Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

# Introducción

## Problemática

Actualmente, el diseño de interfaces hombre-máquina (HMI) demanda soluciones más inclusivas, intuitivas y adaptables, especialmente en contextos donde el uso de dispositivos físicos tradicionales resulta limitado, como en personas con movilidad reducida o en entornos sin teclado disponible, este proyecto plantea una alternativa innovadora: un teclado virtual que puede ser controlado únicamente con el movimiento del dedo, sin contacto directo con la pantalla.

Utilizando tecnologías de visión artificial, como la detección de manos y el reconocimiento de gestos, se logra una interacción fluida y accesible, se implementa un sistema de predicción de palabras mediante modelos de lenguaje basados en trigramas, lo que contribuye a una escritura más rápida y eficiente, mejorando notablemente la usabilidad del sistema.

## Justificación

El desarrollo de interfaces de entrada alternativas se ha vuelto esencial en un mundo donde la accesibilidad, la interacción sin contacto y la personalización de sistemas cobran cada vez más relevancia. Este proyecto responde a la necesidad de ofrecer una solución innovadora que permita la escritura sin necesidad de dispositivos físicos tradicionales como teclados o pantallas táctiles.

En particular, el teclado virtual controlado por gestos de la mano se justifica por su potencial para ser utilizado en diversas situaciones: desde entornos médicos o industriales donde no se pueden utilizar interfaces físicas, hasta casos de personas con movilidad reducida que necesitan métodos de interacción más inclusivos y accesibles.

Además, la incorporación de funciones como el autocompletado de palabras mediante modelos de lenguaje mejora significativamente la velocidad de escritura y reduce el esfuerzo cognitivo del usuario. Esto lo convierte en una herramienta práctica tanto para uso cotidiano como para la investigación en tecnologías de asistencia.

El uso de tecnologías ligeras como MediaPipe para la detección de gestos y modelos de predicción de texto basados en trigramas garantiza un buen rendimiento sin necesidad de grandes recursos computacionales, lo que lo hace viable incluso en equipos con hardware limitado.

En conjunto, este proyecto no solo explora nuevas formas de interacción hombre-máquina, sino que también promueve la inclusión digital, la eficiencia y la innovación tecnológica a través de una solución práctica, económica y adaptable.

# Arquitectura

## Diagrama del sistemaDiagrama El contenido generado por IA puede ser incorrecto.

## Tecnologías implementadas:

* OpenCV: Captura de video, detección de interacción con teclas y dibujo del teclado.
* MediaPipe (Hands): Detección de landmarks de la mano y gestos como la palma abierta.
* Pygame: Sonido de retroalimentación al presionar una tecla.
* Pickle: Carga del modelo de lenguaje basado en trigramas para autocompletado.
* Python: Lenguaje de programación principal.

# Desarrollo y Prototipo

La construcción del prototipo tuvo como objetivo principal crear una interfaz operativa que permitiera a una persona con capacidades motrices limitadas redactar texto mediante el uso exclusivo de una mano frente a una cámara. A diferencia de la propuesta inicial, que consideraba el seguimiento ocular y la detección de parpadeos como método de entrada, el diseño fue reorientado hacia una solución basada en visión artificial enfocada en el reconocimiento de manos. Para ello, se utilizó una cámara estándar junto con herramientas especializadas en inteligencia artificial, lo que facilitó el procesamiento de los gestos necesarios para controlar el teclado virtual.

## Estructura del Prototipo

1. Módulo de detección de la mano y del dedo índice  
A través del uso de la biblioteca *MediaPipe Hands*, se identifica la presencia de una mano dentro del campo visual de la cámara. Mediante los puntos de referencia que proporciona esta herramienta, se localiza específicamente la posición del dedo índice (punto 8), el cual actúa como puntero para interactuar con los elementos de la pantalla. Este método facilita una forma de interacción intuitiva y sin contacto físico.

