



industriales

etsii

Escuela Técnica
Superior
de Ingeniería
Industrial

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería Industrial



Universidad
Politécnica
de Cartagena

MIEMBRO DE



EUROPEAN
UNIVERSITY OF
TECHNOLOGY



CAMPUS MARE NOSTRUM
de Excelencia Internacional

Aplicación de Técnicas de Inteligencia de Negocio para datos de la Automatización de una Planta Industrial

TRABAJO FIN DE MÁSTER

MASTER EN INDUSTRIA 4.0



Universidad
Politécnica
de Cartagena

Autor: Rubén Nicolás Ibáñez

Director: Antonio Guerrero González

Codirector:

Cartagena, septiembre 2022

Resumen

Una empresa de automatización situada en Molina de Segura, que se llama IMA S.L. (Ingeniería Mecánica y Automática S.L.), quiere dar el salto a la Industria 4.0 e implementar técnicas de inteligencia de negocio para facilitar la comercialización y adelantarse a la competencia con estas tecnologías emergentes. Por lo tanto, el objetivo de este proyecto consiste en explicar una posible vía para responder a las necesidades requeridas.

El proyecto consiste en la conexión de una instalación automatizada con tecnología PLC con una pasarela Edge Computing a una base de datos. Esto se realizará mediante el software Node-RED. Posteriormente se visualizarán los datos obtenidos en una página web. Por otro lado, también se hará un procesado y visualización de los datos para la inteligencia de negocio.

Para ello, se subirán los datos mediante la pasarela IOT2050 a Node-RED y se conectará a la base de datos de IBM para la posterior visualización de datos. Luego, se exportarán esos datos y se mostrarán en PowerBI.

Abstract

An automation company located in Molina de Segura, called IMA S.L. (Ingeniería Mecánica y Automática S.L.), wants to make the leap to Industry 4.0 and implement business intelligence techniques to facilitate marketing and stay ahead of the competition with these emerging technologies. Therefore, the objective of this project is to explain a possible way to respond to the required needs.

The project consists of connecting an automated installation with PLC technology with an Edge Computing gateway to a database. This will be done by means of the Node-RED software. Subsequently, the data obtained will be visualized on a web page. On the other hand, the data will also be processed and visualized for business intelligence.

To do this, the data will be uploaded through the IOT2050 gateway to Node-RED and connected to the IBM database for subsequent data visualization. Then, this data will be exported and displayed in PowerBI.

Índice de contenido

Resumen	2
Abstract	2
Índice de figuras	5
Índice de tablas	8
1. INTRODUCCIÓN	9
1.1. Motivaciones	9
1.2. Objetivos del proyecto	10
1.3. Estructura del proyecto	12
2. HERRAMIENTAS, SOFTWARES, SERVICIOS Y DISPOSITIVOS A UTILIZAR O A TENER EN CUENTA	13
2.1. Tecnologías necesarias para el proyecto	13
2.1.1. PLC SIMATIC S7-1200	13
2.1.2. IOT 2050	14
2.1.3. LUTZE ET-SWU5E	15
2.1.4. Node-RED	16
2.1.5. IBM Cloud	17
2.1.6. PowerBI	18
2.2. Otras tecnologías a tener en consideración	20
2.2.1. OPC UA	20
2.2.2. Ignition	23
2.2.3. PAC Control y Groov View	24
2.3. Descripción de distintas bases de datos en la nube	26
2.3.1. Microsoft Azure	26
2.3.2. Amazon RDS	27
2.3.3. Google Cloud SQL	28
2.4. Edge Computing. Distintas pasarelas	29
2.4.1. IOT 2020/40	29
2.4.2. Raspberry Pi 4	30
2.4.3. ESP 32 y ESP 8266	31
2.5. Familia de PLC SIMATIC de SIEMENS	33
2.5.1. S7-200	33
2.5.2. S7-300	34
2.5.3. S7-400	35
2.5.4. S7-1500	36
3. EXPLICACIÓN DEL PROBLEMA, SOLUCIÓN A REALIZAR Y ARQUITECTURA	38
3.1. Funcionamiento del programa cargado en el autómatas	38
3.2. Solución a realizar	38

3.2.1.	Esquema	40
4.	SOLUCIÓN PROPUESTA	41
4.1.	Conexión a Node-RED desde el ordenador y desde la pasarela IOT 2050	41
4.1.1.	Conexión a Node-RED desde un ordenador	41
4.1.2.	Conexión a Node-RED y configuración de la pasarela IOT 2050	43
4.1.2.1.	Configuración de la pasarela IOT 2050	43
4.1.2.2.	Conexión a Node-RED desde la pasarela IOT 2050	49
4.2.	SCADA y variables	52
4.2.1.	Segmentos del programa	52
4.2.2.	Variables SCADA	56
4.3.	Node-RED y conexiones. Subida de datos a paneles de visualización de IBM Cloud	58
4.4.	Configuración de IBM Cloud	65
4.5.	Conexión a Base de Datos, exportación del fichero CSV y visualización en PowerBI	72
4.5.1.	Descarga, configuración y uso de MySQL y HeidiSQL	72
4.5.2.	Subida de datos desde Node-RED a MySQL y exportación del fichero CSV	77
4.5.3.	Creación de paneles de visualización desde PowerBI	83
5.	ANÁLISIS DE RESULTADOS EN PLATAFORMA DE INTELIGENCIA DE NEGOCIO	86
5.1.	Visualización de datos en tiempo real en IBM Cloud	86
5.2.	Visualización de datos históricos en PowerBI	88
6.	CONCLUSIONES	91
7.	TRABAJOS FUTUROS	91
8.	BIBLIOGRAFÍA	93

Índice de figuras

Figura 1. Logo IMA	9
Figura 2. Conexión PLC a IBM Cloud. Del vídeo Publicar datos a la nube por IBM Cloud, Jesús Correa	10
Figura 3. Catálogo completo de IBM Cloud	11
Figura 4. Hardware a utilizar	13
Figura 5. PLC SIMATIC S7-1200	14
Figura 6. IOT 2050	15
Figura 7. LUTZE ET-SWU5E	16
Figura 8. Logo de Node-RED y entorno	16
Figura 9. Logo IBM	17
Figura 10. Logo IBM Cloud	17
Figura 11. Ejemplo del entorno de los paneles de visualización de IBM Cloud	18
Figura 12. Logo PowerBI	18
Figura 13. Arquitectura básica utilizada en BI	19
Figura 14. Ejemplo Dashboard PowerBI	19
Figura 15. Logo OPC UA	20
Figura 16. Diferentes herramientas de OPC UA	20
Figura 17. Pirámide de automatización	21
Figura 18. Funcionalidad de OPC UA desde TIA Portal V16	21
Figura 19. Agregar nueva interfaz del servidor OPC UA	22
Figura 20. Logo Ignition	22
Figura 21. Funcionalidades de Ignition	23
Figura 22. Logo PAC Control	23
Figura 23. Bloques PAC Control	24
Figura 24. Logo Groov EPIC	24
Figura 25. Conexión en Groov View	25
Figura 26. Logo Microsoft Azure	26
Figura 27. Ejemplo de comunicación de Node-RED con Azure	26
Figura 28. Logo Amazon RDS	27
Figura 29. Logo Google Cloud SQL	28
Figura 30. Utilidad de Gateway. Imagen de Aprendiendo Arduino	29
Figura 31. IOT 2020	29
Figura 32. Raspberry Pi 4	30
Figura 33. ESP 32	31
Figura 34. PLC SIMATIC S7-200	33
Figura 35. PLC SIMATIC S7-300	34
Figura 36. PLC SIMATIC S7-400	35
Figura 37. PLC SIMATIC S7-1500	36
Figura 38. Configuración de la conexión con el autómata	38
Figura 39. Nodos de SIEMENS	39
Figura 40. Nodos de IBM Cloud	39
Figura 41. Esquema del proyecto	40
Figura 42. Acceso al cmd	41
Figura 43. Cargando Node-RED	41
Figura 44. Entorno Node-RED desde PC	42
Figura 45. Descarga de nodos	42
Figura 46. Escritura de imagen en IOT 2050	43
Figura 47. Alimentación IOT 2050	44
Figura 48. Introducción de microSD en IOT 2050	44
Figura 49. Configuración Ethernet para IOT 2050	45

Figura 50. PuTTY para acceder a la consola de la pasarela	45
Figura 51. Consola de la pasarela	46
Figura 52. Setup de la pasarela	46
Figura 53. Conexiones a la pasarela	47
Figura 54. Dirección IP de acceso a la pasarela por una vía	48
Figura 55. Dirección IP de acceso a la pasarela por otra vía	48
Figura 56. Entorno Node-RED desde IOT 2050	50
Figura 57. Ping a la pasera IOT 2050	51
Figura 58. Contador 1 TIA Portal	52
Figura 59. Contador 2 TIA Portal	53
Figura 60. Función Calculate TIA Portal	54
Figura 61. Contador de cajas TIA Portal	55
Figura 62. Contador de palés TIA Portal	55
Figura 63. Tabla de variables TIA Portal	56
Figura 64. Bloques de datos TIA Portal	57
Figura 65. Esquema conexión de autómeta a IBM Cloud	58
Figura 66. Configuración nodo s7 in	59
Figura 67. Variables nodo s7 in	60
Figura 68. Rack y Slot del autómeta	61
Figura 69. Configuración conexión s7 in	61
Figura 70. Datos del autómeta mostrados en Node-RED	62
Figura 71. Configuración nodo IoT Watson	63
Figura 72. Organización IBM Cloud	63
Figura 73. Configuración propiedades del nodo IoT Watson	64
Figura 74. Lista de recursos IBM Cloud	65
Figura 75. Internet of Things IBM Cloud	65
Figura 76. Configuración del recurso	66
Figura 77. Lanzamiento del recurso creado	66
Figura 78. Entorno de creación de dispositivos y tipos de dispositivos	67
Figura 79. Creación de un tipo de dispositivo	67
Figura 80. Tipo de dispositivo creado	67
Figura 81. Creación de dispositivo	68
Figura 82. Autenticación del dispositivo	68
Figura 83. Conexión Node-RED a IBM Cloud	68
Figura 84. Dispositivo creado	69
Figura 85. Datos subidos desde Node-RED en tiempo real	69
Figura 86. Datos desglosados	69
Figura 87. Creación de panel de visualización	70
Figura 88. Paneles de visualización disponibles	70
Figura 89. Tipos de tarjetas	71
Figura 90. Panel de visualización IBM Cloud	71
Figura 91. Descarga MySQL	72
Figura 92. Descarga HeidiSQL	72
Figura 93. Acceso a MySQL	73
Figura 94. Creación de sesión en HeidiSQL	73
Figura 95. Creación de base de datos y tabla de variables	74
Figura 96. Variables introducidas a la tabla	74
Figura 97. Autenticación de base de datos	74
Figura 98. Información de base de datos y tabla desde comandos en MySQL	75
Figura 99. Carga de datos en la tabla	75
Figura 100. Datos subidos manualmente	76
Figura 101. Muestra de datos por correspondencia	76

Figura 102. Instalación nodo MySQL	77
Figura 103. Conexión de Node-RED a la base de datos MySQL	77
Figura 104. Convertir variables en flow	78
Figura 105. Datos subidos de manera individual	78
Figura 106. Programación para subir los datos a MySQL	79
Figura 107. Configuración nodo MySQL	79
Figura 108. Datos subidos y visualizados en HeidiSQL	80
Figura 109. Exportar archivo CSV	81
Figura 110. Datos en Excel	82
Figura 111. Importar y cargar datos a PowerBI	83
Figura 112. Manipulación de datos y opción de conexión con otros archivos Excel	84
Figura 113. Columna Tiempo (s) añadida	84
Figura 114. Diferentes tipos de paneles de visualización	85
Figura 115. Paneles IBM Cloud	86
Figura 116. Más opciones de paneles en IBM Cloud	87
Figura 117. Cantidad de datos enviados. Panel interactivo en IBM Cloud	87
Figura 118. Paneles de visualización en PowerBI	88
Figura 119. Más paneles de visualización en PowerBI	88
Figura 120. Paneles interactivos en PowerBI	89
Figura 121. Información detallada en paneles interactivos de PowerBI	90

Índice de tablas

Tabla 1. Características IOT 2050	14
Tabla 2. Características LUTZE ET-SWU5E	15
Tabla 3. Características IOT 2020 e IOT 2040	30
Tabla 4. Características Raspberry Pi 4	31
Tabla 5. Características ESP 32 y ESP 8266	32
Tabla 6. Características SIMATIC S7-200	33
Tabla 7. Características SIMATIC S7-300	34
Tabla 8. Características SIMATIC S7-400	35
Tabla 9. Características SIMATIC S7-1500	37
Tabla 10. Conversión direcciones de variables de Step7 a Node-RED	59

1. INTRODUCCIÓN

En este apartado se realizará una breve descripción del proyecto, comentando, entre otras cosas las motivaciones de por qué decantarse a hacer algo así y los antecedentes de la empresa para tener una mejor perspectiva del salto hacia delante con esta idea. Por otro lado, también se debe hablar de los objetivos a lograr con la realización de este trabajo, y hacer una breve estructura de la memoria del proyecto para tener una idea breve de lo que se va a comentar y discutir aquí.

1.1. Motivaciones

La empresa, conocida como Ingeniería Mecánica y Automática *IMA*, donde estoy realizando prácticas de empresa en colaboración con la *UPCT* y motivado con el *Encuentro TF*, se dedica a la fabricación de maquinaria para otras fábricas, así como paletizadoras, encajadoras, cintas, etc. Sin embargo, tras el auge de la Industria 4.0, IOT y la inteligencia de negocio, quieren dar el salto a esa nueva tecnología.



Figura 1. Logo IMA

El objetivo principal de la empresa consiste en poder entender el funcionamiento de estas nuevas tecnologías y así no depender tanto de empresas externas que les haga el trabajo, y, por tanto, estar algo más especializados. Por ende, lo que quieren conseguir con este proyecto es un proyecto piloto a modo de referencia para que sean capaces de empezar a realizar diversos trabajos relacionados con la Industria 4.0, o al menos tener una primera idea, ya que cada vez la demanda de esta nueva tecnología es mayor.

En resumen, el proyecto que se va a explicar consiste en la subida de datos desde un autómata hacia una base de datos en la nube. Para conseguirlo, se ha optado por hacer uso de una pasarela IOT2050, que subirá los datos de MQTT (almacenados directamente desde el PLC SIMATIC S7-1200 de SIEMENS) a Node-RED local, y de ahí se subirán los datos a una base de datos creada en el servidor de IBM mediante una conexión a IBM Watson con su propio Node-RED, para poder visualizarlos en cualquier momento.

Sin embargo, hay más vías a tener en cuenta que se explicarán brevemente en un apartado posterior para tener una mayor perspectiva del abanico de opciones que existen, como es el uso de *Software Ignition*, la tecnología de *OPC UA* y distintos servidores para bases de datos como podría ser *Google* o *Azure*, entre otros. También se podrían utilizar otras pasarelas como Raspberry Pi, un ESP 32 o ESP 8266, o incluso un Arduino MKR 1010 o 1300. Además, hay distintas tecnologías o aplicaciones para suplir el uso de Node-RED y hacerlo de una manera más simple, pero a su vez menos potente.

Publicar de IBM Cloud (NodeRED) a IoT Gateway (NodeRED)

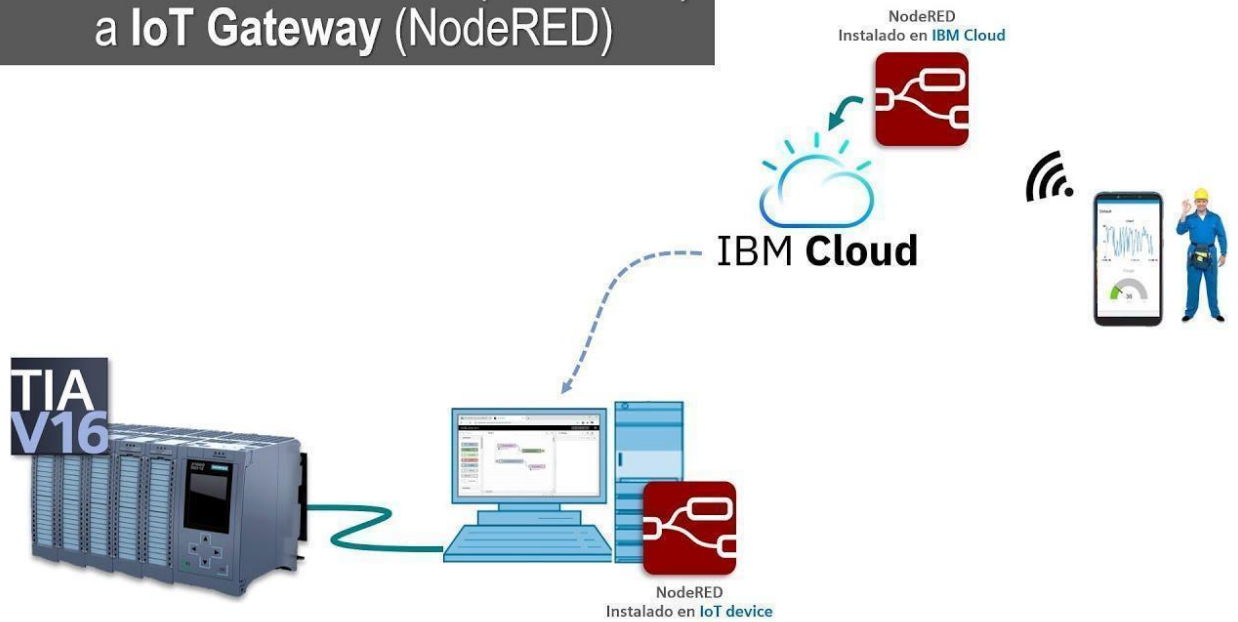


Figura 2. Conexión PLC a IBM Cloud. Del vídeo Publicar datos a la nube por IBM Cloud, Jesús Correa

1.2. Objetivos del proyecto

En este apartado se nombran los principales objetivos marcados para la realización de este TFM. Primero hay que hablar de los objetivos generales del proyecto.

Estoy realizando las prácticas en una empresa que se dedica a la fabricación de distintos dispositivos y maquinaria a nivel industria y que, por lo tanto, dicha maquinaria envía determinadas señales e información. Lo que se quiere llevar a cabo con este proyecto consiste en subir esa información al Cloud para poder disponer de ella, visualizarla y tenerla al alcance de la mano incluso desde distintas plataformas, como puede ser un dispositivo móvil.

Por lo tanto, el objetivo principal de este proyecto consiste en la toma de datos y vinculación de equipos físicos con servicios en la nube de tal manera que todos los datos, información y comandos que se intercambia de manera local esté vinculada con el cloud, en este caso IBM Cloud, ya que este proporciona un amplio abanico de servicios. Entre muchos de esos servicios se encuentra el que se va a utilizar, de IoT (Internet of Things); creación de APIs; conexiones VPNs; bases de datos como en MongoDB, MySQL y PostgreSQL; incluso herramientas de inteligencia de negocio.

Este es el catálogo de servicios que ofrece la plataforma de IBM Cloud. Como se puede apreciar, tienen una inmensa cantidad de herramientas a tener en consideración en función del proyecto que se quiera realizar o el tipo de empresa.



Figura 3. Catálogo completo de IBM Cloud

Sobre los objetivos más específicos y enfocado al problema que se quiere solucionar en este proyecto, se encuentran enumerados los siguientes:

1. Subida de datos desde el autómata a MQTT.
2. Subida de datos de MQTT a Node-RED.
3. Conexión de nuestro Node-RED en local de la pasarela IOT2050 con Node-RED de IBM Watson.
4. Creación de una base de datos en el sistema de IBM Cloud y depositar los datos ahí.
5. Creación de gráficos que monitorizan la cantidad de cajas y palés que pasan a tiempo real con un reseteo en el intervalo de tiempo deseado.
6. Exportación de datos para visualizar históricos con el programa de inteligencia de negocio PowerBI.

1.3. Estructura del proyecto

CAPÍTULO 1. Introducción: En este primer apartado, se comenta brevemente de qué va a ir el proyecto, así como los objetivos que se tienen y las motivaciones para llevar a cabo esta idea.

CAPÍTULO 2. Tecnología a utilizar: Aquí se describe el hardware y todos los programas que se van a utilizar para la realización del trabajo. Además, se hará un breve repaso de otras alternativas que se podrían tener en cuenta y de la familia PLC SIMATIC de SIEMENS.

CAPÍTULO 3. Explicación del problema, solución a realizar y arquitectura: Se programará con TIA Portal un pequeño programa para demostrar cómo se realiza una subida y monitorización de datos a la base de datos de IBM Cloud. El programa no será muy extenso, ya que el objetivo del proyecto no es ese, se pondrán las justas y necesarias para tener algo que poder mostrar y subir. A continuación, se hablará de manera más detallada de la solución escogida.

CAPÍTULO 4. Solución Propuesta: Se realizará el proyecto entero y se enseñará el proceso paso por paso.

CAPÍTULO 5. Análisis de resultados en plataforma de inteligencia de negocio: En este apartado, se explicará cómo mostrar los valores históricos desde otra web generada por un programa de inteligencia de negocio llamado PowerBI. Los datos se pueden importar al programa de dos formas: Exportando desde Node-RED o IBM Cloud a Excel, o bien enviando la información a una base de datos, e importarlo en el programa, o conectar PowerBI con la base de datos, que tenemos también esa opción.

CAPÍTULO 6. Conclusiones y posibles futuros trabajos: Se expondrán los resultados obtenidos y se llegará a una conclusión respecto si merece la pena o no dar el salto a la Industria 4.0. También se hablará de algunas posibles ideas de proyectos a tener en cuenta en el futuro.

2. HERRAMIENTAS, SOFTWARES, SERVICIOS Y DISPOSITIVOS A UTILIZAR O A TENER EN CUENTA

2.1. Tecnologías necesarias para el proyecto

En este apartado se describirán las tecnologías que de las que se va a hacer uso en este proyecto. Desde el hardware hasta todos los programas que se requieren para llevar a cabo la visualización de datos.

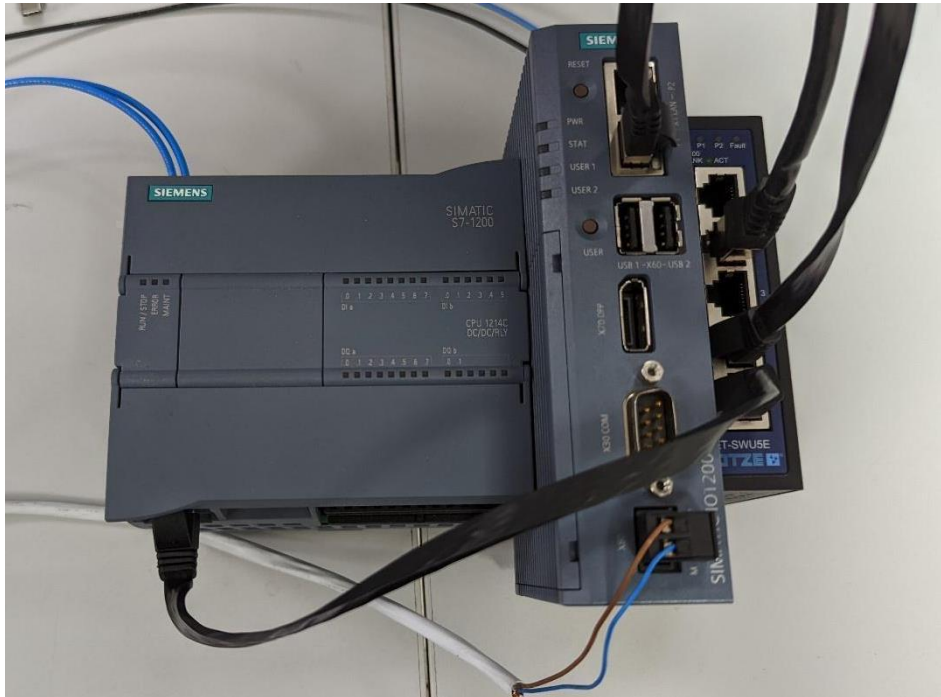


Figura 4. Hardware a utilizar

Como se puede apreciar en la imagen, se ha colocado un PLC SIMATIC S7-1200 de SIEMENS junto a un IOT 2050 en un carril DIN para poder realizar las pruebas de subida y visualización de datos durante el proyecto. Además, se hace uso del ET-SWU5E, de la marca LUTZE, que sirve para comunicar el PLC y la pasarela mediante puertos Ethernet. En este caso ese componente tiene cinco.

2.1.1. PLC SIMATIC S7-1200

El PLC Simatic S7-1200 es el módulo más utilizado en la industria. Consta de 14 entradas digitales y 10 salidas de relé digital. El rango de temperatura de funcionamiento oscila entre los -20°C y los 60°C , por lo que sirve para la amplia mayoría de trabajos en la industria. Posee tres tipos de puertos de comunicación: Ethernet, Profinet y UDP. Por otro lado, la capacidad máxima del programa es de 75 kB y la memoria es de 4 MB. Como es evidente, cumple el Certificado Europeo y sus dimensiones son de 100 x 110 x 75 mm, ya que hay que tenerlo en cuenta a la hora de montaje en un carril DIN de un cuadro eléctrico.



Figura 5. PLC SIMATIC S7-1200

2.1.2. IOT 2050

La pasarela (o Gateway) utilizada para este proyecto es la IOT 2050, más concretamente la versión Basic, cuyo nombre técnico es 6ES7647-0BA00-0YA2. Las principales características, comparándolas con el módulo Advanced son las siguientes:

Características	IOT 2050 Basic	IOT 2050 Advanced
<i>Tensión de alimentación</i>	24 VDC	24 VDC
<i>Procesador</i>	Dual Core AM6528 GP	Quad Core AM6548 HS
<i>RAM</i>	1 GB DDR	2 GB DDR4
<i>eMMc</i>	No	16 GB
<i>Batería RTC</i>	No	Sí
<i>Instalación de SO</i>	Instalado en microSD (Como IOT2040)	Sí
<i>Interfaz Arduino UNO</i>	Sí	Sí
<i>Interfaz Serial</i>	1x Serial (RS 232/485)	1x Serial (RS 232/485)
<i>mPCIe Slot</i>	Sí	Sí
<i>Interfaz Ethernet</i>	2x Ethernet (Gigabit)	2x Ethernet (Gigabit)
<i>Slot Micro SIM para 4G</i>	Sí, con tarjeta adicional	Sí, con tarjeta adicional

Tabla 1. Características IOT 2050



Figura 6. IOT 2050

2.1.3. LUTZE ET-SWU5E

Se utiliza el modelo de LUTZE ET-SWU5E para poder realizar la comunicación por Ethernet, ya que era necesario por la falta de puertos de los que se disponían. Con este dispositivo, se dispone de cinco puertos Ethernet más, pero con dos es más que suficiente (Uno para el PLC y otro para la pasarela).

Sus principales características son las siguientes:

Comunicación	
<i>Estándar</i>	IEEE 802.3 10BASE-T, IEEE 802.3u 100BASE-TX/100BASE-FX
<i>LAN</i>	10/100 Base-TX RJ45 Auto-MDI/MDI-X
<i>Velocidad de transmisión</i>	Max. 10/100 Mbit/s
<i>Longitud del cable (segmento)</i>	Max. 100 m
Supervisión	
<i>Salida de problema</i>	Relé, 1 contacto NA - 1 A @ 24 VDC
<i>Corriente de conexión</i>	1 A @ 24 VDC
<i>Tensión de conexión</i>	120 VAC / VDC
<i>Tensión de aislamiento</i>	500 VDC
Condiciones ambientales generales	
<i>Rango de temperatura de trabajo y almacenamiento</i>	-40°C / 75°C
Datos generales	
<i>Rango de tensión nominal</i>	9 V – 52 V
<i>Corriente nominal</i>	2 W, DC: Max. 0,17 A, AC: Max. 0,15 A
<i>Grado de protección</i>	IP30
<i>Humedad relativa del aire (en funcionamiento)</i>	5% - 90% (sin condensación)
<i>Humedad relativa del aire (almacenamiento)</i>	0% - 90% (sin condensación)

Tabla 2. Características LUTZE ET-SWU5E



Figura 7. LUTZE ET-SWU5E

2.1.4. Node-RED

Para la comunicación entre el PLC con la base de datos en la nube se hará uso de Node-RED, que es una herramienta de programación que sirve para la comunicación de dispositivos de hardware, servicios de internet, etc. Esta herramienta es muy interesante para equipos que se dediquen a IIOT (Internet de las Cosas Industrial) y para la comunicación de equipos con aplicaciones IOT.

Esta tecnología fue desarrollada por IBM en el año 2013. Su principal característica es su sencillez, ya que puedes acceder o conectar un amplio abanico de tecnologías sin tener que profundizar del todo en ellas, por lo que facilita mucho el trabajo.

La estructura mínima del entorno son nodos, que permiten hacer una tarea concreta, como recibir un mensaje MQTT, enviar datos a bases de datos, tanto de manera local como a un servicio en la nube, visualizar dichos datos (en la ventana de la derecha o en el dashboard que ofrecen), comunicar con bots de Telegram, y un largo etcétera.

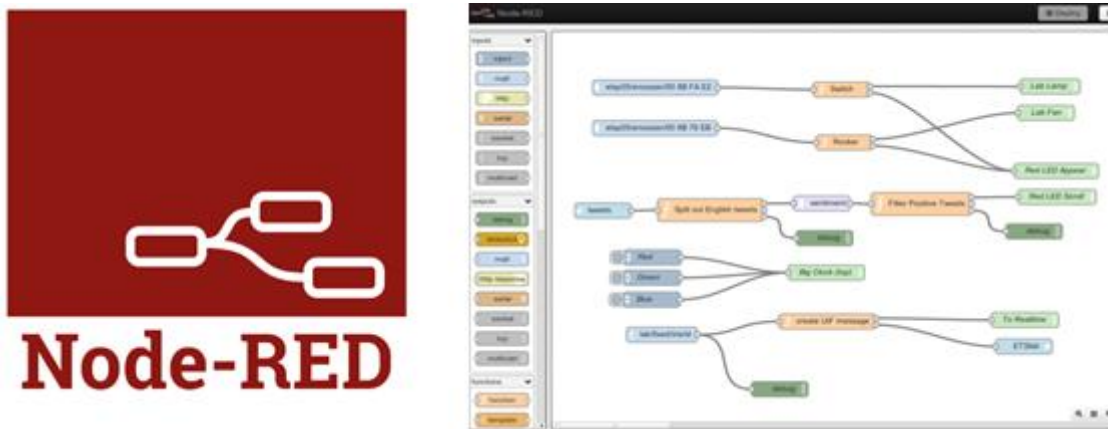


Figura 8. Logo de Node-RED y entorno

Hay diversos dispositivos en los que se puede ejecutar Node-RED. Se puede hacer desde el propio ordenador, desde una pasarela IOT, como puente de enlace entre MQTT y la nube, y otras muchas más. En este proyecto se van a explicar dos maneras, instalando y ejecutando Node-RED desde un PC y desde una pasarela IOT 2050, que es algo más laborioso que la primera alternativa, pero bastante más interesante para proyectos de gran escala.

Para instalar y ejecutar Node-RED hay que hacerlo desde la consola con una serie de comandos, y para entrar y empezar a trabajar se debe conocer la dirección IP del dispositivo en el que se está ejecutando. Para el caso de este proyecto, que se usa un IOT 2050, las direcciones IP de conexión a la pasarela, donde se ejecuta Node-RED es <http://192.168.200.1:1880> o <http://192.168.1.81:1880>, donde los dos primeros grupos (192.168) son los identificadores de red, mientras que los dos últimos (200.1 y 1.81) son los indicadores del HOST. El “1” del último grupo hace referencia al dispositivo que se está utilizando dentro de una conexión, que en este caso es el Gateway. Por otro lado, el término “:1880” hace referencia al número del puerto utilizado.

2.1.5. IBM Cloud

La práctica se va a realizar con la base de datos de IBM Cloud, ya que es de las nubes más seguras para empresas, con funcionalidades tanto de Inteligencia Artificial como de datos. Además, tiene muchísimos recursos para desarrollo de una aplicación IOT, IIOT y más sectores.

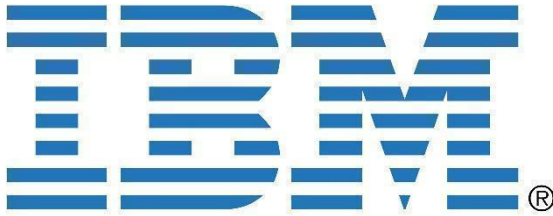


Figura 9. Logo IBM



Figura 10. Logo IBM Cloud

IBM Cloud combina una plataforma como servicio (PaaS) con la infraestructura como servicio (IaaS) para así poder tener una experiencia integrada. Esta plataforma permite soporte desde equipos de desarrollo pequeños hasta grandes empresas. Esta plataforma ha sido diseñada para dar soporte a cualquier necesidad de los usuarios, ya sea un proyecto de trabajo o con un despliegue de la multi nube con IBM Cloud Parks. IBM realiza una combinación entre la nube pública y privada, lo cual aporta una enorme flexibilidad para mover cargas de trabajo en ambas nubes en función de las necesidades del usuario (sean empresariales o tecnológicas).

Por otro lado, IBM utiliza Red Hat OpenShift, que es la plataforma de contenedores de nube híbrida líder de mercado, ya que permite crear una vez y desplegar en cualquier lugar. Por otro lado, con IBM Cloud Satellite, se puede crear un entorno que aporte escalabilidad y flexibilidad bajo la demanda de servicios de la nube pública a las aplicaciones ejecutadas en la nube privada.



