

Experiencias de las E-actividades de Evaluación de las Competencias del nivel básico del área de Programación

Experiences of the E-activities for Evaluation of the Competences of the basic level in the área of Programming

Cerón Garnica, C.¹, Archundia Sierra, E.², Cervantes Márquez, A.P.³, Beltrán Martínez, B.⁴

¹Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla.
Av. 14 sur y Av. Sn. Claudio. s/n. C.U. Puebla, México

²Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla
Av. 14 sur y Av. Sn. Claudio. s/n. C.U. Puebla, México.

³Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla
Av. 14 sur y Av. Sn. Claudio. s/n. C.U. Puebla, México.

⁴Facultad de Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla
Av. 14 sur y Av. Sn. Claudio. s/n. C.U. Puebla, México.

¹mceron@cs.buap.mx, ²etelvina@cs.buap.mx, ³pattyc@cs.buap.mx, ⁴beltran@cs.buap.mx

Fecha de recepción: 19 de junio 2018

Fecha de aceptación: 23 de agosto 2018

Resumen. Las tendencias curriculares internacionales han producido modificaciones en los Planes y Programas de Estudio en las carreras de las Ciencias de la Computación dando énfasis a la evidencia del desarrollo de las competencias del perfil de egreso evaluadas por los organismos acreditadores y la movilidad académica internacional para asegurar la calidad de los mismos. El propósito de este trabajo es presentar la reflexión de los escenarios, e-actividades y herramientas usadas en las asignaturas del área de la programación que han impactado en el desarrollo de las competencias. La metodología utilizada con un enfoque cuantitativo y un diseño cuasi-experimental y longitudinal aplicado a una muestra de estudiantes de la generación 2016 de la Ingeniería en Cs. de la Computación. Finalmente, se presentan los resultados del diagnóstico del desarrollo de las competencias del área de programación y recomendaciones para apoyar el nivel curricular del programa en Ciencias de Computación.

Palabras Clave: Competencias Educativas, Programación, Actividades de Aprendizaje, Pensamiento Computacional.

Summary. The international curricular tendencies have produced modifications in the Plans and Programs of Study in the careers of the Sciences of the Computation giving emphasis in the evidence of the development of the competencies of the profile of graduation evaluated by the accrediting organisms and the international academic mobility to assure the quality of them. The purpose of this project is to present the reflection of the stages, e-activities and tools used in the subjects in the area of programming that have impacted in the development of competencies. The methodology used with a quantitative approach and a design quasi-experimental and longitudinal applied to a sample of students of the 2016 generation of the Engineering in Sciences of the Computation. Finally, are presented the results of the diagnosis of the development of the competencies in the area of programming and recommendations to support the curricular level of the program in Computer Sciences.

Keywords: Educational Competences, Programming, Learning Activities, Computational Thinking.

1 Introducción

Las Instituciones de Educación Superior (IES) han planteado las modificaciones de los planes y programas de estudio con la inclusión de las competencias en los currículos de acuerdo a las necesidades sociales, económicas y laborales. Así como las tendencias internacionales académicas son validar los programas con calidad para aumentar la movilidad académica, dando lugar a una necesidad de las IES de internacionalizar sus programas de estudios. La educación basada en competencias se enfatiza en los saberes usando el objeto de conocimiento para la solución de problemas, siendo una finalidad de las IES lograr que los egresados en las diferentes áreas cumplan con un mínimo de competencias genéricas, disciplinares y específicas, requeridas para la inclusión en el sector productivo del país.

En la Facultad de Ciencias de la Computación (FCC) los programas en plan cuatrimestral están acreditados por el Consejo Nacional de Acreditación de Informática y Computación A.C. (CONAIC), con base a ellos se crearon los planes de estudios a semestres en el año 2016, bajo un enfoque educativo por competencias, articulando los saberes básicos, formativos y optativos. Se consideró importante el uso del Aprendizaje Basado en Problemas (ABA) en escenarios virtuales, a través de las Tecnologías de la Información y Comunicación (TIC).

1.1 Antecedentes

De acuerdo a los resultados y estadísticas del plan de cuatrimestre comprendido del 2009-2016 dentro de los problemas identificados en los estudiantes se encontraron: bajo rendimiento académico y el impacto del

desarrollo de las competencias que afectan a los indicadores de reprobación, deserción y avance de alumnos en las materias de programación de la Ingeniería en Ciencias de la Computación. Se detectó que en el cuarto cuatrimestre la materia con mayor porcentaje de reprobación fue la materia de Estructura de Datos con un 46%, perteneciente al área de Programación, como se muestra en la Figura 1.

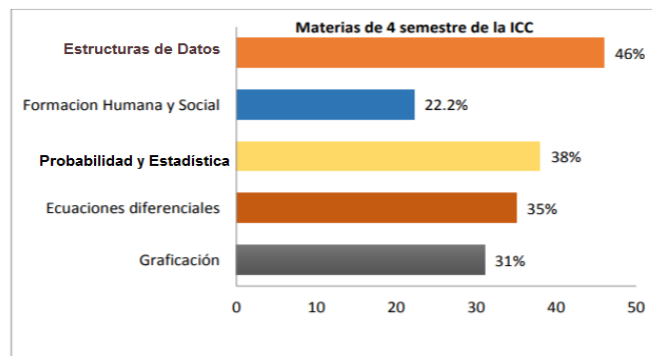


Figura 5. Porcentaje de reprobación de materias (Fuente: Secretaría Académica de la FCC).