2. Interfaz visual del teclado en pantalla  
Se implementó un teclado virtual dividido en filas que contienen letras, números del 0 al 3 y teclas funcionales como "SPACE" y "DEL". Cada tecla se representa como una región rectangular, la cual es dibujada y actualizada dinámicamente. Si el dedo índice permanece sobre una tecla durante más de un segundo, el sistema lo interpreta como una selección o “clic virtual”. Al activar una tecla:

* Se emite un sonido de confirmación.
* Se incorpora el carácter seleccionado al texto que se está escribiendo.
* Si se presiona "SPACE", se agrega un espacio.
* Si se selecciona "DEL", se elimina el último carácter ingresado.

3. Sistema de predicción de palabras  
Mientras el usuario escribe, el sistema sugiere posibles palabras utilizando un modelo de lenguaje basado en trigramas. Estas sugerencias se muestran en pantalla como opciones seleccionables mediante el mismo mecanismo de detección manual. Este componente optimiza la velocidad de escritura, ya que reduce la cantidad de letras necesarias para completar palabras, resultando especialmente útil para usuarios con movilidad reducida.

## Interacción sin contacto y retroalimentación

Una de las cualidades destacadas del prototipo es su capacidad para operar sin necesidad de contacto físico ni el uso de dispositivos especializados. El mecanismo de selección funciona detectando el tiempo que el dedo índice permanece sobre una tecla, lo cual permite una interacción sencilla y accesible. Para complementar esta funcionalidad, se integró un sistema de retroalimentación sonora utilizando la biblioteca *pygame*, que emite un sonido cada vez que se registra una pulsación. Esta señal auditiva proporciona al usuario una confirmación inmediata de sus acciones, lo que resulta especialmente beneficioso para personas con alguna limitación visual.

## Personalización y Adaptabilidad

Durante el proceso de diseño y desarrollo, se implementaron una serie de modificaciones visuales en el teclado con el objetivo de optimizar su legibilidad y garantizar una correcta alineación de los componentes. Entre los cambios más significativos se incluyó el redimensionamiento de ciertas teclas, como la barra espaciadora ("SPACE"), que fue reducida para lograr un equilibrio estético y funcional.

Además, se revisó y ajustó meticulosamente el espaciado entre las columnas de teclas para asegurar una distribución más uniforme y armoniosa. Asimismo, se replanteó por completo la disposición de las filas, corrigiendo un problema previo en el que las teclas tendían a adoptar una forma escalonada o de "escalera", lo cual afectaba negativamente tanto a la estética como a la usabilidad.

Estos refinamientos son fundamentales en el diseño de interfaces accesibles, ya que requieren un equilibrio entre simplicidad, intuición y comodidad visual. Una disposición clara y coherente no solo facilita la navegación, sino que también mejora la experiencia del usuario, especialmente en aquellos casos donde la accesibilidad es una prioridad. Cada detalle, por pequeño que parezca, contribuye a crear un producto más inclusivo y eficiente.

# Implementación (Funciones Clave)

## Código completo

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

## Inicialización y Configuración

Texto

El contenido generado por IA puede ser incorrecto.

Al comenzar la ejecución del programa, se importan las bibliotecas esenciales para el funcionamiento del sistema:

* **cv2** se utiliza para capturar y procesar la imagen de video en tiempo real.
* **mediapipe** permite el reconocimiento de manos mediante modelos de visión artificial.
* **pygame** se encarga de reproducir los sonidos que acompañan la selección de teclas.
* **pickle** se emplea para cargar el modelo de predicción de texto basado en trigramas que fue entrenado con anterioridad.

## Detección de Mano e Índice

El sistema emplea MediaPipe Hands para la detección en tiempo real de la mano. Posteriormente, se obtienen las coordenadas del dedo índice utilizando la siguiente función:Texto

El contenido generado por IA puede ser incorrecto.

## Detección gesto de mano abierta

La función detectar\_gesto\_palma\_abierta evalúa si una mano detectada está en posición de palma abierta, utilizando los puntos de referencia (landmarks) proporcionados por MediaPipe Hands.

