**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA**Logotipo, nombre de la empresa

Descripción generada automáticamente

**FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN**

**MATERIA:** Big Data

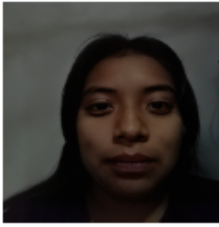
**DOCENTE:** Adrian Rodriguez Aguiñaga

**Proyecto Final:** :

Clasificación de notas periodísticas mediante aprendizaje automático (Teachable Machine)

**Alumn@:**

Cuevas Cortes Viridiana



**Grupo:**

961

# 

## Introducción

Este proyecto consiste en desarrollar un modelo capaz de clasificar imágenes periodísticas en siete categorías informativas utilizando una herramienta de aprendizaje automático entrenado con Teachable Machine.

Las imágenes provienen de mi trabajo real en el área de redacción del medio *Diario Tijuana*,todos los derechos de imagen están a cargo de ellos y de Border Zoom, por lo que representan situaciones de seguridad, política, cultura, educación, clima, estado y noticias locales.

El modelo permite automatizar la clasificación de contenido visual, haciéndolo útil para acelerar los procesos editoriales y mejorar la organización de información.

## Objetivo General

Desarrollar un modelo de clasificación de imágenes que categorice fotografías periodísticas en siete secciones informativas mediante Teachable Machine e integrarlo en una aplicación

## Descripción del Proyecto

El proyecto consiste en una **aplicación web básica** que carga el modelo entrenado con Teachable Machine e identifica, en tiempo real, la categoría correspondiente a una imagen que el usuario sube o captura.  
 La aplicación muestra:

* La imagen seleccionada
* La predicción de la clase
* El porcentaje de confianza

Este sistema permite automatizar tareas de clasificación en redacciones donde se manejan grandes volúmenes de contenido visual.

## 

## Explicación del Modelo de Teachable Machine

Se utilizó un **modelo de Clasificación de Imágenes**, diseñado para identificar patrones visuales y asignar etiquetas basadas en ejemplos entrenados.

**Clases entrenadas**

El modelo clasifica imágenes en las siguientes siete categorías periodísticas:

1. Seguridad
2. Estado
3. Local
4. Clima
5. Educación
6. Política
7. Cultura

Estas clases corresponden a las secciones reales del periódico *Diario Tijuana*.

**Dataset utilizado**

* Las imágenes fueron recolectadas de la biblioteca de medios generado en el área de redacción.
* Se utilizaron entre 20–50 imágenes por categoría (ajustar si tienes el número real).
* Las imágenes incluyen variabilidad en condiciones de iluminación, distancia, ángulos y contexto.

**Entrenamiento del modelo**

* Herramienta: Teachable Machine

**Funcionamiento del modelo**

1. La imagen es cargada por el usuario en la aplicación.
2. El modelo analiza sus características visuales usando convoluciones preentrenadas.
3. Compara la imagen con patrones aprendidos durante el entrenamiento.
4. Genera probabilidades para cada clase.
5. Devuelve la categoría con mayor porcentaje.

## 

## Proceso de Desarrollo y Decisiones Técnicas

### 

**Recolección y organización de datos**

Las imágenes fueron seleccionadas desde archivos reales del periódico para asegurar una clasificación en base a escenarios y contenido visual de acuerdo a la calidad que se utiliza en el medio además, que la herramienta utilizada se basa principalmente en las paletas de colores contrario al contenido de la imagen por lo cual se tuvo que hacer un descarte de ciertas imágenes.

**Entrenamiento en Teachable Machine**

Se eligió Teachable Machine debido a:

* Su facilidad de uso
* Exportación directa a JavaScript
* Modelo ligero para web
* Además de ser requisito para este proyecto la utilización de esta herramienta

**Desarrollo de la aplicación web**

Componentes utilizados:

* **HTML** → estructura
* **CSS** → estilos básicos
* **JavaScript** → carga del modelo y predicciones

**Pruebas y ajustes**

Se probaron imágenes nuevas para evaluar:

* Precisión del modelo
* Errores de predicción
* Comportamiento ante imágenes ambiguas

## 

## Problemática Social Abordada

La saturación de contenido visual en medios digitales genera:

* Sobrecarga de trabajo en las redacciones
* Tiempos prolongados de clasificación manual
* Riesgo de errores humanos
* Retrasos en la publicación de noticias

## Justificación del Impacto

El modelo propuesto ayuda a:

* Acelerar la clasificación de archivos multimedia
* Mejorar la organización interna del periódico
* Facilitar el trabajo de reporteros y editores
* Reducir la carga operativa en la redacción

## Aplicaciones Prácticas del Modelo

1. **Clasificación automática en bases de datos** Organiza imágenes nuevas de forma instantánea.
2. **Asistente para reporteros** Un reportero puede tomar una foto y obtener en segundos la sección correspondiente.
3. **Análisis de tendencias** Permite visualizar qué temas informativos predominan en cierto periodo.

