

# “Gestión de Protocolos de Internet para Aprendizaje Profundo de Datos en Dispositivos IoT Aplicados a Parámetros Ambientales”

“Internet Protocol Management for Deep Data Learning in IoT Devices Applied to Environmental Parameters”  
(Iot = Internet of Things – Internet de las Cosas)\*

Ing. Javier A. Ouret, UCA *Facultad de Ingeniería*  
*Colaborador en el Proyecto para su Trabajo Final: Alumno Luciano Parodi*

## **Informe de Actividades al 15 de Febrero de 2023**

### **Breve introducción sobre el trabajo.**

La gestión de dispositivos interconectados por Internet requiere de múltiples niveles de abstracción en cuanto las aplicaciones y protocolos a utilizar. El número de dispositivos ha crecido exponencialmente en los últimos años forzando el desarrollo de nuevas arquitecturas para poder gestionar la operación y los datos adquiridos. Las tecnologías digitales están presentes en la vida diaria con noticias, información, redes sociales sobre la cuales tenemos un elevado nivel de percepción de su existencia. En el caso de las tecnologías de IoT\* la situación es completamente diferente en cuanto a su percepción, ya que en la mayoría de los casos ni nos damos cuenta de su existencia. IoT\* viene a ser la parte física del mundo digital, monitoreando y colectando datos físicos, operando dispositivos, enviando datos físicos a la nube, etc. Existen dispositivos IoT\* que se ocupan de mostrarnos en tiempo real los precios de cada producto de un supermercado, otros que miden parámetros de medio ambiente, o que controlan el estado del motor de nuestro vehículo, que hacen trazabilidad con GNSS de flotas de transporte, que analizan las vibraciones de las aspas de los generadores solares, que monitorean pacientes en hospitales, etc. El IoT\* tiene una gran diferencia con los dispositivos tradicionales: pueden enviar la información directamente a Internet, ya sea por una conexión directa con su propia dirección IP o por medio de gestores o “brokers” intermediarios a los cuales acceden por Bluetooth o WiFi o por redes celulares 4G/LTE/5G. Otro aspecto es que precisamos nuevos protocolos pues desde el punto de vista de la Internet los protocolos de gestión, control y aprovisionamiento como SNMP/Netconf/TR69 (para el monitoreo, configuración, actualizaciones de firmware, alarmas, etc.) estaban enfocados en dispositivos tradicionales como conmutadores (switches), ruteadores, módems, servidores. El uso de dispositivos IoT\* hizo que el número de nodos y la cantidad de información a gestionar tuviera un aumento exponencial. La gestión ahora involucra también el acceso a datos sin control de estado (stateless) bajo arquitecturas como REST, la gestión de la escalabilidad, el uso de intermediarios (brokers), etc. Por lo tanto, se requiere de técnicas, protocolos y lenguajes adecuados para poder administrar cantidades masivas de recursos, con arquitecturas cliente-servidor con control de estado y modelos de acceso de datos persistentes tienen que ser complementada con modelos sin control de estado, escalables como REST. El objetivo de este trabajo es investigar y proponer técnicas que combinen el uso de protocolos sobre internet para gestión remota de IoT\* con el aprendizaje profundo de datos, para cantidades masivas de dispositivos. Para el desarrollo del trabajo se propone usar sensores de medio ambiente para la captura de datos (CO2, Humedad, Temperatura, etc.), conectados a Internet y gestionados con SQL/MQTT/SNMP/Netconf para el

monitoreo/configuración de los dispositivos y el uso la arquitectura REST con MQTT sobre modelos cliente servidor sin control de estado para las operaciones sobre los datos. A los efectos de evaluar los resultados los sensores registrarán las condiciones ambientales en espacios cerrados utilizados por personas (aulas, oficinas, hospitales, restaurants, cines), que calcularán el nivel de riesgo para la propagación de agentes patógenos. Estos sensores estarán conectados a gateways con acceso a Internet por fibra óptica y 4G/LTE/5G.

## **Palabras claves**

IoT, Internet of Things, IoT Sensor, Raspberry Pi, Rest API, Machine Learning, Artificial Intelligence, Netconf, Carrier Ethernet, MQTT, IoT security.

## **Importancia estratégica.**

La importancia de IoT\* radica en que los usuarios pueden tener el control sobre distintos dispositivos usando Internet, aunque no estén cerca de ellos. De a poco se va conectando a internet todo lo que el ser humano hace, lo que puede permitir mejoras en la productividad, mayor seguridad en la operación de equipamientos y se abren nuevas oportunidades para la creatividad. Al mismo tiempo es importante desarrollar herramientas que permitan a las personas saber qué es lo que se está controlando, cuál información se recolecta y cómo será utilizada. La privacidad y protección de los datos deber ser un objetivo primordial en el control de los despliegues masivos de dispositivos IoT\*. Los protocolos para IoT están normalizados por el IETF (Internet Engineering Task Force), OASIS (Organization for the Advancement of Structured Information Standards), ISO, IEEE, etc.