Con base en lo anterior, los nuevos programas de estudio con enfoque en competencias y plan semestral (ver Figura 2) se diseñó como estrategia para los estudiantes de ingreso otoño 2016, un conjunto de e-actividades para dar seguimiento a su aprendizaje y desarrollo de competencias en el área de la programación usando herramientas y actividades mediante el uso de Escenarios de Aprendizaje para facilitar el aprendizaje autónomo, el pensamiento computacional y fortalecer la autoestima de los estudiantes.

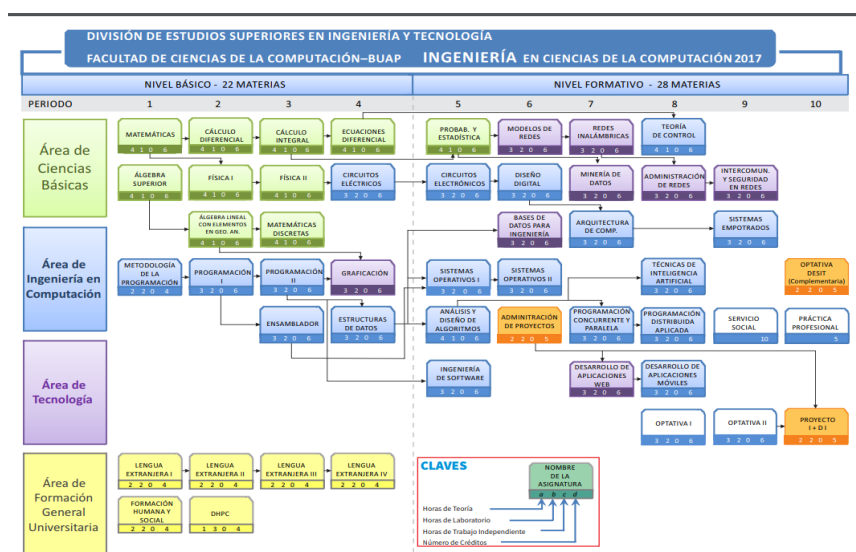


Figura 6. Mapa Gráfico de la Ing. en Cs. de la Computación por semestre.

El propósito de este estudio es analizar y reflexionar sobre las e-actividades y herramientas usadas en el área de la programación que han impactado en el desarrollo de las competencias genéricas y disciplinares en los alumnos en el área de la programación. La metodología que se aplicó fue cuantitativa y el diseño cuasi-experimental y longitudinal con un grupo de control y un grupo experimental, en este último se realizó una intervención educativa con los Escenarios de Aprendizaje. Finalmente se presentan los resultados sobre la evaluación del desarrollo de las competencias disciplinares de la asignatura del punto nodal, Estructuras de Datos, como parte de la autoevaluación y del seguimiento de las competencias del Perfil de Egreso en la Ing. en Cs. de la Computación.

El artículo está organizado de la siguiente forma: en la sección 2 se describen los trabajos relacionados en ambientes y escenarios virtuales de aprendizaje. En la sección 3 se presenta la metodología con enfoque cuantitativo y la prueba de pre-test y pos-test con respecto al desarrollo de las competencias de los estudiantes. En la sección 4 se muestra los resultados obtenidos de las pruebas y por último en la sección 5 se describen las conclusiones y trabajos futuros de la investigación.

2 Fundamentación

En esta sección se describen investigaciones y trabajos en áreas relacionadas con ambientes virtuales de aprendizaje y la evaluación por competencias alineada a las dimensiones del proyecto Tuning de América Latina.

2.1 Escenarios o Ambientes Virtuales de Aprendizaje

Para Ramírez y Burgos [1] un ambiente de aprendizaje enriquecido con tecnología “*permite ofrecer al educador nuevas formas de enseñar y reflexionar sobre su propia práctica educativa, permitiéndole facultar al estudiante en el uso de dicha tecnología para estimular el proceso de aprendizaje*”. Por lo que un escenario de aprendizaje constituye un espacio propicio para que los estudiantes obtengan recursos informativos y medios didácticos para interactuar y realizar actividades encaminadas a metas y propósitos educativos previamente establecidos.

De acuerdo con la experiencia realizada en ingeniería de Universidad de Mar de Plata [2] usando los Ambientes de Aprendizaje Virtuales (AVA) como un aula extendida, recreados en la plataforma Moodle con técnicas de Minería de Datos y el uso de AVA 3D, lograron que los estudiantes desarrollaran mejores habilidades en la solución de problemas y un aprendizaje autónomo, utilizando la técnica de Aprendizaje Basado en Problemas (ABP) que permitió reproducir fenómenos en laboratorios y ser explorados por los estudiantes. Según Martínez [3] los entornos 3D tienen el potencial para desarrollar prácticas que apoyan el aprendizaje del estudiante, de manera fácil y que comparte la experiencia en el entorno. Por otra parte, los entornos virtuales 3D, como por ejemplo Second Life y OpenSim, son comunidades online que simulan espacios físicos en tres dimensiones, similares a la realidad y que permiten a los usuarios, a través de sus avatares, interactuar entre sí y con el entorno, y utilizar, crear e intercambiar objetos. Según Totter [4] se trata de entornos inmersivos, interactivos, personalizables, accesibles y programables.

