



## Docencia 2.0

Juan Julián Merelo, Fernando Tricas

### Informática y ‘mundo real’: Hackatones

Mucho se habla en las enseñanzas técnicas, y en particular en la de la informática, de enlazarlas con la realidad social y empresarial que existe fuera de la academia. Para lograr este enlace tenemos el enfoque tradicional, basado en la realización de trabajos de fin de carrera, prácticas en empresas de muy diverso tipo y el propio desarrollo de las asignaturas, que en la mayoría de los casos no están tan alejadas de la realidad como se afirma. Pero ya llevamos una temporada larga escuchando hablar (y participando, en algunos casos) de nuevas formas de sumergir al alumnado en la realidad del desarrollo de proyectos informáticos. Un hackatón es un maratón de programación, de hack, que estrictamente sería hacer una ñapa o apaño en un proyecto. En general, se trata de llevar a cabo un proyecto de programación en un tiempo limitado. La idea es bastante simple, pero llevarla a cabo no es trivial; integrarla dentro de un currículum de enseñanza de alguna asignatura de informática tampoco. En realidad, cualquier asignatura que tenga prácticas no tendrá demasiado complicado añadirlo, pero para que resulte especialmente útil hacen falta una serie de requisitos. El primero, que efectivamente los contenidos de la asignatura estén enfocados a llevar a cabo un proyecto, que puede ser diseñar un sistema, programarlo o poner en marcha una empresa relacionada con el mismo. No será tan útil en asignaturas que no incluyan esa habilidad, por ejemplo una de contenido matemático. Segundo, que se propongan una serie de proyectos atractivos y reales, donde los alumnos realmente vean el fruto de su trabajo. Tercero, que la evaluación del trabajo sea también lo más real posible. Si los proyectos han sido propuestos por una persona de una empresa, habrá que contar con ella para evaluarlos a ver si han alcanzado los objetivos. Hay que hacer notar aquí que el realismo de los proyectos puede ser tan variable como el de la industria: desde resolver necesidades puntuales de proyectos completos y más complejos a realizar desarrollos puramente prospectivos y exploratorios para probar las posibilidades de determinadas tecnologías; el principal requisito de un proyecto es que tenga objetivos alcanzables dentro del tiempo destinado al hackatón y que se pueda evaluar si se han alcanzado o no los objetivos del mismo. Puede haber un cliente concreto (alguien que quiere conseguir determinada característica para algún producto) o los clientes pueden ser los propios desarrolladores y su deseo puede ser el de agradar al jurado o tribunal (si se plantea

como un concurso) o a la sociedad y personas relacionadas con la convocatoria. Además de la componente de desarrollo informático, un hackatón incluye típicamente algunas otras de las competencias transversales que se reclaman a la Universidad española: formar equipos y trabajar dentro de ellos —a veces ser líder, otras ser un componente más; en cualquier caso, coordinarse para llevar a cabo los objetivos propuestos—, discutir ideas y alternativas, utilizar herramientas de coordinación, trabajar en la comunicación del avance del proyecto mientras se desarrolla, y posteriormente a los otros equipos y personas interesadas en exposiciones más o menos públicas. Finalmente, quién sabe si tomar la decisión de llevar el proyecto a etapas posteriores con su mantenimiento, evolución y tal vez comercialización. En todo caso, como profesores podríamos pensar en realizar un hackatón sobre proyectos que consideremos relevantes (¿puesta al día de alguna funcionalidad para nuestro centro?) pero también podríamos animar al alumnado a participar en hackatones que se organicen en el contexto local. En este último caso suelen aparecer algunos resultados laterales nada despreciables: contacto con la sociedad, mediante el conocimiento por parte de la comunidad que organizó el encuentro de nuestros estudiantes universitarios

---

*JJ Merelo* es catedrático de Universidad en el área de Arquitectura y Tecnología de Computadores, y actualmente director de la Oficina de Software Libre de la UGR. Mantiene un blog desde el año 2002, y lo ha utilizado en clase desde el año 2004; también wikis y, últimamente, agregadores y otras herramientas TIC. Últimamente le ha dado por el *flipped learning*, de lo que se informará debidamente en esta columna.



*Fernando Tricas García* es profesor titular de Lenguajes y Sistemas Informáticos del Departamento de Informática e Ingeniería de Sistemas de la Universidad de Zaragoza. Empezó a estudiar la blogosfera casi cuando aún no existía (allá por el año 2002) y a tratar de integrarla en los cursos y tareas docentes un poco después. Ha impartido numerosas charlas relacionadas con el tema de la Web 2.0. Es actualmente Director de su departamento.



y creación de relaciones; nuestros estudiantes descubren que pueden interaccionar con desarrolladores del 'mundo real' y sacar adelante proyectos a su mismo nivel; también descubren que hay aspectos que desconocen y a los que deberán prestar más atención (típicamente aspectos de comunicación, pero también tecnologías y formas de trabajar con las que no habían tenido contacto); quién sabe si hasta oportunidades laborales y de mayor implicación en nuevos proyectos. Muchas de estos resultados los vimos en el hackatón que se hizo como última práctica de la asignatura Informática Virtual del Grado en Ingeniería Informática en la UGR <http://jj.github.io/IV/documentos/practicas/4.Aplicaciones>. El hackatón final se hizo conjuntamente con otra asignatura, Diseño de Aplicaciones para Internet y se llevó a cabo durante un fin de semana (de viernes por la mañana a lunes por la mañana) en un espacio de coworking, de forma que el espacio de trabajo también fuera lo más real posible. La

idea original partió de una conferencia de Juan Freire en el que se comentaban las experiencias en el MediaLab Prado. Todos los alumnos consideraron la práctica muy positiva o positiva [https://docs.google.com/forms/d/1kzeksc0fWkReZbf3vFp5i77hoasGW8zyoP\\_CmYqOCdE/viewanalytics](https://docs.google.com/forms/d/1kzeksc0fWkReZbf3vFp5i77hoasGW8zyoP_CmYqOCdE/viewanalytics) y, de hecho, pensamos repetirla en años sucesivos y quizás ampliarla a otras asignaturas.

*Todas las columnas de la serie Docencia 2.0 pueden descargarse en formato LaTeX desde <https://github.com/ReVision-Docencia-20/Columns>*

---

©2014 JJ. Merelo, F. Tricas. Este artículo es de acceso libre distribuido bajo los términos de la Licencia Creative Commons de Atribución, que permite copiar, distribuir y comunicar públicamente la obra en cualquier medio, sólido o electrónico, siempre que se acrediten a los autores y fuentes originales