

Estudio retrospectivo de una experiencia de aprendizaje fuera del aula sobre lenguajes y plataformas de programación

Yania Crespo Margarita Gonzalo
Departamento de Informática
Edificio TIC. Campus Miguel Delibes
Universidad de Valladolid
47011 Valladolid
{yania, marga}@infor.uva.es

Resumen

Se presenta un estudio retrospectivo de un evento de aprendizaje fuera del aula sobre lenguajes de programación (nivel iniciación) así como frameworks o plataformas de desarrollo para dicho lenguaje (nivel avanzado) celebrado en los últimos cinco años. Se trata de una experiencia donde un grupo de estudiantes enseña a otros estudiantes. Se aportan datos sobre cada una de las ediciones y se presentan los resultados del estudio retrospectivo para analizar la experiencia desde el punto de vista de los dos tipos de participantes: los asistentes y los instructores. Los datos para el análisis se han obtenido mediante encuestas online. Se concluye que la experiencia es positiva en ambos casos, que los asistentes prefieren este tipo de aprendizaje al aula y que los instructores aprecian la adquisición o mejora de *soft skills*.

Abstract

A five-year retrospective study of an out-of-class learning event on programming languages (initiation level) as well as frameworks or development platforms for that language (advanced level) is presented. It is an experience where a group of students teaches other students. Data are provided on each of the editions and the results of the retrospective study are presented to analyse the experience from the point of view of the two types of participants: the attendees and the instructors. The data for the analysis have been obtained through online surveys. It is concluded that the experience is positive in both cases, attendees prefer this type of learning to the classroom and that instructors appreciate soft skills acquisition or improvement.

Palabras clave

lenguajes de programación, aprendizaje fuera del aula, aprender enseñando, hora del código, desarrollo de *soft skills*

1. Introducción y motivación

HourOfCode es una iniciativa de Code.org que se enmarca en la Semana de Educación en Ciencias de la Computación (Computer Science Education Week). Code.org es una organización sin ánimo de lucro y ha conseguido desde el año 2013 hasta la fecha, con el apoyo de grandes instituciones y empresas, así como de numerosos voluntarios a nivel mundial, que la Ciencia de computación sea parte de la conversación educativa internacional. El término que se usa en este artículo es su traducción al español como “La Hora del Código” pero con ánimo de abreviar se utilizará el acrónimo HoC a partir de sus siglas en inglés.

En el año 2014, habiendo tenido experiencia previa en el año 2013 y en el propio 2014 de conducir eventos HoC destinados a escolares, nos planteamos la posibilidad de celebrar la semana de la Educación en Ciencias de la Computación en el entorno universitario. Se requería realizar una adaptación para celebrar nuestra particular HoC en un entorno donde es habitual que el público objetivo sepa programar en al menos un lenguaje de programación. Desde esa edición de 2014 hasta la fecha hemos celebrado cada año de forma ininterrumpida lo que informalmente llamábamos “nuestra particular HoC”.

Esta secuencia de eventos se ha convertido en un evento arraigado en la Escuela, una “tradición”, y en una experiencia educativa fuera del aula donde se aplica simultáneamente “enseñanza entre estudiantes” (*student teaching* también conocido como *student teach student*), aprendizaje basado en problemas o casos prácticos y que, además, comienza a tener un impacto cada vez mayor en un público de fuera de nuestro centro, consiguiendo un efecto de difusión de los estudios. En esta ponencia presentamos dicha experiencia, su evolución a lo largo de los cinco años, así como algunos resultados de un estudio que hemos realizado para analizarla retrospectivamente.

A continuación, la ponencia se organiza de la siguiente forma. Las Secciones 2 y 3 presentan, respectivamente, algunos trabajos relacionados con el que aquí se presenta, así como las características particulares de

La Hora del Código en nuestro entorno y su evolución. A continuación, en la Sección 4 se explica el diseño y ejecución del estudio. En la Sección 5 se muestran los resultados del estudio sobre la evolución de la oferta formativa. Posteriormente, en la Sección 6 se detallan los resultados obtenidos del estudio de la experiencia enseñanza-aprendizaje desde un punto de vista cuantitativo, así como una valoración cualitativa. En la Sección 7 se presentan algunas lecciones aprendidas. Para finalizar, la Sección 8 expone las conclusiones a las que se ha llegado.

2. Trabajos relacionados

La Hora del Código, organizada a nivel global por Code.org, tiene como objetivos ampliar el acceso a la programación en las escuelas [10] y aumentar la participación de las mujeres y las minorías [9]. Code.org pretende que cada estudiante, de entre 4 a 15 años, en cada escuela tenga la oportunidad de aprender programación, al igual que aprende otras materias tradicionales.

Las Universidades se han involucrado en este evento desde los inicios, por ejemplo, ofreciendo a sus profesores y estudiantes de Informática como profesores para estudiantes de primaria y secundaria como se puede leer en [5], o utilizando los tutoriales online de Code.org para estudiar su impacto en la motivación de los estudiantes universitarios para aprender programación descrito en [3] y [8].

