

Óscar Belmonte Fernández¹,
Carlos Granell Canut²

¹Departamento de Lenguajes y Sistemas Informáticos, Instituto de Nuevas Tecnologías de la Imagen, Universitat Jaume I (Castellón de la Plana); ²European Commission - Joint Research Center, Institute for Environment and Sustainability, Ispra (Italia)

<oscar.belmonte@uji.es>,
<carlos.granell@jrc.ec.europa.eu>

1. Introducción

Minimizar el desajuste entre lo que espera el mundo empresarial y las administraciones públicas frente a la formación que recibe el estudiante en la Universidad, pasa por conocer cual es la visión actual que tiene del programador el mundo empresarial y la administración pública frente al desarrollo de destrezas en las que se enfoca la Universidad para formar a sus estudiantes.

En la medida en que la Universidad sea capaz de reconocer qué es lo que se espera de los estudiantes cuyas destrezas forma, y la empresa y administraciones públicas sean capaces de transmitir estas necesidades, la discrepancia entre lo que se espera y lo que se encuentra se podrá ajustar hasta cierto punto.

Si no existe este diálogo, o si los supuestos por ambas partes son completamente o parcialmente erróneos, no se podrá avanzar en la satisfacción final de todas las partes: empresa-administraciones públicas, universidades y egresados universitarios.

El estudio que se presenta a continuación, intenta, en su modestia, arrojar una primera luz sobre este caso. Los datos de partida sobre los que se basa han sido obtenidos a través de una encuesta pública y difundida a través de *Novática*, en distintos foros sociales y a través de la propia página web de *Novática*.

2. Metodología y datos

La encuesta sobre lenguajes de programación estuvo disponible online en la web de *Novática* durante 5 semanas, desde el día 11 de marzo de 2013 hasta el día 14 de abril de 2013. El número de respuestas obtenidas al cierre de la consulta fue de 151. La encuesta contenía 10 preguntas, las cuales están disponibles en el **anexo 1**.

La lista de preguntas se divide en tres bloques bien diferenciados para intentar obtener alguna respuesta convincente a las siguientes preguntas: ¿Cómo se percibe el rol de la figura del programador? ¿Cómo se percibe la importancia de la figura del programador dentro de un proyecto? ¿Cuáles son las semejanzas y discrepancias en el uso de lenguajes de programación entre la empresa y universidad?

La importancia de la labor del programador. ¿Qué se espera? ¿Cómo se prepara? Análisis desde los lenguajes de programación

Resumen: A partir de los datos extraídos de una encuesta realizada a través de *Novática*, este trabajo pretende analizar, por un lado, los principales valores que perciben los diferentes actores profesionales (empresas, PYMEs, spin-offs, y Administraciones Públicas), con respecto a los valores que poseen los actores de formación (Universidades). Por otro lado, y desde el punto de vista de los lenguajes de programación, cuales son los lenguajes de programación más demandados profesionalmente, y cuales son los lenguajes de programación que se imparten o utilizan en las universidades españolas. Creemos que este análisis, tomado como una medida de la tendencia, puede servir para poner de manifiesto las posibles discrepancias que pueden existir entre el mundo profesional y la Universidad desde el punto de vista de la formación de profesionales con respecto a los lenguajes de programación. Las principales conclusiones del estudio son dos. Por un lado la principal visión que se tiene de un programador en el ámbito profesional es la de un técnico-integrador cuya labor es percibida con una importancia similar a la del resto de profesionales que participan en el desarrollo de proyectos informáticos. Por otro lado, los lenguajes de programación demandados en el mundo profesional se orientan hacia el desarrollo de aplicaciones web (Java, VB/.Net, JavaScript y PHP), mientras que en la Universidad la orientación es hacia los lenguajes de programación orientados a objetos, siendo Java y C++ son los que cuentan con un mayor peso específico.

Palabras clave: Ajuste universidad-empresa, importancia del programador, popularidad de los lenguajes de programación, rol del programador, tendencias de uso de lenguajes de programación.

Autores

Óscar Belmonte Fernández es Doctor en Ciencias Físicas por la *Universitat de València*, profesor titular en el departamento de Lenguajes y Sistemas Informático de la *Universitat Jaume I* de Castellón y miembro activo en el Instituto de Nuevas Tecnologías de la Imagen (INIT) donde desarrolla actualmente su investigación dentro del grupo de Sistemas de Información Geográfica, Geoinfo. Ha participado en distintos proyectos de investigación europeos, así como ha dirigido proyectos en el ámbito nacional y regional. Su línea actual de investigación principal se centra en la Internet de las cosas (IoT), y las redes de sensores. Anteriormente, su principal línea de investigación se centraba en la Informática Gráfica en tiempo real. Es coordinador de la sección técnica "Lenguajes de Programación" de *Novática*.

Carlos Granell Canut se licenció como Ingeniero en Informática por la *Universitat Jaume I* de Castellón en 2000 y recibió el grado de doctor por la misma Universidad en 2006. Durante diez años ha ocupado diversos puestos como investigador en el departamento de Lenguajes y Sistemas Informáticos y en el Instituto de Nuevas Tecnologías de la Imagen (INIT). Desde principios de 2011, se encuentra como investigador postdoctoral en la unidad *Digital Earth and Reference Data del European Commission - Joint Research Center* (EC-JRC), Ispra (Italia). Sus principales líneas de investigación tienen que ver con la aplicación de las tecnologías de información geográfica en el desarrollo de software para plataformas Web que involucran la composición y reutilización de servicios Web, modelos y servicios de geo-procesamiento. Ha participado en diversos proyectos de investigación con fondos públicos, de los cuales destacan los proyectos nacionales España Virtual y los europeos ACE-GIS, AWARE, EuroGEOSS y ENVIROFI. Además, ha realizado diversas estancias de investigación en centros europeos como SINTEF (Noruega), la *Faculty of Geo-Information Science and Earth Observation* de la Universidad de Twente (Holanda) y el *Center for Geospatial Science* de la Universidad de Nottingham (Reino Unido).

La tipología de los encuestados aproximadamente corresponde un 77,5% a la empresa y 22,5% a la universidad. Destaca sobre todo el gran número de respuestas por parte de profesionales de las tecnologías de la información e Informática por parte de la empresa. En particular, la parte izquierda de la **figura 1**

muestra el perfil de la empresa compuesto de gran empresa, pyme y empresas *spin-off*, y Administración Pública. Para los análisis posteriores de los datos, hemos unificado el perfil pyme y *spin-off* puesto que la mayoría de encuestados marcaron ambas respuestas a la vez.

