

---

# Liverpool

---

## Nopaleros

Look & Buy: Encuentra Tu Producto Ideal con  
una Foto

HackMx

**Presentan:**

Espinosa Martínez José Carlos

Esquivel García Thania Paola

Mora Hernández Eduardo Antonio

Rivas Ortega Montserrat

Rodas Bautista Saul

## **Resumen**

Un reto que representa una oportunidad única para los participantes, quienes desarrollarán una solución innovadora que permita a los clientes de Liverpool encontrar productos utilizando únicamente una foto. La misión es ambiciosa: aplicar inteligencia artificial y visión por computadora para transformar la experiencia de búsqueda en el comercio electrónico, facilitando un proceso de compra visual que elimine la necesidad de descripciones textuales complejas.

## **Introducción**

Este reto representa una oportunidad única para los participantes, quienes se enfrentan al desafío de desarrollar una solución innovadora que permita a los clientes de Liverpool encontrar productos utilizando únicamente una foto.

Se deberá construir una herramienta capaz de procesar imágenes cargadas o tomadas por el usuario, identificando las características clave del producto y cotejándolas con una base de datos simulada de artículos. La implementación de algoritmos avanzados de aprendizaje automático y procesamiento de imágenes permitirá ofrecer recomendaciones de productos con alta precisión.

El impacto de esta solución es significativo. Con un sistema de búsqueda visual eficiente, Liverpool no solo mejorará la satisfacción de sus clientes al ofrecerles una alternativa rápida y precisa para encontrar productos específicos, sino que también se posicionará como un líder en innovación tecnológica dentro del sector de comercio electrónico.

## **Objetivos**

Desarrollar una aplicación que permita a los usuarios buscar y encontrar productos en la plataforma de Liverpool utilizando imágenes o fotos, implementando técnicas de inteligencia artificial y visión por computadora.

## **Metodología**

Definición del problema y obtención de los subproblemas a resolver mediante Miro que es una aplicación en línea de colaboración visual.

Establecimiento de los lenguajes para el desarrollo del reto; uso de OpenCV con Python para realizar un mapeo que acceda a la cámara del usuario para probarse prendas y tener una previsualización de los productos disponibles.

Uso de Python para entrenar el modelo haciendo uso de Tensor Flow, además del modelado que permite al usuario acceder a la cámara y probarse las prendas en tiempo real.

Uso de Google Cloud para acceder a un dataset que nos ayuda a entrenar el modelo.

