

## DESARROLLO DE UNA SOLUCIÓN DE IoT Y BIG DATA PARA UN FRIGIDER INTELIGENTE

Juan Antonio Adolfo Zaffaroni<sup>1</sup>, Ramiro Walter Garbarini<sup>2</sup>, Yasnalla Rivero Peña<sup>3</sup>

<sup>1</sup>Universidad Tecnológica Nacional, Argentina y jzaffaroni@dblandit.com, <sup>2</sup>Universidad Tecnológica Nacional, Argentina y rgarbarini@frba.utn.edu.ar, <sup>3</sup>Universidad de Holguín, Cuba y yasnalla@uho.edu.cu

### RESUMEN

Hace años que viene creciendo en el mundo el concepto de Internet de las cosas, que significa la posibilidad de conectar cualquier dispositivo electrónico y/o electrodoméstico a internet para poder ser utilizado, configurado y controlado de forma remota. Además, se dota a estos dispositivos de distintos sensores que generan datos a gran velocidad y volumen, los cuáles deben ser almacenados en plataformas de Big Data, y posteriormente analizados para identificar patrones, encontrar correlaciones entre variables o realizar predicciones en función al comportamiento. En este contexto, una empresa embotelladora de gaseosas y cervezas que entregan en concepto de publicidad frigidiers a negocios de venta de bebidas y alimentos, se encuentra ante la dificultad de obtener información en tiempo real del estado de estos y del control de la cantidad y tipo de mercadería que contienen. Para ello se presenta esta solución que combina un frigider tradicional con sensores de temperatura y humedad, sensor abre puerta, cámara fotográfica para reconocimiento de imágenes, sensores de consumo y tensión, conexión WIFI y 3G en una primera etapa. Incluye además, el desarrollo de una aplicación web/móvil responsiva que permite acceder a diferentes alertas y a la actualización en línea del control de stock de los productos que contenga el frigider utilizando técnicas de procesamiento de imágenes y algoritmos de machine learning. Se utilizó un método de valor de umbral adaptativo y transformaciones morfológicas, el algoritmo de obtención de características SURF para la identificación y contabilización de botellas y un algoritmo de aprendizaje supervisado del estilo SVM.

Palabras clave: Internet de las cosas; Big Data; frigidiers inteligentes; machine learning; reconocimiento de imágenes

### ABSTRACT

The concept of the Internet of Things has been growing in the world for years, which means the possibility of connecting any electronic device and / or household appliance to the internet in order to be used, configured and controlled remotely. In addition, these devices are equipped with different sensors that generate data at high speed and volume, which must be stored on Big Data platforms, and later analyzed to identify patterns, find correlations between variables or make predictions based on behavior. In this context, a bottling company for soft drinks and beers that deliver frigidiers advertising to businesses selling drinks and food, is faced with the difficulty of obtaining real-time information on the status of their and the control of quantity and type of merchandise they contain. This is the solution that combines a traditional frigider with temperature and humidity sensors, door opener, camera for image recognition, consumption and voltage sensors, WIFI and 3G connection in a first stage. It also includes the development of a responsive web / mobile application that allows access to different alerts and the online update of the stock control of the products contained in the frigider using image processing techniques and machine learning algorithms. We used a method of adaptive threshold value and morphological transformations, the algorithm of obtaining SURF characteristics for the identification and accounting of bottles and a supervised learning algorithm of the SVM style.

Keywords: Internet of things; Big Data; smart fridges; machine learning; image recognition.

### 1. INTRODUCCIÓN

A partir de las investigaciones y los descubrimientos de Laplace, Poisson, Faraday, Gauss y Maxwell, el cual predijo la propagación de señales eléctricas a través del espacio, en el año 1900 Guglielmo Marconi transmitió señales de forma inalámbrica a través del océano Atlántico. De esa época en adelante, se desarrollaron diferentes tecnologías de radiofrecuencia las cuales, hoy en día, se pueden integrar en un sólo chip posibilitando la creación de dispositivos muy pequeños capaces de enviar información de forma inalámbrica ya sea para controlar otros dispositivos o para simplemente comunicar la ubicación (como en el caso de los GPS).