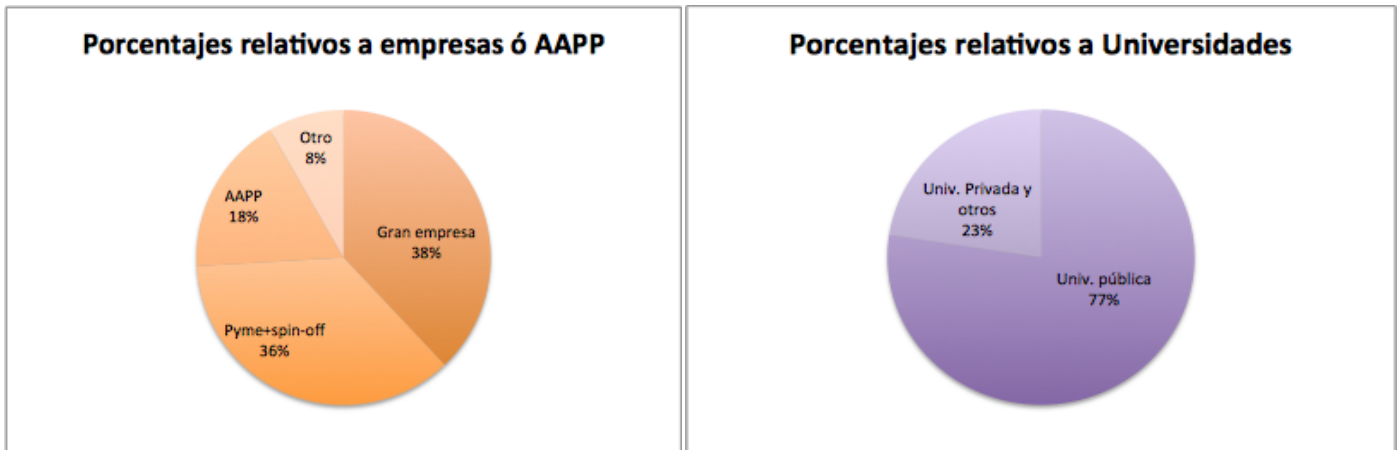


Figura 1. Tipología de los encuestados de la empresa (izquierda) y de la universidad (derecha).

La parte derecha de la **figura 1** muestra la tipología de la universidad, que representa el 22,5 % del total de encuestados. La gran mayoría de los encuestados pertenece a la universidad pública, mientras que el resto engloba encuestados de la universidad privados y otras instituciones de investigación.

3. Resultados¹

3.1. Percepción de la figura del programador

La **pregunta 1** de la encuesta trata de averiguar la etiqueta que mejor define la función de un programador hoy en día. La **figura 2** nos da una primera aproximación de la percepción global de la función del programador atendiendo a la respuesta global de los encuestados.

Se observa que el rol de "técnico" es la perspectiva de un programador que destaca sobre el resto de opciones, posiblemente influenciado por la proliferación de las metodologías de

desarrollo de software como *Scrum*, ya que pone de manifiesto el uso de metodologías o pautas de actuación en proyectos de desarrollo de software. La función de "técnico" viene seguida de cerca por la de "integrador", lo cual también parece lógico dado el entorno actual de desarrollo basado en APIs de distintos proveedores para crear nuevos servicios o productos integrados al estilo de los "mashups". En el lado contrario, la etiqueta de "creativo" no es percibida como la más adecuada para describir las tareas de un programador.

Cabe mencionar que la denominación de "codificador", mucho más tradicional, aún está presente de forma significativa, aunque pierde peso a favor del dúo "técnico-integrador".

En el material suplementario (ver **nota 1**) proporcionamos una serie de gráficas adicionales atendiendo al perfil de los encuestados

cuyo texto descriptivo incluimos a continuación. La serie de figuras 2a, 2b y 2c desgrana por cada perfil mayoritario de la empresa, es decir "Gran empresa", "Pyme + *spin-off*", y "AAPP" los resultados a la **pregunta 1**. De forma resumida, la gran empresa ve al programador como un integrador con dotes de creatividad (figura 2a), la pyme y spin-off como un técnico con cierto toque integrador (figura 2b), y la administración pública como un técnico que incluye tareas de codificador (figura 2c).

La figura 2d del suplemento muestra los resultados de la **pregunta 1** únicamente para el encuestado de "Univ. Pública" ya que el bajo número de respuestas obtenidas para "Univ. Privada" hace no significativo el análisis pormenorizado para este perfil. Curiosamente, la universidad pública parece que no se pone de acuerdo en definir de forma consensuada la figura de un programador, lo cual no deja de ser significado. Coincide en

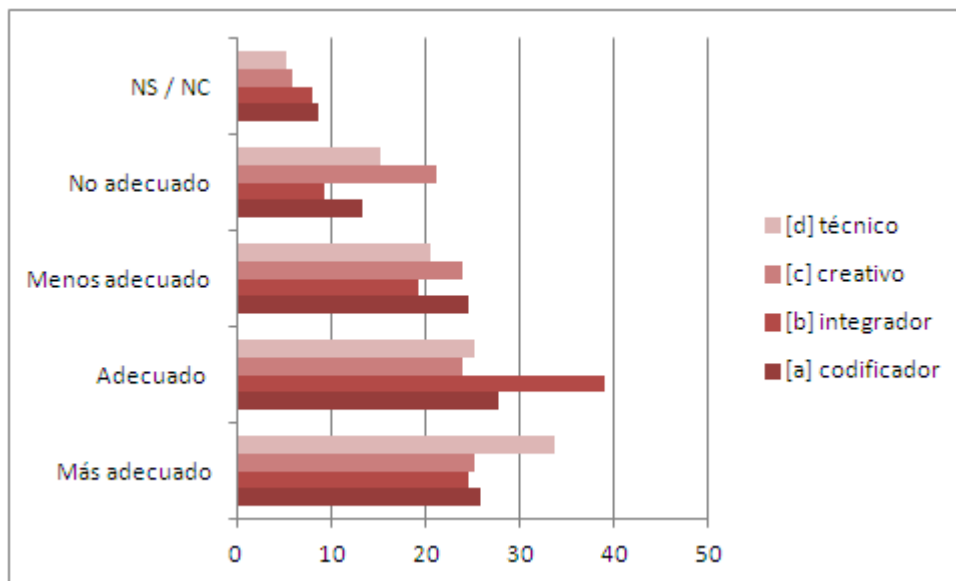


Figura 2. Porcentaje sobre el total de respuestas de cada una de las opciones posibles de la **pregunta 1**: "¿Cuál de las siguientes frases describe mejor la función del programador hoy en día?"

