

UNA ARQUITECTURA DE MICROSERVICIOS PARA DAR SOPORTE A LA CREACIÓN Y EJECUCIÓN DE ACTIVIDADES DE RECOLECCIÓN DE DATOS CON INTERVENCIÓN HUMANA

Jose Arcidiacono¹, Patricia Bazán², Alejandra B. Lliteras^{3,4}

¹ UNLP, Facultad de Informática, ² UNLP, Facultad de Informática, LINTI, ³ UNLP, Facultad de Informática, LIFIA, ⁴ CICPBA

j2arcidiacono@outlook.com, pbaz@info.unlp.edu.ar, alejandra.lliteras@lifia.info.unlp.edu.ar

RESUMEN

DEHIA es una plataforma para la creación y ejecución de actividades de recolección de datos que requieren de la intervención humana para realizar este objetivo.

La plataforma está definida sobre una arquitectura distribuida, basada en microservicios y cuenta con un prototipo funcional probado con un conjunto de usuarios [1].

Este trabajo presenta una versión revisada de la arquitectura respecto de lo presentado en [2] donde se describen cada uno de sus componentes y las capas en que fueron estructurados.

Palabras clave: Arquitecturas distribuidas, Microservicios, Recolección de datos, Intervención Humana

CONTEXTO

En el año 2020 se presentó una tesina de grado [1] proponiendo el diseño de una arquitectura para una plataforma de creación y ejecución de actividades de recolección de datos con intervención humana, junto con un prototipo implementando parte de la funcionalidad propuesta.

La motivación inicial de la tesina respondía a una iniciativa de un Proyecto de Extensión de la Facultad de Informática, “Recicla tu Compu-Recicla tu Mundo”¹ [3], que buscaba la creación de herramientas basadas en

tecnología para acercar a las escuelas las problemáticas de los RAEE (Residuos de Aparatos Eléctricos y Electrónicos).

Para encarar ese trabajo se tomó el enfoque de dos proyectos de investigación ([4], [5]) que proponen distintas estrategias para llevar a cabo actividades con intervención humana mediadas por tecnología, aplicando conceptos de Ingeniería de Software e Interacción Humano-Computador (HCI).

Existe otro trabajo de investigación que presenta una herramienta de autor de actividades educativas mediadas por tecnología móvil, basada en este mismo enfoque [6]. La tesina buscaba presentar un desarrollo nuevo que amplíe la funcionalidad de la herramienta propuesta en [6], tomando nuevos elementos del enfoque [4] y proveyendo mayor robustez así como la posibilidad de abarcar otros campos como la recolección de datos para lo que se conoce como *Ciencia Ciudadana* [7], [8].

Se presentaron los avances de la tesina en WICC 2020 [2] así como en JAIIO 2020 ([9], [10]).

1. INTRODUCCIÓN

Los sistemas distribuidos pueden concebirse como aquellos cuya funcionalidad se encuentra fraccionada en componentes que al trabajar sincronizada y coordinadamente otorgan la visión de un sistema único, siendo la distribución transparente para quien hace uso del sistema. [11]. También se puede definir un sistema distribuido como aquel en

¹ Aprobado y subsidiado en la convocatoria de Proyectos de Extensión Universitaria 2018, de la UNLP. El mismo está destinado a promover la educación ambiental sobre los RAEE en escuelas de la región en el período 2018 al 2019.

el cual componentes de software o hardware ubicados en computadoras en red se comunican y coordinan sus acciones sólo por pasaje de mensajes [12].

Dentro de las arquitecturas distribuidas, se encuentran las arquitecturas de microservicios (MS), las cuales son aquellas en las que una única aplicación está compuesta por un conjunto de pequeños *servicios*, cada cual ejecutándose en su propio proceso y comunicándose por mecanismos ligeros, a menudo una API HTTP [13].

Los microservicios son una alternativa a los sistemas monolíticos, que consisten en un único proceso o unidad de código que resuelve tanto la interfaz, como la lógica de negocio y el acceso a los datos. En el caso de los microservicios, el sistema se descompone en componentes débilmente acoplados y comunicados por pasaje de mensajes.

