

Control de cultivos hortícolas con sensores y actuadores mediante una plataforma en la nube basada en contenedores

Claudio Omar Biale
*Departamento de Informática,
Facultad de Ciencias Exactas,
Químicas y Naturales
Universidad Nacional de Misiones
Apóstoles, Argentina
claudiobiale@fceqyn.unam.edu.ar*

Ariel Lutenberg
*Laboratorio de Sistemas Embebidos,
Facultad de Ingeniería
Universidad de Buenos Aires
Buenos Aires, Argentina
alutenb@fi.uba.ar*

Resumen—En este trabajo se describe el diseño, desarrollo e implementación de un sistema de control de cultivos hortícolas bajo cubierta orientado al uso en ambientes urbanos y semiurbanos. El sistema está compuesto por hardware de bajo costo y una plataforma de software de dos niveles, en los planos local y en la nube. En el plano local los nodos sensores y actuadores recopilan datos y realizan acciones de control. La plataforma en la nube basada en contenedores recopila la información enviada por los nodos y se encarga de impartir órdenes de control de acuerdo a reglas definidas y asociadas a cada nodo. Los datos recopilados y el estado de los actuadores pueden ser visualizados en tiempo real por el usuario mediante una página web.

Palabras Clave—Internet de las cosas, MQTT, CouchDB, EMQX

I. INTRODUCCIÓN

En los últimos años la agricultura ha enfrentado muchos desafíos, desde una creciente población mundial a ser alimentada, hasta requisitos de sostenibilidad y restricciones ambientales debido al cambio climático y el calentamiento global [1].

La existencia de diversos problemas sociales y económicos también amenaza la productividad de la horticultura. El envejecimiento de los productores, el poco interés en las generaciones más jóvenes para convertirse en agricultores, la falta de mano de obra y la reducción de la tierra utilizada para la agricultura pueden provocar una disminución de la productividad [1][2].

Esto obliga a implementar soluciones que permitan modernizar las prácticas agrícolas. En este contexto, la Agricultura 4.0 representa la última evolución de la agricultura de precisión. La misma se encuentra basada en el concepto de agricultura inteligente, donde convergen el uso de Internet de las cosas, computación en la nube, aprendizaje automático para el análisis de grandes volúmenes de datos, vehículos no tripulados y robótica [3][4].

A diferencia de la agricultura de precisión donde las tareas de gestión sólo se basan en datos de acuerdo a su ubicación, en la agricultura inteligente también se consideran los datos de acuerdo al contexto, la situación y los disparadores de eventos [5].

El desarrollo de las tecnologías asociadas a la Agricultura 4.0 han permitido que un invernadero se transforme

en un sistema de producción de plantas de ambiente controlado, propicio para ser usado en agricultura urbana, donde el control y monitoreo remoto usando nodos conectados a Internet brindan información precisa para la toma de decisiones. Esto permite reducir las labores humanas y optimizar la productividad y la calidad de los cultivos; protegiéndolos de plagas, enfermedades y de los efectos adversos de los fenómenos meteorológicos extremos [6][7].

El mercado de invernaderos inteligentes se encuentra en constante crecimiento. En el año 2020 se lo valoró en 1380 millones de dólares y se prevé que alcance los 2820 millones de dólares para el año 2028 [8].

En este contexto, el objetivo de este trabajo radica en el diseño, desarrollo e implementación de un prototipo de sistema que automatice el control y monitoreo de cultivos hortícolas bajo cubierta orientado al uso en ambientes urbanos y semiurbanos.

II. DISEÑO E IMPLEMENTACIÓN

En la Fig. 1 se muestra un diagrama en bloques en el que se pueden observar, de forma simplificada, las interacciones que se producen entre las distintas partes del sistema.

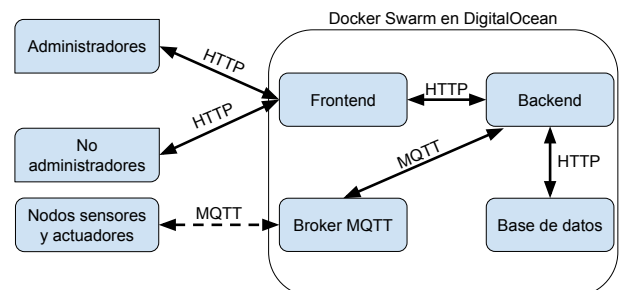


Fig. 1: Diagrama en bloques del sistema.