En el año 1999 surgieron por primera vez los conceptos de Big Data e Internet de las cosas. El primer uso del término Big Data fue en un trabajo académico: "Visually Exploring Gigabyte Datasets in Realtime (ACM)" y el primer uso del concepto "IoT - Internet of Things" (Internet de las Cosas) fue en una presentación de negocios de Kevin Ashton para Procter and Gamble. A partir de ello, en estos últimos casi veinte años, la evolución de plataformas y tecnologías para administrar Big Data y la implementación de IoT ha crecido enormemente. Es importante entender que estos dos conceptos deben ir de la mano, debido a que la cantidad de datos generados por sensores o dispositivos IoT son generados a tanta velocidad y volumen que se requiere para su almacenamiento y posterior análisis de tecnologías orientadas a Big Data que permitan almacenar la información en conjuntos de servidores que trabajan sincronizados como si fuesen una sola unidad de procesamiento y almacenamiento. Para ello, es fundamental el uso de Bases de Datos NoSQL y Plataformas de procesamiento y almacenamiento distribuido. Las empresas comenzaron a aplicar las tecnologías de IoT y Big Data para poder obtener información 360 de los usuarios.

Así, por ejemplo, surgió Nike+: un chip que se pone en los zapatos deportivos y permite sincronizarlo con una aplicación móvil que está conectada a una aplicación web del proveedor en la que se registra información de cómo utiliza el producto el usuario. Por ejemplo: si es un corredor, si es una persona sedentaria o si realiza actividad deportiva esporádicamente. Otras empresas que utilizan estos conceptos son BGH o Carrier, ofreciendo a sus clientes de aires acondicionados conectar sus equipos de aire acondicionado a internet y permitirles controlarlos a través de aplicaciones móviles, obteniendo a cambio información acerca de cómo usan sus dispositivos los usuarios, permitiendo a su equipo de diseño de producto mejorarlos día a día en base a los gustos de los usuarios.

En este contexto, una empresa embotelladora de gaseosas y cervezas que entregan en concepto de publicidad frigidiers a negocios de venta de bebidas y alimentos, se encuentra ante la dificultad de obtener información en tiempo real del estado de sus frigidiers y del control de la cantidad y tipo de mercadería que contienen. Con el objetivo de simplificar lo previamente mencionado, se presenta esta solución que combina un frigidier tradicional con sensores de temperatura y humedad, sensor abre puerta, cámara fotográfica para reconocimiento de imágenes, sensores de consumo y tensión, conexión WIFI y 3G en una primera etapa. (Misra, D. K.) (Rogers, J. W. M., Plett C.)

## 2. METODOLOGÍA

Para la definición del hardware del sistema embebido a utilizar se plantearon dos kits de desarrollo basados en un microcontrolador ARM Cortex-A:

Según datos en la Tabla 1, la Beaglebone Black (*Beaglebone Black - System Reference Manual*) satisface las necesidades de procesamiento requeridas para la solución pero carece de conectividad integrada con una cámara. Asimismo, para proveer escalamiento de componentes (GPS, 3G, sensor de potencia, entre otros), es conveniente tener más de 1 puerto USB (Host) disponible para lo cual se optó por utilizar la Raspberry Pi 3 (*Datasheet Raspberry Pi Compute Module 3*).

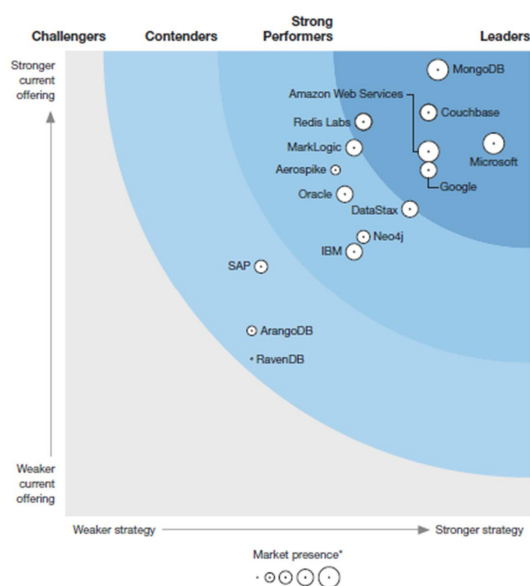
**Tabla 1: Comparación entre opciones de plataforma embebida**