## **Estado actual del conocimiento sobre el tema**

La internet de las cosas (Internet of Things) es un concepto que engloba a la interconexión digital de objetos cotidianos, sensores, software y otras tecnologías con internet. Trata de la conexión a internet con objetos en lugar de con personas, permitiendo la gestión entre dispositivos y el ambiente para mejorar la calidad de vida de las personas. Estos dispositivos van desde objetos domésticos comunes (heladeras, luminarias, audio), o equipamiento médico hasta herramientas industriales sofisticadas, automóviles y aviones. Hay más de 12 mil millones de dispositivos IoT\* conectados en la actualidad, los expertos esperan que este número aumente a 22 mil millones para 2025. El concepto de internet de las cosas fue propuesto en 1999 por el MIT por investigaciones en el campo de la identificación por radiofrecuencia en red (RFID) y tecnologías de sensores. El autor realizo trabajos similares en la UCA en el año 2004 aplicados a la trazabilidad animal por RFID. Los protocolos en IoT\* proveen los mecanismos para instalar, leer, manipular y borrar las configuraciones de cualquier dispositivo conectado en red. Es aplicable a todos los dispositivos conectados en internet, telefonía celular, IoT\*, etc. En la actualidad no existe un mecanismo unificado para acceder y configurar cualquier dispositivo de IoT en Internet. Hay muchos temas a mejorar, como ser: falta de visibilidad y de integración a la seguridad de las redes (se implementan dispositivos IoT\* sin darles la identificación adecuada al resto de los sectores de IT, lo que pueden generar distintos problemas tanto de operación como de seguridad), uso de protocolos abiertos que no están totalmente probados, inyección en la red de grandes volúmenes de datos sin la correcta administración o equipamiento

para procesar la cantidad de datos, escasas pruebas de concepto (POC) y testeos, vulnerabilidad de las APIs. Los protocolos existentes, como SNMP, están orientados a darnos información sobre lo que ocurre en cada equipo, pero no permiten la configuración de los mismos. Tampoco existen repositorios extendidos de las configuraciones ni de su historial. El proceso de implementación de IoT es masivo, pero al mismo tiempo incipiente en cuanto a la infraestructura y metodologías a utilizar para la minería de los datos recogidos en tiempo real. La propuesta de utilizar técnicas de aprendizaje automático de las configuraciones para una adaptación mejorada de las mismas es aún novedosa y no hemos detectado muchos desarrollos en esta línea.

El proceso de automatización para la captura y agrupación (clustering) de los datos de los dispositivos IoT puede usar técnicas de aprendizaje supervisado y no supervisado profundo. El aprendizaje profundo consiste en una serie de técnicas que permiten a los sistemas informáticos predecir, clasificar, ordenar, tomar decisiones y, en general, extraer conocimientos de los datos sin necesidad de definir explícitamente las reglas para realizar esas tareas. Para este aprendizaje de datos de los sensores se puede utilizar K-NN (K-Nearest-Neighbor) que es un algoritmo de aprendizaje del tipo supervisado. Puede usarse para clasificar muestras de datos (valores discretos) o para predecir (regresión, valores continuos). También puede usarse K-means que es un algoritmo de clasificación no supervisado (por cluster) que agrupa objetos en k grupos basándose en sus características. Este agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo. Ambos algoritmos son muy usados en Aprendizaje de Máquina (Machine Learning). Se pueden usar los resultados del aprendizaje para un modelado en 3D del ambiente usando PolyGen o PyTorch para poder establecer el estado localizado de cada parámetro. Se complementa con la utilización de protocolos tipo SNMP/MQTT para la gestión de las configuraciones.