Dedos extendidos:

Verifica que las y-coordinates (posición vertical) de las puntas de los dedos índice, medio, anular y meñique (8, 12, 16, 20) estén por encima de sus respectivas articulaciones intermedias (6, 10, 14, 18). Esto indica que los dedos están estirados.

Pulgar abierto:

Evalúa si el pulgar está separado del resto de la mano, comparando su posición horizontal (x) con la de su articulación (4 vs 3). Se permite que sea hacia la izquierda o derecha, según si es mano izquierda o derecha.

Texto

El contenido generado por IA puede ser incorrecto.

## Interfaz del Teclado Virtual

En esta sección se dibujan los rectángulos y textos que forman cada tecla usando OpenCV, lo que representa la interfaz gráfica real del teclado sobre el video capturado.

Texto

El contenido generado por IA puede ser incorrecto.

## Predicción de Palabras

Esta función toma el texto que el usuario ha escrito (texto) y, usando un modelo de trigramas previamente cargado con pickle, sugiere las 3 palabras más probables que podrían completar la última palabra escrita.

Texto

El contenido generado por IA puede ser incorrecto.

## Captura de video y bucle principal

Inicia la cámara y comienza un ciclo continuo donde se procesa cada fotograma capturado.

Texto

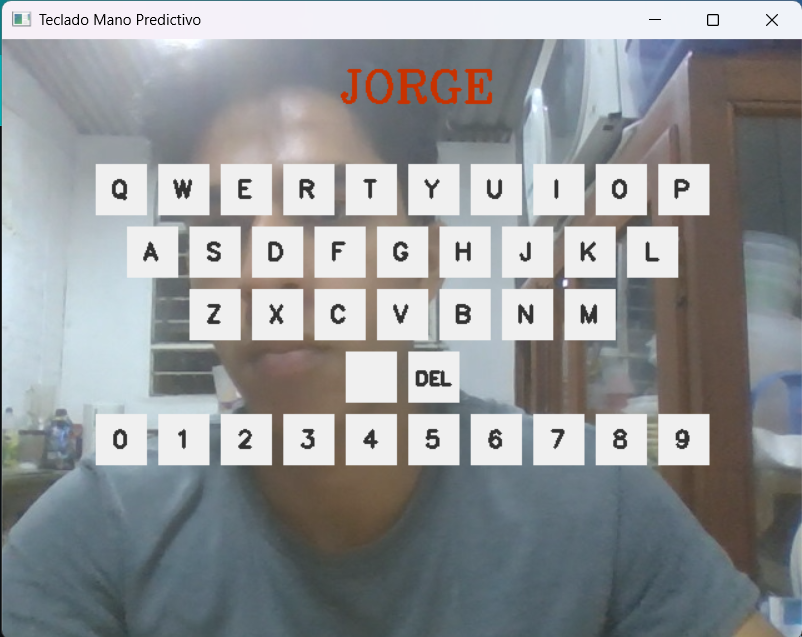
El contenido generado por IA puede ser incorrecto.

# Pruebas

Se realizaron diversas pruebas funcionales con el fin de validar el correcto funcionamiento del teclado virtual con control por gestos y el sistema de autocompletado.

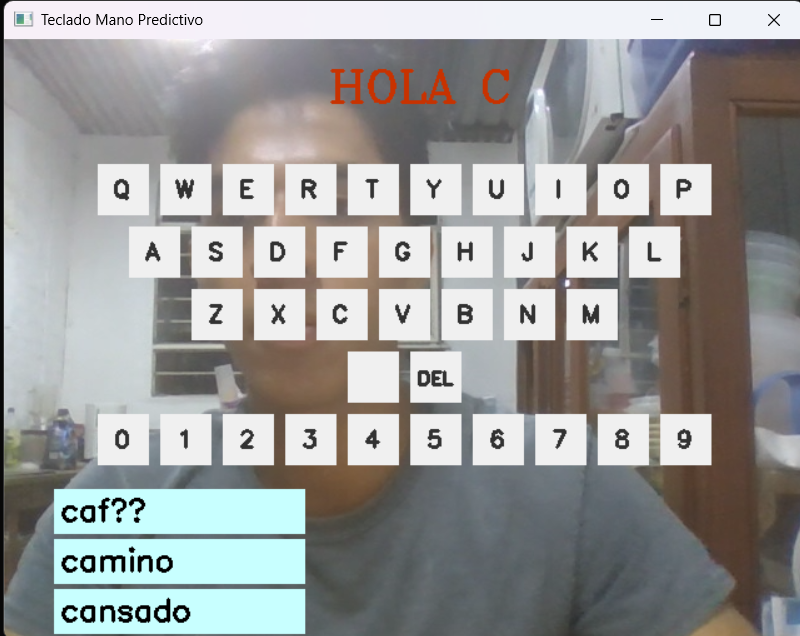
**Caso de Prueba 1: Selección de caracteres individuales**