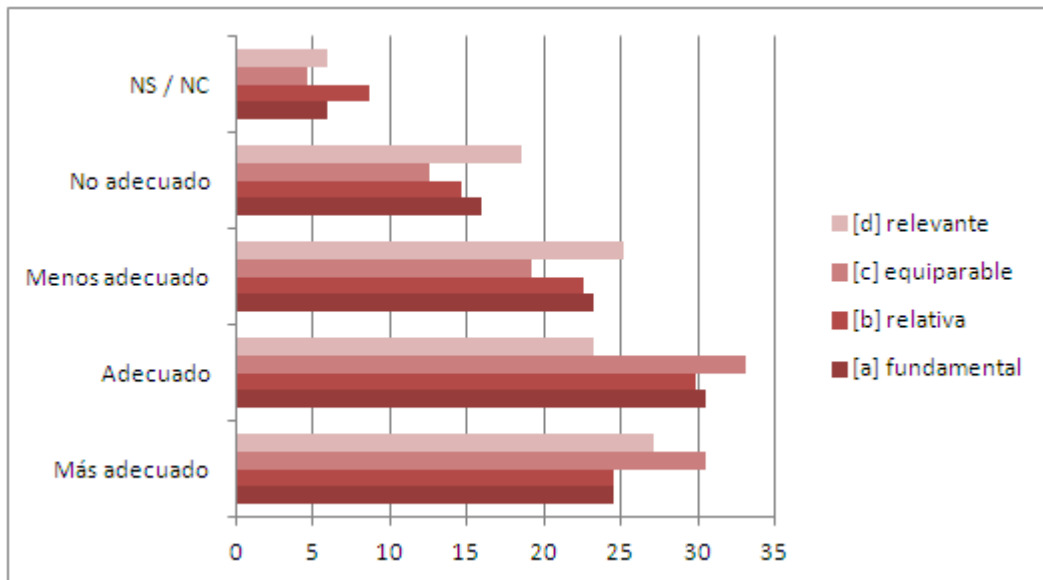


Figura 3. Porcentaje sobre el total de respuestas para cada una de las opciones posibles de la pregunta 2: "La importancia del trabajo del programador en un proyecto".

líneas generales con la visión de la pyme y *spin-off* (figura 2b) y en la percepción global de los encuestados (ver figura 2) en percibir la función del programador como un técnico con cierto toque de integrador.

La serie de figuras 2e, 2f, 2g y 2h muestra el nivel de acuerdo/desacuerdo para cada uno de los tipos de perfiles según el grupo al que pertenece el encuestado para la pregunta 1. Nos encontramos con lo siguiente: el perfil de codificador es el que se percibe como adecuado o más adecuado casi con independencia del grupo que contesta tal (figura 2e). Sobresale el caso de la Administración pública, que percibe como adecuado el perfil de codificador pero en absoluto como el más adecuado.

Con respecto al papel del programador como un integrador, los datos no reflejan una tendencia clara (figura 2f), aunque si atendemos tan sólo al total normalizado, la opinión mayoritaria entre los distintos grupos es que es el perfil que más se adecua a la labor de un programador.

El siguiente perfil es el de creativo, donde sí que se observa una clara tendencia, para la distintos grupos de encuestados y sobre el total normalizado, hacia el papel más adecuado, con la excepción destacada de la Universidad pública que ve este perfil del programador menos adecuado, frente a adecuado y más adecuado (figura 2g).

El último perfil considerado es el de técnico. De nuevo no se encuentra una tendencia clara en los datos de la figura 2h, si atendemos a los datos totales normalizados, aunque sí que hay que destacar el bajo porcentaje de las respuestas del grupo de las Administraciones Públicas para este tipo de perfil como el más adecuado.

3.2. Importancia del trabajo del programador

La pregunta 2 de la encuesta plantea cual es la percepción de la importancia del trabajo del programador en el contexto de un proyecto determinado. En la figura 3 se observa, sobre el total de encuestados, un cierto grado de importancia de la figura del programador en igualdad de condiciones a otros profesionales que forman parte de un mismo proyecto, en detrimento tanto de un rol excesivamente prominente (respuestas [a] y [d]) o superficial y secundario (respuesta [b]) para la consecución de un proyecto.

Esto puede deberse también al despliegue de metodologías de desarrollo ágil que sitúan al desarrollador o programador al mismo nivel que otras funciones y/o perfiles profesionales en un proyecto, enfatizando no tanto la jerarquía vertical sino la colaboración horizontal entre perfiles para llevar a cabo de forma conjunta un proyecto en un tiempo determinado.

La serie de figuras 3a, 3b y 3c muestra cómo se comporta cada perfil de la empresa ("Gran empresa", "Pyme + spin-off", y "Administración pública") con respecto a la pregunta 2 de la encuesta. De forma resumida, la gran empresa sitúa mayoritariamente al programador a un nivel parecido al resto de tareas (figura 3a), donde "equiparable" destaca como la respuesta adecuada y más adecuada. La pyme y *spin-off* consideran fundamental el puesto del programador puesto que justifica e implementa todo el trabajo anterior ante el usuario final (figura 3b). Para el caso de la administración pública, a diferencia de la gran empresa y pymes/*spin-off*, la importancia del trabajo del programador queda relegada a un segundo plano y, en consecuencia, queda un tanto diluido su papel dentro de un proyecto (figura 3c).

La figura 3d, del material suplementario, muestra los resultados de la pregunta 2 únicamente para el encuestado de "Univ. pública". En este caso la universidad diverge con respecto a la empresa, ya que la importancia que otorga al trabajo del programador es menor que la que otorga la empresa en general, situándolo por detrás del trabajo de otros perfiles de un proyecto como el analista y el arquitecto del sistema. Aquí, parece que se mantiene la jerarquía tradicional tal como ilustran los libros de referencia de ingeniería del software y desarrollo de sistemas. En este sentido, la visión de la universidad pública se asemeja con la perspectiva de la administración pública (figura 3c) e incluso incide aún más en considerar la importancia del trabajo del programador como algo relegado a un segundo plano.

La serie de gráficas 3e, 3f, 3g y 3h del material suplementario muestra el nivel de acuerdo/desacuerdo para cada uno de los tipos de perfiles según el grupo al que pertenece el encuestado para la pregunta 2.

En la figura 3e, se muestran los datos agrupados por el grado de importancia del programador en un proyecto obteniendo los siguientes resultados: el papel "fundamental" del programador dentro del desarrollo de un proyecto es considerado como Adecuado o Más adecuado si atendemos al total normalizado. De nuevo, vuelve a destacar la visión de la Universidad pública con respecto al papel del programador.