No hemos encontrado artículos relativos a experiencias similares a la que describimos en esta ponencia de eventos especiales HoC para universitarios con conocimientos sobre programación, aunque sí que existen dos conceptos similares en los elementos que intervienen (estudiantes universitarios de Informática) pero que divergen en la finalidad; estos conceptos son: la mentoría entre iguales (“*peer-mentoring*”) y la tutoría entre iguales (“*peer-tutoring*”). Mientras que la mentoría entre iguales, como bien describe [6], sirve de apoyo en los primeros pasos de los estudiantes noveles en un grado de Ingeniería Informática por parte de estudiantes veteranos, la tutoría de iguales descrita, entre otros, por [2] parte del mismo conjunto de estudiantes, noveles y veteranos, para asesorar y mejorar el aprendizaje de una asignatura. En [1] se habla de *Student Pair teaching* como un caso de estudio de *Pair Teaching*, en el que estudiantes de años anteriores asesoran y enseñan a los estudiantes que comienzan una asignatura para encauzar sus ideas antes de que las lleven a cabo.

Estas experiencias tienen un matiz distinto a la que describimos en este artículo ya que en nuestro caso, estudiantes “expertos” en tecnologías o lenguajes de programación se lanzan a enseñar (previa preparación) a otros estudiantes. Este matiz se ha descrito en [4] como un paradigma emergente en la enseñanza: “*Aprender enseñando*” ya que, según [4] se puede organizar

la educación de forma que los estudiantes tengan oportunidades de aprender enseñando. En la experiencia que describimos en el presente artículo, exponemos un caso en el que un grupo de estudiantes universitarios formados y entrenados en algún lenguaje de programación y/o tecnología, enseñan a otros estudiantes, no importa de qué cursos.

En nuestra Escuela, como en otras universidades, el grupo de estudiantes de ACM realiza entrenamientos de programación competitiva y el Grupo Universitario de Informática realiza algunos cursos y talleres a lo largo del año. Esta experiencia que describimos tiene la particularidad de ser un evento en un único día, con un grupo de talleres simultáneos de iniciación a un lenguaje de programación, y posteriormente avanzados para los que están iniciados en el(los) lenguaje(s) base del taller, y desean realizar desarrollos basados en algún framework o plataforma de programación. Se trata de observar en qué lenguaje o tecnología se concentra más el interés de los asistentes, analizar su evolución y, adicionalmente, analizar la experiencia de aprendizaje en la que unos estudiantes enseñan a otros. No hemos encontrado en nuestras búsquedas análisis parecidos.

3. Particularidades de nuestra HoC

Nuestra HoC se caracteriza por ser un evento en el que no se enseña a programar sino que se pide que el participante asegure que ya sabe programar en algún lenguaje de programación. Todas las otras HoC que conocemos y en las que hemos participado, tienen como objetivo introducir a la programación a los que no saben, o permitir utilizar una hora para programar cuando la actividad principal que se desarrolla gira entorno a otras tareas. En general estos eventos se organizan para escolares y/o sus familiares.

La Hora del Código de la Escuela de Ingeniería Informática de Valladolid (UVa) es una oportunidad para celebrar la Semana de Educación en Ciencias de la Computación aprendiendo y enseñando lenguajes de programación que no se imparten o se usan como medio de enseñanza en los cursos de las titulaciones de la Escuela. A estos talleres los llamamos talleres de iniciación. Desde el punto de vista del instructor, el que enseña, es una oportunidad de prepararse más a fondo en el conocimiento del lenguaje que enseñará en nuestra HoC. Los instructores preparan mini proyectos a través de los cuales se guiará el aprendizaje. Además, como se trata de un público objetivo especial, en nuestra HoC ofrecemos talleres para adentrarse en un framework o plataforma de desarrollo para alguno de los lenguajes de programación introducidos en los talleres de iniciación. A éstos los llamamos talleres avanzados.

La organización se realiza en base a desarrollar primero los talleres de iniciación. Esto ha ido evolucionando de dedicarle una hora a una hora y media o

dos horas. Los talleres avanzados comienzan posteriormente y se planifican para dos horas, dos horas y media o tres horas.

Desde la primera ocasión se adoptó un modelo BYOL (Bring Your Own Laptop). Se utiliza una gran sala disponible en nuestro centro para el estudio y el trabajo individual o en equipo, también disponible para la realización de otras actividades como es este caso. En dicha sala montamos una infraestructura de islas de mesas para cada taller, a las que se lleva la conectividad de electricidad y redes cableadas de datos. Los miembros de la comunidad universitaria pueden utilizar eduroam pero para evitar problema de saturación de la conexión inalámbrica y para permitir conectarse a aquellos que participan provenientes de otras instituciones se realiza el cableado temporal. El uso de esta amplia sala polivalente nos resulta atractivo porque se crea un ambiente de aprendizaje que se contagia y facilita la organización a la hora de verificar la asistencia por cuestiones de certificación de participación así como facilita el paso de los talleres de iniciación a los talleres avanzados.

