ARTÍCULO PRODUCTO DE LA INVESTIGACIÓN

Diagnóstico y reestructuración pedagógica en la enseñanza de programación de computadores en ingeniería de sistemas

Paulo César Ramírez Prada [1]
Ariel Orlando Ortiz Beltrán [2]
Rene Alejandro Lobo Quintero [3]

Recibido: 05-05-2020
Aceptado: 30-06-2020

Resumen

La forma tradicional de enseñar programación ha resultado ser ineficiente a la hora de motivar a las nuevas generaciones de ingenieros para encontrar su camino profesional. En la Universidad Autónoma de Bucaramanga se tomó la decisión de crear una estrategia para resolver este problema, con el objetivo de mejorar las métricas de desempeño sobre los alumnos y profesores, así como orientar el currículo hacia las tendencias específicas de la

1. Ingeniero de Sistemas. Magíster en Gestión, Aplicación y Desarrollo de Software. Docente del programa de Ingeniería de Sistemas de la Universidad Autónoma de Bucaramanga (UNAB), Colombia. Miembro del Grupo de Tecnologías de Información (GTI), líneas de investigación: ingeniería de software, programación distribuida y arquitectura empresarial.
Correo electrónico: pramirez206@unab.edu.co
ORCID: http://orcid.org/0000-0001-5421-9890

2. Ingeniero de Sistemas. Magíster en Creación Multimedia. Docente del programa de Ingeniería de Sistemas de la Universidad Autónoma de Bucaramanga (UNAB), Colombia. Miembro del Grupo de Tecnologías de Información (GTI), líneas de investigación: inteligencia artificial, ingeniería de software y gamificación.
Correo electrónico: aortiz875@unab.edu.co
ORCID: http://orcid.org/0000-0003-1522-2362

1. Ingeniero de Sistemas. Magíster en Ingeniería de Sistemas y Computación. Docente del programa de Ingeniería de Sistemas de la Universidad Autónoma de Bucaramanga (UNAB), Colombia. Miembro del Grupo de Preservación e Intercambio Digital de Información y Conocimiento (Prisma), líneas de investigación: desarrollo móvil, ingeniería de software y gamificación.
Correo electrónico: rlobo@unab.edu.co
ORCID: http://orcid.org/0000-0003-2989-5357

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

disciplina (apps, IoT, web, IA y videojuegos) y, por último, hacer una revisión de las herramientas, plataformas y lenguajes de programación que estaban en tendencia a nivel mundial para incluirlos en el plan docente en conjunto con una metodología de gamificación experimental diseñada por los mismos docentes. Como resultado, las calificaciones generales mejoraron, las tasas de deserción disminuyeron y la población estudiantil en general aumentó.

Palabras clave: currículo, deserción, aprendizaje, ingeniería de sistemas.

Diagnosis and pedagogical restructuring in the teaching of computer programming in systems engineering

Abstract

The traditional way of teaching programming has proven to be inefficient in motivating new generations of engineers to find their career path. At the Autonomous University of Bucaramanga, the decision was made to create a strategy to solve this problem, with the aim of improving performance metrics on students and teachers, as well as orienting the curriculum towards specific trends in the discipline (apps, IoT , web, AI and video games) and, finally, to review the tools, platforms and programming languages that were in trend worldwide to include them in the teaching plan in conjunction with an experimental gamification methodology designed by the teachers themselves . As a result, overall grades improved, dropout rates decreased, and the overall student population increased.

Keywords: curriculum, dropout, learning, systems engineering.

Introducción

En el departamento de Santander, Colombia, es notable la falta de motivación de los jóvenes para estudiar la carrera de ingeniería de sistemas. De acuerdo al Ministerio de Educación: "el problema actual con los programas de ingeniería es que la demanda por profesionales ha ido en aumento, mientras que los bachilleres muestran una preocupante falta de apetencia por estudiarlos" (La República, 2010). La forma tradicional de enseñar programación de computadoras en general está bastante desfasada con las tecnologías y metodologías pedagógicas que se mantienen vigentes en la actualidad. Adicionalmente, la programación se enseña a una generación que es esencialmente diferente, caracterizada dentro del grupo de los llamados centennials, que están a la caza de una educación ágil, práctica, autónoma, ajustada a sus necesidades individuales. Otro factor importante es que, en las facultades colombianas de ingeniería de sistemas, los contenidos del programa académico no se corresponden con las tendencias tecnológicas globales actuales. Esto se reflejó durante varios años en la Universidad Autónoma de Bucaramanga, donde una gran cantidad de estudiantes abandonaron la escuela en los primeros semestres en busca de una que se ajustara más a sus ideales profesionales. Además, el número de estudiantes de primer ingreso era muy bajo mantentiendo una poblacion estudiantil en la carrera de poco más de 80 estudiantes (figura 1). Sin embargo, en el primer semestre de 2015 el Gobierno colombiano decidió invertir en las carreras de TIC facilitando el ingreso para estudiantes de bajos recursos a través de becas, lo que incrementó la poblacion; debido a este fenomeno se aumentó la necesidad de una reforma pedagógica de fondo para estar a la altura de los retos incipientes.