## Resultados Obtenidos

* El modelo logra identificar correctamente la mayoría de las imágenes de prueba.
* Las clases más fáciles de identificar son Seguridad y Política por sus elementos visuales claros.
* La clase Cultura puede presentar confusiones si su contexto visual se parece a eventos políticos o educativos.
* La precisión general es adecuada para un primer prototipo funcional.

# 

## Conclusiones

El proyecto demuestra que es posible implementar un sistema de clasificación automática de imágenes para uso periodístico utilizando Teachable Machine y tecnologías web accesibles.

La aplicación facilita el trabajo en redacción editorial, mejora la eficiencia y abre la puerta a integrar más automatización en medios locales.

Además, se logró un aprendizaje significativo sobre modelos de clasificación, organización de datasets y uso práctico de redes neuronales en aplicaciones web.

## Enlace al Video (YouTube)

[*https://youtu.be/rIf4oZ\_9qro*](https://youtu.be/rIf4oZ_9qro)

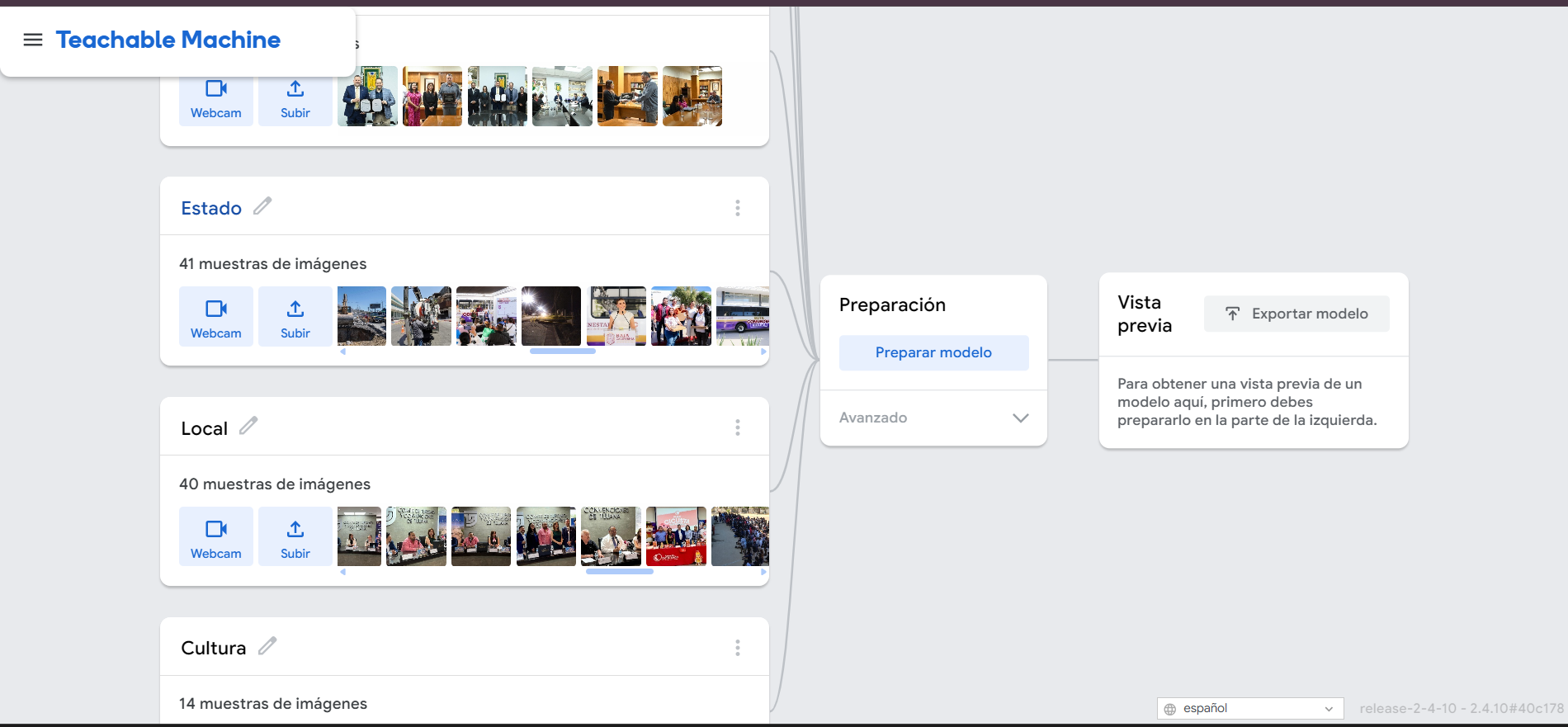
## Repositorio del Proyecto

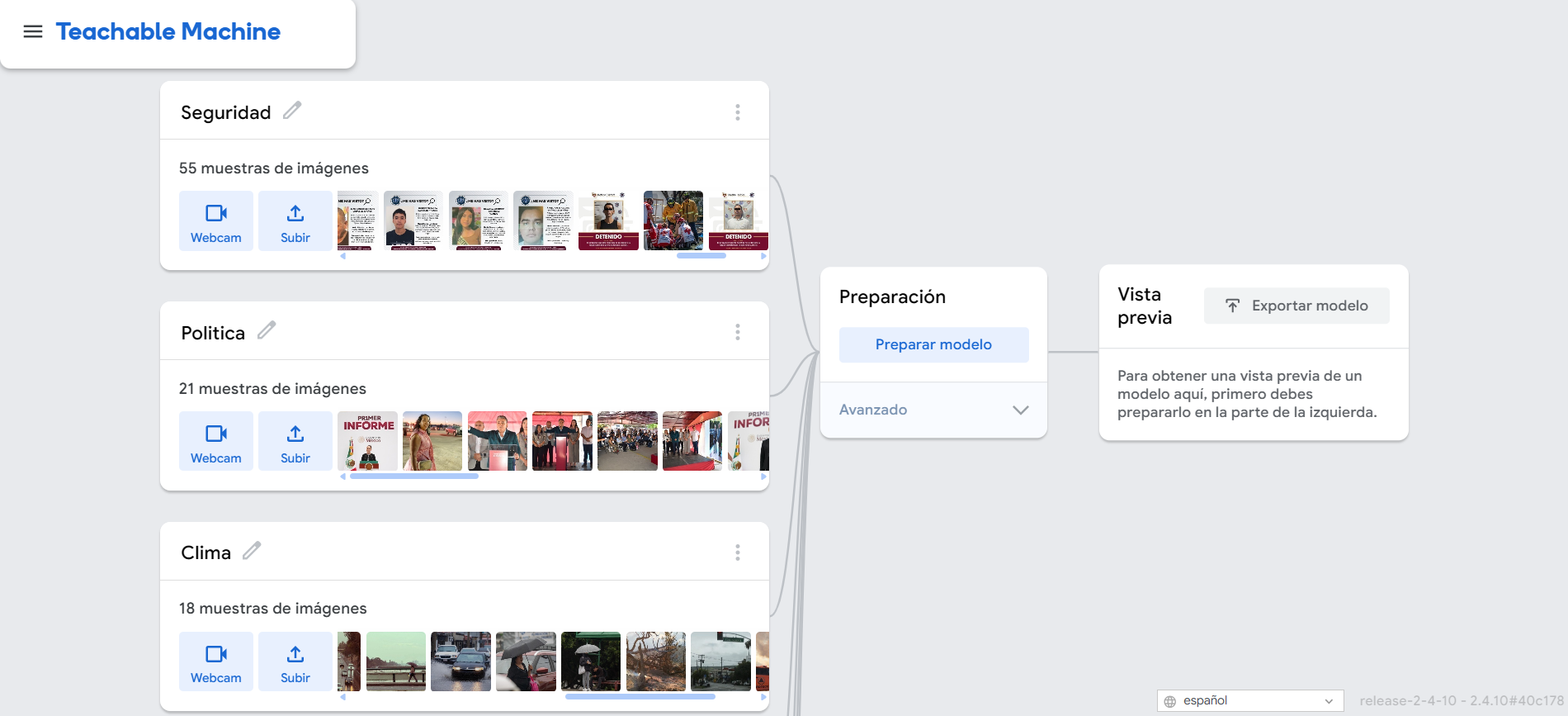
[*https://github.com/Viridiana-2203301/Proyecto-Big-Data.git*](https://github.com/Viridiana-2203301/Proyecto-Big-Data.git)

