# PROYECTO: RECORRIDO VIRTUAL DE LA FIME EN AFRAME

INTRODUCCIÓN

Anteriormente se había realizado la creación del recorrido virtual de la Facultad de Ingeniería Mecánica y Eléctrica con la herramienta Unity para generar el entorno virtual con modelos 3D, esto con el fin de ayudar a los alumnos de nuevo ingreso, así como docentes nuevos y personal administrativo a encontrar con más facilidad los edificios y los salones, los departamentos y áreas recreativas. Este proyecto fue un éxito, el inconveniente era que solo lo podían disfrutar las personas que visitaran las instalaciones de la facultad, ya que este se encontraba instalado en un equipo de cómputo que se encuentra en el pasillo principal de la FIME.

Para resolver esta problemática el Laboratorio de Realidad Virtual de la FIME decidió implementar el recorrido virtual en HTML, esto tendría un alcance mucho mayor puesto que no habría que descargar ningún instalador o una aplicación como tal, no se necesitaría de un buen procesador o tarjeta gráfica para que pudieran visualizarlo. Esto sonaba bastante atractivo y funcional, pero se descubrió que el proyecto pesaba demasiado como para implementarlo en HTML, lo cual procedió a generar nuevas ideas como el bajar la calidad de los objetos 3D, limitar las áreas, etc.

Aquí es donde entra AFRAME.

Aframe es un framework en javascript para generar web en un entorno virtual. Este framework usa la arquitectura ECS (Entity Component System), usada en el desarrollo de juegos donde cada objeto es una entidad.

PLANTEAMIENTO DEL PROBLEMA

El Recorrido Virtual de la FIME actualmente presenta problemas para poder implementarse en una página web, por lo cual, se propone hacer un proyecto desde cero, optando por utilizar el framework “Aframe” y hacer uso de imágenes equirectángulares, las cuales sustituyen los modelos 3D que el recorrido tenía.

OBJETIVO

Implementar el Recorrido Virtual de la Facultad de Ingeniería Mecánica y Eléctrica en HTML utilizando el framework AFRAME.

Hacer que el proyecto en HTML pese menos de 100Mb.