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

Figura 1. Número de estudiantes matriculados por semestre desde 2011.

AÑO	MATRICULADOS	
	Primer Ingreso	Total
2011-II	6	88
2012-I	16	88
2012-II	11	86
2013-I	20	86
2013-II	13	84
2014-I	12	83
2014-II	11	80
2015-I	72	137
2015-II	15	140
2016-I	98	219
2016-II	117	304
2017-I	75	358
2017-II	17	335
2018-I	39	364
2018-II	10	348

Fuente: elaboración propia.

El propósito de este proceso de desarrollo y reevaluación del currículo es adaptar el programa de Ingeniería de Sistemas de la Universidad Autónoma de Bucaramanga a la realidad tecnológica y metodológica internacional teniendo en cuenta las tendencias más demandadas por la industria y la academia locales, involucrando participativamente a todos los actores del proceso: alumnos, profesores y directivos.

Por lo tanto, se propuso una revisión de los objetivos y la filosofía del programa, se llevó a cabo un proceso de vigilancia tecnológica, identificando las tendencias que marcarían el enfoque práctico del mismo con el fin de iniciar la transición de la forma tradicional de enseñanza en la universidad, donde las clases se impartían principalmente en cátedras masivas, a una metodología orientada a proyectos. Desde los primeros semestres, los estudiantes convierten sus propias ideas en software real. Adicionalmente, se construyó una metodología flexible basada en educación por competencias y estrategias lúdicas para crear un flujo de motivación dentro de los cursos. Finalmente, también se definieron nuevos criterios de desempeño y evaluación para estudiantes y maestros en los cursos de programación.

Contexto

En las materias relacionadas con la ingeniería de sistemas y ciencias de la computación se han realizado varios esfuerzos por incluir tanto gamificación como juegos serios en los procesos de enseñanza de programación, logrando impactos muy positivos entre los estudiantes. Por ejemplo, Braulio Lambruschini y Waldy Pizarro (2015, pp. 295-299) presentan una estrategia pedagógica implementada en la Universidad San Martín de Porres de Lima (Perú), con la cual logran mejorar los niveles de atención, las calificaciones y la asistencia a clase; para ello mezclan un sistema de aprendizaje online llamado Schoology e incorporan elementos de gamificación como lo son: puntos de experiencia y niveles en las clases presenciales.

Gamificación y juegos serios

Gamification (en lengua española "gamificación" o "ludificación") sugiere en este sentido, a poder utilizar elementos del juego, y el diseño de juegos, para mejorar el compromiso y la motivación de los participantes. El concepto definido por Deterding, Dixon, Khaled y Nacke en el artículo *Gamification: Toward a Definition* en 2011, se refiere al uso de elementos de diseño de juegos en contextos que no son de juego. Hablamos de un campo relativamente nuevo pero con un rápido crecimiento. El concepto *gamification* es diferente a serious game: mientras que el segundo describe el diseño de juegos que no tienen el firme objetivo de entretener a los usuarios, las experiencias o ejemplos "gamificados" simplemente emplean algunos elementos de los juegos como reglas, mecánicas, etc. El primer uso y documentación del término se realizó en el año 2008, pero este no fue generalizado sino hasta el segundo semestre de 2010. Se presume que fue Nick Pelling quien introdujo el término mucho antes, en el año 2003, cuando escribió un trabajo como consultor para una empresa de fabricación de hardware. Sin embargo, el concepto en sí es no nuevo, por ejemplo, el uso

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

de "insignias" o "medallas" ha sido utilizado desde años atrás en el ejército. Un ejemplo de ello es la utilización de medallas por los líderes de la Unión Soviética, que entregaban estas a aquellos soldados que realizaba un buen trabajo, y como un sustituto de los incentivos económicos (Espinosa,Eguia 2016).

En los años recientes la gamificación ha empezado a tener mayor protagonismo en la educación, tanto en la educación media como en la superior; al hacer una revisión de la literatura se pueden encontrar que a partir del 2010 se empezó a utilizar el término ampliamente, definiéndolo inicialmente como "el uso de elementos de diseño de juegos en contextos externos" (Deterding, Dixon, Khaled y Nacke, 2011). Los proyectos encontrados, a pesar de pertenecer a múltiples disciplinas tales como artes, medicina, derecho, ingeniería, tienen en común el objetivo de motivar a los estudiantes a aprender.

