



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería de la Salud



## TFG del Grado en Ingeniería de la Salud

### Gemelo digital UCI Documentación Técnica

Presentado por Pablo Peña María  
en Universidad de Burgos

9 de junio de 2025

Tutor: Telmo Miguel Medina



---

# Índice general

---

<b>Índice general</b>	i
<b>Índice de figuras</b>	iii
<b>Índice de tablas</b>	iv
<b>Apéndice A Plan de Proyecto Software</b>	1
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	2
<b>Apéndice B Documentación de usuario</b>	7
B.1. Requisitos software y hardware para ejecutar el proyecto. . . . .	7
B.2. Instalación / Puesta en marcha . . . . .	9
B.3. Manuales y/o Demostraciones prácticas . . . . .	15
<b>Apéndice C Manual del desarrollador / programador / investigador.</b>	19
C.1. Estructura de directorios . . . . .	19
C.2. Compilación, instalación y ejecución del proyecto . . . . .	20
C.3. Pruebas del sistema . . . . .	21
C.4. Instrucciones para la modificación o mejora del proyecto . .	22
<b>Apéndice D Descripción de adquisición y tratamiento de datos</b>	25
D.1. Descripción formal de los datos . . . . .	25
D.2. Descripción clínica de los datos. . . . .	26
<b>Apéndice E Manual de especificación de diseño</b>	27
E.1. Planos . . . . .	27

<b>Apéndice F Especificación de Requisitos</b>	<b>31</b>
F.1. Diagrama de casos de uso . . . . .	31
F.2. Explicación casos de uso . . . . .	31
<b>Apéndice G Estudio experimental</b>	<b>39</b>
G.1. Cuaderno de trabajo . . . . .	39
G.2. Configuración y parametrización de las técnicas . . . . .	39
G.3. Detalle de resultados . . . . .	40
<b>Apéndice H Anexo de sostenibilización curricular</b>	<b>41</b>
H.1. Conclusión . . . . .	43
<b>Bibliografía</b>	<b>45</b>

---

# Índice de figuras

---

B.1. Archivo del proyecto mostrado en el menú de Revit . . . . .	9
B.2. Vista 3D del bloque técnico C . . . . .	10
B.3. Interfaz de Autodesk Revit con el modelo de la UCI cargado y la pestaña “Gestionar” visible. . . . .	10
B.4. Creación de base de datos desde MySQL. . . . .	11
B.5. Verificación de la generación de valores de temperatura y humedad en la base de datos. . . . .	11
B.6. Importación de las bibliotecas necesarias en el script de Dynamo.	12
B.7. Demostración de la ejecución periódica en Dynamo (esquina inferior izquierda). . . . .	13
B.8. Código introducido en MySQL. Parte 1 . . . . .	13
B.9. Código introducido en MySQL. Parte 2. . . . .	14
B.10. Código introducido en MySQL. Parte 3. . . . .	14
B.11. Código introducido en el nodo Python Script. . . . .	15
B.12. Localización del sensor en la vista 3D de la UCI. . . . .	16
B.13. Visualización de los valores de Temperatura y Humedad en las propiedades del sensor. . . . .	17
B.14. Visualización de los valores de Temperatura y Humedad en la vista de planta (2D) de la UCI. . . . .	18
E.1. Plano de la UCI. . . . .	28
E.2. Plano de la UCI con las UTAs incluidas. . . . .	29
E.3. Plano ampliado de la UCI con las UTAs incluidas. . . . .	30
F.1. Diagrama de casos de uso. . . . .	31

---

# Índice de tablas

---

A.1. Planificación temporal del Trabajo Fin de Grado . . . . .	2
A.2. Coste estimado de personal . . . . .	3
A.3. Coste total estimado del proyecto . . . . .	4
B.1. Requisito Funcional 1: Abrir el modelo BIM . . . . .	7
B.2. Requisito Funcional 2: Consultar valores ambientales . . . . .	8
B.3. Requisito Funcional 3: Insertar sensor virtual . . . . .	8
B.4. Requisito Funcional 4: Eliminar sensor . . . . .	8
B.5. Requisito Funcional 5: Crear nuevo parámetro . . . . .	8
B.6. Requisito Funcional 6: Cerrar Revit y guardar cambios . . . . .	8
F.1. CU-1 Abrir el modelo BIM en Revit. . . . .	32
F.2. CU-2 Consultar valores ambientales en el modelo. . . . .	33
F.3. CU-3 Insertar un nuevo sensor en el modelo. . . . .	34
F.4. CU-4 Eliminar sensor del modelo. . . . .	35
F.5. CU-5 Crear un nuevo parámetro para el sensor. . . . .	36
F.6. CU-6 Cerrar el modelo y salir del programa. . . . .	37

## *Apéndice A*

---

# **Plan de Proyecto Software**

---

## **A.1. Introducción**

En este apartado se muestra la planificación temporal para el desarrollo del proyecto y la planificación económica que supone. Además, se recoge la viabilidad legal del mismo.

## A.2. Planificación temporal

Fecha	Tarea realizada	Duración estimada
1 – 10 febrero	Búsqueda y revisión de documentación sobre BIM, Revit, Dynamo, sensores e IoT	~10 días
11 – 20 febrero	Diseño conceptual del sistema: definición del alcance, objetivos y herramientas	~10 días
21 febrero – 10 marzo	Implementación fallida del sistema y análisis de alternativas	~2 semanas
11 – 20 marzo	Investigación sobre conexión con bases de datos	~1 semana
21 marzo – 15 abril	Familiarización con Revit y modelado del sensor y los parámetros	~3 semanas
16 – 25 abril	Configuración de parámetros en Revit, vinculación con Dynamo y lectura desde MySQL	~1 semana
26 abril – 5 mayo	Creación de etiquetas personalizadas y visualización de valores en Revit	~10 días
16 abril – 24 mayo	Redacción de la memoria	~1 mes
26 abril – 28 mayo	Elaboración de anexos	~1 mes
21 – 31 mayo	Revisión final, corrección ortográfica y maquetación en LaTeX	~1 semana

Tabla A.1: Planificación temporal del Trabajo Fin de Grado

## Planificación económica

### Costes de personal

Para estimar el coste asociado al desarrollo del proyecto, se ha considerado la contratación de un ingeniero de la salud titulado, encargado de realizar todas las tareas de diseño, implementación, validación y documentación. Se ha tomado como referencia el salario medio bruto de un ingeniero biomédico en España, estimado en unos 29.000 € anuales [Jobted, 2024]. Prorrateando este importe a una dedicación de tres meses a jornada completa, se calcula un salario bruto mensual de 2.418,33 €.

A este importe se le han aplicado los costes empresariales correspondientes a la Seguridad Social, en función de las bases de cotización del Régimen General [Social, 2023]. El desglose aparece en la Tabla A.2.

Concepto	Coste (€)
Sueldo bruto mensual	2.418,33
Contingencias comunes (23,6 %)	570,72
Contrato temporal (6,7 %)	162,02
FOGASA (0,2 %)	4,83
Formación profesional (0,6 %)	14,51
<b>Total mensual</b>	<b>3.170,41</b>
<b>Total para 3 meses</b>	<b>9.511,23</b>

Tabla A.2: Coste estimado de personal

### Costes de software

Si el proyecto es llevado a cabo por un estudiante, Autodesk ofrece una licencia temporal gratuita para usar Revit, y en ese caso todas las licencias serían gratuitas. Suponiendo que el proyecto se realiza en un entorno profesional y no se dispone de esta ventaja, este sería el software necesario:

- **Autodesk Revit 2025:** licencia comercial individual anual estimada en 3.150 € [Autodesk, 2025].
- Dynamo (incluido en Revit).
- MySQL Workbench.
- Python 3 y paquetes `mysql.connector`, `clr`, entre otros.
- Overleaf (versión gratuita).
- GitHub y GitHub Desktop.
- ChatGPT (versión gratuita para generación de scripts y apoyo técnico).

### Costes de hardware

El desarrollo del TFG se ha realizado íntegramente en un entorno simulado, por lo que no ha sido necesario adquirir sensores reales ni otros componentes físicos. El único hardware indispensable ha sido un ordenador personal, que se amortiza de forma proporcional al uso durante el proyecto.