En la primera edición se optó por abordar algunos de los talleres de iniciación utilizando tutoriales online pero esta opción se descartó a partir de la segunda edición para permitir aprovechar mejor la condición de los asistentes de saber programar en algún lenguaje. Por otra parte, se decidió que siempre la metodología sería basada en proyectos, mini proyectos en Iniciación y proyectos un poco más ambiciosos en los Avanzados. Los tutoriales online se atascan en detalles que para un participante ya conocedor de los fundamentos de la programación son irrelevantes y se pierde tiempo.

Para los talleres, se preparan entornos de desarrollo en la nube basados en Codebox¹ en el que quedaba desplegada toda la instalación necesaria para que un asistente se conecte y tenga ya instalado todo lo se utiliza en el taller y probadas todas las dependencias para que funcionasen sus primeras aplicaciones.

En las siguientes URLs se encuentran ejemplos del material preparado para un taller de iniciación y uno avanzado, respectivamente:

- <https://trello.com/c/huL5ziEJ>
- <https://trello.com/c/9NB7Xc8n>

Los instructores son principalmente alumnos. Alumnos que enseñan a alumnos. Estos alumnos, en general aunque no necesariamente, como puede ver se más adelante, están vinculados a diferentes asociaciones: Grupo Universitario de Informática (GUI), UVaCoders (grupo de programación competitiva), otros agrupados a través de la delegación de alumnos, que acuden a la llamada general a voluntarios.

¹<https://github.com/CodeboxIDE/codebox>

4. Diseño del estudio

Analizar nuestra HoC retrospectivamente tras 5 ediciones a través de las siguientes preguntas:

RQ1 ¿Cuál ha sido la evolución del interés por los talleres de los diferentes lenguajes de programación, plataformas y frameworks?

RQ2 ¿Cómo ha sido la experiencia de enseñanza-aprendizaje?

Dado que se trata de una experiencia de estudiantes enseñando a estudiantes, nos interesa analizar en ambos casos desde ambas perspectivas, cuando es el estudiante el que enseña y cuando es un asistente que es enseñado por otro estudiante. Ambas preguntas se analizan entonces desde las siguientes perspectivas: punto de vista del alumno asistente, punto de vista del alumno instructor.

Para responder a RQ1 se utilizarán los registros de datos almacenados de las distintas ediciones con la oferta formativa y el número de inscritos.

Para responder a RQ2 se realizarán dos encuestas que se enviarán por email, una para los asistentes y otra para los instructores.

4.1. Ejecución

Se realizó la recopilación de datos y se crearon archivos csv con la oferta, instructores e inscritos por cada año. Se diseñaron dos cuestionarios cuyas preguntas y respuestas se pueden obtener en estas dos URLs:

- Punto de vista del instructor:
<https://gitlab.inf.uva.es/snippets/29>
- Punto de vista del asistente:
<https://gitlab.inf.uva.es/snippets/30>

Se recolectaron todas las direcciones de correo electrónico de los inscritos en las cinco ediciones. De aquellos que participaron el día del evento pero no se habían inscrito no se dispone de la dirección de contacto. Se envió a todas estas direcciones el enlace a un formulario online con una encuesta para los asistentes. Se cuenta con dos problemas. El primer problema se tuvo con las cuentas de correo institucionales para el caso de los alumnos ya graduados. La mayoría de estas cuentas ya no existen. El segundo problema es que realmente no sabemos a cuántos les ha llegado la encuesta online.

5. Resultados sobre la evolución del interés en lenguajes y tecnologías (RQ1)

En primer lugar se analiza la oferta formativa puesto que ésta refleja el interés y el dominio de los instructores por los lenguajes, plataformas y frameworks. Las dos primeras ediciones se describen sin cifras pues no contamos con datos precisos.

5.1. Evolución de la oferta formativa

En la primera edición, en el año 2014, se lanzó una oferta formativa planteada como talleres introductorios de una hora de duración al estilo tradicional de HoC para introducirse a las particularidades de los siguientes lenguajes: Ruby, Golang, Python, Javascript, Android y Php. Se propuso que, aquellos que quisiesen profundizar más, pudiesen continuar con los talleres avanzados por la tarde. En estos talleres se apostó por un aprendizaje basado en problemas o casos prácticos como programar un servidor web en Ruby o Php, etc. Había una relación 1-1 talleres de iniciación-avanzados.

La segunda edición, año 2015, lanzó una oferta formativa de talleres de iniciación a: Python, Golang, Processing, C++, R, JavaScript, Ruby, Php y HTML5+CSS. En esa edición se introdujo Python en iniciación para responder a las peticiones que nos hicieron llegar los estudiantes. El taller de HTML5+CSS3 se añadió para ofrecer una iniciación a potenciales participantes del taller avanzado identificado como 2015-3 en el Cuadro 1. Los talleres de iniciación continuaron teniendo una hora de duración. Ese año se decidió concentrar esfuerzos en los talleres avanzados y no ofertar uno por cada lenguaje de programación ofertado en iniciación. La razón de esta decisión fue, por una parte, permitir ampliar la oferta de iniciación sin vincularla con la necesidad de ampliar cada lenguaje en talleres avanzados y, por otra parte, concentrar a varios instructores colaborando para impartir un mismo taller avanzado. De esta manera, se le dio forma a la idea de los talleres avanzados que se aplicaría en las siguientes ediciones: aprender el uso de una tecnología (framework, plataforma) vinculada con uno o varios lenguajes de programación, guiándose por un caso práctico.