Se pueden encontrar en las principales conferencias de educación a nivel mundial que el tema de la gamificación es aplicado en diferentes campos educativos. Por ejemplo, Bouki, Economou y Kathrani (2014) muestran un exitoso caso de implementación de gamificación para los estudiantes de derecho en el cual logran mezclar la metodología de análisis legal IRAC con elementos tales como tareas, preguntas, premios, motivaciones, elementos de rol. Para esto desarrollan una plataforma multimedia en la cual los estudiantes pueden ir conociendo detalles del caso a través de múltiples juegos.

También está el proyecto presentado por Rojas, Cowan, Kapralos y Dubrowski (2013), en el cual los autores desarrollan un sistema de aprendizaje por internet llamado OPEN, orientado a la enseñanza en medicina y que incluye elementos clásicos de educación virtual tales como foros de discusión, blogs, mensajes entre participantes, posibilidad de subir y descargar contenidos. Posteriormente, emplean un *framework* para aplicar gamificación llamado MRC.

Este *framework* planteado también por Rojas et al. (2014) contiene 4 fases que deben ser desarrolladas para lograr una exitosa implementación:

- Teoría y modelamiento: se explora la teoría relevante al sujeto (elemento a ser gamificado) revisando la literatura y el modelo o framework pedagógico que soportará las actividades a realizar. Adicionalmente, se define el tipo de intervención; para esto se debe entender el contexto y el objetivo. Se identifican los componentes de la intervención y los mecanismos que influencian los resultados.

- Prueba piloto: en esta fase se debe analizar con todos los interesados la pertinencia, probable aceptación, y relación costo-beneficio de la intervención; para esto se utilizan encuestas y grupos focales, y adicionalmente se diseña el protocolo de pruebas que será utilizado.

- Evaluación: en esta etapa se determina la efectividad de la intervención diseñada anteriormente, y se usa esto para encontrar su justificación en el mundo real.

- Implementación: en esta etapa final, la solución desarrollada en las etapas anteriores es implementada en el mundo real.

Finalmente, otro caso de gamificación aplicada en la enseñanza de la programación es desarrollado por Swacha y Baszuro (2013), quienes plantean un sistema completo de aprendizaje en el cual los estudiantes pueden ver los contenidos del curso, participar en retos, misiones y ejercicios; adicionalmente, los estudiantes van obteniendo medallas según su desempeño realizando ciertas acciones, como resolver ejercicios de primero, resolver determinada cantidad de ejercicios, terminar misiones antes del tiempo indicado, entre otros.

En el campo de los juegos serios tenemos a Tsalikidis y Pavlidis (2016), quienes desarrollaron jLegends, un videojuego de rol *online* llamado jLegends (JavaScript Legends) con el cual se puede facilitar la enseñanza de conceptos de programación y del lenguaje Javascript. Para esto emplean elementos de juegos de rol tales

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

como razas, clases, puntos de ataques, puntos de vida, entre otros, y los incluyen en una aventura en donde los estudiantes deben programar para ir avanzando en el juego. Los autores desarrollaron esta herramienta con modos de uso para docentes en clase y para estudiantes en casa.

Aprendizaje basado en problemas

El aprendizaje basado en problemas (Schmidt, 1983; Hmelo-Silver, 2004; Schmidt et al., 2011) es una metodología docente en la que los estudiantes abordan temas de una asignatura en el contexto de la resolución de problemas realistas, generalmente complejos y multifacéticos. Es una pedagogía centrada en los estudiantes, en el sentido de que son ellos quienes han de identificar qué saben, qué necesitan aprender, y cómo y dónde obtener la información que les permita resolver el problema planteado. Los estudiantes, de este modo, son expuestos a una situación simulada de trabajo profesional real que involucra aspectos de procedimiento, normativos y éticos que han de ser considerados para alcanzar el resultado esperado. Trabajando mediante la combinación de diferentes estrategias de aprendizaje para entender la naturaleza del problema, y las restricciones y opciones para su resolución, definir las variables de entrada, y analizar los puntos de vista planteados, los estudiantes aprenden a negociar y elegir las decisiones a llevar a cabo. En este contexto, el profesor actúa como facilitador de aprendizaje proporcionando las bases de conocimiento apropiadas para el problema, modelando el proceso de resolución y monitorizando el aprendizaje durante tal proceso. El profesor tiene que animar y apoyar a los estudiantes a aumentar su confianza para afrontar el entendimiento y resolución del problema.

A través del aprendizaje basado en problemas los estudiantes adquieren las siguientes competencias:

- Resolución de problemas de la vida real, lo cual implica objetivos, contenidos, contextos y dificultades

cambiantes que fomentan el desarrollo de habilidades, iniciativas y entusiasmo en entornos de trabajo.