Asumiendo un precio de compra de 900 € y una vida útil de 6 años, se ha empleado la fórmula de amortización lineal:

$$\text{Amortización}_{3 \text{ meses}} = \frac{900 \times 3}{12 \times 6} = 37,50 \text{ £}$$

### Resumen de costes totales

Elemento	Coste (€)
Coste de personal (3 meses)	9.511,23
Amortización equipo informático	37,50
Licencia Revit comercial	3.150,00
<b>Total estimado</b>	<b>12.698,73</b>

Tabla A.3: Coste total estimado del proyecto

### Viabilidad legal

El desarrollo del presente proyecto ha requerido el uso de diversas herramientas software. En el contexto profesional en el que se enmarca, todas las licencias comerciales se han gestionado conforme a sus condiciones de uso:

- **Autodesk Revit 2025:** empleado con licencia comercial individual, apta para entornos profesionales no académicos [Autodesk, 2025].
- **Dynamo:** herramienta de código abierto incluida en Revit.
- **MySQL Workbench:** distribuido bajo licencia pública general (GPL).
- **Python:** software libre con licencia Python Software Foundation (PSF).
- **Overleaf, GitHub y ChatGPT:** empleados en sus versiones gratuitas con fines de documentación y asistencia técnica.

### Protección de datos

El sistema implementado no gestiona información personal ni sensible, ya que los datos de temperatura y humedad han sido generados artificialmente. No obstante, se han seguido buenas prácticas en cuanto al tratamiento de datos, previendo que una futura integración con sistemas reales exigiría el cumplimiento de la legislación vigente en materia de protección de datos, concretamente la Ley Orgánica 3/2018 (LOPDGDD) y el Reglamento (UE) 2016/679 (RGPD) [de España, 2018, y Consejo de la Unión Europea, 2016].

**Cumplimiento normativo**

El proyecto cumple con la legislación vigente en materia de propiedad intelectual, licencias de software y tratamiento de datos. Además, todas las fuentes de información utilizadas se han citado adecuadamente, respetando la normativa académica de integridad y honestidad científica.



## *Apéndice B*

---

# **Documentación de usuario**

---

## **B.1. Requisitos software y hardware para ejecutar el proyecto.**

Con el objetivo de garantizar la correcta ejecución del sistema desarrollado, se detallan a continuación los requisitos mínimos recomendados tanto a nivel de software como de hardware:

### **Requisitos funcionales**

RF-01	Abrir el modelo BIM
Descripción	El sistema debe permitir al usuario abrir el archivo del modelo BIM en Autodesk Revit para acceder al entorno del gemelo digital.
Importancia	Alta. Es imprescindible para iniciar el trabajo de visualización y supervisión ambiental.
Prioridad	Alta

Tabla B.1: Requisito Funcional 1: Abrir el modelo BIM

<b>RF-02</b>	<b>Consultar valores ambientales</b>
Descripción	El sistema debe permitir al usuario visualizar los valores actuales de temperatura y humedad integrados en el modelo BIM.
Importancia	Alta. Es el objetivo principal del sistema de gemelo digital: proporcionar datos ambientales actualizados de forma visual.
Prioridad	Alta

Tabla B.2: Requisito Funcional 2: Consultar valores ambientales

<b>RF-03</b>	<b>Insertar sensor virtual</b>
Descripción	El sistema debe permitir al usuario insertar nuevas instancias de sensores virtuales en el modelo BIM.
Importancia	Media. Permite ampliar el sistema a nuevas zonas del entorno hospitalario o planificar futuras implementaciones.
Prioridad	Media

Tabla B.3: Requisito Funcional 3: Insertar sensor virtual

<b>RF-04</b>	<b>Eliminar sensor</b>
Descripción	El sistema debe permitir al usuario eliminar sensores previamente insertados del modelo BIM.
Importancia	Media. Facilita la gestión del modelo eliminando elementos que ya no son necesarios o representativos.
Prioridad	Media

Tabla B.4: Requisito Funcional 4: Eliminar sensor

<b>RF-05</b>	<b>Crear nuevo parámetro</b>
Descripción	El sistema debe permitir crear y asignar nuevos parámetros personalizados a las familias de sensores virtuales.
Importancia	Media. Permite ampliar el modelo a nuevas variables ambientales según evolucione el sistema.
Prioridad	Media

Tabla B.5: Requisito Funcional 5: Crear nuevo parámetro

<b>RF-06</b>	<b>Cerrar Revit y guardar cambios</b>
Descripción	El sistema debe permitir cerrar el modelo y guardar los cambios realizados por el usuario.
Importancia	Baja. Es una funcionalidad básica para conservar el trabajo hecho durante la sesión.
Prioridad	Baja

Tabla B.6: Requisito Funcional 6: Cerrar Revit y guardar cambios

## B.2. Instalación / Puesta en marcha

Para ejecutar correctamente el sistema descrito en este proyecto, es necesario seguir los siguientes pasos de instalación y configuración inicial:

### 1. Instalación del entorno BIM

1. Descargar e instalar **Autodesk Revit 2025** desde el sitio oficial de Autodesk: <https://www.autodesk.es/products/revit/overview>.
2. Abrir el modelo BIM correspondiente al proyecto (archivo BT-C-1.rvt) que contiene la representación de la UCI.



Figura B.1: Archivo del proyecto mostrado en el menú de Revit.

Por defecto, se abre la vista 3D "BTC 3D", en la cual vemos la planta -1, correspondiente a la UCI. Esta proviene de un recorte de sección de una vista original y más amplia en la que se representa un bloque técnico entero. Para visualizar esta segunda vista, es necesario abrir el Navegador de Proyectos y en la pestaña **Vistas 3D** hacer doble click en "**{3D}**".

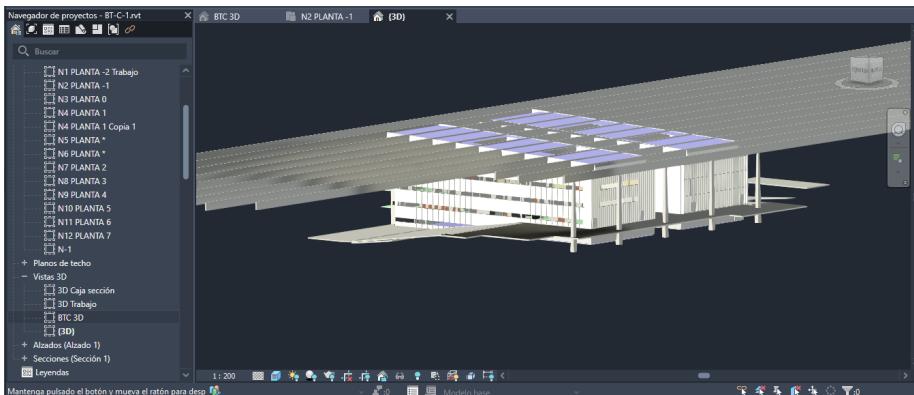


Figura B.2: Vista 3D del bloque técnico C.

3. Verificar que **Dynamo** está disponible e integrado en Revit (normalmente se encuentra en la pestaña “Gestionar”).

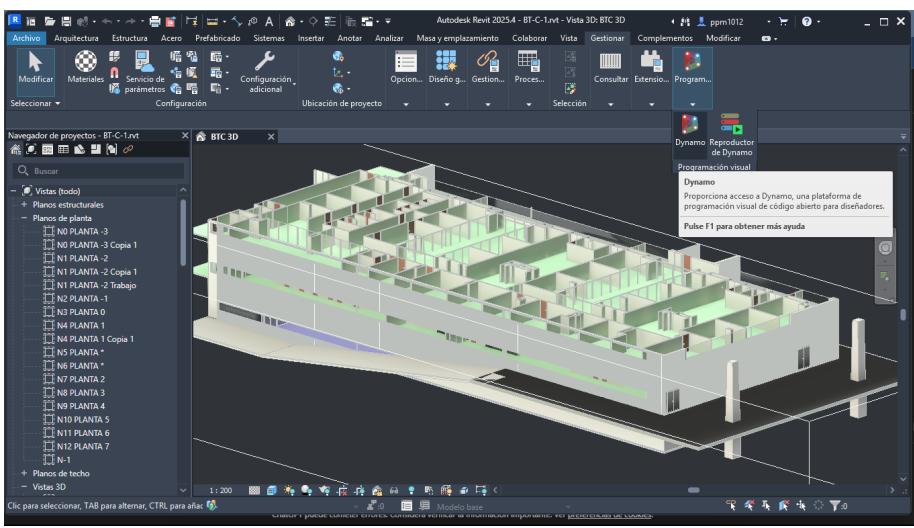


Figura B.3: Interfaz de Autodesk Revit con el modelo de la UCI cargado y la pestaña “Gestionar” visible.