Las aportaciones de Sparrow, Blevins y Brenner [5] (2011), afirman que: “Los entornos 3D permiten el diseño de actividades didácticas prácticas, contextualizadas y basadas en problemas reales que pueden ser introducidas en la formación impartida en las facultades, de manera colaborativa por los estudiantes, pero requiere de la delimitación de las competencias y de las actividades a desarrollar en estos entornos”. Lo cual conlleva a que utilicen el aprendizaje basado en evidencias, elementos claves para medir el nivel de desempeño o logro de las competencias y un aprendizaje por experiencias.

En términos generales se pueden distinguir cuatro elementos esenciales: herramientas o medios de interacción, el proceso de interacción, la comunicación entre los sujetos o usuarios, contenidos y espacios donde se llevan a cabo las actividades o interacciones. Es por eso que las nuevas tecnologías cumplen dos funciones básicas especialmente vinculadas con el aprendizaje: *la mediación cognitiva y la aceleración de la percepción estímulos sensoriales*. Con los AVA 3D permiten diseñar distintos escenarios de interacción, logrando simulaciones que reflejan contextos reales y significativos para el aprendizaje. En este sentido los estudiantes interactúan de diversas formas en el escenario (alumno-alumno, alumno-profesor, alumno-contenidos/recursos) según su preferencia o estilo de aprendizaje, lo cual es esencial para un aprendizaje significativo. Pensamiento computacional y herramientas de apoyo

El pensamiento computacional propuesto por Wing [6], es una metodología basada en la implementación de los conceptos básicos de las ciencias de la computación para resolver problemas cotidianos y realizar tareas rutinarias. De tal manera que los problemas son resueltos mediante la ayuda de las computadoras aplicando la algoritmia. Wing describe con detalle las características y propiedades del pensamiento computacional siendo las siguientes: reformular un problema a uno parecido que sepamos resolver por reducción, encuadrarlo, transformar, simular, pensar recursivamente, procesar en paralelo, interpretar códigos de datos. Así como generalizar, reconocer ventajas, desventajas del solapamiento, juzgar un programa por simplicidad de diseño; utilizar la abstracción y descomposición en un problema complejo para su representación de manera fácil.

Por otra parte, el ISTE [7] lo define como “*un enfoque para resolver un determinado problema que empodera la integración de tecnologías digitales con ideas humanas, no reemplaza el énfasis en creatividad, razonamiento o pensamiento crítico, pero refuerza esas habilidades al tiempo que realza formas de organizar el problema de manera que la computadora pueda ayudar*”. Dentro de las herramientas de apoyo que más se utilizan para el desarrollo del pensamiento computacional algorítmico y programación encontramos a: Scratch, que emplea bloques gráficos, donde cada bloque representa un elemento del lenguaje de programación, como son: estructura de control, operadores, variables, funciones, etc., mientras Alice y Greenfoot permiten solucionar problemas y representarlos de forma más sencilla, introduciendo conceptos de programación orientados a objetos. Así también, se modelan soluciones que el estudiante experimenta sin tener un trasfondo de programador lo que ha llevado a generar que los estudiantes utilicen en distintos niveles educativos el uso de estas herramientas computacionales. Otras herramientas online, como CodeStudio apoyan en comunidad a aprender programación y PseInt, que es una herramienta simple e intuitiva para el uso de pseudo-lenguaje el cual aporta fundamentos algorítmicos.

2.2 Evaluación y Competencias disciplinares en la Informática o Ciencias de la Computación

La evaluación de programas y cursos de formación es una de las actividades más relevantes y significativas en todo proceso de gestión y planificación de la educación formal. Por lo cual se considera a la evaluación como: “un proceso sistemático de información, no improvisado, que requiere organizar sus fases, temporalizar sus secuencias, administrar los recursos, construir o seleccionar los instrumentos, técnicas y métodos, así como criterios que implican un juicio de valor orientados hacia la toma de decisiones” [8].

La evaluación por competencias o evaluación del desempeño afirma Frade, que “implica que tengamos dos pies cuando evaluamos, la evaluación formativa (de proceso) y la evaluación sumativa (de producto)”, los cuales se comunican y generan los aprendizajes logrados. Por otra parte, Argudín [9] considera que este enfoque de Educación Basado en Competencias es holístico, trata de integrar las experiencias de la vida real, con un propósito de desarrollar habilidades mediante tareas y funciones básicas donde se domine los conocimientos, habilidades, actitudes y valores que determinan el logro de la competencia mientras que Perrenoud define la competencia como “la capacidad de movilizar recursos cognitivos para hacer frente a un tipo de situaciones con buen juicio, a su debido tiempo, para definir y solucionar verdaderos problemas” [10]. Esto implica tres elementos sustantivos en la competencia: situaciones, los recursos cognitivos y soluciones reales. Según Ordóñez [11] identifica algunos beneficios por aplicar un modelo basado en competencias en las instituciones: desarrolla personas competitivas, saberes en los alumnos y ayuda a garantizar la calidad en los egresados.