- Resolución eficiente de problemas, desarrollando la habilidad de encontrar y usar recursos apropiados.
- Aprendizaje autónomo, desarrollando habilidades de aprendizaje autodirigido y automotivado, y capacidades de análisis proactivo.
- Automonitorización de aprendizaje, continuamente supervisando y validando la idoneidad del conocimiento y las tareas de resolución sobre los problemas.
- Trabajo en equipo, colaborando eficientemente como miembros de un equipo, y desarrollando habilidades de comunicación y liderazgo, además de capacidades sociales y éticas.
- Estas competencias difieren de aquellas adquiridas en una metodología docente clásica, y de este modo implican técnicas de evaluación alternativas, entre otras:

  – Evaluación personal, que ayuda a los estudiantes a pensar más cuidadosamente sobre lo que saben, lo que no saben, y lo que necesitan aprender para completar ciertas tareas planteadas.
  – Evaluación por compañeros, que ayuda a los estudiantes a experimentar situación del mundo real fuera del aula, en las cuales han de colaborar y evaluar el trabajo de otros.
  – Evaluación por el profesor, que ayuda a los estudiantes a entender cómo interactuar de forma efectiva en grupo y les anima a explorar diferentes ideas.
  – Presentaciones orales, que permiten a los estudiantes practicar sus habilidades comunicativas, que serán muy importantes en situaciones laborales futuras.
  – Informes, que permiten a los estudiantes practicar sus habilidades de redacción.

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

Aprendizaje cooperativo

El aprendizaje cooperativo (Johnson & Johnson, 1975; Johnson *et al*., 1988; Brown & Ciuffetelli, 2009) es una metodología docente que consiste en organizar en el aula actividades en las que los estudiantes deben trabajar en grupos para completar tareas de forma colectiva. A diferencia del aprendizaje individual o autónomo, en el cooperativo el proceso de aprendizaje de un estudiante se enriquece —o incluso es proporcionado— por los recursos y habilidades de sus compañeros de grupo, y de la comunicación que mantenga con ellos, e. g. pidiendo y compartiendo información, evaluando ideas y monitoreando el trabajo de cada miembro de grupo (Chiu, 2000; Chiu, 2008). De este modo, un estudiante tiene éxito en el aprendizaje si el resto del grupo también lo tiene. Además, en este escenario el papel del profesor pasa de proporcionar información a facilitar la adquisición y asimilación de información por parte de los propios estudiantes (Cohen, 1994; Chiu, 2004).

Para que una actividad de aprendizaje cooperativo sea efectiva debe tener presentes dos características principales: a) los estudiantes han de trabajar para alcanzar un objetivo o reconocimiento de grupo, y b) tal objetivo o reconocimiento de grupo depende del aprendizaje de cada estudiante (Brown y Ciuffetelli, 2009). De este modo, al diseñar tareas de una actividad de aprendizaje cooperativo, las funciones y responsabilidades de cada miembro de un grupo han de estar bien definidas y delimitadas. Los estudiantes deben saber perfectamente de qué tienen que estar a cargo y responder ante el grupo. Es más, aquello de lo que un estudiante es responsable no puede completarse por cualquier otro de sus compañeros. Así, todos los miembros del grupo, velando por el éxito de este, deben participar en la actividad dando lo mejor de ellos mismos.

Brown y Ciuffetelli (2009) plantean cinco elementos básicos esenciales que una actividad de aprendizaje cooperativo (formal) ha de tener:

- Interdependencia positiva.
- Interacción cara a cara.
- Exigibilidad individual/responsabilidad personal.
- Habilidades cooperativas.
- Autoanálisis de grupo.

El aprendizaje cooperativo requiere por tanto que los estudiantes se involucren en actividades de grupo que no sólo hacen incrementar el aprendizaje, sino que también producen otros beneficios, como la mejora del desarrollo de relaciones y habilidades sociales. Investigaciones recientes demuestran resultados abrumadoramente positivos del aprendizaje cooperativo. Así, por ejemplo, el estudio realizado por Tsay y Brandy (2010) reporta que estudiantes que participaron en actividades cooperativas exhibiendo comportamientos colaborativos y proporcionando retroalimentación constructiva tuvieron mejores calificaciones académicas en exámenes finales. Slavin (2010) refuerza los resultados de Tsay y Brandy demostrando que el aprendizaje cooperativo aumenta la autoestima de los estudiantes, enriquece la percepción de ellos sobre sus compañeros, y rompe barreras étnicas e ideológicas, fomentando interacciones positivas y relaciones de amistad.

Aplicación de la reforma

En los cursos especializados en enseñanza de programación ofertados antes del año 2015, los lineamientos pedagógicos definidos respondían a la visión de enseñanza en ingeniería que predominaba en los principales centros educativos superiores del país a finales del siglo pasado. Esto debido a que la planta docente estaba constituida principalmente por docentes con varios años de experiencia que habían dejado atrás su proceso formativo hace bastante tiempo; además, no se había discutido y unificado una

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

metodología propia y, finalmente, por la falta de herramientas tecnológicas efectivas para acompañar el aprendizaje de los alumnos.