A partir de la tercera edición se cuenta con datos completos de inscripciones por cada taller ofertado. Los Cuadros 3 (Iniciación) y 4 (Avanzados) resumen la evolución de la oferta formativa a lo largo de los años 2016, 2017 y 2018 así como la demanda en términos de número de inscritos con el objetivo de apreciar la evolución del interés de los inscritos tanto en lenguajes de programación como en tecnologías. Se indica NO cuando no se ofertó el taller y CL cuando se ofertó pero fue cancelado por ausencia de demanda.

En 2016 valoramos como un error la introducción de Android en la fase de Iniciación fundamentalmente porque no es un lenguaje de programación como tal. Decidimos mantenernos en la idea original de que los talleres de Iniciación son para iniciarse a lenguajes de programación y los avanzados son para aprender tecnología (frameworks y plataformas).

En esta tercera edición, se extendió la duración de los talleres de iniciación y se continuó con la propuesta de que los talleres avanzados se extendiesen hasta media tarde.

La introducción de lenguajes más vinculados a algu-

nos fabricantes como C# a Microsoft y Swift a Apple se nos hizo llegar como petición de algunos grupos de alumnos con interés. En nuestros estudios los lenguajes que se utilizan no están vinculados a ningún fabricante de sistema operativo/hardware. El taller de C# tuvo mucha demanda y asistencia, pero por el contrario, el taller de Swift no. Hablando con los alumnos llegamos a la conclusión de que era debido a la necesidad de disponer de hardware y software de Apple. Esto también influyó en la baja participación en el taller avanzado referido como 2017-4. Estos talleres podrían haberse cerrado por baja inscripción pero se decidió impartirlos por comenzar a fomentar un grupo de interés en desarrollo para iOS.

Durante la edición de 2017, y posterior a ella, se nos hizo llegar la petición para que se ofreciese en nuestra HoC un taller de introducción a Java, a pesar de que tuviese fuerte presencia en las asignaturas de las titulaciones de la Escuela debido a que podían venir asistentes de otros centros e instituciones y también a que muchos alumnos del centro optan por este tipo de talleres para reforzar su aprendizaje.

La introducción de Perl en la oferta de 2018 provino de un mensaje recibido por los organizadores pidiendo un taller de este lenguaje. Por el contrario, la propuesta de introducir Kotlin surgió de la propia organización para introducir una novedad. Sin embargo, antes de la fecha de cierre de inscripciones se cancelaron ambos talleres de la oferta por ausencia de inscritos. Estableciendo un número mínimo de inscritos para impartir el taller, nos aseguramos de no desperdiciar recursos humanos, de modo que los instructores de talleres que no tienen inscritos pueden unirse de apoyo en otros talleres más mayoritarios.

Los talleres avanzados ofertados a lo largo de 2016, 2017, 2018 se enumeran en el Cuadro 1. En esta última edición (2018) como novedad se introdujo un Taller conjunto Iniciación-Avanzado de "Introducción a Gherkin y automatización de pruebas para BDD" propuesto por un alumno que se ofreció voluntario a prepararlo y contó con 3 inscritos. Gherkin no es un lenguaje de programación por lo que no se incluye en los cuadros resumen.

5.2. Evolución de la demanda

En el año **2014** se inscribieron previamente 27 pero el día D se acercó mucha gente con curiosidad y participó sin estar inscrita previamente. En esa ocasión no registramos muchos datos. En el año **2015** se contó con el total de participantes sumando la asistencia a talleres de iniciación y a los avanzados. En este curso participaron un total de 108 asistentes. La forma en la que se guardaron los datos nos ha permitido recuperar los asistentes y sus correos electrónicos pero no los talleres a los que asistió cada uno. **A partir de 2016** ya se cuenta con los datos pormenorizados de participación por cada taller de iniciación y avanzado.