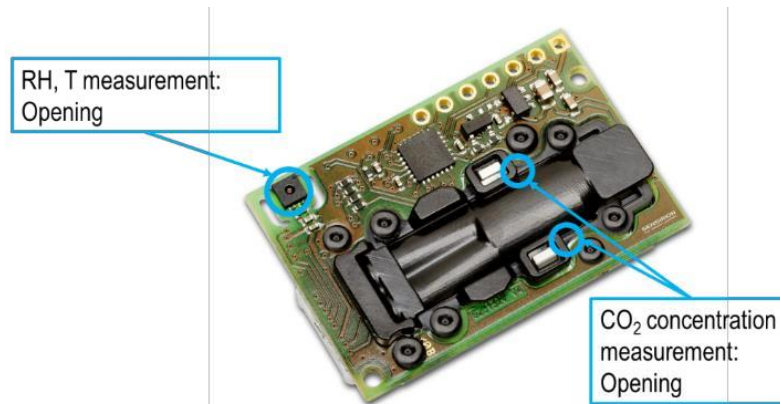
Del lado de los dispositivos IoT (clientes sin control de estado) la variedad de opciones de conexión es tan grande que existen múltiples equipos que se adaptan en menor o mayor medida a cada situación de acuerdo con la ubicación geográfica del dispositivo. Encontramos así ruteadores, switches, Access Points, módems, demarcadores de servicio, REM-OOB (Out Of Band managers), etc. En dispositivos de despliegue masivo como los de IoT, para gestionarlos es necesario combinarlos con otros ecosistemas y protocolos como MQTT, CoAP, y modelos clientes servidor sin control de estado sobre HTTP/HTTPs (Rest API). A nivel comercial servicios en la nube como AzureIoT o AWS IoT son otras opciones para la gestión masiva de dispositivos. MQTT son las siglas de Message Queuing Telemetry Transport. Se trata de un protocolo de mensajería ligero para usar en casos de clientes que requieran un consumo energético mínimo, que están conectados a redes no fiables o con recursos limitados en cuanto al ancho de banda. Se utiliza principalmente para comunicaciones de máquina a máquina (M2M) o conexiones del tipo de Internet de las cosas. MQTT fue estandarizado como código abierto por medio de la organización para el avance de estándares de información estructurada (OASIS) en 2013. OASIS aún gestiona el estándar MQTT [28]. REST es un concepto para la transferencia de estados de representación de datos que busca aumentar la capacidad de HTTP para operar sobre datos en forma escalable.

## **Objetivos, Metodología e Hipótesis de la Investigación**

- Investigar las mejores prácticas de aprendizaje automático para analizar los procesos de configuración, monitoreo de dispositivos y captura de datos ambientales de los dispositivos IoT interconectados por internet por medio de protocolos como MQTT y SNMP con arquitectura REST. Para obtener datos se realizará la captura y análisis de parámetros de ambiente (CO2,

humedad, temperatura) donde interactúen gran cantidad de personas, a los efectos de evaluar la seguridad de ese ambiente frente a la existencia y propagación de elementos patógenos.

- Realizar minería de datos de las redes de sensores operativas utilizando el aprendizaje automático de los parámetros seleccionados. El aprendizaje automático se considera un campo de inteligencia artificial que utiliza técnicas estadísticas para que los sistemas informáticos puedan "aprender" (por ejemplo, mejorar progresivamente el rendimiento en una tarea específica) a partir de los datos [26]. Para que el proceso de aprendizaje sea autosostenido, necesitamos utilizar técnicas que puedan procesar la información de los enlaces de la red con una mínima intervención humana externa. Para evaluar y obtener los datos necesarios para este modelo se adoptó el algoritmo K-NN de los vecinos más próximos (k-nearest neighbors). El objetivo es encontrar los grupos de datos disponibles de los enlaces operativos [25]. Los datos se presentan para su evaluación como una agrupación suave (los grupos pueden superponerse). Uno de los beneficios tanto de K-NN y como de GMM es que producen parámetros válidos para la selección de grupos en cada iteración [28]. Este trabajo es una variante de la primera versión sobre el tema presentada en CACIDI 2018 [15], aplicado a los datos capturados por sensores IoT de CO<sub>2</sub>, Temperatura y Humedad relativa. Otra variante será utilizando el algoritmo K-Means para aprendizaje no supervisado.
- Con la información obtenida proponer acciones sobre otros dispositivos IoT para regular la ventilación del lugar, la capacidad operativa y de prevención contra la propagación de los agentes patógenos, etc. Para demostrar los resultados del trabajo es necesario realizar capturas en entornos reales y hostiles por lo que se propone usar sensores industriales de medio ambiente para la captura de esos datos (CO<sub>2</sub>, Humedad, Temperatura, etc.), por medio de API REST, en conjunto con SNMP/Netconf para el monitoreo/configuración de los dispositivos y la arquitectura REST con MQTT con modelos cliente servidor sin control de estado para las operaciones sobre los datos. Los sensores registrarán las condiciones ambientales en espacios cerrados utilizados por personas (por ejemplo aulas, oficinas, hospitales, restaurants, cines).
- Desarrollar un prototipo funcional basado en Python, SQLAlchemy, MariaDB remoto, que implemente la utilización de aprendizaje supervisado y no supervisado, y SNMP/MQTT/REST para el monitoreo, transporte de comandos de configuración y captura de datos ambientales de sensores IoT que, en base a la información obtenida, tome decisiones que impacten en el entorno. Con la información obtenida ejecutar acciones para regular la ventilación del lugar, la capacidad operativa, y de prevención para evitar la propagación de los agentes patógenos.
- Como plataforma de desarrollo e implementación a nivel de hardware se trabajará con:
  - Sensores IoT SCD30 que consisten en una cavidad óptica con fuente de luz, detectores para medir la concentración de gas CO<sub>2</sub>, un sensor SHT31 para medir humedad y temperatura, montados sobre un PCB con la electrónica necesaria para realizar las lecturas y el procesamiento básico de datos. Las moléculas de CO<sub>2</sub> pasan de abajo hacia arriba por un difusor dentro de la cavidad óptica.