Bajo esta metodología tradicional, los objetivos de aprendizaje y las competencias se mantenían relativamente proporcionales a los actuales, siendo sesgadas hacia el aprendizaje teórico del pensamiento algorítmico por el contexto de las condiciones enumeradas anteriormente. Sin embargo, las actividades de aprendizaje necesarias para cumplir estos objetivos eran esencialmente diferentes, predominando la clase magistral como primer mecanismo de transferencia de conocimiento, siendo reforzada por talleres y evaluaciones individuales (la mayoría de las veces sin la posibilidad de interacción con equipos de cómputo reales).

Sin desconocer las ventajas de la enseñanza tradicional, después de una gran crisis desencadenada por la falta de bachilleres interesados en ingresar al programa de Ingeniería de Sistemas, junto con tasas elevadas de mortalidad académica de los estudiantes actuales y la constante queja por su baja motivación, se define la actualización en varios niveles del programa curricular. Específicamente para el interés de este artículo se hablará de las decisiones tomadas en dos niveles (macro y micro) que se consideran fundamentales para el desarrollo de una gamificación efectiva.

Para que un proceso de gamificación tenga efecto, debe evaluar y posteriormente ajustar sus parámetros a la medida del contexto donde va a ser implementado. A pesar de que la definición aceptada en general de gamificación se limita a "la introducción de mecánicas de juego en entornos no lúdicos" (Oliva, 2016), si estas mecánicas no son coherentes con la realidad inmediata y si no están dadas en un entorno que soporte los procesos de calidad —en este caso la excelencia en la formación profesional para la de ingeniería de sistemas— es de esperarse que no sean efectivas, aun cuando puedan ser buenas en sí mismas. Por ejemplo, si se implementa un sistema de medallas que reconozca ciertas habilidades específicas de un estudiante durante un curso, pero este estímulo no tiene ningún valor fuera del curso, se perderá,

y junto con él, la motivación del estudiante a coleccionar estas medallas en ocasiones posteriores a la primera vez. En otro ejemplo más específico, por más que se generen estrategias lúdicas para la enseñanza del pensamiento algorítmico, si no existen los equipos y no se planean dentro de la metodología formas de traducir ese pensamiento en la creación de software concreto con el cual el estudiante pueda experimentar y recrear situaciones de la vida laboral cotidianas, no se lograrán cumplir los objetivos completamente. Por lo tanto, además de generar mecánicas de juego adecuadas, en paralelo se tomaron decisiones en un nivel superior para que todo el proceso fuera integral.

Para esto se planteó una estrategia con tres puntos principales: en primera lugar, al no tener datos del progreso de los estudiantes a lo largo de su carrera, resultaba difícil plantear soluciones específicas ante las dificultades que pudieran tener a lo largo del programa; por lo tanto, para los diferentes cursos que involucraran habilidades de programación se planteó la propuesta de realizar pruebas de diagnóstico al inicio y final de cada curso, para medir el estado en el que llegaban los estudiantes sobre los conocimientos básicos de programación (flujo de datos, estructuras de control, arreglos y matrices, funciones, clases y objetos). Los cursos seleccionados para estas pruebas fueron: Fundamentos de Programación, Programación de Computadores, Estructuras de Datos, Análisis de Algoritmos.

Además de esto, no solo los datos sobre los estudiantes resultaban insuficientes, sino que además sobre los profesores solo se contaba con la información cualitativa y subjetiva de la evaluación docente por parte de los estudiantes; por lo tanto, no se podían plantear estrategias concretas de mejora y acompañamiento para que su labor fuera más efectiva y satisfactoria. Este problema se veía aumentado entre los profesores de cátedra, quienes compartían mucho menos tiempo y espacios comunes con el resto de los docentes de planta. Las medidas que se han venido tomando al respecto son: por un lado, el seguimiento de los resultados académicos (tanto en las pruebas de diagnóstico como en las notas de cada curso) de los estudiantes que han pasado por los diferen-

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

tes profesores, buscando patrones y tendencias específicas que ayuden a revelar las falencias actuales; y por otro lado, la creación de espacios donde los docentes comparten sus experiencias y de manera conjunta (incluyendo a los docentes de cátedra) definen objetivos, contenidos y herramientas para el desarrollo de cada semestre.

En segundo lugar, antes de pensar en las mecánicas de juego como tal, fue necesario replantear los objetivos de cada curso teniendo en cuenta que se quería perfilar el programa de Ingeniería de Sistemas hacia el desarrollo de aplicaciones móviles, internet de las cosas, desarrollo web y videojuegos. Por esta razón era fundamental que los contenidos, las herramientas tecnológicas y las actividades de aprendizaje estuvieran alineadas con este nuevo modelo que se estaba gestando. Para el curso de Fundamentos de Programación se decidió agregar un módulo de aprendizaje de desarrollo de aplicaciones móviles con App Inventor 2. El curso de Programación de Computadores se enfocó en las líneas de móviles, web y escritorio, entre otros.