En el 2013, el Proyecto de ALFA Tuning de América Latina [12] muestra los resultados obtenidos de los países formadores de las carreras en Informática agrupando a la licenciatura e ingeniería en: Sistemas, Informática, Computación, Ciencias Informáticas, Ciencias de la Computación, Redes Informáticas, Sistemas Computacionales, Tecnologías Computacionales, Sistemas e Informática, Sistemas y Computación, Software, Redes y Telecomunicaciones, Informática Administrativa, entre otras. La duración de las carreras varía entre 8 y 12 semestres, siendo 10 semestres la más frecuente. La población que estudia dichas carreras es de 413,577 estudiantes, los datos muestran que Brasil y México aportan más de la mitad (55%). México tiene 106,000 estudiantes en las carreras de informática, es decir, el 26% y Brasil tiene el 29% de la demanda de esta carrera.

Así también, se definieron las dimensiones de las competencias de la carrera de Informática en: aspectos disciplinares, ejercicio de la profesión y la responsabilidad social [12].

1. En la dimensión de aspectos disciplinares están agrupadas en 4 áreas:
 - Fundamentos de informática: se enfoca en
 - Aplicar el conocimiento de ciencias de la computación, de tecnologías de la información y de las organizaciones, para desarrollar soluciones informáticas.
 - Aplicar el enfoque sistémico en el análisis y resolución de problemas.
 - Gestión y liderazgo: se enfoca en
 - Desempeñar diferentes roles en proyectos informáticos, en contextos multidisciplinarios y multiculturales, tanto locales como globalizados.
 - Asimilar los cambios tecnológicos y sociales emergentes.
 - Comprender y aplicar los conceptos éticos, legales, económicos y financieros en la toma de decisiones y en la gestión de proyectos informáticos.
 - Calidad e innovación: se enfoca en
 - Concebir, diseñar, desarrollar y operar soluciones informáticas basándose en principios de ingeniería y estándares de calidad.
 - Aplicar estándares de calidad en el desarrollo y evaluación de soluciones informáticas.
2. La Dimensión del ejercicio profesional enmarca las competencias:
 - Capacidad de aplicar los conocimientos en la práctica.
 - Capacidad de abstracción, análisis y síntesis, capacidad de identificar, plantear y resolver problemas.
 - Capacidad de aprender y actualizarse permanentemente.
 - Conocimiento sobre el área de estudio y la profesión.
 - Capacidad de investigación.
 - Capacidad para organizar y planificar el tiempo.
 - Capacidad para formular y gestionar proyectos.
 - Capacidad para trabajar en equipos.
 - Habilidad para trabajar en contextos internacionales.
 - Capacidad de comunicación en un segundo idioma.
3. La Dimensión de responsabilidad social expresa la relación esperada del profesional con el contexto socio-cultural, ético y medio-ambiental de su espacio de actuación. Las competencias de esta dimensión son:
 - Compromiso ético.
 - Responsabilidad social y compromiso ciudadano.
 - Compromiso con la preservación del medio ambiente.
 - Compromiso con su medio socio-cultural.
 - Valoración y respeto por la diversidad y multiculturalidad.

Por lo cual, las competencias profesionales de la Ing. en Cs. de la Computación consideradas para el Plan Semestral son ocho (8) y clasificadas de acuerdo a las dimensiones del Meta-Perfil del Área de Informática del Proyecto Tuning de América Latina, se observan en la Tabla 1.

Tabla 1. Alineación de las dimensiones de Tuning y competencias del perfil.

Dimensión	Competencias Profesionales del Perfil del Ing. en Cs. de la Computación
Aspectos Disciplinarios	2. Diseña soluciones de sistemas de cómputo soportadas en modelo de procesos, metodologías y herramientas para resolver problemas.
<i>Fundamentos de Informática</i>	4. Integrar elementos de software en la construcción de soluciones aplicando modelos matemáticos que permita utilizar eficientemente los recursos de hardware.
<i>Calidad e Innovación:</i>	1. Aplica la metodología para la gestión de proyectos adecuada que permita la creación de productos de calidad. 5. Diseña soluciones creativas e innovadoras por medio del análisis, síntesis e implementación en sistemas de cómputo que cumplan con los estándares de calidad.
<i>Gestión y Liderazgo</i>	8. Analizar los principales problemas de su área, identificando los conocimientos necesarios y herramientas adecuadas para proponer soluciones y divulgar los resultados obtenidos.
Ejercicio Profesional	3. Resuelve problemas de automatización y control de procesos a través del uso de conocimientos de matemáticas, software y hardware en el funcionamiento en un entorno interdisciplinario. 7. Aplica los avances tecnológicos más eficientes en el área de desarrollo de aplicaciones de software, tratamiento de datos, redes de computadoras y sistemas empotrados, control digital y robótica con el fin de proponer soluciones innovadoras a problemas científico-tecnológico del país.
Responsabilidad Social:	6. Interactuar con el usuario y atendiendo sus necesidades con el fin de generar soluciones en su competencia

Con base en lo anterior se definen las competencias genéricas y disciplinares a evaluar del Área de Programación del Perfil de Ing. en Cs. de la Computación (ver Tabla 2).

Tabla 2. Competencias en el área de Programación.

