



TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE TLAXIACO

REALIDAD AUMENTADA

PRACTICA 8. MARCADORES POR IMÁGENES

CARRERA:

INGENIERIA EN SISTEMAS COMPUTACIONALES

INTEGRANTES DEL EQUIPO:

SANDRA GABRIELA VELASCO GUZMÁN

RAEL GABRIEL BAUTISTA

AMILKAR VLADIMIR REYES REYES

ARNOL JESUS CRUZ ORTIZ

DOCENTE

ING. JOSE ALFREDO ROMAN CRUZ

Tlaxiaco, Oax., 28 de Abril de 2025.



“Educación, ciencia y tecnología, progreso día con día”®

INDICE DE CONTENIDO

INTRODUCCIÓN	3
OBJETIVO	3
MATERIALES	3
DESARROLLO	4
PREPARACIÓN DEL ENTORNO	4
ESTRUCTURACIÓN BÁSICA DEL ARCHIVO HTML	5
CONFIGURACIÓN DE LA ESCENA:	6
ESTRUCTURA FINAL DE LA CARPETA DE LA PRACTICA DE MARCADORES POR IMÁGENES	6
MARCADOR PERSONALIZADO POR IMAGEN UTILIZADO PARA EL RECONOCIMIENTO	7
PRUEBA DEL FUNCIONAMIENTO	7
SUBIDA A UN SERVIDOR WEB	8
CONCLUSIÓN	9
BIBLIOGRAFÍAS	9

TABLA DE ILUSTRACIONES

Ilustración 1 Creacion del index.html	4
Ilustración 2 Codigo index.html	5
Ilustración 3 Estructura final del marcador por imagen	6

INTRODUCCIÓN

La Realidad Aumentada (RA) permite superponer elementos virtuales en el mundo real a través de la cámara de un dispositivo. Esta práctica tiene como finalidad introducir el uso de marcadores por imágenes como punto de anclaje para mostrar contenido tridimensional utilizando tecnologías web como A-Frame y AR.js. Estas herramientas permiten desarrollar experiencias inmersivas de manera sencilla y sin necesidad de conocimientos profundos en gráficos 3D, lo que facilita la implementación de RA directamente desde el navegador.

OBJETIVO

Desarrollar una aplicación web de realidad aumentada que reconozca imágenes como marcadores y despliegue objetos 3D en tiempo real usando A-Frame y AR.js.

MATERIALES

- ✓ Computadora o smartphone con cámara funcional.
- ✓ Navegador web moderno (Chrome, Firefox).
- ✓ Editor de texto (Visual Studio Code, Sublime Text, etc.).
- ✓ Conexión a internet para cargar las librerías desde CDN.
- ✓ Imagen personalizada para usar como marcador NFT.

DESARROLLO

Para llevar a cabo esta práctica, se utilizó la biblioteca AR.js en conjunto con A-Frame, un framework basado en HTML que permite construir experiencias de realidad virtual y aumentada de manera sencilla. El objetivo fue desarrollar una pequeña aplicación web que reconociera un marcador visual y mostrara una imagen superpuesta.

PREPARACIÓN DEL ENTORNO

Se creó una carpeta de proyecto denominada RA_MarcadorImagen, la cual contiene un archivo HTML principal y los recursos necesarios para la prueba. No se requiere servidor local, ya que la tecnología puede ejecutarse directamente desde el navegador, siempre que se habiliten los permisos de cámara.

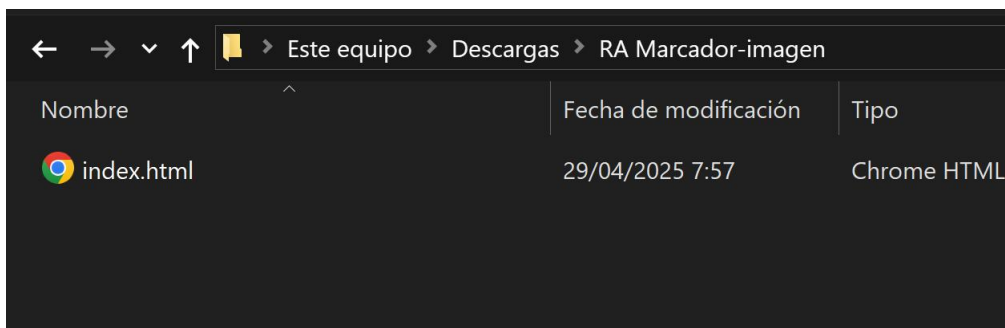


Ilustración 1 Creacion del index.html

En esta práctica estoy desarrollando una experiencia de realidad aumentada basada en marcadores (Image Markers) utilizando **A-Frame** y la librería **AR.js**, con integración de un modelo 3D animado en formato GLB. El propósito principal es detectar un marcador personalizado (archivo .patt) a través de la cámara web y, una vez identificado, visualizar un modelo 3D sobre el mismo.