* **Objetivo:** Verificar que el sistema detecte la posición del dedo índice y registre correctamente la selección de una tecla.
* **Procedimiento:** El usuario posiciona la punta del dedo sobre una tecla del teclado virtual durante al menos un segundo.
* **Resultado esperado:** La tecla se marca como seleccionada, se reproduce el sonido de clic y el carácter correspondiente se agrega al texto.
* **Resultado obtenido:** Funciona correctamente. La selección es precisa y sensible al tiempo.



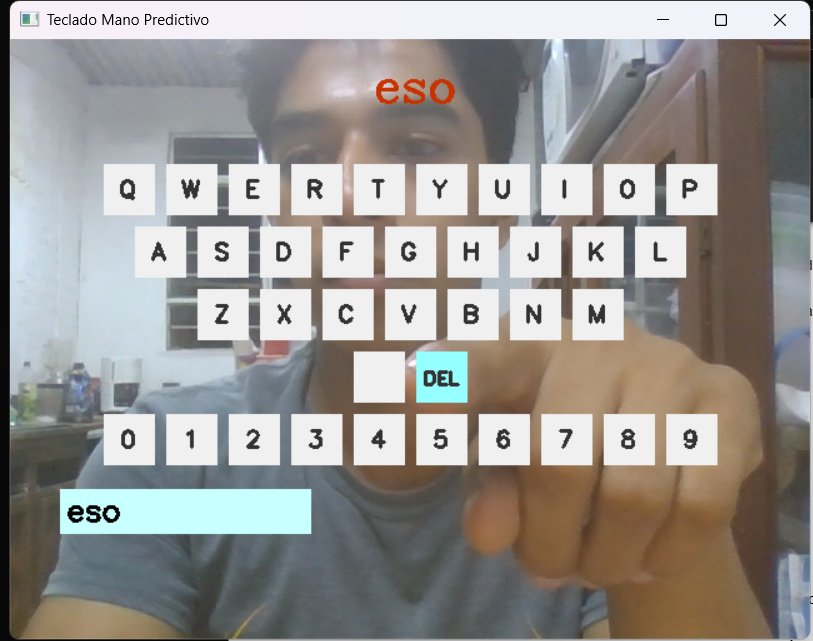
**Caso de Prueba 2: Inserción de espacio**

* **Objetivo:** Validar que el carácter de subrayado \_ sea interpretado como un espacio en blanco.
* **Procedimiento:** El usuario mantiene el dedo sobre la tecla \_ por más de un segundo.
* **Resultado esperado:** Se inserta un espacio en el texto.
* **Resultado obtenido:** Espacio insertado correctamente.