Concepto	Raspberry Pi 3	Beaglebone Black
Procesador	Broadcom BCM2837	AM3358/9
Núcleo CPU	4x Cortex-A7	Cortex-A8 Dual PRU (200MHz)
Frecuencia CPU	1.2GHz	1GHz
RAM	1GB	512MB
Memoria FLASH	No	4GB
Almacenamiento	microSD	microSD
Ethernet	100Mbit	100Mbit
USB	4x 2.0 Host	1x 2.0 Host, 1x 2.0 Device
Salida de video	HDMI, compuesto	HDMI
Expansión	40 pines: GPIO,I2C,SPI,UART	2x46 pines: GPIO,ADC,I2C,SPI, UART,CAN
Dimensiones	85x56mm	86x53mm

Se hicieron pruebas de estrés en la recolección de datos desde los sensores conectados, se tuvieron en cuenta manejo de excepciones a lo largo del código del sistema embebido (recolectores de datos, orquestados de procesos, envío de datos a las APIs) así como la posibilidad de reiniciarse a sí mismo si surgiera alguna falla en la ejecución del proceso maestro (orquestador de procesos). También se incluyó, dentro del almacenamiento interno de la Raspberry Pi 3, persistencia de datos de hasta 90 días, con las fotos del interior del frigider y de las métricas asociadas (sensor abrepuerta, sensor temperatura y presión, medidor de potencia, posición geoespacial y estado de botellas).

Para el análisis y la definición de las Bases de Datos se analizaron las mismas a partir de diferentes criterios, escalamiento horizontal, replicación, sharding, consistencia eventual, particionamiento de red, latencia, entre otros. Para la selección de los motores de Base de Datos NoSQL se tomaron en cuenta, además, su posicionamiento en el mercado en comparación con otras Bases de Datos (Figura 1) (*Yuhanna, Noel*) del mismo segmento: MongoDB en el segmento de las Bases de Datos NoSQL Documentales (*Mongodb.org*) y Cassandra en las Bases de Datos Column Family (*Cassandra.apache.org*).

**Big Data NoSQL**  
 Q1 2019



**Figura 1 - Forrester Wave™: Big Data NoSQL, Q1 2019**

Se hicieron pruebas de carga con más de 600.000.000 eventos en varios días y se evaluó la performance de ingesta y de consulta con la herramienta JMeter (*JMeter*). Para los datos de configuración y definición de producto se realizaron consultas por diferentes atributos, las cuáles fueron satisfactorias. Dentro del marco de la detección de objetos, se consideraron distintos métodos de procesamiento de imágenes, comúnmente utilizados. Los ensayos se realizaron utilizando casos de prueba obtenidos con un prototipo de fridger. Se le dió prioridad a la precisión, antes que a la complejidad de la posible solución.

### 3. RESULTADOS Y DISCUSIÓN

El primer desafío es el de definir las diferentes Base de Datos en las que se persisten todos los datos de la plataforma: métricas registradas desde los sensores de cada fridger, configuración de dispositivos y configuración de usuarios. La cantidad de frigidgers a manejar se estima entre 100000 y 150000, distribuidos en diferentes zonas del país, cada uno posee 6 sensores, de los cuáles se tomará información cada 15 segundos, siendo 34.560 inserciones por día por cada fridger, sumando un total de entre 3.456.000.000 y 5.184.000.000 inserciones por día. Para ello se requiere una Base de datos que resuelva la grabación de series de tiempo a gran velocidad y permita distribuir los datos en múltiples nodos de un cluster de servidores sin que eso cause una degradación en la performance. Los requisitos de almacenamiento de los datos de configuración de frigidgers, dispositivos y usuarios varían con respecto a los de las métricas previamente mencionados. En este caso los puntos claves no son la velocidad de grabación o volumen de la información, sino su disponibilidad y estructura variable de los datos a persistir. Se requieren Bases de

Datos que puedan replicarse en varios servers y que permitan rápidamente modificar su esquema y realizar consultas por cualquier atributo. Se hicieron diferentes análisis y benchmarks entre bases de datos Relacionales y Bases de Datos NoSQL (*Abramova, V., Bernardino, J., & Furtado, P.*) (*Nayak, A., Poriya, A., & Poojary, D.*). Para almacenar las series de tiempo se seleccionó Cassandra (*Cassandra.apache.org*), una Base de Datos NoSQL de Tipo Column Family que, mediante el uso de su concepto de Wide Rows, permite por cada Dispositivo y Sensor, tener una serie de tiempo almacenada con los datos censados cada 15 segundos. Para el almacenamiento de la información de frigidiers, dispositivos y usuarios, se optó por utilizar MongoDB (*Mongodb.org*), por ser una Base de Datos NoSQL de Tipo Documental y schemaless, que permite realizar consultas y operaciones por cualquier atributo del dato e implementar escalamiento horizontal de forma nativa permitiendo replicación y distribución geográfica de los datos.