Figura 11. Ejemplo del entorno de los paneles de visualización de IBM Cloud

2.1.6. PowerBI



Figura 12. Logo PowerBI

El término Business Intelligence (BI) se refiere al uso de distintas estrategias y herramientas para transformar información en conocimiento o algo tangible, para poder facilitarle a la empresa la toma de decisiones sobre cualquier aspecto. Para ello, se hará uso de PowerBI, un servicio de análisis de negocios de Microsoft, que proporciona servicios basados en la nube. Un aspecto positivo de esta herramienta es a la hora de su uso, ya que es un programa muy sencillo y muy intuitivo, se aprende a usar realmente rápido. Otro aspecto a tener en cuenta es a la hora de la visualización de datos, ya que sus dashboards pueden ser interactivos y estar relacionados unos con otros según cómo interese mostrar o manipular los datos.

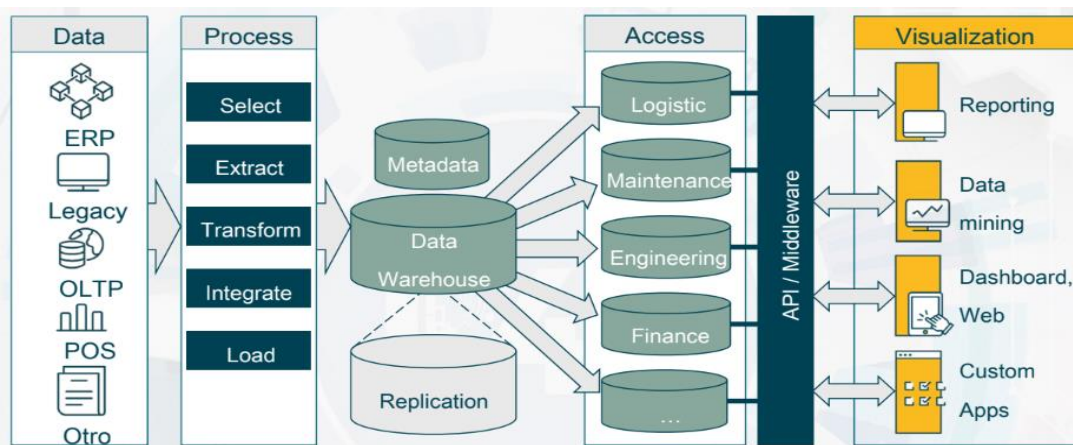


Figura 13. Arquitectura básica utilizada en BI

- Los sistemas de planificación de recursos empresariales (ERP – Enterprise Resource Planning) integran y manejan muchos de los negocios asociados con la producción de bienes y servicios.
- El sistema Legacy es un sistema informático que utiliza el usuario y no se requiere reemplazar a día de hoy.
- OLTP es el Procesamiento de Transacciones en Línea (OnLine Transaction Processing) que facilita y administra aplicaciones para entrada de datos y recuperación o procesamiento de transacciones.
- POS (Point of Sale) permite gestionar tareas relacionadas con la venta.
- Los metadatos son datos que escriben otros datos. Un grupo de metadatos se refiere a un grupo de datos que describen el contenido informativo de un objeto o recurso.
- El almacén de datos, o Data Warehouse, guarda colecciones de datos orientadas a un ámbito determinado. Está integrado, no es volátil y variable en el tiempo, que ayuda a la toma de decisiones en una empresa o proyecto.

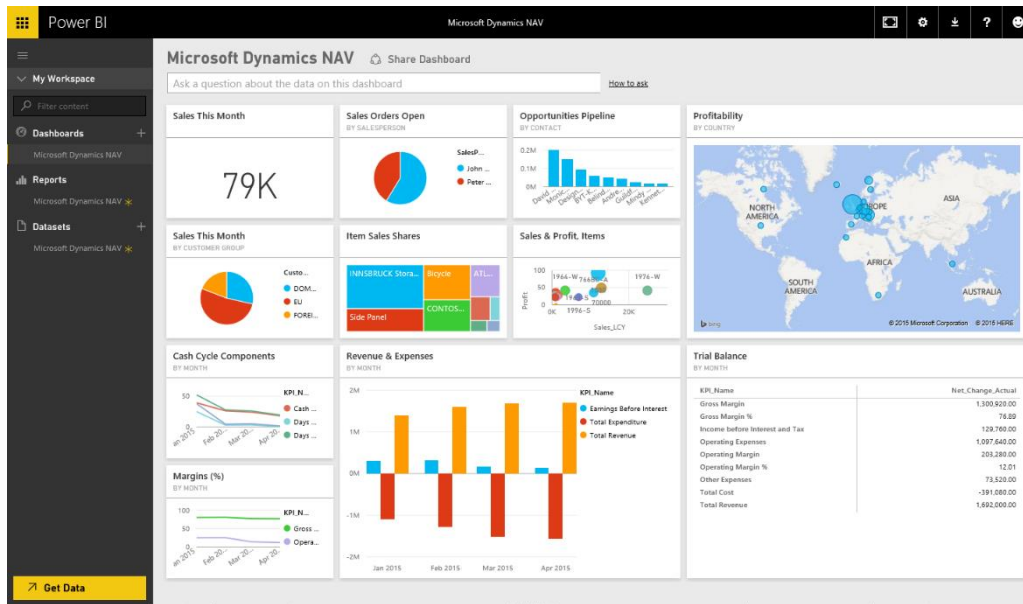


Figura 14. Ejemplo Dashboard PowerBI

2.2. Otras tecnologías a tener en consideración

Evidentemente, esta estrategia seleccionada no es la única que existe, ya que hay muchísimas vías a elegir para poder monitorizar o controlar cualquier dispositivo. A continuación, en este apartado, se va a comentar alguna de esas tecnologías, ya que se pueden tener en consideración en un futuro para diferentes proyectos dependiendo del presupuesto, comodidad, tecnologías que disponga la empresa, etc.

Las tecnologías que se van a mencionar van desde distintos softwares a utilizar para el control o monitorización de datos; como OPC UA, Ignition, PAC Control y Groov View; otras bases de datos; como Microsoft Azure, Amazon RDS o Google Cloud SQL; y distintas pasarelas que puedan conectar a Node-RED.

2.2.1. OPC UA



Figura 15. Logo OPC UA

OPC (OLE for Process Control) tiene como objetivo entregar un estándar de comunicación abierto e independiente. Su ámbito es la industria, ya que se encuentra en software industrial como SCADA, Historiadores, etc. Sin embargo, OPC tiene ciertos inconvenientes, por lo que también existe OPC UA, que tiene ciertas ventajas. Las más interesantes son las siguientes:

- OPC UA se puede implementar en cualquier sistema operativo, incluido Android, mientras que OPC solo en Windows.
- Los servidores de OPC UA pueden estar en el mismo PLC, mientras que los de OPC son drivers que hay que instalar.
- OPC UA permite la encriptación de las comunicaciones con certificados de seguridad.

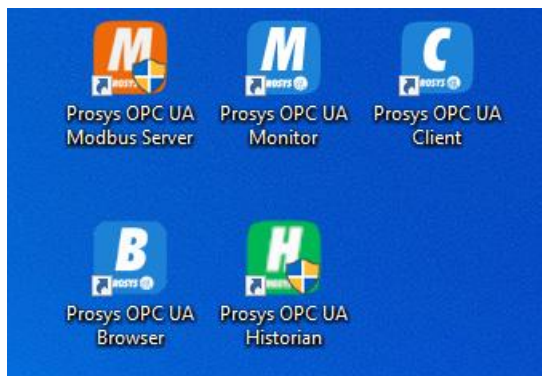


Figura 16. Diferentes herramientas de OPC UA

Sobre las herramientas de la Figura 16, una breve explicación de cada una:

1. **Modbus Server:** Es el Servidor OPC. Es un conector entre el mundo OPC con los protocolos nativos de una fuente de datos. Es bidireccional y puede tanto leer como escribir datos.
2. **Client:** Se conecta al Servidor OPC y se puede observar cómo va evolucionando la información. También se pueden manipular las variables.
3. **Monitor:** Es el Dashboard de OPC, aquí se pueden visualizar todos los datos en tiempo real en distintos KPI al gusto del usuario.
4. **Browser:** Es un navegador donde se pueden encontrar todos los valores históricos de las variables.
5. **Historian:** Con esta herramienta se puede subir los datos de OPC a cualquier base de datos. Si se quisieran visualizar estos valores, se podrían usar herramientas como HeidiSQL, que es un visor de bases de datos.

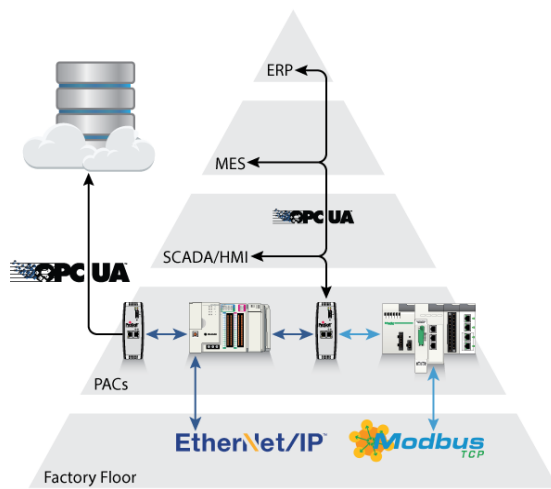


Figura 17. Pirámide de automatización

Para comunicar un autómata (en este caso el PLC de SIEMENS SIMATIC S7-1200) con OPC UA a través de TIA Portal, hay que entrar a la carpeta del modelo del PLC, El 1214C DC/DC/Rly en este caso, y ahí a la carpeta de “Comunicación OPC UA”. A continuación, se selecciona la opción de agregar una nueva interfaz de servidor.

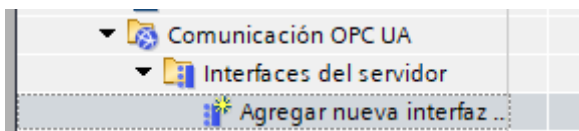


Figura 18. Funcionalidad de OPC UA desde TIA Portal V16

Posteriormente, se acepta la pestaña emergente, y simplemente habría que introducir las variables deseadas que se quieran controlar desde el Monitor y Browser de OPC UA.

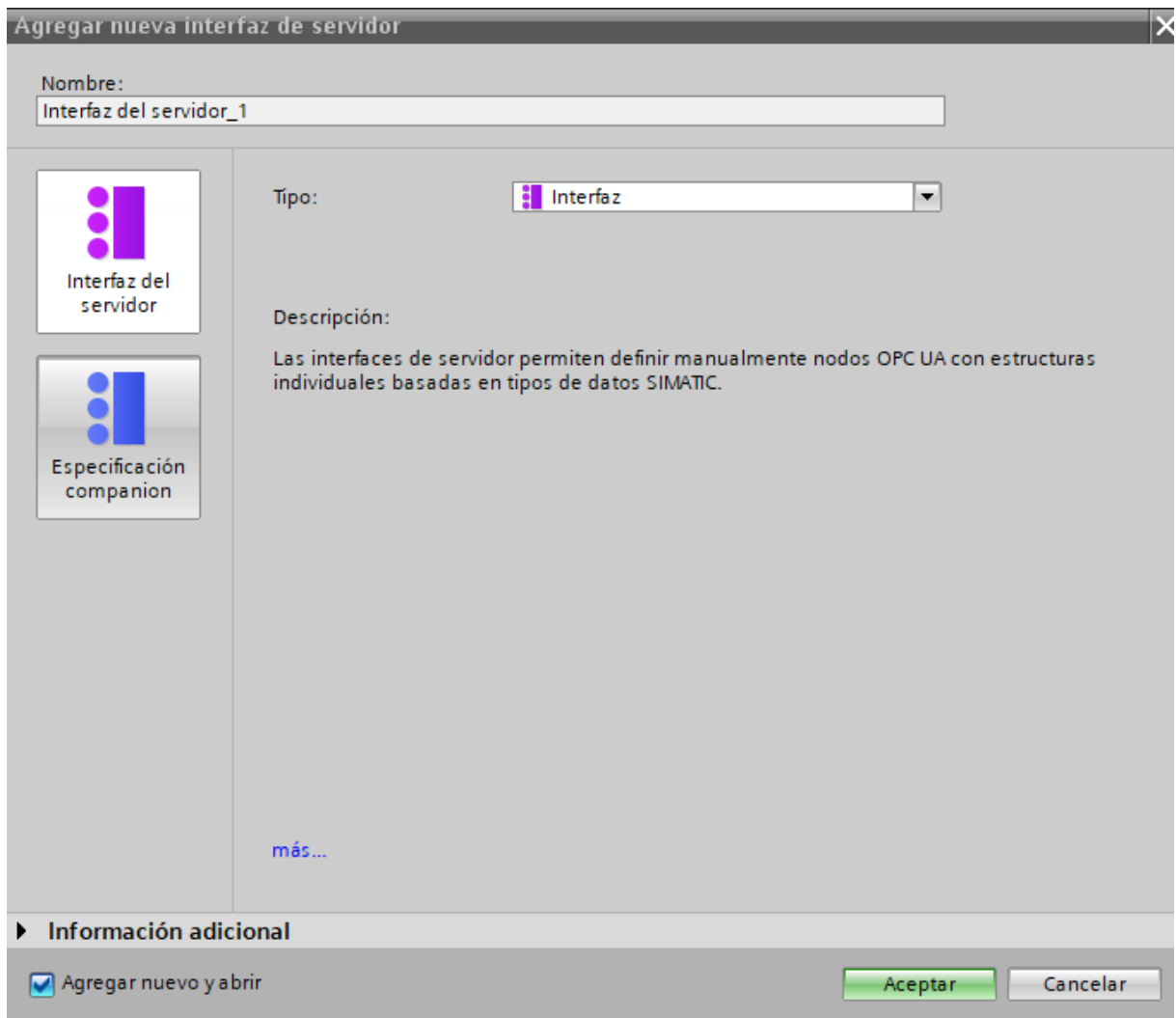


Figura 19. Agregar nueva interfaz del servidor OPC UA

2.2.2. Ignition



Figura 20. Logo Ignition

Ignition es un software de nueva generación, accesible, escalable y centrado en los datos. Es un HMI que permite construir pantallas de estado y control en tiempo real para monitorizar o controlar cualquier máquina. También es un SCADA, ya que permite la adquisición y almacenamiento de datos en una base de datos, además de poder hacer alarmas desde una ubicación central. Por otro lado, también es un MES (Manufacturing Execution Systems), que es la capa intermedia entre el ERP y la planta de producción, y provoca soluciones como OEE, SPC, seguimiento y trazabilidad o de gestión de tareas.

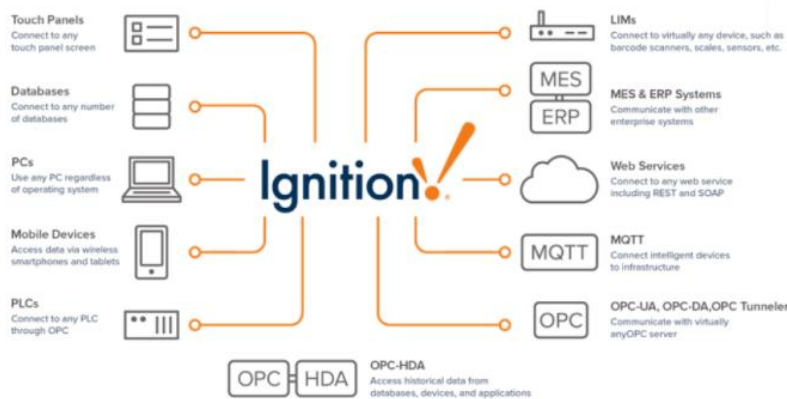


Figura 21. Funcionalidades de Ignition

Con Ignition se pueden hacer multitud de funcionalidades, las principales son las siguientes:

- **HMI SCADA:** Se puede abrir el tiempo de ejecución o a un cliente en el PC que se quiera sin importar el sistema operativo. También en smartphones o paneles táctiles.
- **Historiador:** Puede registrar datos en cualquier base de datos SQL.
- **Base de datos:** Se pueden construir aplicaciones de bases de datos. Básicamente se ejecuta una base de datos como un CRM (Customer Relationship Management).
- **Alarma:** También se puede crear una alarma con sus notificaciones a través de mensajes de audio, SMS o correos electrónicos.
- **Mix and Match:** Se pueden combinar todas estas funcionalidades, y otras más, en un mismo proyecto para poder visualizarlas todas en la misma pantalla.

2.2.3. PAC Control y Groov View

PAC Control

Figura 22. Logo PAC Control

PAC Control es un lenguaje de control de diagramas de flujo de Opto 22 para diagramas de control y entradas y salidas de Groov EPIC y SNAP PAC. Se utiliza para aplicaciones de control de procesos, IIOT y monitorización remota, entre otras muchas funciones.

Los bloques que se utilizan son los siguientes:

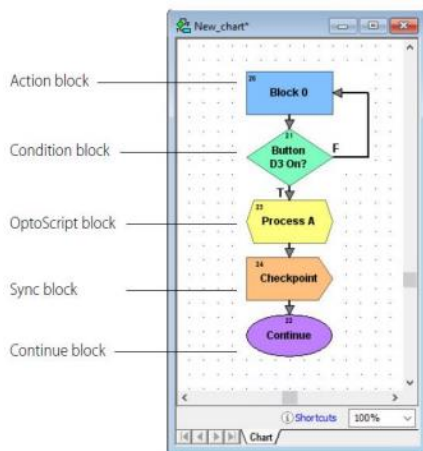


Figura 23. Bloques PAC Control

- **Bloque de acción:** Son rectangulares y realizan el trabajo de la estrategia, como encender o apagar las cosas, o establecer variables. Pueden tener muchas entradas, pero solo una salida.
- **Bloque de condición:** Tienen forma de diamante y contienen preguntas o condiciones para controlar el flujo lógico de la estrategia. Pueden tener muchas entradas, pero solo dos salidas, True o False. Pueden contener más de una condición con las funciones AND (Y) y OR (O), con la primera deberán ser las dos condiciones ciertas, mientras que con la segunda con sólo una validez.
- **Bloque de continuación:** Forma ovalada. No tienen ninguna función en el flujo, simplemente almacenan el nombre del siguiente bloque a ejecutar. Pueden tener varias entradas, pero ninguna salida.
- **Bloque de OptoScript:** Bloques hexagonales que contienen código OptoScript, como su nombre indica. Este lenguaje se utiliza para simplificar tareas como cálculos matemáticos, manejo de cadenas y ciclos y condiciones complejas. Tienen más de una entrada, pero ninguna salida.
- **Bloque de sincronización:** Solamente estará visible cuando la compatibilidad con la redundancia del controlador está habilitada. Sincroniza controladores redundantes en ubicaciones estratégicas en un gráfico con estos bloques



Figura 24. Logo Groov EPIC

Por otro lado, está Groov View. Esto es una herramienta de interfaz de operador muy fácil de utilizar y que está conectada a infinidad de sistemas, simplemente necesita un navegador web para poder conectarse a dispositivos, crear la interfaz y monitorizar o controlar dichos dispositivos. En su interfaz se pueden utilizar etiquetas o tags de cualquier sensor, equipo y software que esté conectado al sistema Grov EPIC.

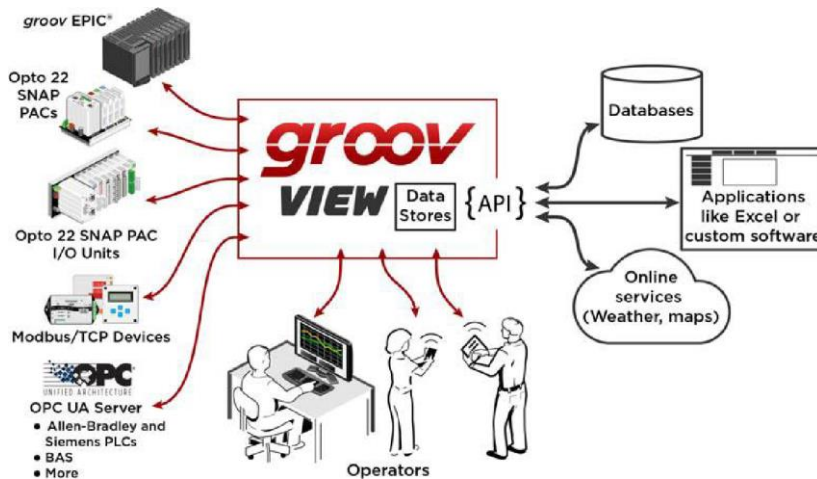


Figura 25. Conexión en Groov View

Groov View se puede conectar a distintos tipos de fuentes de datos y los visualiza para los usuarios. En la imagen se puede ver como Groov View conecta las fuentes de datos y los usuarios. A la izquierda de la foto se ven las fuentes de datos, mientras que a la derecha están los usuarios que escriben datos en un almacén de datos Groov. En un sistema de Groov EPIC también se puede tener etiquetas de Node-RED, Ignition, OPC UA y MQTT como fuentes de datos. Por otro lado, en la parte inferior de la foto se encuentran los operadores, que interactúan con la interfaz de Groov View y pueden gestionar a todos los usuarios, además de elegir qué se puede y qué no se puede ver o hacer.

2.3. Descripción de distintas bases de datos en la nube

Aparte de IBM Cloud, hay infinidad de servicios que te ofrecen bases de datos en la nube, los cuales también hay que tener en cuenta ya que cada una tiene unas determinadas características, distintos servicios y, evidentemente, distintos precios que hay que considerar antes de lanzarse a un nuevo proyecto de esta categoría.

Se podría estar hablando de una innumerable cantidad de bases de datos en la nube, pero no es la intención. Por tanto, se comentarán tres de las más importantes del mercado, aparte de la IBM que ha sido explicada con anterioridad. Se explicarán brevemente los servicios de Microsoft Azure, Amazon RDS y Google Cloud SQL. Hay más opciones, pero esas son unas de las más conocidas y usadas tanto dentro como fuera de la industria.

2.3.1. Microsoft Azure



Figura 26. Logo Microsoft Azure

La plataforma de Microsoft Azure consta de más de 200 productos y servicios en la nube que están diseñados para proponer soluciones a una infinidad de problemas actuales y en el futuro. Se pueden crear, diseñar, ejecutar un gran abanico de aplicaciones desde la nube. Hay una gran cantidad de opciones, desde programar en Linux y en nginx, que es un servidor web de código abierto, hasta crear las bases de datos que se desee tanto de Linux como de Windows, y conectarlas a una base de datos compartida. También existe una herramienta, Cognitive Service, que sirve para crear un IA bot, lo cual puede llegar a ser extremadamente útil para determinadas aplicaciones o programas.

Una de las grandes ventajas de Azure es que tiene una interfaz realmente sencilla de utilizar, por lo que no habría problemas de entendimiento, incluso siendo principiante. Además, es extremadamente fácil conectar este servicio con otras tecnologías. Por ejemplo, PowerBI es capaz de extraer información de su base de datos de una manera bastante rápida y cómoda. Por otro lado, si se quiere conectar con otros dispositivos, gracias a Node-RED es extremadamente fácil, con el módulo "Azure IoT Hub".

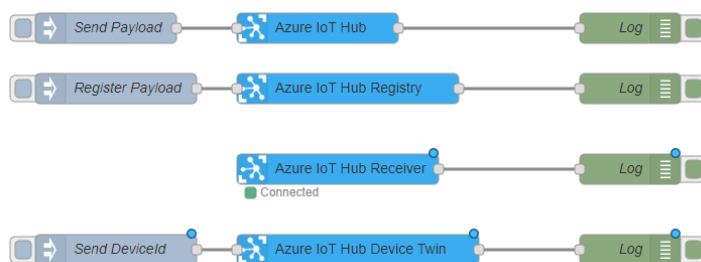


Figura 27. Ejemplo de comunicación de Node-RED con Azure

2.3.2. Amazon RDS

Amazon RDS es un servicio de base de datos con seis motores distintos de base de datos, se puede elegir cualquiera de ellos: Amazon Aurora, MySQL, MariaDB, PostgreSQL, Oracle y Microsoft SQL Server. Por tanto, con este servicio, se puede utilizar prácticamente cualquier código o aplicación. Se encarga de las tareas habituales de las bases de datos, como la búsqueda y reparación de errores, copias de seguridad, etc.



Figura 28. Logo Amazon RDS

Las principales características de este servicio son:

- **Menor carga administrativa:** Es muy fácil de utilizar, se puede usar la consola de administración de AWS, la interfaz de línea de comandos de Amazon RDS o hacer llamadas a la API de una manera muy sencilla. Además, Amazon RDS garantiza que la base de datos esté siempre actualizada con los últimos parches.
- **Rendimiento:** El almacenamiento es una opción de almacenamiento SSD que ofrece una línea de base de 3 IOPS por cada GB, y con capacidad de ráfagas de hasta 3000 IOPS por encima del valor de la línea de base.
- **Escalabilidad:** Es capaz de escalar los recursos para ampliar o reducir la implementación hasta los 32 vCPU y 244 GiB de RAM. El escalado es cuestión de minutos. Todos los motores mencionados anteriormente ofrecen un máximo de 64 TB de almacenamiento, excepto Microsoft SQL Server, que ofrece hasta las 16 TB.
- **Disponibilidad y durabilidad:** La función de copia de seguridad se puede configurar hasta un máximo de 35 días. Las implementaciones Multi-AZ mejoran la disponibilidad y durabilidad de las instancias de base de datos, por lo que se hacen más apropiadas para cargas de trabajo de bases de datos de producción.
- **Seguridad:** Amazon RDS soporta el uso de SSL para proteger los datos de tránsito. Además, admite el cifrado de datos transparente en SQL Server y Oracle, estando el segundo integrado con AWS CloudHSM.
- **Manejabilidad:** Ofrece métricas de Amazon CloudWatch sin ningún cargo adicional, utilizando la consola de administración de RDS. También ofrece monitorización optimizada, con acceso a más de 50 métricas de CPU, memoria, E/S y rendimiento. Además, también envía notificaciones vía email o SMS mediante Amazon SNS.

2.3.3. Google Cloud SQL

Google Cloud SQL es una de las soluciones que proporciona Google para la gestión de base de datos. Consiste en la gestión de base de datos como MySQL, PostgreSQL y Microsoft SQL Server. Con este servicio se pueden usar las herramientas Datastore o Firestone, aunque antes había que familiarizarse con el Marketplace.



Figura 29. Logo Google Cloud SQL

Las características principales y ventajas de este servicio son las siguientes:

- **Seguridad y cumplimiento:** Promete el encriptado de todos los datos, estando tanto en reposo como en tránsito, conectividad privada con la nube virtual y acceso a la red protegido mediante firewall. Por otro lado, cumple los estándares SSAE 16, ISO 27001, PCI DSS y HIPAA.
- **Escalabilidad a la medida del usuario:** Con una llamada a la API es más que suficiente para escalar las instancias del usuario.
- **Solución gestionada:** Este servicio garantiza que las bases de datos sean fiables, seguras y escalables, además de automatizar la copia de seguridad, instalación de parches de encriptado y el aumento de la capacidad. Tiene disponibilidad en casi toda la totalidad del mundo.
- **Solución integrada:** Se puede acceder a las instancias desde cualquier aplicación, como App Engine, Compute Engine, y muchas más. Con BigQuery se puede consultar directamente las bases de datos de datos de Cloud SQL.
- **Migraciones sencillas a Cloud SQL:** Database Migration Service (DMS) permite migrar la base de datos de producción a Cloud SQL con un tiempo mínimo de inactividad. No es necesario ni aprovisionar, gestionar o monitorizar los recursos de la migración, ya que DMS usa las funciones nativas de replicación de MySQL y PostgreSQL para que el proceso sea lo más fiable y fiel posible. Este proceso no requiere de ningún coste adicional.

2.4. Edge Computing. Distintas pasarelas

A continuación, en este apartado, se explicarán brevemente distintas pasarelas que se podían utilizar en sustitución del IOT 2050. Si bien es cierto que la pasarela escogida es una de las mejores opciones del mercado, tiene un coste que hay que asumir, y a veces eso no es posible. Por tanto, es conveniente mencionar y comentar las características principales de otras opciones que podrían salvar de más de un apuro en función de la exigencia del proyecto.

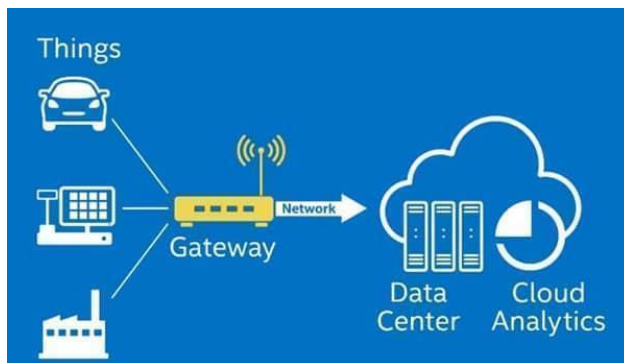


Figura 30. Utilidad de Gateway. Imagen de Aprendiendo Arduino

2.4.1. IOT 2020/40

En primer lugar, se va a hablar de la familia de IOT. Para este trabajo se ha elegido el IOT 2050, pero también existen los IOT 2020 e IOT 2040. Estas opciones, si bien son menos potentes que la que se ha seleccionado, son también más económicas. Por tanto, si el proyecto no es tan exigente con la pasarela, puede ser que sea una buena opción rebajar las prestaciones y así abaratar costes.



Figura 31. IOT 2020

Como se ha comentado, la diferencia de presupuesto es notable. Por un lado, tenemos el IOT 2020, que cuesta en torno a los 180€, luego está el IOT 2040, costando aproximadamente 220€, solamente unos 40€ de sobrecoste en comparación. Sin embargo, si se mira el precio del IOT 2050, se puede ver que cuesta cerca de los 430€, por lo que se estaría hablando de que duplica el precio del IOT 2040, por lo que puede no ser una muy buena opción si la programación a realizar no es muy demandante.

Las principales características de estas dos pasarelas se van a mostrar en la tabla siguiente:

Características	IOT 2020	IOT 2040
<i>CPU</i>	Intel Quark x1000 (x86 400 MHz)	Intel Quark x1000 (x86 400 MHz) con funciones de seguridad
<i>Capacidad de memoria</i>	256 kB, 512 MB, 8 FLASH MB	256 kB, 1 GB, 8 FLASH MB
<i>Tipo de memoria</i>	DDR3 RAM, SRAM	DDR3 RAM, SRAM
<i>Protocolo de comunicación</i>	Ethernet	Ethernet
<i>Tipo de puerto COM</i>	Ethernet RJ45	Ethernet RJ45, RS232/485 conmutable
<i>Extensión</i>	Arduino, mPCIe	Arduino, mPCIe
<i>Características integradas</i>	LED	Batería RTC en búfer, LED, Vigilancia
<i>Temperaturas de funcionamiento</i>	0°C a 50°C	0°C a 50°C

Tabla 3. Características IOT 2020 e IOT 2040

2.4.2. Raspberry Pi 4

Raspberry Pi es un microordenador que tiene capacidad de realizar una ingente cantidad de tareas a pesar de tener un precio base de solamente 35 \$, por lo que puede ser una opción interesante para la Industria 4.0, ya que es capaz de correr Node-RED y, con su precio y versatilidad con autómatas que no sean solo de SIEMENS, puede ser una pasarela a tener en consideración. Por tanto, se ha considerado necesario mencionar brevemente esta placa, ya que puede ser muy útil para proyectos futuros y de bajo coste y requerimiento de hardware.



Figura 32. Raspberry Pi 4

Las principales características de este dispositivo son las siguientes:

Características	Raspberry Pi 4
<i>CPU</i>	ARM Cortex-A72
<i>Frecuencia de reloj</i>	1.5 GHz
<i>GPU</i>	VideoCore VI (con soporte para OpenGL ES 3.x)
<i>Memoria</i>	LPDDR4 SRAM 1 GB a 35 \$ 2 GB a 45 \$ 4 GB a 55 \$
<i>Conectividad</i>	Bluetooth 5.0

	Wi-Fi 802.11ac Gigabit Ethernet
<i>Puertos</i>	GPIO 40 pines 2x micro HDMI 2x USB 2.0 2x USB 3.0 CSI (cámara Raspberry Pi) DSI (panel táctil) Micro SD Conector de audio Jack USAB-C (alimentación)

Tabla 4. Características Raspberry Pi 4

2.4.3. ESP 32 y ESP 8266

Los módulos ESP 32 y ESP 8266 son módulos Wi-Fi y Bluetooth (el ESP 8266 no) son una opción aún más económica que permiten subir datos a bases de datos en la nube. Para el caso de estas herramientas, también se podría usar Node-RED, pero hay infinidad de aplicaciones web que podrían desempeñar la función de visualización rápida de datos como Thinger.io, Adafruit y Blink, entre otras. Son opciones mucho menos potentes que el resto, pero para aplicaciones de IOT pequeñas o en casa podrían ser extremadamente interesantes, ya que tienen un muy bajo coste, ambas inferior a 10€.



Figura 33. ESP 32

Las principales características de ambas placas son las siguientes que se muestran en la tabla:

<i>Características</i>	<i>ESP 32</i>	<i>ESP 8266</i>
<i>Procesador</i>	32 bit Xtensa LX6 Dual-Core	32 bit Xtensa LX6 1 core
<i>Frecuencia de reloj</i>	160 a 240 MHz	80 MHz
<i>SRAM</i>	520 kB	50 kB
<i>Conectividad</i>	Wi-Fi 802.11 b/g/n hasta 150 Mbps Bluetooth v4.2 BR/EDR y BLE	Wi-Fi 802.11 b/g/n hasta 72.2 Mbps
<i>GPIO</i>	36	17
<i>DAC</i>	2 canales de 8 bits	No
<i>ADC</i>	SAR de 12 bits	SAR de 10 bits

<i>SPI/I2C/I2S/UART</i>	4/2/2/3	2/1/2/2
<i>Sensor táctil</i>	Sí (8 canales)	No
<i>Sensor temperatura</i>	Sí	No
<i>Protocolos de red</i>	IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT	IPv4, TCP/UDP/HTTP/MQTT
<i>Temperatura de funcionamiento</i>	-40°C / 85°C	-40°C / 125°C

Tabla 5. Características ESP 32 y ESP 8266

2.5. Familia de PLC SIMATIC de SIEMENS

Para la realización del proyecto, como bien se ha comentado al principio de la memoria, se va a trabajar con un autómata de SIEMENS, más concretamente el SIMATIC S7-1200. Sin embargo, desde la misma marca, hay distintas gamas en la misma familia de autómatas que sería interesante conocer o tener en consideración para diferentes tipos de proyectos (desde proyectos más grandes u optimistas hasta otros mucho más sencillos). Por lo tanto, en este apartado se van a mencionar y explicar brevemente el resto de la familia de autómatas de SIEMENS, mencionando distintos modelos existentes para cada gama. El S7-1200 ya fue explicado anteriormente, por lo que no se va a volver a mencionar aquí. Los autómatas que se van a mostrar a continuación son los siguientes: S7-200, S7-300, S7-400 y S7-1500, teniendo cada uno distintos modelos con distintas características.

