

Recepción: 11/10/2012
Aprobación: 26/11/2012

LA CONSTRUCCIÓN DE DIDÁCTICAS PARA LAS CIENCIAS DE LA COMPUTACIÓN: INTERCULTURALIDAD Y GLOBALIZACIÓN

Jesús Insuasti
Universidad de Nariño

RESUMEN

Mientras los productos de las Ciencias de la computación se generan en corto tiempo dado el desarrollo de la ciencia *per se*, las reflexiones en cuanto a la enseñanza de dichas ciencias se han caracterizado por desarrollarse a largo plazo. Este artículo realiza una exploración de corte intercultural sobre experiencias interesantes en materia de la enseñanza en las Ciencias de la computación en el caso específico de la didáctica del diseño de software en diferentes partes del planeta. Tras dicho estudio previo, realizado a través de una investigación cualitativa en el marco de la docencia universitaria, el artículo establece los parámetros de reflexión al tenor de los procesos de globalización, brindando algunas ideas para propuestas futuras en materia de didácticas específicas en el campo, con un pensamiento crítico. Se concluye que el aporte de las líneas de pensamiento crítico contribuye a la contextualización de las didácticas inmersas en procesos de globalización.

Palabras clave: intercultural, globalización, didácticas, enseñanza, Ciencias de la computación.

THE CONSTRUCTION OF TEACHING FOR COMPUTER SCIENCES: INTERCULTURALITY AND GLOBALIZATION

Jesús Insuasti
University of Nariño

ABSTRACT

While the products of computer science have been generated in a short time, given the development of science *per se*, the reflections concerning the teaching of these sciences have been characterized by long-term development. This article provides an intercultural scan of interesting experiences in the field of computer science education, specifically in the case of the didactic of software design in different parts of the world. After a previous study, conducted through qualitative research in the context of University teaching, the article sets the parameters of reflection to the tenor of the globalization process, offering some proposed future ideas specific to the field of teaching with critical thinking. It concludes that critical thinking contributes to the contextualization of the didactic field immersed in the globalization processes.

Key words: intercultural, globalization, didactic, teaching computer science.

INTRODUCCIÓN

Los avances tecnológicos en Ciencias de la computación llevan un ritmo muy acelerado; en este sentido, es común ver hoy por hoy que ciertos productos de naturaleza tecnológica computacional manejan sus tiempos de vigencia en lapsos muy cortos. Es así como se puede observar que ciertos elementos –tanto hardware como software– adquiridos por el consumidor, son utilizados por un periodo corto de tiempo –generalmente meses o incluso semanas– cuando de repente aparece otro elemento que lo mejora o incluso lo substituye totalmente por razones de obsolescencia del elemento anterior.

Este ritmo vertiginoso de la evolución tecnológica puede ser observado desde la perspectiva comercial asociado a la “cultura” del consumismo, si puede llamarse de esta manera. Sin embargo, al alejarse momentáneamente de su producción técnica y tecnológica, las Ciencias de la computación –al igual que cualquier ciencia– son tratadas en escenarios académico-científicos. Es aquí donde los avances en materia de didácticas de la enseñanza de las ciencias computacionales muestran una notoria diferencia evolutiva con respecto a las producciones tecnológicas que de ella se desprenden.

Este artículo esta soportado en los resultados de una investigación titulada “Didáctica del Diseño de Software”, que produjeron un libro en el que se plantean los lineamientos generales de didácticas en computación con base en los resultados de la investigación previa. Como temática central, se abordan algunos aspectos interculturales a tenerse en cuenta en la búsqueda de construcciones didácticas al canon de las ciencias computacionales. Una mirada hacia una colección de experiencias en materia de enseñanza en diferentes partes del mundo, ha permitido esbozar los elementos comunes no sólo en materia de contenidos sino de metodologías en el campo de la didáctica. Es importante resaltar que mundialmente se ha establecido una serie de lineamientos y directrices en materia de contenidos sobre la enseñanza de las Ciencias de la computación; sin embargo, son realmente poco visibles los aportes referentes a reflexiones asociadas al ejercicio propio de la enseñanza.

1. UNA MIRADA A LA INTERCULTURALIDAD Y EL FENÓMENO DE LA GLOBALIZACIÓN

El conocimiento manejado dentro de ambientes académico-científicos traspasa las barreras del espacio; en este sentido, dicho conocimiento trasciende los espacios geográficos donde es apropiado alrededor del planeta, más aún cuando se cuenta con tecnologías de la información y las comunicaciones que facilitan el

fenómeno de divulgación del conocimiento. Es así como se ve desfigurado –pero nunca acabado– el concepto de localidad, para entrar en un escenario de “puesta en común” en ámbito global¹.

Teniendo en cuenta las Ciencias de la computación como el escenario principal donde se exploran varios conceptos en este artículo, los contenidos que forman parte de los planes de estudio de las diferentes disciplinas asociadas a dichas ciencias se encuentran estandarizados bajo unos lineamientos claros. En materia de contenidos, organizaciones de corte internacional han podido establecer la base de conocimientos que permite permear la estructurabilidad de los contenidos en los diferentes planes de estudio; entre dichas organizaciones se encuentra ACM (*Association for Computing Machinery*), IEEE-CS (*Institute of Electrical and Electronics Engineers – Computer Society*), y AIS (*Association for Information Systems*). Juntas han elaborado un documento completo donde se establecen los lineamientos para la organización de los currículos asociados a las Ciencias de la computación².

Aquí inicia el primer “choque” conceptual en materia de educación, puesto que la recomendación, que realizan ACM, IEEE y AIS, centra su atención en los contenidos de forma explícita; y aun así, estas organizaciones le llaman currículo a la organización de contenidos que ellas proponen³. Hoy por hoy, en la arena educativa se entiende por currículo a todos los elementos que de manera articulada forman parte de los procesos educativos y de formación. En este orden de ideas, el concepto de currículo se constituye como un constructo teórico de gran complejidad, que sobrepasa la mera definición de un plan de estudios. Dicha complejidad se manifiesta en una visión particular que tiene el gobierno colombiano cuando afirma, a través de su Ley General de Educación, que el currículo forma parte de un proceso educativo que contempla “un conjunto de criterios, planes de estudio, programas, metodologías y procesos que contribuyen a la formación integral del individuo así como a la construcción de una identidad regional, local y nacional”⁴.

Esta complejidad que presenta el currículo va de la mano con los contextos donde los currículos se aplican; de acuerdo a esto, existen factores culturales que

1. DUNNING, J. *Regions. Globalization, and the knowledge-based economy*. Oxford: Oxford University Press, 2002. p. 34, University of Liverpool Catalogue – EBSCOhost. Fecha de consulta: 26 de agosto de 2012.
2. ACM, IEEE Computer Society and AIS. *Computing curricula: the overview report*. 2005. En Internet: http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf. Fecha de consulta: septiembre 11, 2012.
3. TAN, G., & VENABLES, A. “Survival mode: the stresses and strains of Computing Curricula Review”, *Journal Of Information Technology Education*, enero, 2008: IIP33-IIP43, p. 34, Education Research Complete, EBSCOhost, Fecha de consulta: 26 de agosto de 2012.
4. CONGRESO DE LA REPÚBLICA DE COLOMBIA, Ley 115 de 1994 (Febrero 8), Ley General de Educación. Artículo 76. En: Diario Oficial. No. 41214 del 8 de febrero de 1994.

intervienen en los procesos educativos y marcan la diferencia en su desarrollo. Así, pues la elaboración curricular presenta grandes niveles de complejidad, donde la cultura y los contextos inciden en su construcción; de esta forma, existen múltiples conceptos de currículo que se abordan desde diferentes ópticas, según el paradigma conceptual y las corrientes de pensamiento subyacentes⁵.