Con respecto a la consideración de importancia "relativa" del trabajo del programador en un proyecto, por detrás de la importancia del analista funcional, el diseñador técnico y/o el arquitecto del sistema, no existe una tendencia clara, aunque sí que es esta posición la que

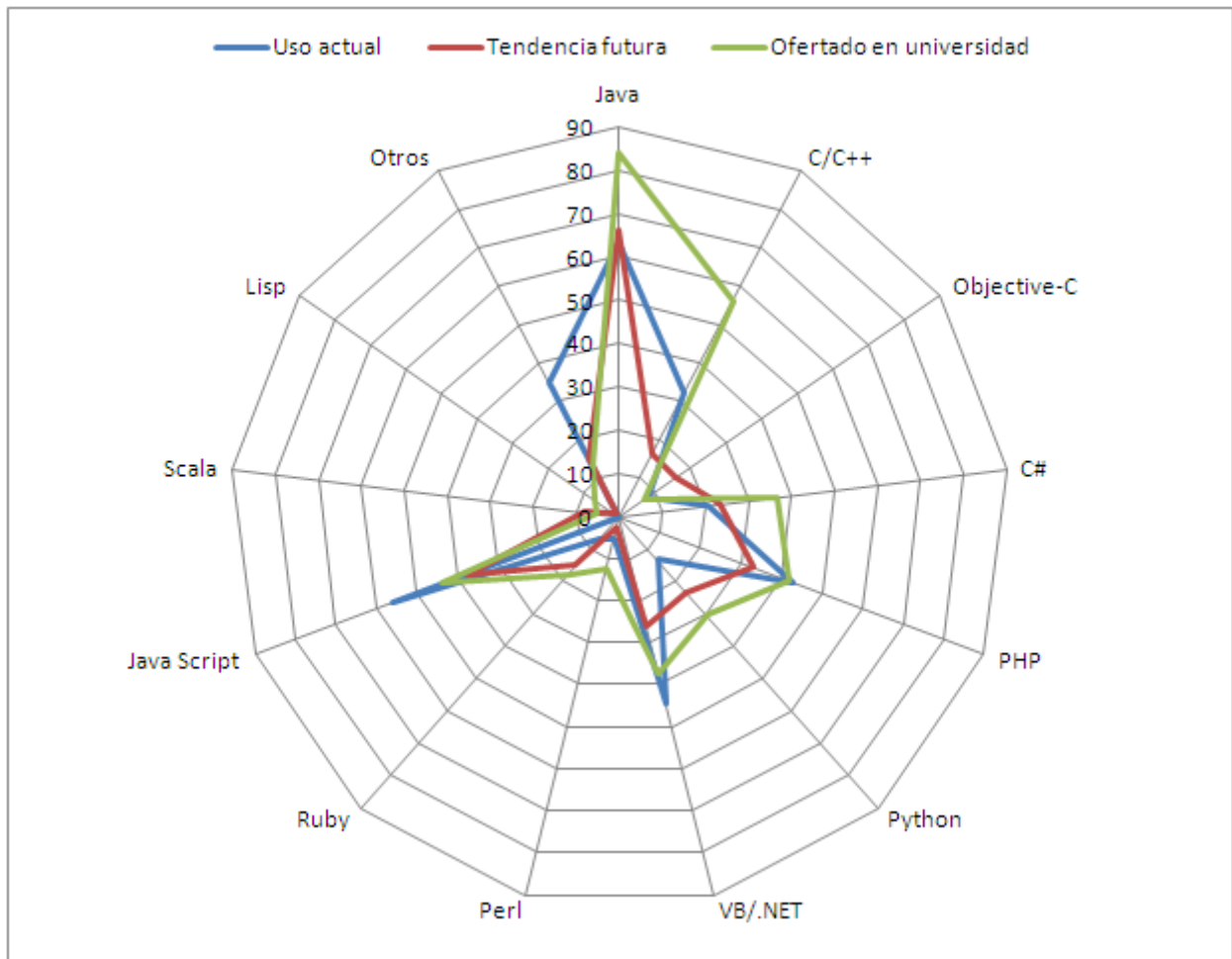


Figura 4. Porcentajes sobre las respuestas a las preguntas 4, 5 y 6: "Uso actual", "Tendencia futura" y "debería ofertarse en la universidad", respectivamente. La suma de los porcentajes es superior al 100% debido a respuestas múltiples.

tiene una mayoría si se atiende a los totales normalizados (figura 3f).

Si atendemos a los totales, se observa una suave tendencia a considerar el trabajo del programador dentro de un proyecto al mismo nivel que el resto de tareas de un proyecto (figura 3g).

Finalmente, no hay una clara tendencia a que el trabajo del programador en un proyecto sea considerado como relevante (figura 3h). Aunque sí que destaca la opinión de "Más adecuado" a este grado de importancia que da la Gran empresa y en particular las PYMEs y *spin-off*, frente a la opinión de la Universidad pública que lo considera "Menos adecuado".

3.3. Uso y tendencias de los lenguajes de programación desde la empresa

El segundo bloque de preguntas de la encuesta (preguntas 4-6) se dirige directamente al colectivo de la empresa (parte izquierda de la figura 1), con la intención de arrojar algo de luz sobre los lenguajes de programación utilizados hoy en día en proyectos empresariales, cuál será la tendencia en un futuro no muy lejano, y qué lenguajes de programación de-

berían ser ofertados y estudiados en la universidad para mejorar la preparación de los futuros informáticos, siempre desde la óptica de la muestra de encuestado pertenecientes a la empresa.

La gráfica de radar en la figura 4 nos da unas primeras pistas de los lenguajes de programación más influyentes según el colectivo de empresa sobre estas tres tendencias: "uso actual", "tendencia futura", "ofertados en universidad". Los picos más pronunciados, tanto en el uso actual como a corto plazo, se observan con los dúos Java - VB/.Net y JavaScript - PHP, que forman el núcleo duro de lenguajes para el desarrollo de aplicaciones empresariales, siendo Java y JavaScript los predilectos actualmente.

El caso de C/C++ y Python es interesante puesto que su evolución a corto plazo sigue caminos opuestos. Mientras C/C++ se desinfla a corto plazo, Python parece que cobrará más importancia en aplicaciones futuras. El resto de lenguajes, tanto funcionales, lógicos y declarativos como C# y Objective-C, no resultan significativos ahora ni a corto plazo según los datos de la encuesta.

Cabe mencionar por último el uso de otros lenguajes de programación no recogidos en la lista de posibles respuestas pero sí mencionados por los encuestados a través de la opción "Otro", y que parecen aún tener un rol importante en el mundo empresarial. Destacan en esta categoría Cobol, Pl/sql y Html/Css (uso actual), Cobol y Html5 (tendencia futura), y Cobol y Pascal (ofertado en universidad).

La serie de figuras 5, 6 y 7 analiza el uso y tendencia de los lenguajes de programación atendiendo al perfil de empresa de los encuestados.

Considerando el perfil de "Gran empresa" (ver figura 5), los resultados se asemejan al total de encuestados del grupo empresa (figura 4), salvo que el uso de Php queda bastante diluido.