2.5.1. S7-200

Es la gama más baja de la familia SIMATIC de SIEMENS. Con este autómata se pueden realizar distintas aplicaciones de control. Hay varios modelos a elegir, que son los siguientes: CPU 221, CPU 222, CPU 224, CPU 224XP, CPU 226. Estos modelos ofrecen soporte para la CPU, con opción de desactivado de edición de funcionamiento para así tener más memoria de programas. El CPU 224XP soporta entradas y salidas analógicas en placas, además de tener dos puertos de comunicaciones, mientras que el CPU 226 incluye filtros de entrada y captura de pulsos.

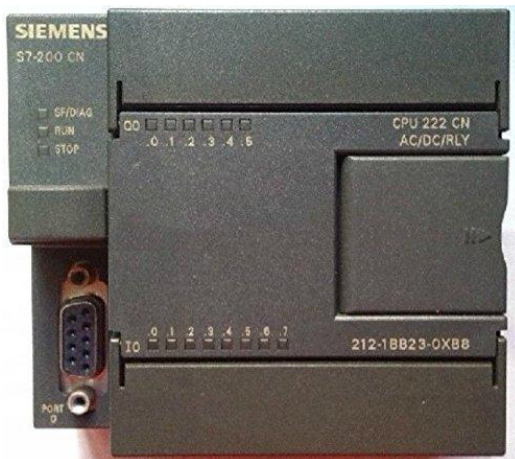


Figura 34. PLC SIMATIC S7-200

Las principales características son:

Características	SIMATIC S7-200
<i>Tensión</i>	120 Vac, 230 Vac
<i>Puerto de comunicación</i>	RS 485
<i>Memoria</i>	2 kB (de datos) y 4 kB (de programa)
<i>Idioma de programación utilizado</i>	AWL, FUP, Ladder Logic
<i>Corriente de salida</i>	2 A
<i>Temperatura máxima de funcionamiento</i>	+ 45°C
<i>Número de puertos de comunicación</i>	1

Tabla 6. Características SIMATIC S7-200

2.5.2. S7-300

Esta gama de autómatas procesa datos e instrucciones de una manera rápida y clara. Dispone de 24 entradas digitales y 4 analógicas, por lo que puede recibir medidas físicas y proporciona un amplio abanico de posibilidades a la hora de programar. Es muy práctico para la automatización industrial, robótica y automoción, ya que dispone de 4 contadores procesando datos a 60kHz por segundo. Sus dimensiones son muy pequeñas, siendo de 120 mm (A) x 125 mm (L) X 130 mm (P), por lo que puede ser una muy buena opción en usos donde el espacio sea muy limitado. Su montaje e instalación es muy fácil, en un carril DIN.



Figura 35. PLC SIMATIC S7-300

Sus principales características son las siguientes:

Características	SIMATIC S7-300
Número de entradas	24 (digital) y 4 (analógicas)
Número de salidas	16 (digital) y 2 (analógicas)
Tipo de Red	USB
Tipo de puerto de comunicación	RS 442 y RS 485
Categoría de tensión	24 Vdc
Corriente de salida	500 mA
Memoria	192 kB (integrados) y 8 MB (de carga)
Tipo de montaje	Carril DIN
Lenguaje de programación utilizado	FDB, GRAPH, HiGraph®, LAD, SCL
Rango de temperatura de funcionamiento	0°C / +60°C

Tabla 7. Características SIMATIC S7-300

2.5.3. S7-400

Estos autómatas sirven para cualquier tipo de industrias, con un gran rendimiento para crear soluciones. Es usado principalmente en industrias de manufactura y de procesos. Esta serie es la ideal

para realizar tareas con un tráfico de grandes cantidades de datos, ya que tiene una velocidad de procesamiento muy alta, además de unos tiempos de operación muy pequeños, que ayuda a reducir los tiempos de ciclo de cualquier máquina en todo tipo de industria. Además, El S7-400 integra un BU, de alta velocidad, por lo que realiza una conversión muy eficaz de los módulos periféricos centrales. La programación se realiza a través de TIA portal y existen cuatro modelos en esta gama, que son: S7 412, S7 414, S7 416 y S7 417.



Figura 36. PLC SIMATIC S7-400

Sus principales características son las siguientes:

Características	SIMATIC S7-400
<i>Número de salidas</i>	16
<i>Longitud de cable</i>	Sin pantalla: máx. 600 m Con pantalla: máx. 1000 m
<i>Tensión nominal de carga L1</i>	De 20 V a 128 Vdc
<i>Protección a inversión de polaridad</i>	Sí, mediante fusibles
<i>Intensidad total de la salida</i>	Hasta 40°C: máx. 16 A (21 A con bandeja de ventiladores) Hasta 60°C: máx. 8 A (14 A con bandeja de ventiladores)
<i>Aislamiento galvánico</i>	Sí, entre los canales y el bus posterior
<i>Diferencia de potencial admisible</i>	250 Vac
<i>Potencia disipada del módulo</i>	típ. 10 W
<i>Retardo de salida</i>	De '0' a '1': 2 ms De '1' a '0': 13 ms
<i>Fusibles de repuesto</i>	Sí, de 8 A y 250 V
<i>Frecuencia de conmutación</i>	Bajo carga óhmica: máx. 10 Hz Bajo carga inductiva: máx. 0,5 Hz

Tabla 8. Características SIMATIC S7-400

2.5.4. S7-1500

Esta es la gama más alta de la familia de autómatas SIMATIC. Está perfectamente integrado con TIA Portal, por lo que ofrece una ingeniería con una eficiencia máxima. Ofrece un rendimiento del sistema avanzado, además de PROFINET como interfaz integrada estándar. En esta gama se puede instalar, conectar, incluso ponerse en servicio, de una manera fácil y rápida. Es el autómata con los tiempos de reacción más cortos, y con un tiempo de procesamiento de comandos de hasta 1ns en la CPU.

En el S7-1500 la información detallada de errores está en la funcionalidad de diagnóstico, que está integrado en el sistema del mismo autómata. Estos errores se visualizan como información de texto sin formato, que ofrece total transparencia de la planta. Esta gama, además, tiene integrado el sistema SIMATIC Safety Integrated, para aplicaciones estándar y de seguridad, permitiendo la conexión entre dispositivos PROFIsafe Geräten mediante PROFIBUS y PROFINET.

En esta gama hay tres distintos modelos: S7 1511, S7 1513 y S7 1516.



Figura 37. PLC SIMATIC S7-1500

Sus principales características son las siguientes:

Características	SIMATIC S7-1500
<i>Grado de protección</i>	IP20
<i>Temperatura ambiente</i>	Montaje horizontal: hasta 60°C (50°C con pantalla) Montaje vertical; hasta 40°C
<i>Humedad relativa</i>	De 5% a 95% sin condensación
<i>Pérdidas típicas de potencia</i>	5,7 W (7 W el S7-1516)
<i>Tensión de alimentación</i>	24 Vdc
<i>Memoria</i>	S7-1511: 150 kB (integrada) y 1 MB (datos) S7-1513: 300 kB (integrada) y 1,5 MB (datos) S7-1516: 1 MB (integrada) y 5 MB (datos)
<i>Memoria de carga</i>	2 GB

<i>Número de contadores S7</i>	2048
<i>Número de temporizadores S7</i>	2048
<i>Número de puertos</i>	2
<i>RJ 45 (Ethernet)</i>	Sí
<i>PROFINET IO-Controller</i>	Sí
<i>PROFINET IO-Device</i>	Sí
<i>Tipo de puerto de comunicación</i>	RS 485
<i>Número máximo de conexiones</i>	S7-1511: 96 S7-1513: 128 S7-1516: 256
<i>Lenguajes de programación</i>	KOP, FUP, AWL, SCL

Tabla 9. Características SIMATIC S7-1500

3. EXPLICACIÓN DEL PROBLEMA, SOLUCIÓN A REALIZAR Y ARQUITECTURA

En este apartado se comentará la idea y el procedimiento que se va a tomar para realizar el proyecto. Incluyendo una breve descripción del programa que se va a cargar en el autómata, y una explicación de cómo se va a realizar la subida de datos a la nube para tenerlos monitorizados en todo momento. No se va a mostrar el procedimiento realizado, eso se hará posteriormente. Simplemente se explicará paso por paso y de una manera breve qué es lo que se quiere realizar y las metas a lograr en este proyecto.

3.1. Funcionamiento del programa cargado en el autómata

Se realizará un breve programa en TIA Portal simulando una cinta en el que se van transportando cajas hasta llenar un palé de capacidad para diez cajas. Mediante un contador (que simula la cantidad de cajas) y un reloj a una frecuencia de 1 Hz para simular la frecuencia que van pasando las cajas, se simulará el llenado de palés. Posteriormente, con otro contador, se irá acumulando el número de palés que se van completando. Además, se incluirán distintas variables como podría ser interruptores de seguridad, parada de emergencia y demás.

Como se puede ver, el programa va a ser simple, ya que no es el objetivo principal del proyecto, lo que verdaderamente interesa es la subida y monitorización de datos en todo momento.

3.2. Solución a realizar

Una vez realizado el programa en TIA Portal, el siguiente paso es comunicar esto con Node-RED. Para ello, primero, el autómata debe estar conectado a la misma red local que la pasarela IOT 2050. Por lo tanto, se le debe asignar la dirección IP 192.168.200.X, al igual que en el Gateway. En este caso, se le asigna la dirección 192.168.200.5, siendo la misma máscara de subred también, la 255.255.255.0.

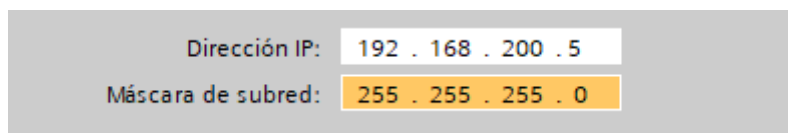


Figura 38. Configuración de la conexión con el autómata

Con el autómata estando en la misma red local que la pasarela, ya se puede comunicar y subir todo tipo de datos e información a Node-RED. Aparte de las funciones básicas ya comentadas, los nodos que se van a usar son los de comunicación con el PLC, “s7 in” principalmente, ya que el único interés en este proyecto es en el de subir datos a la nube, no el de mandar información o controlar el autómata de manera remota. Esto, si se quisiera hacer, habría que hacerlo con el nodo “s7 out” para enviar información a una variable del autómata y el de “s7 control” para realizar la función a enviar requerida.

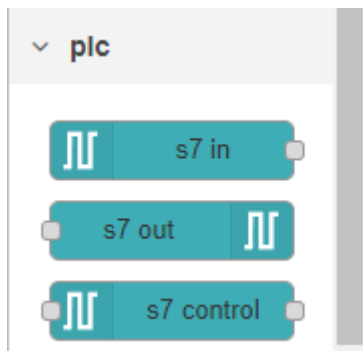


Figura 39. Nodos de SIEMENS

Otros aspectos a tener en cuenta son los que se muestran en la siguiente imagen, los nodos de entrada y salida para realizar la conexión con IBM Watson y la base de datos de IBM Cloud. El primer nodo sirve para enviar la información a IBM Cloud, mientras que el segundo (que no se va a utilizar en este proyecto) serviría para recibir instrucciones de la propia base de datos, lo que combinándolo con “s7 out”, por ejemplo, se podría monitorizar o controlar cualquier máquina de manera remota. Gracias a estos nodos, se podrá enviar toda la información deseada desde el autómata hasta la base de datos de IBM Cloud y, por lo tanto, poder visualizarlos desde el dashboard del servicio que ofrecen con sus paneles de visualización.

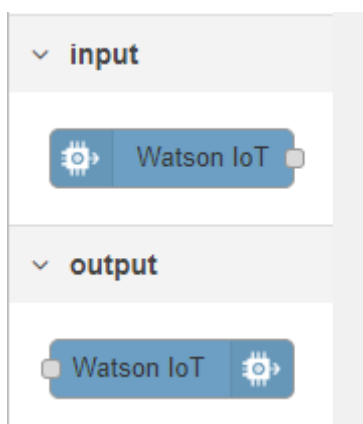


Figura 40. Nodos de IBM Cloud

3.2.1. Esquema

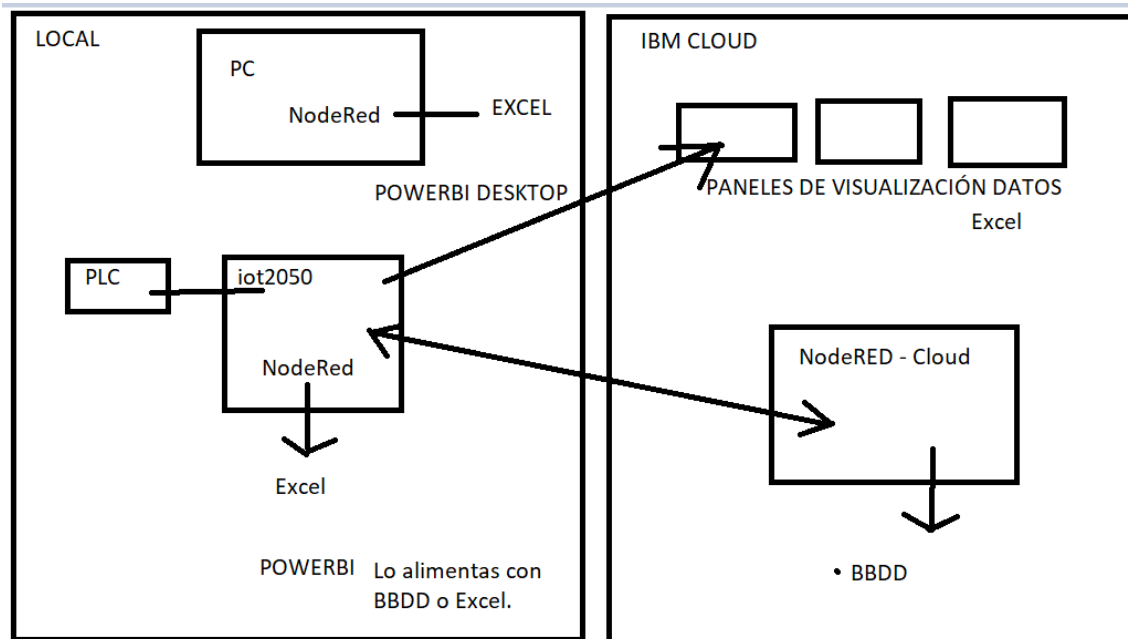


Figura 41. Esquema del proyecto

Este sería el esquema a seguir para realizar el proyecto. La parte de la izquierda está todo el hardware conectado a una misma red local, siendo el dibujo de arriba la ejecución de Node-RED desde cualquier ordenador, que se explicará posteriormente, y lo segundo la ejecución de Node-RED desde la pasarela IOT 2050, que es más complejo, pero también se explica al completo en este proyecto. El objetivo del trabajo consiste en enviar los datos del autómatas a los paneles de visualización de IBM Cloud. Para ello, hay que utilizar algunos nodos mencionados previamente. La otra vía consiste en conectar el Node-RED de la pasarela con IBM Watson, que ejecuta otro Node-RED, esta vez en la nube, lo cual expandirá mucho más el proyecto, y se podría intercalar con otros distintos.

Para terminar, se pueden subir y procesar los datos con PowerBI para hacer un estudio de negocio y ver los valores históricos que se han ido almacenando.

4. SOLUCIÓN PROPUESTA

4.1. Conexión a Node-RED desde el ordenador y desde la pasarela IOT 2050

4.1.1. Conexión a Node-RED desde un ordenador

Hay muchas vías para conectarse a Node-RED, como se ha visto en el esquema del apartado anterior. Una de ellas, es desde el propio PC en el que se está trabajando. No es necesaria para realizar el proyecto, pero se va a realizar también debido a que también se puede trabajar desde ahí.

Lo primero que se tiene que hacer es pulsar la tecla Windows+R, y así se abrirá la pestaña de ejecución de comandos. En esta se escribe "cmd" y se abrirá la consola.

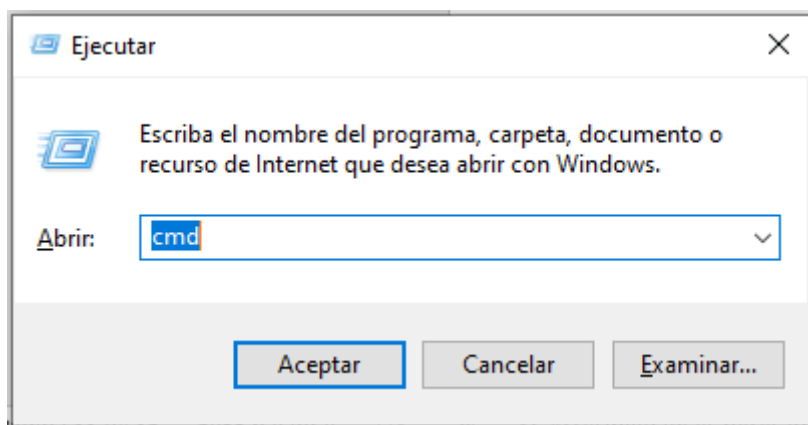
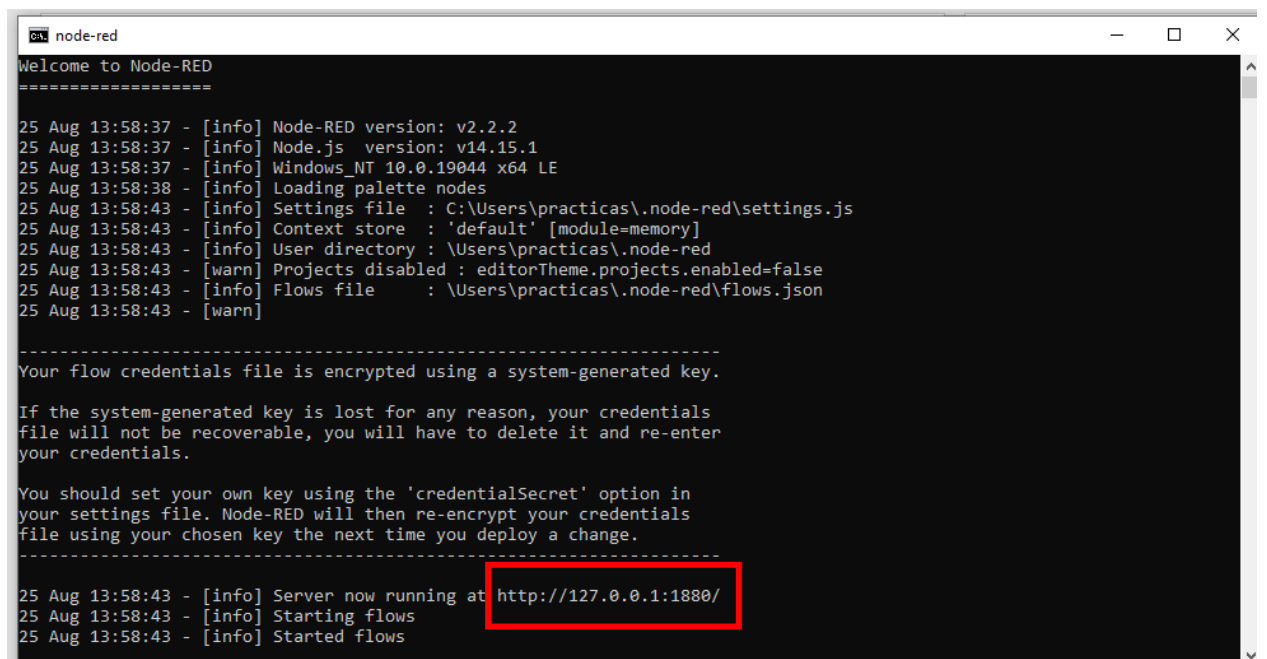


Figura 42. Acceso al cmd

En el caso del ordenador donde se ha realizado el ejemplo, Node-RED ya está instalado, por lo que en la captura no se muestra. Si se necesitara instalar, simplemente hay que escribir el comando "npm install -g - -unsafe-perm node-red". Una vez hecho eso, se ejecuta escribiendo "node-red".



```
node-red
Welcome to Node-RED
=====
25 Aug 13:58:37 - [info] Node-RED version: v2.2.2
25 Aug 13:58:37 - [info] Node.js version: v14.15.1
25 Aug 13:58:37 - [info] Windows_NT 10.0.19044 x64 LE
25 Aug 13:58:38 - [info] Loading palette nodes
25 Aug 13:58:43 - [info] Settings file : C:\Users\practicas\.node-red\settings.js
25 Aug 13:58:43 - [info] Context store : 'default' [module=memory]
25 Aug 13:58:43 - [info] User directory : \Users\practicas\.node-red
25 Aug 13:58:43 - [warn] Projects disabled : editorTheme.projects.enabled=false
25 Aug 13:58:43 - [info] Flows file : \Users\practicas\.node-red\flows.json
25 Aug 13:58:43 - [warn]

-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----
25 Aug 13:58:43 - [info] Server now running at http://127.0.0.1:1880/
25 Aug 13:58:43 - [info] Starting flows
25 Aug 13:58:43 - [info] Started flows
```

Figura 43. Cargando Node-RED

Como se puede apreciar en la imagen, hay una línea que dice que el servidor está funcionando en la dirección: <http://127.0.0.1:1880/>, por lo que simplemente se copia esa dirección en un buscador cualquiera. Los nodos instalados se ven en la parte de la izquierda, donde se pueden ver los más comunes como “inject”, “debug”, “switch” o “function”, entre muchos otros.

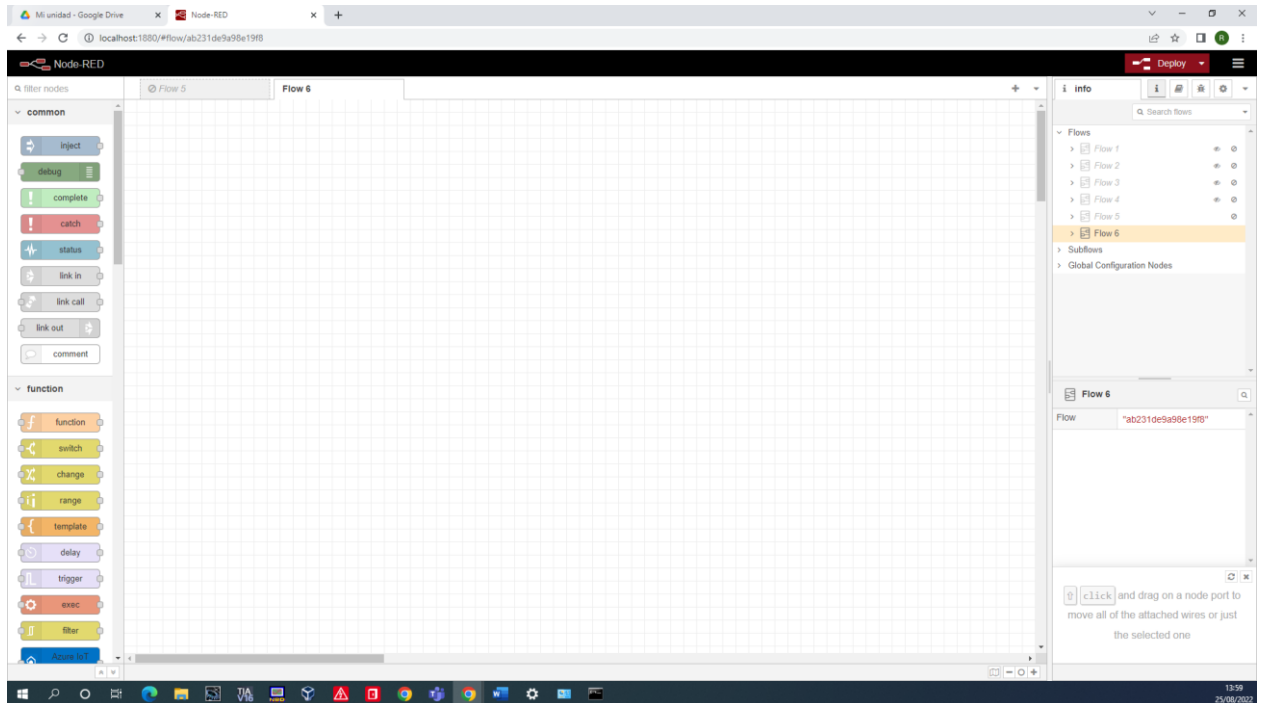


Figura 44. Entorno Node-RED desde PC

Si se quisieran instalar otros nodos, ya que hay infinidad de funciones, hay que dirigirse a la pestaña de arriba a la derecha y clicar en “Manage palette”, y buscar los nodos en el buscador que aparecerá, como se ve en la siguiente imagen. Habrá que buscar e instalar los nodos de “S7 in”, para comunicarse y recibir información del PLC, además del nodo “IBM IoT Device”, para conectarse con el servidor de IBM Cloud.

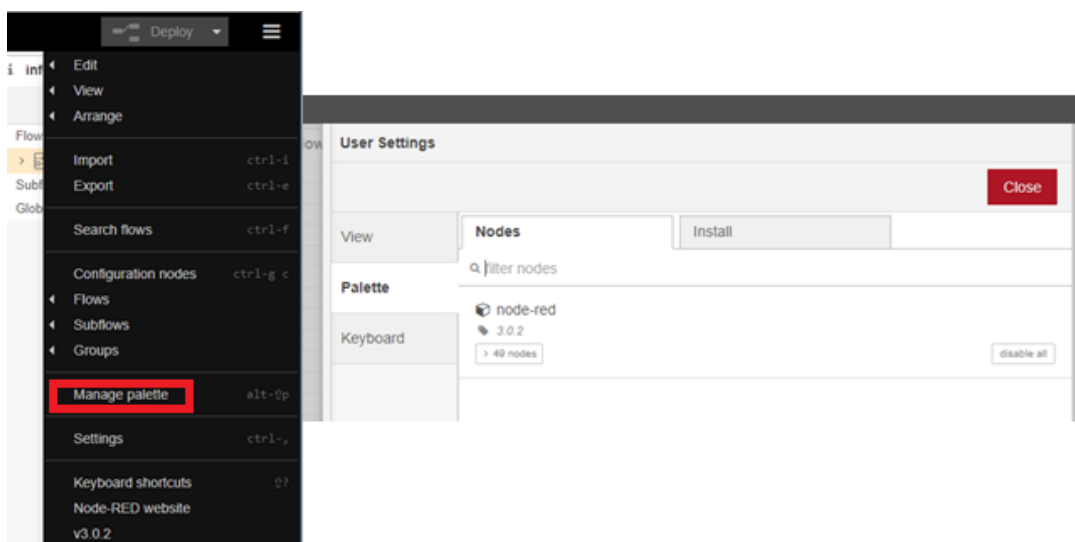


Figura 45. Descarga de nodos

4.1.2. Conexión a Node-RED y configuración de la pasarela IOT 2050

4.1.2.1. Configuración de la pasarela IOT 2050

La manera explicada anteriormente, aunque si bien es cierto que puede proporcionar muchas soluciones, no es la más indicada para este proyecto, ya que interesa no tener un dispositivo, en este caso un ordenador de por medio. Además, ya que se está usando una pasarela del calibre como es la IOT 2050, que es de las mejores del mercado, es conveniente aprovecharla y sacarle el mayor partido posible.

Por lo tanto, en este apartado se va a exponer cómo ejecutar Node-RED a través de una pasarela de SIEMENS, siendo igual de aplicable para la IOT 2020 e IOT 2040.

Lo primero que se hace es alimentar la pasarela e insertar una tarjeta microSD como se ve en la siguiente imagen. La pasarela se alimenta con 24 voltios. En la página de SIEMENS proporcionan instrucciones de cómo realizar este paso. Por otro lado, hay que instalar la imagen en el microSD que se encuentra dentro de la pasarela para poder utilizarlo.

Para realizar la instalación de la imagen del IOT 2050 en la tarjeta microSD, lo primero que hay que asegurarse es que esta esté vacía, si no habría que formatearla y dejarla como nueva. Una vez eso, se descarga la imagen desde la página de SIEMENS, con el siguiente enlace:

<https://support.industry.siemens.com/cs/document/109741799/downloads-for-simatic-iot20x0?dti=0&lc=en-WW>

Luego, se tiene que descargar el archivo .zip necesario para la pasarela, que en este caso es el siguiente:

https://support.industry.siemens.com/cs/attachments/109741799/IOT2050_Example_Image_V1.2.2.zip

Posteriormente se tiene que descomprimir y utilizar descargar el programa *Win32 Disk Imager* para poder montar la imagen en la unidad microSD. Este programa se puede descargar desde el siguiente enlace: <https://sourceforge.net/projects/win32diskimager/>. Una vez descomprimido el archivo y teniendo el programa, se introduce la dirección donde se ha guardado la imagen y dónde se quiere escribir. Una vez realizado esto se introduce la tarjeta en el slot y se alimenta la pasarela como se puede observar en las siguientes imágenes.

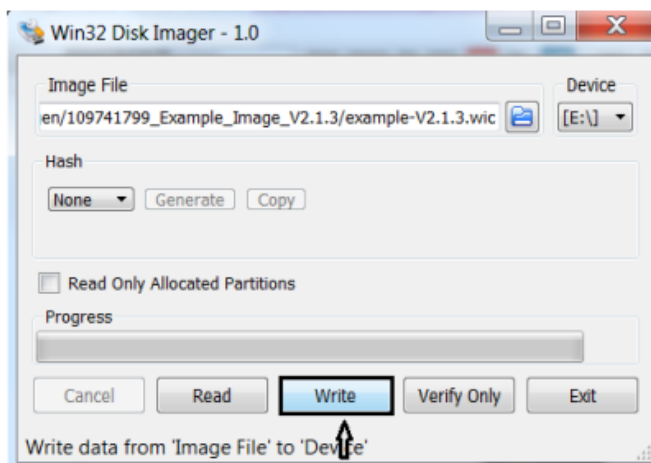


Figura 46. Escritura de imagen en IOT 2050



Figura 47. Alimentación IOT 2050

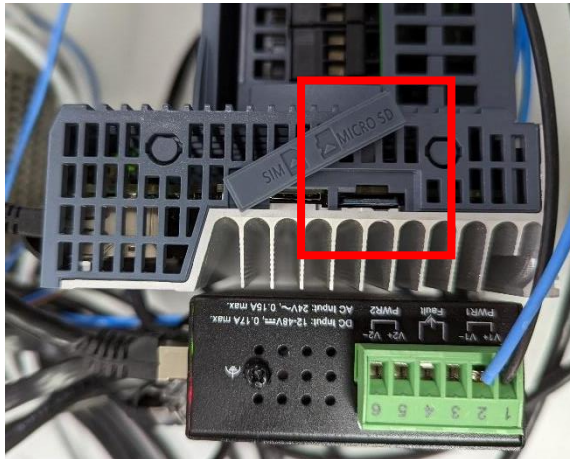


Figura 48. Introducción de microSD en IOT 2050

Ya se ha alimentado la pasarela, por lo que el led del SD debe parpadear, y una vez deje de hacerlo querrá decir que se ha completado la instalación de la imagen en el IOT 2050.

Para poder establecer conexión por primera vez hay que hacerlo de manera local. La primera vez que se instala la tarjeta SD, el puerto Ethernet de comunicación X1 P1 LAN tiene la dirección IP 192.168.200.1. Si se quiere establecer conexión, hay que configurar la dirección IP del puerto Ethernet de la siguiente manera: 192.168.200.X. La X puede ser cualquier número excepto el definido para la pasarela.

Los dos primeros grupos de una dirección IP (192.168) son los identificadores de red, mientras que los dos últimos son los identificadores del HOST. Para poder tener conexión entre múltiples dispositivos, los tres primeros grupos tienen que ser iguales, en este caso 192.168.200.X. El último grupo debe ser distinto, ya que sirve para identificar cada dispositivo en la conexión.

Para configurar la IP del puerto Ethernet hay que ir a conexiones de red, concretamente en el apartado de "Ethernet". A continuación, hay que ir a "Cambiar opciones del adaptador", para acabar seleccionando las propiedades de "Realtek USB GbE Family Controller". A continuación, se hace doble clic en "Protocolo de Internet versión 4 (TCP/IPv4)". En esa pestaña se establece la dirección IP 192.168.200.74 y la máscara de subred 255.255.255.0, tal y como se puede apreciar en la siguiente imagen.