Finalmente, se discutieron y unificaron las herramientas, plataformas y lenguajes que estuvieran en tendencia actual y se decidió incluirlos en el plan de enseñanza de manera integrada con los contenidos de los cursos. Por esta razón, a lo largo de los diferentes cursos en general los estudiantes tienen contacto con PseInt (pseudolenguaje), App Inventor 2 (código en bloques), Java, Processing, C#, y Python.

De la mano con las actualizaciones curriculares que aún a fecha de hoy se vienen implementando, se empieza un periodo de experimentación y afirmación de diferentes mecánicas lúdicas apuntando a un desarrollo de los cursos que vincule a los estudiantes de manera emocional e intelectual para su mayor aprovechamiento. Las mediciones que se realizan hasta la fecha sobre estas mecánicas son: índices de asistencia, índices de permanencia en los cursos, notas promedio, rango máximo de notas por curso, rango mínimo de notas por curso.

- ***Sistema de puntos para la evaluación:*** buscando un giro en las actividades evaluativas tradicionales, en las cuales los estudiantes reciben calificaciones positivas y negativas por sus esfuerzos, se plantea la utilización de un modelo acumulativo, donde toda acción que realiza el estudiante de manera satisfactoria le suministra puntos (nunca le quita), de tal manera que la asociación esfuerzo-resultado es siempre positiva, siendo más fácilmente asimilada por los estudiantes, y se reduce el impacto negativo que una nota inferior al 3.0 suele tener en una evaluación. En general, se utilizó un total de puntos múltiplo de 5 (5000, 50 000, etc.) para poder utilizar un mapeo lineal directo a la nota definitiva acumulada.

- ***Bancos de Retos:*** de manera progresiva, se ha redefinido desde una perspectiva lúdica el concepto de "deber" o "tarea". Resaltando los valores de la superación personal, la autonomía y la autodidáctica incrementalmente se han cambiado los ejercicios y tareas tradicionales, secuenciales y con fecha límite inmediata, por bancos de ejercicios donde los estudiantes encuentran diferentes retos con varios niveles de dificultad que están abiertos desde que se comienza cualquier módulo del curso. La idea es que tengan la posibilidad de desarrollarlos de manera autónoma, a veces en tiempo de clase pero en general en sus horas de trabajo extraclase, y a su propio ritmo vayan encontrando caminos de aprendizaje que los lleven a asimilar los conocimientos de manera práctica.

- ***Dinámicas competitivas:*** la competición es una dinámica lúdica fundamental para el desarrollo de nuevas habilidades personales. Debe ser balanceada con dinámicas colaborativas para enriquecer aún más al estudiante y humanizarlo en sus labores profesionales. Varias de las actividades propuestas para obtener

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

puntos tienen su fundamento en el trabajo individual, soportadas por herramientas tecnológicas como:

– *Kahoot, que permite realizar quices contrarreloj de una manera dinámica y digital.* Otra mecánica que se ha popularizado entre los docentes es la de dinamizar la entrega de puntos según el tiempo y calidad del trabajo de cada estudiante. Por ejemplo, ante el reto de desarrollar X algoritmo, el primero en entregar recibirá 50 puntos, el segundo recibirá 50 solo si presenta alguna funcionalidad adicional, el tercero deberá hacer lo mismo respecto al segundo y así.

– *Dinámicas colaborativas:* educar para la colaboración en la ingeniería resulta extrañamente novedoso. El sistema educativo tradicionalmente recompensa los logros individuales y concibe las actividades grupales como complementarias; sin embargo, a la hora del ejercicio profesional, el desarrollo de software resulta altamente colaborativo. Trabajando en muchos casos en equipos no solo con colegas programadores sino con profesionales de todos los campos del conocimiento. Acerca de la colaboración, en algunos cursos se implementó un sistema de colaboración entre pares en el que los estudiantes con mejores notas y aptitudes sociales eran delegados con la responsabilidad de trabajar en horario extraclase resolviendo dudas y preparando material de apoyo para sus compañeros de curso a cambio de recibir bonificaciones académicas y exenciones de actividades de clase. Ellos además fomentaban la cooperación a través de las redes sociales y la resolución conjunta de problemas algorítmicos.

– *Exploración:* el aprendizaje y el descubrimiento están íntimamente ligados. De igual manera, el interés que un aprendiz puede tener sobre un nuevo tema depende de manera directa del reto y novedad que este represente para él. Uno de los grandes problemas de la educación tradicional es que estandariza las rutas de aprendizaje a criterio del docente, dejando vulnerables a quienes presentan un esquema de conocimientos diferentes a los que este planeta como estándar. La solución propuesta ante esto fue plantear diversas actividades que se pudieran desarrollar de una manera autónoma y no lineal junto con los contenidos del curso. Por ejemplo, los bancos de retos pueden estar abiertos desde el inicio del curso hasta la fecha de subida de notas al sistema. Otra posiblidad probada en clases es no cerrar de manera estricta todos los contenidos, sino permitir que de alguna manera los integrantes del curso tengan poder de decisión sobre lo que quieren aprender a continuación.