Una de las manifestaciones principales de la corriente postmoderna es el fenómeno de la globalización⁶. En ambientes académicos, la globalización ha tenido grandes impactos; centrándose en las Ciencias de la computación, el conocimiento manejado en ambientes de Educación Superior a nivel de pregrado es producido principalmente en Estados Unidos y Europa, con aportes de Asia. Al tenor de esta realidad, los contenidos propuestos a través del documento llamado *Computing Curricula* de ACM, IEEE y AIS regulan el hilo conductor de las líneas de formación a través de las diferentes disciplinas relacionadas con las Ciencias de la computación. Éste quizás es el fenómeno de mayor relevancia –e incluso, de mayor visibilidad– en materia de globalización inmersa en escenarios académicos, debido a que no sólo se “exporta” el contenido de formación académica, sino que se adjunta la tecnología asociada para manejar dichos contenidos. En otras palabras, se ha estipulado –de común acuerdo a nivel mundial– que la formación de profesionales en las Ciencias de la computación no debería responder en forma exclusiva a resolver las problemáticas locales; en su lugar, la formación de dichos profesionales apunta principalmente a la solución de problemas de naturaleza global. Sin embargo, la libertad de la que gozan las instituciones de educación superior en el mundo hace que la orientación de sus procesos de formación en materia de Ciencias de la computación goce de un toque de autonomía.

Es importante ver al fenómeno de globalización como un momento histórico y no como una realidad absoluta e inmodificable. De esta forma, las circunstancias de esta tendencia de unificación mundial han generado diversos fenómenos de resistencia, incluyendo anacronismos, fundamentalismos y tradicionalismos, tal como lo destaca Milian Guillermo Enríquez⁷. No obstante, en materia académica, las Ciencias de la computación están inmersas dentro de este imaginario ideológico, donde sus contenidos claramente reflejan un panorama de universalización. Un ejemplo claro de estas acciones puede ser evidenciado en la utilización de lenguajes de naturaleza computacional (incluyendo las propuestas teóricas, así como las implementaciones prácticas). Desde la base conceptual teórica, existe

5. USCÁTEGUI DE JIMÉNEZ, Aura Mireya, GOYES MORENO, Isabel. Reformas educativas y currículos universitarios. En: Itinerantes Revista del Área de Pedagogía, Currículo y Didáctica del Doctorado en Educación de RUDECOLOMBIA, Editorial Universidad del Cauca, v.1 fasc. 2002, p. 90.

6. ENRÍQUEZ, Milian Guillermo. La filosofía intercultural andina en la posmodernidad y la globalización. En: Cuadernos de Filosofía Latinoamericana 28.97, 2007; p. 16.

7. *Ibíd.*, p. 17.

un marco de referencia para facilitar la modelación de sistemas de cualquier naturaleza: este es el caso de UML (*Unified Modeling Language*), que es utilizado globalmente para describir modelos de sistemas a través de símbolos y diagramas. Con este ejemplo, se cumple fielmente la filosofía de estandarización de conceptos a fin de que un diagrama en UML sea entendido por cualquier profesional, sin importar su lengua materna ni su contexto en espacio y tiempo.

Al momento de “armar” un plan de estudios en las diferentes disciplinas asociadas a las Ciencias de la computación, el común denominador es ver el referente internacional propuesto a través del documento llamado *Computing Curricula* de ACM, IEEE y AIS. Tal documento es considerado la principal carta de navegación en materia de contenidos. Este grado de estandarización teórico-conceptual ha dado pie para el desarrollo de procesos de globalización en ambientes académicos. Sin embargo, teniendo en cuenta la complejidad curricular existente de acuerdo a cada contexto, ese “atreimiento” de globalizar las cosas ha tenido su centro en los contenidos, mas no en las estrategias pedagógicas. En materia de didácticas, es preciso hablar de una “libertad” condicionada a los lineamientos propios de cada institución, departamento, programa, e incluso de cada profesor. Para bien o para mal –dependiendo de la lente con que se mire el fenómeno–, son poco visibles las propuestas en materia de didáctica de las Ciencias de la computación desde un punto de vista intercultural en escenarios de globalización. A fin de realizar un acercamiento a este interesante tópic, una investigación fue realizada en el año 2010, cuyo objeto de estudio fue la construcción de una didáctica específica para el diseño de software contemplando experiencias interculturales en varios puntos del planeta. Se debe entender que se trata de una sola parte (podría asumirse como curso o conjunto de cursos) dentro de la formación en Ciencias de la computación, dicha parte es el diseño de software, para lo cual se presenta un resumen de los resultados de dicha investigación en el capítulo a continuación.

2. EXPERIENCIAS INTERCULTURALES ALREDEDOR DE LA ENSEÑANZA EN CIENCIAS COMPUTACIONALES

Los antecedentes que sirvieron como punto de partida de la investigación fueron contemplados desde un ámbito regional, nacional e internacional. Las experiencias interculturales fueron provistas por Daniel Jackson (*Massachusetts Institute of Technology* - USA), Ralph Johnson (*University of Illinois at Urbana Champaign* - USA), Dewayne E. Perry (*University of Texas at Austin* - USA), Jonathan Aldrich (*Carnegie Mellon University* - USA), Tevfik Bultan (*University of California at Santa Barbara* - USA), George Camdea (*EPFL - Swiss Federal Institute of Technology* - Suiza), Alexander L. Wolf (*Imperial College London* - Reino Unido), David Notkin (*University of Washington* - USA), Premkumar T. Devanbu (*Univer-*

sity of California at Davis - USA), Kenneth M. Anderson (*University of Colorado at Boulden - USA*), John Keklak (*Boston University - USA*), P.G. Kluit (*Delft University of Technology - Holanda*), Paolo Bottoni (*University of Rome "La Sapienza" - Italia*), Diego Hernán Montoya Bedoya (*Escuela de Ingeniería de Antioquia - Colombia*), Alexander Barón (*Universidad de Nariño - Colombia*), Carlos Cobos (*Universidad del Cauca - Colombia*) y Armando Rafael Acuña Martínez (*Fundación Universitaria San Martín - Colombia*).

La investigación tuvo una duración de dos años y los resultados fueron consolidados en 2010. Ésta buscó fundamentar los elementos básicos en la construcción de una didáctica para el curso (o los cursos) de diseño de software desde una perspectiva intercultural. Esta investigación exploró los contenidos de los cursos de diseño de software, los cuales fueron impartidos por 15 profesores internacionales y 4 nacionales, de acuerdo con la lista presentada anteriormente. De igual forma, a través de un análisis cualitativo y exploratorio de los estudios de caso, se lograron identificar los componentes generales de las didácticas teniendo en cuenta aquellos casos exitosos desde el punto de vista profesoral. Tanto el análisis estadístico descriptivo como el análisis de naturaleza cualitativa permitieron identificar y describir los aspectos positivos y negativos de las prácticas docentes en dichos cursos, así como los elementos que deberían formar parte de la didáctica propuesta.

Con el ánimo de contextualizar la temática referente a los cursos de diseño de software, se debe entender que, en Ciencias de la computación, el software aparece como la entidad que permite “materializar” una solución a pesar de que la definición misma de software lo hace parecer como la parte lógica de los sistemas de información computacionales⁸. El concepto de ingeniería de software aparece históricamente en 1968 en una conferencia que fue mantenida en su discusión con el nombre de “software crisis”⁹; en este entorno se empieza a vislumbrar el término de ingeniería de software como una disciplina para la construcción de soluciones computacionales basadas en software, bajo unos criterios claros de calidad, productividad y profesionalismo¹⁰. En términos prácticos y teniendo en cuenta la situación actual, el diseño de software se aborda desde un paradigma orientado a objetos –no siendo el único paradigma– dadas las evidencias de uso y producción a nivel profesional en los ámbitos laborales, científicos y académicos.

8. ADAMS, A., & McCRINDLE, R. Pandora's box: social and professional issues of the information age. New Jersey: Wiley NJ, 2008.

9. NAUR, P., & RANDELL, B. Software engineering: report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels: Scientific Affairs Division, NATO, 1969. p. 5.