El sistema se desplegó en el proveedor de servicios en la nube DigitalOcean donde se conformó un clúster de Docker Swarm entre las distintas máquinas virtuales o droplets y está constituido por cuatro aplicaciones:

- Una aplicación web (frontend) desarrollada utilizando Svelte,

- Un servidor REST (backend) desarrollado utilizando Fastify,
- Una base de datos documental CouchDB y
- Un broker MQTT EMQX.

A nivel de protocolos, en la capa de aplicación se usan HTTP y MQTT. El primero se utiliza para la interacción de usuarios de tipo administrador y no administrador con el frontend, la comunicación entre el frontend y el backend, y entre el backend y la base de datos CouchDB. Por otro lado, MQTT es el protocolo utilizado para la comunicación entre los nodos sensores y actuadores con el broker MQTT, y entre este y el backend.

Con el objetivo de que las imágenes del frontend y backend se encuentren disponibles para los nodos del clúster se configuró un repositorio privado en Docker Hub.

Para la gestión de los contenedores dentro del clúster se utilizó un conjunto de scripts de Bash y Portainer Community Edition, una herramienta de gestión de contenedores de código abierto.

Las funcionalidades existentes en la aplicación del frontend permiten gestionar nodos, tipos de sensores, tipos de actuadores, reglas y usuarios. La gestión de nodos incluye el cambio en el tiempo entre envíos de datos, el acceso al dashboard, a la asociación de reglas, al historial de mediciones y al historial de logs.

En el backend se implementaron rutas y plugins que aceptan solicitudes del frontend y del broker MQTT. Se definió la forma en que el backend y los nodos sensores y actuadores se comunican utilizando el protocolo MQTT.

En la etapa inicial se desarrolló una aplicación en Node.js que simula el funcionamiento de un nodo sensor y actuador. La aplicación publica cada determinado tiempo un mensaje MQTT con valores aleatorios correspondientes a sensores de humedad ambiente, humedad de suelo y temperatura ambiente, y tiene como actuador una bomba de agua.

Se ha desarrollado un prototipo de nodo sensor y actuador sobre un módulo ESP32-WROOM-32, constituido por los siguientes componentes:

- Sensor de temperatura y humedad ambiente DHT22.
- Sensor de humedad del suelo HL-69.
- Válvula solenoide.
- Relés.
- Transformador de 12 V.

Para el desarrollo del firmware se utilizó Mongoose OS un framework y sistema operativo de tiempo real de código abierto que facilita el desarrollo de firmware aplicado a Internet de las cosas.

En [9] se encuentra disponible la memoria del trabajo final de posgrado donde se especifican cuestiones como el modelo de datos y el detalle de las acciones asociadas a nodos y backend en la comunicación usando el protocolo MQTT.

III. EVALUACIÓN

Para el despliegue del sistema en DigitalOcean se utilizaron dos máquinas virtuales o droplets provistas de 1 CPU compartida, 1 GB de memoria RAM, 25 GB de disco SSD y 1000 GB de transferencia.

Con el sistema desplegado se realizaron pruebas manuales de escalamiento horizontal del broker EMQX y

la base de datos CouchDB. Ambas pruebas consistieron en agregar un nuevo contenedor, realizar las acciones correspondientes para el armado del clúster y verificar que el clúster se haya conformado.

Adicionalmente se realizó una prueba de integración y funcionamiento de todas las partes. Los pasos de configuración de la prueba consistieron en:

- Acceder a la aplicación web con un usuario de tipo “administrador”.
- Crear un nodo.
- Crear los tipos de sensores correspondientes a temperatura ambiente, humedad ambiente y humedad del suelo.
- Crear un tipo de actuador correspondiente a una bomba de agua con funcionamiento en modo “on-off”.
- Definir dos reglas que permitan:
 - Activar la bomba de agua si la humedad del suelo es menor o igual al 30 %.
 - Desactivar la bomba de agua si la humedad del suelo es mayor o igual al 50 %.
- Asociar las reglas al nodo.
- Programar el prototipo de nodo sensor y actuador.