año-id	Lenguaje	Taller Avanzado
2015-1	Java	"Android: Aprende a desarrollar Apps Android Nativas usando Android Studio"
2015-2	JavaScript	"Aplicaciones móviles híbridas: Aprende a crear tu App híbrida con tecnologías web usando Ionic (AngularJS + Apache Cordova) y su servidor con ExpressJS (NodeJS)"
2015-3	JavaScript Php	"Aplicaciones web: Aprende a crear una web multiplataforma con responsive design usando Bootstrap + HTML5 + JavaScript + Php"
2016-1	Python	"Recogida y tratamiento de datos de la Web con Python, SQLAlchemy y Beautiful Soap"
2016-2	JavaScript	"Aplicaciones web y móviles independientes de plataforma basadas en Angular, Node JS y bootstrap"
2016-3	Go	"Concurrencia en Go"
2017-1	JavaScript	"Bots conversacionales con MSBotFramework y TypeScript"
2017-2	Go	"Cómo construir un chat en tiempo real en GoLang"
2017-3	Python	"Desarrollo basado en Flask para Python"
2017-4	Swift	"Introducción a Machine Learning con iOS"
2018-1	Java	"Programación de aplicaciones móviles para Android"
2018-2	JavaScript	"Desarrollo de aplicaciones web con Node.js"
2018-3	Python	"Desarrollo basado en Flask para Python"
2018-4	JavaScript	"Desarrollo de aplicaciones web con Vue framework"
2018-5	Python	"Desarrollo de Bots de Telegram con Python"

Cuadro 1: Talleres avanzados en las últimas tres ediciones clasificados por año y lenguaje de programación requerido.

El Cuadro 2 muestra un resumen de la participación en las ediciones de nuestra HoC. Se incluye una columna con el número de respuestas obtenidas a la encuesta enviada a inscritos en cada edición. En los Cuadros 3 y 4 se muestran respectivamente los datos desglosados con los que se cuenta (las últimas 3 ediciones) de las inscripciones solicitadas para los diferentes talleres.

	Iniciación	Avanzados	Ponentes	Total	Resp.
2014	?		7	?	1
2015	108		12	120	6
2016	72	26	13	111	14
2017	67	24	18	109	27
2018	89	28	21	138	40

Cuadro 2: Participantes en La Hora del Código por año con los datos disponibles y cantidad de respuestas obtenidas a la encuesta enviada a inscritos.

6. Resultados del estudio sobre la experiencia de enseñanza-aprendizaje (RQ2)

6.1. Perspectiva del asistente

Se han recibido un total de 52 respuestas. La mayor representación de asistentes se han obtenido para los cursos más recientes, 2018, 2017 y 2016 por ese orden. Sin embargo, de los asistentes a nuestra particular HoC del 2014 solamente se consiguió obtener una respuesta y 6 respuestas del 2015. En el Cuadro 2 se muestra la distribución por ediciones de los asistentes que responden a la encuesta. De éstos, algunos han repetido participación en diferentes ediciones: uno repite en 2015 y 2016; cuatro en 2016 y 2017; once repiten en 2017 y en 2018; cuatro repiten en 2016, 2017 y 2018; tres repiten en 2015, 2016, 2017 y 2018; y uno repite en 2014, 2015, 2016 y 2017. Es decir, 16 de 52 repiten

2 veces, 7 de los 52 repiten 3 veces y 4 de 52 repiten 4 veces su participación en nuestra HoC. Es decir, el 52 % de los que responden han repetido participación.

Previendo la posibilidad de aquellos que solamente habían participado en la última edición de 2018, se introdujo una pregunta para saber cuántos eran y si repetirían en 2019. De los 36 que solamente han participado en 2018, 29 dicen que sí participarían el 2019, 6 dicen que tal vez y solamente 1 dice que no participaría otra vez. Con estas cifras se concluye que la mayoría de los que participan repiten o tienen intención de repetir experiencia.

A los asistentes a los talleres de iniciación a un lenguaje de programación se les pregunta si la experiencia del aprendizaje de un lenguaje de programación en nuestra HoC resulta mejor que en clase. De los 52 preguntados, 48 fueron a los talleres de iniciación y 37 de ellos respondieron que la experiencia resultaba mejor que en clase, 4 peor y 7 igual, es decir que para más de un 77 % la experiencia fue mejor.

A los asistentes a los talleres avanzados de programación con frameworks/bibliotecas y plataformas se les pregunta si la experiencia del aprendizaje de un framework o plataforma en nuestra HoC resulta mejor que en clase. De los 52 encuestados, 23 participaron en los talleres avanzados y de ellos 17 respondieron que era mejor, 4 igual y 2 peor, es decir más de un 73 % respondió que la experiencia mejor.

A los asistentes que son estudiantes de la Escuela se le ha preguntado si les gustaría que hubiesen más experiencia de aprendizaje en las clases donde son otros estudiantes los que enseñan a sus compañeros. Eliminando 4 respuestas que corresponden a estudiantes de fuera de la Escuela, Las respuestas fueron que a 37 alumnos sí que les gustaría, a 10 les era indiferente y a 1 persona no le gustaría.

Por otra parte resulta interesante saber cómo de inspirador/motivador es el evento, si después de participar en los talleres como asistente se ha animado ya en al-

	C++	C#	Golang	Java	JavaScript	Kotlin	Perl	Php	Python	R	Ruby	Rust	Scala	Swift
2016	17	NO	8	NO	16	NO	NO	3	16	2	10	NO	NO	NO
2017	6	9	6	NO	5	NO	NO	7	10	NO	NO	5	2	2
2018	10	5	NO	14	15	CL	CL	8	14	7	NO	9	3	CL

Cuadro 3: Inscritos en los Talleres de Iniciación a Lenguajes de Programación con los datos disponibles. Se indica NO cuando no se ofertó y C cuando se canceló por falta de inscripciones.