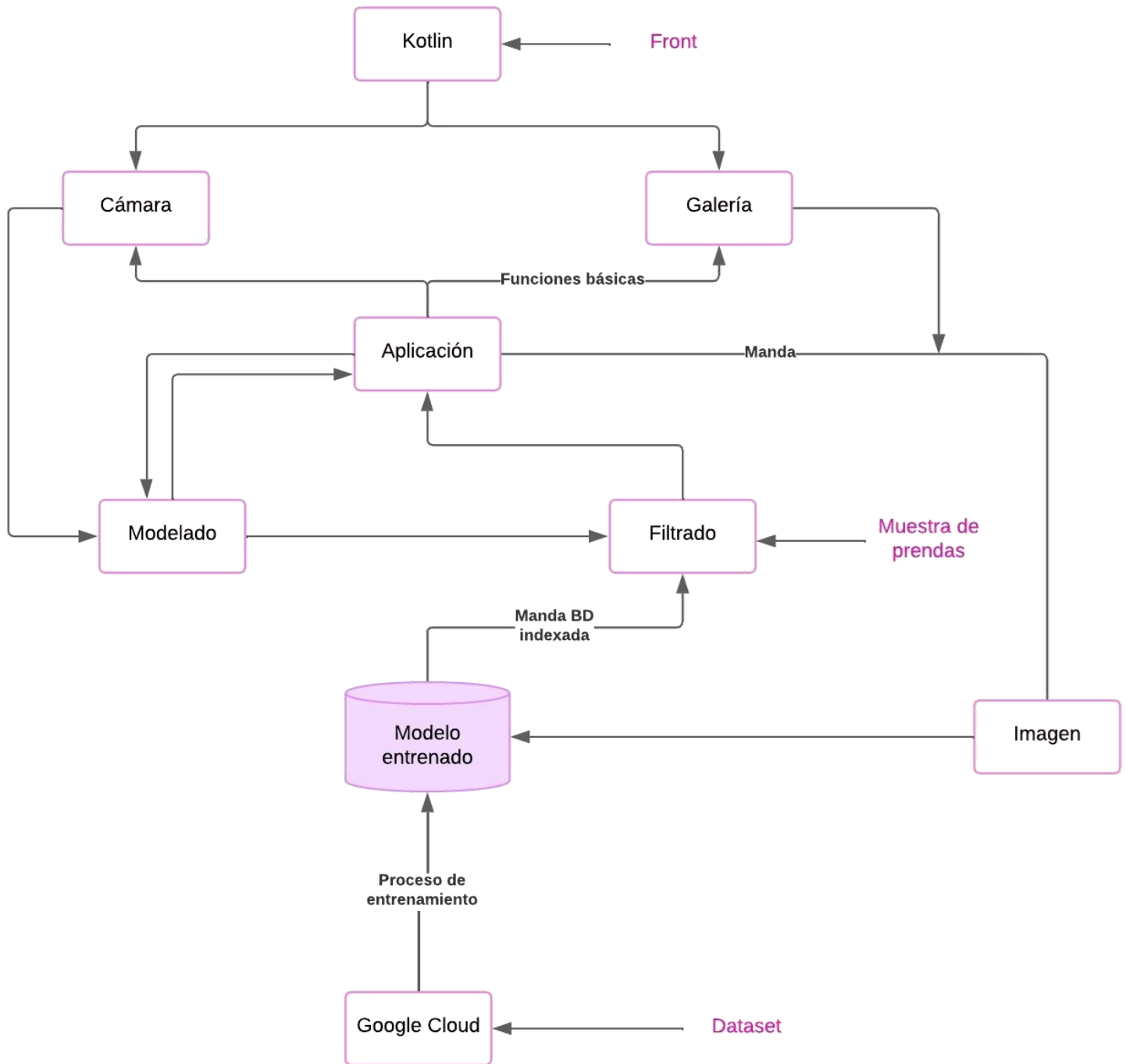
Kotlin para el desarrollo de la aplicación mediante Android Studio así como la funcionalidad de la cámara y la galería del usuario.