10. SOMMERVILLE, I. Software Engineering, 9th Edition, Boston: Pearson Education/Addison-Wesley, MA., 2011. p. 5.

La idea de sistemas orientados a objetos se remonta a la década de 1960, cuando fueron utilizados inicialmente en algunos entornos gráficos experimentales. Estas ideas luego emigraron a los lenguajes de programación a finales de 1960 con el lenguaje Simula. Varios métodos de análisis y diseño utilizando conceptos Orientados a Objetos comenzaron a aparecer en la década de 1980. Cada método tiene su propia notación o lenguaje de modelado, lo que condujo a cierta confusión y en dicha confusión surge UML como el comienzo de un esfuerzo para combinar el trabajo de dos dirigentes, metodólogos Grady Booch y Jim Rumbaugh, en Corporación Rational. Casi al mismo tiempo, el Object Management Group –OMG– lanza una convocatoria de propuestas para un lenguaje de modelado de objetos estándar. La notación que Booch y Rumbaugh habían desarrollado fue presentada por Rational para el OMG y, tras una serie de deliberaciones y revisiones, se ha convertido en el estándar actual en materia de diseño de software, sin importar el proceso de software a seguir. Posteriormente se une a la dupla otro autor, llamado Ivar Jacobson, y con él se han producido una serie de documentación sobre la especificación del lenguaje de modelado unificado para el diseño de software¹¹.

La investigación a la que hace referencia este artículo clarifica el uso del lenguaje UML como herramienta de modelación. Sin embargo, varios enfoques del proceso de software son utilizados en el desarrollo de los cursos de diseño de software; entre los cuales se tiene preferencia por: *Waterfall Process*, *Iterative-Incremental Process*, *RUP*, *CBSD* y *Agile Methodologies* (tales como *eXtreme Programming*, *Scrum* y *AUP*, entre otros). En términos generales, la mayoría de profesores explora en el desarrollo de sus clases dos o tres procesos de software; sin embargo, los profesores hacen énfasis en uno de ellos, de acuerdo al criterio de cada uno de ellos. El común denominador es el uso de UML como herramienta de modelado, sin importar qué proceso de software es seguido; ya la rigurosidad –y extensión– de dicha documentación depende del tipo de proceso de software que se desea trabajar¹².

Es importante aclarar que, del listado de procesos software preferidos por los profesores, RUP es el proceso más utilizado en los ambientes académicos consultados. En este sentido, se observa que RUP es una creación de instancias de USDP –*Unified Software Development Process*–, ya que el USDP fue desarrollado por Ivar Jacobson, Jim Rumbaugh y Grady Booch, todos los expertos de procesos conocidos que trabajaban para la Corporación Rational. La verdad es que el USDP

11. OMG Inc. The object management group home page. En Internet: <http://www.omg.org/>. 2007. Fecha de consulta: 5 de septiembre de 2012.

12. BENNET, S., McROBB, S., & FARMER, R. *Object-Oriented Systems Analysis and Design Using UML*. 4th Edition, McGraw-Hill Higher Education, UK., 2010. p. 610.

y RUP tienen una raíz común del trabajo original del proceso llevado a cabo en Rational y la obra de Ivar Jacobson. En 2003, Rational Corporation fue adquirida por IBM. RUP es ahora un producto de IBM y es el componente del proceso de *Rational Method Composer* de IBM¹³.

Es importante resaltar que dentro del resumen de aspectos positivos y negativos de las prácticas docentes se ven reflejados los elementos culturales, como la valoración de la puntualidad, la afinidad para trabajar en equipo, el grado de madurez académica, los aspectos motivacionales, las actitudes frente a la solución de problemas, el desarrollo de las competencias comunicativas, entre otros. Los cuadros semánticos producto del análisis cualitativo se presentan en la Figura 1, de acuerdo con los aspectos positivos y negativos de las prácticas docentes.

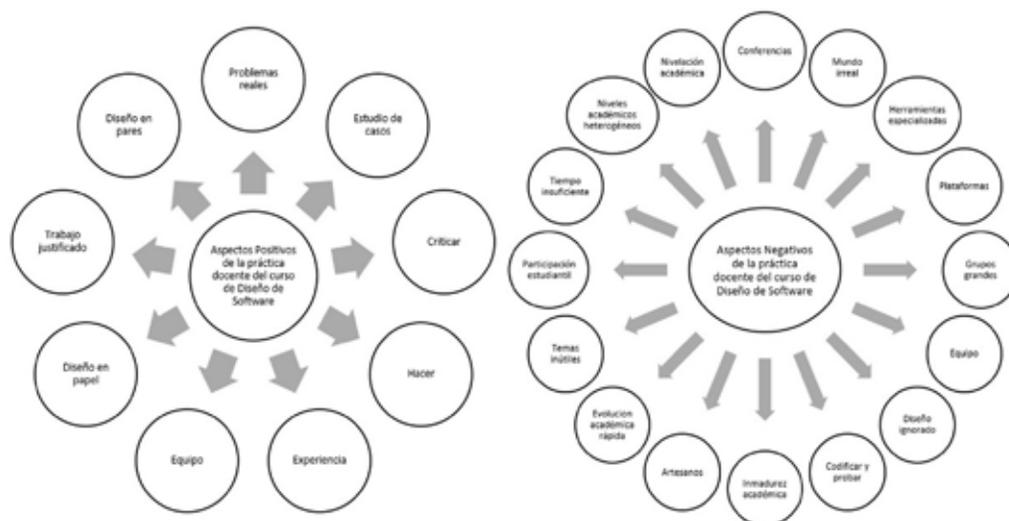


Figura 1. Aspectos Positivos y Negativos de las prácticas docentes¹⁴

Teniendo en cuenta el análisis cuantitativo a través de las encuestas procesadas y el análisis cualitativo a fin de determinar las categorías relevantes para la investigación, la didáctica propuesta logró hacer una síntesis sobre los estilos de enseñanza que se inscriben dentro de una corriente de pensamiento asociada a modelos pedagógicos.

13. IBM, Rational Method Composer. En Internet: <http://www-306.ibm.com/software/awdtools/rmc/index.html>, 2006. Fecha de consulta: 11 de septiembre de 2012.

14. JIMÉNEZ, Robinson, MARTÍNEZ, Álvaro. & INSUASTI, Jesús. Lineamientos para una didáctica en computación – Caso: Curso de Diseño de Software, Alemania: Editorial Académica Española en colaboración con AV Akademikerverlag GmbH & Co., 2012. pp. 103-104.

En cuanto a la identificación de los modelos pedagógicos utilizados, es evidente que se empieza a marcar la tendencia hacia aquellos de naturaleza progresista e incluso conductista. Es relevante hacer mención de la naturaleza epistemológica de las Ciencias de la computación en el marco de una corriente de pensamiento positivista instrumental. Por formar parte del campo de estudio de las ingenierías, las Ciencias de la computación comparten las características que, según Habermas, son propias de las áreas del conocimiento relacionadas con las ciencias empírico-analíticas. Se debe recordar que las ciencias empírico-analíticas se distinguen por su tratamiento del dominio-objeto, que se rige por la ley predecible como regularidades que permiten ciertos tipos de técnicas metodológicamente controladas de investigación, que sería inadecuada para las ciencias histórico-hermenéuticas o crítico-sociales¹⁵.

Al tenor de estas ideas, las 19 universidades fueron clasificadas por su modelo pedagógico de acuerdo con 7 escenarios de aplicación, los cuales son: la intención, el método, las metas, el concepto de desarrollo, los contenidos, la relación profesor-estudiante y la evaluación. Frente a cada uno de estos elementos que integran el escenario académico, se identificó la posición bien sea conductista o progresista. El estudio develó que no existe pensamiento o intención romántica, ni socialista, ni tradicional. A decir verdad, en estos resultados se puede comprobar que, dado el enfoque epistemológico dentro del paradigma positivista con un enfoque empírico-analítico de las ciencias computacionales, el modelo pedagógico que predomina en los escenarios académicos consultados sigue una filosofía de instrucciones dentro de los parámetros conductistas. Los resultados del análisis cualitativo están en la Figura 2.