Tipo de Competencia	Competencias a Evaluar	Materias
Genéricas	C1-Trabajo en equipo C2- Capacidad de resolución de problemas C3-Aprendizaje Autónomo	-Metodología -Programación I -Programación II
Disciplinares	C4- Diseña soluciones de sistemas de cómputo soportadas en modelo de procesos, metodologías y herramientas para resolver problemas. C5-Aplicar el enfoque sistémico en el análisis y resolución de problemas (usando modelos matemáticos y hardware)	-Ensamblador -Estructuras de Datos

El perfil del egresado demuestra las competencias que tendrá al finalizar sus estudios, lo cual es un proceso que se va formando en el estudiante en cada una de las áreas, con respecto a la de Programación en la Tabla 3, se observa las competencias algorítmicas que se evalúan de acuerdo a las materias impartidas, las cuales aportarán al desarrollo de las competencias conforme el estudiante vaya cursándolas, siendo el nodo modular de control la materia de Estructuras de Datos en el nivel básico de formación, lo cual permite valorar los logros de desempeño alcanzados por los estudiantes a nivel curricular.

Tabla 3. Materias que contribuyen al desarrollo de las competencias.

Disciplinar Básica Programación	Competencias algorítmicas en el nivel básico <i>Fundamentos de Cs. de la Computación o Informática</i>				
	C1	C2	C3	C4	C5
Materia					
Metodología de la Programación	20%	20%	20%	20%	
Ensamblador	20%	20%	20%	20%	50%
Programación I	20%	20%	20%	20%	
Programación II	20%	20%	20%	20%	
Estructuras de Datos	20%	20%	20%	20%	50%
Total	100%	100%	100%	100%	100%

3 Metodología

La metodología utilizada con un enfoque cuantitativo, un diseño cuasi-experimental y longitudinal, ya que se aplicó a una muestra conformada de dos grupos de alumnos de la Generación 2016, que cursan actualmente el cuarto semestre de la Ingeniería en Cs. de la Computación. Ambos grupos conformados inicialmente por 45 alumnos, la sección 101 se considera el grupo experimental conformado por 17 mujeres y 28 hombres. La sección 102 se considera el grupo control, conformado por 15 mujeres y 30 hombres, con el cual se trabaja de manera tradicional. Para la evaluación se diseña una prueba de aptitud académica y el registro de actas finales la cual se aplicó para medir los saberes tanto del grupo de control y experimental, al final del cuarto semestre, cabe destacar que la sección 101 solo dos se dieron de baja mientras que en la sección 102 seis en dicho periodo. Ambos grupos presentan deserción, aunque la reprobación es mayor en el grupo de control, por lo que finalmente la muestra se conformó de 15 estudiantes de cada grupo a la cual se les aplicó en cada semestre una evaluación de seguimiento del desarrollo de las competencias de acuerdo a las competencias genéricas y disciplinares enunciadas de cada asignatura.

3.1 Diseño y experiencia en escenarios de aprendizaje

La intervención educativa se realiza a la sección 101 en un periodo del 2016-2018, mediante el uso de Escenarios de Aprendizaje en 2D y 3D. El diseño conceptual de los escenarios, las actividades y las secuencias didácticas se utilizó el modelo de diseño centrado en la evidencia [13] que comprende las siguientes fases:

- Fase 1. Competencias: se definen las competencias genéricas y disciplinares del área de programación
- Fase 2. Escenarios: Se definen los espacios de interacción y las situaciones para el aprendizaje (alumno-docente, alumno-alumno, alumno-contenidos/recursos educativos). Para lo cual se brindaron espacios virtuales afines a los de la Facultad, logrando escenarios generales como: *Escenario 1: aula de proyección*, *Escenario 2: laboratorio de simulaciones y prácticas* y *Escenario 3: Biblioteca digital*, espacio de manejo de recursos digitales, de asesorías y para trabajar en equipo. Los escenarios diseñados primeramente en VRML y usando Moodle, después se integraron OpenSim y Sloodle.
- Fase 3. e-actividades: se refiere al diseño de las actividades online, utilizando la Taxonomía de Marzano para alcanzar el desarrollo de las competencias, teniendo las evidencias de los aprendizajes esperados como son: e-portafolios (prácticas de laboratorio), e-solución (problemario) y e-proyectos (resolución de problemas y aplicaciones). Cada una se describe detalladamente las características y tareas a realizar en cada escenario. Así, como su forma de evaluación mediante rubricas, listas de cotejo, autoevaluación y co-evaluación. Para las e-actividades el desarrollo del pensamiento computacional y programación, se enfocó en que el estudiante adquiriera ciertas habilidades: construir un algoritmo, resolver un problema planteado, traducirlo a un lenguaje de programación específico y probarlo en la computadora. Las herramientas utilizadas fueron: Scratch, Pseint, Alice y Greenfoot en las materias de Metodología de la Programación, Programación I, y II para facilitar los saberes conceptuales, procedimentales y actitudinales. Para Ensamblador y Estructuras de Datos se utilizaron objetos de aprendizaje en la plataforma.

A continuación, se presentan el diseño de los escenarios similares a la realidad de tal forma que el estudiante pueda navegar de manera natural con respecto a los objetos existentes. En la Figura 3, se muestra el diseño del edificio CCO3-FCC para interactuar con los escenarios de aprendizaje, los estudiantes pueden visitar y recorrer el edificio para trabajar en las actividades diseñadas en cada asignatura.