Entre las ventajas de los microservicios a nivel mantenimiento, se tiene que la funcionalidad relacionada está en el mismo servicio, y un cambio en esa funcionalidad no afecta a los demás servicios [14].

Otra característica de los microservicios es que permiten usar una o varias tecnologías diferentes en cada servicio, lo que permite adaptarse a las necesidades de cada funcionalidad. Sin embargo, puede ser beneficioso mantener una diversidad limitada de tecnologías cuando se piensa que el equipo de trabajo debe poder lidiar con todas ellas.

Se diseñó e implementó una plataforma con una arquitectura basada en microservicios. En este trabajo se completa la propuesta que se hizo en [2].

Se tiene un modelo en cuatro capas, las cuales pueden observarse en la Figura 1 y se describirán a continuación:

- La primera capa, la de *clientes*, contiene los componentes que permiten a los usuarios la interacción con el resto de la plataforma. En esta

capa se cuenta con dos tipos de clientes. Desde el cliente Web se definen las actividades (funcionalidad a la cual pueden acceder los usuarios denominados “autores”) y también pueden resolverse de manera web (funcionalidad a la cual pueden acceder los “clientes web”). El cliente Móvil, es decir una aplicación para celular, también permite la ejecución de actividades. Este cliente es accedido por los “usuarios móviles”.

- La capa de compuertas contiene los componentes tecnológicos que permiten a los clientes comunicarse de manera transparente con los servicios (que serán descritos más adelante). Se tiene una compuerta por cada tipo de cliente de acuerdo a las necesidades de comunicación de cada uno.

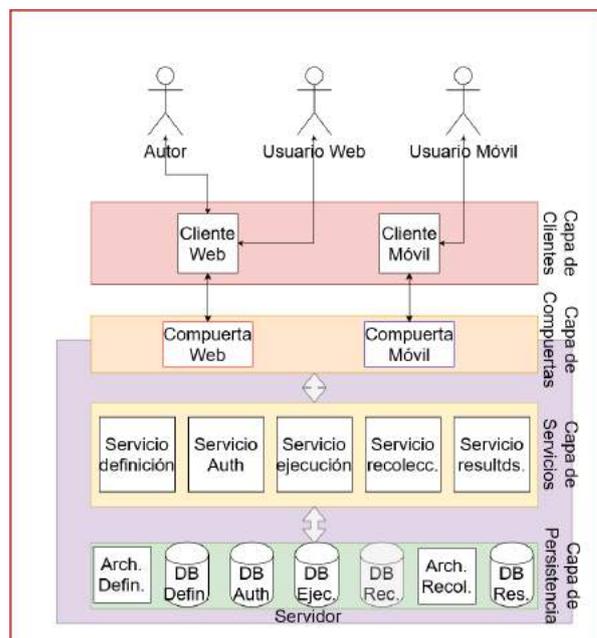


Figura 1. Arquitectura por capas de la plataforma

- La capa de servicios contiene los servicios, en este caso *microservicios* que brindan la funcionalidad principal de la plataforma: el servicio de

definiciones gestiona la gestión, listado y descarga de actividades; el servicio Auth (autenticación-autorización) gestiona los tokens de acceso para autenticar a los usuarios y autorizarlos a realizar determinadas tareas; el servicio de ejecución que mantiene el estado de ejecución de las actividades en Cliente Web; el servicio de recolección que recibe las respuestas producto de la resolución de las actividades; y finalmente, el servicio de resultados, que expone los datos de forma de poder ser consumidos para generar resúmenes, estadísticas y tablas.

- La última capa, la de persistencia, contiene los mecanismos para hacer permanentes los datos de los distintos servicios. En una arquitectura de microservicios, los datos son privados a los servicios, de modo que cada servicio tiene su propia base de datos, y dos de ellos además tienen sistemas de archivos. Además, son todas bases de datos SQL a excepción del servicio de recolección, que necesita una base de datos NoSQL.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

La tesina de grado que sustenta este trabajo dio origen a la definición de varias líneas de investigación y desarrollo y que abordan diferentes áreas, como 1- Ingeniería de Software, 2- Interacción Humano-Computador, 3 - Usabilidad y Accesibilidad tanto web como móvil, 4- Arquitecturas Distribuidas, 5- Workflows, 6 - Herramientas de autor, 7 - Testing y 8- Recolección de datos.

