user_name`. Si no hay valor, se muestra “-”.
- **Tipo:** texto.

3. Prenda A

- **Qué muestra:** etiqueta de la prenda detectada + nombre de archivo.
- **Origen:** `detA_label` (p. ej., “Camisa”, “Pantalón”...) y `imgA_path` (se muestra `basename`).
- **Fallbacks:** si no hay `detA_label` se usa “A”; si no hay ruta, se muestra “-”.
- **Tipo:** texto multilínea (etiqueta + archivo).

4. Color A

- **Qué muestra:** chip de color + texto “nombre (HEX)”.
- **Origen:** `a_color_hex` (para el chip) y `a_color_name` (para el nombre).
- **Cómo se obtiene:**
 - En cada evaluación se calcula el color principal de A (RGB) y se pasa a **HEX**.
 - El nombre básico (“gris”, “verde”, etc.) se infiere con una regla por **HSV** (ver “Reglas de cálculo”, abajo).
- **Tipo:** chip + texto.

5. Patrón A

- **Qué muestra:** una de `liso`, `lunares`, `cuadros`, `estampado`.
- **Origen:** `a_pattern` calculado al momento de la predicción con **heurística robusta** (ver “Reglas de cálculo”).
- **Tipo:** texto.

6. Prenda B

- **Igual que Prenda A**, pero con `detB_label` y `imgB_path`.
- **Notas:** cuando B no trae patrón guardado, la vista lo **recalcula al vuelo** si el archivo existe (`detect_pattern(read_bgr(imgB_path))`).



7. Color B

- **Igual que Color A**, pero con `b_color_hex` y `b_color_name`.

8. Patrón B

- **Igual que Patrón A**, pero para B. También se recalcula si faltaba.

9. Resultado

- **Qué muestra:**  si `pred == 1` (COMBINAN) o  si `pred == 0` (NO COMBINAN).
- **Origen:** `pred` predicho por el modelo.
- **Tipo:** icono.

10. Conf.

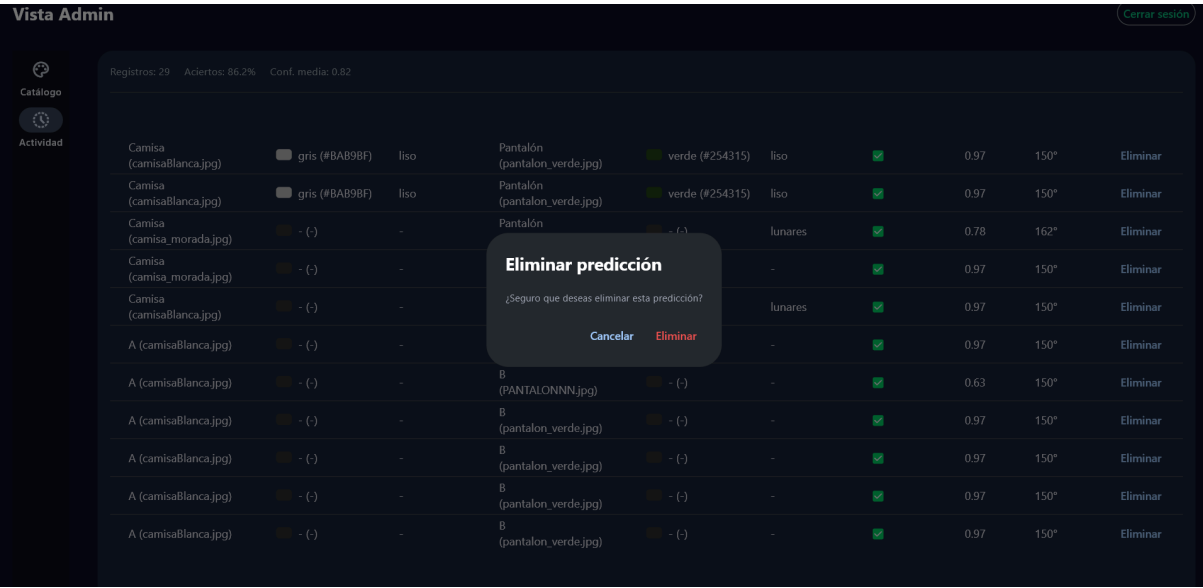
- **Qué muestra:** confianza del modelo en la clase positiva (0.00–1.00).
- **Origen:** `pred_proba`.
- **Cómo se calcula:** se intenta `model.predict_proba(X)[:,1]`; si no está disponible, se usa `decision_function` o `predict` como fallback.
- **Tipo:** número con 2 decimales.