– *Storytelling*: a la hora de implementar narrativas en el aula de clase, se descubrió que existen dos formas independientes de hacerlo. Por un lado, una narrativa inmersiva e integral en la que todo elemento referente a la clase tiene un equivalente semántico en el espacio de la narrativa aplicada. Las notas, las tareas, los talleres, las medallas, incluso el estilo narrativo de los exámenes deben corresponder a elementos del universo que se está construyendo para encapsular el curso de tal manera que se mantenga en la medida de lo posible el estado de suspensión de la incredulidad. Por otro lado, una estrategia más débil implicó no hacer analogías de todos los elementos del curso, sino coherencias temáticas a la hora de plantear

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

actividades concretas. Por ejemplo, ¿cómo cada tema del curso nos puede ayudar en la creación de nuestro propio planeador de viajes?

– *Juegos de rol:* en los juegos de rol, los actores pasan a representar de alguna manera un papel arquetípico. Esta representación puede ser dada a través de avatares, actuación, o simplemente el ejercicio del rol en sí. Como se mencionó anteriormente, la selección de los mejores estudiantes para que jugaran en el rol de guardianes resultó en una descentralización de la estructura jerárquica del conocimiento, en la medida en que tomaban estos el papel de docentes asistentes a cambio de beneficios especiales. Otro ejemplo de implementación de esta mecánica es en la definición de trabajos en equipos para el desarrollo de software. La experiencia muestra que es muy difícil que todos los integrantes de un equipo sean implementadores de código. Por lo tanto, definir roles adecuados dependiendo del perfil de cada estudiante resulta vital para el máximo aprovechamiento de la experiencia. Por ejemplo, los cuatro roles del proceso creativo de Roger von Oech: explorador, artista, juez y guerrero.

– *Actividades físicas:* finalmente, el desarrollo de actividades que impliquen movimientos y el uso del cuerpo resulta interesante para generar procesos de asociación inusuales. En la práctica se realizaron carreras de observación a través de las cuales los estudiantes debían recorrer el campus de la universidad buscando pistas para solucionar diferentes problemas; buscaban fragmentos de código escondidos, enlaces a páginas de recursos previamente diseñadas, ayudas visuales, para al finalizar el recorrido resolver un gran problema de manera colectiva utilizando las herramientas

conseguidas a lo largo de la actividad. [Inicio de lista de 2.º nivel]

Resultados

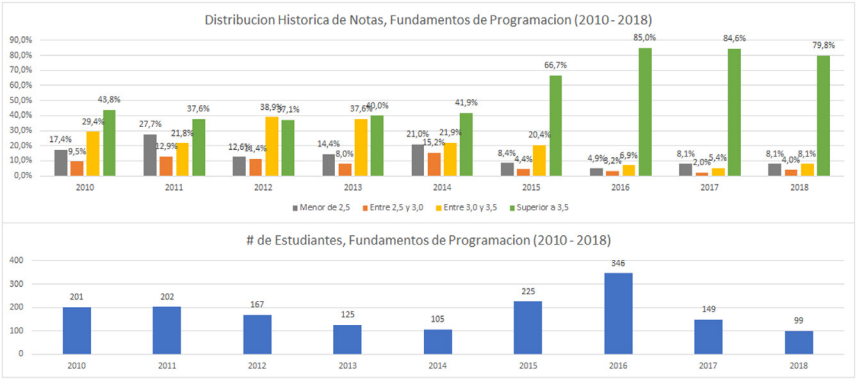
Después de la nueva etapa de implementación de políticas, las tasas de deserción han disminuido y la población estudiantil en general ha aumentado. Los puntajes de satisfacción han mejorado sus resultados. Las métricas y los estándares se han establecido para el primer año de cursos de programación y el proceso continúa progresivamente. El interés y el estado de ánimo general de los estudiantes han mejorado considerablemente. También se creó un espacio anual para la divulgación y el intercambio de los mejores proyectos de cursos de ingeniería de sistemas (Ingeniotic) y, finalmente, el programa logró un mayor reconocimiento a escala nacional.