Figura 3. Ambiente real y virtual del edificio de la facultad.

En el Escenario 1, el estudiante interactúa en el aula que dispone de las secuencias didácticas, recursos, y materiales. Al ingresar al aula, selecciona si desea trabajar con el Moodle para revisar las clases, generar sus evidencias de aprendizaje o pide una asesoría virtual por medio del chat o participa en los foros, logrando promover un aprendizaje autónomo y reflexivo.

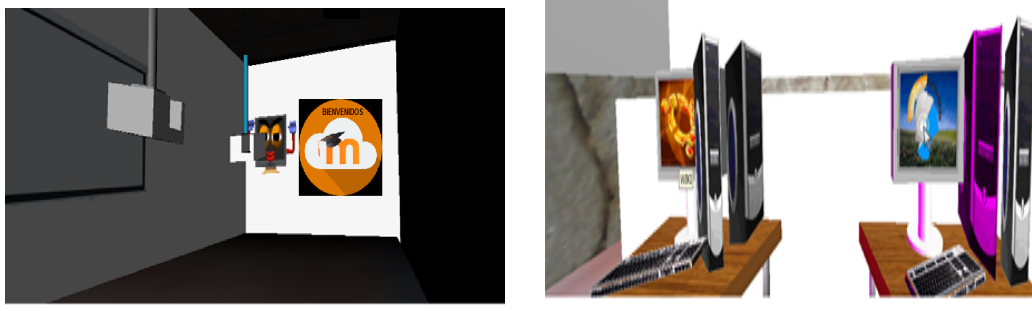


Figura 4. Escenario 1 para interactuar en aula de proyecciones con las secuencias didácticas virtual.

En el Escenario 2, el estudiante interactúa en el laboratorio de simulaciones y prácticas que debe realizar, selecciona las herramientas de trabajo como: Pseint, Alice y Greenfoot. Para las e-actividades: e-soluciones y e-proyectos, el estudiante usa la estrategia de aprendizaje ABP e interactúa con algún lenguaje de programación para realizar soluciones informáticas y enfatiza el trabajo en equipo (Ver Figura 5).

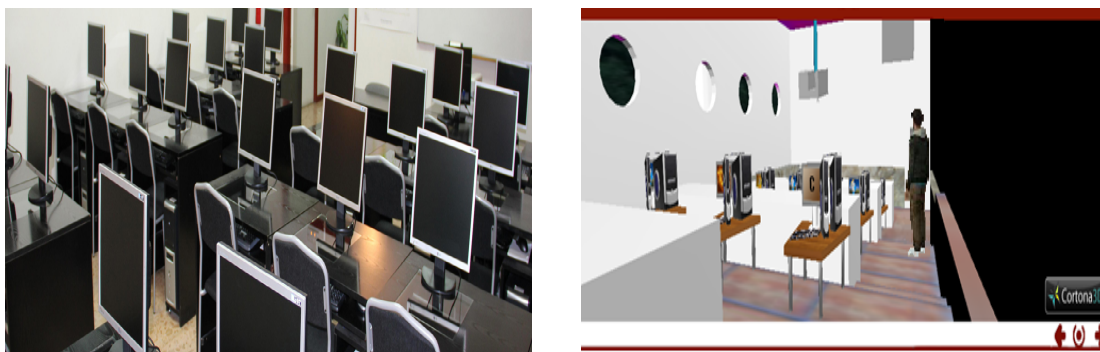


Figura 5. Escenario 2 de Laboratorio de simulaciones y prácticas real y virtual.

El Escenario 3, permite al estudiante interactuar con recursos digitales de varias disciplinas, consultar materiales, puede reservar una mesa o equipo de cómputo para trabajar en equipo, participar en foros y chat, buscar información en internet, y preparar con sus compañeros una propuesta de trabajo e incluso interactuar con otros docentes y solicitar asesorías del docente (Ver Figura 6).

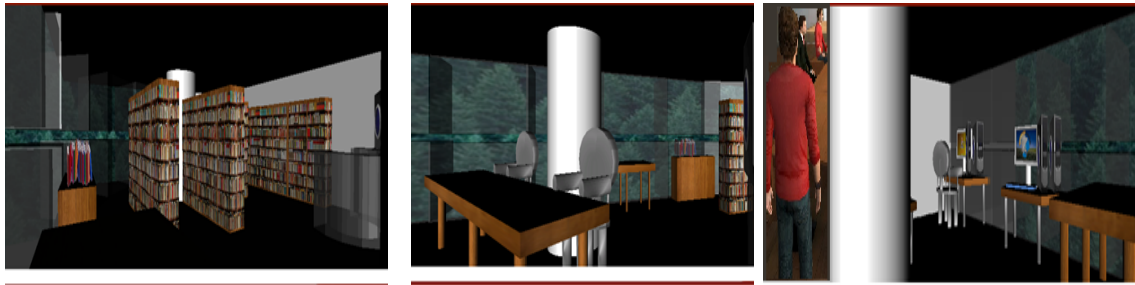


Figura 6. Escenario 3 de Biblioteca digital, recursos educativos y área de trabajo en equipo virtual.

Con respecto a otros escenarios poder interactuar con utilizar simuladores e incluso a otros cursos de MOOC que permitió mayor dinamismo para solucionar problemas (ver Figura 7).