	Android (Java)	X (Go)	Y (JavaScript)	Z (Python)	iOS (Swift)
2016	NO	2	13	11	NO
2017	15	8	4	10	2
2018	10	NO	6	7+5	CL

Cuadro 4: Inscritos en los Talleres Avanzados de Tecnologías (frameworks y plataformas) con los datos disponibles. X, Y y Z se refieren a tecnologías y entre paréntesis está el lenguaje de programación base (ver Cuadro 1).

guna edición a ser instructor de taller o si cree que se animaría en un futuro. En este caso, solo 15 han respondido afirmativamente, 19 respondieron que tal vez se animarían a ser instructor y 18 no querrían serlo.

Se ha preguntado: ¿Podrías aportar aspectos más positivos que ves en La Hora del Código que celebramos en la Escuela de Ingeniería Informática de Valladolid (UVa)?

Se han recogido 11 respuestas y se destaca:

- la amplia oferta de lenguajes y frameworks que hay
- la oportunidad que se brinda de aprender algo nuevo de programación de forma gratuita
- lo positivo de cómo el alumnado se involucra en el aprendizaje
- la actitud de los participantes, el compañerismo y la manera de llegar a gente fuera de clase
- la cercanía de los instructores así como sus respuestas rápidas y ejemplos autoexplicativos

Alguno de los que responde indica haber tenido una sensación de satisfacción e inspiración.

Se recogen también críticas y sugerencias. Los alumnos sugieren, por ejemplo, ofrecer un nivel intermedio entre los talleres de iniciación y avanzados, un taller más práctico que los de iniciación sin llegar a la complejidad de los avanzados. En cuanto a las críticas que se han recibido destaca el poco tiempo disponible para la actividad, proponiéndose que se dedique una mañana entera para que se pueda trabajar asimilando todo un poco mejor. También se comenta que para realizar el evento se podría haber elegido los laboratorios de la Escuela, criticándose por parte de un encuestado los problemas de infraestructura de red que hay que solventar y la acústica de la sala.

6.2. Perspectiva del instructor

En este caso, al igual que en el anterior apartado, hemos tenido muchas menos respuestas de las esperadas. Una de las razones ha sido porque sólo almacenamos las direcciones de correo institucionales y la mayor parte de los alumnos que han sido instructores

ya han finalizado sus estudios. La mayor parte de las respuestas corresponden a aquellos que han participado en las dos últimas ediciones.

Se pregunta en qué edición o ediciones ha participado como instructor de algún taller para, por una parte, observar a cuáles de los instructores hemos podido llegar. No hemos conseguido llegar bien a los de la edición de 2014 y a los de la de 2015. Por otra parte, con esa pregunta se puede establecer en el perfil del encuestado si se trata de los que han repetido experiencia o los que solamente han participado en una ocasión. De las 19 respuestas, 8 instructores han repetido en al menos dos ediciones y 11 solo han participado en una edición.

A los instructores se les ha consultado si habían participado como asistente en algún taller de alguna edición de La Hora del Código antes de ser instructor de taller. Es muy interesante saber que 8 instructores no habían participado, eso quiere decir que, los alumnos se animan a enseñar aunque no hayan participado nunca; además, los 11 restantes habían participado al menos en una ocasión.

Un aspecto que nos interesaba mucho analizar desde el punto de vista del estudiante que ha sido instructor de taller es la adquisición/desarrollo de *soft skills* [7] que permite este tipo de experiencia y si los estudiantes eran conscientes de ello. Con una respuesta mayoritaria, 18 instructores piensan que sí que les ha servido, y solo 1 piensa que le ha servido solo un poco. También se pregunta si en su caso particular considera que le ha ayudado y por último, en caso de respuesta positiva a la pregunta anterior, se pide que reflexionen y aporten, mediante escritura libre, algunas habilidades o destrezas que creen que ser instructor de taller en nuestra particular HoC les ha ayudado a adquirir o a desarrollar. De las 19 respuestas, 14 sí que lo consideran, 4 consideran que ha sido un poco y 1 no lo piensa.

De cara a extrapolar el interés despertado por este tipo de actividad de estudiante que enseña a estudiante, se ha consultado si les gustaría tener más oportunidades de enseñar a compañeros en el ámbito de las clases de los estudios. A 15 estudiantes les gustaría tener esa oportunidad, a 2 solo un poco y a otros 2 no les gusta-

ría.

Por otra parte, algo que nos ha parecido interesante y significativo es el perfil de estos estudiantes, que dan el paso a ser instructores de talleres que nos parecía vinculado con una tendencia al asociacionismo. Para corroborar este punto, se ha preguntado si antes o después de haber participado en La Hora del Código el estudiante se había unido a alguno de los grupos de la Escuela: Grupo Universitario de Informática de la UVa (GUI), Grupo UVaCoders de Programación Competitiva u otro. De ellos, la mayoría de las respuestas afirmativas, 17, correspondían a alumnos que ya estaban en una asociación y solo 2 no pertenecían a ninguna.