El código HTML utiliza el framework **A-Frame** para manejar escenas en 3D y la extensión **AR.js** para habilitar la funcionalidad de realidad aumentada. Las funcionalidades clave que implementa esta práctica son:

Carga de modelo 3D: A través de la etiqueta `<a-assets>` se precarga un archivo GLB (asset.glb) que contiene el modelo animado (en este caso, un personaje como Bowser).

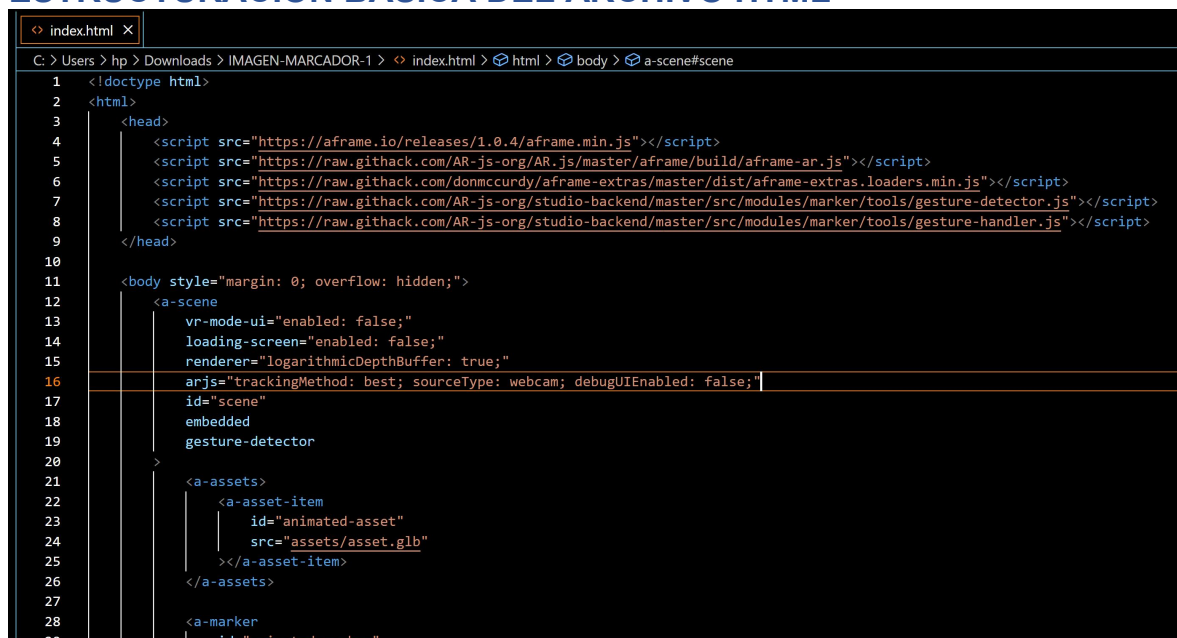
Detección de marcador personalizado: Se utiliza un marcador con tipo pattern definido por el archivo marker.patt, lo que permite que el modelo solo aparezca cuando se detecta este patrón visual con la cámara.

Animación del modelo: El modelo está animado gracias al componente animation-mixer, que permite que la animación integrada en el archivo .glb se reproduzca de forma continua (loop: repeat).

Interacción por gestos: Se incluyen scripts de detección y manejo de gestos (gesture-detector.js y gesture-handler.js) que permiten manipular el modelo (por ejemplo, moverlo, escalarlo o rotarlo con gestos táctiles o del mouse).

Cámara virtual: Se añade una entidad <a-entity camera> para definir la perspectiva del usuario dentro de la escena aumentada.

ESTRUCTURACIÓN BÁSICA DEL ARCHIVO HTML



```
1 <!doctype html>
2 <html>
3   <head>
4     <script src="https://aframe.io/releases/1.0.4/aframe.min.js"></script>
5     <script src="https://raw.githubusercontent.com/AR-js-org/AR.js/master/aframe/build/aframe-ar.js"></script>
6     <script src="https://raw.githubusercontent.com/donmccurdy/aframe-extras/master/dist/aframe-extras.loaders.min.js"></script>
7     <script src="https://raw.githubusercontent.com/AR-js-org/studio-backend/master/src/modules/marker/tools/gesture-detector.js"></script>
8     <script src="https://raw.githubusercontent.com/AR-js-org/studio-backend/master/src/modules/marker/tools/gesture-handler.js"></script>
9   </head>
10
11   <body style="margin: 0; overflow: hidden;">
12     <a-scene
13       vr-mode-ui="enabled: false;"
14       loading-screen="enabled: false;"
15       renderer="logarithmicDepthBuffer: true;"
16       arjs="trackingMethod: best; sourceType: webcam; debugUIEnabled: false;"
17       id="scene"
18       embedded
19       gesture-detector
20     >
21       <a-assets>
22         <a-asset-item
23           id="animated-asset"
24           src="assets/asset.glb"
25         ></a-asset-item>
26       </a-assets>
27
28       <a-marker
```

Ilustración 2 Código index.html

Librerías base: Se carga A-Frame, que permite crear escenas 3D en HTML, y AR.js, que integra funcionalidades de realidad aumentada (AR) para detectar marcadores y renderizar modelos 3D sobre ellos.

aframe-extras.loaders: Esta extensión permite cargar modelos en formato .glb o .gltf, lo cual es fundamental para mostrar modelos 3D animados en la escena.

gesture-detector y gesture-handler: Estas herramientas permiten detectar gestos (como arrastrar, hacer zoom o rotar) sobre los objetos 3D cuando el usuario interactúa con el modelo en pantalla (por ejemplo, con mouse o pantalla táctil).