## 

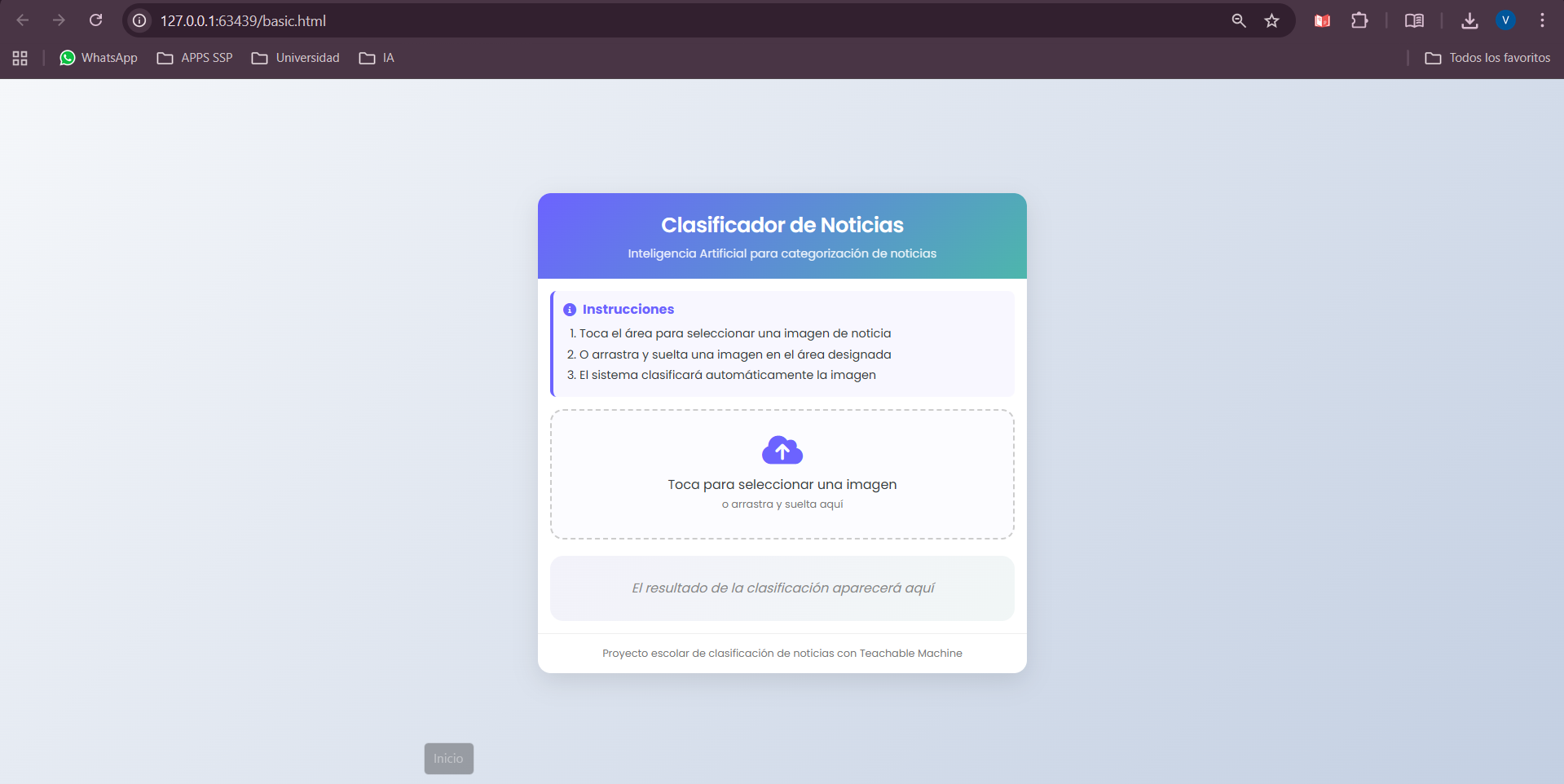
## Anexos

* **Capturas del entrenamiento:**