Los datos obtenidos del perfil "Pyme + *spin-off*" son más variopintos (ver figura 6). Aunque mantiene la misma sintonía en cuanto al núcleo duro de lenguajes de programación (Java, VB/.Net, JavaScript, Php), irrumpen con fuerza un par de lenguajes que pasan casi desapercibidos por el resto de

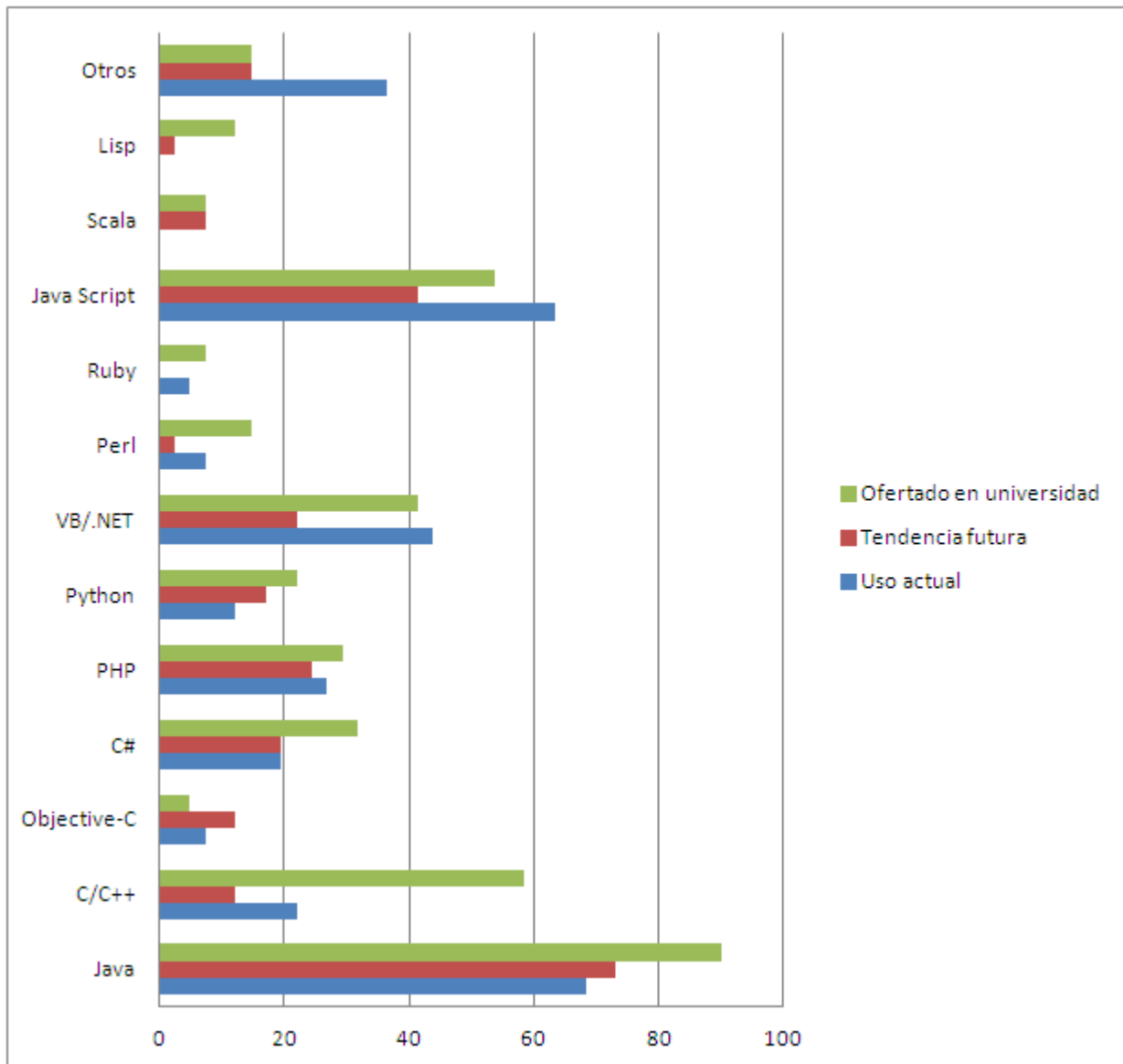


Figura 5. Uso, tendencia futura, y necesidad de ofertarse en la universidad por parte del perfil Gran empresa. La suma de los porcentajes es superior al 100% debido a respuestas múltiples.

perfiles de empresa. El primero es C#, orientado a la creación de aplicaciones basadas en la plataforma .Net de Microsoft. El segundo es Objective-C, el cual se utiliza para el desarrollo de aplicaciones para dispositivos móviles con sistema operativo iOS de Apple. El auge de Objective-C en realidad no sorprende por la tendencia actual al desarrollo de aplicaciones sobre plataformas móviles, un producto casi necesario para cualquier empresa pyme o *spin-off* dedicado a las tecnologías de la información.

En cuanto a la "Administración pública" (ver figura 7), destaca el bajo porcentaje de uso de VB/.Net y JavaScript en comparación con los resultados arrojados por el total de encuestados de la empresa (figura 4). Destaca en este perfil el repunte de Python como lenguaje que se utiliza actualmente, el cual se percibe por los tres perfiles empresariales

como un lenguaje prometedor y al alza en un futuro.

3.4. Uso y tendencia de los lenguajes de programación desde la universidad

El segundo bloque de preguntas de la encuesta (preguntas 8-10) se dirige al colectivo de la universidad pública (parte derecha de la figura 1).

La figura 8 desglosa los lenguajes de programación que se imparten actualmente como docencia reglada en la universidad pública, cuáles de ellos se recomendarían a futuros estudiantes, y cuáles se perciben como los más demandados por la empresa, siempre desde la perspectiva de la universidad pública.

A diferencia del grupo de empresa, que identifica un núcleo de 4 lenguajes, los encuestados de la

universidad pública claramente se decantan por la docencia de Java y C/C++, relegando JavaScript a una tercera posición muy rezagada con respecto a los dos primeros.

Un detalle significativo tras comparar las figuras 4 y 8 es la coincidencia entre los lenguajes utilizados por la empresa y los que la universidad pública piensa que demanda la empresa.

Otra vez, el núcleo formado por Java, VB/.Net, JavaScript y Php encaja en ambos colectivos, siendo Java elegido por unanimidad de los encuestados de la universidad pública.