- El sensor se conecta a las entradas/salidas I2C de una microcomputadora Raspberry Py. El conjunto IoT se ensambla dentro de un gabinete protegido para luego ser instalado en el ambiente a monitorear.
- El conjunto sensor IoT/Raspberry Py se puede conectar a Internet de varias formas:
  - Por 4G/LTE utilizando una placa RBM33G de Mikrotik con módulos LTE Quectel EC25.
  - Por fibra óptica utilizando un demarcador de servicio y un conversor. El demarcador utiliza el chipset Microsemi VSC5619EV y módulos SFPs.
  - Directamente desde la microcomputadora Raspberry PI a un moden WiFi., ruteadores y conmutadores (switches) de uso masivo y.
- A nivel de software de base se utilizarán sistemas operativos abiertos basados en Linux, OpenWRT, Python, MicroPython y herramientas para monitoreo y captura de datos como RFC2544, Wireshark, Ostinato, MaríaDB. El proceso de automatización para la captura y agrupación (clustering) de los datos estará basado en técnicas de aprendizaje supervisado y no supervisado profundo.
- Para la interconexión, en los equipos núcleo de las redes en Internet (operadores), utilizaremos los protocolos SNMP/Netconf (Network Configuration Protocol) [11] para monitorear, instalar, manipular y eliminar configuraciones de dispositivos. Se utilizarán llamadas a procedimientos remotos RPC (Remote Procedure Calls), JSON y XML (Extensible Markup Language) para la codificación de los datos y los mensajes del protocolo.
- En base a lo anterior realizar pruebas de concepto (POC) para:
  - Monitoreo, configuración y recolección de datos de operación de dispositivos IoT conectados a puertos I/O digitales en módulos Raspberry basados en Linux Raspbian, los cuales se conectan a switches de borde Ethernet o Carrier Ethernet sobre enlaces físicos con fibra óptica y cobre, y con respaldo sobre 4G/LTE, en el laboratorio descrito en este documento.
  - El sensado de parámetros ambientales utilizando Python, SQLAlchemy y el sensor SCD\_30 Adafruit, con prototipo de extracción de datos por medio de aprendizaje profundo

- Configurar un gestor SNMP/MQTT para monitoreo y gestión de parámetros de la red de dispositivos IoT.
- El método por seguir toma como base la línea de investigación aplicada a un modelo cliente-servidores sin control de estado por medio Rest API. En teoría las herramientas automatizadas de aprendizaje profundo generan una gran fracción de reglas de extracción que de otro modo se escribirían manualmente, dejando al usuario con el poder de definir algunas reglas de extracción que no se pueden aprender o que necesitan un ajuste manual [7]. Aprendizaje supervisado y no supervisado con K-NN y K-Means.