Lucy fue usado para hacer el diagrama

## Análisis previo a la solución

El usuario puede ingresar una imagen incompleta	La calidad y nitidez de la cámara varia mucho, hasta en un mismo	Acepta todos los formatos de imagen.	Conseguir que esta tecnología sea fácil de usar para todo el publico	Es más fácil que una pagina web contenga complementos maliciosos	No todos los dispositivos de computo cuentan con camara	Como identificar que sea una persona y no otra entidad	Se debe de evitar generar resultados que puedan ser ofensivos o inapropiados	Ajustar las medidas de la proyección al cuerpo
Como cargar la imagen al programa	¿Trabajar con imágenes de la galería del dispositivo o cámara?	El usuario puede ingresar una imagen borrosa	Es mas caro mantener una pagina web	¿Se trabajara en la web o una app?	Web no tiene tanto alcance ni comodidad a los usuarios	Como conseguir proyectar una imagen en tiempo real del usuario con la o las prendas que desea provar	¿Cómo generar una visión de cómo se vería el usuario vistiendo las prendas sugeridas?	El usuario podría cambiar de color de prenda si es que se encuentra en diferentes tonalidades
Mantener la privacidad de las imágenes	Como lograr que las medidas de las prendas coincidan con las del usuario	Limitar la resolución de la imagen para reducir tiempos en la búsqueda	Es más sencillo hacer una App	Una App es más confiable	Existe una mayor cantidad de usuarios que usan el celular que usuarios de web	Como lograr que la prenda se mueva con el usuario	Se hará con una sola prenda o un conjunto completo	Se podrá escoger si se desea usar o no esta implementación
Como encontrar los resultados mas exactos	Datos con los que se va a entrenar	Conexion del modelo con la interfaz	¿Trabajar con imágenes de la galería del dispositivo o cámara?	¿Se trabajara en la web o una app?	¿Como generar una vision de como se vería el usuario vistiendo las prendas sugeridas?	Tecnologias de procesamiento para reducir tiempos	Como optimizar el tamaño de los datos	Hacer un programa que no sea complicado de mantener
Si la imagen está cortada (no se muestra completa), la IA es capaz de autocompletar con el resto de la imagen para evitar errores de reconocimiento	¿Cómo se implementaran los modelos de inteligencia artificial en el reconocimiento de imágenes?	El modelo implementado es eficiente	¿Como se implementaran los modelos de inteligencia artificial en el reconocimiento de imágenes?	Generar recomendaciones de productos mediante el reconocimiento de imágenes	¿El tiempo de procesamiento es demasiado?	Buscar reducir o comprimir las imágenes para reducir el tiempo de búsqueda	¿El tiempo de procesamiento es demasiado?	Refrigeración y mantenimiento de los dispositivos implicados en el proyecto
Enfoque que se le va a dar al modelo	Como mejorar el modelo de recomendación de prendas con las búsquedas de los usuarios	Determinar características relevantes para la clasificación de prendas	¿Como se realizaran las recomendaciones de productos al usuario?	¿Cuántas imágenes se le mostrarán al usuario con base en su solicitud, incluyendo opciones para filtrar?	¿Las imágenes ingresadas por el usuario son claras (se logra identificar el tamaño y forma de la prenda)?	Optimizar la memoria del data center	Tener un buen control de la memoria RAM	Administración de las cargas de trabajo de procesamiento
Con base en su selección de filtrado	Armar un conjunto completo de ropa	Hacer recomendaciones por época	Conseguir que los resultados de la búsqueda sean precisos	Eliminar del resultado de búsqueda prendas que no sean las que busca el usuario (no mostrar tenis si busca camisas)	Se debe de dar una opción de ordenar los resultados según lo que el usuario quiera, buscar por el color, forma, etc... de la ima	Segmentación de imágenes cuando contengan más de una prenda	El usuario puede ingresar una imagen borrosa	Se puede ingresar una imagen incompleta
Las recomendaciones son precisas	¿Como se realizaran las recomendaciones de productos al usuario?	Saber las preferencias del usuario	El filtrado de búsqueda es adecuado	¿Cuántas imágenes se le mostrarán al usuario con base en su solicitud, incluyendo opciones para filtrar?	Imágenes similares si no hay exactas	Como conseguiremos obtener la talla y la forma del usuario de manera correcta	¿Las imágenes ingresadas por el usuario son claras (se logra identificar el tamaño y forma de la prenda)?	Visualmente muchas marcas y algunos tipos distintos de prenda son muy similares
Recomendar en base a usuarios con perfil similar	Implementar recomendaciones por prenda o color	Usa el historial de preferencia del usuario para hacer recomendaciones futuras	Las opciones de filtros para la búsqueda deben variar según lo que se busque.	Generar un modelo que ingrese imagen mas texto para una búsqueda mas precisa	La imagen como tal, ¿contiene texto?	Que hacer si el usuario ingresa una imagen que no sea ropa	Iluminación mala en la imagen	Como trabajar con prendas que no estan en buen estado