No se obtiene el mismo grado de concordancia en el caso contrario, es decir, los lenguajes que deberían ofertarse en los planes de estudios según la empresa y los que realmente se ofrecen en la universidad. Aquí, los empresa-

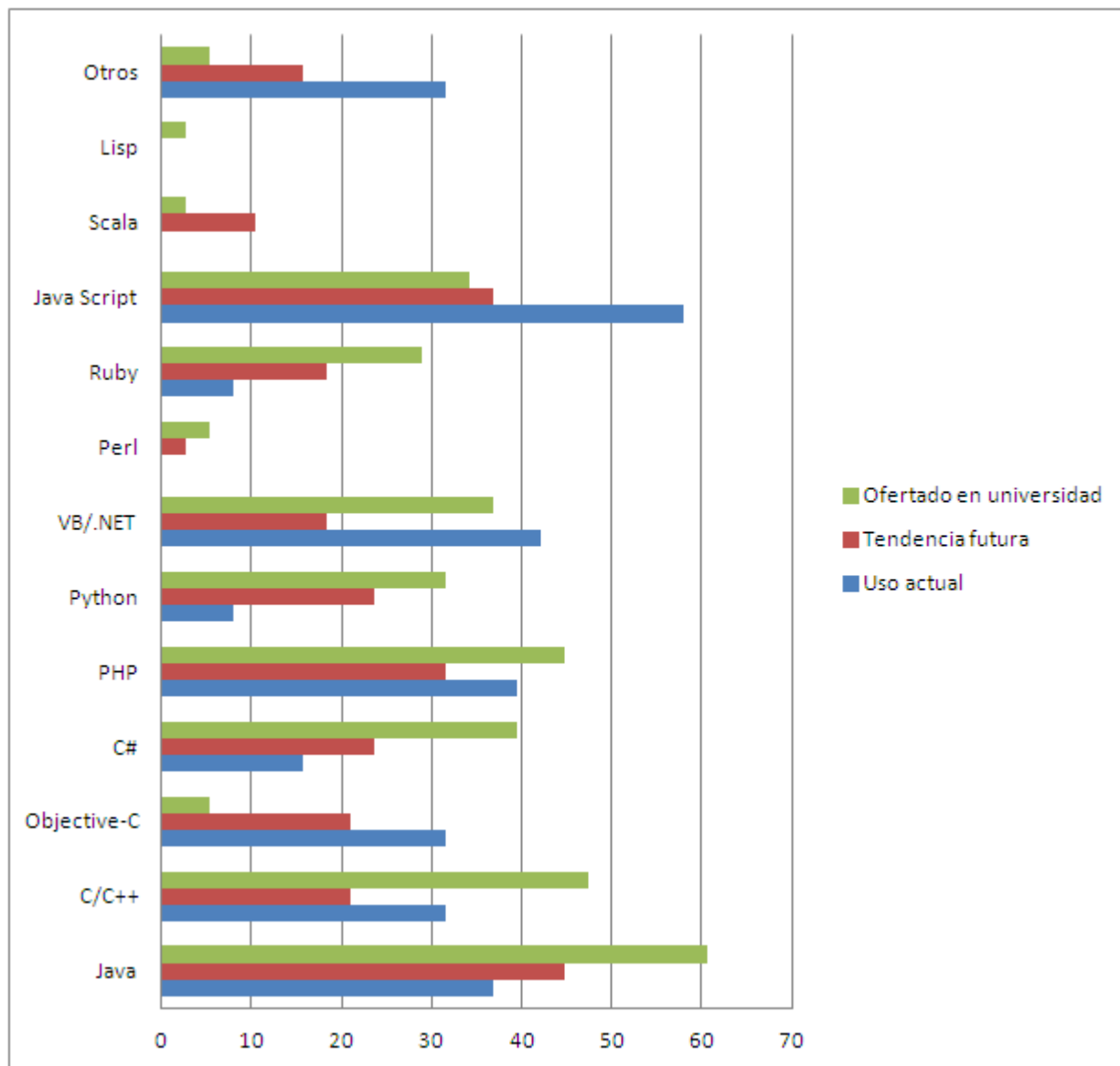


Figura 6. Uso, tendencia futura, y necesidad de ofertarse en la universidad por parte del perfil Pyme y *spin-off*. La suma de los porcentajes es superior al 100% debido a respuestas múltiples.

rios echan de menos más docencia en VB/.Net, Php y C#. Curiosamente, Objective-C no aparece reflejado en la docencia actual ni como petición para incluirse en planes docentes en el futuro.

4. Comentarios finales

Según los resultados mostrados por nuestra encuesta, con respecto a la percepción de la figura del programador hoy en día (**pregunta 1**), gana la faceta "técnico-integrador" de forma global. Con respecto a la importancia del programador dentro de un equipo, prevalece la visión de equiparable con otros roles del proyecto.

Esto sitúa a la figura del programador como un profesional que migra desde una visión de codificador a un técnico-integrador y que trata de "tú a tú" y como iguales a otros roles

profesionales que típicamente han prevalecido sobre la tarea del programador. En general se trata de un cambio de tendencia en las relaciones entre los miembros de un grupo, donde se prima la colaboración y la interacción horizontal entre sus miembros en aras de conseguir el éxito del proyecto.

Con respecto a qué lenguajes se utilizan actualmente en proyectos de desarrollo, las empresas se decantan sin ninguna duda por el cuarteto Java, VB/.Net, JavaScript y Php, con ligeras diferencias cuando se analizan los datos por perfiles empresarial. Los reyes indiscutibles en la docencia universitaria son Java y C/C++. Aquí sí se detecta un desfase entre lo que se oferta actualmente en las universidades públicas y lo que se demanda en la empresa. Finalmente, las familias de lenguajes funcionales y lógicos [1][2]) tienen

poca acogida tanto en la empresa como en la docencia universitaria y no se percibe un cambio de tendencia en el futuro.

Resulta extremadamente difícil estimar el nivel de popularidad de los distintos lenguajes de programación de manera automática. La empresa Tiobe Software [3] utiliza como índice de popularidad el número de páginas web dedicadas a cada lenguaje de programación, encontradas por distintos buscadores web. Por su parte, en la página web PYPL *Popularity of Programming Language index* [4] se utilizan como índice de popularidad las búsquedas realizadas sobre manuales de cada lenguaje de programación analizado.