## 2. Instalación de base de datos y generación de datos

1. Instalar MySQL Server y MySQL Workbench desde <https://dev.mysql.com/downloads/>.
2. Crear la base de datos utilizando el script SQL proporcionado en el anexo técnico del proyecto.

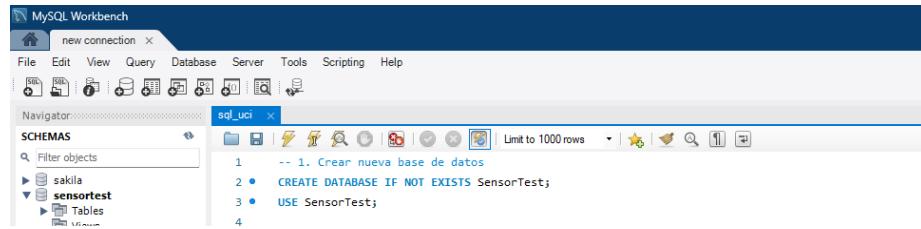


Figura B.4: Creación de base de datos desde MySQL.

3. Asegurarse de que el servicio MySQL está en ejecución.
4. Verificar que los valores de temperatura y humedad se generan automáticamente a través del procedimiento almacenado y el evento programado.

The screenshot shows the 'Result Grid' view in MySQL Workbench. The grid displays data from a table with columns: ID, SensorID, Timestamp, Temperatura, and Humedad. The data is as follows:

	ID	SensorID	Timestamp	Temperatura	Humedad
▶	237	1	2025-05-15 16:22:19	23.40	49.05
	236	1	2025-05-15 16:22:15	22.82	54.12
	235	1	2025-05-15 16:22:10	22.24	44.73
	234	1	2025-05-15 16:22:05	22.62	38.02
	233	1	2025-05-15 16:22:00	22.74	58.35

Below the grid, it says 'LecturasAmbiente 5 x Result 6'.

Figura B.5: Verificación de la generación de valores de temperatura y humedad en la base de datos.

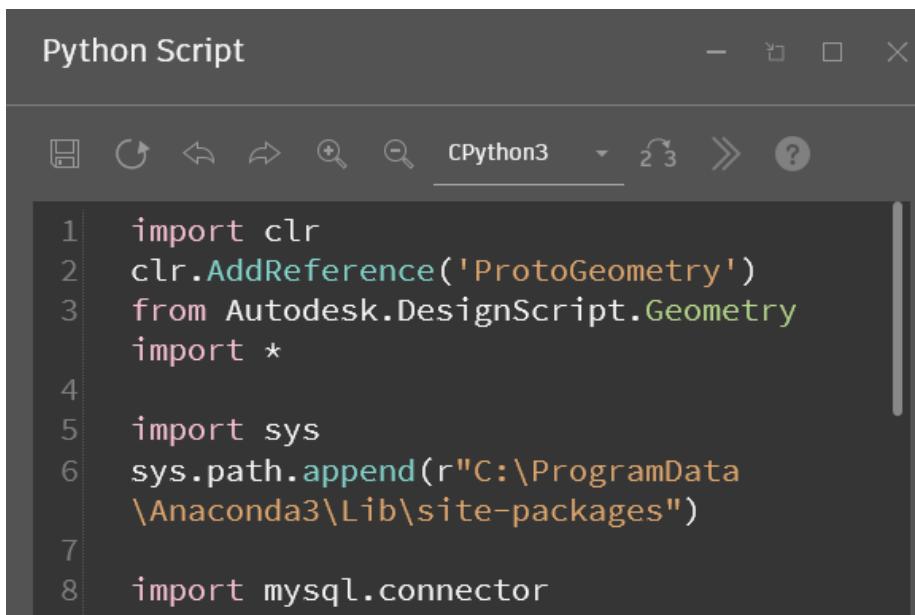
## 3. Instalación de entorno Python

1. Descargar e instalar Python 3.10 o superior desde <https://www.python.org>.

2. Instalar la biblioteca de conexión con MySQL mediante el siguiente comando:

```
pip install mysql-connector-python
```

3. Verificar que el intérprete de Python tiene acceso a las bibliotecas necesarias, incluyendo `clr` si se ejecuta desde el entorno de Dynamo.
4. Si se emplea Anaconda, añadir la ruta de sus paquetes a `sys.path` dentro del script Python en Dynamo.



The screenshot shows the 'Python Script' interface in Dynamo. The title bar says 'Python Script'. The toolbar includes icons for file operations, search, and help. A dropdown menu shows 'CPython3'. The main code area has the following content:

```

1 import clr
2 clr.AddReference('ProtoGeometry')
3 from Autodesk.DesignScript.Geometry
4 import *
5
6 import sys
7 sys.path.append(r"C:\ProgramData\Anaconda3\Lib\site-packages")
8 import mysql.connector

```

Figura B.6: Importación de las bibliotecas necesarias en el script de Dynamo.

#### 4. Conexión entre base de datos y modelo BIM

1. Abrir el archivo .dyn de Dynamo suministrado.
2. Editar las credenciales de conexión a la base de datos (host, usuario, contraseña) si fuera necesario.
3. Ejecutar el script en modo periódico (por ejemplo, cada 5 segundos) para actualizar automáticamente los parámetros ambientales del modelo.

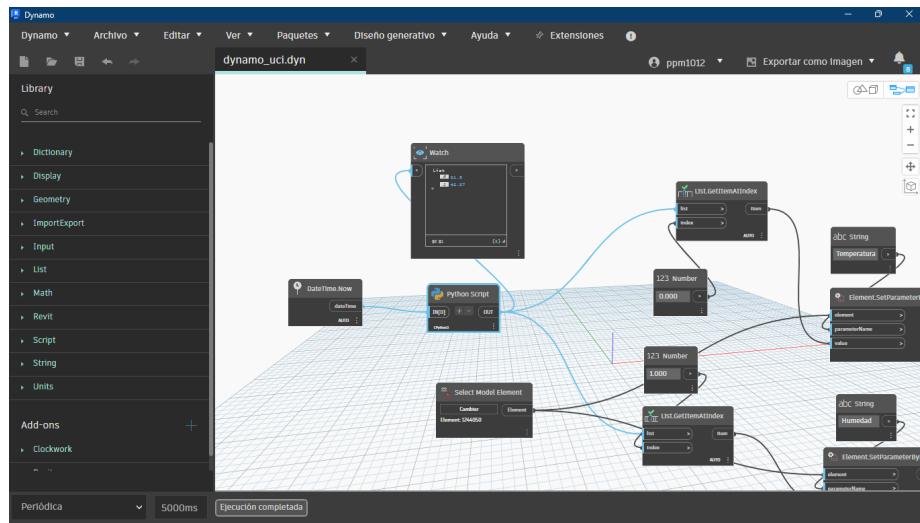


Figura B.7: Demostración de la ejecución periódica en Dynamo (esquina inferior izquierda).

Una vez completados estos pasos, el sistema estará plenamente operativo. El modelo de Revit mostrará en tiempo real los datos ambientales que se van generando y almacenando en la base de datos.