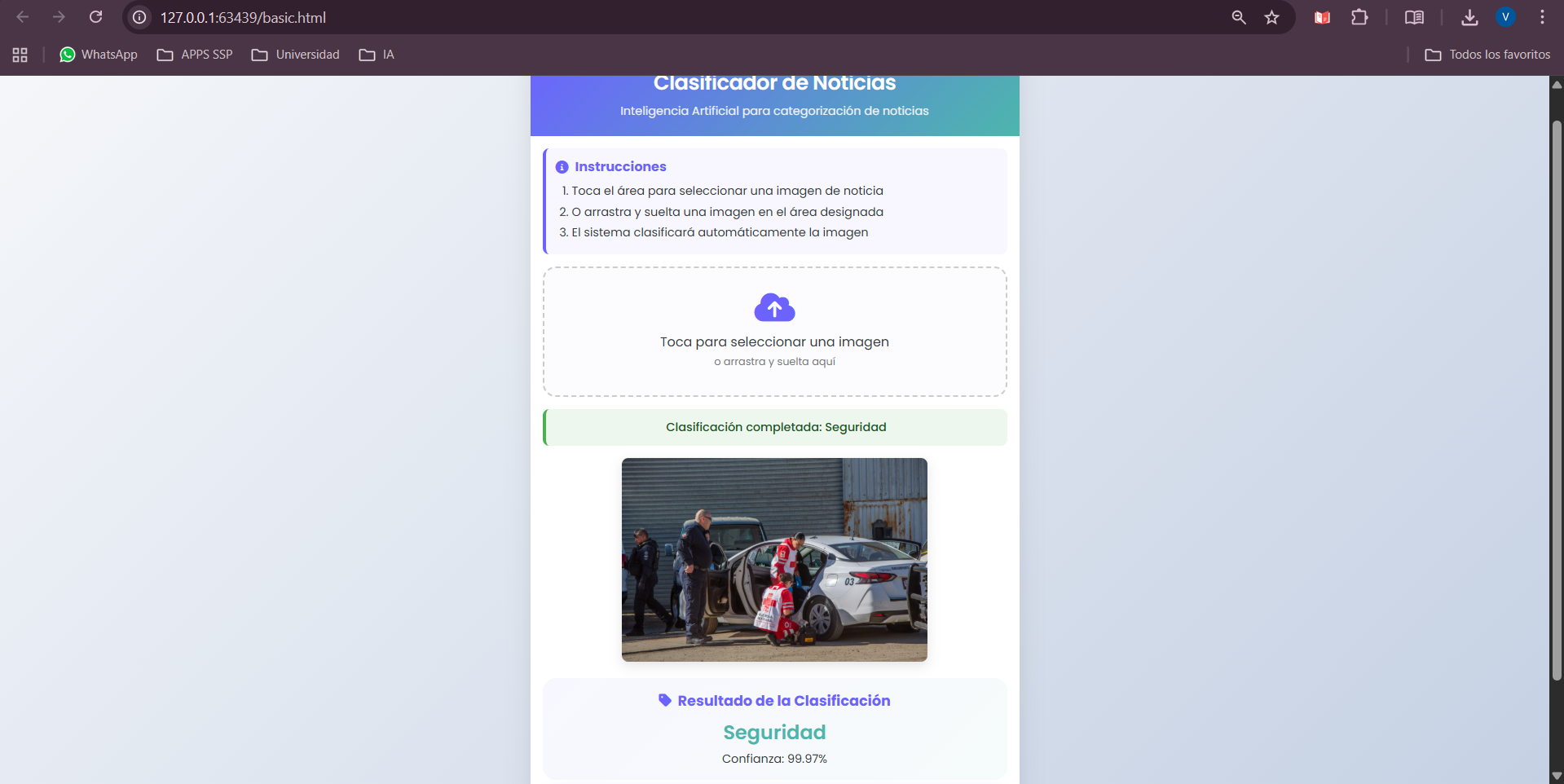
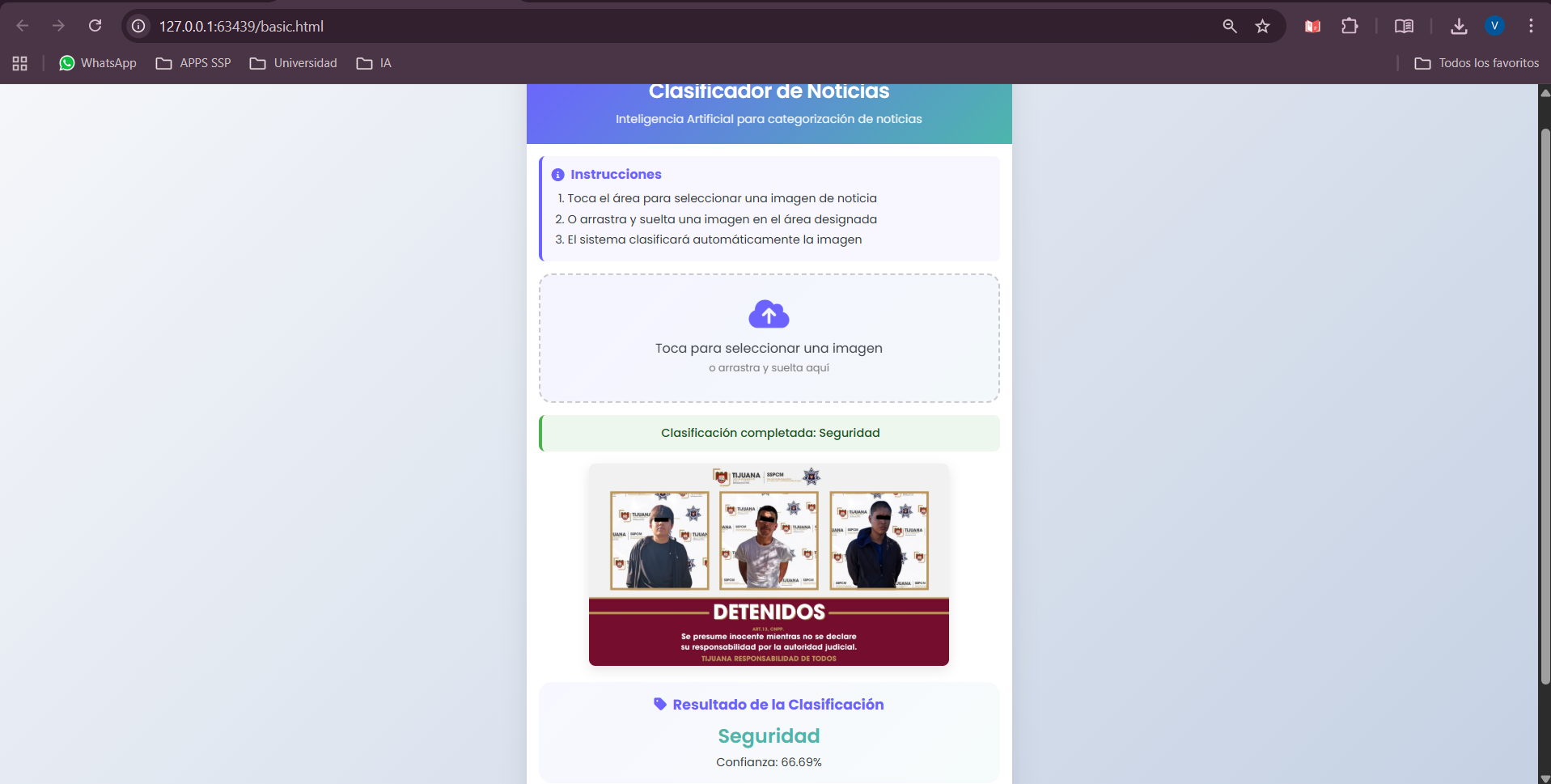
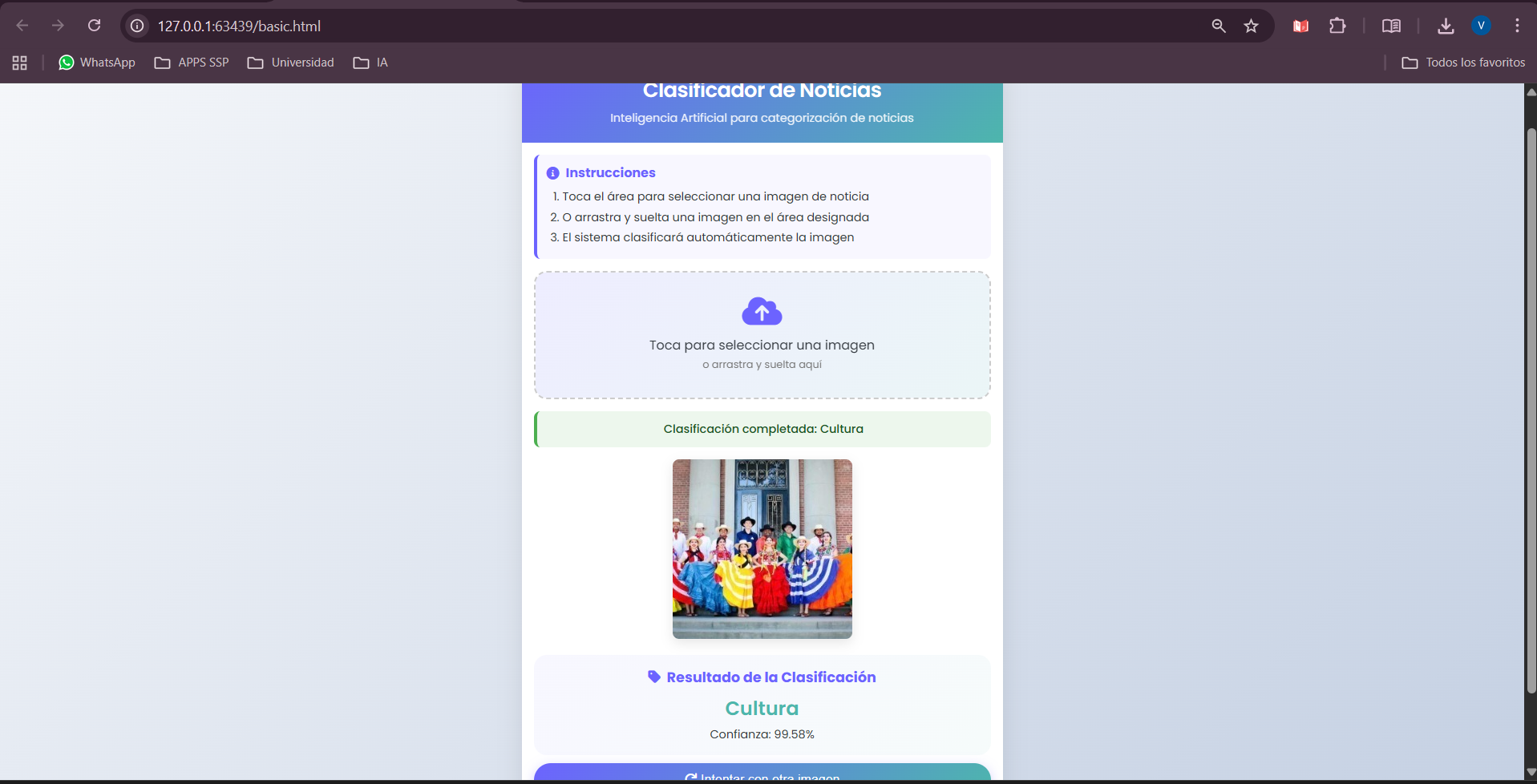