Finalmente, la herramienta que proporciona *The Transparent Language Popularity Index* [5] permite automatizar el proceso de crea-

visiones sobre Lenguajes de Programación

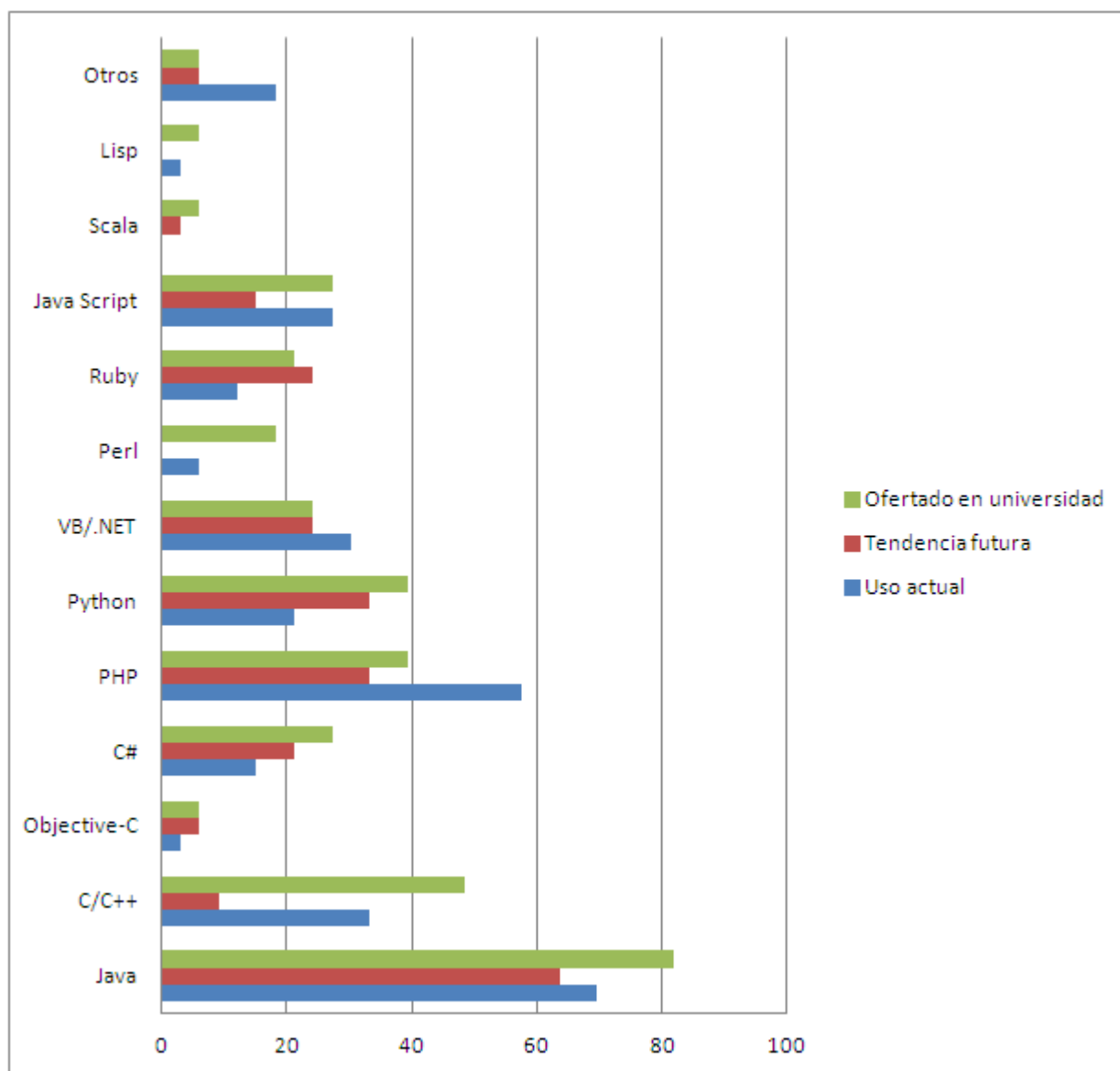


Figura 7. Uso, tendencia futura, y necesidad de ofertarse en la universidad por parte del perfil Administración Pública. La suma de los porcentajes es superior al 100% debido a respuestas múltiples.

ción de estos índices. Como resumen, si atendemos a los lenguajes que ocupan los primeros puestos, prácticamente encontramos los mismos lenguajes de programación que han predominado en nuestro estudio: C, Java, Objective-C, Php, C#, Python, Javascript. Como conclusión final en este caso, y restringiendo el análisis a este conjunto de lenguajes de programación, es claro que la demanda por parte de empresas, PYMES-spin-offs, y AAPP, coincide con estos índices de popularidad de ámbito mundial.

Volviendo a la herramienta mencionada, la tendencia a largo plazo del uso de lenguajes de programación en el desarrollo de proyectos informáticos cubre desde el año 2002 hasta el presente. De las gráficas que se publican en su página web se pueden observar varias tendencias a lo largo de estos años: por un lado que inicialmente existían dos grandes grupos, el

formado por los lenguajes de programación Java, C y C++ y el formado por el resto de lenguajes de programación. Dentro del primer grupo destaca la caída en el uso del lenguaje de programación C++ a partir del año 2004, lo cual se refleja también por nuestros encuestados pertenecientes a la empresa. Por otra parte, y dentro de este primer grupo, también se observa una ligera tendencia a la baja tanto de C como sobre todo de Java.

En el grupo de "otros lenguajes de programación", destaca la fuerte subida, desde el año 2009 del lenguaje de programación Objective-C, que recordamos es el utilizado para el desarrollo de aplicaciones para dispositivos móviles con sistema operativo iOS de Apple, lo cual no se refleja en los datos de la encuesta llevada a cabo, con la significativa excepción de las empresas pymes y *spin-off*.