3. SÍNTESIS DE ELEMENTOS COMUNES EN EL CAMPO DE LA DIDÁCTICA

Después de un análisis exhaustivo de las experiencias recogidas en dicha investigación, la didáctica propuesta para el curso de Diseño de Software se estructuró con base en ocho dimensiones: Enfoque Pedagógico, Objetivos Pedagógicos, Contenidos, Actividades de Clase, Actividades de Trabajo Independiente, Recursos Educativos, Espacio Físico y Evaluación. Las dimensiones que sugiere la investigación pretenden abarcar la mayoría de los aspectos que integran una didáctica específica. Todas estas dimensiones se encuentran estrechamente relacionadas, tal como se puede observar en el mapa conceptual general expresado en la Figura 3.

15. HABERMAS, Jürgen, Knowledge and human interests. Toronto, Canada: Beacon Press, 1972, p. 368.

Rank	Universidad	Intención	Método	Metas	Concepto de desarrollo	Contenidos	Relación profesor-estudiante	Evaluación
01	MIT	C	P	P	P	C	P	P
02	Universidad de Illinois - Urbana	C	P	-	P	C	P	P
05	Universidad de Texas at Austin	C	C	C	C	C	P	C
06	Universidad Carnegie Mellon	P	P	P	P	C	P	P
10	Universidad de California Santa Bárbara	P	P	P	P	C	P	P
14	EPFL - Swiss Inst. Fed. tecn.	P	P	P	P	C	P	P
17	Colegio Imperia de Londres	P	P	P	P	C	P	P
18	Universidad de Washington	P	P	P	P	C	P	P
31	Universidad de California Davis	P	P	P	P	C	P	P
35	Universidad de Colorado Boulder	P	P	C	P	C	P	P
36	Universidad de McGill	C	P	P	P	C	C	P
40	Universidad Duke	C	P	P	P	C	C	P
59	Universidad de Boston	C	P	P	P	C	C	P
62	Universidad Tecnológica Delft	-	-	-	-	-	-	-
79	Universidad La Sapienza Roma	C	P	P	P	C	C	P
-	Escuela de Ingeniería de Antioquia	P	P	P	P	C	P	P
-	Universidad de Nariño	P	T	P	T	C	P	C
-	Universidad del Cauca	P	C	P	P	C	C	P
-	Universidad San Martín	P	P	P	P	C	P	P
CONVENCIONES								
	Modelo Pedagógico	Conductista		Progresista		Tradicional		
	Letra asociada	C		P		T		

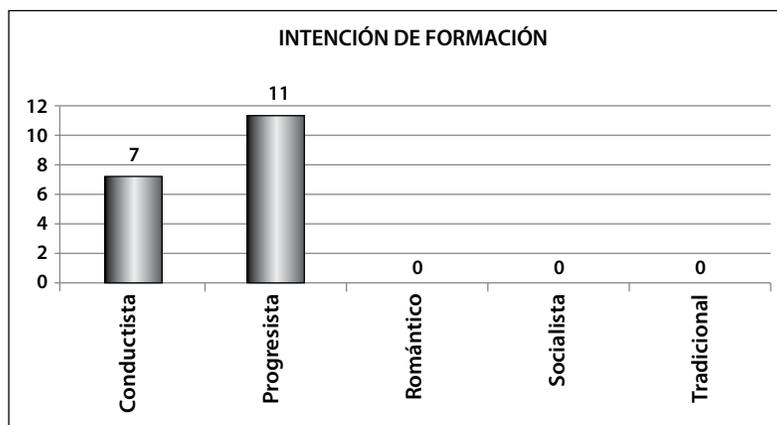


Figura 2. Resultados del análisis cualitativo sobre los modelos pedagógicos de las universidades¹⁶.

16. JIMÉNEZ, Robinson, MARTÍNEZ, Álvaro. & INSUASTI, Jesús. Lineamientos para una didáctica en computación – Caso: Curso de Diseño de Software, Alemania: Editorial Académica Española, en colaboración con AV Akademikerverlag GmbH & Co., 2012, p. 105.

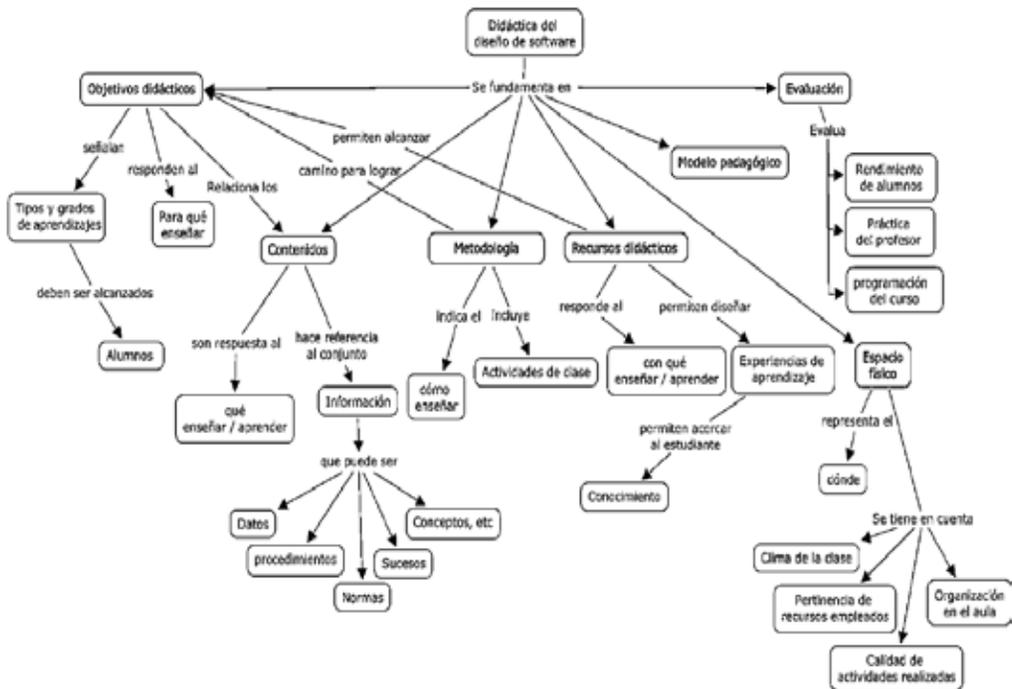


Figura 3. Mapa conceptual de la Didáctica propuesta para el curso de Diseño de Software¹⁷.

La Didáctica propuesta para el curso de Diseño de Software (DDS) es el resultado del análisis y síntesis usando tres fuentes de información: Profesores nacionales y extranjeros en el campo del diseño de software, el documento disciplinar *The Joint Task Force for Computing Curricula 2005*, y la fundamentación teórica sobre aprendizaje activo. El aprendizaje activo fue abordado desde la mirada de Barrado y otros autores, los cuales afirman que “aquellas estrategias pedagógicas son las que propician una actitud activa del estudiante en clase, en contraposición con lo que ocurre en el método expositivo clásico, en el que el alumno se limita a tomar notas de lo que ve en la pizarra (se dice que en una clase expositiva la información pasa directamente de las notas del profesor a la libreta del alumno, sin pasar por sus cabezas)”¹⁸.

Las diferentes culturas consultadas muestran elementos comunes en materia de las prácticas de enseñanza-aprendizaje que, a juicio de sus docentes,

17. JIMÉNEZ, Robinson, MARTÍNEZ, Álvaro. & INSUASTI, Jesús. Lineamientos para una didáctica en computación – Caso: Curso de Diseño de Software, Alemania: Editorial Académica Española, en colaboración con AV Akademikerverlag GmbH & Co., 2012. p. 115.

18. BARRADO, C. et al. Siete experiencias de aprendizaje activo, Universitat Politècnica de Catalunya - Departament d'Arquitectura de Computadors, UPC Press, En internet: <http://people.ac.upc.edu/leandro/pubs/UPC-DAC-2001-11.pdf>, 2001. Fecha de consulta: 11 de septiembre de 2012. p. 20.

son consideradas como exitosas. Entre aquellos elementos comunes a través de diferentes culturas está la elaboración estructurada del syllabus con objetivos pedagógicos explícitos, la preferencia por la forma de redacción de objetivos o metas en lugar de competencias, el uso del lenguaje UML como estándar mundial en modelación, el uso de clase magistral acompañada con laboratorios de prácticas, la preferencia por las tareas y lecturas complementarias como trabajo independiente, el uso de una amplia documentación impresa y digital, y el desarrollo de un proyecto de clase como estrategia más usada para la evaluación.