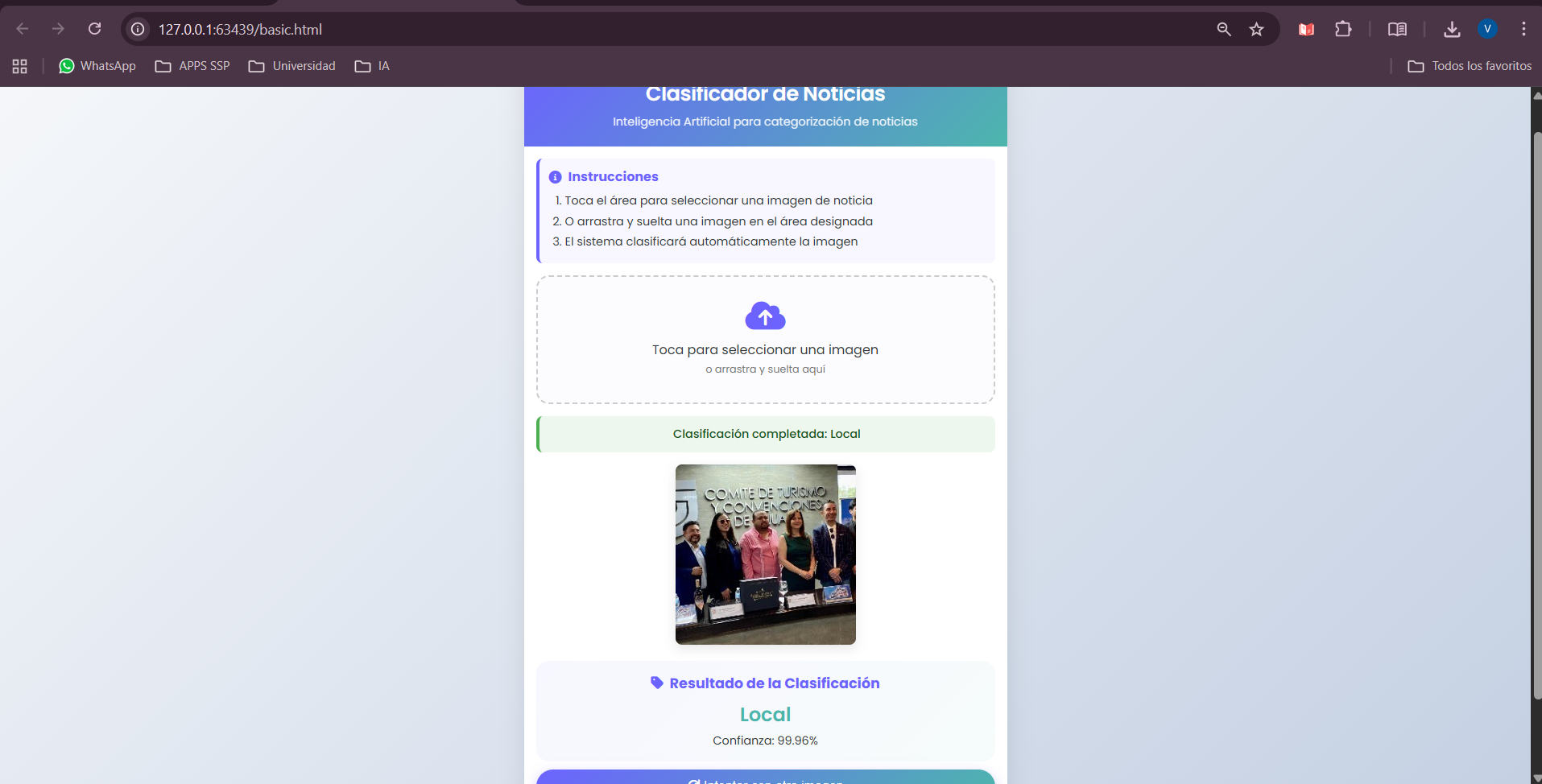
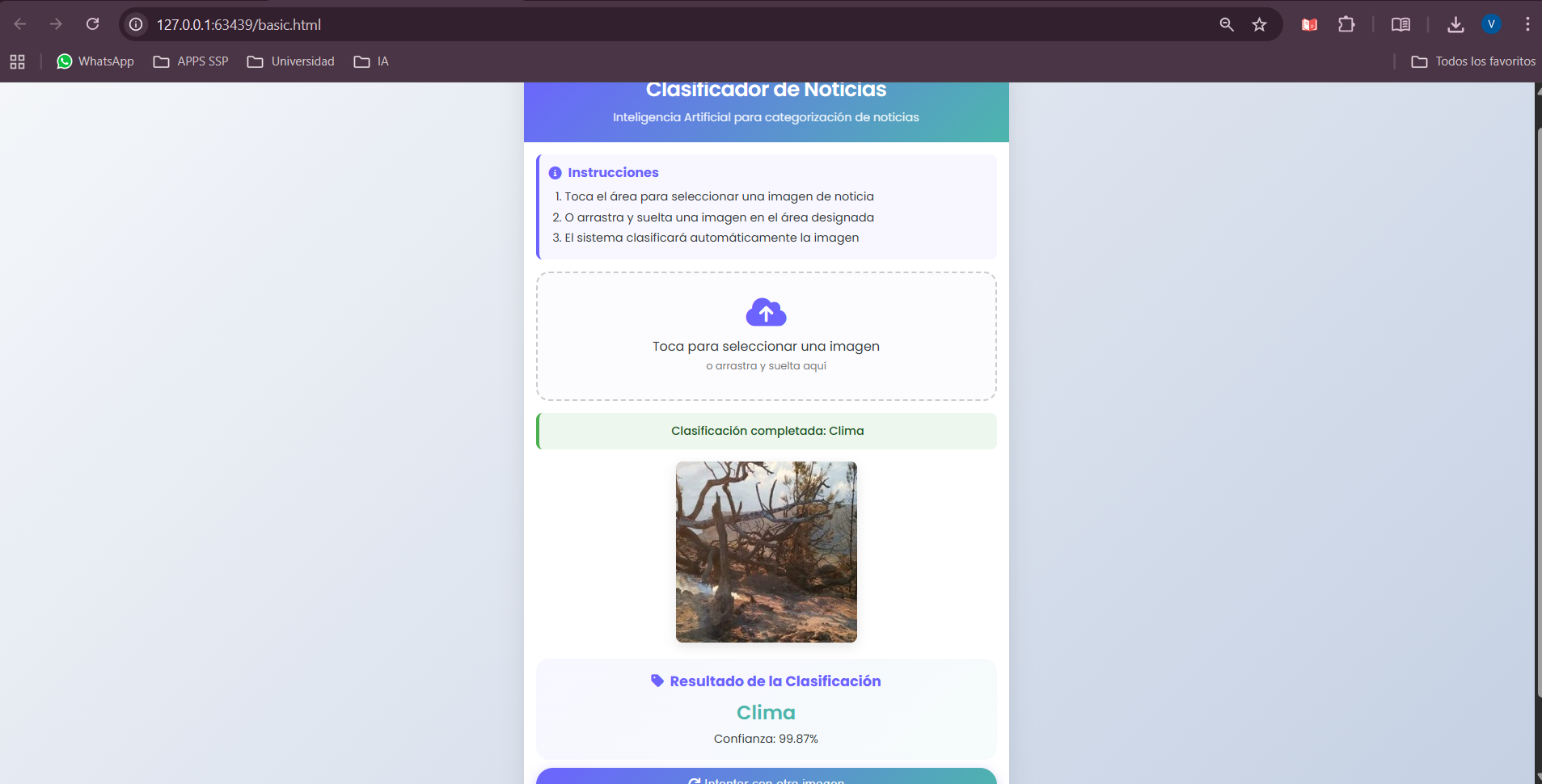