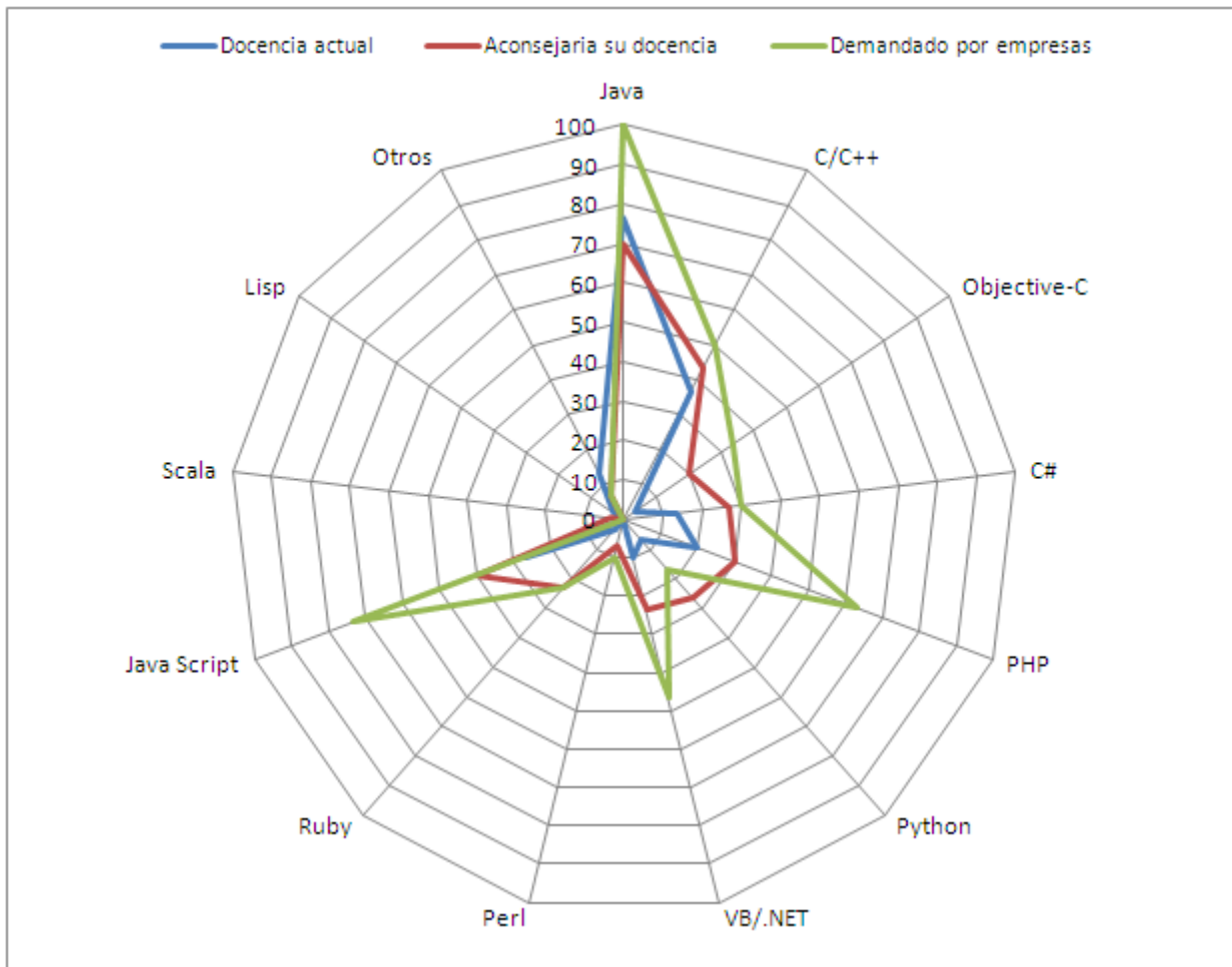


Figura 8. Porcentajes de las preguntas 8 (Docencia actual), 9 (Aconsejaría su docencia) y 10 (demandado por la empresa) por parte del perfil universidad pública. La suma de los porcentajes es superior al 100% debido a respuestas múltiples.

Referencias

[1] R. Peña. Los lenguajes de programación en perspectiva. *Novática* num. 222, págs. 7-11, 2013.
 [2] M. Montenegro. La programación funcional. *Novática* num. 222, págs. 14-19, 2013.
 [3] TIOBE Software. <<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>>.
 [4] pyDatalog. *PYPL Popularity of Programming Language index*. <<https://sites.google.com/site/pydatalog/pypl/PyPL-Popularity-of-Programming-Language>>.
 [5] The Transparent Language Popularity Index. <<http://lang-index.sourceforge.net/#grid>>.

Nota

¹ Por restricciones de espacio, no hemos podido incluir en el PDF de este número de *Novática* todas las gráficas generadas para dar soporte a nuestras conclusiones. Por ello, hemos añadido en nuestra web un PDF suplementario con gráficas adicionales de los datos presentados y discutidos en esta sección. Las figuras numeradas con letras (a, b, c, ...) se incluyen en dicho análisis suplementario: <<http://www.ati.es/novatica/2013/223/Nv222-Suplemento-Encuesta-LP.pdf>>.

visiones sobre Lenguajes de Programación

Anexo 1: Preguntas contenidas en la encuesta

Pregunta 1. ¿Cuál de las siguientes frases describe mejor la función del programador hoy en día? (Puntúe de 2 a 5 cada uno de los enunciados según su adecuación al momento actual, siendo 1 No sabe / No contesta, y 2 la más adecuada).

[a. Es un **codificador** cuya misión es reescribir una especificaciones expresadas en lenguaje natural en un determinado lenguaje de programación.]

[b. Es un **integrador** cuya función principal es la orquestación de llamadas a *funciones que vienen implementadas a través de frameworks, librerías de funciones, servicios web, etc.*]

[c. Es un **creativo** cuya misión es transformar una especificación escrita en lenguaje natural en una nueva especificación (comprensible para un compilador) bajo criterios de eficacia y eficiencia en cuya aplicación sus experiencias previas y su talento natural resultan fundamentales.]

[d. Es un **técnico** que, a partir de unos requisitos previos, obtiene unos programas o módulos de ejecución, siguiendo unas metodologías o pautas de actuación que son comunes con las de otros profesionales que participan en proyectos afines.]

Pregunta 2. La importancia del trabajo del programador en un proyecto es: (Puntúe de 2 a 5 cada uno de los enunciados según su adecuación al momento actual, siendo 1 No sabe / No contesta, y 2 la más adecuada).

[a. Es **fundamental** puesto que justifica e implementa todo el trabajo anterior ante el usuario final. Si la programación no resulta adecuada y sólida el "edificio" entero del proyecto corre peligro de desmoronarse.]

[b. Es **relativa** situándose por detrás de la importancia del trabajo realizado por el analista funcional, el diseñador técnico y/o el arquitecto del sistema.]

[c. Se sitúa a un **nivel parecido** del resto de tareas puesto que ese trabajo se integra con el realizado en las etapas anteriores permitiendo así corregir defectos de concepción o malos entendidos con el usuario final.]

[d. Es **relevante** puesto que la solidez de los algoritmos y soluciones diseñadas durante la fase de programación es un componente importante del buen funcionamiento del producto final.]

Pregunta 3. ¿En qué tipo de empresa trabaja?

- a. Gran empresa.
- b. Pyme.
- c. Spin-off.
- d. Administración pública.
- e. Otro

Pregunta 4. ¿Con que LP (uno o varios) trabaja actualmente?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro

Pregunta 5. ¿Con qué LP tendería a trabajar a partir de ahora por considerarlos con mejor futuro?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro

Pregunta 6. ¿Que LP (uno o varios) cree que deberían estudiarse en la Universidad para que el estudiante llegue mejor preparado a la empresa?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro

Pregunta 7. Si no trabaja en la Universidad, finalizar la encuesta 7. ¿En qué tipo de universidad trabaja?

- a. Pública.
- b. Privada.
- c. OPI.
- d. Otras Instituciones de investigación.

Pregunta 8. ¿Qué LP (uno o varios) enseña o a cuales se aplican mejor a las materias que enseña?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro

Pregunta 9. ¿Que LP (uno o varios) aconsejaría aprender (o profundizar en sus características) a sus alumnos?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro

Pregunta 10. ¿Qué LP (uno o varios) cree que demandan más las empresas, o están más en consonancia con las necesidades del mercado, actualmente?

Java; C/C++; Objective-C; C#; PHP; Python; VB/.NET; Perl; Ruby; JavaScript; Scala; Lisp; Otro