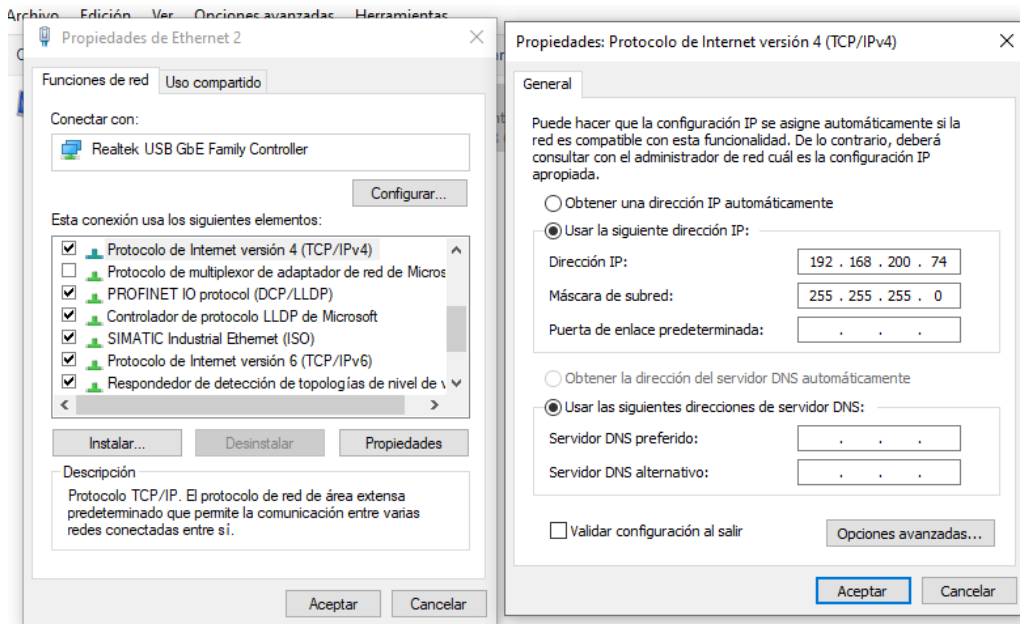


Figura 49. Configuración Ethernet para IOT 2050

Una vez realizado esto, hay que instalar el software de comunicación por protocolo SSH “PuTTY”. Cuando se tiene instalado, hay que añadir la dirección IP de la pasarela: 192.168.200.1.

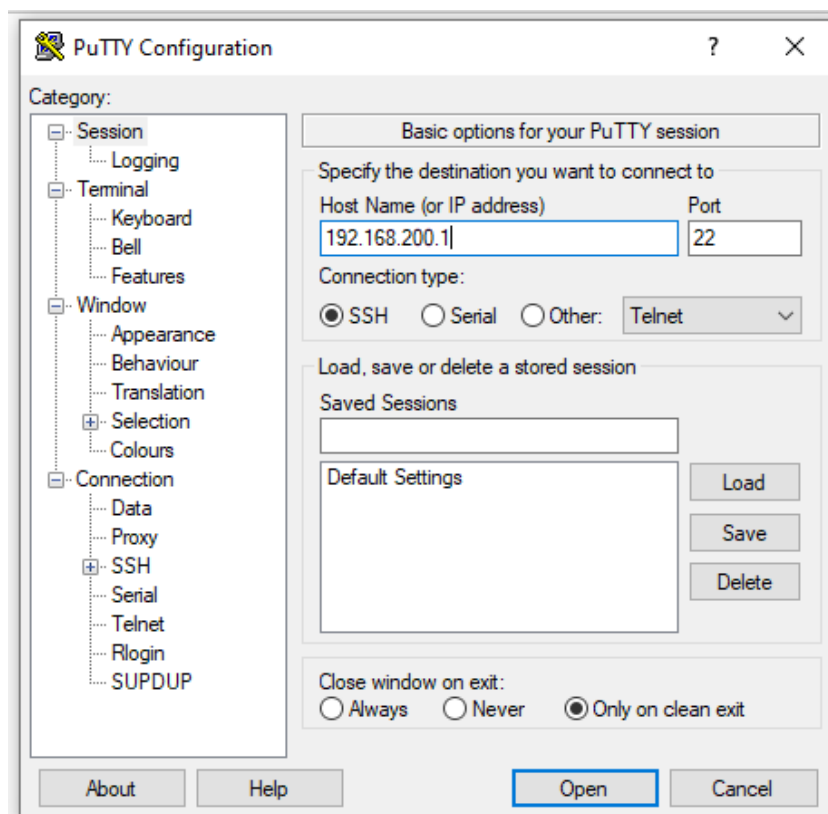


Figura 50. PuTTY para acceder a la consola de la pasarela

Una vez eso, se abrirá la consola de la pasarela IOT 2050, donde se introduce “root” en el apartado de “login as:”, luego, en la contraseña se escribe “root” también. Una vez hecho eso se pide que se cambie la contraseña, y se puede poner cualquiera.

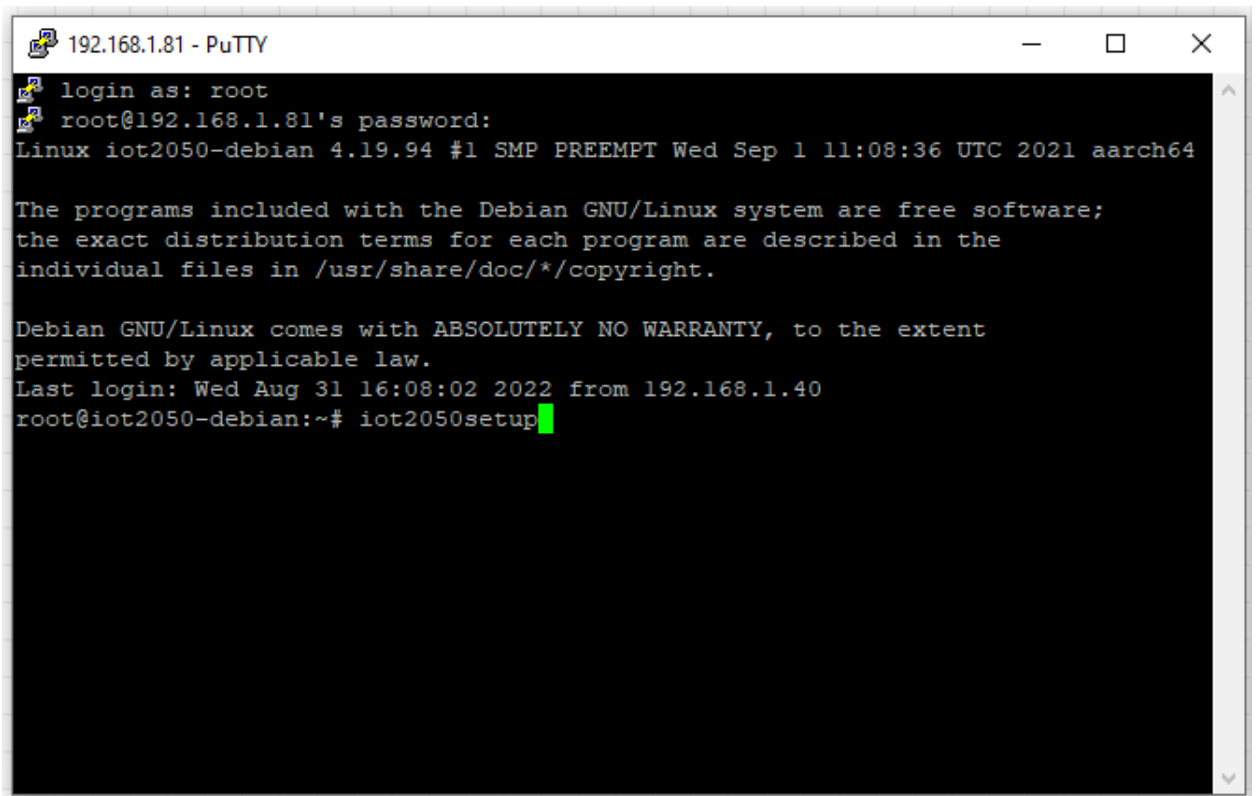


Figura 51. Consola de la pasarela

Para entrar a la configuración de la pasarela, se hace desde el comando "iot2050setup". Dentro de dicha configuración se pueden cambiar distintos aspectos del IOT 2050, como cambiar el Hostname, la contraseña o la zona horaria; entre varias cosas.

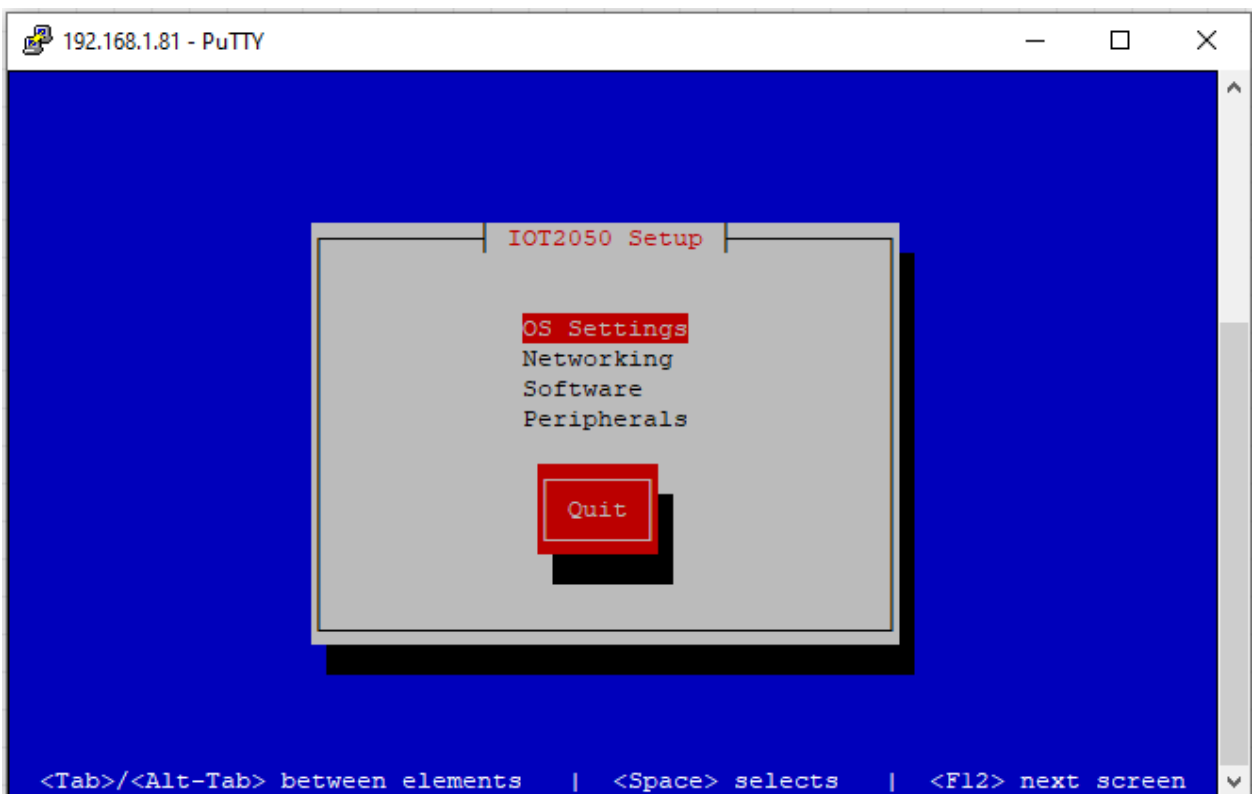


Figura 52. Setup de la pasarela

Desde la pestaña de “Networking” se activan y configuran al gusto las conexiones deseadas para acceder al IOT. En el caso de este proyecto hay dos: la local, con la IP 192.168.200.1; y la de la empresa, la cual es 192.168.1.81. Se puede acceder desde ambas tanto a Node-RED como a la consola de la pasarela, además de proporcionarle internet, ya que es necesario hacerlo.

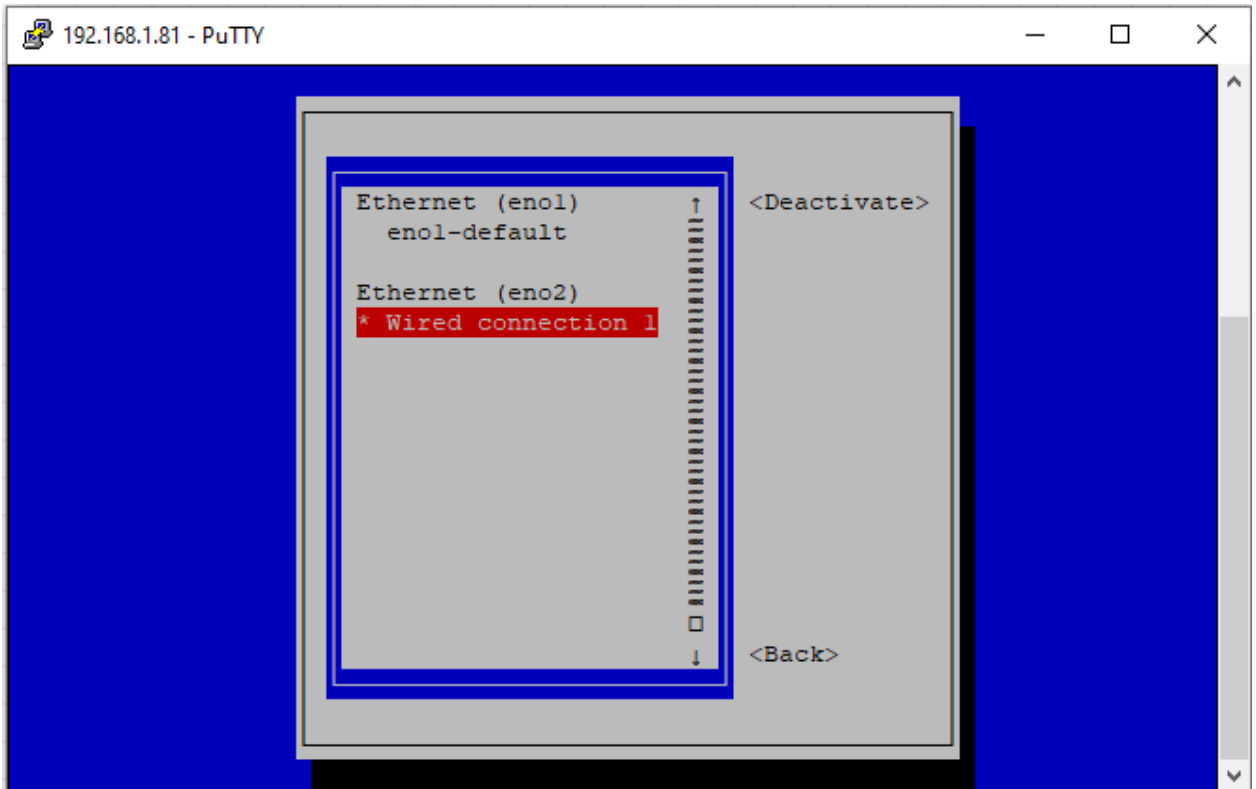


Figura 53. Conexiones a la pasarela

Como se puede ver en las siguientes dos imágenes, tal y como se ha comentado, se puede acceder a la pasarela IOT 2050 desde dos direcciones IP: 192.168.200.1 y 192.168.1.81. A la segunda dirección, además hay que añadirle la de los servidores DNS y la de Gateway, si no la conexión jamás se podrá realizar.

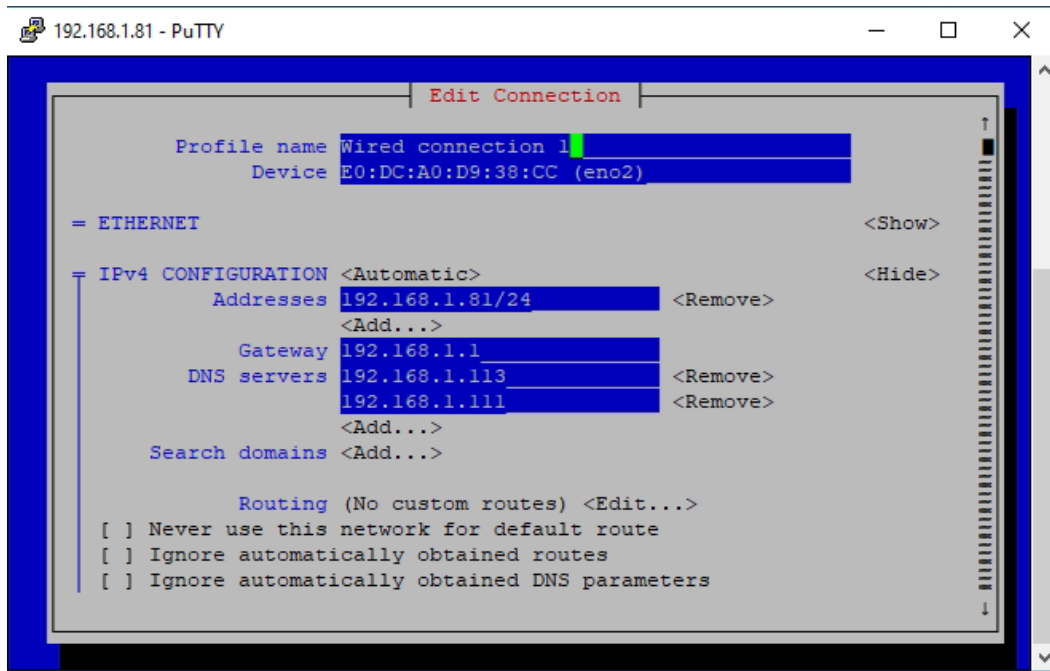


Figura 54. Dirección IP de acceso a la pasarela por una vía

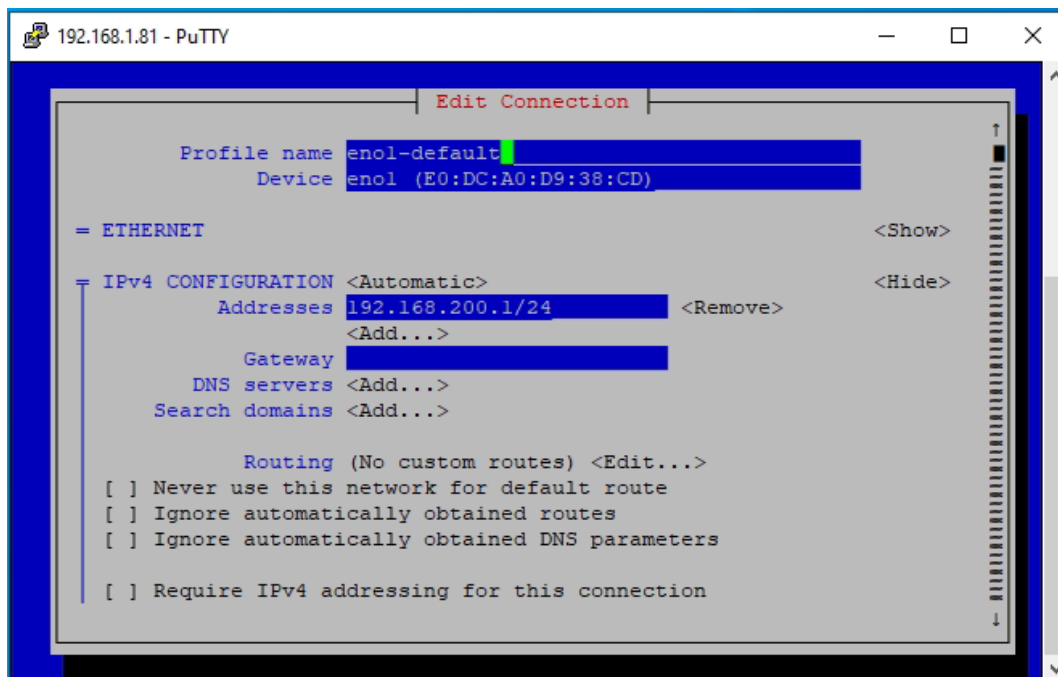


Figura 55. Dirección IP de acceso a la pasarela por otra vía

Se puede acceder tanto a Node-RED como a la configuración de la pasarela por una u por otra, no habría ningún tipo de problema. Además, si el usuario lo prefiere, podría incluso desactivar una de las dos entradas para entrar únicamente mediante la misma dirección IP.

4.1.2.2. Conexión a Node-RED desde la pasarela IOT 2050

Para instalar Node-RED en la pasarela IOT 2050, lo primero que hay que hacer es modificar un archivo “arch.conf”, que se encuentra en la carpeta “opkg”, que a su vez está en la carpeta “etc”. Eso se realiza con el siguiente comando:

```
nano /etc/opkg/arch.conf
```

Una vez dentro de ese archivo hay que añadir lo siguiente:

```
arch all 1  
arch any 6  
arch noarch 11  
arch i586 12  
arch quark 13  
arch x86 14  
arch i586-nlp-32 16  
arch i586-nlp-32-intel-common 21  
arch iot2050 26
```

Para guardar la modificación realizada, hay que pulsar “CTRL+X”, “yes” y “Enter”. A continuación, una vez realizado eso, hay que añadir los repositorios con un nuevo archivo que se llama “iotdk.conf” con el siguiente comando:

```
nano /etc/opkg/iotdk.conf
```

Dentro de él se escribe lo siguiente:

```
src iotdk-all http://iotdk.intel.com/repos/2.0/iotdk/all  
src iotdk-i586 http://iotdk.intel.com/repos/2.0/iotdk/i586  
src iotdk-quark http://iotdk.intel.com/repos/2.0/iotdk/quark  
src iotdk-x86 http://iotdk.intel.com/repos/2.0/iotdk/x86
```

A continuación, se agrega el repositorio MRAA con la siguiente línea de código:

```
echo "src mraa-upm http://iotdk.intel.com/repos/3.5/intelgalactic/opkg/i586" >  
/etc/opkg/mraa-upm.conf
```

Una vez hecho eso, se actualiza mediante:

```
opkg update
```

Ya estaría todo listo, simplemente quedaría por instalar node.js para poder usar Node-RED- Eso se realiza con el comando de a continuación:

```
opkg install nodejs
```

Una vez eso, se instala Node-RED con el mismo comando que se utiliza con un PC cualquiera:

npm install -g --unsafe-perm node-red

Ya estaría todo listo, por lo que siempre que la pasarela IOT 2050 esté en funcionamiento, Node-RED estará en ejecución. Para acceder a este hay que utilizar el enlace <http://192.168.200.1:1880> o por <http://192.168.1.81:1880> La dirección IP es la que se le ha asignado a la pasarela desde un primer momento, y el “:1880” es el número del puerto utilizado.

Finalmente, como se puede apreciar en la siguiente imagen, Node-RED está ejecutándose correctamente desde la pasarela IOT 2050.

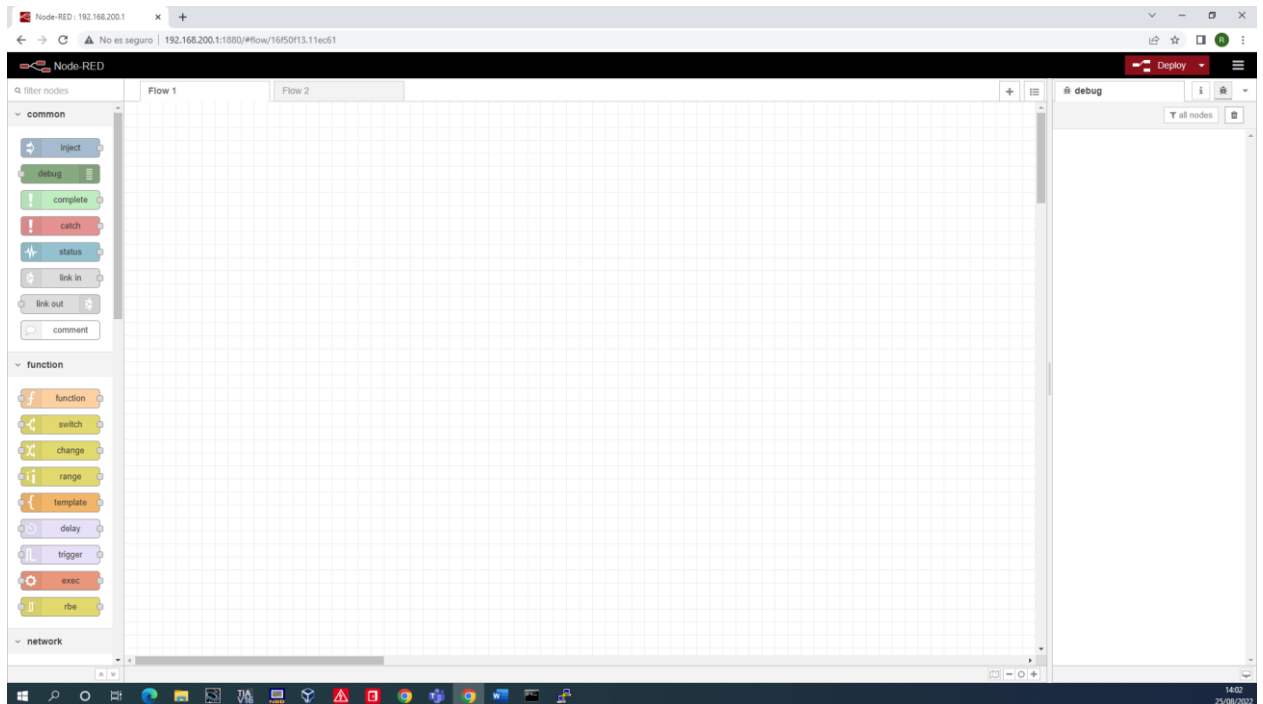
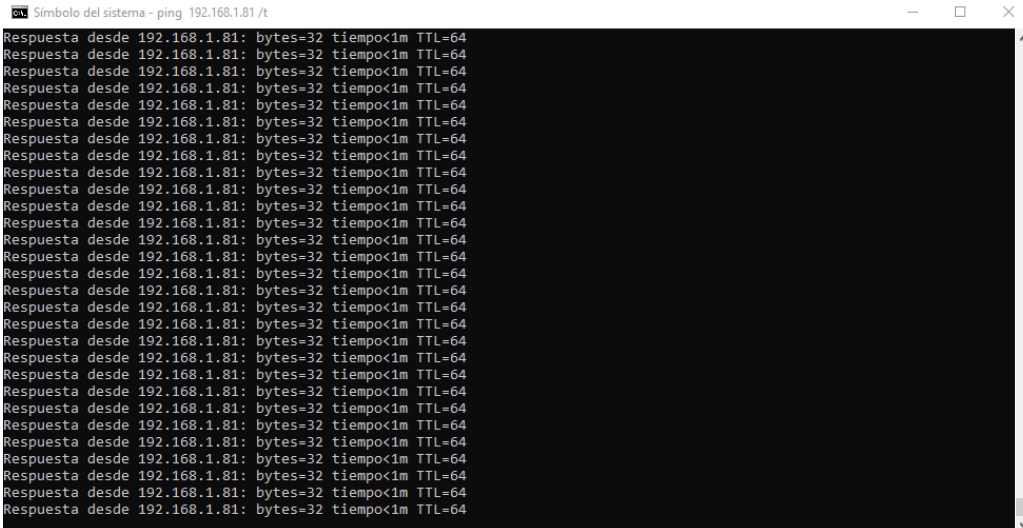


Figura 56. Entorno Node-RED desde IOT 2050

Como se puede ver también, si desde el cmd se intenta hacer ping con la pasarela IOT 2050, se puede ver que hay comunicación entre los dispositivos, por lo que se debería poder acceder tanto a Node-RED como a la configuración de la pasarela sin ningún tipo de problema desde la dirección IP 192.168.1.81, en este caso.



```
Símbolo del sistema - ping 192.168.1.81/t
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.1.81: bytes=32 tiempo<1m TTL=64
```

Figura 57. Ping a la pasera IOT 2050

Para instalar los nodos deseados para realizar este proyecto, se puede hacer de dos formas, o desde dentro de la consola o desde el mismo Node-RED.

En el caso de hacerlo desde la consola hay que introducir los siguientes comandos:

npm install node-red-contrib-s7

npm install node-red-node-watson

En el caso de querer hacerlo desde “Mange Palette”, en Node-RED, hay que introducir lo siguiente en el buscador:

node-red-contrib-s7

node-red-node-watson

4.2. SCADA y variables

4.2.1. Segmentos del programa

Se ha creado un pequeño programa en TIA Portal sin la finalidad de ser utilizado en ningún dispositivo. Este programa tiene como finalidad únicamente ser utilizado con carácter educativo para poder guiar y servir de ejemplo para cualquiera que quiera o necesite subir datos a la nube. Sin embargo, se va a realizar brevemente una descripción genérica del programa que se ha subido al autóata, para así entender mejor lo obtenido en IBM Cloud.

En primer lugar, el segmento 3 consiste en un contador (IEC_Counter_0_DB, %DB3) que va incrementando el valor en el bloque de datos Real 1. Ese valor va incrementando según los pulsos que van llegando según la variable %M3.5, que manda señales a pulsos cada segundo (Frecuencia 1 Hz). Se ha puesto un valor de '10' a PV, lo que quiere decir que cuando el contador llegue a 10, se encenderá la salida Q y, por tanto, funcionará Tag_7, aunque al final esta variable no se usa para nada. Por otro lado, se dispone una variable que resetea el contador cuando el bloque de datos llegue a 50, y se empezará el ciclo de nuevo.

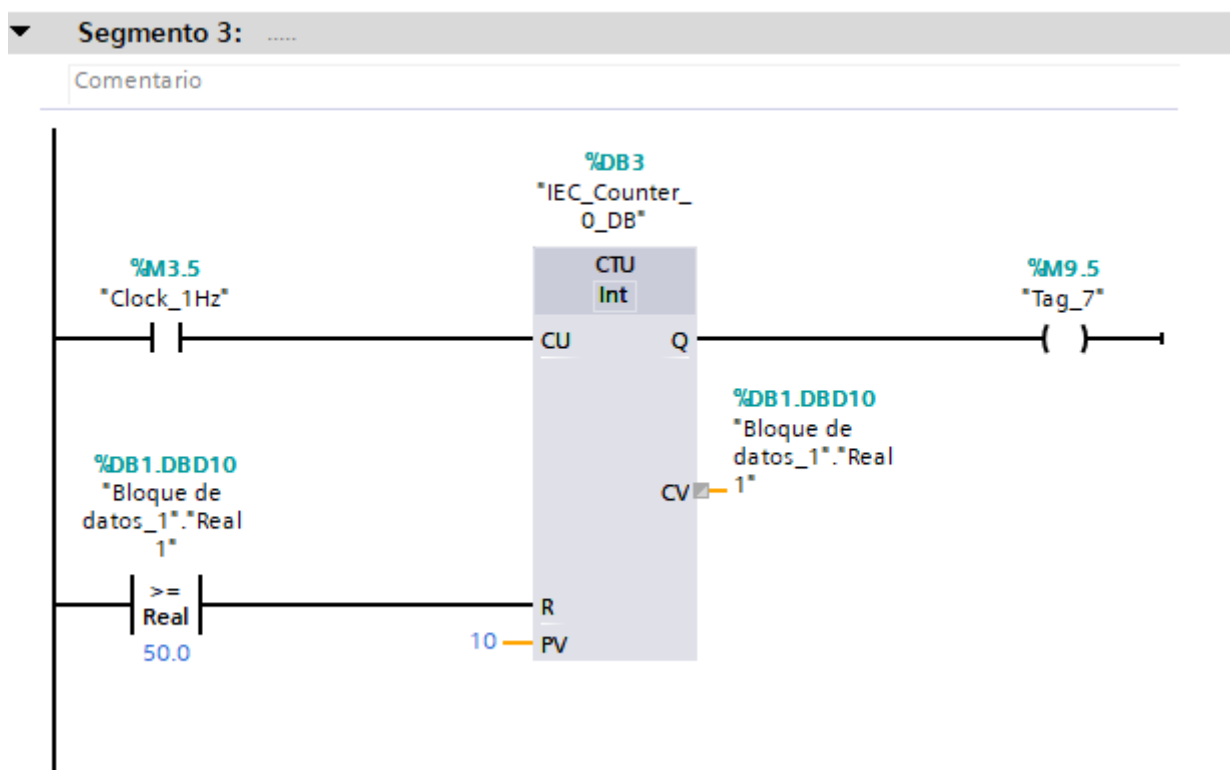


Figura 58. Contador 1 TIA Portal

En el segmento 4 se hace exactamente lo mismo, con un contador nuevo, poniendo el mismo valor en PV que en el segmento anterior, de '10'. Cogemos un nuevo bloque de datos "Real 2" y se pone como condición que el contador se resetee cuando este bloque de datos llegue a 50. Además, el contador recibe pulsos cada 0,1 segundos, a una frecuencia de 10 Hz, a diferencia del otro que era cada segundo. Aquí no se introduce ninguna variable como Tag_7, ya que no va a tener ninguna utilidad. Se usará para demostrar que se puede ver desde IBM si el programa y la máquina está funcionando correctamente mediante 'true' y 'false'.

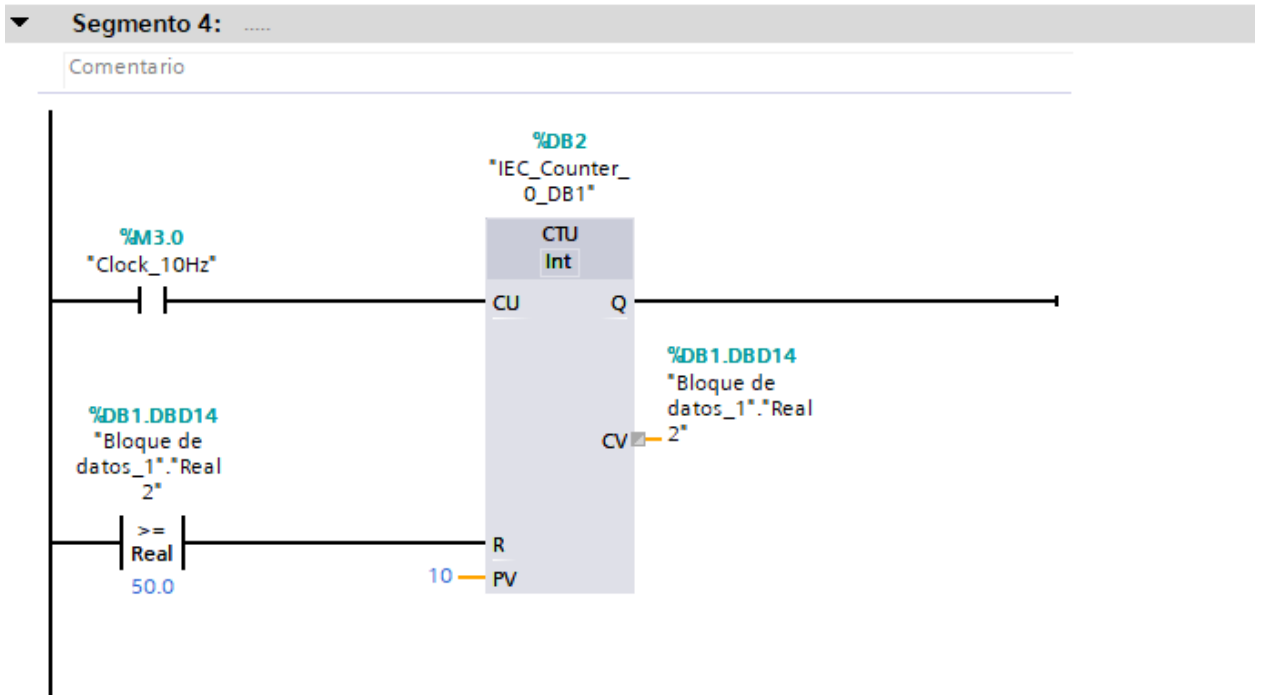


Figura 59. Contador 2 TIA Portal

Estos dos segmentos se van a utilizar para crear una función mediante el bloque “Calculate”, del segmento 5. La finalidad de este nuevo segmento es poder mostrar mediante los paneles de visualización de IBM Cloud un gráfico donde el valor de salida vaya variando constantemente con subidas y bajadas, en función a las variables de entrada definidas en los dos segmentos anteriores. Tanto las variables de entrada como las de salida son enteros que pueden ser tanto valores fijos (como IN 2 e IN 3, en la imagen de abajo) como variables (IN 1, IN 4 y OUT). Los valores de entrada IN 1 e IN 4 van variando en función a los segmentos 3 y 4, con el valor que van recibiendo constantemente las variables Real 1 y Real 2 del bloque de datos. Por otro lado, las variables IN 2 e IN 3 son fijas y tienen un valor entero cada una de 100 y 5, respectivamente.