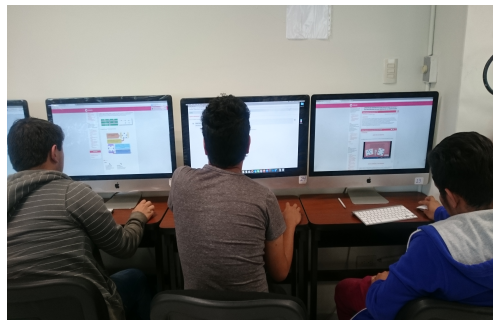


Figura 7. Uso de simulaciones en Estructuras de Datos con el grupo experimental.

4 Resultados

Uno de los objetivos de este estudio es mostrar los resultados obtenidos de su rendimiento académico (en actas) con respecto al grupo experimental después de la intervención y la comparación con el grupo de control al final del curso, los cuales fueron evaluados en el periodo comprendido de otoño 2016 a primavera 2018, como se observa en la Figura 6, el grupo experimental tiene un promedio de rendimiento académico de 8.6 mientras que el de control de 7.7 lo que conlleva a firmar que el nivel de dominio en la solución de problemas se está desarrollando el grupo experimental sección 101, obteniendo aprendizajes profundos y estratégicos parte esencial del pensamiento computacional, de acuerdo a la experiencia y comentarios de los estudiantes los escenarios principalmente propiciar el desarrollo en el área de programación.

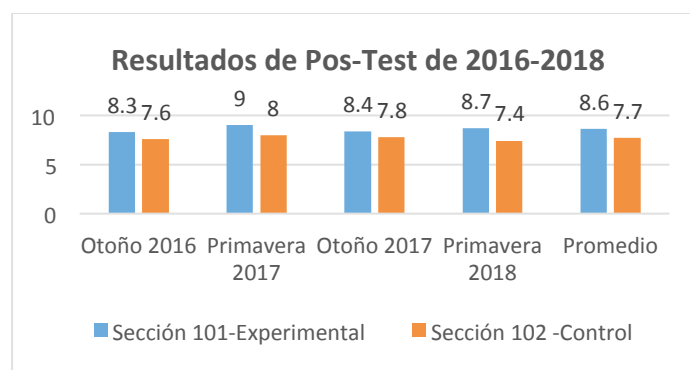


Figura 6. Promedio de las materias del área de programación del 2016-2018

Con respecto a las competencias definidas en la Tabla 4, de acuerdo a la escala de Likert se consideró tres niveles de desempeño de las competencias al perfil: Principiante, Intermedio y Avanzado.

Para evaluar los contenidos se maneja en tres puntajes: 1) Bajo, menor a la media, 2) Regular, dentro de la media y 3) Alto, superior a la media, lo cual permite llevar el seguimiento académico del proceso de aprendizaje. Logrando identificar las competencias que se aplicaron al área de Programación en el nivel básico:

- C₁-Trabajo en equipo
- C₂- Capacidad de resolución de problemas.

- C₃-Aprendizaje Autónomo
- C₄-Diseña soluciones de sistemas de cómputo soportadas en modelo de procesos, metodologías y herramientas para resolver problemas
- C₅-Aplicar el enfoque sistémico en el análisis y resolución de problemas

A partir de lo anterior, se analiza el nivel de desarrollo de las competencias de los estudiantes del grupo experimental (15 alumnos) donde se muestra que la mayoría se encuentra en proceso del desarrollo de sus competencias, siendo la C₄, la que alcanzó un 70.5%, lo que nos indica que el estudiante es capaz de diseñar soluciones y el pensamiento computacional se está aplicando mientras que la C₁ se debe fortalecer para que logre su desarrollo. Asimismo, se observa un porcentaje mínimo de estudiantes que su competencia es baja, lo cual se requiere trabajar estrategias para apoyar su formación, tal como como se observa en la Tabla 4.

Tabla 4. Nivel de las competencias del Grupo Experimental nodo modular Estructuras de Datos.

Competencia	Iniciada	En Proceso	Desarrollada
	Bajo	Regular	Alto
Competencia ₁	15%	53%	32%
Competencia ₂	16.6%	63.4%	20%
Competencia ₃	12%	56.5%	31.5%
Competencia ₄	8%	70.5%	21.5%
Competencia ₅	12%	56.5%	31.5%

5 Conclusiones y trabajos futuros

Una de las principales contribuciones de estudio es reflexionar sobre el desarrollo de las competencias en el área de programación de la Ingeniería en Cs. de la Computación, logrando aprendizajes y evidencias en el diseño de algoritmos, modelar soluciones de problemas, y el uso de un lenguaje de programación específico, esto implicó un seguimiento en un grupo y una intervención por semestre, proporcionando al estudiante un aprendizaje contextual y real para fundamentar un pensamiento computacional mediante el uso de escenarios de aprendizaje logrando soluciones computacionales de manera innovadora y creativa que se busca ir formando en el Ingeniero en Cs. de la Computación.