11. ΔHue°

- **Qué muestra:** diferencia de tono (en grados) entre los **colores principales** de A y B.
- **Origen:** `hue_pair_note` (entero).
- **Cómo se calcula:**
 - Se obtiene el **tono H** (0–360) de los colores principales de A y B.
 - Se aplica distancia circular: $d = \min(|H1-H2|, 360-|H1-H2|) \rightarrow$ rango **0–180°**.
 - Se guarda como entero (ej. “104°”, “150°”).
- **Tipo:** número entero + símbolo “°”.

12. Acciones

- **Qué hace: Eliminar** quita esa fila del CSV.
- **Cómo funciona:** se abre confirmación; si se acepta, se carga el CSV completo, se hace `drop(index=csv_index)` y se guarda.
- **Notas:** el índice que se usa es el **índice original** del CSV (se preserva al invertir el orden), por eso el borrado es preciso.



Vista Admin Cerrar sesión

Registros: 29 Aciertos: 86.2% Conf. media: 0.82

Catálogo Actividad

Prenda A	Color A	Patrón A	Prenda B	Color B	Patrón B	Resultado	Conf.	ΔHue°	Acciones
Camisa (camisaBlanca.jpg)	gris (#B9B8BE)	liso	Camisa (PANTALONNN.jpg)	gris (#DFDD09)	lunares	✓	0.90	150°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	estampado	✓	0.89	104°	Eliminar
Prenda A (camisaBlanca.jpg)	gris (#D7D6DC)	liso	Prenda B (pantalónGris.avif)	gris (#424244)	lunares	✓	0.67	10°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	estampado	✓	0.89	104°	Eliminar
Camisa (camisaGris.webp)	gris (#646669)	estampado	Vestido (imagenLunares.jpg)	negro (#1C191B)	lunares	✓	0.89	104°	Eliminar
Camisa (camisaBlanca.jpg)	gris (#B9B8BE)	liso	Pantalón (pantalón_verde.jpg)	verde (#254315)	lunares	✓	0.97	150°	Eliminar
Prenda A (camisa_morada.jpg)	indigo (#422A69)	liso	Prenda B (pantalónGris.avif)	gris (#424244)	lunares	✗	0.37	22°	Eliminar
Camisa (camisa_morada.jpg)	indigo (#422A69)	liso	Camisa (PANTALONNN.jpg)	gris (#DFDD09)	lunares	✓	0.88	138°	Eliminar
Camisa (camisaBlanca.jpg)	gris (#B9B8BE)	liso	Vestido (imagenLunares.jpg)	negro (#1C191B)	estampado	✓	0.89	104°	Eliminar

Manual del Administrador

Introducción

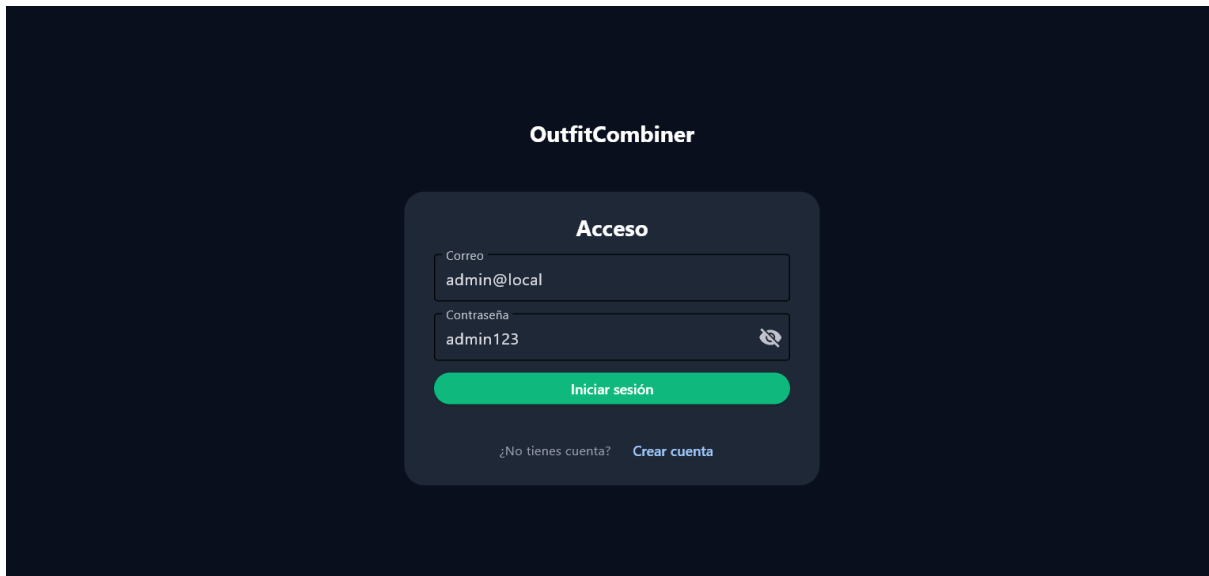
El **Administrador** tiene un rol **operativo y acotado** dentro de OutfitCombiner. Sus responsabilidades se limitan a:

- **Historial:** ver el registro de predicciones realizadas por los usuarios (sin editar resultados).
- **Catálogo:** **ingresar** (alta) y **eliminar** (baja) elementos del catálogo de colores/prendas, incluyendo su etiqueta de recomendación para cada tipo de daltonismo.