De todas las áreas mencionadas anteriormente, este trabajo aborda la de arquitecturas distribuidas y define una línea

de investigación que apunta a mejorar aspectos de los microservicios.

Una arquitectura distribuida con microservicios trae beneficios pero también conlleva nuevos desafíos. Un ejemplo es la distribución por componentes en contenedores independientes que aportan a la interoperabilidad pero que aumenta la complejidad operacional y requiere automatización y monitoreo.

En este sentido, la mejora en torno a los microservicios que aborda la línea de trabajo apunta a considerar requerimientos no funcionales (escalabilidad, seguridad) a partir de un mejor aprovechamiento de las capacidades que ofrecen los microservicios, sin perder de vista el mantenimiento, despliegue y monitoreo de los mismos.

El estilo arquitectónico de los microservicios aborda el desafío de las arquitecturas monolíticas tradicionales donde el software se construye en forma de bloque complejo e indivisible. Para aprovechar los beneficios del estilo arquitectónico de microservicios, es necesario utilizar tecnologías alineadas con las características de los microservicios [15]. La contenerización (*containerization*) se ha convertido en un formato de implementación popular para microservicios, y Docker [16] es el contenedor líder siendo una plataforma que empaqueta todo lo necesario para implementar y ejecutar microservicios.

Por otra parte, para permitir orquestar y gestionar la implementación de contenedores, se requieren plataformas específicas, siendo Kubernetes, un caso de ellas. Kubernetes [17] es la plataforma líder para la automatización implementación, escalado y administración para contenedores aplicaciones y facilita la tarea de los desarrolladores, manejando la complejidad de implementar la resiliencia de sus aplicaciones y les permite concentrarse en la lógica de negocio de las aplicaciones [18].

3. RESULTADOS ESPERADOS / OBTENIDOS

Se propuso una arquitectura distribuida basada en microservicios que permitirá la gestión y ejecución de actividades recolección de datos con intervención humana. Se plantearon los lineamientos para una plataforma que cumpliera con esta funcionalidad y fuera accesible para usuarios sin conocimientos de programación. La plataforma debía favorecer la reutilización de conocimiento (actividades, tareas, planificaciones) propio y de otros usuarios.

Se pretendía desarrollar un prototipo funcional de esta plataforma aplicándolo en un caso de estudio que permitiera hacer pruebas con usuarios.

Finalmente se pudo implementar el prototipo basado en la arquitectura y los lineamientos propuestos, y se tomó como caso de estudio una necesidad del Programa E-Basura de la Universidad Nacional de La Plata². Se instanció el prototipo para representar con actividades uno de los formularios³ de recepción de donaciones del Programa. Se realizaron pruebas de usabilidad con usuarios experimentados y no experimentados en la configuración de herramientas. Las pruebas indicaron que la plataforma podría ser usada por personas sin conocimientos de programación, llegando a niveles de usabilidad aceptables [1].

4. FORMACIÓN DE RECURSOS HUMANOS

El trabajo de investigación realizado por la primera autora de este artículo ha permitido consolidar varias de las líneas de investigación del actual proyecto de investigación del LINTI “De la Sociedad del Conocimiento a la Sociedad 5.0: un abordaje

tecnológico y ético en nuestra región”⁴ que entre sus amplios objetivos específicos propone promover buenas prácticas de seguridad en las distintas plataformas de despliegue de aplicaciones garantizando la disponibilidad, integridad y confidencialidad de la información.

En particular la temática de la tesina será continuada por su autora desde el enfoque de la arquitectura a través de una propuesta de doctorado a presentarse en 2021.

