

DOCUMENTACIÓN TÉCNICA DEL PROYECTO

"Integración de Realidad Virtual y Sensorización del DIC"

Índice

1. Introducción.....	2
2. Desarrollo del modelo 3D de DIC en Blender	2
3. Implementación de experiencia VR con Babylon.js	3
4. Montaje y configuración de sensores con Arduino	3
5. Instalación y configuración del broker MQTT en Raspberry Pi	4
6. Diseño y configuración de la base de datos InfluxDB.....	4
7. Creación del dashboard en Grafana para visualización de datos.....	5
8. Modelo compatible con metodología BIM.....	6

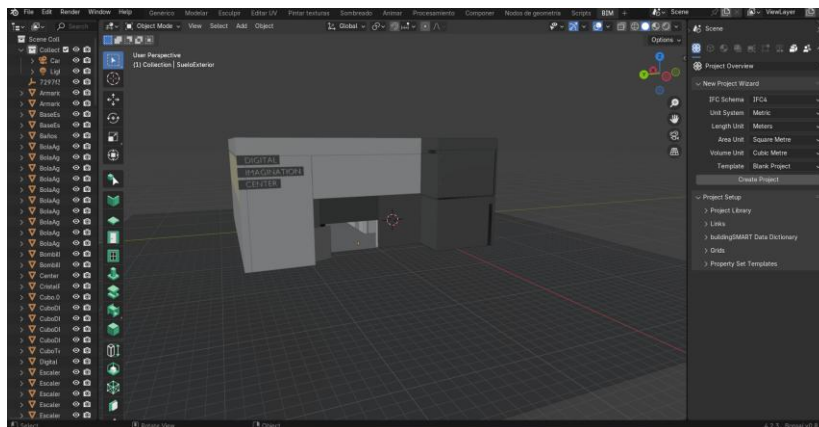
1. Introducción

El proyecto busca implementar un sistema de sensorización y visualización de datos en tiempo real en el edificio DIC del ITA, además de crear una experiencia de realidad virtual satisfactoria y agradable mediante el navegador web. A través de la recopilación de datos ambientales, como temperatura, humedad, niveles de CO2, ruido ambiental e iluminación, se busca optimizar el uso de calefacción e iluminación mediante decisiones automatizadas. La integración con realidad virtual permitirá una representación interactiva y en 3D del edificio y sus condiciones ambientales.



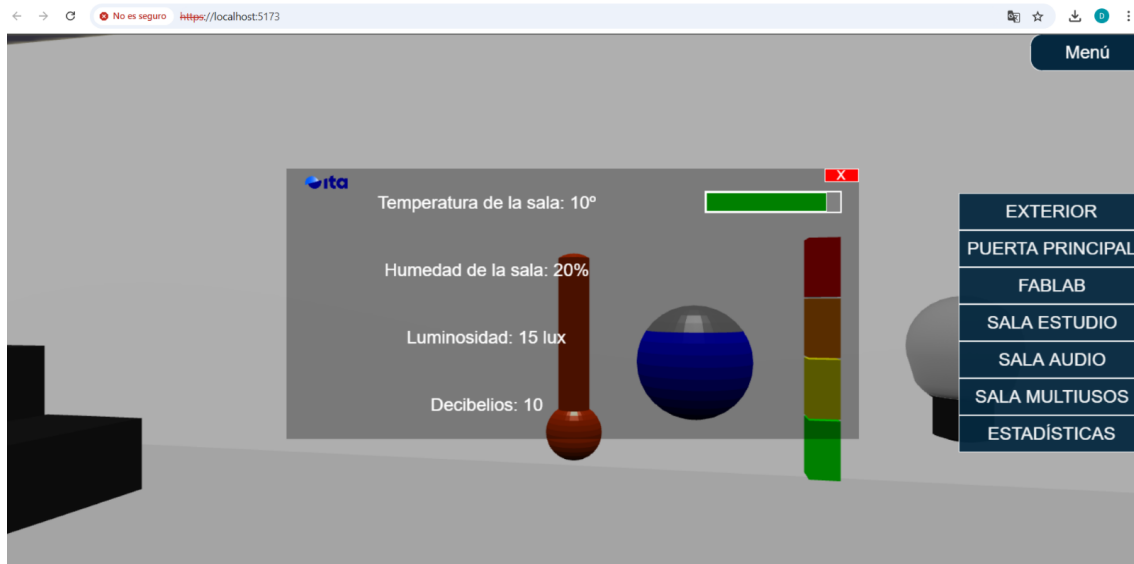
2. Desarrollo del modelo 3D de DIC en Blender

El modelo 3D del edificio DIC se ha diseñado en Blender, utilizando herramientas de modelado y texturización avanzadas. Se han aplicado técnicas de optimización para garantizar un rendimiento fluido en la web. El modelo incluye detalles arquitectónicos esenciales y es compatible con entornos de realidad virtual.