Cabe resaltar que la tendencia en el desarrollo de la práctica en materia de Diseño de Software se basa en ejemplos de la vida real; así, el estudio de caso frecuentemente se utiliza para tal menester. En la parte negativa de la práctica docente, se encontró que existen factores de naturaleza académico-administrativa que intervienen en el aprendizaje de los estudiantes que asisten a los cursos de Diseño de Software. Entre dichas desventajas se encuentra que los grupos inician clases con diferentes niveles de conocimiento, falta de tiempo asignado al desarrollo del trabajo presencial, falta de motivación en los estudiantes, la programación escasa para desarrollo de trabajos en grupo, la evolución vertiginosa de la profesión en términos técnicos, el apego por la metodología de prueba-y-error, entre otros.

Lo cierto es que, en materia de contenidos, los hallazgos de la investigación apuntan hacia un consenso global fuertemente marcado por las autoridades mundiales que regulan las disciplinas asociadas a las Ciencias de la computación. Teniendo en cuenta su enfoque empírico-analítico de las Ciencias de la computación, los profesores, sin importar su asiento cultural prefieren el estudio de casos como estrategia de aplicación práctica de la fundación teórica; de igual forma, se trata de crear escenarios de participación activa del estudiante tanto en el trabajo presencial como en el trabajo independiente. En conclusión, dentro de la corriente de pensamiento conductista-progresista, los profesores de diferentes culturas comulgan con una metodología de ejemplificación de las soluciones basadas en contextos reales. Son notorios los esfuerzos que han manifestado los profesores encuestados y entrevistados en materia de propiciar espacios de participación activa del estudiantado; sin embargo, dicha participación se ve reflejada únicamente en el desarrollo del trabajo independiente. En esta situación, sin excepción todos los profesores manifestaron su deseo de "inclinarse la balanza" hacia ambientes de aprendizaje activo, pero la realidad muestra que dichas intenciones se quedan en ese nivel puesto que el arraigo de las bases conductistas-progresistas es demasiado fuerte en dicha academia. Por eso, la propuesta como "producto" final de la investigación integra elementos del aprendizaje activo a fin de enriquecer la práctica docente para que la mera intención de los profesores trascienda a realidades tangibles.

4. LA CRÍTICA

Además de elementos comunes que trascienden las diferencias culturales encontradas en la investigación, se hallaron otros elementos que forman parte de la heterogeneidad intercultural y que inciden en los procesos de enseñanza aprendizaje. Iniciando con el manejo del tiempo, se encontró que las culturas de Estados Unidos y de Europa son supremamente estrictas en la definición de horarios y las fechas y horas de entrega de trabajos y proyectos (*deadlines*); algo similar sucede en los escenarios colombianos investigados, pero de forma no tan estricta. Otro punto interesante es la orientación hacia el trabajo en equipo, promovido en escenarios extranjeros; en contraste por las evidencias de la muestra en Colombia, se observa a esta actividad como opcional y no como prioritaria. El uso de tecnología de soporte, como ambientes virtuales de aprendizaje y el uso de herramientas tecnológicas en línea, depende de la cultura y la edad del profesor; en este particular, la investigación encontró que culturalmente es mucho más aceptable el uso de tecnología de soporte para la enseñanza y el aprendizaje de los cursos de Diseño de Software en las culturas de Estados Unidos y de Europa que en Colombia; preferiblemente los profesores que aún no superan los 40 años de edad suelen optar por este tipo de ayudas educativas.

Se ha dicho con anterioridad que los contenidos –que de hecho, tienen una orientación conductista más que estructuralista o progresista– han sido estandarizados en respuesta a los fenómenos de globalización siguiendo parámetros de desarrollo de competencias laborales; con esto queda abierto el debate sobre la estandarización de la práctica docente frente a dichos fenómenos de globalización. Históricamente se ha podido evidenciar que el avance de las Ciencias de la computación *per se* es supremamente rápido, e incluso vertiginoso. En materia de conocimiento específico, quizás son las Ciencias de la computación escenario que más desactualización presenta en periodos cortos de tiempo. No sucede igual con las reflexiones en materia pedagógica; así, lo muestra la escasa productividad de conocimiento en dichas esferas.

Es probable que esa falta de reflexión en materia de mejoramiento de la práctica docente, así como de las metodologías para facilitar el aprendizaje, hayan “encallado” a las Ciencias de la computación dentro de una corriente de pensamiento positivista inmersa exclusivamente en pedagogías conductistas y estructuralistas. Es preciso, a estas alturas de la “vida académica”, tomar un aire de reflexión y establecer los elementos de juicio apropiados para la construcción crítica de nuevas didácticas a fin de alcanzar ese fin anhelado de la educación que es la formación del ser humano. Sin embargo, si se ve la situación con el lente positivista, se podría afirmar que las pedagogías utilizadas en las Ciencias de la computación han dado excelentes resultados (resultados funcionales e instrumentales propios de la corriente de pensamiento positivista); en ese orden de

ideas, quizás la postura de efectividad de las pedagogías conductistas y estructuralistas sea la más apropiada. Pero valdría la pena escudriñar los intereses que se encuentran detrás de alcanzar ese fin último con las pedagogías mencionadas.

Si la intención detrás de todo el aparato educativo de las Ciencias de la computación se orienta exclusivamente a la producción de tecnología con fines lucrativos, ergo, los modelos pedagógicos actuales con que se trabaja son suficientes. Pero, ¿qué hay del papel de la tecnología en el desarrollo humano? Y ¿qué se entiende por desarrollo humano? Se abre de esta forma el campo para la reflexión, donde intervienen el mundo de la subjetividad, el mundo de la interpretación crítica y argumentada, el mundo donde el texto y el contexto se entrelazan a fin de crecer juntos en un ambiente de incertidumbre.

La investigación que se menciona cumplió su objetivo principal, que consistía en la construcción de una propuesta didáctica ajustada a una serie de realidades manifestadas por aquellos que participaron en su construcción como fuentes primarias. Se logró producir un acercamiento hacia pedagogías que se desprenden de las corrientes conductistas y estructuralistas, con la intención de explorar posturas ideológicas cercanas a los ámbitos sociales. Con todo esto, la propuesta fue retroalimentada a las 19 universidades del planeta que participaron en la su construcción; sin embargo, hasta ahora, la didáctica no ha sido probada en contextos reales.

Si bien es cierto que los elementos integrados en dicha didáctica provienen de las que podría llamarse “buenas experiencias” (en términos de los profesores que impartieron dichos cursos) en materia de la enseñanza del Diseño de Software desde una perspectiva intercultural, está pendiente su validación a fin de determinar si realmente dicha didáctica responde a las necesidades evidenciadas en los escenarios académicos. Esta situación da pie a una subsecuente investigación para determinar qué tan conveniente sería la búsqueda de la construcción de didácticas dentro de ambientes de estandarización de la práctica docente en respuesta a los fenómenos de globalización. ¿Se estaría hablando de una violación a la libertad de cátedra? Las propuestas vienen siendo lo que la gente quiera hacer de ellas; en este sentido, una propuesta sugiere y argumenta, pero finalmente, la decisión es de aquellos individuos que deseen plasmar en la realidad aquellos imaginarios planteados en dichas propuestas.