El código desarrollado para crear y almacenar los datos en MySQL es el siguiente:

```

1      -- 1. Crear nueva base de datos
2 •  CREATE DATABASE IF NOT EXISTS SensorTest;
3 •  USE SensorTest;
4
5      -- 2. Crear tabla nueva
6 •  CREATE TABLE IF NOT EXISTS LecturasAmbiente (
7          ID INT AUTO_INCREMENT PRIMARY KEY,
8          SensorID INT NOT NULL DEFAULT 1,
9          Timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
10         Temperatura DECIMAL(5,2),
11         Humedad DECIMAL(5,2)
12     );

```

Figura B.8: Código introducido en MySQL. Parte 1

```

14      -- 3. Crear procedimiento almacenado con generación robusta
15  DELIMITER $$ 
16 •  DROP PROCEDURE IF EXISTS InsertarLectura;
17  CREATE PROCEDURE InsertarLectura()
18  BEGIN
19      -- declarar como DOUBLE para que RAND() funcione correctamente
20      DECLARE temp DOUBLE;
21      DECLARE hum DOUBLE;
22
23      -- generar valores directamente como DOUBLE
24      SET temp = ROUND(21 + (RAND() * (24 - 21)), 2);
25      SET hum = ROUND(30 + (RAND() * (60 - 30)), 2);
26
27      INSERT INTO LecturasAmbiente (SensorID, Temperatura, Humedad)
28          VALUES (1, temp, hum);
30  END;
31  $$ 
32  DELIMITER ;
33

```

Figura B.9: Código introducido en MySQL. Parte 2.

```

35 •  -- 4. Activar el event scheduler (si aún no está activo)
36  SET GLOBAL event_scheduler = ON;
37
38      -- 5. Crear evento programado que ejecuta el procedimiento cada 5 segundos
39 •  DROP EVENT IF EXISTS InsertarLecturaCada5s;
40
41 •  CREATE EVENT InsertarLecturaCada5s
42      ON SCHEDULE EVERY 5 SECOND
43      STARTS CURRENT_TIMESTAMP
44      DO CALL InsertarLectura();
45
46      -- 6. Verifica si la tabla recibe datos
47      -- Espera unos segundos y ejecuta:
48
49
50 •  SELECT * FROM LecturasAmbiente ORDER BY ID DESC;
51 •  SHOW EVENTS FROM SensorTest;
52
53      -- descomentar la siguiente línea para parar
54      -- ALTER EVENT InsertarLecturaCada5s DISABLE;

```

Figura B.10: Código introducido en MySQL. Parte 3.

Y el código incluido en el nodo Python Script de Dynamo es:

```

1 import clr
2 clr.AddReference('ProtoGeometry')
3 from Autodesk.DesignScript.Geometry import *
4
5 import sys
6 sys.path.append(r"C:\ProgramData\Anaconda3\Lib\site-packages")
7
8 import mysql.connector
9
10 # Conexión a la base de datos
11 conn = mysql.connector.connect(
12     host="127.0.0.1",
13     user="root",
14     password="contraseña",
15     database="sensortest"
16 )
17 cursor = conn.cursor()
18
19 # Leer Último registro
20 cursor.execute("SELECT Temperatura, Humedad FROM lecturasambiente ORDER BY ID DESC LIMIT 1")
21 resultado = cursor.fetchone()
22
23 if resultado:
24     temperatura = float(resultado[0])
25     humedad = float(resultado[1])
26 else:
27     temperatura = None
28     humedad = None
29
30 cursor.close()
31 conn.close()
32
33 OUT = temperatura, humedad
34

```

Figura B.11: Código introducido en el nodo Python Script.

### B.3. Manuales y/o Demostraciones prácticas

Una vez realizados todos los pasos descritos en el apartado anterior de instalación y puesta en marcha, el sistema queda completamente preparado para su funcionamiento.

Para comprobar que el flujo de datos funciona correctamente, basta con abrir el modelo BIM en Autodesk Revit y verificar que los valores de los sensores simulados se están actualizando en tiempo real. Para ello, se deben seguir los siguientes pasos:

1. Abrir el modelo de la UCI en Revit.
2. Acceder a Dynamo desde la pestaña “Gestionar” y asegurarse de que el archivo .dyn correspondiente está ejecutándose en modo periódico.

3. Seleccionar el sensor insertado en la vista 3D. En la siguiente imagen se rodea de amarillo.



Figura B.12: Localización del sensor en la vista 3D de la UCI.

4. Comprobar en las propiedades (presionando dos veces la tecla "P") del ejemplar que los campos Temperatura y Humedad muestran valores dinámicos que se actualizan cada pocos segundos.

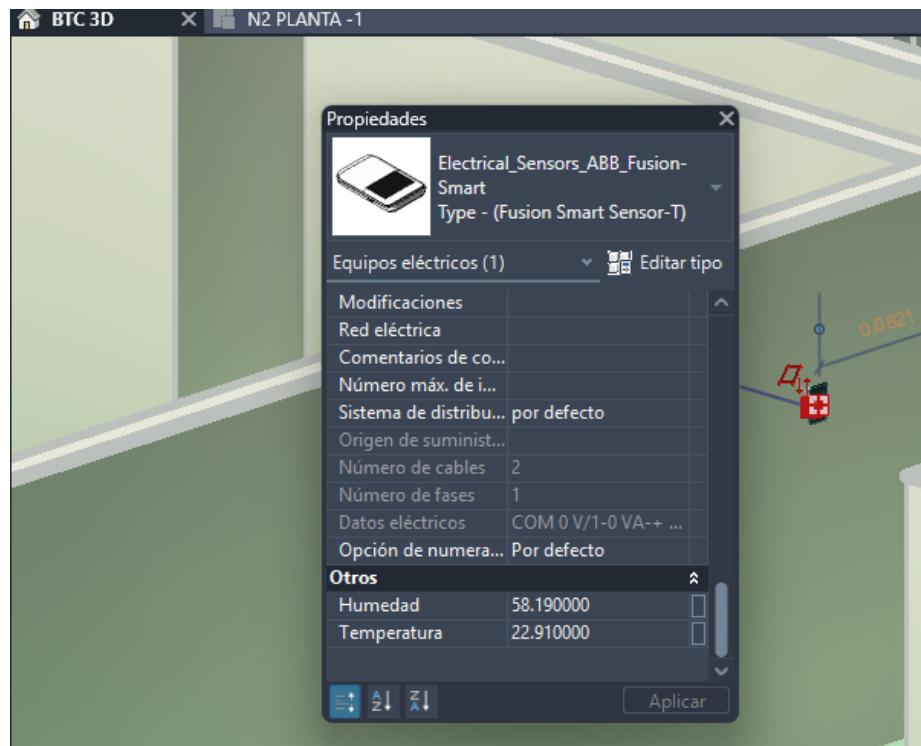


Figura B.13: Visualización de los valores de Temperatura y Humedad en las propiedades del sensor.

5. También se pueden visionar estos valores de forma más visual abriendo en Revit desde el Navegador de Proyectos > Vistas > Planos de planta > N2 PLANTA -1.

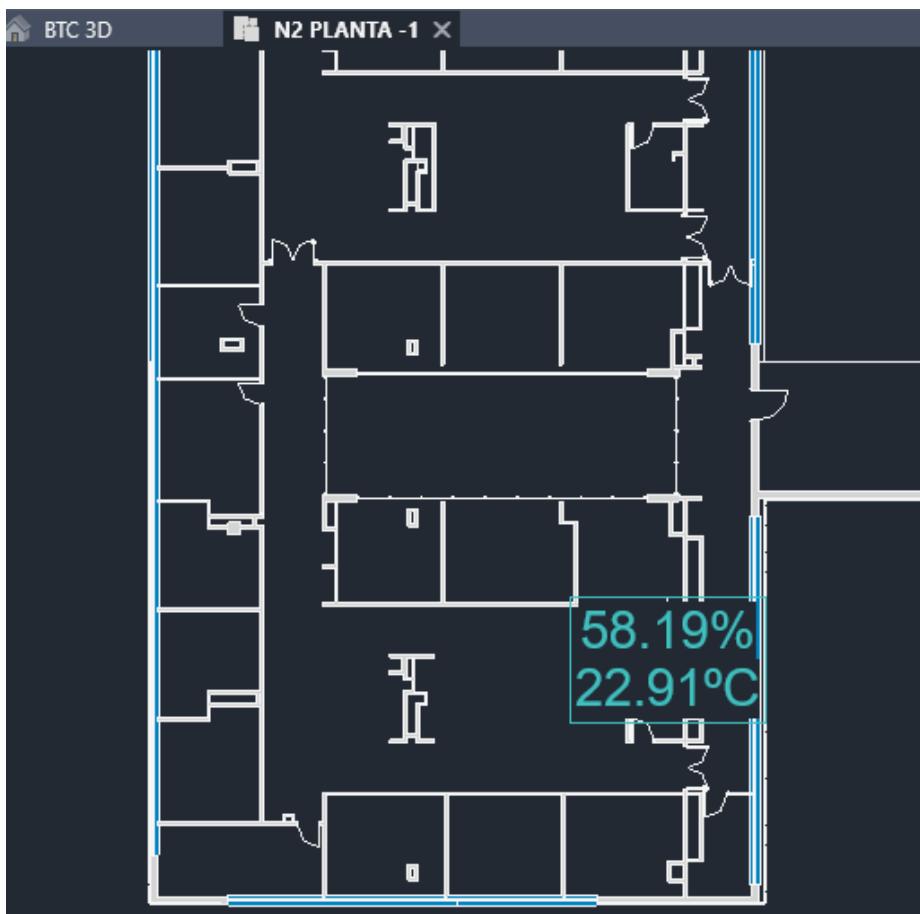


Figura B.14: Visualización de los valores de Temperatura y Humedad en la vista de planta (2D) de la UCI.