CONFIGURACIÓN DE LA ESCENA:

- vr-mode-ui: Desactiva el modo de realidad virtual para enfocarse solo en AR.
- loading-screen: Oculta la pantalla de carga.
- logarithmicDepthBuffer: Mejora la precisión del renderizado en objetos lejanos.
- arjs: Configura la cámara como fuente (sourceType: webcam) y selecciona el mejor método de tracking.
- embedded: La escena está incrustada en la página web.
- gesture-detector: Habilita la detección de gestos.

ESTRUCTURA FINAL DE LA CARPETA DE LA PRACTICA DE MARCADORES POR IMÁGENES

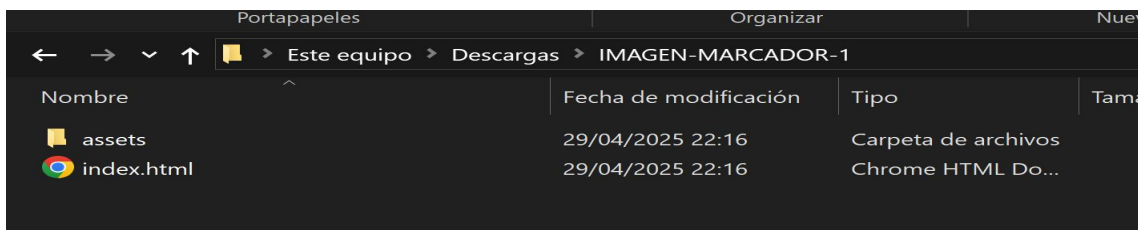
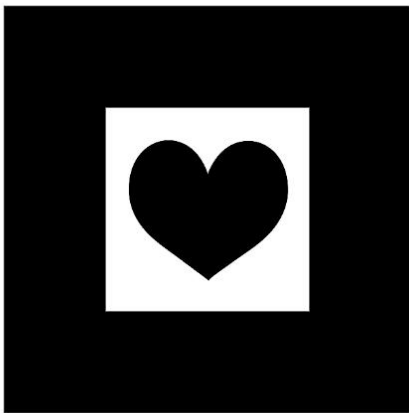


Ilustración 3 Estructura final del marcador por imagen

MARCADOR PERSONALIZADO POR IMAGEN UTILIZADO PARA EL RECONOCIMIENTO

En esta práctica se utilizó un **marcador personalizado** de tipo pattern para el reconocimiento de imagen mediante la cámara del dispositivo. Este tipo de marcador es definido por un archivo con extensión .patt, el cual contiene la codificación del patrón visual que AR.js debe reconocer para activar la visualización del modelo 3D correspondiente.

El uso de un marcador personalizado tiene como finalidad sustituir los marcadores genéricos (como el marcador Hiro o Kanji) por una imagen única, permitiendo crear experiencias más personalizadas e identificables dentro de la aplicación de realidad aumentada.



PRUEBA DEL FUNCIONAMIENTO

Para ello, se utilizó un recurso disponible en línea mediante una URL externa (archivo hospedado en GitHub Pages).



SUBIDA A UN SERVIDOR WEB

Para poder acceder a la aplicación desde cualquier dispositivo, el proyecto fue subido a <https://github.com>, y luego publicado mediante GitHub Pages, permitiendo acceder al archivo index.html a través de una URL pública.

<https://sandaravelasco.github.io/Marcador-Imagen/>

CONCLUSIÓN

Esta práctica permitió explorar la aplicación de la realidad aumentada en entornos web utilizando tecnologías accesibles y de código abierto. Se comprendió el funcionamiento de los marcadores por imagen y su utilización para posicionar contenido digital en el espacio físico a través de la cámara. La combinación de A-Frame con AR.js demostró ser una herramienta poderosa y sencilla para iniciar el desarrollo de experiencias RA interactivas, permitiendo visualizar objetos tridimensionales al reconocer una imagen, lo cual abre las puertas a aplicaciones en educación, entretenimiento, publicidad, entre otras.

BIBLIOGRAFÍAS

- ✓ A-Frame Official Site: <https://aframe.io>
- ✓ AR.js GitHub Repository: <https://github.com/AR-js-org/AR.js/>
- ✓ NFT Marker Creator: <https://carnaux.github.io/NFT-Marker-Creator/>
- ✓ Mozilla Mixed Reality: <https://mixedreality.mozilla.org/>
- ✓ Documentación de A-Frame + AR.js: <https://ar-js-org.github.io/AR.js-Docs/>