Una vez configurado el entorno de prueba se procedió a verificar que el sistema responda a cada una de las reglas asociadas al nodo. Para que se cumpla la primera regla de prueba se dejó el sensor de humedad del suelo sin contacto con agua y se verificó el estado del relé y los cambios a nivel del dashboard del nodo. De igual modo, para que se cumpla la segunda regla de prueba se colocó el sensor de humedad del suelo en agua y se verificó el estado del relé y los cambios a nivel del dashboard del nodo. Como paso final se procedió a verificar que los datos se almacenaron correctamente en la base de datos.

IV. COMPARACIONES CON OTROS TRABAJOS

Pliot [10] permite la obtención de datos mediante una red de sensores instalados en el invernadero que transmiten los datos de forma inalámbrica para su almacenamiento en la nube. Mediante una aplicación web o un celular se puede acceder a visualizar los datos actuales e históricos y a la administración de alertas. Soporta el cálculo de índices y algoritmos de predicción de acuerdo a los datos recolectados. La empresa ofrece la instalación, mantenimiento y calibración de equipos que son otorgados en comodato.

Seedmech [11] posibilita la obtención de datos de una huerta o invernadero mediante estaciones meteorológicas, estos datos se almacenan en la nube. Mediante una aplicación web o un celular se puede acceder a visualizar los datos actuales, históricos y distintos valores calculados. La disponibilidad de sensores y del sistema de riego viene dado por el modelo de estación meteorológica pero es posible agregar sensores adicionales. Hay modelos que disponen de un sistema de alertas mediante SMS. Las estaciones meteorológicas y los sensores son provistos por la empresa.

En [12] se describe el desarrollo de una plataforma compuesta por un servidor web NGINX, una aplicación desarrollada en Node-Red, un broker MQTT EMQX y una base de datos MySQL. Se utilizó como visor de datos a

Grafana y se desarrolló una aplicación en Python que se encarga de almacenar los datos que llegan al broker en la base de datos. Los nodos fueron implementados sobre un módulo ESP32 constituido por una electroválvula, un led RGB, un módulo GPS y sensores de temperatura y humedad ambiente y de humedad del suelo.

En [13] se desarrolló un sistema compuesto por nodos y un gateway conectados mediante el protocolo LoRa. El gateway se encarga de recibir los datos enviados por cada nodo y de enviarlos a una base de datos Firebase donde también se configuró un dashboard para una visualización amigable de la información. Los nodos se implementaron en módulos Arduino Uno y el gateway en una placa HELTEC LoRa 32.

En el trabajo desarrollado los datos se almacenan en la nube en servidores que son administrados por la empresa y no en servidores manejados por terceros como es el caso de Pliot y Seedmech. El soporte a distintos tipos de actuadores en Seedmech es limitado porque solo se proporciona de forma adicional el uso de actuadores para riego. Los tipos de sensores soportados por Pliot y Seedmech viene limitado porque se pueden usar aquellos provistos por las respectivas empresas. En [12] y [13] no se especifica la forma de adicionar nuevos sensores o actuadores. En el trabajo desarrollado se soportan sensores y actuadores a decisión del usuario final siempre que se cumpla con el protocolo de comunicación y que los tipos de sensores y actuadores se encuentren registrados en el sistema. Pliot permite administrar un sistema de alertas y Seedmech tienen soporte en algunos modelos de estaciones meteorológicas, este soporte no se encuentra en la implementación del trabajo desarrollado. Pliot, Seedmech y del libro de referencia.[13] soportan el uso de protocolos de comunicación adicionales pero estos protocolos no son necesarios para ambientes urbanos o semiurbanos. En [13] el control de cambio en los actuadores se encuentra especificado en el código de los nodos. En [12] el cambio en los actuadores se puede realizar de forma manual desde el dashboard o en caso de pérdida de conexión de acuerdo a lo especificado en el código de los nodos. En el trabajo desarrollado el cambio en los actuadores se produce de acuerdo a las reglas asociadas a cada nodo.