De esta forma, existe la posibilidad de abrir una puerta al mundo de la subjetividad, de la interpretación, de racionalidades no instrumentales, dando una mirada a los impactos en la sociedad. Si de buscar cambios sociales se trata en pro del desarrollo humano, entendido desde una visión holística de múltiples racionalidades, entonces la teoría crítica tiene bastantes elementos que aportar a la construcción dialéctica de ambientes de enseñanza y aprendizaje. Con una

perspectiva humanizadora y de servicio social superpuesta sobre intereses materiales y meramente técnicos. Bien decía Max Horkheimer, considerado como una de las figuras más representativas de la corriente de pensamiento crítico, que la teoría crítica es una teoría social orientada hacia la crítica y la transformación de la sociedad en su conjunto, en contraste con la teoría tradicional orientada sólo a la comprensión o explicación de la realidad. Horkheimer quería distinguir la teoría crítica como forma radical y emancipadora de la teoría marxista, criticando tanto el modelo de la ciencia propuesto por el positivismo lógico y lo que él y sus colegas vieron como el positivismo y el autoritarismo encubierto del marxismo ortodoxo y el comunismo¹⁹.

Entonces, la visión original de Max Horkheimer se ajusta a las ideas expuestas en la investigación mencionada en este artículo. Sea esta la oportunidad para incursionar en un campo que ha sido realmente poco explorado en los ambientes de enseñanza de las Ciencias de la computación desde una perspectiva crítico-social. Es claro que no sería viable desprenderse de forma radical de la racionalidad heredada del positivismo lógico en cuanto se refiere a contenidos de Ciencias de la computación; de hecho, esta relación es natural dada la epistemología propia de dichas ciencias. Se trata más bien de orientar los procesos de enseñanza de los contenidos (técnicos e instrumentales) antes mencionados a fin de establecer marcos de reflexión sobre sus intereses y fines últimos. Para nadie es extraño que la tecnología tiene su impacto en la sociedad; la tecnología *per se* no es ni buena ni mala, es lo que es y nada más: un instrumento producto de procesos científicos. El uso que se le da en la sociedad causa impactos que, dependiendo del lente con que se miren, pueden ser beneficiosos o nocivos.

Con esta reflexión, el rol del profesor en los procesos de enseñanza de las Ciencias de la computación experimenta una transformación en actitud profunda. Por consiguiente, las didácticas asociadas a dichos espacios deberán ser reflejo del cambio actitudinal y comportamental que experimentaría el docente frente a una posición crítico social. Al abrigo de estas ideas, se hace necesario recordar los postulados de Erich Fromm cuando escribe sobre las relaciones entre el conocimiento y el desarrollo de las sociedades desde una perspectiva sicoanalítica. Fromm, en su texto *The Revolution of Hope, toward a humanized technology*, hace alusión a las características que forman parte del constructo social en el marco de una explosión tecnológica que incide en el desarrollo del ser humano; Fromm hace una advertencia sobre los peligros de la mecanización tecnológica, que pueden opacar la naturaleza humana. En este sentido, en su libro Fromm propone un movimiento social radical de redescubrimiento de la

19. HORKHEIMER, Max. Traditional and critical theory. En: Critical theory: selected essays, 1937, pp. 188-243.

esencia humana, frente a la degeneración social causada por los impactos tecnológicos²⁰. Quizás estas posturas tan radicales fueron el detonante para disparar el criticismo sobre sus obras, por parte de Chomsky y Marcuse. A pesar de sus ideas revolucionarias para la época, Fromm tiene grandes aportes, rescatados ahora dentro de la intencionalidad de construir didácticas apropiadas a las Ciencias de la computación con un toque crítico social, dado que se cuestionan constantemente los intereses ocultos de la producción tecnológica, en contraste con los ideales de desarrollo humano. Las posturas de Fromm son iniciadas desde una base puramente psicoanalítica; sin embargo, los aportes que hace a la teoría crítica son de valor incalculable. Así, las reflexiones en profundidad sobre los intereses detrás de la producción tecnológica deben ser contempladas en los procesos de construcción de didácticas apropiadas para las Ciencias de la computación, en estos días.

¿Es, entonces, pertinente una estandarización de la práctica docente en las Ciencias de la computación? Ciertamente, esta pregunta carece de respuesta inmediata; haciendo una reflexión dentro de la temática de los intereses ocultos del conocimiento, la tecnología, como producto del conocimiento impartido, manejado y creado en los ambientes científico-académicos de las ciencias computacionales, tendría que llevar las vestiduras de una racionalidad social de pertinencia para el desarrollo humano, entendido como los escenarios históricos para el crecimiento integral del ser. Dicha estandarización o unificación de criterios en materia de didácticas, esto es la parte de enseñanza de las ciencias computacionales, podría ser factible (e incluso más saludable) con una visión humanizadora en pro del crecimiento de la sociedad bajo principios de democracia y libertad, con respeto a las diferencias, buscando la igualdad social y tratando de eliminar las brechas de separación de los grupos culturales. Parece utópica esta postura ideológica; no obstante, existen espacios posibles donde puede llevarse a cabo este cambio desde una mirada a los conceptos sobre praxis que forman parte del corpus teórico del filósofo neo-marxista Karel Kosik. En su libro *Dialectics of the Concrete*, Kosik presenta una postura crítica frente a las diferentes problemáticas del ser humano y su mundo; la praxis es entendida como un escenario donde la teoría reafirma su sentido en la aplicación del conocimiento en la realidad desde una perspectiva de complejidad. Los contextos sociales priman a la hora de dar sentido a la praxis; apartándose un poco de la racionalidad cartesiana, las ciencias deberían encontrar en la praxis su camino y no su fin último. Quizás la búsqueda desesperada de la explicación y el entendimiento de la realidad, muy propia de las ciencias empírico-analíticas ha cegado la visión general que trasciende al

20. FROMM, Erich. *The revolution of hope, toward a humanized technology*, Harper and Row, New York, 1968. p. 160.

contexto cuando la praxis se orienta hacia fines más altruistas. Kosik afirma que "La praxis se funde con todo el hombre y lo determina en su totalidad"²¹; en este sentido, ¿no debería la producción tecnológica fundirse en una sociedad en pro del crecimiento del ser únicamente? De esta manera, la producción tecnológica debería ser concebida dentro de esta racionalidad de crecimiento del ser humano, en lugar de la mera instrumentalización con ánimo de lucro a fin de sostener un modelo económico-social llamado capitalismo, siendo este último interpretado como un momento histórico y no como una realidad absoluta e intransformable.

La praxis, es entonces, el centro neurálgico donde se debe profundizar en reflexión en materia de didácticas para las Ciencias de la computación. Recordando a Husserl, la diferencia existente entre las intencionalidades de la teoría y la práctica dentro de los procesos de transformación humana ha marcado un antecedente interesante en la filosofía idealista que se produjo en el siglo pasado. Husserl establece un "rescate" de la filosofía como fuente indispensable para el desarrollo de las ciencias, postura que al parecer, había sido desdibujada por los extremos objetivos del positivismo lógico y el naturalismo; despojándose de la investidura del sicologismo para abrazar la corriente fenomenológica, Husserl utilizó esta última como método de enseñanza e incluso como ciencia misma al adentrarse en el mundo de la filosofía. Husserl mantiene una postura crítica frente a lo que denominó la crisis de las ciencias, donde promueve la fenomenología como la alternativa para el conocimiento de la realidad de forma objetiva, sin descuidar su esencia no sólo a través de la explicación de los hechos, como lo determina exclusivamente la corriente positivista.

En síntesis, los aportes de Husserl son también aceptables en materia de construcción de didácticas adecuadas para las Ciencias de la computación, el estudio de la naturaleza de las cosas (en este caso, el desarrollo tecnológico) desde su esencia y reflejado en su intencionalidad, hace un valioso aporte desde la crítica a los procesos de enseñanza en Ciencias de la computación. Dichos constructos didácticos podrían ser abordados desde las dimensiones de la intersubjetividad y los principios de empatía incluidos en la teoría de Husserl²². El mundo de la vida es la realidad compleja e incluso caótica en la cual se desenvuelve el ser humano; el cambio de intencionalidad detrás de la producción tecnológica tendría la oportunidad de reflejar transformaciones sociales de grandes proporciones. Sería interesante aplicar el concepto fenomenológico de la experiencia y describir su utilidad para la comprensión de las experiencias con la tecnología. La propuesta del enfoque teórico y metodológico de la fenomenología como marco

21. KOSIK, Karel. Praxis and Totality. En: *Didactics of the Concrete: A Study on problems of Man and World*, Volumen 53, Reidel Publishing Company, USA, 1976; pp. 176.