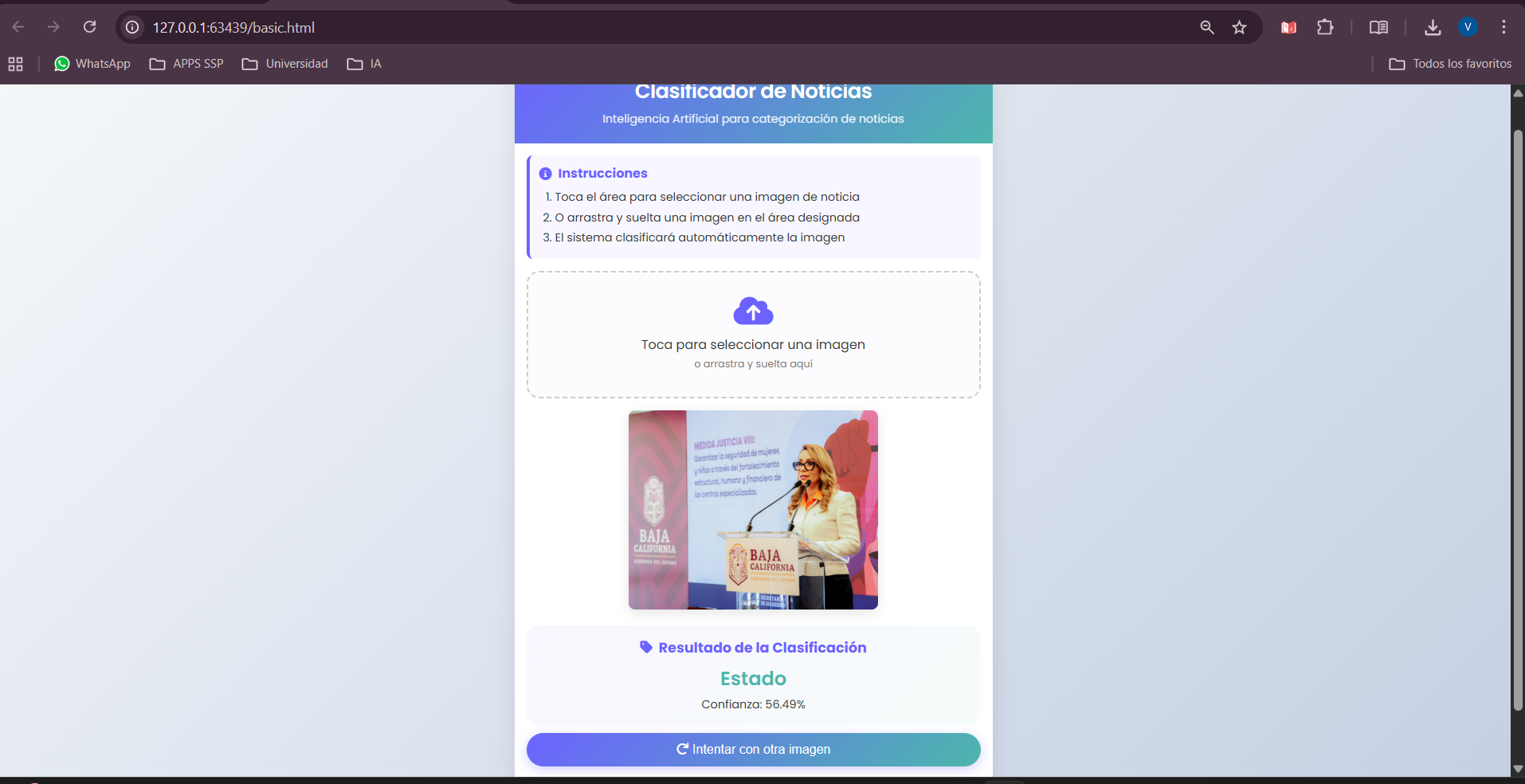
****

****

* **Capturas de la interfaz web:**

****

* **Ejemplos de predicciones:**

****

* **Estructura del código:**
  + **Index.html**
  + **Codigo.js**