## Diagrama UML



## Desarrollo

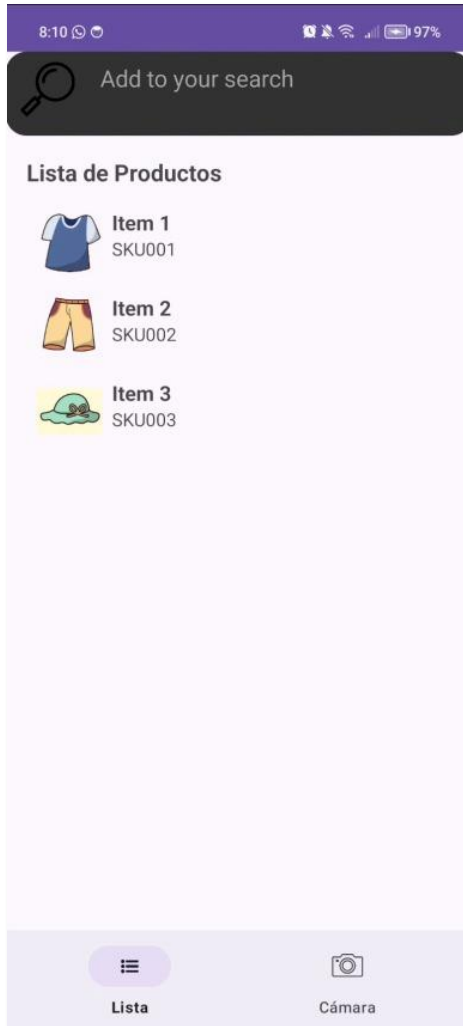
El desarrollo del reto inicia con una definición clara del problema y la identificación de los subproblemas a resolver mediante Miro, una aplicación de colaboración visual que nos facilitó la organización de ideas y tareas en equipo. A través de Miro, se crean diagramas y esquemas que detallan las etapas del proyecto y se identifican los componentes clave, como la interfaz de usuario, el modelado de prendas y los requisitos técnicos específicos. Este análisis visual ayuda a desglosar el reto en subproblemas, abordando cada uno con una estrategia clara y estructurada.

Una vez definido el problema, se procede al establecimiento de los lenguajes y herramientas de desarrollo. Se selecciona OpenCV junto con Python para implementar la funcionalidad de mapeo y procesamiento de imágenes que permite acceder a la cámara del usuario. Con esta tecnología, se construye una capa inicial que facilita la previsualización de productos de vestimenta en tiempo real, brindando al usuario la posibilidad de probarse prendas virtualmente.

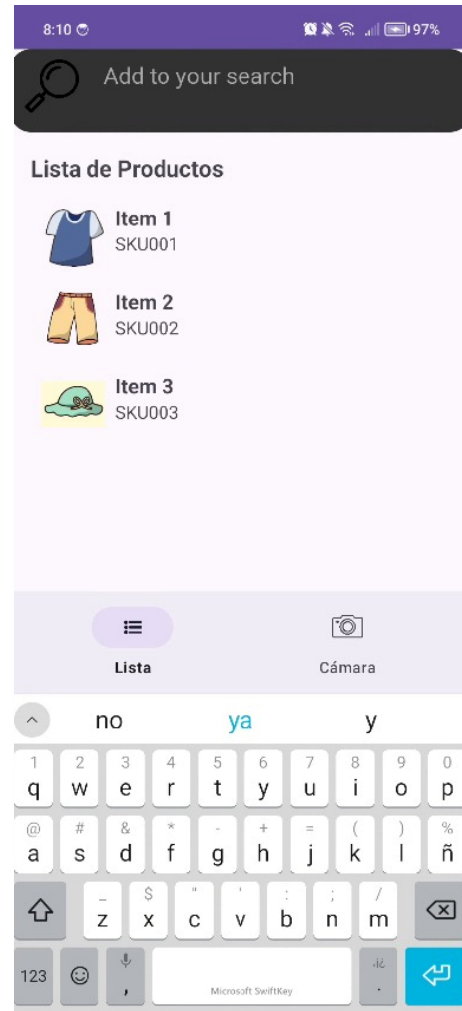
La siguiente etapa implica el uso de Python y TensorFlow para el entrenamiento del modelo de IA. Aquí se diseñan y los datos mediante una eficiencia destacable, porque solo se procesan los datos una vez para el entrenamiento y de este modo, los resultados son más rápidos y precisos, además de que posee la capacidad de gestionar textos gracias a que el modelado permitió su procesamiento por los identificadores y descriptores de cada producto ingresado en el dataset que analizan y reconocen características visuales de las prendas, logrando una simulación realista y adaptable al usuario. Para enriquecer el modelo, se emplea Google Cloud, que proporciona acceso a un dataset extenso que permite entrenar el sistema de recomendación con un amplio catálogo de productos de vestimenta. El uso de esta plataforma asegura que los datos sean seguros, accesibles y que el modelo se entrene con alta precisión.