22. HUSSERL, Edmund. *Cartesian meditations: An introduction to phenomenology* (D. Cairns, Trans.). Kluwer Academic Publishers, Alemania, 1982, pp. 157.

para el desarrollo de una didáctica adecuada a las Ciencias de la computación sería bastante atractiva para los seguidores de un pensamiento social y para la unificación de criterios en materia de metodologías de enseñanza para las Ciencias de la computación. Los antecedentes interesantes de la aplicación de la fenomenología a las didácticas de las ciencias empírico-analíticas son muy escasos y, por lo general, se han realizado en Europa; no obstante, los resultados son muy alentadores cuando se tiene a la tecnología como catalizador de la investigación con una conciencia social claramente definida²³.

Si bien es cierto que los contenidos de los cursos de Ciencias de la computación están previamente establecidos por organizaciones de corte internacional, la esencia de dichas ciencias puede ser explorada desde la mirada filosófica²⁴. En este sentido, ¿qué se entiende por la filosofía de las ciencias computacionales? Pues bien, del hecho de que a la mayoría de los estudiantes de las Ciencias de la computación no les es familiar la filosofía de la ciencia, los estudiantes suelen recibir una introducción a una serie de cuestiones centrales en tal asunto, tales como los fundamentos de la ciencia, sus supuestos centrales y limitaciones, sus implicaciones, y lo que constituye el progreso científico. Son tres preguntas básicas que aparecen a menudo en la filosofía de la ciencia. Se introducen las cuestiones ontológicas, epistemológicas y metodológicas: “¿Qué es real?”, “¿Cómo podemos llegar a conocer acerca de la realidad?”, y “Por qué principios podemos formar el conocimiento?”²⁵. Adicionalmente a las tres preguntas propuestas por los investigadores Denzin y Lincoln, se sumaría la cuarta pregunta sobre la intencionalidad expresada por las corrientes críticas y sociales: ¿Cuál es el fin último de las producciones tecnológicas en contextos sociales?

5. ¿QUÉ HACER AL RESPECTO?

Finalmente, ¿qué hacer frente a la encrucijada de las propuestas didácticas donde la interculturalidad y la globalización dejan una huella profunda? Quizás una actitud crítica contribuya a desenredar dicha problemática, puesto que históricamente se ha conocido que son pocos los aportes en materia de reflexiones pedagógicas aplicadas a las ciencias computacionales. La respuesta radica en no quedarse con las manos cruzadas y en adoptar una postura crítica frente a los fenómenos asociados a tales ambientes.

23. CILESIZ, Sebnem. A Phenomenological Approach to Experiences with Technology: Current State, Promise, and Future Directions for Research. *Educational Technology Research And Development*, 59(4), En: ERIC, Ipswich, MA., Fecha de Consulta: Septiembre 11, 2012. pp. 487-510.

24. TEDRE, Matti. “Know Your Discipline: Teaching the Philosophy of Computer Science”, *Journal Of Information Technology Education*, 6 (1), En: ERIC, Ipswich, MA. Fecha de Consulta: Septiembre 11 de 2012, 2007, pp. 105-122.

25. DENZIN, N.K. & LINCOLN, Y.S. *Handbook of qualitative research*. London, UK: SAGE. 1994.

En términos de la conciencia social con la que se forman los profesionales en Ciencias de la computación, a través del tiempo, diferentes disciplinas han desarrollado sus propios códigos de ética como guía sobre las responsabilidades, derechos y deberes; por ejemplo, hay un juramento hipocrático en Ciencias de la Salud, y las Ciencias Jurídicas tienen ciertas normas que representan los valores éticos hacia el ejercicio de la profesión. Con esto en mente, la computación no puede ser la excepción a este comportamiento.

ACM (*Association for Computing Machinery*), como una sociedad fundamentada en el área de la computación y disciplinas afines de ámbito educativo y científico, ha desarrollado su propio Código de Ética y Conducta Profesional. Esta sociedad señaló varios puntos -sobre el liderazgo moral, y aspectos profesionales- para recomendar buenas prácticas relacionadas con trabajos alrededor de la computación; es preciso destacar algunos de ellos: Contribuir a la sociedad y el bienestar humano, evitar el daño a otros, y cumplir los contratos, acuerdos y responsabilidades asignadas²⁶.

De acuerdo con las ideas anteriores, se podría concluir que los profesionales de las Ciencias de la computación tienen altas responsabilidades en contextos reales, debido a tres razones principales:

En primer lugar, la historia ha demostrado que los seres humanos son seres sociales, por lo que la vida comunitaria es la naturaleza humana. Para Karl Marx, el hombre es la totalidad de sus relaciones sociales²⁷. En este sentido, los profesionales de diferentes disciplinas juegan papeles significativos en la sociedad, donde los “antiguos” valores recuerdan que se debe buscar el bien común para el desarrollo del potencial humano y su rol en una sociedad justa y equitativa. Lamentablemente, no todas las personas piensan de esta manera; con esto, las dos frases: “Contribuir a la sociedad y el bienestar humano” y “Evitar el daño a los demás”, declaradas por ACM, se aplicarían en su totalidad dentro de un comportamiento racional y, en consecuencia, generarían diferentes responsabilidades dentro de su desarrollo.

En segundo lugar, debido al rápido crecimiento tecnológico en las Ciencias de la computación, más y más desarrollos en esta disciplina se aplican en la vida diaria. Dentro de este contexto, prácticamente todos los escenarios han sido cubiertos por las soluciones tecnológicas generadas desde las Ciencias de la computación donde las personas interactúan todos los días. En consecuencia, las implicaciones de las responsabilidades en la sociedad se plantean para cada

26. ACM. ACM Code of ethics and professional conduct. En Internet: <http://www.acm.org/about/code-of-ethics>, Fecha de Consulta: Septiembre 11, 2012, 1992

27. Vease tesis No 6. MARX, Karl. *Theses On Feuerbach*. 1st Publication 1924. Institute of Marxism-Leninism, Moscow, 1845.

innovación y/o adaptación tecnológica. Por eso, es preciso apoyar la idea de hacer un énfasis explícito en el código de ética para los profesionales en Ciencias de la computación. Estell y Christensen muestran una interesante propuesta sobre cómo articular el Código de Ética propuesto por ACM con nuevos profesionales²⁸; sin embargo, dicha propuesta corre el riesgo de ser un poco superficial, porque un simple rito de graduación –tal como un juramento al momento de la ceremonia de grado– puede ser tomado sin “sentirlo” intensamente. Los juramentos son un estilo de vida y no una serie de frases que se las llevaría el viento. En concordancia con lo anteriormente expuesto, dicho Código de Ética debe ser no sólo un curso dentro de los planes de estudio, sino que debe ser una estrategia de formación transversal en todos los programas de estudios, a fin de hacer cumplir el verdadero significado del Código de Ética a través de la educación formal y bajo una pedagogía crítico-social.

Finalmente, las organizaciones tienen sus reglas y estrategias para alcanzar los objetivos. De acuerdo con esta lógica, ACM afirmó que los profesionales deben “respetar los contratos, acuerdos y responsabilidades asignadas” con la intención principal de articular comportamientos éticos y productividad. En una lectura más profunda, ACM dijo que esta iniciativa no debe ir en contra de las implicaciones éticas o morales, ni contradecir todos los demás puntos hechos en su Código de Ética. Esta es el área de la ética empresarial.

Así, todo el desarrollo de las ciencias computacionales genera impactos en la sociedad, que pueden ser buenos o malos, dependiendo de los puntos de vista. La pregunta es: ¿qué está haciendo el mundo académico con el fin de formar profesionales conscientes del impacto que causan a la sociedad? Parece que la respuesta más acertada está del lado de la formulación de didácticas apropiadas que incidan en la conciencia social de los profesionales en formación. Entonces, es necesaria la vinculación de diferentes paradigmas y corrientes de pensamiento en la estructuración de currículos y, más aun, de didácticas donde no solo prevalezca la visión del positivismo lógico, sino que se articulen y se conjuguen elementos de las corrientes crítico-sociales y, ¿por qué no?, también las de corte histórico-hermenéutico. No obstante, es una tarea muy difícil, pero no imposible.