Respecto a la salida OUT, esta es una función variable y dicha función es la siguiente:

$$OUT = (IN 1 \times IN 2) + (IN 3 \times IN 4)$$

Aparentemente la función simplemente tendría que crecer y crecer, pero no es así. Si bien es cierto que la función con esas variables debería tener una pendiente ascendente todo el rato y los contadores de los segmentos 3 y 4 son crecientes, no siempre va a crecer la variable de salida ya que los contadores reciben pulsos a ritmos diferentes (10 veces más rápido el segmento 4) y, por tanto, siempre habrá uno reseteándose mucho más rápido y creando ciertas perturbaciones en el crecimiento de la gráfica. Por otro lado, llegará un punto en el que ambos se resetearán a la vez, por lo que OUT será 0, y volverá a empezar el ciclo de subida.

Como he dicho anteriormente, estos tres segmentos explicados hasta ahora servirán como ejemplo para ver cómo visualizar datos desde el servidor de IBM Cloud y no tienen mucha dificultad en cuanto programación en TIA Portal, ya que no es ese el objetivo principal del proyecto.

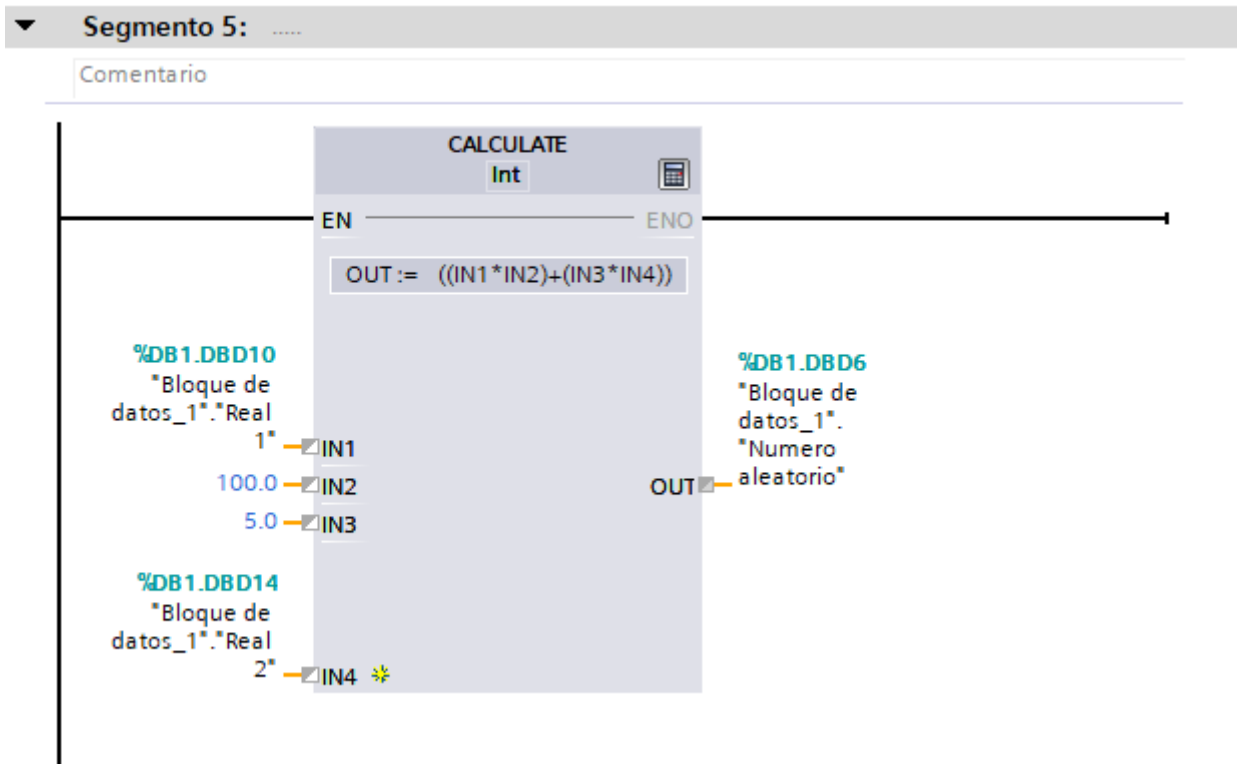


Figura 60. Función Calculate TIA Portal

Después de estos primeros tres segmentos, se han realizado dos más haciendo un pequeño ejercicio práctico. Con los segmentos 6 y 7 se quiere simular una paletizadora. En el primero de los dos, se ha puesto un contador ascendente donde se simula que van llegando cajas a la paletizadora y esta llena los palés. Las cajas se simulan con pulsos de reloj cada segundo (1 Hz), pero se podría poner cualquier otra frecuencia. Se ha considerado que el palé tenga una capacidad de hasta 50 cajas. Por tanto, se pone el valor '50' como parámetro de referencia, una vez se llegue a esa cantidad se resetea el contador. El número de cajas que van saliendo se contabilizan con un bloque de datos llamado Real 3.

Aparte de resetear el contador del segmento 6, se enviará un pulso al contador de palés llenos en el segmento 7 con la función QU a la salida del contador, que es una de las distintas funcionalidades proporcionadas.

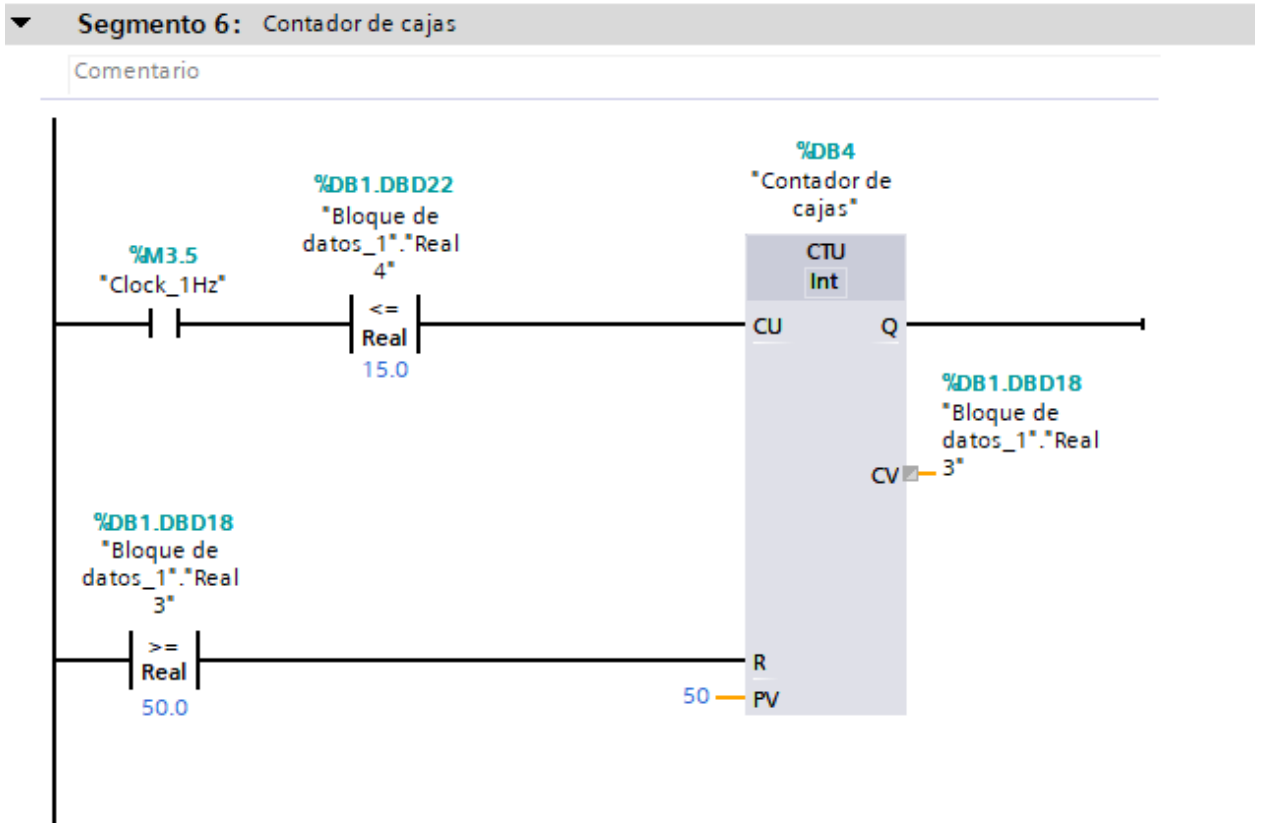


Figura 61. Contador de cajas TIA Portal

Como se acaba de comentar, para que el contador de palés del segmento 7 crezca y considere que un palé está lleno, se usa la variable “Contador de cajas”.QU, que solamente dejará enviar señal al contador cuando la salida del contador 6 esté operativa. Esta solo enviará señal durante 1 segundo, ya que en cuanto el contador de cajas llega a 50 se resetea.

El número de palés llenos se va acumulando en el bloque de datos Real 4 y, cuando este llegue a 15, se parará el proceso para que no sigan pasando más cajas y no se llenen más palés. Eso se ha hecho con el comparador introducido al lado de la señal de reloj.

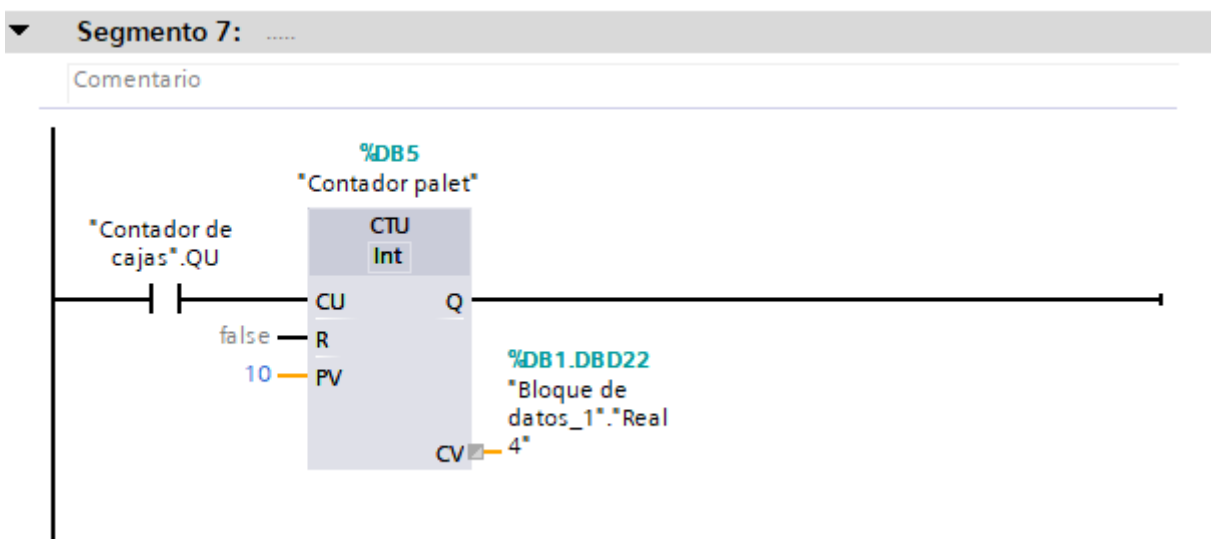


Figura 62. Contador de palés TIA Portal

4.2.2. Variables SCADA

En este apartado se muestran las variables que se han utilizado para la programación en TIA Portal. Muchas no se han llegado a mostrar, dado a que solo ha sido a modo de prueba previamente. Hay distintos tipos de frecuencia para los relojes, pero al final solo se ha utilizado el de 1 Hz y 10 Hz, además de distintas variables booleanas que han quedado sin utilizar como AlwaysTRUE y AlwaysFALSE, entre otras.

Tabla de variables estándar									
	Nombre	Tipo de datos	Dirección	Rema...	Acces...	Escrib...	Visibl...		
1	Tag_2	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
2	Tag_1	Bool	%M10.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	System_Byte	Byte	%MB4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
4	FirstScan	Bool	%M4.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
5	DiagStatusUpdate	Bool	%M4.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
6	AlwaysTRUE	Bool	%M4.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
7	AlwaysFALSE	Bool	%M4.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
8	Clock_Byte	Byte	%MB3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
9	Clock_10Hz	Bool	%M3.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
10	Clock_5Hz	Bool	%M3.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
11	Clock_2.5Hz	Bool	%M3.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
12	Clock_2Hz	Bool	%M3.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
13	Clock_1.25Hz	Bool	%M3.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
14	Clock_1Hz	Bool	%M3.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
15	Clock_0.625Hz	Bool	%M3.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
16	Clock_0.5Hz	Bool	%M3.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
17	Tag_4	Bool	%M10.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
18	Tag_5	Bool	%M10.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
19	Tag_3	Int	%MW100	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
20	Tag_6	Bool	%M10.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
21	Tag_7	Bool	%M9.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
22	Tag_8	Int	%MW101	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
23	Tag_9	Bool	%M10.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
24	<Agregar>			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Figura 63. Tabla de variables TIA Portal

Sobre los bloques de datos, los de incremento palet, segundos y minutos no se han llegado a utilizar para este proyecto, pero sí se probaron previamente. Los bloques de datos de nombre y con variable Real y la del número aleatorio sí se han llegado a utilizar tanto para almacenar valores de los contadores como para la calculadora con la que se ha creado una función a mostrar en los paneles de visualización de IBM Cloud.

Bloque de datos_1										
	Nombre	Tipo de datos	Offset	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..	Comen...
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Incremento palet	Int	0.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Incremento segundos	Int	2.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Incremento minutos	Int	4.0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	Numero aleatorio	Real	6.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	Real 1	Real	10.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	Real 2	Real	14.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	Real 3	Real	18.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	Real 4	Real	22.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	Real 5	Real	26.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	Real 6	Real	30.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	Real 7	Real	34.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	Real 8	Real	38.0	0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
14	Flancos	Struct	42.0		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Figura 64. Bloques de datos TIA Portal

4.3. Node-RED y conexiones. Subida de datos a paneles de visualización de IBM Cloud

En esta parte del proyecto, se va a mostrar la programación de los nodos con Node-RED para poder comunicarse con el servicio de IBM Cloud. Para ello, tenemos que hacer uso de los nodos mencionados en apartados anteriores: “s7 in”, para enviar los valores a Node-RED y “Watson IOT”, el de output, para enviar esos mismos datos de IBM Cloud. Por otro lado, se ha introducido el nodo de “debug”, con el que se puede comunicar y mostrar datos por pantalla. Con este último nodo se puede saber si se están recibiendo datos y si estos son correctos.

El esquema utilizado para este proyecto es el siguiente:

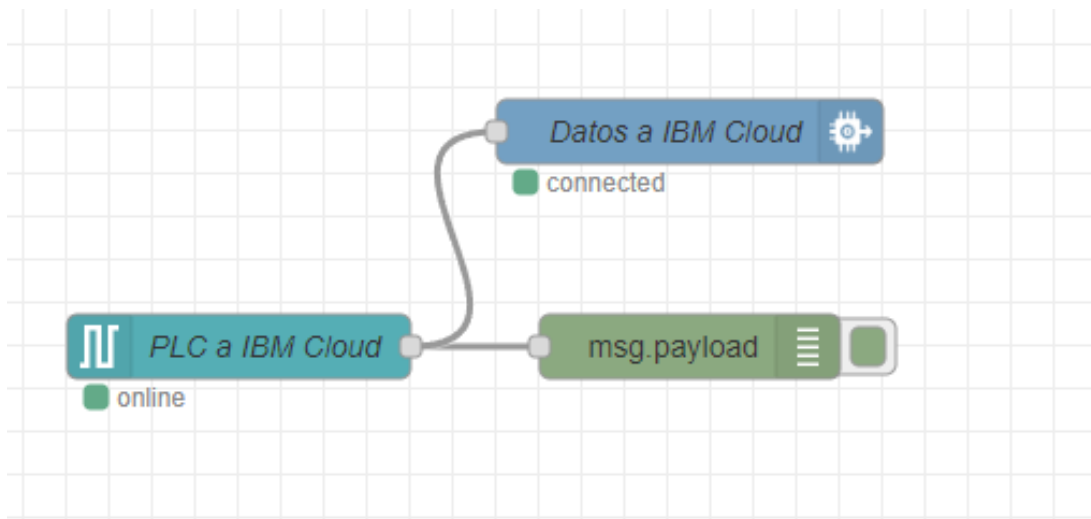


Figura 65. Esquema conexión de autómata a IBM Cloud

Lo primero que se realiza es la configuración del nodo que comunica el autómata (y la pasarela IOT 2050) con Node-RED. Lo primero, aunque no es necesario, es ponerle un nombre al nodo, en este caso se le ha puesto “PLC a IBM Cloud”, por si se hicieran más apartados y no confundir términos. Luego, hay que editar el endpoint y poner las características del PLC en uso, además de las variables que se quieren poner. Para ello hay que darle al lápiz que está al lado de la pestaña de PLC.

Por otro lado, para poder visualizar todas las variables seleccionadas desde el mismo nodo, en “mode” hay que seleccionar “All variables”, aunque se podría elegir cualquiera que se desee. Sin embargo, en este caso interesa más el poder mostrarlas todas desde el mismo nodo, ya que hace el programa y la subida de datos mucho más simple y cómoda, pero ambas opciones son válidas.

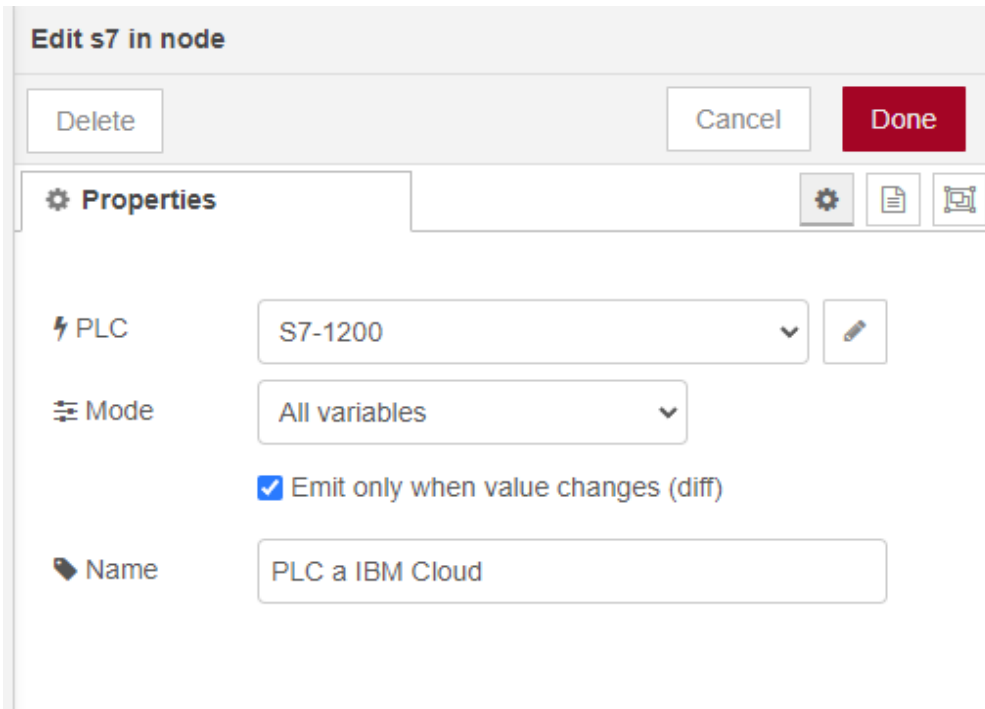


Figura 66. Configuración nodo s7 in

Como se ha dicho, hay que escribir las características del PLC y determinar las variables a mostrar. Para ello, primero se va a la opción de “Variables” y desde ahí se escribe la lista de las variables que se van a utilizar en el proceso.

En la columna de la derecha se escribe el nombre que se quiera mostrar para clasificar los datos de esa variable. En ese caso se han puesto “Calculate OUT”, “Número de cajas” y “Número de palés”.

En cambio, la columna de la izquierda hace referencia de la dirección de las variables en TIA Portal. Sin embargo, no se escriben exactamente igual, hay que hacer una pequeña corrección a la hora de su escritura. En la tabla siguiente se va a mostrar las principales direcciones tanto en TIA Portal como en Node-RED:

Address	Equivalente en Step7	Tipo de Dato JS
DB5,X0.1	DB5.DBX0.1	Booleano
DB23,B1 o DB23,BYTE1	DB23.DBB1	Número
DB100,C2 o DB100,CHAR2	DB100.DBB2	String
DB42,I3 o DB42,INT3	DB42.DBW3	Número
DB21,R7 o DB21,REAL7	DB21.DBD7	Número
QI12 o AI12	QW12 o AW12	Número (output)
MRW20	MW20	Número (memoria)

Tabla 10. Conversión direcciones de variables de Step7 a Node-RED

Por lo tanto, las variables utilizadas para este trabajo son las siguientes que se muestran en la imagen:

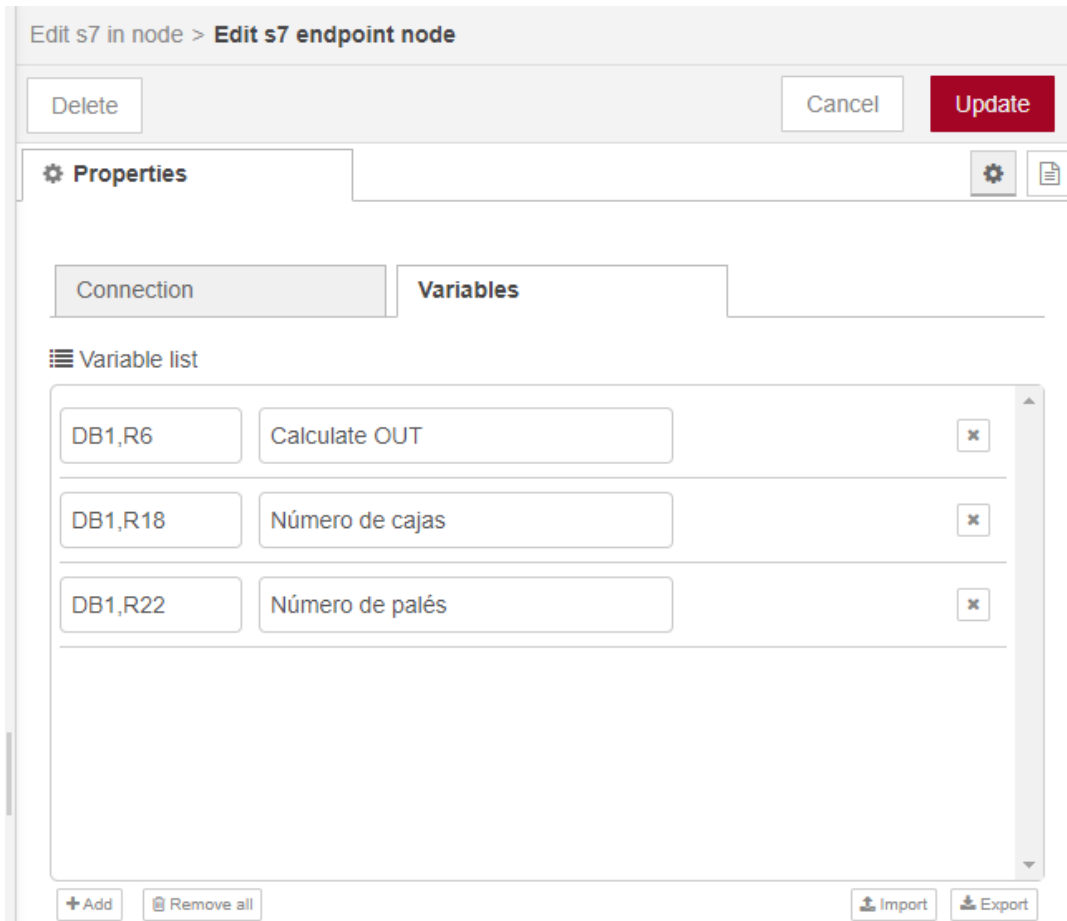


Figura 67. Variables nodo s7 in

Posteriormente, hay que configurar la conexión del nodo con el S7-1200 al que se le ha cargado el programa. Para ello, hay que ir al apartado de conexiones. Lo primero es seleccionar que la comunicación se hace mediante Ethernet y la dirección IP del autómatas, que es 192.168.200.5, tal y como se puede apreciar en el apartado 3.2 de este documento. El puerto 102 o pone Node-RED por defecto, no hace falta tocarlo.

A Continuación, se selecciona el modo Rack/Slot y se tiene que seleccionar en cuáles está colocado el autómatas desde el programa. En este caso se encuentra en el Rack 0 y Slot 1. Esto se puede ver desde aquí:



Figura 68. Rack y Slot del autómeta

Por último, se pone el tiempo de ciclo y timeout deseado, en este caso de 500 ms y 1500 ms, respectivamente. Además, hay que ponerle un nombre para guardar esta configuración, que es “S7-1200” por ejemplo.

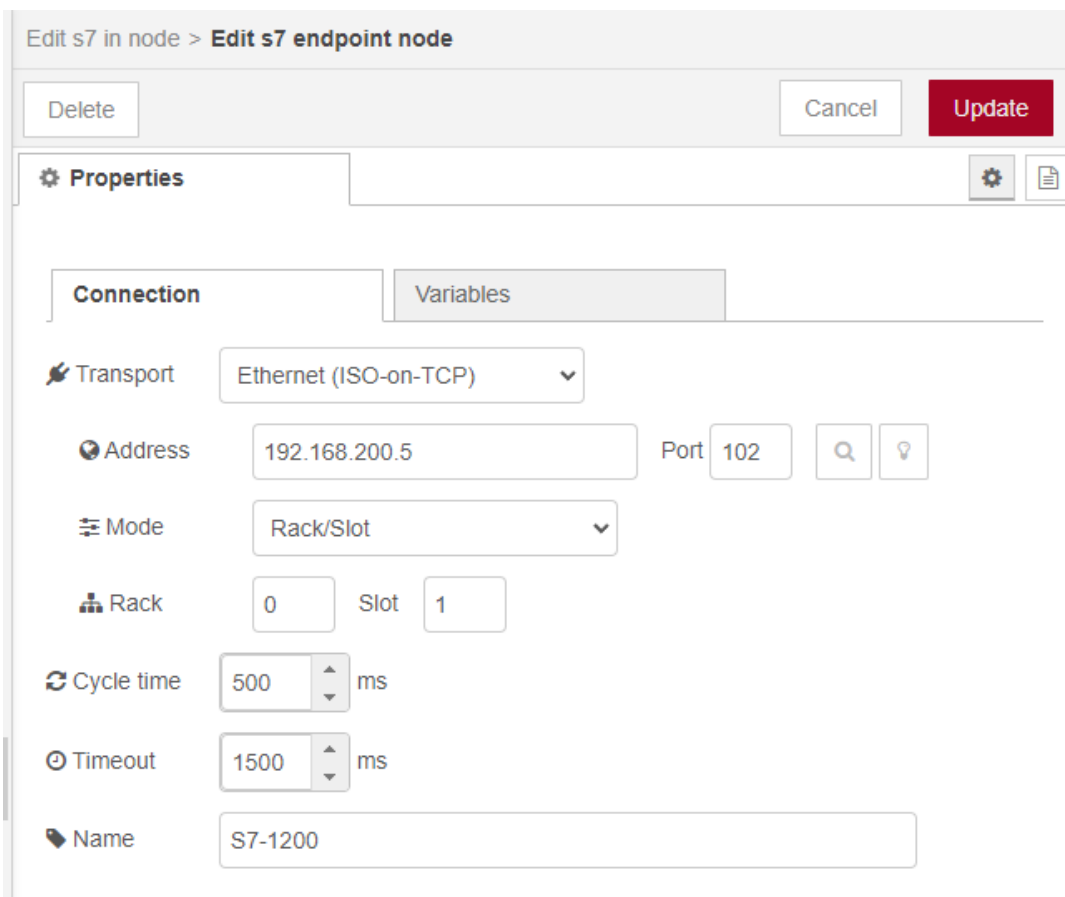


Figura 69. Configuración conexión s7 in

Con esto, ya estaría el nodo que comunica el autómeta S7-1200 con Node-RED correctamente configurado y se deberían de subir los datos en tiempo real.

En esta imagen se puede ver que todo funciona correctamente, y gracias al debug se puede apreciar los valores que se van recibiendo desde la ventana de Node-RED. Los datos aparecen como en la parte derecha de la foto, pero se puede desplegar y verse de una manera mucho más ordenada, tal y como se ve a la izquierda.

Aparte de proporcionar la información recibida, también te dice de qué tipo es el mensaje, que en este caso es un objeto, ya que las variables creadas en TIA Portal (y definidas en Node-RED) son de tipo Real, no enteros. El objetivo siguiente es subir estos valores a una base de datos y al servicio de IBM Cloud.



Figura 70. Datos del autómata mostrados en Node-RED

Como se ha dicho, el objetivo a continuación consiste en ser capaces de subir los datos desde Node-RED al servicio de IBM Cloud. Para ello, hay que hacer uso del nodo “Watson IoT”. En la configuración de este nodo hay dos opciones para conectarlo, como “Device” o “Gateway”. En este caso, ya que se está haciendo uso de una pasarela IOT 2050, hay que conectarlo como Gateway. Luego, hay que seleccionar la opción “Registered”, escribir un nombre para el nodo (“Datos a IBM Cloud”, en este caso), escribir el tipo y la ID del servicio, que son “IOT2050” y “Datos”, para ese caso en concreto. Una vez realizado eso, hay que configurar las credenciales para poder subir la información al servicio en la nube. Para ello hay que clicar en el lápiz al lado de “Credentials”.