El desarrollo de la API y aplicación web/móvil responsiva que permite acceder a información de los frigidiers como alertas (puerta abierta, baja de temperatura, corte de consumo, cambios de tensión) y actualización en línea del control de stock, se realizó basado en el stack MERN (MongoDB, Express, React y NodeJS). (*Mern.io*) Esta combinación de tecnologías estaban alineadas con varias de las previamente mencionadas, y su velocidad de desarrollo, curva de aprendizaje baja y compatibilidad entre los componentes está ampliamente demostrada en el mercado.

El control de stock, basado en algoritmos de procesamiento de imágenes y machine learning, se realiza en base al reconocimiento de tapas y en dos etapas: Primero, una red neuronal pre-entrenada realiza la detección de bordes (HED). Se optó por este tipo de solución robusta ante cambios en la iluminación. Los bordes permiten la identificación de tapas dentro de la imagen. Se optó por una estrategia de detección de tapas debido al espacio reducido, limitadas posiciones (y ángulos) posibles de la cámara dentro del frigider y la oclusión que provocan las botellas unas a otras. Segundo, las regiones de la imagen que contienen alguna tapa son analizadas por separado utilizando un segundo algoritmo de machine learning, SVM (Support-Vector-Machine). Finalmente, los resultados son enviados mediante la red celular 3G a la API, para ser visualizados desde la aplicación web.

Como objetivos alcanzados con la solución desarrollada se pueden enunciar: detectar la cantidad y caracterizar las botellas presentes, detectar la apertura de puerta y posteriormente identificar el cambio de stock de botellas, medir la potencia monofásica consumida por el frigider y pronosticar, en base al consumo de potencia, fallas en su funcionamiento.

A continuación, en las Figuras 2-4, se muestran algunas pantallas de la solución desarrollada.

Comentado [1]: +cschiffmacher@dblandit.com tenes la foto de los contornos? y de la detección de eclipse (una mas o menos acertada)?

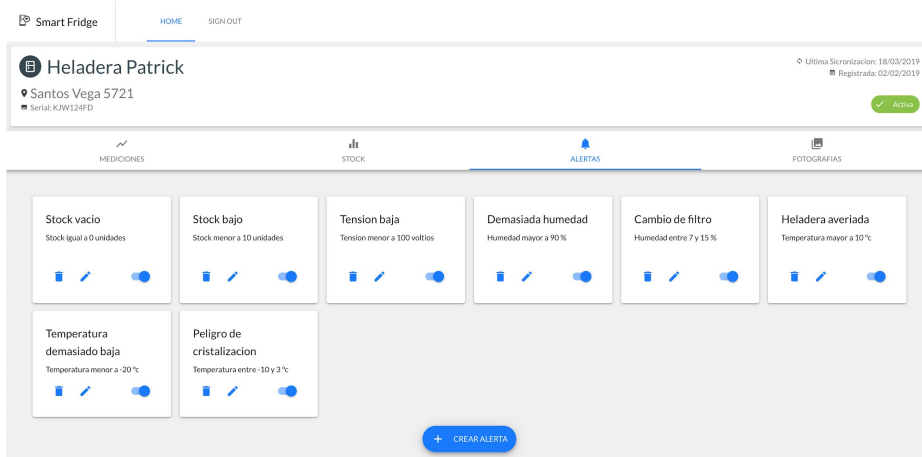


Figura 2. Pantalla de Administración de Alertas.

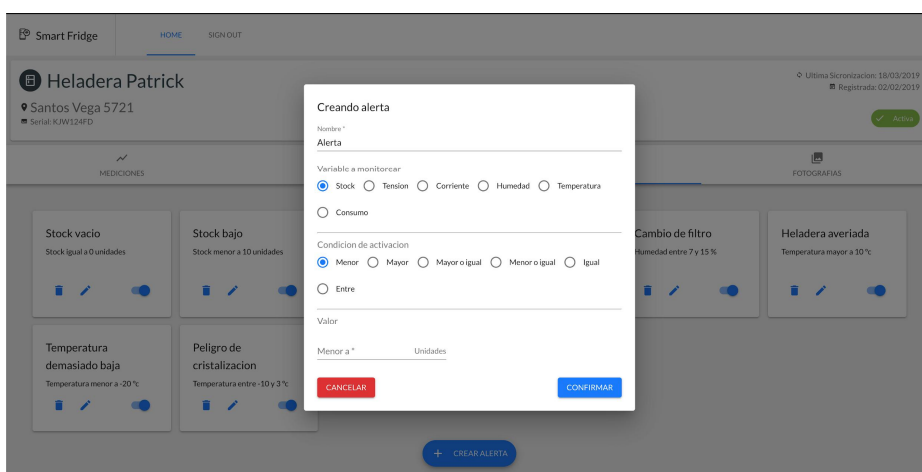


Figura 3. Pantalla de Configuración de Alertas.