## **Informe de actividades.**

### **Etapas 1 (Marzo 2022 – Agosto 2022)**

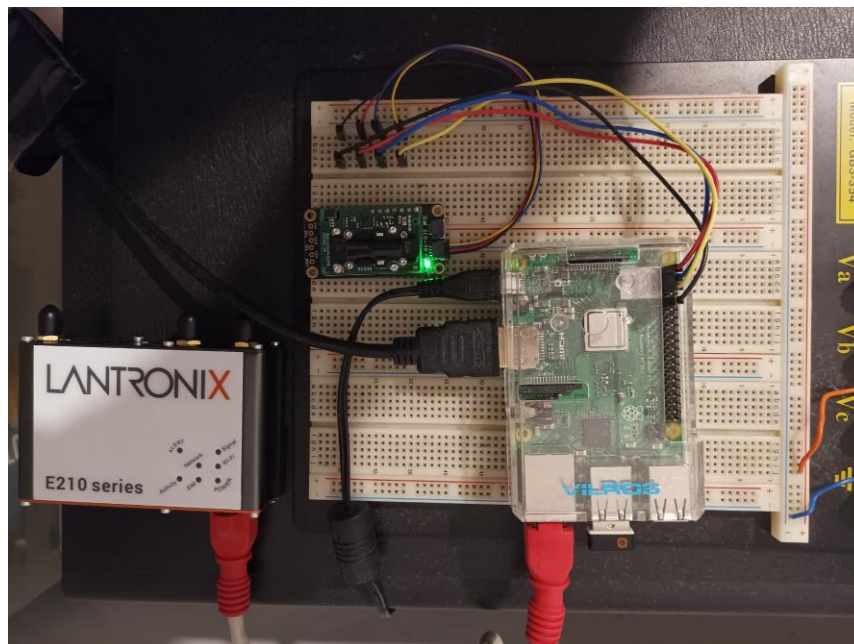
- Se realizó la revisión de antecedentes y trabajos relacionados. La mayoría de los trabajos están aplicados a temas de seguridad. Algunos con temas generales a continuación:
  - M. Mohammadi, A. Al-Fuqaha, S. Sorour and M. Guizani, "Deep Learning for IoT Big Data and Streaming Analytics: A Survey," in IEEE Communications Surveys & Tutorials, vol. 20, no. 4, pp. 2923-2960, Fourthquarter 2018, doi: 10.1109/COMST.2018.2844341.
  - Safa Ben Atitallah, Maha Driss, Wadii Boulila, Henda Ben Ghézala, Leveraging Deep Learning and IoT big data analytics to support the smart cities development: Review and future directions, Computer Science Review, Volume 38, 2020, 100303, ISSN 1574-0137.
  - S. Yao et al., "Deep Learning for the Internet of Things," in Computer, vol. 51, no. 5, pp. 32-41, May 2018, doi: 10.1109/MC.2018.2381131.
- Revisión e investigación de las normas vigentes aplicables a este proyecto.
  - Marco arquitectónico: el enfoque de IEEE P2413-2019 es desarrollar un estándar para el marco arquitectónico para Internet de las cosas, que incluye descripciones de varios dominios de IoT, definiciones de abstracciones de dominios de IoT e identificación de puntos en común entre diferentes dominios de IoT. El marco arquitectónico definido en este estándar promoverá la interacción entre dominios, la interoperabilidad del sistema de ayuda y la compatibilidad funcional.
  - Armonización y seguridad de IoT: El IEEE 1451-99 se centra en el desarrollo de un estándar para la armonización de dispositivos y sistemas de Internet de las cosas (IoT). Este estándar define un método para el intercambio de datos, la interoperabilidad y la seguridad de los mensajes a través de una red, donde los sensores, actuadores y otros dispositivos pueden interoperar, independientemente de la tecnología de comunicación subyacente.
  - Rendimiento y calidad de los sensores: los sensores son fundamentales para el ecosistema de IoT con un gran volumen de diferentes sensores integrados en un marco complejo. IEEE 2700 propone un marco común para la terminología, las unidades, las condiciones y los límites de la especificación del rendimiento del sensor. IEEE P2510 define medidas de calidad, controles, parámetros y definiciones para datos de sensores relacionados con implementaciones de Internet de las cosas (IoT).
  - Serie IEEE Std 802.3 en Ethernet
  - Serie IEEE Std 802.11 en LAN inalámbrica
  - Serie IEEE Std 802.15 sobre redes de área personal inalámbricas
  - Serie IEEE Std 802.16 sobre mejoras de movilidad de acceso inalámbrico de banda ancha
  - Serie IEEE Std 1451, direccionamiento de sensores (adoptado por ISO/IEC)
  - Serie IEEE Std 1547 sobre el manejo de recursos distribuidos en sistemas de energía eléctrica
  - Serie IEEE Std 1609 sobre transporte inteligente
  - Serie IEEE Std 1888, que aborda las ubicuas redes de control de la comunidad verde
  - IEEE 802.15.4 es un estándar técnico que define el funcionamiento de una red de área personal inalámbrica de baja velocidad (LR-WPAN). Especifica la capa física y el control de acceso a los medios para LR-WPAN y es mantenido por el grupo de trabajo IEEE 802.15, que definió el estándar en 2003.[1] Es la base para las