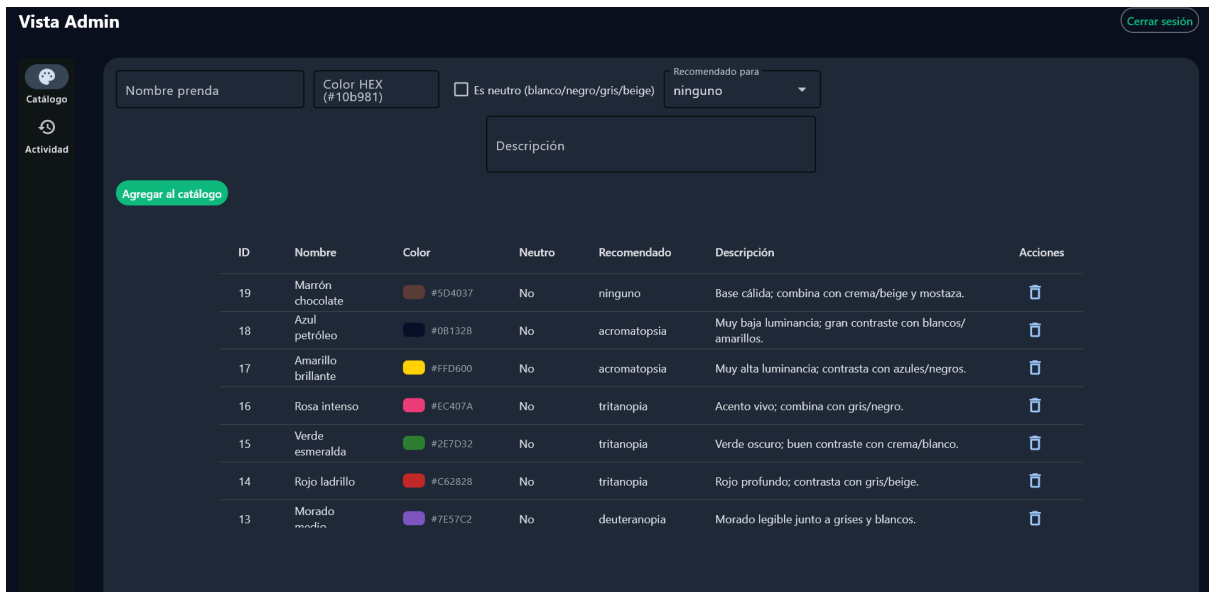
Instalación y Configuración

Acceso inicial

1. Abra la aplicación OutfitCombiner.
2. Inicie sesión con su **usuario y contraseña de administrador**.



3. Verifique en la barra lateral que aparecen las secciones **Catálogo** y **Actividad**.























Vista Admin

Cerrar sesión

Catálogo

Actividad

Registros: 55 Aciertos: 67.3% Conf. media: 0.70

Fecha	Usuario	Prenda A	Color A	Patrón A	Prenda B	Color B	Patrón B	Resultado
2025-09-04 00:11	Alva	Camisa (camisaMoradDD.jpg)	 indigo (#422A69)	liso	Prenda B (pantalónGrisDD.avif)	 gris (#424244)	liso	✗
2025-09-04 00:06	Alva	Camisa (camisaAzul.png)	 azul (#1E2943)	liso	Prenda B (pantalónGrisDD.avif)	 gris (#424244)	liso	✗
2025-09-04 00:05	Alva	Camisa (camisaAzul.png)	 azul (#1E2943)	liso	Pantalón (pantalón_verde.jpg)	 verde (#254315)	lunares	✓
2025-09-04 00:03	Alva	Camisa (camisaAzul.png)	 azul (#1E2943)	liso	Prenda B (pantalónGrisDD.avif)	 gris (#424244)	liso	✗
2025-09-04 00:02	Alva	Camisa (camisaGris.webp)	 gris (#646669)	estampado	Prenda B (pantalónGrisDD.avif)	 gris (#424244)	liso	✗
2025-09-03 23:59	Alva	Camisa (camisa_morada.jpg)	 indigo (#422A69)	liso	Prenda B (pantalónGris.avif)	 gris (#424244)	liso	✗
2025-09-03 23:56	fer	Pantalón (camisaRojoOscuro.jpg)	 rojo (#3F0D19)	liso	Camisa (pantalónMarron.jpg)	 naranja (#E241C)	liso	✓
2025-09-03 23:55	fer	Camisa (camisaRoja.jpg)	 rojo (#B70708)	liso	Pantalón (pantalón_verde.jpg)	 verde (#254315)	lunares	✓
2025-09-03 23:52	fer	Camisa (camisaGris.webp)	 gris (#646669)	estampado	Prenda B (pantalónGris.avif)	 gris (#424244)	liso	✗
2025-09-03 23:52	fer	Camisa (camisaAzul.png)	 azul (#2F61B1)	liso	Pantalón (pantalón_verde.jpg)	 azul (#243C63)	liso	✗