Esta sencilla comprobación confirma que el sistema está leyendo los datos generados en la base de datos MySQL y que estos se integran correctamente en el modelo BIM. La visualización directa en Revit valida así el correcto funcionamiento del flujo completo del gemelo digital.

## *Apéndice C*

---

# **Manual del desarrollador / programador / investigador.**

---

## **C.1. Estructura de directorios**

Los ficheros que componen este trabajo se encuentran en el [repositorio de GitHub](#). El contenido de cada uno de ellos se detalla a continuación:

- **Carpeta demostracion:** carpeta que contiene un vídeo de demostración del funcionamiento del prototipo completo.
  - **video\_demostracion.mp4:** grabación de pantalla donde se observa la ejecución coordinada de MySQL, Dynamo y Revit, así como la visualización correcta de los valores de humedad y temperatura.
- **Carpeta img:** carpeta que incluye todas las imágenes utilizadas en la memoria y en los anexos.
- **Carpeta memoria\_y\_anexos:** carpeta que contiene los documentos PDF finales de la memoria y de los anexos.
  - **memoria.pdf:** documento PDF que contiene la memoria completa del Trabajo de Fin de Grado.
  - **anexos.pdf:** documento PDF que contiene todos los anexos complementarios a la memoria.
- **Carpeta tex:** carpeta que incluye todos los archivos fuente en formato LaTeX correspondientes a los apartados de la memoria y los anexos.

- **codigo\_python.py**: script en lenguaje Python utilizado en el nodo *Python Script* de Dynamo para conectarse a la base de datos y recuperar los datos ambientales.
- **sql\_uci.sql**: archivo SQL que contiene la creación de la base de datos y tabla, así como el evento programado en MySQL para la generación automática de datos simulados de temperatura y humedad.
- **README.md**: archivo de presentación del repositorio en GitHub.
- **anexos.tex**: archivo LaTeX que contiene la estructura principal del documento de los anexos.
- **memoria.tex**: archivo LaTeX que contiene la estructura principal del documento de la memoria.
- **bibliografia.bib**: archivo BibTeX que recoge la bibliografía utilizada en la memoria.
- **bibliografiaAnexos.bib**: archivo BibTeX que recoge la bibliografía utilizada en los anexos.
- **BT-C-1.rvt**: archivo del modelo Revit de la UCI empleado para el proyecto. Debido a su gran tamaño, no se encuentra en el repositorio de GitHub, pero se incluye en la copia digital entregada del Trabajo de Fin de Grado.

## C.2. Compilación, instalación y ejecución del proyecto

Este apartado no requiere instrucciones adicionales a las ya descritas en el apartado de instalación y puesta en marcha. El proyecto no necesita ser compilado, ya que está formado por una integración de herramientas que funcionan mediante scripts interpretados y conexión entre sistemas.

Una vez ejecutadas las fases de instalación y configuración inicial, el sistema queda completamente operativo. A partir de ese momento, el único requisito para los usuarios a los que está destinado este proyecto (como personal de mantenimiento o sanitario) es acceder al modelo BIM en Autodesk Revit y visualizar los parámetros establecidos asociados a los sensores simulados.

Por tanto, desde el punto de vista del usuario final, el funcionamiento del sistema es transparente y no requiere conocimientos técnicos avanzados más allá del uso básico de Revit para la navegación y consulta de propiedades de los elementos.

## C.3. Pruebas del sistema

Las pruebas se han centrado en cuatro aspectos principales:

### 1. Verificación de generación de datos

Se comprobó que el procedimiento almacenado en MySQL genera correctamente valores de temperatura y humedad dentro del rango esperado (21–24 °C para temperatura, 30–60 % para humedad). También se validó que el evento programado ejecuta dicho procedimiento automáticamente cada cinco segundos, insertando registros nuevos en la tabla.

### 2. Conexión entre Python y la base de datos

A través del script incluido en Dynamo, se ejecutó la conexión a la base de datos usando el módulo `mysql.connector`. Se validó que los datos se extraen correctamente en cada ejecución periódica, sin interrupciones ni errores de conexión, siempre que el servidor esté activo.

### 3. Actualización de parámetros en Dynamo y Revit

Se verificó que los valores leídos desde la base de datos son tratados correctamente por el script Python y que se asignan sin errores a los parámetros de temperatura y humedad de cada sensor en el modelo BIM. El nodo `Watch` de Dynamo confirmó la correcta recepción de los datos, y Revit reflejó estos valores automáticamente.

### 4. Robustez del sistema ante interrupciones

Se probó la interrupción manual del servidor de base de datos durante la ejecución. Se comprobó que el sistema no se bloquea, sino que deja de actualizar los valores hasta que se restablece la conexión. Esta prueba valida que el comportamiento es estable y predecible incluso en situaciones imprevistas.

Los resultados de estas pruebas demuestran que el sistema cumple con su objetivo funcional: representar en tiempo real, de forma simulada, los parámetros ambientales de una UCI hospitalaria en un entorno BIM.

## C.4. Instrucciones para la modificación o mejora del proyecto

Este proyecto ha sido diseñado con un enfoque modular y extensible, lo que permite su adaptación o ampliación futura tanto a nivel técnico como funcional. A continuación se describen algunas instrucciones y recomendaciones para facilitar su mejora:

### 1. Sustitución de datos simulados por sensores reales

Una de las principales mejoras posibles consiste en reemplazar la generación artificial de datos mediante eventos en MySQL por la lectura en tiempo real de sensores IoT físicos. Para ello se recomienda:

- Utilizar sensores compatibles con microcontroladores como ESP32 o Raspberry Pi.
- Establecer la comunicación mediante protocolo MQTT o HTTP, conectando los datos a un servidor intermedio que actualice la base de datos.
- Adaptar el script Python en Dynamo para leer datos de dicho servidor o directamente desde MQTT si se emplea una solución integrada.

### 2. Mejora de la interfaz y visualización

Para facilitar el uso por parte de personal sanitario o técnico no especializado, se podrían incorporar mejoras como:

- Automatización del arranque del sistema (por ejemplo, mediante scripts .bat que inicien todos los servicios).
- Incorporación de gráficos o indicadores visuales dentro del modelo BIM (por ejemplo, colores asociados a rangos de valores).
- Desarrollo de una interfaz web conectada a la base de datos, que permita consultar el estado ambiental sin necesidad de abrir Revit.

### **3. Seguridad y acceso a datos**

En una futura implementación real, sería necesario:

- Añadir cifrado en la comunicación entre sensores y servidor (por ejemplo, MQTT sobre TLS).
- Proteger la base de datos con usuarios específicos y control de accesos.
- Considerar la integración con normativas de seguridad como RGPD o LOPDGDD.

Estas recomendaciones permiten adaptar el sistema a escenarios reales, ampliar su funcionalidad y facilitar su reutilización en futuros trabajos dentro del ámbito de la ingeniería de la salud.



## *Apéndice D*

---

# **Descripción de adquisición y tratamiento de datos**

---

## **D.1. Descripción formal de los datos**

Los datos utilizados corresponden a registros simulados de parámetros ambientales, concretamente temperatura y humedad relativa. Estos datos se almacenan en una base de datos relacional MySQL en una tabla denominada **LecturasAmbiente**.

Cada registro de la tabla representa una lectura de un sensor ambiental y contiene los siguientes campos:

**ID:** Identificador único de la lectura (entero, clave primaria, autoincremental).

**SensorID:** Identificador del sensor correspondiente (entero, no nulo, valor por defecto 1).

**Timestamp:** Marca temporal que indica el momento de la lectura (tipo DATETIME, con valor por defecto igual a la fecha y hora actual).