- especificaciones Zigbee,[2] ISA100.11a,[3] WirelessHART, MiWi, 6LoWPAN, Thread y SNAP, cada una de las cuales amplía aún más el estándar al desarrollar las capas superiores que no están definidas en IEEE 802.15.4 . En particular, 6LoWPAN define un enlace para la versión IPv6 del Protocolo de Internet (IP) sobre WPAN, y es utilizado por capas superiores como Thread.
- ISO/IEC 30165: Internet de las cosas (IoT): el marco de IoT en tiempo real proporciona una guía para implementar un sistema RT-IoT para evitar las dificultades que suelen ocurrir durante el desarrollo de sistemas en tiempo real.
  - Protocolos y estándares comunes: HTTP (Protocolo de transferencia de hipertexto), LoRaWan (Red de área amplia de largo alcance), Bluetooth. Malla, NFC (Comunicación de campo cercano), Wi-Fi, Transporte de telemetría de cola de mensajes (MQTT), Protocolo de aplicación restringida (CoAP), Protocolo de cola de mensajes avanzado (AMQP), Protocolo de comunicación de máquina a máquina (M2M), Mensajería extensible y Protocolo de presencia (XMPP), Servicio de distribución de datos (DDS), Z-Wave.
  - MQTT Versión 5.0 Estándar OASIS. 07 marzo 2019.
- Selección y ensayo de herramientas de software JetBrains PyCharm, NumPy, SQLAlchemy, Flask, MariaDB.
    - Para la operación de los sensores IoT, la captura de datos, la geolocalización, la tunelización de tráfico en forma segura y el almacenamiento de datos en SQL se utiliza el lenguaje Python 3.7 con las librerías correspondientes.
    - Para la presentación en el navegador se utiliza Javascript. La lectura de datos se realiza en el servidor utilizando Node JS.
    - Para repositorio y control de versiones se utiliza GitHub.
    - IDE para desarrollos: Visual Studio Code, PyCharm, WLS, Thorny (Raspberry).
  - Configuración de servidor en hosting remoto para recibir la captura de los datos ambientales.
    - Servidor con Shared Hosting, Hosting Package profesional, Server Name server153, cPanelVersion 102.0 (build 26), Apache Version 2.4.55, PHP Version 5.6.40, MySQL Version 10.3.37-MariaDB-log-cll-lve
    - Architecture x86\_64, Operating System Linux.
  - Configuración de servidor en hosting remoto para recibir la captura de los datos ambientales.
    - Se configuró el servidor para acceder a la base de datos en MySQL en forma segura por medio del tunelizado con SSHTunnel desde Python con SQLAlchemy como librería de funciones SQL.
  - Preparación de los entornos virtuales de redes venv sobre Python para facilitar la replicación de datos.
    - Se prepararon entornos de Python “virtualenv” parcialmente aislados que permiten instalar paquetes para que los use cada aplicación en particular, en lugar de instalarlos en todo el sistema.
  - Desarrollo de la librería REST API con Flask y SQL Alchemy con las rutas de acceso a las funciones CRUD (Create-Read-Update-Delete) para el manejo de los datos de los sensores utilizando SQL, con almacenamiento remoto en una base MariaDB.
    - Se desarrollaron módulos para la captura de los datos desde sensores de ambiente con arquitectura IoT y se realizaron pruebas para el almacenamiento masivo de los datos.

### **Etapas 1 (Septiembre 2022 – Febrero 2023)**

- Preparación del laboratorio . Como plataforma de desarrollo e implementación a nivel de hardware se trabajará con Sensores IoT SCD30 Sensirion (Suiza) y ruteador IoT Lantronix E210 para la conectividad a internet por medio de 4G/LTE/5G o WiFi o WAN. Estos equipos fueron adquiridos con fondos propios del investigador.

- Los sensores SCD30 realizan la medición de CO2 por el principio de NDIR de 2 mediciones comparadas. Estabilidad de medición a largo plazo por medio de canales de referencia incorporados. Medición compensada de temperatura de la señal del sensor de CO2. Sensores de humedad y temperatura integrados. El sensor utiliza tecnología CMOSens® de Sensirion. Interfaz calibrada y linealizada UART e I2C.
- El ruteador E210 posee la capacidad de conexión por LTE Cat-1 M1 Celular / WAN con cable / LAN / WiFi con conmutación por fallas y balanceo de carga. Se puede hacer la conversión de protocolo de bus de campo: Modbus, DNP3, IEC 104, DLMS, Dual Sim. Se puede hacer programación y procesamiento a nivel de equipos de borde de la red. Se implementa esquema de tunelización VPN con esquemas de conmutación por error hacia la red celular/WAN/Wi-Fi. La conmutación por errores es automática para garantizar la conectividad a Internet.
- El sensor se conecta a una Raspberry Pi con el sistema operativo Linux Raspbian por el bus I2C integrado. Una versión similar del PCB Raspberry Pi fue testeado en ambientes agresivos dentro de un gabinete NEMA IP67 con temperaturas extremas y polvo durante 3 años sin verse afectado su funcionamiento.