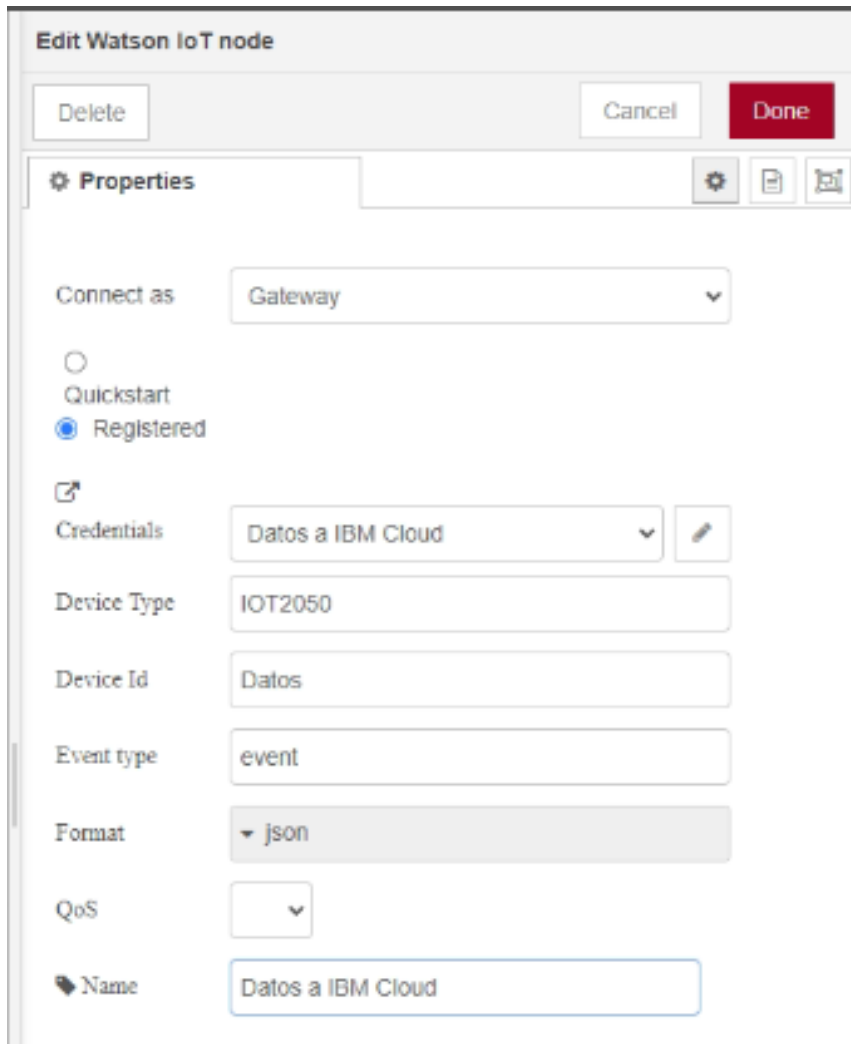


Figura 71. Configuración nodo IoT Watson

Una vez dentro de las credenciales. Lo primero que hay que introducir es el código de la organización, el cual se otorga a la cuenta del usuario una vez se lanza un recurso. En este caso es “ypl07a” y eso se puede ver en la siguiente imagen:

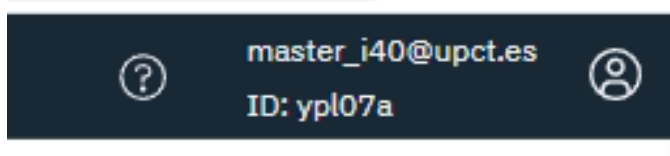
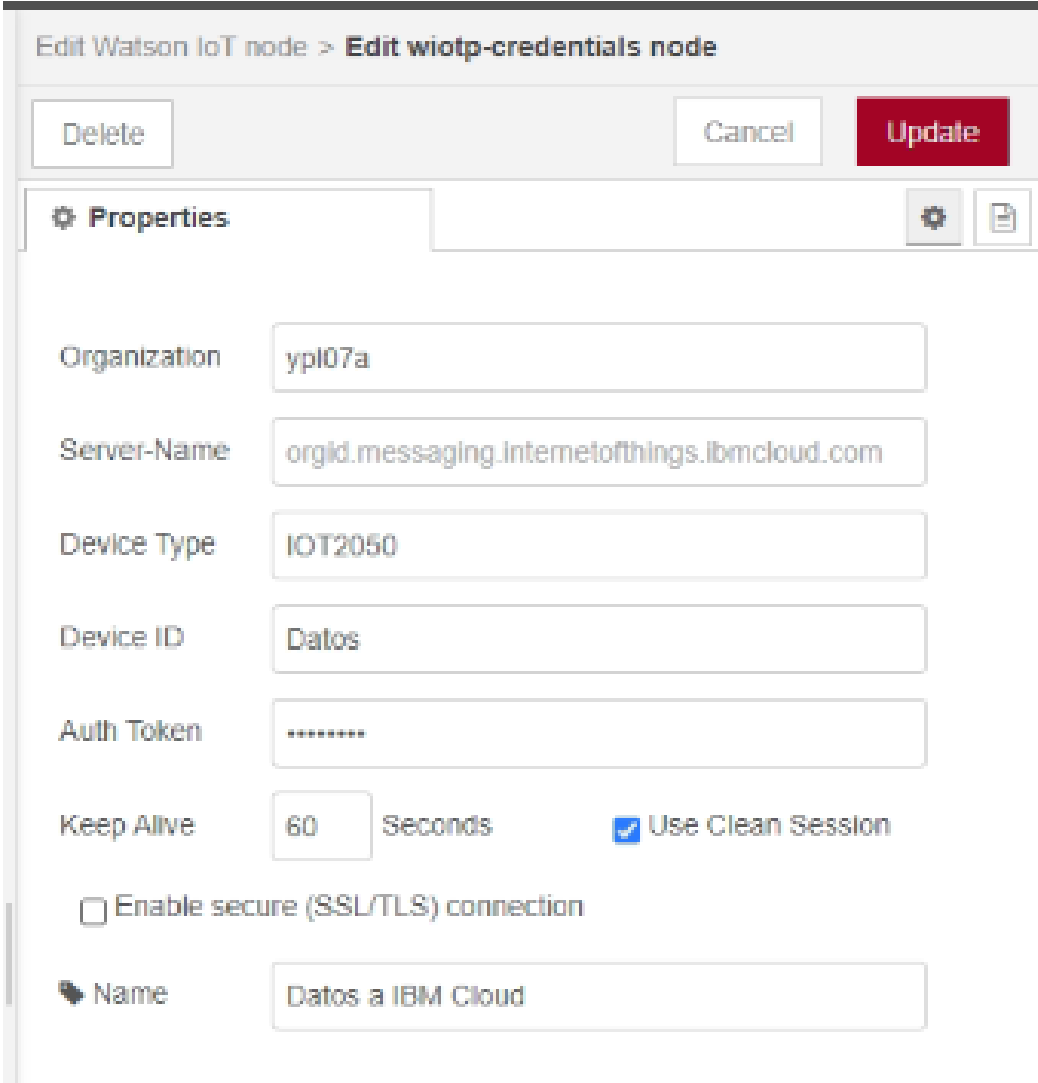


Figura 72. Organización IBM Cloud

Luego, se escribe el tipo de dispositivo, que es “IOT2050” y la ID del dispositivo, que es “Datos”. Posteriormente, se escribe la contraseña de autenticación seleccionada en “Auth Token” y se introduce un nombre a estas credenciales. En este caso “Datos a IBM Cloud”.



Edit Watson IoT node > **Edit wiotp-credentials node**

Delete Cancel Update

⚙ Properties

Organization ypl07a

Server-Name orgid.messaging.internetofthings.ibmcloud.com

Device Type IOT2050

Device ID Datos

Auth Token

Keep Alive 60 Seconds Use Clean Session

Enable secure (SSL/TLS) connection

👤 Name Datos a IBM Cloud

Figura 73. Configuración propiedades del nodo IoT Watson

Una vez hecho esto, se actualiza, se acepta todo y se despliega Node-RED con todos los nodos actualizados. Tras todo este procedimiento, los datos deberían estar enviándole desde el autómata hasta el servicio de IBM Cloud.

4.4. Configuración de IBM Cloud

En este apartado se va a explicar el procedimiento realizado en el servicio de IBM Cloud para poder visualizar los datos en los paneles de visualización que proporciona la misma empresa. Lo primero que hay que hacer es dirigirse a la lista de recursos. En la imagen se ve creado el que se ha utilizado, pero si se empieza desde cero no habrá ningún recurso en la lista. Hay que clicar en “Crear recurso” y, en el catálogo con infinidad de servicios que aparecen, buscar “Internet of Things Platform”, que es el que se necesita para subir y monitorizar los datos desde la nube de IBM.



Figura 74. Lista de recursos IBM Cloud

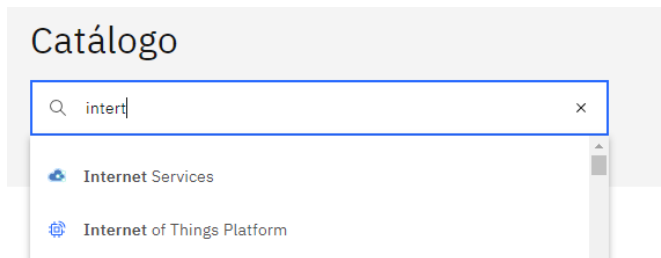


Figura 75. Internet of Things IBM Cloud

Una vez seleccionada dicha herramienta, hay que seleccionar una ubicación. En este caso se ha elegido la más cercana a la ubicación de la empresa para tener muchas menores latencias. A continuación, se selecciona uno entre los varios planes que ofrece este servicio de IBM. Se ofrecen tres tarifas, de más barata a más cara:

1. Gratuito: Ofrece un máximo de 500 dispositivos conectados, 500 enlaces de aplicaciones y un máximo de 200 MB de datos intercambiados. Al ser una versión de prueba, los servicios se suprimen tras 30 días de actividad.
2. 2500 \$ USD/Plan: Es el nivel de capacidad 1, que incluye hasta 5000 dispositivos de sensor y 1 millón de sucesos de análisis.
3. 8000 \$ USD/Plan: Es el nivel de capacidad 2, incluyendo un total de 25000 dispositivos de sensor y 5 millones de sucesos de análisis.

En este caso, al estar con la versión gratuita de estudiante, sólo es ofertada la versión gratuita, por lo que se selecciona esa. Además, tampoco se le va a dar un uso a día de hoy más allá del nivel educativo.

A continuación, se selecciona un grupo de recursos, que al estar con una versión de prueba se tiene solo uno por defecto, llamado “Default”. Se aceptan los términos de condiciones y se crea el servicio.

Seleccione una ubicación

Frankfurt (eu-de)

Seleccione un plan de precios

Los precios mostrados no incluyen impuestos. Los precios mensuales que se muestran son para el país o ubicación: [Estados Unidos](#)

Plan	Características	Tarifas
Lite	<p>Incluye un máximo de 500 dispositivos registrados y un máximo de 200 MB de métricas de datos cada uno</p> <p>Un máximo de 500 dispositivos registrados</p> <p>Máximo de 500 enlaces de aplicación</p> <p>Un máximo de 200 MB de datos intercambiados, datos analizados y datos límite analizados cada uno</p> <p>El plan de servicio Lite para Internet of Things Platform incluye un máximo de 500 dispositivos registrados y un máximo de 200 MB de datos intercambiados, datos analizados y datos límite analizados al mes para cada uno.</p> <p>Los servicios del plan Lite se suprimen tras 30 días de inactividad.</p>	Gratis

Configurar su recurso

Nombre de servicio: Internet of Things Platform-6i

Selección de grupo de recursos: Default

Etiquetas: Ejemplos: env:dev, version-1

Etiquetas de gestión de acceso: Ejemplos: access:dev, proj:version-1

Resumen

Internet of Things Platform Gratuito

Ubicación: Frankfurt

Plan: Lite

Nombre de servicio: Internet of Things Platform-6i

Grupo de recursos: Default

Instancia existente del plan Lite

Solo puede tener 1 instancia de plan Lite de este servicio por grupo de recursos. [Suprima](#) la instancia de plan Lite actual de Default grupo de recursos para crear una nueva o [visualice la instancia existente](#).

He leído y acepto los siguientes acuerdos de licencia: [Condiciones](#)

Crear

Añadir a estimación

Figura 76. Configuración del recurso

Ahora que está creado el recurso, aparece la siguiente pantalla diciendo que el servicio está activo. Se daría a lanzar y estaría listo, ya se podría utilizar.

Lista de recursos /

Internet of Things Platform-4y Activo proyecto1 IAM | proyecto...

Gestionar

Plan

Conexiones

Empecemos con IBM Watson IoT Platform

Conecte, controle y gestione dispositivos de forma segura. Cree rápidamente aplicaciones IoT que analicen datos del mundo físico.

Lanzar Docs

Figura 77. Lanzamiento del recurso creado

El servicio ya está lanzado. Ahora, lo que hay que hacer es crear los dispositivos y el tipo de dispositivo. Lo primero que hay que crear es el tipo de dispositivo, por lo que se va a explicar cómo realizar ese proceso principalmente.

Primero, hay que clicar en “Tipos de dispositivo” y darle a la opción de “Añadir tipo de dispositivo” para incluir la pasarela, que es la que comunica el autómatas con IBM Cloud.

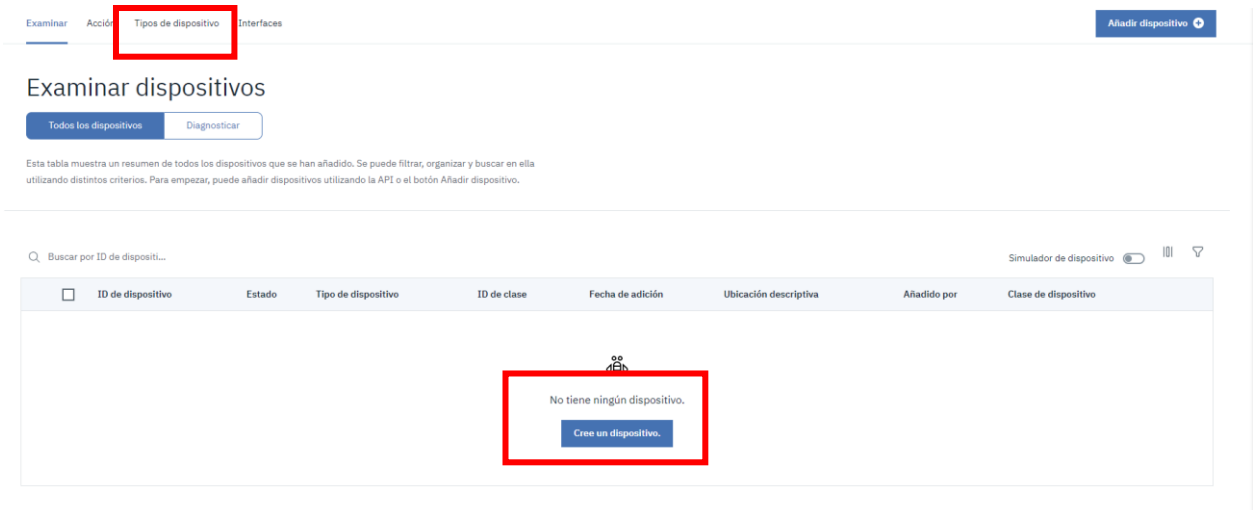


Figura 78. Entorno de creación de dispositivos y tipos de dispositivos

Después de seleccionar la opción de “Añadir Tipo de Dispositivo”, se elige la opción de Dispositivo, en el caso de subir los datos desde un ordenador, o la de Pasarela, en caso de hacerlo desde ese dispositivo. En este caso, al usar un IOT 2050, se selecciona la Pasarela como tipo.

Posteriormente, se pone el nombre del tipo, en este caso “IOT2050” y la descripción, que se ha puesto la misma. Una vez se acepta y se avanza, aparecerá un resumen con la información del dispositivo.



Figura 79. Creación de un tipo de dispositivo

Ahora, volviendo a la ventana donde aparecen los tipos de dispositivos, debería aparecer el creado con el nombre “IOT2050”. Se ve claramente que es así.

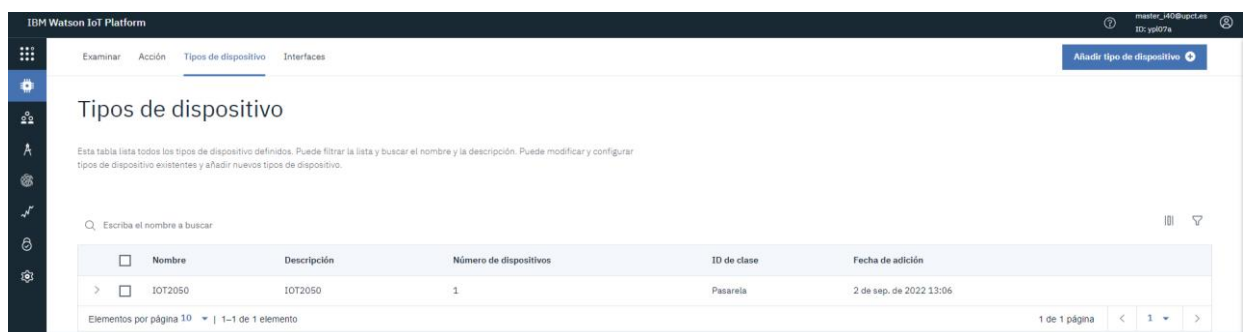


Figura 80. Tipo de dispositivo creado

Una vez creado el tipo, hay que añadir el dispositivo. Como se puede ver en la siguiente imagen aquí hay que realizar un par de cosas más, pero el proceso sigue siendo muy rápido. Lo primero que hay que hacer es otorgarle una identidad. Para ello, hay que definir el tipo de dispositivo, y ahí se selecciona el que se acaba de crear, para poder estar conectado con Node-RED. Luego, se crea una ID, con el nombre que se desee. En este caso será “Datos”.

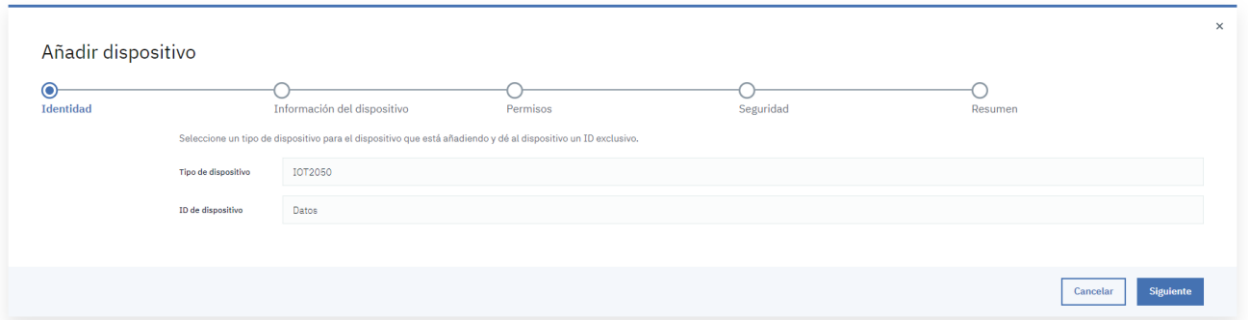


Figura 81. Creación de dispositivo

En el siguiente paso aparece la información del dispositivo y un resumen con todos los datos recogidos. Pero antes de eso, hay que crear una contraseña como señal de autenticación. Esta contraseña es la que se mete en el nodo de “Watson IoT”, como se ha visto anteriormente. Por lo tanto, es muy importante no olvidarla, ya que si no será imposible acceder al dispositivo.

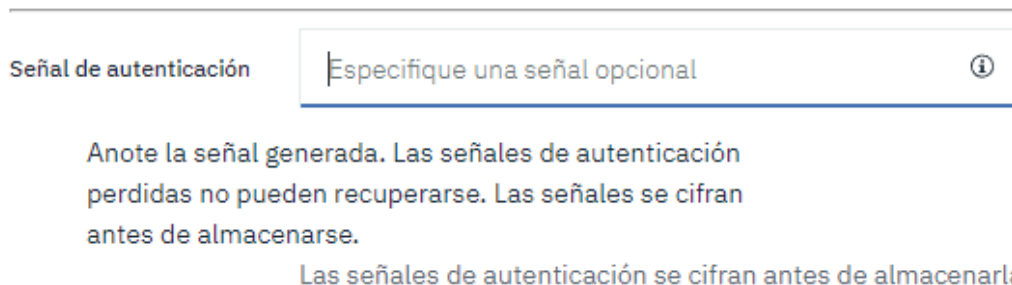


Figura 82. Autenticación del dispositivo

Este es el esquema utilizado en Node-RED para subir información a IBM Cloud. Está explicado todo su procedimiento en el apartado anterior.

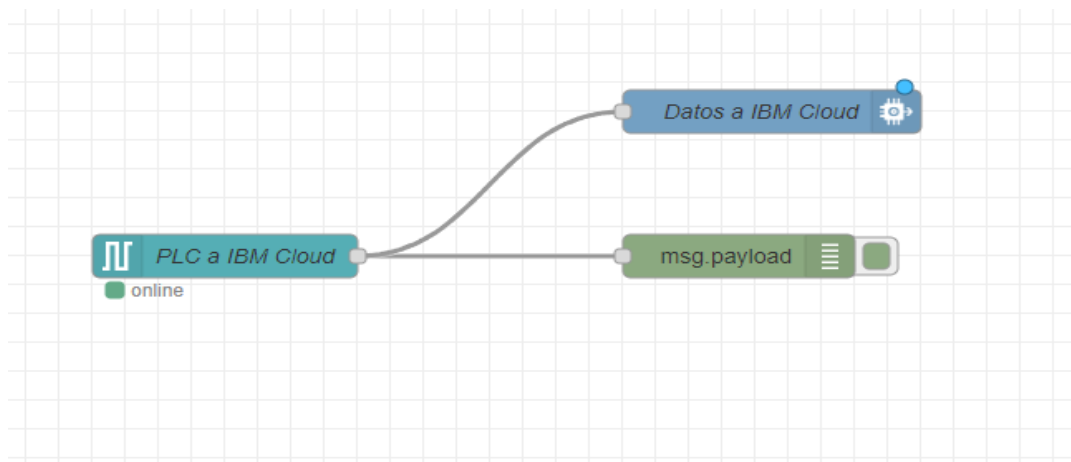


Figura 83. Conexión Node-RED a IBM Cloud

En la siguiente imagen se puede ver que se ha creado el dispositivo, por lo que ya se puede trabajar con él una vez se realice la conexión con Node-RED, que está explicado anteriormente.

ID de dispositivo	Estado	Tipo de dispositivo	ID de clase	Fecha de adición	Ubicación descriptiva	Añadido por	Clase de dispositivo
Datos	Desconectado	IOT2050	Pasarela	2 de sep. de 2022 13:08		master_140@upct.es	

Figura 84. Dispositivo creado

Como se puede apreciar, los datos están siendo subidos al servicio en formato json. Por lo que toda la configuración tanto de Node-RED como de los dispositivos de IBM Cloud ha funcionado correctamente.

Los sucesos recientes listados muestran la corriente activa de datos que entran y salen en este dispositivo.

Suceso	Valor	Formato	Último recibido
event	{"d":{"Calculate OUT":2405,"Número de cajas":4...	json	hace unos segundos
event	{"d":{"Calculate OUT":2380,"Número de cajas":4...	json	hace unos segundos
event	{"d":{"Calculate OUT":2255,"Número de cajas":4...	json	hace unos segundos
event	{"d":{"Calculate OUT":2230,"Número de cajas":4...	json	hace unos segundos
event	{"d":{"Calculate OUT":2105,"Número de cajas":4...	json	hace unos segundos

Figura 85. Datos subidos desde Node-RED en tiempo real

Si se selecciona uno de los datos subidos, se puede ver la carga útil de suceso, con la información detallada y bien desglosada de los datos que se han subido a la nube.

Carga útil de suceso

Nombre de suceso event
Hora de recepción 6 de sep. de 2022 13:02

```
1 | {  
2 |   "d": {  
3 |     "Calculate OUT": 4010,  
4 |     "Número de cajas": 8,  
5 |     "Número de palés": 7  
6 |   }  
7 | }
```

Figura 86. Datos desglosados

Una vez hecho eso, el siguiente y último paso consiste en la creación de paneles de visualización para poder monitorizar dicha información. Para lograr esto, hay que dirigirse a la opción de paneles, que es la primera opción que aparece en la columna de la izquierda. Se le pone un nombre para diferenciar

las tarjetas introducidas en dicho panel., que en este caso será “Subida de datos a IBM Cloud”, una breve descripción si se desea y los miembros que pueden editar dichos paneles de visualización.

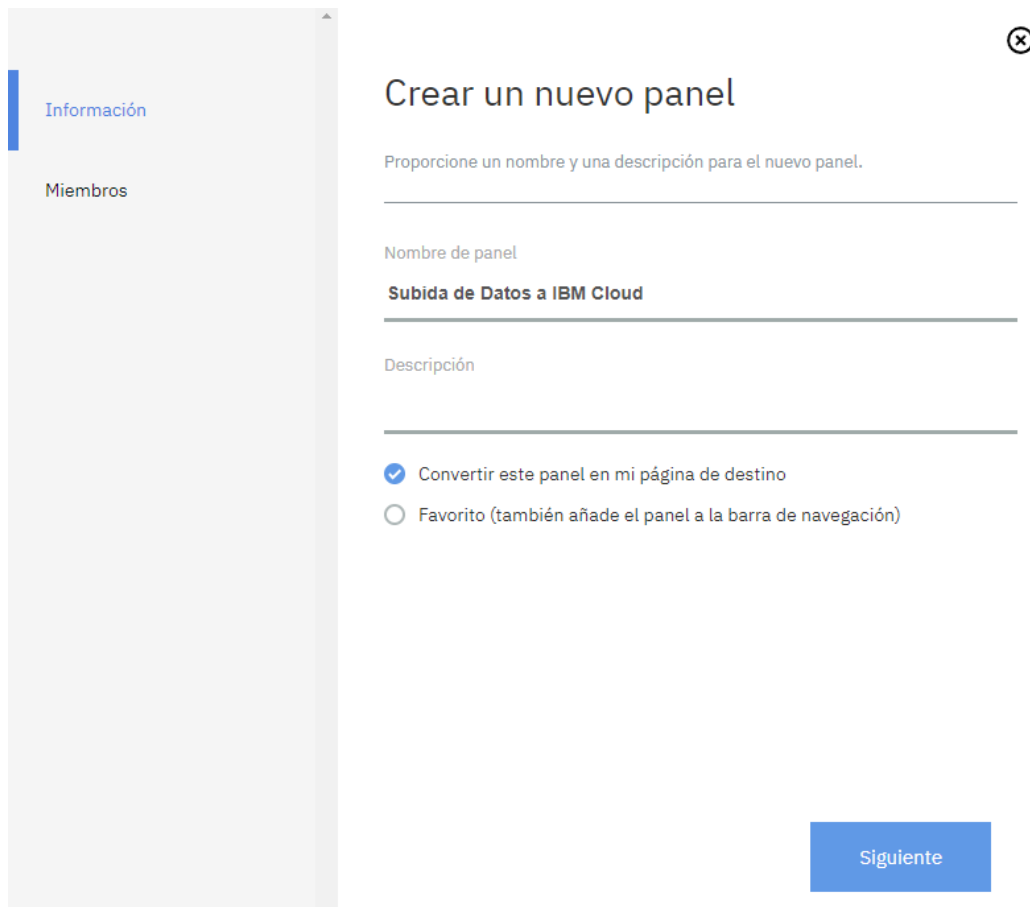


Figura 87. Creación de panel de visualización

Ya se ha creado el nuevo panel, es el primero que se ve en la imagen de abajo, donde dice que no tiene tarjetas, ya que aún no se ha instalado ninguna.

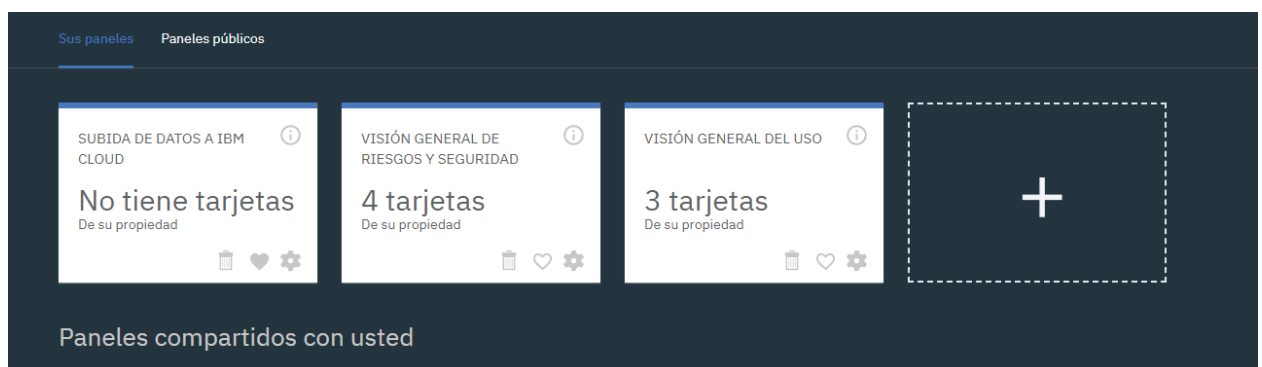


Figura 88. Paneles de visualización disponibles

Entrados al panel, evidentemente este se encontrará vacío. Para empezar a introducir tarjetas para visualizar la información del autómata, hay que seleccionar la opción de “Crear tarjeta”. El servicio de IBM nos ofrece distintos tipos de tarjetas que se pueden utilizar sin restricciones en función de cómo convenga mostrar ese dato, así como un valor, con un gráfico de barras, de líneas, etc. Todas las funcionalidades ofrecidas se pueden apreciar en la siguiente imagen.



Crear tarjeta

Tipo de tarjeta
Seleccionar tipo de tarjeta

Dispositivos



Figura 89. Tipos de tarjetas

Para el caso de este las tarjetas que se muestran en la siguiente figura, se han hecho uso de varias. Para las tres primeras, que muestran el número de cajas, palés y el valor de Calculate OUT en tiempo real, se ha hecho uso de la tarjeta conocida como “Valor”. Por otro lado, para las dos de abajo, mientras que en el gráfico de la izquierda se muestra en un mismo gráfico el número de cajas y palés, en el de la derecha se puede observar el valor de Calculate OUT. Ambos gráficos están en función del tiempo, que se encuentra en el eje x, siendo la hora local del momento en el que se estaban visualizando los datos.

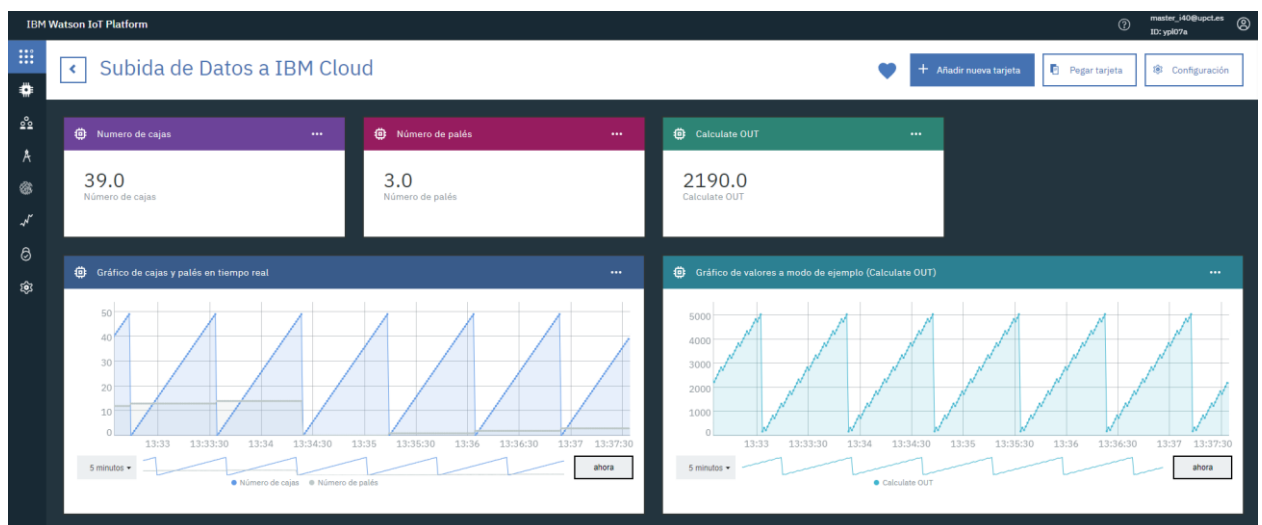


Figura 90. Panel de visualización IBM Cloud

4.5. Conexión a Base de Datos, exportación del fichero CSV y visualización en PowerBI

4.5.1. Descarga, configuración y uso de MySQL y HeidiSQL

En este apartado se va a realizar la subida de datos e información del autómeta a una base de datos mediante Node-RED. Para ello, se hará uso de MySQL y el gestor de base de datos HeidiSQL.

Lo primero, es realizar la instalación de las herramientas. El link de descarga de MySQL es el siguiente: <https://dev.mysql.com/downloads/mysql/>, una vez ahí dentro hay que pinchar en “Go o Download Page” e instalar el archivo seleccionado. Una vez se realiza la instalación, se instala solo el servidor de MySQL, ya que también se puede instalar la versión de cliente, o ambas.

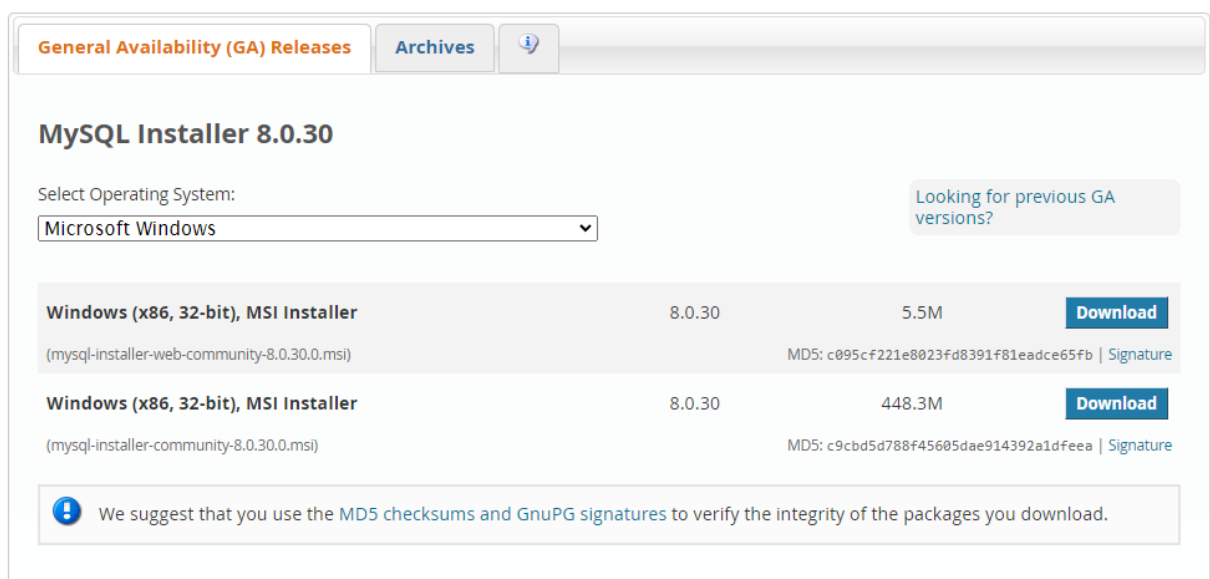


Figura 91. Descarga MySQL

Para instalar el gestor de base de datos HeidiSQL, hay que acceder al siguiente enlace de descarga: <https://www.heidisql.com/download.php?download=installer>. El archivo se descarga automáticamente nada más entrar a la página de descarga de la web oficial.

Download HeidiSQL 12.1, released on 13 Aug 2022

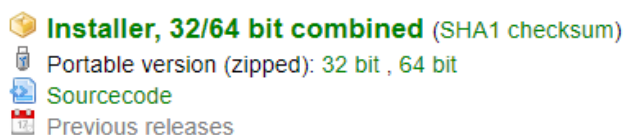


Figura 92. Descarga HeidiSQL

Una vez realizadas ambas descargas e instalaciones, se procede a realizar la subida a la base de datos. Lo primero que hay que hacer es escribir la contraseña seleccionada en la ventana de comandos de MySQL, y así estar dentro del monitor MySQL.

