

Programación web en varias capas para bases de datos curadas: un estudio de caso

Cross-tier web programming for curated databases: a case study

Simon Fowler, y otros

International journal of digital curation, ISSN 1746-8256, Vol. 16, n. 1, 2021

Las bases de datos curadas se han convertido en valiosas fuentes de información para varias disciplinas científicas y, como resultado del trabajo manual de los expertos, a menudo se transforman en importantes trabajos de referencia. Características tales como el seguimiento de la procedencia, el archivo y la cita de datos son consideradas importantes para las bases de datos curadas, no obstante, implementar tales prestaciones supone un desafío y los proyectos de bases de datos pequeños frecuentemente carecen de los recursos necesarios para hacerlo. Una aplicación de bases de datos científicas no es solo la base de datos relacional en sí, sino también un ecosistema de aplicaciones web que muestran los datos y otras que permiten la curación de los mismos. El soporte de funciones avanzadas de curación requiere cambiar todos estos componentes y actualmente no existe una forma de proporcionar dichas prestaciones de manera reutilizable. Los lenguajes de programación en varias capas permiten a los desarrolladores crear una aplicación web en un lenguaje único y uniforme. En consecuencia, las consultas y actualizaciones de la base de datos pueden escribirse en el mismo lenguaje que el resto del programa, siendo posible proporcionar funciones de curación a través de transformaciones del programa. Como paso hacia esta meta, es importante establecer que las bases de datos curadas pueden ser realmente implementadas en un

lenguaje de programación en varias capas. En este artículo describimos un estudio de caso de este tipo: reimplementación de la interfaz de una base de datos científica real, la *IUPHAR/BPS Guide to PHARMACOLOGY* (GtoPdb), utilizando el lenguaje de programación por capas Links. Mostramos cómo algunas de las características del lenguaje de programación, tales como la consulta integrada de lenguaje, simplifican el proceso de desarrollo y descartan errores comunes. A través de una evaluación de corrección funcional automatizada, probamos que la implementación de Links mantiene correctamente la funcionalidad de la versión oficial. A través de una evaluación comparativa del rendimiento, demostramos que la implementación de Links realiza menos consultas a la base de datos, si bien el tiempo necesario para gestionarlas es comparable al de la versión oficial de Java. Asimismo, aunque existen ciertos sobrecostos al usar Links debido a su falta de madurez en comparación con Java, la versión de Links puede emplearse como un estudio de caso de una prueba de concepto del uso de programación en varias capas en bases de datos curadas.

<https://doi.org/10.2218/ijdc.v16i1.735>

Traducción del resumen de la propia publicación