**Temperatura:** Valor de la temperatura en grados Celsius (número decimal con precisión de hasta dos cifras decimales).

**Humedad:** Valor de la humedad relativa en porcentaje (número decimal con precisión de hasta dos cifras decimales).

Los datos se generan de forma automática mediante un procedimiento que simula el funcionamiento de sensores físicos reales. A intervalos regulares, se insertan nuevas filas en la tabla con valores aleatorios dentro de rangos

coherentes con condiciones ambientales habituales en entornos hospitalarios. Este procedimiento permite emular un flujo continuo de información, similar al de un sistema IoT operativo.

## D.2. Descripción clínica de los datos.

Desde un punto de vista clínico, el control de la temperatura ambiental es fundamental para garantizar el confort térmico de los pacientes, especialmente aquellos en estados críticos con capacidad termorreguladora reducida. En entornos hospitalarios, se recomienda mantener la temperatura en rangos que oscilen entre 21 y 24 °C, dependiendo de las características específicas de cada zona. En una UCI, mantener una temperatura adecuada puede prevenir episodios de hipotermia o hipertermia, contribuyendo al bienestar y a una recuperación más eficaz.

Por otro lado, la humedad relativa también juega un papel esencial. Niveles de humedad demasiado bajos pueden provocar sequedad de mucosas, irritación de las vías respiratorias y aumentar el riesgo de infecciones nosocomiales, mientras que una humedad excesiva favorece la proliferación de hongos, bacterias y virus en el ambiente. Por ello, los valores clínicamente recomendados para una UCI se sitúan entre 30 % y 60 %, buscando un equilibrio que minimice los riesgos tanto para pacientes como para el material médico sensible.

Además, un entorno ambiental controlado permite la correcta conservación de equipamiento médico y reduce el riesgo de fallo en dispositivos electrónicos críticos. Por ello, las unidades de tratamiento de aire integradas en el sistema HVAC (calefacción, ventilación y aire acondicionado) del hospital suelen incluir sensores de temperatura y humedad, así como sistemas de filtración y renovación de aire, que deben funcionar de forma continua y eficiente.

## *Apéndice E*

---

# **Manual de especificación de diseño**

---

## **E.1. Planos**

Este primer plano se corresponde con la planta de la UCI. En el BOX 1 (número de local 488) es donde hemos insertado el sensor dentro de Revit, ya que como vamos a ver en los siguientes planos, es donde se encuentra la UTA que hemos cogido como ejemplo (UTA-20, marcada en amarillo su localización).

El segundo plano muestra la distribución de las UTAs y conductos de ventilación. Como es complicado reconocer a cual nos referimos, en el tercer plano se amplia el plano y se redondea la UTA-20 en amarillo.

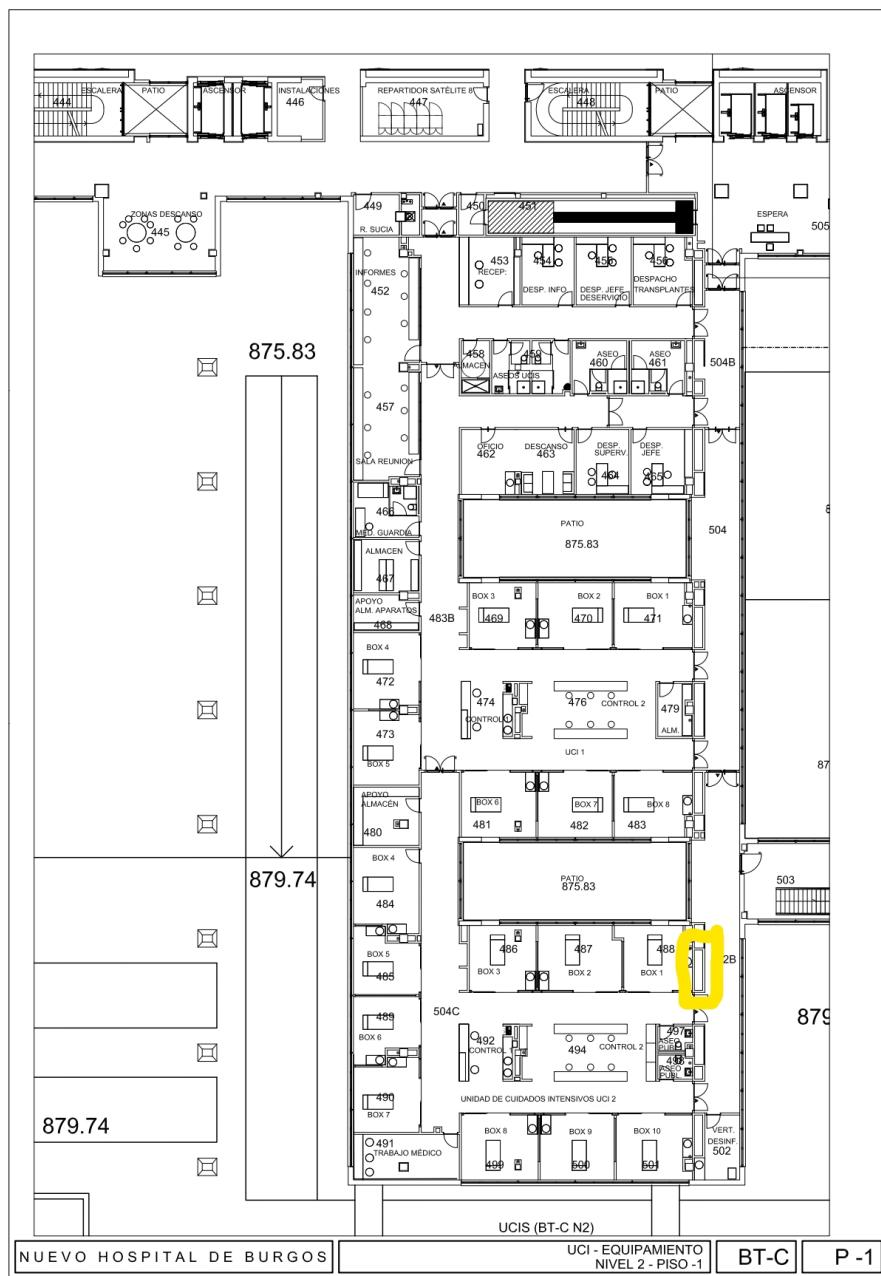


Figura E.1: Plano de la UCI.