CÓDIGO

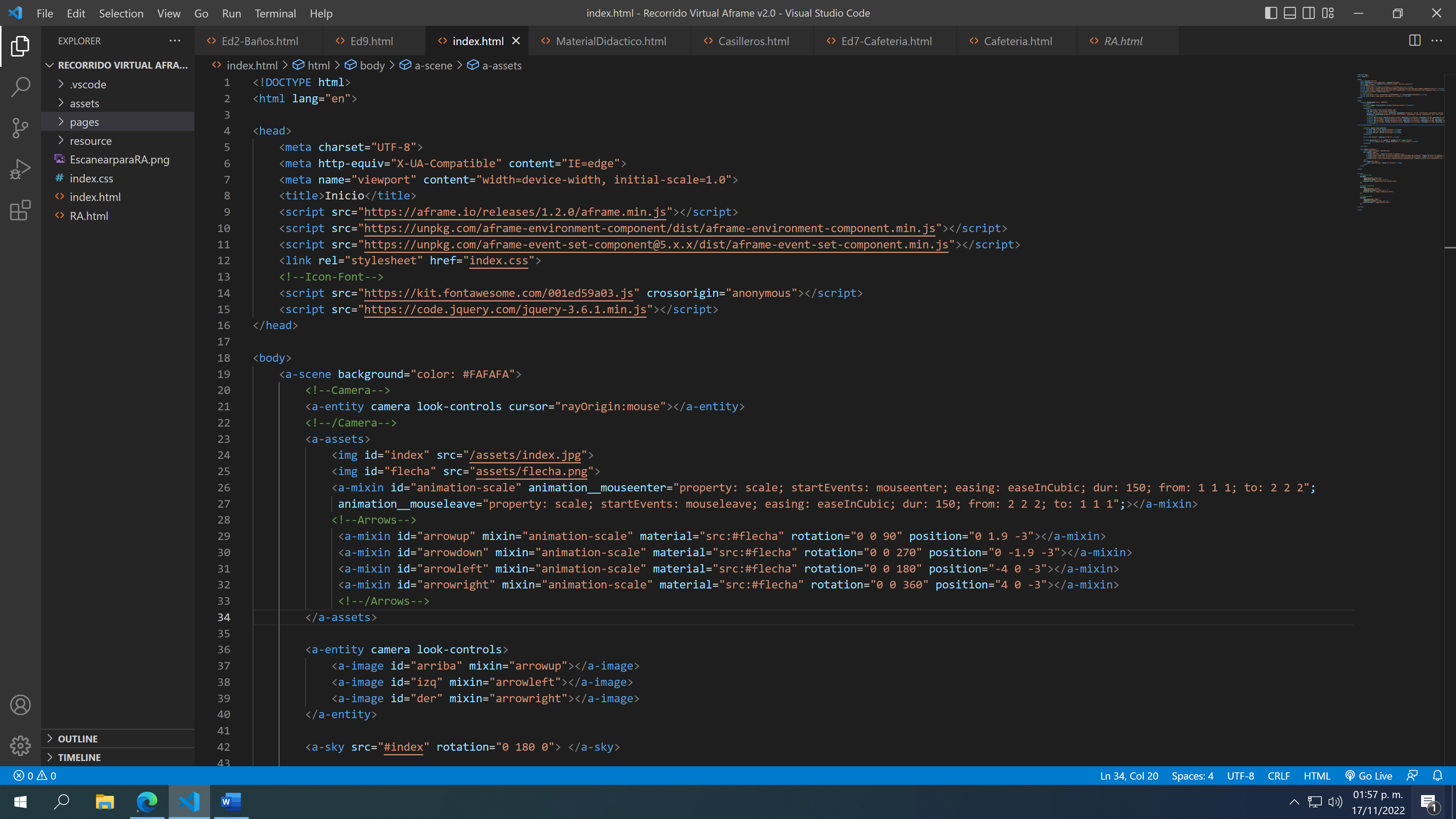


Ilustración : index general del proyecto.

Como podemos observar en la Ilustración 1, abrimos el proyecto definiendo que es una página HTML, enseguida tenemos el *Head* en donde especificamos las librerías que estamos utilizando, entre las cuales se encuentra la liga de “Aframe” que es la que nos dará vida a todo el proyecto.

Después, nos encontramos con el *Body* y aquí es donde empezamos a hacer uso de las bibliotecas de Aframe.

<a-scene> Se utiliza para crear una escena, y dentro de esta irán todos los objetos 3D e imágenes que vayamos a necesitar para crear nuestro entorno VR.

<a-entity> Se usa para crear un objeto vacío, aquí puede ir cualquier tipo de objeto, hasta incluso una imagen jpg, png, etc.

<a-assets> Dentro de esta etiqueta se definen las imágenes, objetos y funciones que se vayan a utilizar dentro de la escena.

<img> Se utiliza para definir una imagen, esta etiqueta solo se utiliza dentro de los Assets ya que esta es la que nos indica que se mandará llamar esta imagen más adelante. En esta etiqueta no se pueden definir parámetros para las imágenes como lo viene siendo “postition”, “rotation”, y “scale”.

<a-mixin> Esta etiqueta se usa para crear funciones repetitivas. Esto es de gran ayuda para optimizar el código y establecer la función en un solo lugar. Los Mixin siempre deben llevar “id”, pues es con este que se mandará llamar más adelante.

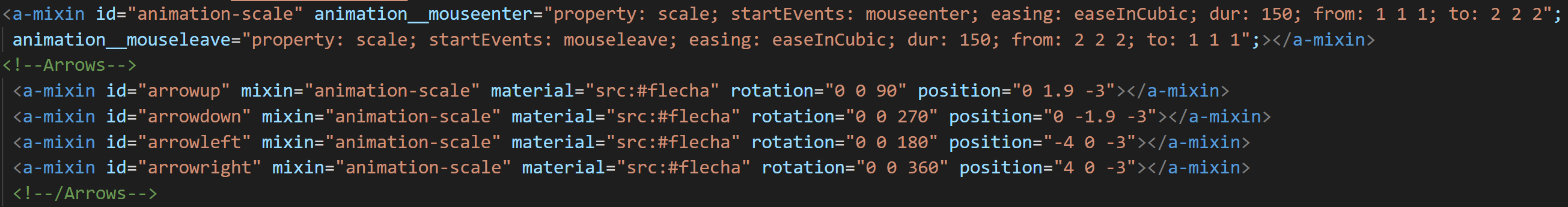
Nota: Si bien, la etiqueta <img> se podría decir que es parecida a la etiqueta <a-mixin>, a diferencia que la primera solo se usa para definir imágenes y, como ya dijimos, no se pueden definir parámetros.

<a-sky> Se utiliza para agregar colores de fondo o imágenes de 360° a la escena. Un sky es una gran esfera con un color o una textura que nosotros mismos podemos agregarle.

<a-plane> Esta crea una superficie geométrica en donde, por lo general, se agregan textos ya que funciona como un pequeño recuadro que resalta estos.

<a-image> Es la función en donde se manda llamar la imagen que definamos en la etiqueta <img>; dentro de esta etiqueta también se puede mandar llamar la imagen sin tener que definirla previamente, pero una buena práctica indica que las imágenes deben cargarse desde un inicio agregándolas en los Assets.

Sabiendo esto, analicemos el siguiente código:



El primer Mixin que podemos ver se utilizó para definir la animación que hará un objeto cuando pasemos el mouse encima de él; en este caso estamos definiendo que tendrá la propiedad “scale”, y que el evento se iniciará cuando el cursor pasa por encima de él. También podemos observar que tiene una duración de 150 milisegundos