Es importante aclarar, que los docentes en general del área están acostumbrados a trabajar con un enfoque tradicional, desaprovechando las habilidades digitales nativas del estudiante. Por otra parte, el uso de escenarios 2D, 3D, plataformas y herramientas digitales facilitan el aprendizaje y el trabajo docente. Una de las recomendaciones es establecer estrategias que apunten las competencias mediante talleres, seminarios y cursos on-line como los MOOC, que pueden acompañar en su proceso de aprendizaje al estudiante, logrando escenarios dinámicos y diversos. Asimismo, se requiere en los docentes una formación permanente de estrategias tecno-pedagógicas actuales.

La evaluación de competencias requiere identificar si se está logrando el desarrollo de las competencias, la propuesta de un nodo evaluador fue la materia de Estructuras de Datos para verificar si se están logrando en el nivel básico del área de programación del perfil de egreso de Ing. en Cs. de la Computación. El evaluar en un nodo permite identificar que estudiantes requieren apoyo y atención en su formación, esto conlleva a que se debe establecer un nodo de evaluación curricular interna y externa para poder valorar las competencias y tomar decisiones que fortalezcan el diseño curricular y las estrategias de mejora en la calidad educativa.

Así también se analizó la alineación de las competencias del perfil profesional del Ing. en Cs. de la Computación con el meta perfil del Proyecto Tuning de América Latina para las carreras de informática, donde se pudo identificar que el perfil del ingeniero de la FCC-BUAP se inclinan más a las competencias disciplinares que a las profesionales empresariales. Lo cual implica que se requieren enfocar las competencias para el ejercicio de la profesión para que pueda insertarse al sector productivo de acuerdo de los empleadores.

La perspectiva de la investigación es realizar el análisis completo y estadístico de las competencias del plan semestral de la carrera de ingeniera y de las otras licenciaturas ofertadas en la unidad académica, usando un nodo de evaluación cuyos resultados cuantitativos y cualitativos permitan llevar a cabo las actualizaciones y modificaciones curriculares sustentadas, dando lugar a fortalecer el perfil de egreso de acuerdo a las necesidades profesionales de los empleadores y las tendencias académicas internacionales.

Agradecimientos

Agradecemos a la Vicerrectoría de Investigación y Estudios de Posgrado de la BUAP, por el apoyo al trabajo de investigación y recursos otorgados. Al CONAIC por su valoración de los programas, al Cuerpo Académico CA-277 y a la Facultad de Ciencias de la Computación (BUAP).

Referencias

1. Ramírez, M.; Burgos, A.J.V.: Recursos Educativos Abiertos en Ambientes Enriquecidos con Tecnología: Innovación en la Práctica Educativa, Tecnológico de Monterrey, México. <https://repositorio.itesm.mx/handle/11285/573608> (2010). Accedido el 18 de Septiembre de 2017.
2. Huapaya, C.; Lizarralde, F.; Arona, G.; Vivas, J., Massa, S.; Bacino, G; Rico, G.; Evans, F.: *Uso de Ambientes Virtuales de Aprendizaje en la Enseñanza de la Ingeniería. XVIII Congreso Argentino de Ciencias de la Computación. Red de Universidades con Carreras en Informática.* <http://hdl.handle.net/10915/23651> (2012). Accedido el 15 de Agosto de 2017
3. Martínez, R.: *Conexión de entornos de aprendizaje.* Barcelona, España: UDC, pp. 91-139 (2017)
4. Totter, E.; Raichman, S.: Creación de espacios virtuales de aprendizaje en el área Ciencias Básicas en carreras de Ingeniería. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología.* 4, pp. 40-46 (2009)
5. Sparrow, J. L., Blevins, S. J. y Brenner, A. M. (2011). Faculty development for and in virtual worlds. En R. Hinrichs y C. Wankel (Eds.): *Transforming virtual world learning*, United Kingdom, pp. 47-65 (2011)
6. Wing, J.: Computational thinking. *Communications of the ACM*, Vol. 49, No.3, pp. 33-35 <http://dx.doi.org/10.1145/1118178.1118215> (2006). Accedido el 15 de Agosto de 2017
7. ISTE: *Computational Thinking Leadership Toolkit*, first edition. Computer Science Teachers Association (CSTA) and International Society for Technology in Education (ISTE). <http://www.iste.org/docs/ct-documents/ct-leadershiptoolkit.pdf?sfvrsn=4> (2011). Accedido el 10 de Agosto de 2017.
8. Posavac, E.J.: *Program Evaluation. Methods and Case Studies*, Prentice Hall, New Jersey (1989)
9. Argudín, Y.: *Educación Basada en Competencias.* Editorial Trillas, México (2005)
10. Perrenoud, P.: *Diez nuevas competencias para enseñar.* Editorial Grao, México (2004)
11. Ordonez, B.: Competency-Based Education: Changing the Traditional College Degree Power, Policy, and Practice. *New Horizons In Adult Education & Human Resource Development*, Vol 26, No. 4, pp.47-53 (2014)
12. Lino, C. J.: Proyecto Tuning América Latina. Educación Superior en América Latina: reflexiones y perspectivas en Informática. Universidad de Deusto. http://tuningacademy.org/wp-content/uploads/2014/02/RefInformatics_LA_SP.pdf (2013) Accedido el 15 de Agosto de 2016.
13. Mislevy, R.J.: *Evidencecentered design for simulationbased assessment.* Los Ángeles: *The National Center for Research on Evaluation, Standards, and Student Testing* (CRESST). Los Ángeles: University of California. (2011)