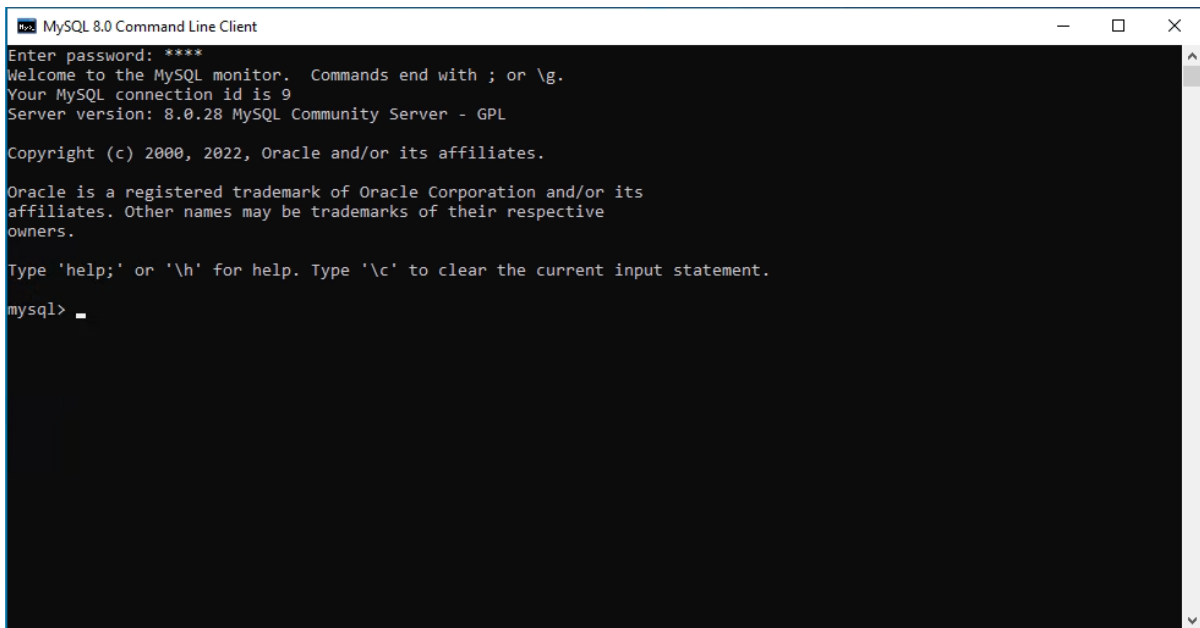


Figura 93. Acceso a MySQL

Para usar el gestor de base de datos, el primer paso es crear una nueva sesión, llamada “SubidaDatosTFM”, a la que se le pone la dirección IP del dispositivo, el usuario, que es “root” y la contraseña puesta. El puerto es el 3306, pero lo pone por defecto HeidiSQL. Luego, se abre y ya debería estar dentro del gestor.

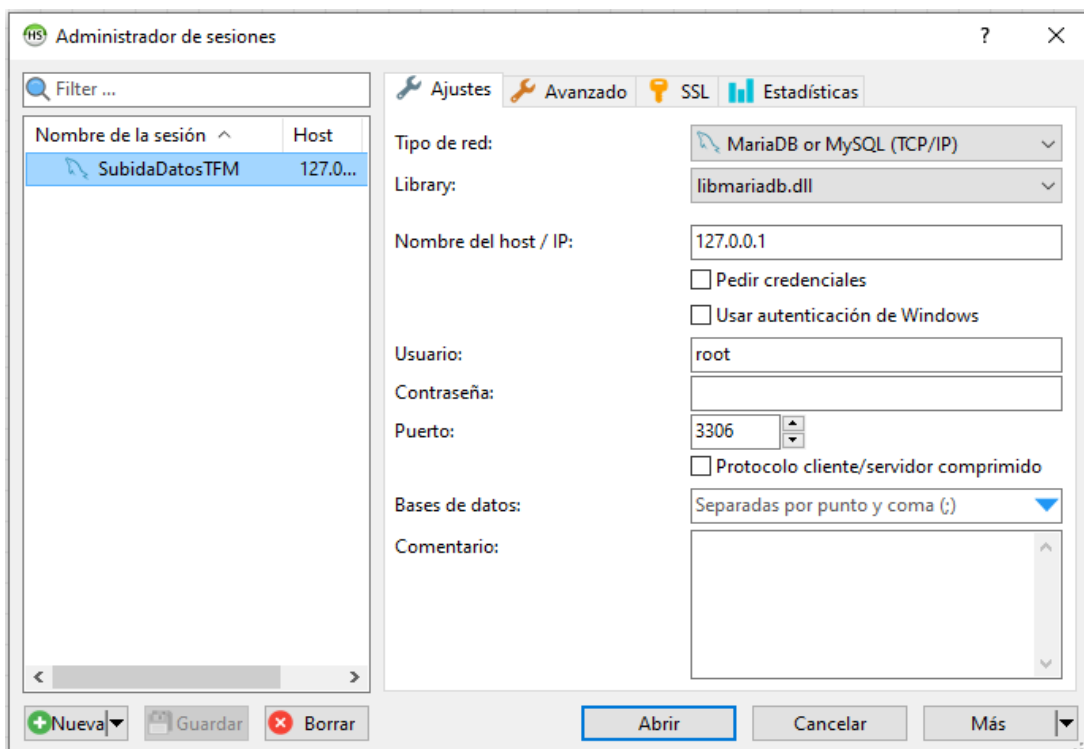


Figura 94. Creación de sesión en HeidiSQL

Una vez dentro, creamos una nueva base de datos, a la cual se la ha denominado “basedatosnodered”, y a la que se le ha creado una tabla llamada “tabla1”.

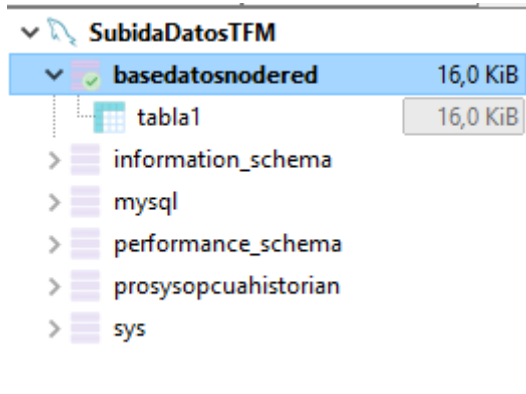


Figura 95. Creación de base de datos y tabla de variables

Cuando se selecciona la base de datos creada, hay que dirigirse a la pestaña de “Tabla: subida de datos” y a “Básico”. Desde ahí hay que agregar tres filas: Calculate OUT, Número de cajas y Número de palés, para este caso en concreto. El tipo de los datos es entero, por lo que se pone “INT” y la relación Longitud/Conjunto la pone la misma HeidiSQL por defecto, y es de 10.

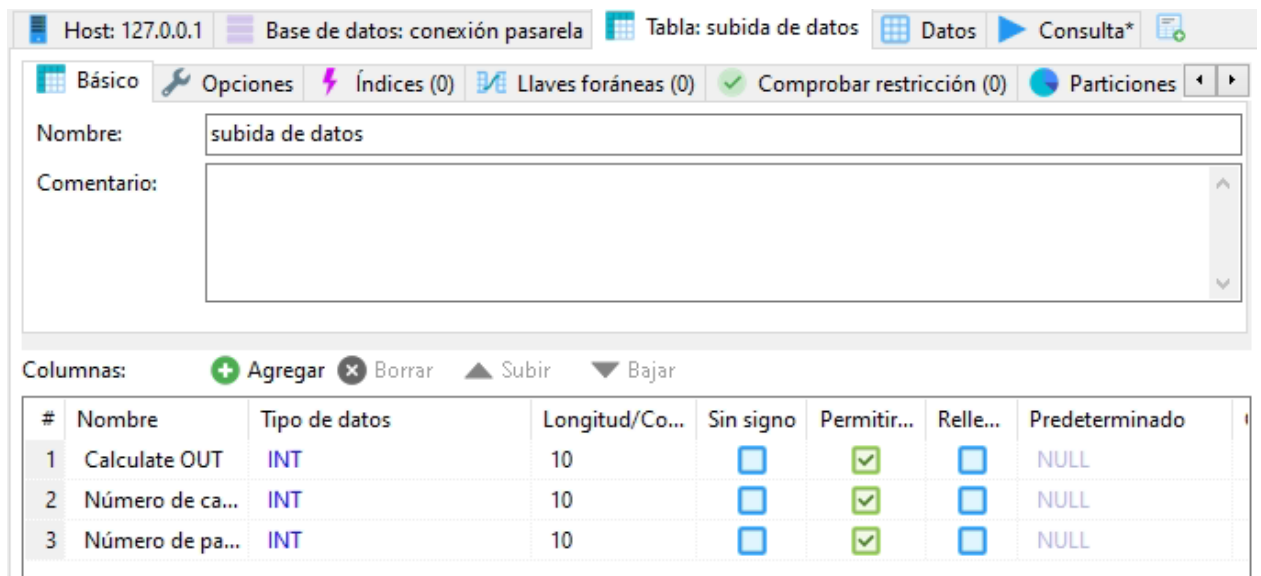


Figura 96. Variables introducidas a la tabla

A continuación, en la ventana de “Consulta”, se escribe el siguiente código para autenticar la base de datos:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH  
`subida de datos`mysql_native_password BY 'UPCT'
```

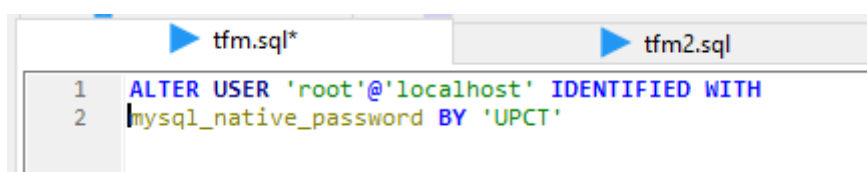


Figura 97. Autenticación de base de datos

Desde la ventana de comandos se puede ver que está creada la base de datos “basedatosnodered” en MySQL. También se comprueba la existencia de la tabla1 y su descripción, donde se ven las tres variables a visualizar en HeidiSQL.

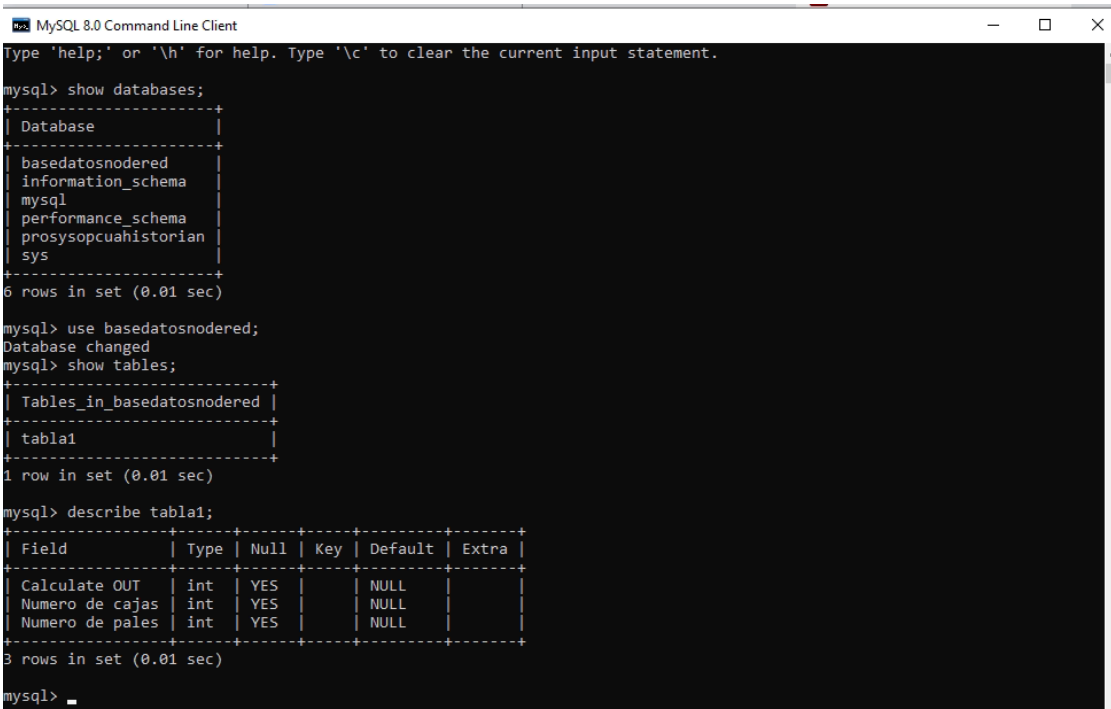
Para poder ver dicha información, hay que utilizar los siguientes comandos:

show databases;

use basedatosnodered;

show tables;

describe tabla1;



```
mysql> show databases;
+-----+
| Database |
+-----+
| basedatosnodered |
| information_schema |
| mysql |
| performance_schema |
| prosysopcuahistorian |
| sys |
+-----+
6 rows in set (0.01 sec)

mysql> use basedatosnodered;
Database changed
mysql> show tables;
+-----+
| Tables_in_basedatosnodered |
+-----+
| tabla1 |
+-----+
1 row in set (0.01 sec)

mysql> describe tabla1;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Calculate OUT | int | YES | | NULL | |
| Numero de cajas | int | YES | | NULL | |
| Numero de pales | int | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)

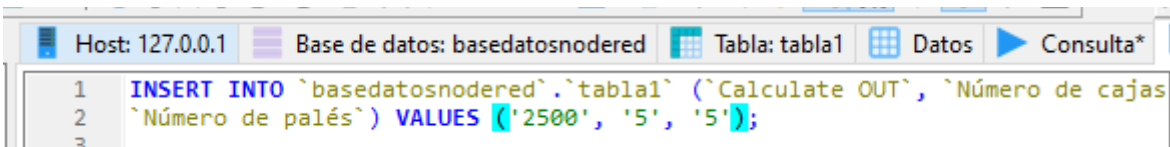
mysql>
```

Figura 98. Información de base de datos y tabla desde comandos en MySQL

Como se puede apreciar, la base de datos y la tabla creados en el gestor HeidiSQL están dentro de MySQL, por lo que ya se pueden utilizar.

Lo siguiente que se hace, para probar que la base de datos funciona, es cargarle un valor a la tabla, el cual se hace con el siguiente código:

***INSERT INTO `basedatosnodered`.`tabla1` (`ID`, `DatoFloat`,
`DatoEntero`, `Cadena`) VALUES ('2', '2.5', '5', 'Hola NodeRed');***



```
Host: 127.0.0.1 Base de datos: basedatosnodered Tabla: tabla1 Datos Consulta*
1 INSERT INTO `basedatosnodered`.`tabla1` (`Calculate OUT`, `Número de cajas
2 `Número de palés`) VALUES ('2500', '5', '5');
3
```

Figura 99. Carga de datos en la tabla

Tras escribir el código hay que darle ejecutar, y se puede ver en la pestaña de datos que este se ha cargado correctamente. El primero de la imagen se ha introducido de manera manual, y los otros tres han sido ejecutando el código anterior varias veces.

Calculate OUT	Número de cajas	Número de palés
2.400	4	4
2.500	5	5
2.500	5	5
2.500	5	5

Figura 100. Datos subidos manualmente

Con el siguiente comando se consigue leer las últimas tres inserciones de la base de datos:

`SELECT *FROM tabla1 ORDER BY ID DESC LIMIT 2;`

```
1 SELECT *FROM tabla1 ORDER BY "Calculate OUT" DESC LIMIT 3;
```

Calculate OUT	Número de cajas	Número de palés
2.400	4	4
2.500	5	5
2.500	5	5

Figura 101. Muestra de datos por correspondencia

4.5.2. Subida de datos desde Node-RED a MySQL y exportación del fichero CSV

Para poder conectar Node-RED a una base de datos de MySQL, lo primero que hay que hacer es decargar el siguiente nodo desde manage palette:

node-red-node-mysql

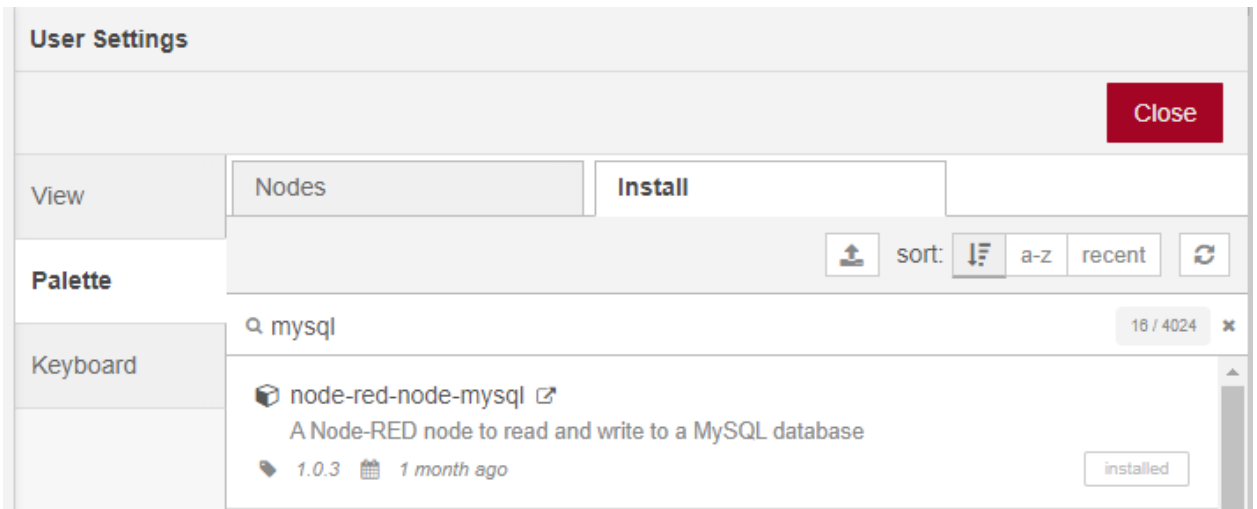


Figura 102. Instalación nodo MySQL

El programa creado es el que se ve en la siguiente imagen. Los tres primeros nodos, los que envían información a Node-RED, son en realidad el mismo nodo utilizado en apartados anteriores, el “s7 in”. La única diferencia es que aquí en vez de mostrar las tres variables agrupadas, se han mostrado de manera individual cada una. Posteriormente, se utiliza el nodo “function”, donde se convierten las variables Calculate OUT, número de cajas y número de palés en flow. Por último, estas se muestran con un debug para cada una.

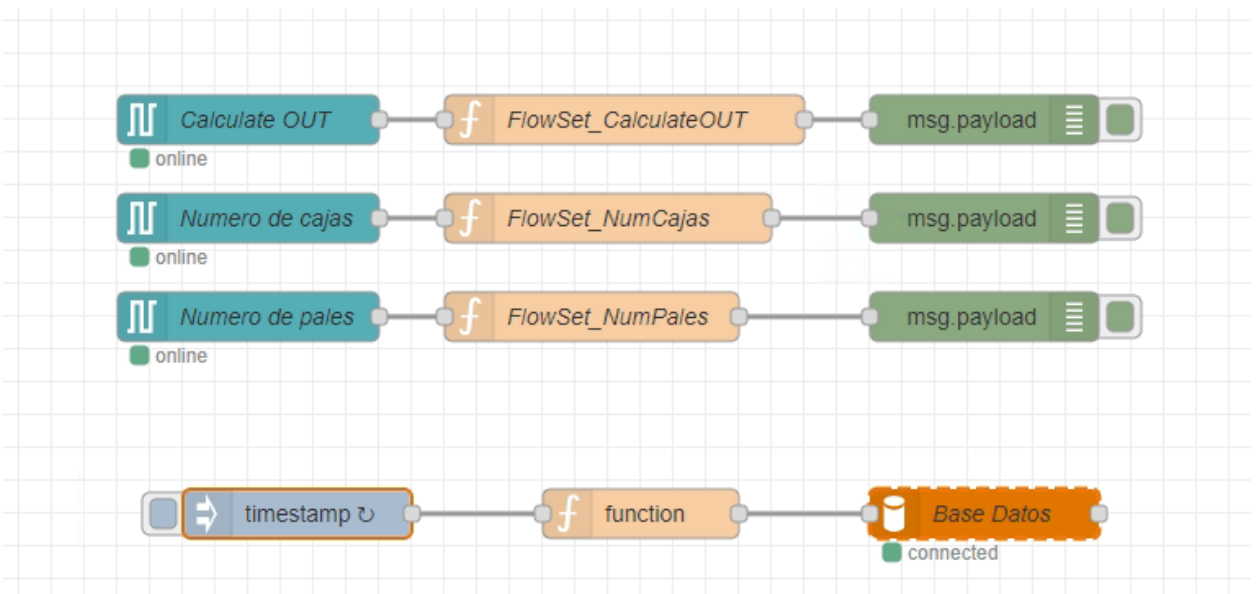


Figura 103. Conexión de Node-RED a la base de datos MySQL

```
1 flow.set("CalculateOUT",msg.payload);  
2 return msg;  
  
1 flow.set("NumCajas",msg.payload);  
2 return msg;  
  
1 flow.set("NumPales",msg.payload);  
2 return msg;
```

Figura 104. Convertir variables en flow

Tal y como se puede ver en la siguiente figura, la información llega sin ningún tipo de problema desde el autómata y, a diferencia de en los otros apartados, cada valor llega separado.

```
6/9/2022, 18:51:27 node: aa1e92da186a5d7a  
Número de cajas : msg.payload : number  
22  
  
6/9/2022, 18:51:28 node: 4cab071634e4b93b  
Calculate OUT : msg.payload : number  
530  
  
6/9/2022, 18:51:28 node: aa1e92da186a5d7a  
Número de cajas : msg.payload : number  
23  
  
6/9/2022, 18:51:29 node: 4cab071634e4b93b  
Calculate OUT : msg.payload : number  
680
```

Figura 105. Datos subidos de manera individual

A continuación, la cuarta línea del código consiste en mandar los datos anteriores a la base de datos MySQL creada. Para lograr dicho objetivo, lo primero que se introduce es un “timestamp”, el cual envía pulsos cada segundo, para que la base de datos esté constantemente recibiendo información. Luego, se vuelve a usar el nodo “function”, pero esta vez para enviar la información a MySQL y poder verla desde el gestor HeidiSQL. Para lograrlo, hay que introducir el siguiente código, que lo único que hace es enviar los valores a las columnas creadas de la tabla1 con la función “flow.get”, para obtener los datos de las variables.