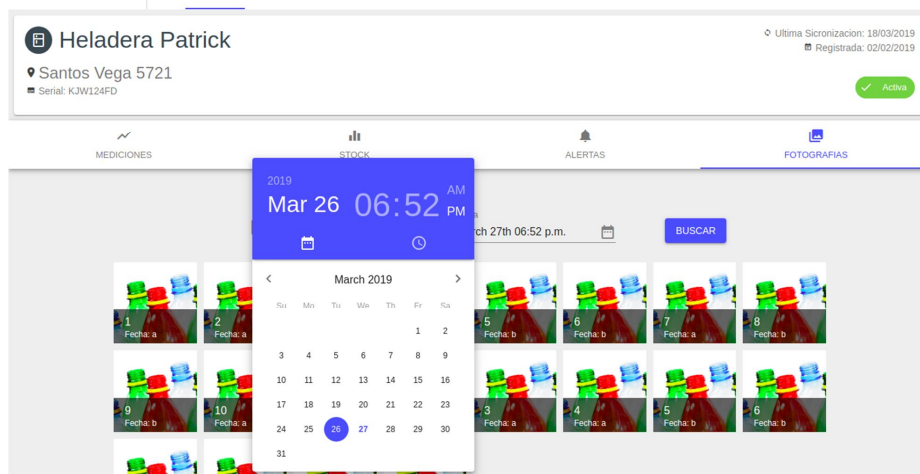


Figura 4. Consulta de Fotografías del Contenido del Frigider.

#### 4. CONCLUSIONES

Con la solución desarrollada apuntamos a resolver el problema que tiene la empresa embotelladora de gaseosas y cervezas de no poder contar con información de cómo utilizan sus frigidiers los comercios en los cuáles se instalan en concepto de promoción de la marca. Esta solución les permitirá contar en tiempo real con imágenes del contenido del frigider y con el estado del stock de botellas que el mismo posee permitiendo, por ejemplo, la reposición eficiente de mercadería. Además, pueden prevenir fallas del motor del frigider comparando su curva de consumo con las curvas de consumo estándar para motores de frigidiers.

En cuanto a la solución pudimos resolver los algoritmos utilizados para identificación de objetos, así como también las bases de datos para almacenar la información de los distintos sensores aproximadamente cada 15 segundos en una base de datos NoSQL column family, Cassandra (*Cassandra.apache.org*), y almacenar la información de configuración y definición de productos en una base de datos NoSQL Documental, MongoDB (*Mongodb.org*) (Zaffaroni, J.; Róttoli, G.D.; López Nocera, M.; Pollo Cattaneo, M.F.). Ambas Bases de Datos están diseñadas para escalar horizontalmente y correr en Cluster de Servidores, permitiendo a la solución escalar acompañando el crecimiento del negocio. (Bugiotti, F., & Cabibbo, L.) (Strauch, C., Sites, U. L. S., & Kriha, W.) (Hecht, R., & Jablonski, S.) (Moniruzzaman, A. B. M., & Hossain, S. A.)

#### 5. CONFLICTO DE INTERESES

Los autores consideran que no existe conflicto de interés con la investigación presentada.