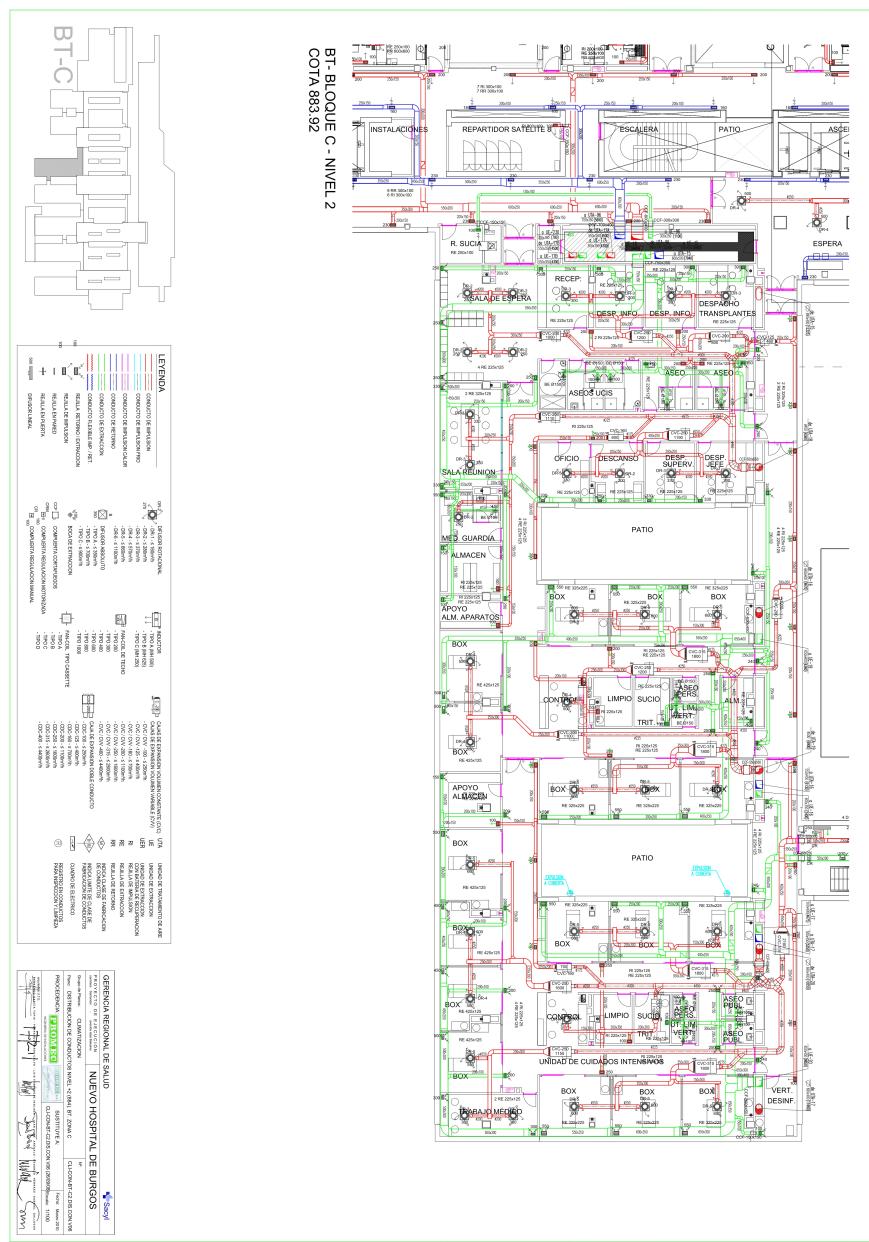


Figura E.2: Plano de la UCI con las UTAs incluidas.

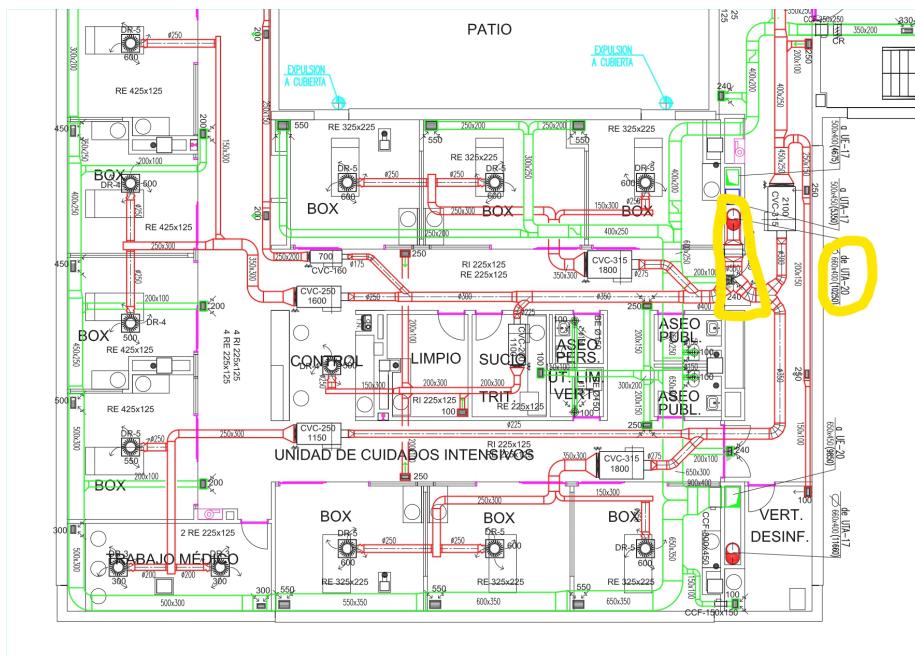


Figura E.3: Plano ampliado de la UCI con las UTAs incluidas.

## *Apéndice F*

---

# Especificación de Requisitos

---

## F.1. Diagrama de casos de uso

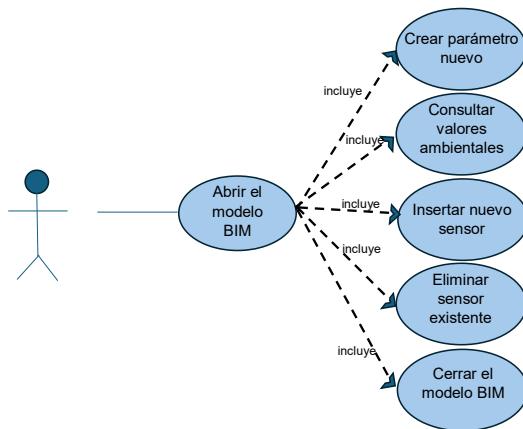


Figura F.1: Diagrama de casos de uso.

## F.2. Explicación casos de uso

<b>CU-1</b>	Abrir el modelo BIM en Revit
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-01
<b>Descripción</b>	El usuario abre Autodesk Revit y carga el archivo del modelo correspondiente a la UCI.
<b>Precondición</b>	Autodesk Revit debe estar instalado en el equipo. El archivo del modelo debe estar disponible localmente o en red.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Iniciar Autodesk Revit.</li> <li>2. Seleccionar y abrir el archivo del modelo.</li> </ol>
<b>Postcondición</b>	El modelo está cargado y listo para su consulta o edición.
<b>Excepciones</b>	El archivo no se encuentra, está corrupto o no es compatible con la versión de Revit.
<b>Importancia</b>	Alta

Tabla F.1: CU-1 Abrir el modelo BIM en Revit.

<b>CU-2</b>	Consultar valores ambientales en el modelo
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-02
<b>Descripción</b>	El usuario navega por el modelo y consulta los valores de temperatura y humedad asignados a los sensores virtuales.
<b>Precondición</b>	El modelo debe estar cargado y los datos deben estarse actualizando desde la base de datos.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Navegar por el modelo en planta o vista 3D.</li> <li>2. Seleccionar un sensor y consultar sus propiedades.</li> </ol>
<b>Postcondición</b>	El usuario visualiza los últimos valores ambientales almacenados.
<b>Excepciones</b>	El sensor no está correctamente vinculado o los datos no se están actualizando.
<b>Importancia</b>	Alta

Tabla F.2: CU-2 Consultar valores ambientales en el modelo.

<b>CU-3</b>	Insertar un nuevo sensor en el modelo
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-03
<b>Descripción</b>	El usuario inserta una nueva instancia de la familia de sensores en una ubicación deseada del modelo BIM.
<b>Precondición</b>	El modelo debe estar abierto y contener la familia de sensores cargada.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Activar la herramienta de colocación de componentes.</li> <li>2. Seleccionar la familia de sensor.</li> <li>3. Colocar el sensor en la ubicación deseada.</li> </ol>
<b>Postcondición</b>	El modelo contiene un nuevo sensor listo para recibir datos.
<b>Excepciones</b>	La familia de sensores no está disponible o no se puede colocar en esa vista.
<b>Importancia</b>	Media

Tabla F.3: CU-3 Insertar un nuevo sensor en el modelo.

<b>CU-4</b>	Eliminar sensor del modelo
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-04
<b>Descripción</b>	El usuario elimina una instancia de sensor del modelo cuando ya no es necesaria.
<b>Precondición</b>	El modelo debe estar abierto y editable.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Seleccionar el sensor a eliminar.</li> <li>2. Pulsar la tecla Supr o usar el botón de eliminar.</li> </ol>
<b>Postcondición</b>	El sensor ha sido eliminado del modelo.
<b>Excepciones</b>	El sensor está bloqueado o no se tiene permiso de edición.
<b>Importancia</b>	Media

Tabla F.4: CU-4 Eliminar sensor del modelo.

<b>CU-5</b>	Crear un nuevo parámetro para el sensor
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-05
<b>Descripción</b>	El usuario crea un nuevo parámetro compartido para añadir una nueva variable ambiental al sensor.
<b>Precondición</b>	El usuario debe tener acceso a los parámetros compartidos y a la edición de familias.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Acceder a "Gestionar» "Parámetros compartidos".</li> <li>2. Crear un nuevo parámetro (por ejemplo, CO2).</li> <li>3. Asignarlo a la familia de sensores.</li> </ol>
<b>Postcondición</b>	El sensor contiene un nuevo parámetro visible en Revit.
<b>Excepciones</b>	No se encuentra el archivo de parámetros compartidos o no se tiene acceso de edición.
<b>Importancia</b>	Media

Tabla F.5: CU-5 Crear un nuevo parámetro para el sensor.