Por último, en ambos casos se pide una valoración global en una escala de 1 a 5 de su experiencia como asistente/instructor, obteniéndose de media una valoración de 4,38 y 4,42, respectivamente. Una valoración que consideramos muy positiva.

7. Lecciones aprendidas

Organizativamente, nuestra particular HoC supone un enorme esfuerzo cada año que solamente es posible gracias a un gran equipo estable en parte, pero que se consigue renovar cada año. Es necesaria la colaboración de asociaciones y agrupaciones estudiantiles, profesores, dirección del centro, técnicos de la Escuela así como el apoyo de la institución.

La semana internacional de Educación en Ciencias de la Computación se celebra tradicionalmente en diciembre, en unas fechas que, en nuestro entorno, coinciden con una habitual concentración de días festivos y no lectivos. Por ello, tanto dicha semana como la siguiente no son una buena opción. Por lo que se ha optado por celebrar nuestra HoC en la última semana del mes de Noviembre.

Asimismo, desde la Unión Europea en 2015 se asienta la Semana de la Programación de la UE, conocida como la EU Code Week, que se viene celebrando en Octubre. Debido a las características de nuestro evento donde son estudiantes los que imparten talleres a otros estudiantes, tampoco es posible trasladar nuestra HoC a Octubre por no tener suficiente tiempo para prepararlo.

La búsqueda de un horario en el que mayoritariamente puedan participar asistentes e instructores sin pérdida de sus clases, fundamentalmente en el caso de los alumnos de nuestro centro, no ha sido difícil gracias a un espacio de 2 horas de 12:00 a 14:00 todos los miércoles reservado para este tipo de actividades. Los mayores éxitos de participación se han dado cuando el horario ha incluido una parada intermedia para comer finalizando a las 17:00 como muy tarde.

El aprendizaje de los organizadores en las ediciones de 2014 y 2015, así como la necesidad de coordinar un equipo numeroso y variado de forma ágil nos llevó a la creación de un tablero Trello en el año 2016 que se ha ido mejorando en las ediciones posteriores. En

<https://trello.com/b/yEZZGid7> se pone a disposición el esquema de este tablero para ser replicado y tener una base de partida si se desea organizar un evento de estas características en otro entorno.

La primera edición de 2014 se realizó solamente para miembros de la Escuela de Ingeniería Informática. En el año 2015 se decidió abrirlo al resto de miembros de la UVa y se planteó el problema de comprobar si eran o no miembros UVa. Se decidió pedir el correo electrónico institucional como email de contacto y en caso de no ser así se rechazaba la inscripción. Esto incrementó el trabajo de gestión de inscripciones. En el siguiente año se pensó en brindar la oferta formativa para ámbitos externos a la UVa pero dando preferencia a miembros UVa. Esta labor fue más difícil aún porque se sumó la comprobación de pertenencia a la institución a la gestión de dos listas paralelas, la de los preferentes y el resto de inscritos. En la dos últimas ediciones se reafirmó la vocación de brindar la oferta formativa desde la Universidad hacia el entorno y se decidió no definir prioridades en la inscripción. Toda esta toma de decisiones la hacen los propios estudiantes que ayudan en la organización e imparten los talleres, moderada por los profesores que conducen la actividad.

Otro aspecto relevante ha estado relacionado por una parte con la protección de datos y simultáneamente con la pérdida de datos. Los primeros eventos de 2014 y de 2015 se organizaron de forma muy artesanal. En 2014 se realizó una pequeña aplicación web por los técnicos de la Escuela para inscribirse en el evento identificándose mediante el LDAP. El evento tuvo mayor asistencia que inscritos a través de esa aplicación. El día de celebración se acercaron y participaron muchos más y no hemos podido recomponer correctamente las cifras finales de participación. En 2015 se realizó la gestión mediante una conocida app para la gestión de eventos con la que tuvimos un problema para ajustarnos a las características particulares de nuestra HoC y el problema que representaba la protección y custodia de los datos. A partir del 2016 se cuenta con una instancia a nivel de toda la institución para la gestión de eventos lo cual nos ha permitido, por una parte, asumir correctamente la política de protección y custodia de datos de la institución así como, por otra parte, no perder datos para análisis y estudios posteriores.

Por último, y no por ello menos importante, se debe destacar la necesidad de cuidar activamente las buenas relaciones entre los diferentes estamentos del centro así como dedicar esfuerzos a diferentes formas de reconocimiento de la labor realizada. Los créditos de libre configuración son una forma de premiar de algún modo a los estudiantes que actúan como instructores de talleres pero, hay que ser consciente de que hay un límite en la cantidad de créditos de este tipo que se pueden acumular. Dado que muchos repiten de un año a otro, además de que suelen desarrollar un perfil que les hace participar en otras actividades de esta naturaleza, es in-

mediato llegar a que el estudiante no lo hace por eso, no es un incentivo. Es relevante entonces comprender por qué lo hace, y trabajar en incentivar y motivarlos en este sentido: realizar reconocimientos en público, emitir certificados que acreditan la impartición de talleres, escribir recomendaciones para los estudiantes que así lo piden para sus procesos de selección de cara al mercado laboral, motivar en la importancia de la adquisición y desarrollo de *soft skills* son las acciones realizadas en esta dirección.