## 6. REFERENCIAS BIBLIOGRÁFICAS

- Abramova, V.; Bernardino, J.; Furtado, P. (2014). *Experimental evaluation of NoSQL databases* (p. 1, 6). International Journal of Database Management Systems
- Arora, R.; Aggarwal, R.R. (2013). *Modeling and Querying Data in MongoDB* (p. 4). International Journal of Scientific and Engineering Research
- Beaglebone Black - System Reference Manual*. En *git hub* [en línea]. [Consulta: 27 de febrero de 2019]. Recuperado de: [https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual#\\_Table\\_2\\_BeagleBone\\_Black\\_Features](https://github.com/beagleboard/beaglebone-black/wiki/System-Reference-Manual#_Table_2_BeagleBone_Black_Features)
- Bugiotti, F.; Cabibbo, L. (2013). *A Comparison of Data Models and APIs of NoSQL Datastores*. Roma: Dipartimento di Ingegneria della Università di Roma
- Datasheet Raspberry Pi Compute Module 3*. En *raspberrypi* [en línea]. [Consulta: 29 de febrero de 2019]. Recuperado de: [https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi\\_DATA\\_CM\\_1p0.pdf](https://www.raspberrypi.org/documentation/hardware/computemodule/datasheets/rpi_DATA_CM_1p0.pdf)
- Hecht, R.; Jablonski, S. (2011). *NoSQL evaluation: A use case oriented survey* (p 336-341). Hong Kong: 2011 International Conference on Cloud and Service Computing
- Jmeter - Aplicación para testing de rendimiento*. En *JMeter* [en línea]. [Consultado: 10 de marzo de 2019]. Recuperado de: <https://jmeter.apache.org/>
- López, D. (2012). *Análisis de las posibilidades de uso de Big Data en las organizaciones*. Santander, España: Universidad de Cantabria
- MERN - Stack de Desarrollo*. En *Mern* [en línea]. [Consultado: 6 de marzo de 2019]. Recuperado de: <https://mern.io>
- Misra, D.K. (2004). *Radio-Frequency and microwave communication circuits* (p. 1-2). Milwaukee, Wisconsin: University of Wisconsin
- Moniruzzaman, A.B.M.; Hossain, S.A. (2013). *Nosql database: New era of databases for big data analytics-classification, characteristics and comparison*
- MongoDB - NoSQL Database*. En *MongoDB* [en línea]. [Consultado: 5 de marzo de 2019]. Recuperado de: <https://www.mongodb.com>
- Nayak, A.; Poriya, A.; Poojary, D. (2013). *Type of NOSQL databases and its comparison with relational databases* (p. 5, 16-19). New York, USA: International Journal of Applied Information Systems
- Rogers, J.W.; Plett C. (2010). *Radio Frequency Integrated Circuit Design* (p. 11-19). Norwood, Massachusetts
- Strauch, C.; Sites, U.L.S.; Kriha, W. (2011). *NoSQL databases. Lecture Notes*. Stuttgart: Stuttgart Media University





*The Apache Cassandra Project* (2016). En *Apache Cassandra* [en línea]. [Consulta: 5 de marzo de 2019]. Recuperado de: <https://cassandra.apache.org/>

Yuhanna, Noel (2019) *The Forrester Wave: Big Data NoSQL Q1 2019* (Reporte Técnico)

Zaffaroni, J.; Róttoli, G.D.; López Nocera, M.; Pollo Cattaneo, M.F. (2016). *Estudio Comparativo de características de versiones de la base de datos NoSQL Documental MongoDB* (Reporte Técnico) [en línea] [Consulta: 2 de marzo de 2016]. Recuperado de: <https://sistemas.frba.utn.edu.ar/grupogemis/Trabajos/Reportes%20Tecnicos/GEMIS-TD-2016-03-TR-2016-03.pdf>

#### **SOBRE LOS AUTORES**

Ing. Juan Zaffaroni, Ingeniero en Sistemas de Información (UTN). Director de la Diplomatura en Arquitecturas de Big Data Aplicadas dictada en la Universidad Tecnológica Nacional – Extensión Universitaria, docente de las materias Gestión de Datos, Diseño de Sistemas e Implementación de Bases de Datos NoSQL en la Carrera Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional en Argentina. CEO de la empresa DBlantIT, consultora en sistemas con foco exclusivo en Big Data. Presentó trabajos en distintos Congresos de Ingeniería.

Ramiro W. Garbarini, Ingeniero en Sistemas de Información (UTN). Jefe de Laboratorios de Sistemas del Departamento de Ingeniería en Sistemas de Información, UTN. Docente Adjunto de "Análisis de Sistemas" y de "Sistemas de Información Geográfica", UTN. Docente Adjunto del área "Teoría en Sistemas", UNNOBA. Vicepresidente del Capítulo DAMA Argentina. Director I+D, Socio en Ecologix SRL – Argentina.

Yasnalla Rivero Peña, Ingeniero en Ciencias Informáticas. Profesor Universitario de la Universidad de Holguín, Cuba. Master en MAIPA - Matemática Aplicada e Informática para la Administración.