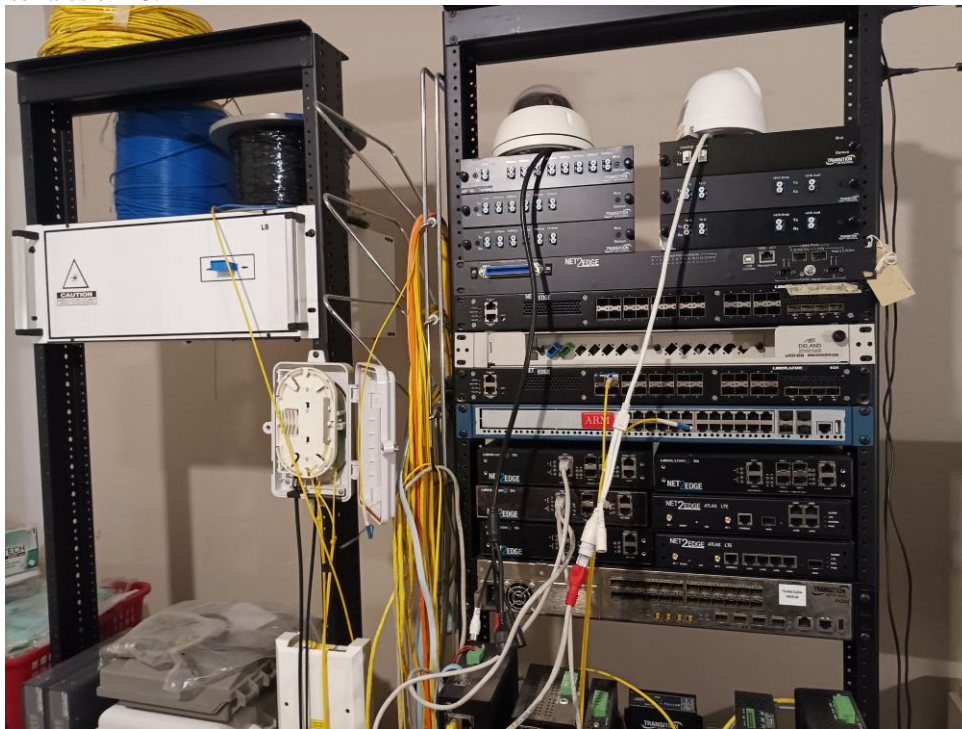


- Desarrollo del prototipo funcional de acuerdo con lo descrito en “Metodología”
  - El código desarrollado y detalle de los prototipos está disponible en: <https://github.com/jaouret/IoT01>.
  - El acceso es privado. Enviar email a [javierouret@uca.edu.ar](mailto:javierouret@uca.edu.ar) para recibir permiso de lectura.
- Selección y ensayo de las herramientas de inteligencia artificial para aprendizaje profundo.
- Ensayo sobre las configuraciones cargadas en las Raspberry Pi.
  - Se realizaron ensayos de capturas con las configuraciones y código durante 2 meses en forma satisfactoria. Todo el desarrollo pueden ejecutarse en ambientes Windows, Linux, MacOS o plataformas móviles.
- Escribir y presentar un paper sobre lo anteriormente expuesto.
  - Paper en desarrollo para ser enviado a las conferencias y publicaciones.



## Informe del laboratorio utilizado

- El laboratorio utilizado para las investigación y desarrollo de este trabajo consta de los siguientes elementos:
  - Switch Carrier Ethernet 2.0, 1 x OLT GPON, 10 x ONT/ONU GPON, 2 x SFP ONU, 2 x SFP OTDR, 4 x SFP 1G Ethernet, bobinas de impulso de fibra óptica, splitters 1x4 + 1x8, instrumental de medición (OTDR , OLS, OPM). NID Atlas LTE, NID N2E 304/306. Liberator 4424.
  - Laptops con Windows 10, Debian Linux, Ubuntu Linux y Windows Linux Subsystem.
  - Software para generación y análisis de tráfico (Wireshark, Ostinato, RFC 2544, Y.1564)
  - Software para desarrollo IDE PyCharm , Visual Studio Code, plataformas de CVS Gitlab, Python, SQLAlchemy, hosting remoto con MySQL. Raspeberry PI 3 con Raspbian OS, MicroPython y servidor MySQL.
  - Sensor IC2 con NIR / HVAC para la captura de parámetros ambientales. Router E210 para completar la configuración de los sensores IoT con acceso redundante a internet por LTE/4G/5G/WiFi/WAN con puertos seriales e I2C.



## Congresos a los que se enviará el paper en 2023

- *Nota: seguramente se agreguen otros congresos durante el año en curso.*
- NPLAY 2023. Marzo de 2023. Córdoba Argentina.
- International Conference On Intelligent Computing And Control Systems ICICCS. Julio 2023.
- 53 JAIIO. Septiembre de 2023 Sede de Posgrado de UNTREF, CABA
  - Fecha límite para presentación de trabajos: Abril de 2023.

## Publicaciones a la cuales se estará enviando el paper en el 2023

- *IEEE Journal on Selected Areas in Communication.*
- *ACM Transactions on Internet Technology.*
- *IEEE Internet of Thing Journal.*