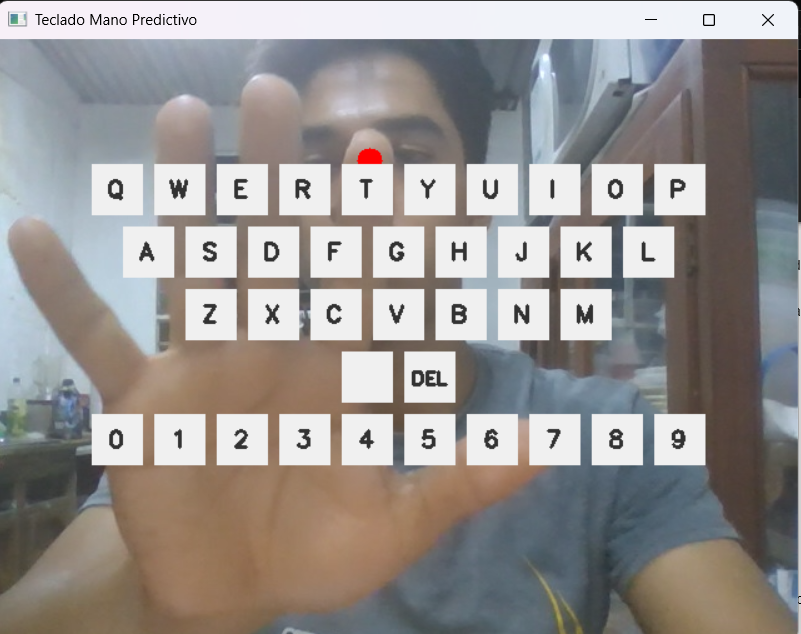
**Caso de Prueba 3: Eliminación de caracteres**

* **Objetivo:** Confirmar que la tecla DEL elimina el último carácter escrito.
* **Procedimiento:** El usuario posiciona el dedo sobre la tecla DEL durante más de un segundo.
* **Resultado esperado:** Se elimina el último carácter del texto.
* **Resultado obtenido:** El texto se modifica eliminando correctamente el último carácter.



**Caso de Prueba 4: Gesto de mano abierta (limpieza de texto)**

* **Objetivo:** Verificar que el sistema borre todo el texto cuando se detecta la palma completamente abierta.
* **Procedimiento:** El usuario abre completamente la mano dentro del área visible de la cámara.
* **Resultado esperado:** El texto actual se borra y se reinicia el sistema de sugerencias.
* **Resultado obtenido:**Texto limpiado correctamente.



# Conclusiones

El sistema desarrollado demuestra que es posible implementar un teclado virtual controlado mediante gestos de la mano utilizando visión por computadora, permitiendo la escritura sin contacto físico y con asistencia inteligente a través de autocompletado. La interacción mediante la punta del dedo índice, junto con el modelo de predicción basado en trigramas, permite una experiencia fluida y accesible.

## Limitaciones detectadas:

* **Precisión de detección en condiciones variables de luz:** El rendimiento del sistema puede verse afectado por cambios en la iluminación, generando errores en la detección de la mano o en la posición del dedo.
* **Tiempo de selección de tecla:** El tiempo de espera (1 segundo) necesario para confirmar una tecla puede resultar lento para usuarios experimentados, afectando la velocidad de escritura.
* **Limitación a una sola mano:** El sistema actual solo reconoce una mano a la vez, lo que restringe el potencial de entrada con ambas manos o la navegación con otra.
* **Modelo de predicción básico:** El modelo de lenguaje basado en trigramas, aunque funcional, es limitado en vocabulario y contexto. No aprende del usuario ni corrige errores comunes.
* **Dependencia del fondo visual:** Fondos complejos o ruidosos pueden afectar la precisión del seguimiento del dedo y de la palma.

## Mejoras futuras recomendadas:

**Optimización del tiempo de interacción**, permitiendo una configuración dinámica del tiempo necesario para "presionar" una tecla (según experiencia o preferencia del usuario).

**Implementación de un modelo de lenguaje más avanzado**, como un modelo neuronal tipo RNN o incluso el uso de modelos preentrenados ligeros para mejorar las sugerencias contextuales.

**Soporte para gestos adicionales**, como desplazamiento del teclado, cambio de idioma o borrado por gestos específicos.

**Integración de retroalimentación visual o sonora más clara**, como iluminación de teclas seleccionadas con animación o variaciones de sonido según la acción.

**Compatibilidad multiplataforma**, optimizando el rendimiento para que funcione de forma eficiente en distintos sistemas operativos y con distintas cámaras.

**Modo de calibración inicial**, que permita adaptar el teclado virtual a diferentes resoluciones, distancias o tamaños de mano.