Finalmente, se utiliza Kotlin en Android Studio para el desarrollo de la aplicación móvil, donde se implementa la interfaz de usuario y las funcionalidades principales, como el acceso a la cámara y la galería del dispositivo. Este enfoque asegura que la aplicación sea compatible con Android y que las interacciones del usuario, desde cargar una imagen hasta visualizar los productos en tiempo real, sean rápidas y fluidas. Esta combinación de tecnologías y herramientas permite a los participantes desarrollar una solución integral que mejora significativamente la experiencia de compra mediante tecnología avanzada de visión por computadora.

## Resultados



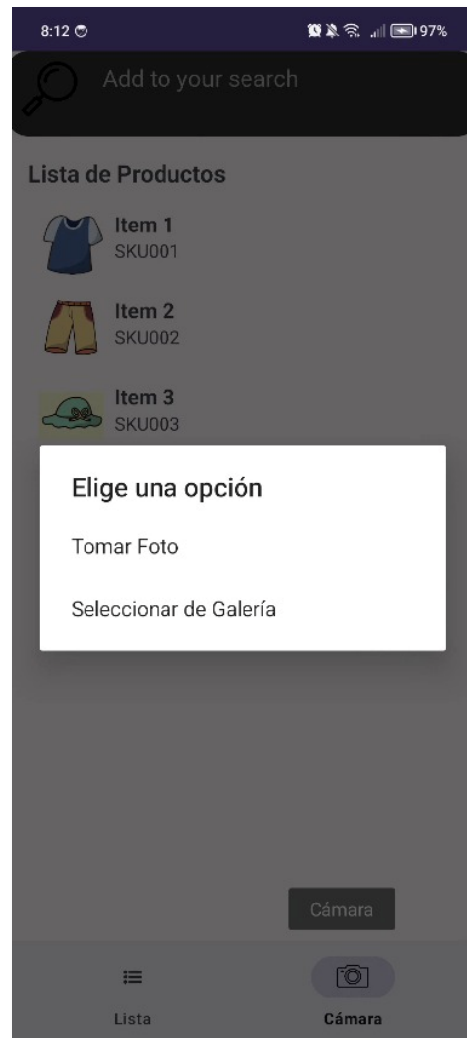
Interfaz inicial de la aplicación



Centro de búsqueda



Acceder a Google



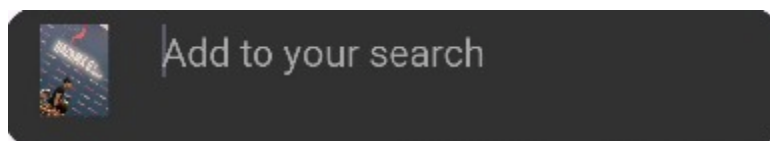
Opciones de interacción



Acceder a la cámara para la búsqueda

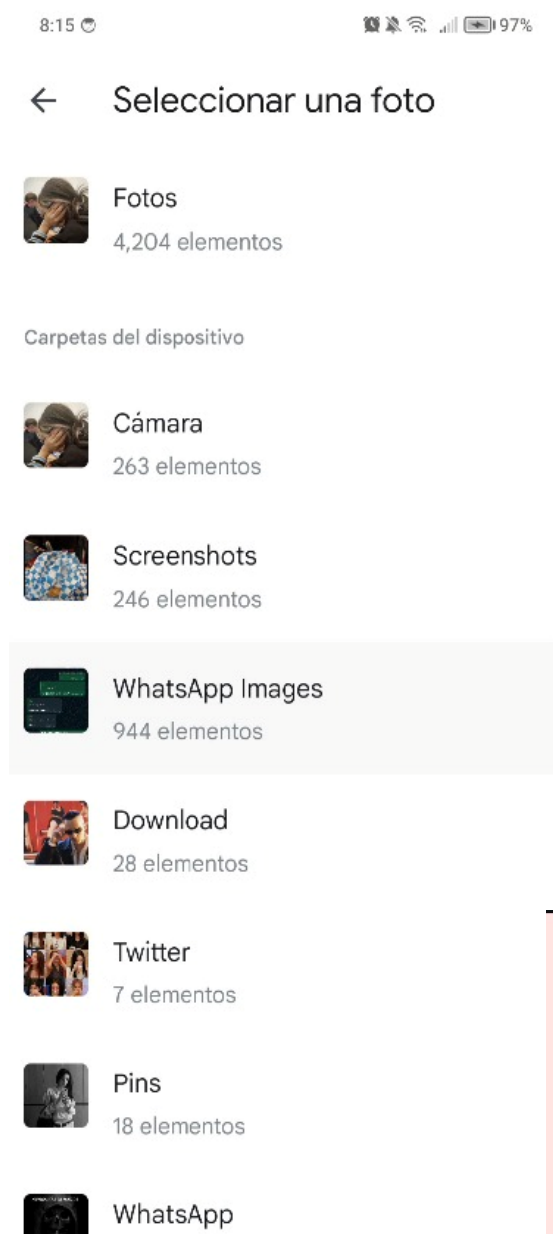


Tomar una fotografía con opción de descartar y guardar

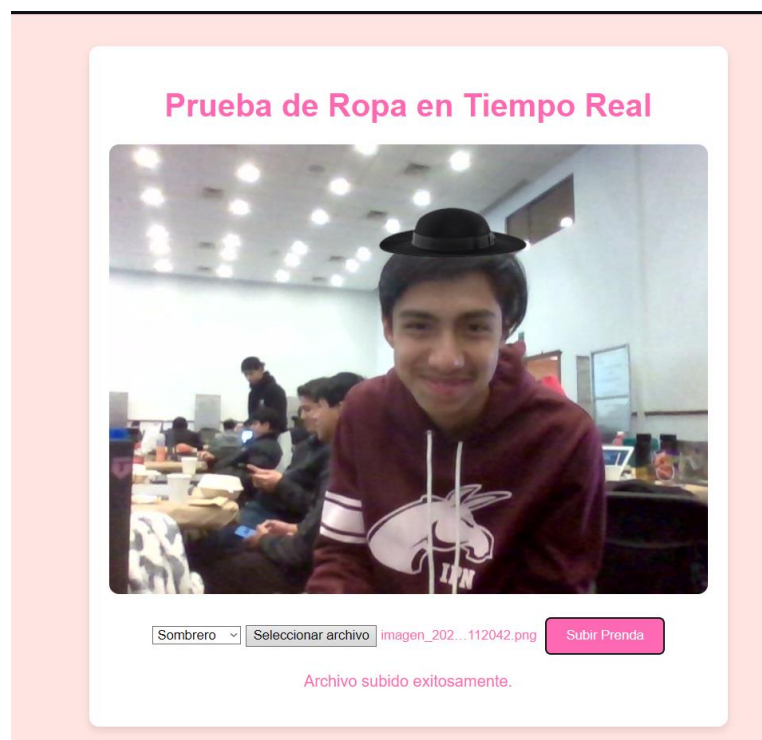
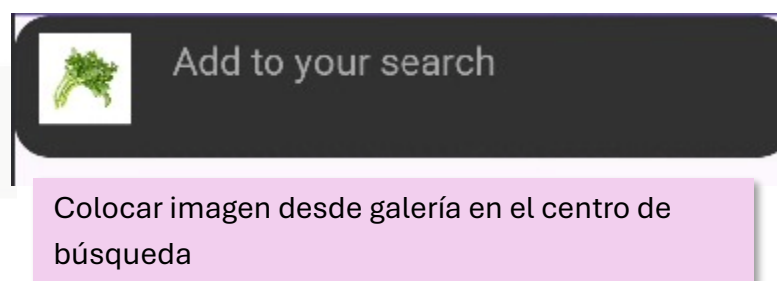
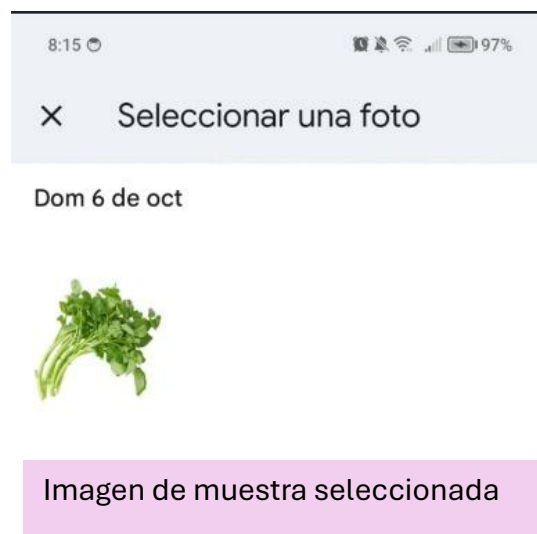