8. Conclusiones y trabajo futuro

En respuesta a RQ1, los lenguajes con más estabilidad en la demanda en iniciación son JavaScript y C++ así como por Python (aunque este último forma parte de asignaturas en la formación reglada de la Escuela). Se aprecia el auge del interés por R y Scala. En el aspecto más avanzado en tecnología se reafirma el interés por plataformas y frameworks para JavaScript y Python, así como para el binomio Java+Android (aunque este último forma parte de asignaturas en la formación reglada de la Escuela)

En respuesta a RQ2, puede decirse que el planteamiento de tener como instructores a estudiantes ha sido una gran idea. Los instructores han profundizado en su conocimiento de los lenguajes o frameworks enseñando a sus propios compañeros y esta experiencia ha permitido que mejoren en sus destrezas o *soft skills*. La mayor parte de estos instructores formaban parte de una asociación de la Escuela, lo que enfatiza el valor de este tipo de grupos y asociaciones que se involucran de forma muy activa en todas las actividades de la Escuela.

La organización de un evento de este tipo en una Escuela de Ingeniería Informática es una experiencia satisfactoria. Se cuenta con estudiantes, profesores y PAS que año tras año se animan a participar como asistentes, instructores o miembros de la organización. En los últimos tres años han sido más de 380 personas entre todos. Los estudiantes que han asistido a los talleres se han dado cuenta del esfuerzo de los instructores, han participado activamente en los talleres y algunos de ellos se han convertido posteriormente en instructores. Han valorado positivamente estos talleres y tienen intención de repetir en los próximos años.

Como trabajo futuro se plantea realizar estudios sobre la motivación de los estudiantes que se presentan voluntarios como instructores así como de la adquisición de *soft skills* por parte de los mismos.

Agradecimientos

La preparación y despliegue de los entornos de desarrollo en la nube listos para usar en los talleres ha sido posible gracias a Javier Provecho, primero alumno y luego antiguo alumno de nuestra Escuela, y a Javier R. Aparicio, técnico del centro. En general mu-

chas gracias al Grupo Universitario de Informática de la UVA (GUI), al Grupo UVaCoders de Programación Competitiva, a la delegación de alumnos, los técnicos y la dirección de la Escuela de Ingeniería Informática de Valladolid (UVA), y a todos los que han hecho posible a lo largo de estos 5 años la realización de nuestra particular Hora del Código. Esta ponencia ha sido financiada por CompuEdu, y los PIDs UVA 38 y 39.

Referencias

- [1] R. Andersson and Bendix L. Extreme teaching - a framework for continuous improvement. *Computer Science Education*, 16(3), 2006.
- [2] L. Chang, C. Yang, M. Chiang, and G. Lee. Effectiveness of peer tutoring: Computer science students' perspective. In *FECS12: The 2012 International Conference on Frontiers in Education: Computer Science and Computer Engineering*, 2012.
- [3] J. Du, H. Wimmer, and R. Rada. Hour of Code: Can it change students' attitudes toward programming? *Journal of Information Technology Education: Innovations in Practice*, 15:53–73, 2016.
- [4] D. Duran. *Aprender a enseñar. Evidencias e implicaciones educativas de aprender enseñando*. Narcea, 2014.
- [5] N. A. Escherle, S. Ramirez-Ramirez, A. Basawapatna, D. Assaf, A. Repenning, C. Maiello, Y. Endo, and J. Nolasco-Flores. Piloting computer science education week in Mexico. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education, SIGCSE '16*, pages 431–436, 2016.
- [6] M. García García, M. Gaya López, and P. Velasco Quintana. Mentoría entre iguales: alumnos que comparten experiencias y aprendizaje. In *JENUI 2010. "XVI Jornadas de Enseñanza Universitaria de la Informática"*, pages 119–126, 2010.
- [7] B. Llamas, M. D. Storch de Gracia, L. F. Mazadiego, J. Pous, and J. Alonso. Assessing transversal competences as decisive for project management. *Thinking Skills and Creativity*, 31:125–137, 2019.
- [8] S. Nikou and A. Economides. Measuring student motivation during the "hour of code" activities. In *Proceedings - IEEE 14th International Conference on Advanced Learning Technologies, ICALT 2014*, 2014.
- [9] C. Wilson. Hour of Code: We can solve the diversity problem in computer science. *ACM Inroads*, 5(4):22–22, 2014.
- [10] C. Wilson. Hour of Code: Maryland, Washington and San Francisco move to support computer science. *ACM Inroads*, 6(3):14–14, 2015.