```
1 mensaje = "INSERT INTO tabla1 (Calculate OUT, Numero de cajas, Numero de pales)";  
2 mensaje = mensaje + "VALUES (";  
3 mensaje = mensaje + ""+flow.get("CalculateOUT")+","+flow.get("NumCajas")+"";  
4 mensaje = mensaje + ""+flow.get("NumPales")+"";  
5 mensaje = mensaje + ")";  
6  
7 msg.topic = mensaje;  
8 return msg;
```

Figura 106. Programación para subir los datos a MySQL

Una vez realizado eso, hay que configurar el nodo “mysql”. Lo primero que hay que hacer es poner la dirección del dispositivo. En este caso, al ser desde un mismo dispositivo, la 127.0.0.1. Luego, se introduce el usuario, que es “root”, y la contraseña creada. Para terminar, hay que introducir también la base de datos a la que se va a escribir la información, que es “basedatosnodered”.

The screenshot shows the configuration interface for a MySQL node in Node-RED. The title bar reads "Edit mysql node > Edit MySQLdatabase node". At the top, there are three buttons: "Delete", "Cancel", and "Update". Below this is a "Properties" section with a gear icon and a document icon. The properties are listed as follows:

Host	127.0.0.1
Port	3306
User	root
Password
Database	basedatosnodered
Timezone	±hh:mm
Charset	UTF8
Name	BBDD

Figura 107. Configuración nodo MySQL

Ya estaría realizada la configuración y la programación desde Node-RED, por lo que se despliega el programa y la información debería estar llegando constantemente a la base de datos. Para poder corroborarlo, hay que ir a HeidiSQL y recargar la pestaña de datos.

Se ha dejado un par de minutos para que almacene algo de información para poder trabajar sobre ella y, como bien se puede apreciar, esta ha llegado correctamente a la base de datos.

basedatosnodered.tabla1: 123 filas en total (aproximadamente)

Calculate OUT	Numero de cajas	Numero de pales
4.735	5	3
4.885	6	3
5.035	7	3
185	8	3
335	9	3
235	10	3
385	11	3
535	12	3
685	13	3
835	14	3
735	15	3
885	16	3
1.035	17	3
1.185	18	3
1.335	19	3
1.235	20	3
1.385	21	3
1.535	22	3
1.685	23	3
1.835	24	3
1.735	25	3
1.885	26	3

Figura 108. Datos subidos y visualizados en HeidiSQL

PowerBI se puede conectar a la base de datos MySQL y trabajar directamente con ella, pero para este caso se van a exportar las filas para trabajar desde un Excel. Para ello, se selecciona toda la tabla, se presiona clic derecho y se selecciona la opción de importar filas.

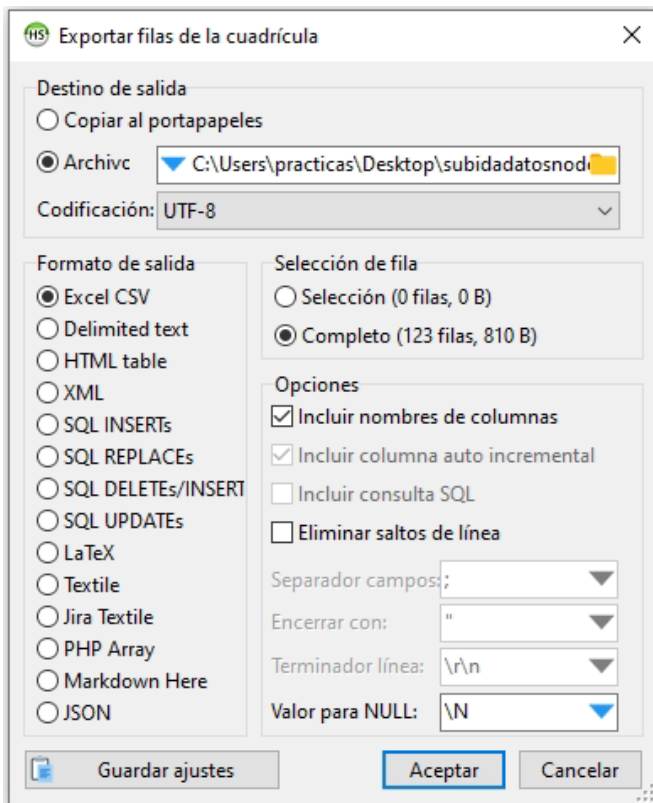


Figura 109. Exportar archivo CSV

Se selecciona la ubicación donde se desea exportar el archivo, se elige el formato, que va a ser Excel CSV. Luego, se indica que va a ser una selección de fila completa, para cogerlas todas y se acepta la configuración. El archivo se habrá guardado correctamente en el ordenador del usuario.

Se comprueba en la dirección donde se ha ubicado el archivo, y tal y como se observa en la imagen, el archivo ha sido exportado de manera correcta y ya se puede trabajar con él en PowerBI.

	A	B	C
1	Calculate OUT	Numero de cajas	Numero de pales
2	4735	5	3
3	4885	6	3
4	5035	7	3
5	185	8	3
6	335	9	3
7	235	10	3
8	385	11	3
9	535	12	3
10	685	13	3
11	835	14	3
12	735	15	3
13	885	16	3
14	1035	17	3
15	1185	18	3
16	1335	19	3
17	1235	20	3
18	1385	21	3
19	1535	22	3
20	1685	23	3
21	1835	24	3
22	1735	25	3
23	1885	26	3
24	2035	27	3
25	2185	28	3
26	2335	29	3
27	2235	30	3
28	2385	31	3
29	2535	32	3
30	2685	33	3
31	2835	34	3
32	2735	35	3
33	2885	36	3

Figura 110. Datos en Excel

4.5.3. Creación de paneles de visualización desde PowerBI

Para trabajar con PowerBI, lo primero que hay que hacer es importar el archivo excel obtenido desde la base de datos MySQL. Antes de cargar el fichero, esta herramienta nos muestra la tabla para comprobar si está correcta, antes de empezar a manipularla.

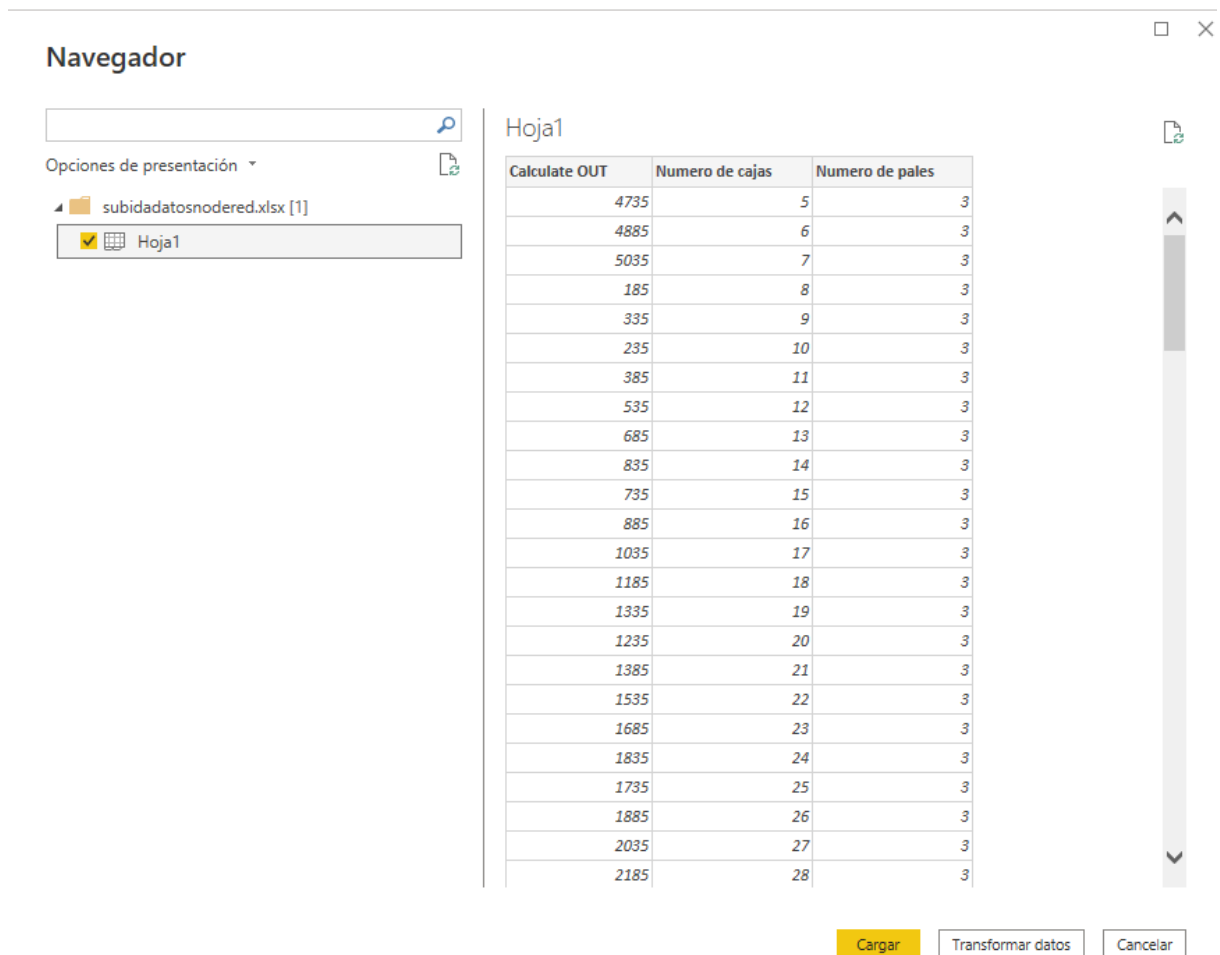


Figura 111. Importar y cargar datos a PowerBI

En la parte izquierda del programa, aparecen tres símbolos. El primero de los tres es el entorno en el que se va a trabajar e introducir los paneles de visualización. El segundo, por otro lado, muestra al usuario la tabla de valores que se ha cargado, la cual se puede manipular introduciendo nuevas columnas, borrando otras, creando relación entre una columna y otra, y un largo etcétera. Por último, el tercer símbolo muestra las hojas de datos cargadas en el programa. Esto es una herramienta bastante interesante, aunque ahora no se le va a sacar un gran provecho, ya que se pueden relacionar distintas variables de distintos Excel cargados en el programa, lo cual podría derivar en unos paneles de visualización mucho más interesantes y llamativos para el usuario o cliente final.

Calculate OUT	Numero de cajas	Numero de pales
473.5	5	3
488.5	6	3
503.5	7	3
18.5	8	3
33.5	9	3
23.5	10	3
38.5	11	3
53.5	12	3
68.5	13	3
83.5	14	3
73.5	15	3
88.5	16	3
103.5	17	3
118.5	18	3
133.5	19	3
123.5	20	3
138.5	21	3
153.5	22	3
168.5	23	3

Figura 112. Manipulación de datos y opción de conexión con otros archivos Excel

Antes de trabajar con el entorno, como se ha mencionado, se han manipulado los datos cargados. Se ha añadido una columna de tiempo, en segundos, para así poder mostrar las variables en función del tiempo, que es lo que verdaderamente interesa ahora mismo en este proyecto. En la imagen de abajo se puede ver la nueva columna integrada.

Calculate OUT	Numero de cajas	Numero de pales	Tiempo (s)
473.5	5	3	1
488.5	6	3	2
503.5	7	3	3
18.5	8	3	4
33.5	9	3	5
23.5	10	3	6
38.5	11	3	7
53.5	12	3	8
68.5	13	3	9
83.5	14	3	10
73.5	15	3	11
88.5	16	3	12
103.5	17	3	13
118.5	18	3	14
133.5	19	3	15

Figura 113. Columna Tiempo (s) añadida

Para introducir paneles, tenemos que hacer uso de las herramientas de visualización que se muestran en la siguiente imagen. En comparación a IBM Cloud, PowerBI ofrece una mayor variedad de herramientas. Entre ellas se encuentran las más conocidas, como gráficos de líneas o de áreas. Por otro lado, también hay gráficos de columnas y de barras, además de una combinación bastante interesante entre gráficos de barras con gráficos de líneas.

Aparte de todo ese tipo de gráficos, hay uno bastante llamativo, que es el mapa coroplético, el cual representa los datos no solo en un eje, que también, sino en un mapa, sea del lugar que sea. Esta herramienta puede considerarse bastante importante en inteligencia de negocios, ya que, a la hora de abrir una empresa, fábrica o, simplemente a la hora de comercializar un producto, es muy indispensable saber lo propensa que puede llegar a ser una ubicación en utilizar unos servicios u otros.

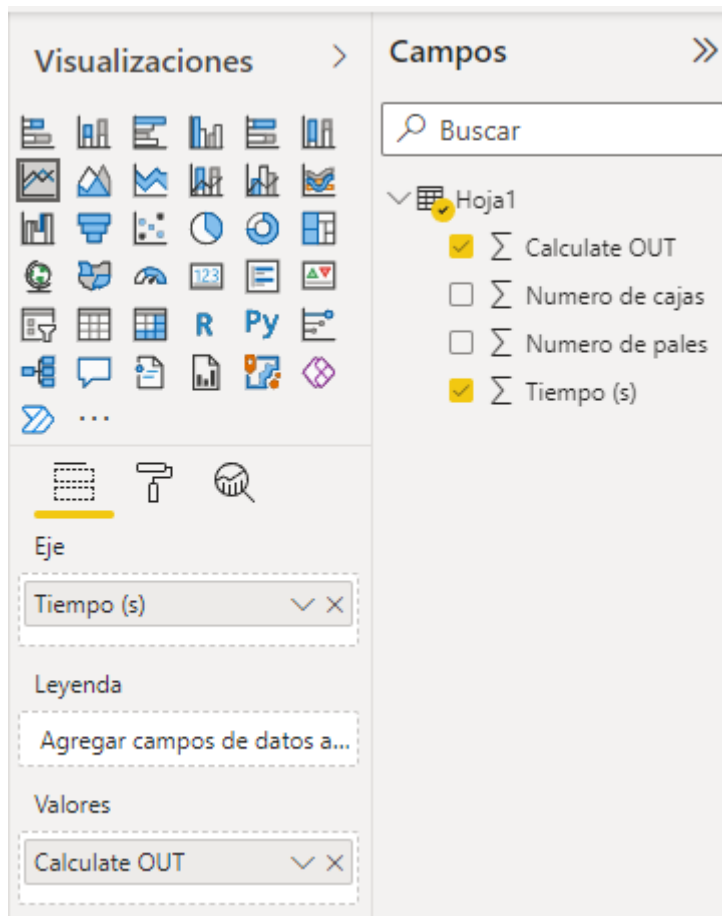


Figura 114. Diferentes tipos de paneles de visualización

5. ANÁLISIS DE RESULTADOS EN PLATAFORMA DE INTELIGENCIA DE NEGOCIO

En esta parte del trabajo, se van a mostrar los paneles realizados en los servicios de IBM Cloud y PowerBI, y así poderlos comparar un poco, ya que cada uno tiene sus ventajas e inconvenientes.

5.1. Visualización de datos en tiempo real en IBM Cloud

Estos son los paneles que se han creado para IBM Cloud. Aparte de los mostrados en el apartado anterior, aquí se han creado algunos más para poder ver cuánto ofrece este servicio. Además de los explicados anteriormente, hay más opciones. Por ejemplo, en vez de poner tres tarjetas para mostrar el valor en texto en tiempo real, se puede poner únicamente uno donde se muestran todas las propiedades del dispositivo conectado a la nube.

Además, hay tarjetas como puede ser la nombrada como “Indicador”, que muestra los valores en un rango de mínimo a máximo. En este caso, se ha utilizado para la variable Calculate OUT y es la tarjeta que se encuentra a la izquierda de la imagen.

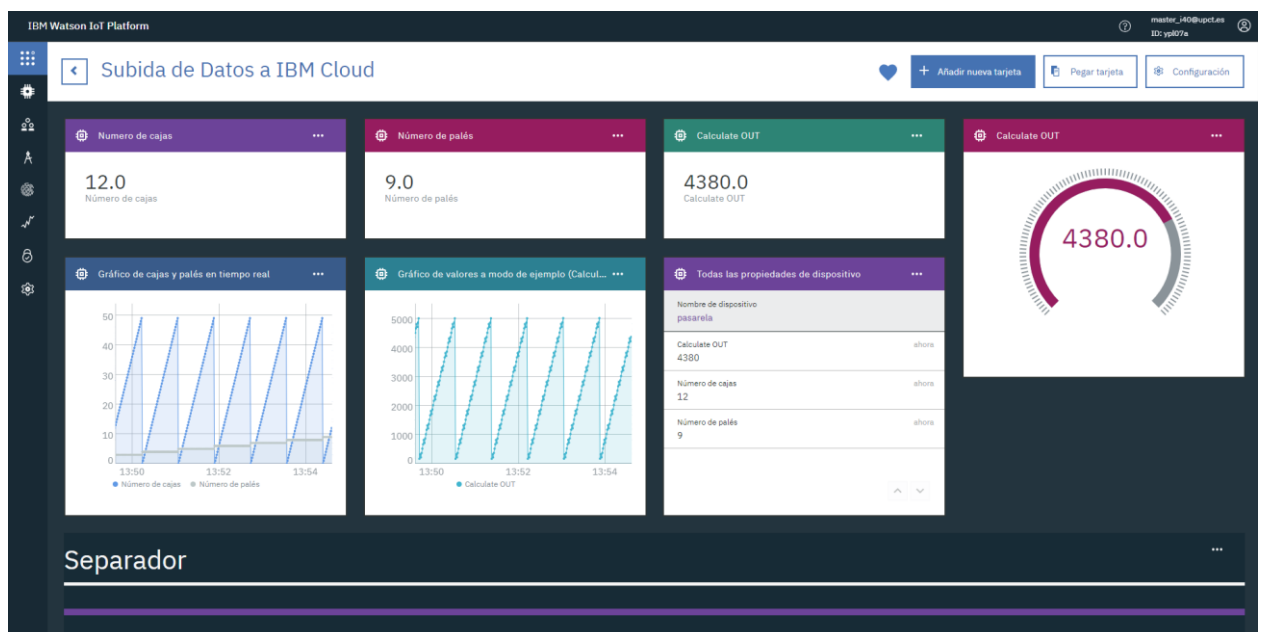


Figura 115. Paneles IBM Cloud

Aparte de todos los paneles ofrecidos, IBM también proporciona un separador, el cual, aunque tampoco sea indispensable, puede llegar a ser útil en ciertas situaciones si se quiere diferenciar varios procesos. Así, por tanto, no se confundirá al usuario final y puede dar una imagen de mayor orden en la distribución de los paneles.

Por otro lado, IBM también ofrece ciertos paneles que indican al usuario los datos que se han transferido a la nube en el tiempo. En la imagen se puede apreciar distintas maneras de ver dicha información en los paneles. Se puede ver desde una tarjeta simple, que muestre los datos transferidos en un día, en el mes y en el mes anterior. También se puede hacer algo más detallado, que es mostrar el valor de datos subidos cada día. Por último, y la mejor opción, es representar dichos valores en un gráfico

que muestre la evolución de los datos transferidos cada día, así se puede apreciar de una manera visual y clara cuánto se ha trabajado con la plataforma de IBM Cloud y los picos más altos del mes.

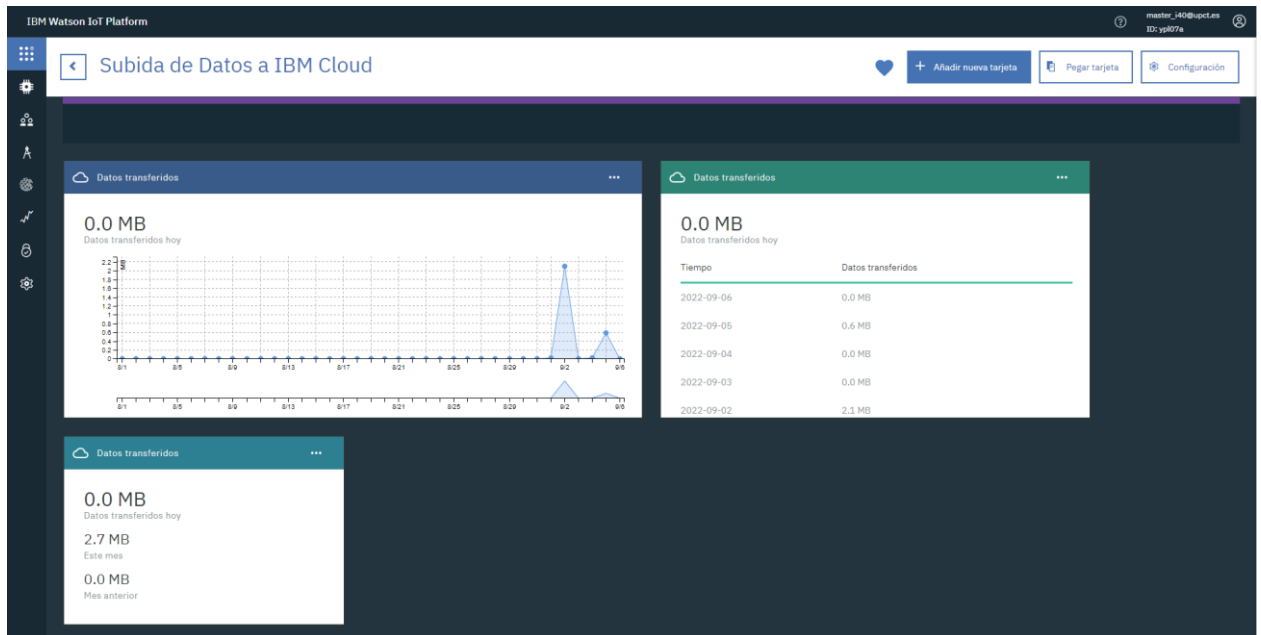


Figura 116. Más opciones de paneles en IBM Cloud

Una característica a destacar de estos paneles es que son interactivos, por lo que si el usuario se aproxima al punto que le interesa, podrá ver la información detallada.

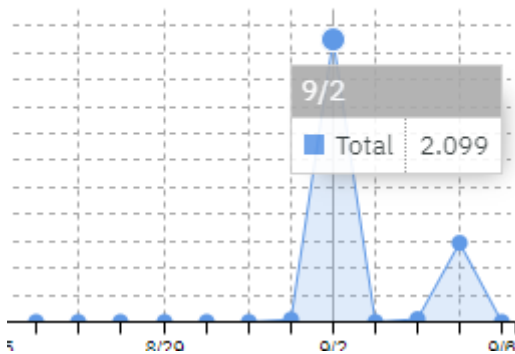


Figura 117. Cantidad de datos enviados. Panel interactivo en IBM Cloud

Por ejemplo, aquí se puede apreciar que el 2 de septiembre se envió una cantidad de datos de 2,099 MB.

Tras ver esto, se puede concluir que IBM Cloud es una opción muy óptima para visualizar y monitorizar valores en tiempo real. Además, la configuración de sus paneles es extremadamente sencilla, solo hay que seleccionar las variables a mostrar y no genera ningún tipo de problemas en ese aspecto.

5.2. Visualización de datos históricos en PowerBI

En cuanto los paneles de visualización de PowerBI, se han realizado los siguientes:

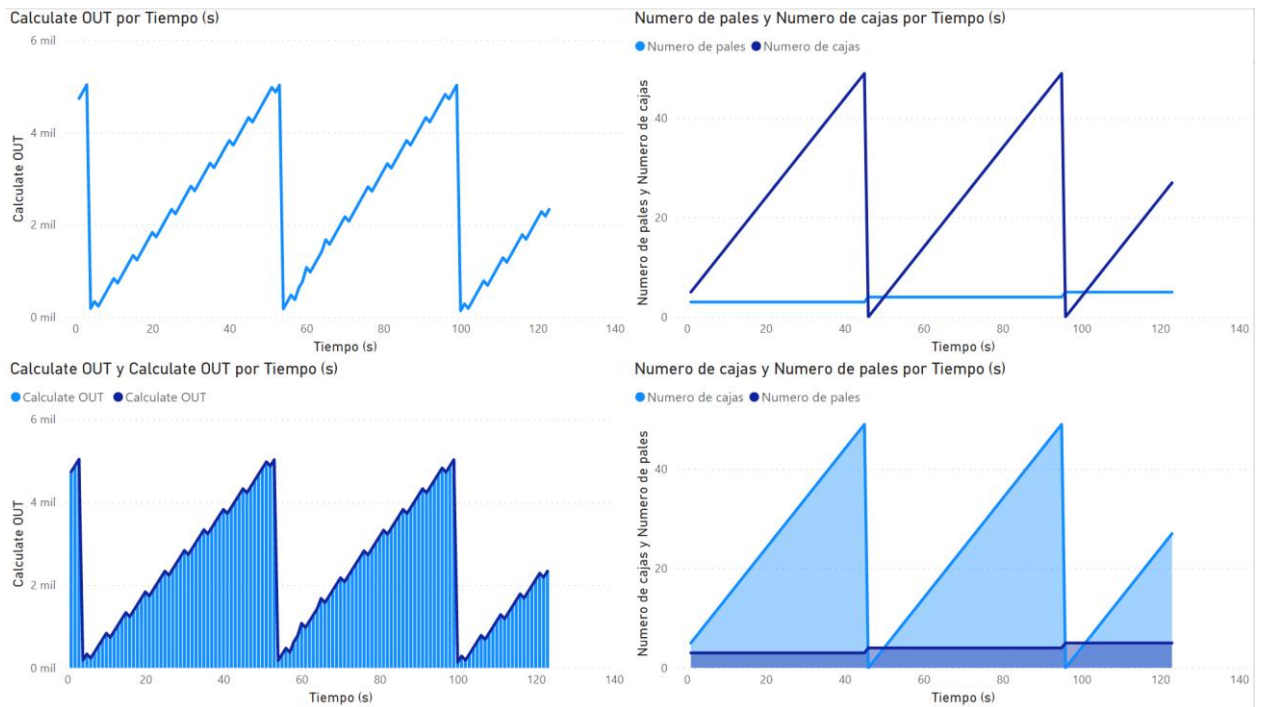


Figura 118. Paneles de visualización en PowerBI

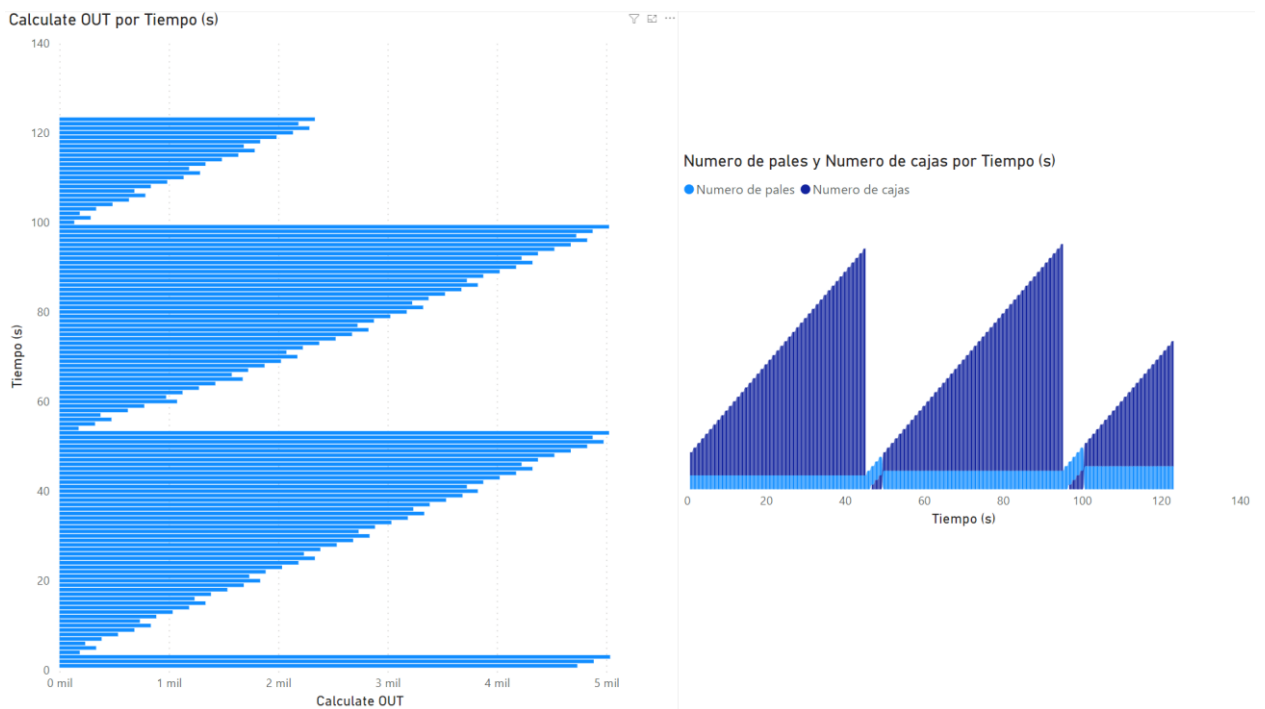


Figura 119. Más paneles de visualización en PowerBI

En comparación a IBM Cloud, PowerBI tiene un abanico de paneles más amplio que el anterior, por lo que podría ser una mejor opción a la hora de visualizar determinada información. Sin embargo, el proceso para poder mostrar los datos en el programa es bastante más complicado, ya que no solo hay que ser capaces de enviar la información a una base de datos, sino que luego hay que conectarla a PowerBI, o, en su defecto, importar un CSV con el que poder trabajar y manipular dentro del programa.

Una ventaja que tiene PowerBI con respecto al servicio de IBM Cloud consiste en que es bastante más manipulable, ya que se pueden intercambiar las variables o manipularlas. Además, también se pueden poner como se desee en el gráfico. Se puede colocar tanto en el eje x como en el eje y, meter más de una variable por eje, leyendas, líneas de regresión, y un largo etcétera.

Al igual que pasa con IBM Cloud, en PowerBI también se puede ver información detallada de ciertos puntos que puedan ser interesantes para cualquier usuario. La diferencia, en beneficio de PowerBI, es que con esta herramienta, cada vez que se selecciona un punto, este también se muestra en todas las gráficas donde esté incluido, por lo que se puede ver una relación aún mayor entre gráficos y entender el contexto de cualquier dato con muchísima más facilidad.

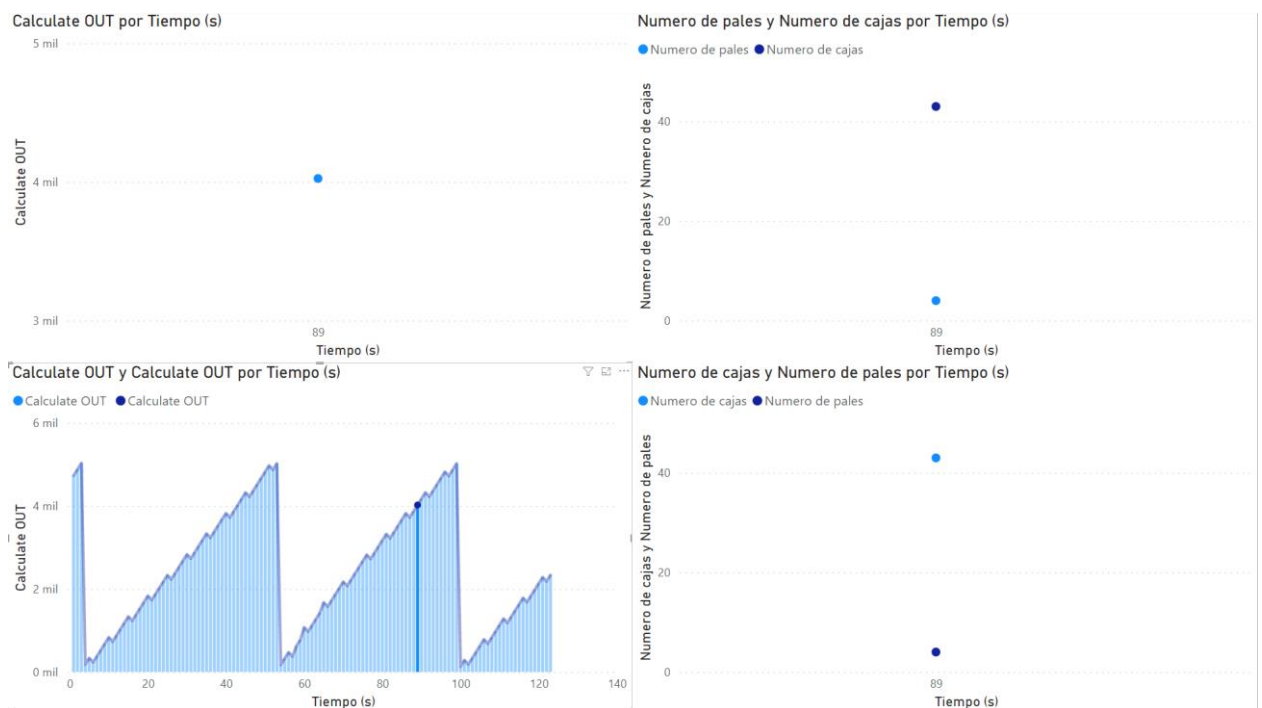


Figura 120. Paneles interactivos en PowerBI

Por ejemplo, tal y como se puede ver en la siguiente imagen, se selecciona una barra en el segundo 89 del proceso y, con eso, se indica el valor de la variable Calculate OUT, que es 4025. Además, también proporciona información del número de cajas (43) y palés (4) en ese mismo instante. Teniendo en cuenta que estas otras dos variables pertenecen a otra gráfica, es una característica bastante notoria y a tener en consideración.

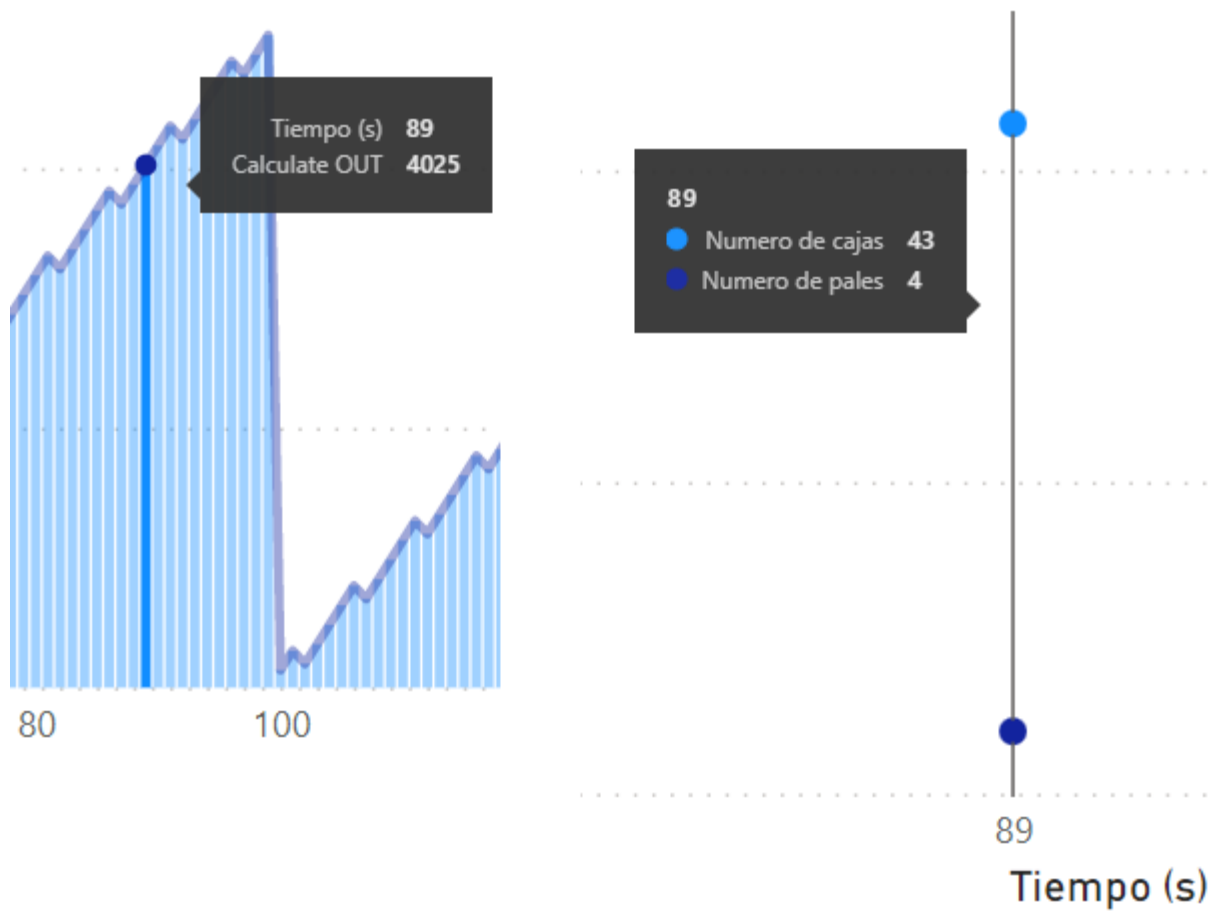


Figura 121. Información detallada en paneles interactivos de PowerBI

Concluyendo, se puede decir que la plataforma de PowerBI ofrece más herramientas que IBM Cloud, además de más funcionalidades y mayor detalle. Sin embargo, es bastante más difícil de manejar y no muestra valores en tiempo real, solo históricos. Por tanto, si se quiere monitorizar un autómeta en tiempo real, esta no es la herramienta idónea. Sin embargo, si lo que se desea es obtener un histórico de ventas o información de ciertas máquinas a lo largo del tiempo, este programa es idóneo para prácticamente cualquier usuario interesado.

6. CONCLUSIONES

Como conclusiones obtenidas tras la realización de este Trabajo Final de Máster, considero que se han cumplido los objetivos explicados en el primer capítulo del trabajo.

En el segundo capítulo se ha hecho una recopilación de todos los dispositivos, softwares y servicios que podrían ser útiles en el salto de cualquier empresa a la Industria 4.0. En este capítulo se ha recorrido desde la familia de autómatas de SIEMENS, las pasarelas IOT; múltiples programas como OPC UA e Ignition; y distintas bases de datos, como IBM Cloud y Microsoft Azure. Por otro lado, se ha comentado distintos dispositivos con el que se podría llevar perfectamente a cabo distintos proyectos de escala menor y de menor exigencia.

En el resto de capítulos, del 3 al 5, es donde se ha encontrado la parte principal del proyecto, explicando la arquitectura a utilizar, y su procedimiento desde cero. En esta parte se ha mostrado paso a paso cómo configurar y montar un dispositivo IOT 2050 desde 0, creando su imagen y configurando la dirección IP para conectarse a él. Posteriormente, se ha mostrado cómo poder acceder a Node-RED desde la misma pasarela y la instalación de distintos nodos para poder realizar el proyecto.

Por último, se ha detallado cómo se realiza el proceso de subida de datos desde un autómata hasta el servicio de IBM Cloud. Eso pasa desde cargar el programa en el S7-1200 desde TIA Portal a cómo trabajar desde el servicio de Internet de las cosas ofrecido en el catálogo de IBM Cloud. Entre medias, se han configurado los nodos que comunican el entorno local del autómata y pasarela con la nube, donde se encuentran infinidad de proyectos y utilidades gracias a IBM Cloud, y muchos más servicios.

En resumen, considero que esta memoria de un proyecto piloto puede servir de referencia a prácticamente cualquier empresa que se quiera animar a dar el salto a IIOT y a la Industria 4.0. Por lo tanto, puedo decir que estoy bastante satisfecho del trabajo realizado tanto en las prácticas de empresa como en el Proyecto de Final de Máster.

7. TRABAJOS FUTUROS

Sobre posibles futuros trabajos, gracias al salto proporcionado con la llegada de IOT y la Industria 4.0, se ha abierto la ventana de oportunidades para una infinidad de trabajos y nuevos proyectos. Este proyecto recoge lo básico para poder dar los primeros pasos con estas nuevas herramientas, pero sin embargo hay mucho más allá.

Lo primero a comentar es que una vez se empieza a realizar un proyecto de esta envergadura, puedes llegar a encontrar un amplio abanico de posibilidades y muchísimas maneras de realizar cualquier proyecto, ya que, aparte de lo realizado en esta memoria, se pueden incluir muchas cosas más. El mismo servicio de IBM tiene su propio entorno Node-RED, en la nube, por lo que un posible proyecto podría ser en llevar la subida de información no solo a una base de datos, sino a ese entorno de Node-RED en la nube, ya que desde ahí se podría entrelazar con multitud de proyectos similares, y las posibilidades se incrementan de una manera soberbia. Por otro lado, también hay funcionalidades más simples, sencillas y que llevan menos tiempo, como podría ser la comunicación del usuario con cualquier autómata desde Telegram, ya que hay distintas herramientas que permiten al usuario poder recibir información en tiempo real de la máquina. Además, con esta herramienta, se podría incluso mandar información desde el mismo

dispositivo móvil cambiando, por ejemplo, la velocidad de llenado de un depósito, frecuencia de un motor, y un largo etcétera.

Por otro lado, no solo hay que cerrarse en este entorno de SIEMENS con un PLC S7-1200 y un IOT 2050, ya que hay muchos más hardwares a tener en consideración que, aun pudiendo llegar a ser menos potentes, pueden llegar a cumplir con determinadas tareas sin ningún tipo de problemas. Los principales dispositivos han sido mencionados en el capítulo 2 de la memoria.

Saliendo del entorno de SIEMENS, el dispositivo que más me ha llamado la atención y es con el que me gustaría profundizar también en un futuro, es la Raspberry Pi 4. Considero que, aun teniendo un hardware bastante más limitado que el de cualquiera de la familia de IOT 2000, podría ser una buena opción, ya que también dispone de Node-RED, por lo que se podría realizar el mismo proyecto con esa tarjeta. Además, es bastante más económico, ya que la versión más cara no llega a los 100€, mientras que el IOT 2020, que es el más barato de la familia, supera los 200€, duplicando el precio. Por tanto, aun siendo una opción peor, el tema económico en proyectos con presupuesto más limitado podría ser un factor diferencial para decantarse por uno u otro.

8. BIBLIOGRAFÍA

- 1) Información del PLC de SIEMENS SIMATIC S7-1200.
<https://new.siemens.com/mx/es/productos/automatizacion/systems/industrial/plc/s7-1200.html>
- 2) Información de IOT 2050.
<https://es.rs-online.com/web/p/gateways-iot/2017731>
- 3) Información de LUTZE ET-SWU5E.
<https://www.luetze.com/es-es/catalogo/Control/Comunicaci%C3%B3n-industrial/772004/>
- 4) Información de IOT 2020.
<https://es.rs-online.com/web/p/gateways-iot/1244037>
- 5) Información de IOT 2040.
<https://es.rs-online.com/web/p/gateways-iot/1244038>
- 6) Información de Raspberry Pi 4.
<https://www.xataka.com/ordenadores/raspberry-pi-4-caracteristicas-precio-ficha-tecnica>
- 7) Información de ESP 32.
<https://www.prometec.net/instalando-esp32/>
- 8) Información de ESP 8266.
<https://aprendiendoarduino.wordpress.com/2017/09/12/que-es-esp8266/>
- 9) Información del PLC de SIEMENS SIMATIC S7-200.
https://es.rs-online.com/web/p/controladores-plcs-y-automatas/4886735?cm_mmc=ES-PPC-DS3A-_google-_DSA_ES_ES_Automatizaci%C3%B3n+y+Control+de+Procesos_Index-_Controladores+%26+PLCs+y+Aut%C3%B3matas%7C+Products-_DYNAMIC+SEARCH+ADS&matchtype=&aud=821594433763:dsa-1642115163260&gclid=Cj0KCQjw0oyYBhDGARIsAMZEuMv5iQOw9IyHb5jIbO38yGU9f8j58Y5CS-YJ9zuktNjNzKXH2LVM83YaAvHQEALw_wcB&gclsrc=aw.ds
- 10) Información del PLC de SIEMENS SIMATIC S7-300.
<https://es.rs-online.com/web/p/controladores-plcs-y-automatas/7877992>
- 11) Información del PLC de SIEMENS SIMATIC S7-400.
<https://support.industry.siemens.com/cs/mdm/1117740?c=87898482571&lc=es-NI>
- 12) Información del PLC de SIEMENS SIMATIC S7-1500.
https://es.rs-online.com/web/p/controladores-plcs-y-automatas/8199717?cm_mmc=ES-PPC-DS3A-_google-_DSA_ES_ES_Automatizaci%C3%B3n+y+Control+de+Procesos_Index-_Controladores+%26+PLCs+y+Aut%C3%B3matas%7C+Products-_DYNAMIC+SEARCH+ADS&matchtype=&aud=827186183886:dsa-1642115163260&gclid=Cj0KCQjw0oyYBhDGARIsAMZEuMt39kU_42w--iBu7FHiolG_OBA8VaNGEtKI6CXwuXp8AjNx_vS5S4EaAm0UEALw_wcB&gclsrc=aw.ds

13) Instalación pasarela IOT 2050.

<https://support.industry.siemens.com/cs/document/109741799/downloads-for-simatic-iot20x0?dti=0&lc=en-WW>

14) Instalación imagen sobre IOT 2050.

https://support.industry.siemens.com/cs/attachments/109741799/IOT2050_Example_Image_V1.1.1.zip

15) IOT – Publicar datos (Arduino) a la nube por IBM Cloud – Watson. Vídeo del canal de YouTube de Jesús Correa – PERU.

<https://www.youtube.com/watch?v=x97NHVnuRO8&t=835s>

16) MySQL Server - PLC S7 1200 (Crear bases de datos para almacenar informacion de PLC Siemens). Vídeo del canal de YouTube de Jesús Correa – PERU.

<https://www.youtube.com/watch?v=i7sGt47Fpvc&t=1462s>