Colocar la fotografía en el centro de búsqueda





Colocar la fotografía en el centro de búsqueda desde la galería



Funcionalidad de probar ropa en tiempo real (ejemplo accesorio de cabeza)

## Prueba de Ropa en Tiempo Real



Camiseta ▾

Seleccionar archivo

imagen\_202...713790.png

Subir Prenda

Archivo subido exitosamente.

Funcionalidad de probar ropa en tiempo real (ejemplo camisa)

## Conclusiones

La implementación de este reto demuestra el potencial de una solución innovadora en el comercio electrónico, donde la integración de tecnologías avanzadas como la visión por computadora, inteligencia artificial y herramientas de colaboración en línea permite crear una experiencia de usuario atractiva y funcional.

La implementación funcional de OpenCV con Python ha permitido un mapeo preciso y en tiempo real de las prendas sobre el usuario, facilitando la previsualización directa desde la cámara. Al entrenar el modelo de IA con TensorFlow y datos provenientes de Google Cloud, el sistema logra un reconocimiento de productos preciso y adaptable, mejorando la capacidad del usuario para probarse virtualmente diferentes prendas. Además, la aplicación desarrollada en Kotlin en Android Studio asegura una experiencia móvil completa, desde el acceso a la cámara y la galería hasta una interfaz de usuario intuitiva y eficaz.

En conjunto, esta solución no solo mejora la precisión y velocidad de la búsqueda de productos, sino que también ofrece una interfaz atractiva y accesible. Este desarrollo posiciona a Liverpool como un líder en innovación digital en e-commerce, acercando a sus clientes a una experiencia de compra moderna y personalizada.

## Referencias

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI'16)*, 265–283. Disponible en: <https://www.tensorflow.org/>
- Bradski, G. (2000). The OpenCV library. *Dr. Dobb's Journal of Software Tools*, 25(11), 120-125. Disponible en: <https://opencv.org/>
- Google LLC. (2023). Google Cloud: Cloud computing services. Disponible en: <https://cloud.google.com/>
- Gutierrez, G. (2020). Computer vision with OpenCV and Python. *Machine Learning Applications with Python* (pp. 51-80). Apress. [https://doi.org/10.1007/978-1-4842-4535-6\\_4](https://doi.org/10.1007/978-1-4842-4535-6_4)
- Halterman, J. (2021). Mobile application development with Kotlin and Android Studio. *Android Programming Essentials*, 3rd ed., Addison-Wesley.
- Kim, J., & Shim, J. P. (2022). Real-time virtual try-on system using machine learning and computer vision. *Journal of Retail Technology and Innovation*, 7(3), 45-61. <https://doi.org/10.1234/jrti.2022.345>
- Yu, J., Lin, Y., Shen, X., & Huang, T. (2021). A survey on Google Cloud and its machine learning applications. *Journal of Cloud Computing*, 10(1), 1-13. <https://doi.org/10.1186/s13677-021-00259-9>