V. CONCLUSIONES

En este trabajo se completó el diseño, desarrollo e implementación de un prototipo de sistema escalable que automatiza el control y monitoreo de cultivos hortícolas bajo cubierta orientado al uso en ambientes urbanos y semiurbanos.

El sistema se desplegó en un proveedor de servicios en la nube. El uso de un clúster de Docker Swarm permitió que el sistema sea escalable horizontalmente y use los servicios de descubrimiento y balanceo de carga. Se comprobó el funcionamiento del escalado horizontal y la conformación de un clúster tanto en la base de datos como también en el broker MQTT.

A futuro se pretende mejorar el prototipo de nodo sensor y actuador, extender el sistema de reglas, automatizar la configuración del clúster al escalar horizontalmente los servicios, analizar alternativas en la forma de almacenamiento de los valores de sensores y actuadores en la base

de datos, trasladar el control de reglas de activación de actuadores al borde de la red y desarrollar un módulo que almacene los datos al borde de la red en situaciones donde no exista conexión de Internet, este módulo puede extenderse para definir en que caso se transfieren datos a la nube.

REFERENCIAS

- [1] Rafael Simionato et al. «Survey on connectivity and cloud computing technologies: State-of-the-art applied to Agriculture 4.0». En: *Revista Ciência Agronômica* 51 (2021).
- [2] Lukas Zagata y Lee-Ann Sutherland. «Deconstructing the ‘young farmer problem in Europe’: Towards a research agenda». En: *Journal of Rural Studies* 38 (2015), págs. 39-51.
- [3] Sara Oleiro Araújo et al. «Characterising the Agriculture 4.0 Landscape—Emerging Trends, Challenges and Opportunities». En: *Agronomy* 11.4 (2021), pág. 667.
- [4] Verónica Saiz-Rubio y Francisco Rovira-Más. «From smart farming towards agriculture 5.0: A review on crop data management». En: *Agronomy* 10.2 (2020), pág. 207.
- [5] Cor Verdouw et al. «Digital twins in smart farming». En: *Agricultural Systems* 189 (2021), pág. 103046.
- [6] Antonis Tzounis et al. «Internet of Things in agriculture, recent advances and future challenges». En: *Biosystems engineering* 164 (2017), págs. 31-48.
- [7] Redmond R Shamshiri et al. «Advances in greenhouse automation and controlled environment agriculture: A transition to plant factories and urban agriculture». En: *International Journal of Agricultural and Biological Engineering* 11.1 (2018), págs. 1-22.
- [8] Verified Market Research. *Smart Greenhouse Market Size, Share, Trends, Opportunities & Forecast*. <https://www.verifiedmarketresearch.com/product/global-smart-greenhouse-market-size-and-forecast-to-2025/>. Abr. de 2022. (Visitado 06-05-2022).
- [9] Laboratorio de Sistemas Embebidos Universidad de Buenos Aires. *Consulta de Trabajos Finales - Listado Completo*. <https://lse.posgrados.fi.uba.ar/trabajo-final/archivo-historico/consulta-de-trabajos-finales-listado-completo>. (Visitado 06-06-2023).
- [10] Pliot Smart Agriculture. *Pliot*. <https://pliot.ar/>. (Visitado 24-11-2022).
- [11] Seedmech Latinamerica SRL. *Home — SEED-MECH*. <https://www.seedmech.com/>. (Visitado 24-11-2022).
- [12] Facundo A. Chazarreta et al. «Diseño de una plataforma para servicios IoT con aplicaciones en el sector productivo». En: *Libro de Trabajos del CASE 2022, Congreso Argentino de Sistemas Embebidos, Argentina*. 2022.
- [13] Matías N. Mardegan et al. «Sistema embebido escalable para el monitoreo de parámetros en invernaderos mediante IoT». En: *Libro de Trabajos del CASE 2022, Congreso Argentino de Sistemas Embebidos, Argentina*. 2022.