<b>CU-6</b>	Cerrar el modelo y salir del programa
<b>Versión</b>	1.0
<b>Autor</b>	Pablo Peña María
<b>Requisitos asociados</b>	RF-06
<b>Descripción</b>	El usuario guarda los cambios realizados y cierra Autodesk Revit.
<b>Precondición</b>	El modelo debe estar abierto y el usuario debe tener permisos de guardado.
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. Guardar el modelo.</li> <li>2. Cerrar el archivo.</li> <li>3. Salir de Revit.</li> </ol>
<b>Postcondición</b>	El programa se cierra y los cambios se conservan.
<b>Excepciones</b>	El archivo está protegido o hay conflictos al guardar.
<b>Importancia</b>	Baja

Tabla F.6: CU-6 Cerrar el modelo y salir del programa.



## *Apéndice G*

---

# **Estudio experimental**

---

## **G.1. Cuaderno de trabajo**

Durante el desarrollo del sistema se probaron diferentes métodos y herramientas hasta llegar a la solución final. A continuación se enumeran algunas decisiones clave y configuraciones descartadas:

- Se valoró inicialmente la exportación del modelo a Unity para su integración con sensores, pero la pérdida de calidad visual al convertir archivos .fbx motivó su abandono.
- Se probó el uso de Excel con macros para generar datos simulados, pero la dificultad de sincronización con Dynamo y la falta de robustez justificaron la migración hacia una base de datos.
- Se experimentó con la conexión directa desde Dynamo a MQTT, pero se optó por una solución más estable mediante lectura desde una base de datos MySQL.

## **G.2. Configuración y parametrización de las técnicas**

La arquitectura final se configuró con los siguientes parámetros principales:

- Base de datos MySQL con evento que genera un nuevo valor cada 5 segundos.

- Rango simulado: temperatura entre 21–24 °C y humedad entre 30–60 %.
- Script Python embebido en Dynamo con conexión periódica a MySQL.
- Ejecución de Dynamo cada 5 segundos en modo periódico.

### **G.3. Detalle de resultados**

Aunque el sistema no fue sometido a validaciones estadísticas ni pruebas con usuarios, se verificó funcionalmente que los valores generados se almacenan correctamente en la base de datos y son reflejados automáticamente en el modelo BIM. Estas pruebas han servido como validación técnica del flujo de datos y del comportamiento esperado del sistema de gemelo digital.

## *Apéndice H*

---

# **Anexo de sostenibilización curricular**

---

Durante el desarrollo de este Trabajo de Fin de Grado he tratado de integrar conocimientos técnicos con una perspectiva más amplia, que incluye los principios de la sostenibilidad definidos por la CRUE. En este anexo, presento una reflexión sobre cómo la sostenibilidad ha estado presente de manera transversal en mi trabajo, no solo en los objetivos del proyecto, sino también en la manera en que se ha diseñado y ejecutado.

### **Sostenibilidad ambiental**

Uno de los principales valores del proyecto ha sido el uso de un modelo digital para representar entornos hospitalarios, lo que permite reducir la necesidad de ensayos físicos y visitas técnicas, minimizando desplazamientos, consumo energético y generación de residuos. La integración de datos ambientales como la temperatura y la humedad dentro del modelo Revit facilita la supervisión remota y el control del entorno, promoviendo una gestión más eficiente de la climatización y, por tanto, un uso más racional de los recursos energéticos.

Asimismo, aunque el sistema actual utiliza datos simulados, en un contexto real permitiría detectar desviaciones térmicas o de humedad que pueden comprometer la eficiencia energética del sistema HVAC, facilitando la toma de decisiones para corregirlas de manera precoz. Esta anticipación contribuye a reducir consumos innecesarios y prolongar la vida útil de los equipos, alineándose con los principios del diseño sostenible.

## **Sostenibilidad social y sanitaria**

El proyecto también contribuye indirectamente a la sostenibilidad social, en tanto que mejora las condiciones ambientales de las Unidades de Cuidados Intensivos, espacios críticos donde la estabilidad térmica e higrométrica tiene un impacto directo en la salud y la recuperación de los pacientes. Mejorar la monitorización de estas condiciones favorece un entorno más seguro tanto para el personal sanitario como para los pacientes, lo cual redunda en una mejor calidad asistencial.

En particular, este trabajo propone una arquitectura que, en una posible implementación futura, podría incluir sensores reales y análisis en tiempo real. Esto podría permitir actuar rápidamente ante situaciones de riesgo (por ejemplo, alta humedad que favorezca infecciones), mejorando la respuesta hospitalaria y contribuyendo a la seguridad del paciente, que es uno de los ejes de la sostenibilidad social en el ámbito sanitario.

## **Sostenibilidad tecnológica y económica**

Desde el punto de vista tecnológico, el sistema desarrollado hace uso de herramientas abiertas o de amplio acceso dentro del sector, lo cual favorece su replicabilidad y reduce costes de implementación. Además, se ha apostado por una arquitectura modular y escalable, lo que facilita su adaptación a otros entornos o necesidades, prolongando así su vida útil y evitando desarrollos desde cero en futuras fases.

El uso de datos simulados permite realizar pruebas y demostraciones sin necesidad de equipamiento físico costoso, lo cual favorece la sostenibilidad económica del proyecto en fases de prototipo. Esta estrategia también permite una formación más accesible para estudiantes o profesionales, contribuyendo a la difusión del conocimiento sin grandes inversiones.

## **Desarrollo de competencias en sostenibilidad**

Durante el desarrollo del proyecto he adquirido y puesto en práctica varias competencias relacionadas con la sostenibilidad:

Capacidad para diseñar soluciones técnicas que integren criterios de eficiencia energética y responsabilidad medioambiental, como la optimización del control ambiental en entornos hospitalarios.

Competencia en evaluar el impacto social y sanitario de decisiones técnicas, considerando el bienestar de los usuarios finales del sistema (pacientes, personal sanitario).

Habilidad para trabajar con herramientas digitales de forma responsable, fomentando el uso de tecnologías abiertas y de bajo impacto económico.

Conciencia crítica sobre la necesidad de planificar proyectos tecnológicos pensando en su durabilidad, escalabilidad y posible reutilización.

Estas competencias me han permitido no solo cumplir con los objetivos técnicos del TFG, sino también hacerlo con una visión más integral y responsable, alineada con los Objetivos de Desarrollo Sostenible (ODS), especialmente el ODS 3 (Salud y bienestar), el ODS 9 (Industria, innovación e infraestructura) y el ODS 12 (Producción y consumo responsables).

## **H.1. Conclusión**

La sostenibilidad ha sido un eje transversal en este Trabajo de Fin de Grado. Aunque el objetivo principal ha sido demostrar la viabilidad técnica de integrar datos ambientales en un modelo BIM de una UCI, he procurado que cada decisión técnica estuviese alineada con los principios de eficiencia, responsabilidad y mejora del entorno que definen la sostenibilidad. Este enfoque me ha permitido no solo desarrollar un sistema funcional, sino también reflexionar sobre el papel de la ingeniería en la construcción de un futuro más equitativo y sostenible.



---

## Bibliografía

---

- [Autodesk, 2025] Autodesk (2025). Comprar revit 2025 – licencia comercial. Disponible en: <https://www.autodesk.es/products/revit/subscribe> [Último acceso: mayo de 2025].
- [de España, 2018] de España, G. (2018). Ley orgánica 3/2018, de protección de datos personales y garantía de los derechos digitales (lopddg). Disponible en: <https://www.boe.es/buscar/act.php?id=BOE-A-2018-16673> [Último acceso: mayo de 2025].
- [Jobted, 2024] Jobted (2024). ¿cuánto cobra un ingeniero biomédico? (sueldo 2024). Disponible en: <https://www.jobted.es/salario/ingeniero-biom%C3%A9dico> [Último acceso: mayo de 2025].
- [Social, 2023] Social, S. (2023). Bases y tipos de cotización 2023. Disponible en: <https://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/36537> [Último acceso: mayo de 2025].
- [y Consejo de la Unión Europea, 2016] y Consejo de la Unión Europea, P. E. (2016). Reglamento (ue) 2016/679 del parlamento europeo y del consejo, relativo a la protección de las personas físicas (rgpd). Disponible en: <https://eur-lex.europa.eu/eli/reg/2016/679/oj> [Último acceso: mayo de 2025].