CONCLUSIONES

La exploración de los resultados de la investigación en materia de caracterizar los elementos que deberían formar parte de una didáctica en Ciencias de la computación ha permitido develar nuevos horizontes en materia de aplicación

28. ESTELL, J. & CHRISTENSEN, K. 'The need for a new graduation rite of passage', Communications of the ACM. February 2011. 54(2). 2011, pp.113-115.

alternativa de corrientes de pensamiento complementarias a la racionalidad positivista.

Siendo la investigación de corte eminentemente teórico-conceptual, es menester generar los mecanismos para la validación de dichas propuestas en contextos reales; de este modo, fenómenos de interpretación, de subjetividad y de crítica serán necesarios para contextualizar la propuesta, a fin de estimar la incidencia de los impactos en la sociedad y en la formación del individuo.

Son los espacios propicios del Doctorado en educación, donde las propuestas en materia de innovaciones curriculares y didácticas se materializan desde una perspectiva crítica. La búsqueda constante de las soluciones a las problemáticas sociales motiva el desarrollo de proyectos de esta naturaleza. Con esto se tiene que los escenarios de construcción dialéctica, donde intervienen diferentes posturas ideológicas para “sacar adelante” las propuestas de impacto social y cultural, son los puntos de partida para la generación de conocimiento donde su intencionalidad deberá estar formulada en función de las corrientes de pensamiento participantes.

Si bien es cierto que la corriente de pensamiento crítico hace importantes aportes desde la mirada de la praxis en contexto para las didácticas en Ciencias de la computación, a su vez es notoria la dificultad de establecer los horizontes de pensamiento decolonial dada la naturaleza de dichas ciencias; en este sentido, la producción teórico-práctica, que se materializa generalmente en tecnología computacional, ya nace con un propósito de masificación de su uso, atravesando las barreras culturales donde dichas innovaciones o adaptaciones tecnológicas serán implementadas.

Por otro lado, es evidente que la producción de conocimiento en las Ciencias de la computación se encuentra centralizada especialmente en Estados Unidos de América, Europa y algunos países de Asia, sin desconocer la producción científica que se realiza en el resto del mundo. Con este panorama, el fenómeno de colonialidad sigue manifestándose por la posición hegemónica de aquellos lugares que producen tecnología de punta sobre aquellos que la consumen. Dada esta situación, el pensamiento decolonial en Ciencias de la computación enfrenta grandes retos de acuerdo con la enorme dependencia tecnológica de los países que se consideran en vía de desarrollo.

Si bien es cierto que las reflexiones sobre la enseñanza de las Ciencias de la computación no han tenido el mismo nivel de desarrollo y producción que la ciencia *per se*, para un mundo en constante cambio, la sociedad necesita nuevos planteamientos desde la academia con el objeto de afrontar los retos en materia de apropiación social del conocimiento encaminada hacia la exploración de nuevos horizontes de desarrollo colectivo. Es el momento de actuar, ¡y la educación es un camino posible!

BIBLIOGRAFÍA

ACM. ACM Code of ethics and professional conduct. En Internet: <http://www.acm.org/about/code-of-ethics>, Fecha de Consulta: septiembre 11, 2012, 1992.

ACM, IEEE Computer Society and AIS. Computing curricula: the overview report. 2005. En Internet: http://www.acm.org/education/education/curric_vols/CC2005-March06Final.pdf. Fecha de Consulta: septiembre 11, 2012.

ADAMS, A., & McCRINDLE, R. Pandora's box: social and profesional issues of the information age, Wiley, New Jersey, NJ, 2008, p. 635.

BARRADO, C. et al. Siete experiencias de aprendizaje activo, Universitat Politecnica de Catalunya - Departament d'Arquitectura de Computadors, UPC Press, En Internet: <http://people.ac.upc.edu/leandro/pubs/UPC-DAC-2001-11.pdf>, 2001, p. 20.

BENNET, S.; McROBB, S.; & FARMER, R. Object-Oriented systems analysis and design using UML. 4th Edition, McGraw-Hill Higher Education, UK. 2010, p. 688.

CILESIZ, Sebnem. A Phenomenological Approach to Experiences with Technology: Current State, Promise, and Future Directions for Research. Educational Technology Research And Development , 59(4), En: ERIC, Ipswich, MA. Fecha de Consulta: Septiembre 11, 2012. pp. 487-510.

CONGRESO DE LA REPÚBLICA DE COLOMBIA, Ley 115 de 1994 (Febrero 8). Ley General de Educación. En: Diario oficial, Número 41214 del 8 de febrero de 1994.

DENZIN, N.K. & LINCOLN, Y.S. Handbook of qualitative research. London, UK: SAGE. 1994.

DUNNING, J. Regions, globalization, and the knowledge-based economy: Oxford: Oxford University Press, 2002. p. 506, University of Liverpool Catalogue - EBSCOhost, Fecha de consulta: agosto 26, 2012.

ENRÍQUEZ, Milian Guillermo. "La filosofía intercultural andina en la posmodernidad y la globalización". En: Cuadernos de Filosofía Latinoamericana 28.97, 2007, pp. 11-19.

ESTELL, J. & CHRISTENSEN, K. "The need for a new graduation rite of passage", Communications of the ACM. February 2011. 54(2). 2011, pp. 113-115.

FROMM, Erich. The revolution of hope, toward a humanized technology. New York: Harper and Row, 1968, p. 160.

HABERMAS, Jürgen. Knowledge and human interests. Toronto, Canada: Beacon Press, 1972, p. 368.

HORKHEIMER, Max. "Traditional and critical theory", En: Critical theory: selected essays, 1937, pp. 188-243.

HUSSERL, Edmund. Cartesian meditations: an introduction to phenomenology (D. Cairns, Trans.). Alemania: Kluwer Academic Publishers, 1982, p. 157.

IBM, Rational Method Composer. En Internet: <http://www-306.ibm.com/software/awdtools/rmc/index.html>, 2006, Fecha de Consulta: 11 de septiembre de 2012.

JIMÉNEZ, Robinson; MARTÍNEZ, Álvaro & INSUASTI, Jesús. Lineamientos para una didáctica en computación – Caso: Curso de Diseño de Software, Alemania: Editorial Académica Española, en colaboración con AV Akademikerverlag GmbH & Co., 2012, p. 153.

KOSIK, Karel. "Praxis and totality", en: Didactics of the concrete: a study on problems of man and world, Volumen 53. USA: Reidel Publishing Company, 1976, p. 176.

MARX, Karl. Theses on Feuerbach. 1st publication 1924. Moscow: Institute of Marxism-Leninism, 1845.

NAUR, P., & RANDELL, B. Software engineering: report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968. Brussels: Scientific Affairs Division, NATO, 1969, p. 231.

OMG Inc. The object management group home page. 2007. En Internet: <http://www.omg.org/> Fecha de consulta: septiembre 5 de 2012.

SOMMERVILLE, I. Software Engineering, 9th Edition. Boston, MA: Pearson Education/Addison-Wesley, 2011, p. 773.

TAN, G., & VENABLES, A. "Survival mode: the stresses and strains of Computing Curricula Review", Journal Of Information Technology Education, (enero 2008): IIP33-IIP43, Education Research Complete, EBSCOhost, Fecha de consulta: agosto 26, 2012.

TEDRE, Matti. "Know your discipline: teaching the philosophy of computer science", Journal of Information Technology Education, 6 (1), En: ERIC, Ipswich, MA. Fecha de consulta: septiembre 11 de 2012, 2007, pp. 105-122.

USCÁTEGUI DE JIMÉNEZ, Aura Mireya; GOYES MORENO, Isabel. "Reformas educativas y currículos universitarios". En: Colombia Itinerantes Revista del Área de Pedagogía, Currículo y Didáctica del Doctorado en Educación de RUDECOLOMBIA, Editorial Universidad del Cauca, v.1 fasc. 2002, pp. 89-99.