Configuración básica que sí hace el Admin

- **Catálogo**
 - Alta de elementos con: **Nombre**, **Color HEX** (formato **#RRGGBB**), **Es neutro** (blanco/negro/gris/beige), **Recomendado para** (**ninguno**, **protanopia**, **deuteranopia**, **tritanopia**, **acromatopsia**) y **Descripción** opcional.
 - Baja de elementos mediante el botón **Eliminar** en la tabla.
- **Verificación visual**
 - Tras agregar, valide que el chip de color y el HEX se muestren correctamente en la tabla.

Uso Básico (según alcance del Admin)

1) Ver Historial (Actividad)

- Abra **Actividad** en la barra lateral.
- En la parte superior verá estadísticas resumidas (cantidad de registros recientes, confianza media, etc.).

- En la tabla podrá **ver** por fila:
 - **Fecha y usuario**, nombres de prenda detectada (A/B)
 - **Color A/B** (nombre y **HEX**)
 - **Patrón A/B** (liso, lunares, cuadros, estampado)
 - **Resultado** (Combinan/No combinan), **Confianza** y ΔHue°
- **Nota:** el Admin **no edita** predicciones desde aquí (solo consulta).

2) Administrar Catálogo

- Abra **Catálogo** en la barra lateral.
- Complete los campos y pulse **Agregar al catálogo**:
 - **Nombre prenda:** ej. “Camisa azul acero”, “Pantalón caqui”.
 - **Color HEX:** ej. **#1E3A8A**. Debe incluir **#** y 6 dígitos hexadecimales.
 - **Es neutro:** marque si es blanco/negro/gris/beige.
 - **Recomendado para:** seleccione **ninguno** o el tipo de CVD aplicable.
 - **Descripción** (opcional): ej. “Aporta contraste con azules medios”.
- Para **eliminar** un elemento, use el icono **Eliminar** en la fila correspondiente.

Impacto en la app del usuario: cuando una pareja sale **NO COMBINAN**, la UI del usuario toma el **color A** como base y sugiere alternativas del **Catálogo** ordenadas por complementariedad (ΔH), contraste, neutralidad y coincidencia con el modo CVD. Si el catálogo está vacío o mal cargado, **no habrá sugerencias**.

Monitorización y Mantenimiento (dentro del rol)

Qué revisar periódicamente

- **Actividad:** que el historial se esté llenando (indica uso normal).

- **Catálogo:** que existan suficientes elementos para cada CVD y **neutros** (blanco/negro/gris/beige), pues suelen desbloquear opciones seguras.

Buenas prácticas

- **Calidad de datos del catálogo:**
 - Use **HEX válidos** (**#RRGGBB**).
 - Marque **neutro** solo cuando aplique (blanco/negro/gris/beige).
 - Rellene **Recomendado para** cuando el color sea claramente útil a cierto CVD.
- **Cobertura mínima recomendada:**
 - Neutros: **#FFFFFF**, **#F5F5DC** (beige), **#D9D9D9** (gris claro), **#000000**.
 - Por CVD: agregue 4–6 colores útiles por tipo (ej. azules/cianes para prot/deut, amarillos/olivas para trit, tonos claros/medios para acromatopsia).

Mensajes comunes y cómo proceder

- “No hay elementos en el catálogo o no se encontraron sugerencias.”

Funciones Avanzadas del Admin (dentro del alcance)

1. Curaduría por CVD

Mantenga un set de colores “seguros” por tipo de daltonismo. Asigne “**Recomendado para**” para influir en la priorización de sugerencias cuando al usuario le salga “NO COMBINAN”.

2. Política de neutros

Los **neutros** elevan la probabilidad de buenas combinaciones. Asegúrese de que el catálogo incluya varios neutros con ligeras variaciones de brillo (blanco roto, gris medio, antracita, beige).