3. Implementación de experiencia VR con Babylon.js

Para la experiencia VR, se utilizó Babylon.js, un motor gráfico basado en WebGL. El modelo 3D fue exportado en formato GLB desde Blender, pudiendo importarlo en el proyecto BabylonJS. BabylonJS nos permite interactuar con cada elemento creado en Blender por separado a través de su nombre, lo que nos permitirá un control total dentro de nuestro entorno Web. Además, se añadieron paneles de datos para poder visualizar la situación de la sensorica del DIC, juntos a distintos meshes creados expresamente para representar esos datos de manera más visual. La iluminación es la más estándar para gastar los menos recursos posibles.



Además cuenta con un menú de teletransporte para poder moverte libremente y de manera rápida.

4. Montaje y configuración de sensores con Arduino

Para la adquisición de datos ambientales, se han instalado varios sensores conectados a placas ARDUINO UNO R4 WIFI. Los sensores empleados son:

- DHT11: mide temperatura y humedad.

- MQ135: detecta niveles de CO2.
- TSL2561: mide la intensidad de luz ambiental.
- GY-SPHO645: micrófono para medir el ruido ambiental.

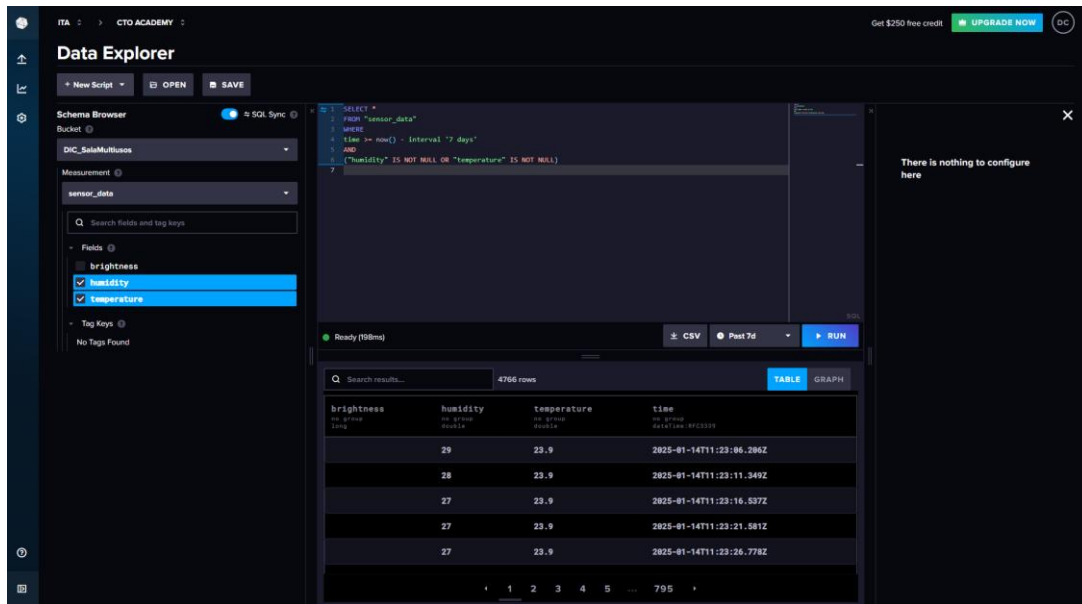
Cada sensor ha sido programado en Arduino utilizando librerías específicas para la adquisición y transmisión de datos vía MQTT.

5. Instalación y configuración del broker MQTT en Raspberry Pi

El broker MQTT, basado en Mosquitto, se instaló en una Raspberry Pi, permitiendo la comunicación entre los sensores y la base de datos. Se configuraron topics específicos para cada tipo de sensor, asegurando una transmisión eficiente de los datos. A través de un script en Python configuramos a que topic se suscribe, y a través de unas credenciales de influxdbcloud, podemos mandar los datos e insertarlos en el bucket correspondiente, siendo cada bucket una sala a sensorizar del DIC.

6. Diseño y configuración de la base de datos InfluxDB

La base de datos utilizada es InfluxDB, diseñada para manejar series temporales. Los datos se almacenan con marcas de tiempo y se organizan en buckets, facilitando consultas y análisis históricos.



7. Creación del dashboard en Grafana para visualización de datos

InfluxDBCloud permite la implementación de Grafana a través de Grafana Cloud de manera fácil, pudiendo reflejar los datos guardados en gráficas muy detalladas.



8. Modelo compatible con metodología BIM.

A través de un plugin de blender llamado Bonsái, nos permite crear un entorno de trabajo enfocado a la metodología BIM y al desarrollo de estructuras, edificios más detallados y precisos, pudiendo añadir clases IFC, el cuál es el estándar de BIM, para poder trabajar con el modelo en otros softwares dedicados a BIM, como puede ser BIMVision