## Bibliografía

- [1] Changzhi, D., Zhang, H., Arens, E., & Zhiwei, L. (Marzo de 2017). Machine learning approaches to predict thermal demands using skin temperatures: Steady-state conditions. *Building and Environment*, Volume 114, págs. 1-10.
- [2] Giardina, P., Sambo, N., Dallaglio, M., Bernini, G., Carrozzo, G., Cugini, F., & Castoldi, P. (Marzo de 2017). Configuring monitoring entities through NETCONF and YANG in control and hierarchical management planes. *Optical Fiber Communication Conference 2017*. Los Ángeles, California, Estados Unidos: OSA
- [3] IEEE. (2013). Are we ready for SDN? Implementation challenges for software-defined networks. *IEEE Communications Magazine*.
- [4] Shafiq, M., Yu, X., & Wang, D. (2017). Network Traffic Classification Using Machine Learning Algorithms. *Advances in Intelligent Systems and Interactive Applications*, págs. 621-627.
- [5] Syahputra, R. (2017). Distribution Network Optimization Based on Genetic Algorithm. *Journal of Electrical Technology UMY*.
- [6] Zander, S., & Schmoll, C. (05 de Junio de 2018). Network Measurement and Accounting Meter (NETMATE). Obtenido de <https://github.com/DanielArndt/netmate-flowcalc>
- [7] Jussi Myllymaki, Jared Jackson. Robust Web Data Extraction with XML Path Expressions. IBM Research Division. 2002.
- [8] Jamuna .A, Vinodh Edwards S.E.(2013). Efficient Flow based Network Traffic Classification using Machine Learning. *International Journal of Engineering Research and Applications (IJERA)*
- [9] T.Kalaiselvi1, P.Shanmugaraja.(2016). Internet Traffic Classification Using supervised Learning Algorithms – A Survey. *International Research Journal of Engineering and Technology*.
- [10] RFC 6241. Network Configuration Protocol (NETCONF). (2011). Internet Engineering Task Force (IETF)
- [11] RFC 1157. A Simple Network Management Protocol (SNMP). (1990). Network Working Group.
- [12] Javier Ouret. (2018). Modelización Normalizada de la Calidad de Servicio en Redes de Acceso ME y GPON.
- [13] Javier Ouret. (2019) . Reconfiguración Automática de Parámetros de Calidad de Servicio Dispositivos por medio del Protocolo NETCONF. 48 JAIIO. Salta. Argentina.
- [14] RFC for NETCONF.
- [15] Disponible: <https://www.yumaworks.com/tools/yang-compiler/netconf-details>
- [16] RFC 8040. RESTCONF Protocol. Internet Engineering Task Force (IETF). 2017.
- [17] Netopeer2 – The NETCONF Toolset.
- [18] Disponible: <https://github.com/CESNET/Netopeer2>
- [19] NetworkX
- [20] Disponible: <https://networkx.github.io/documentation/stable/tutorial.html>
- [21] TensorFlow – Open-source software library for dataflow and differentiable programming across a range of tasks. Disponible: <https://www.tensorflow.org/tutorials/>
- [22] MOTT Disponible; <https://mqtt.org/>
- [23] Raspberry PI. Disponible: <https://www.raspberrypi.org/forums/viewtopic.php?t=196010>
- [24] Sysrepo Docker. Disponible en: <https://hub.docker.com/r/sysrepo/sysrepo-netopeer2>
- [25] Dempster, A.P.; Laird, N.M.; Rubin, D.B. "Maximum Likelihood from Incomplete Data via the EM Algorithm". *Journal of the Royal Statistical Society, Series B*. 39, 1977, pp 1–38.
- [26] Samuel, Arthur. "Some Studies in Machine Learning UsiDisponibleng the Game of Checkers", *IBM Journal of Research and Development*, 1959.
- [27] Padhraic Smyth, "Mixture Models and the EM Algorithm", Department of Computer Science, University of California, Irvine, 2017
- [28] Disponible: <https://www.paessler.com/es/it-explained/mqtt>
- [29] Disponible: <https://cen.acs.org/biological-chemistry/infectious-disease/es-El-experto-en-aerosoles-JosLuis/99/i1>
- [30] Disponible: <https://www.cerem.es/blog/porque-el-internet-de-las-cosas-iot-promete-cambiar-la-vida-del-ser-humano>
- [31] T. Teleszewski1 · K. Gładyszewska-Fiedoruk1. "The concentration of carbon dioxide in conference rooms: a simplified model and experimental verification". Published online: 27 May 2019.
- [32] Measuring Carbon Dioxide Inside Buildings. Disponible: <https://www.energy.wsu.edu/>
- [33] Jordi Salazar y Santiago Silvestre. "Internet de las cosas". Publicado por: České vysoké učení technické v Praze. Fakulta elektrotechnická Dirección de contacto: Technická 2, Praha 6, Czech Rep.
- [34] Disponible: <https://www.kellton.com/kellton-tech-blog/your-complete-guide-to-iot-protocols-and-standards-2022>
- [35] M. A. Al-Garadi, A. Mohamed, A. K. Al-Ali, X. Du, I. Ali and M. Guizani, "A Survey of Machine and Deep Learning Methods for Internet of Things (IoT) Security," in *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1646-1685, thirdquarter 2020, doi: 10.1109/COMST.2020.2988293.