5. BIBLIOGRAFÍA

[1] Arcidiacono, J (2020). DEHIA: una plataforma liviana para definir y ejecutar actividades con intervención humana basadas en workflows. Tesina de grado. Facultad de Informática, UNLP

[2] Arcidiacono, J., Lliteras, A. B., & Bazán, P. A. (2020). Plataforma para la definición y ejecución de actividades orientadas a la recolección y análisis de datos, con intervención humana. In XXII Workshop de Investigadores en Ciencias de la Computación (WICC 2020, El Calafate, Santa Cruz).

[3] Proyecto Recicla tu Compu-Recicla tu mundo. Proyecto de extensión de la UNLP, Facultad de Informática, LINTI. <http://bit.ly/3eoYw0e>

[4] Lliteras, A. B. (2015). Un enfoque de modelado de actividades educativas posicionadas que contemplan elementos concretos. Tesis de maestría. Facultad de Informática, UNLP.

[5] Lliteras, A. B., Grigera, J., dal Bianco, P. A., Corporaal, F. M., & Gordillo, S. E. (2018). Challenges in the design of a customized location-based mobile learning

²<https://e-basura.unlp.edu.ar/>

³<https://encuestas.linti.unlp.edu.ar/index.php/369463/lang-es>

⁴ Programa de Incentivos a la Investigación del Ministerio de Educación de la Nación

application. In 2018 XIII Latin American Conference on Learning Technologies (LACLO) (pp. 315-321). IEEE.

[6] Dal Bianco, P. A., Mozzon Corporaal, F., Lliteras, A. B., Grigera, J., & Gordillo, S. E. (2019). MoLE: A web authoring tool for building mobile learning experiences. In XXV Congreso Argentino de Ciencias de la Computación (CACIC 2019, Universidad Nacional de Río Cuarto).

[7] Bonney, R., Cooper, C. B., Dickinson, J., Kelling, S., Phillips, T., Rosenberg, K. V., & Shirk, J. (2009). Citizen science: a developing tool for expanding science knowledge and scientific literacy. *BioScience*, 59(11), 977-984.

[8] Strasser, B. J., Baudry, J., Mahr, D., Sanchez, G., & Tancoigne, E. (2019). "Citizen Science"? Rethinking Science and Public Participation. *Science & Technology Studies*, 52-76.

[9] Arcidiacono J, Bazán P y Lliteras AB (2020). Arquitectura de microservicios distribuidos para una plataforma que orquesta actividades orientadas a la recolección y análisis de datos, con intervención humana en JAIIO 2020.

[10] Arcidiacono J, Lliteras AB. y Bazán P (2020). DEHIA, una plataforma para la generación y ejecución de actividades de recolección de datos con intervención humana aplicada en el Programa E-Basura en JAIIO 2020.

[11] Bazán P. et al. (2017). *Arquitecturas, Servicios y Procesos Distribuidos. Una Visión desde la construcción del software*. Libro de Cátedra. Editado por EDULP (Editorial de la

UNLP). ISBN 978-950-34-1520-7
<https://bit.ly/3chD0rh>

[12] Colouris, G et al. (2000) *Distributed Systems Concepts and Design 3e*. Addison-Wesley. ISBN- 13: 978-0132143011

[13] Fowler, Martin, and James Lewis. "Microservices a definition of this new architectural term." URL: <http://martinfowler.com/articles/microservices.html> (2014).

[14] Newman, Sam. *Building microservices: designing fine-grained systems*. O'Reilly Media, Inc., 2015.

[15] N. Dragoni et al., "Microservices: Yesterday, Today, and Tomorrow," in *Present and Ulterior Software Engineering*, M. Mazzara and B. Meyer, Eds. Cham: Springer International Publishing, 2017, pp. 195–216.

[16] "Docker" [Online]. Disponible: <https://www.docker.com/>.

[17] "Kubernetes," Kubernetes. [Online]. Disponible: <https://kubernetes.io/>.

[18] Vayghan, L. A., Saied, M. A., Toeroe, M., & Khendek, F. (2019, July). Microservice based architecture: Towards high-availability for stateful applications with Kubernetes. In 2019 IEEE 19th International Conference on Software Quality, Reliability and Security (QRS) (pp. 176-185). IEEE.