La motivación es un factor vital a la hora de realizar procesos educativos; en el caso de los jóvenes que inician su vida profesional con la carrera de ingeniería de sistemas es muy importante lograr incrementar la motivación desde etapas tempranas a fin de evitar, entre otros factores, deserción y baja comprensión de los temas impartidos, que tiene como resultado bajos niveles de aprendizaje. A fin de mejorar la motivación en el programa de Ingeniería de Sistemas y las asignaturas de la línea de programación de computadores, desde 2015 se han llevado a cabo algunos esfuerzos individuales para incluir la gamificación como un componente pedagógico en el aula de clase a fin de lograr una mejor motivación de sus estudiantes. Si bien estos esfuerzos han sido buenos y se ha podido evidenciar una mejora tanto en las notas finales de los estudiantes (figuras 2 y 3), como en la motivación de los estudiantes por profundizar en temas de programación.

DOI: https://doi.org/10.15332/s1657-107X

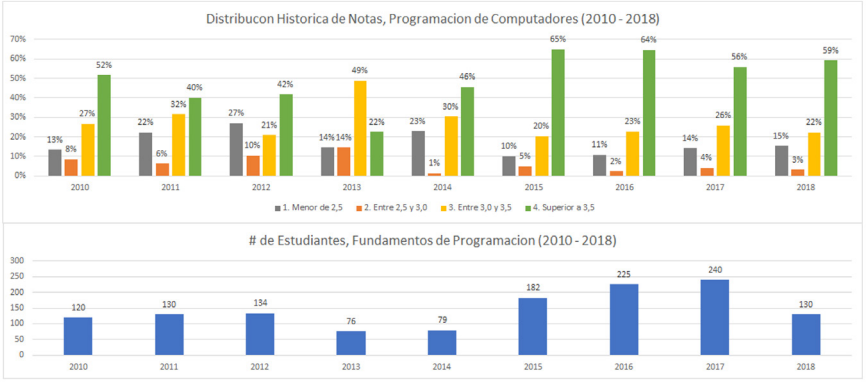
ISSN: 1657-107X
e-ISSN: 2500-5421

Figura 2. Estadísticas de notas y población estudiantil en el curso de Fundamentos de Programación.



Fuente: sistema Banner Académico (febrero de 2019).

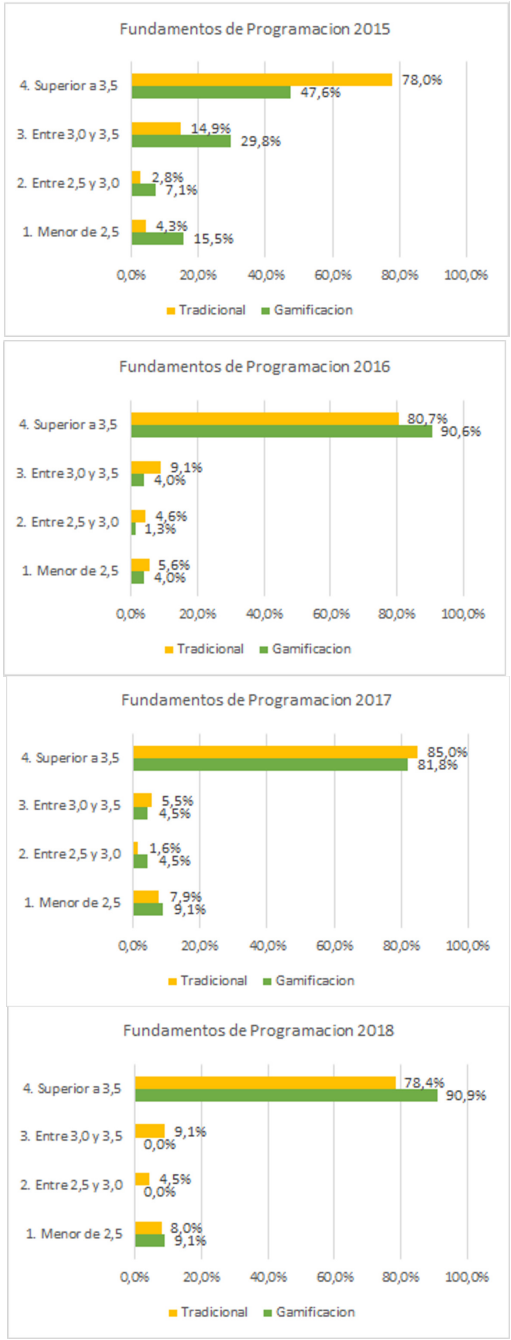
Figura 3. Estadísticas de notas y población estudiantil en el curso de Programación de Computadores.



Fuente: sistema Banner Académico (febrero de 2019).

Los siguientes gráficos (figura 4) representan la distribución de notas en los cursos de fundamentos de programación que implementaron metodología tradicional o metodología basada en gamificación en el aula. Para el 2015, un total de 84 alumnos (37,7 % del total) recibieron clases con metodología basada en gamificación; así mismo, en 2016, 149 (43,1 %), en 2017, 22 (14,8 %) y en 2018, 11 (11,1 %).

Figura 4. Distribución de notas finales en los cursos de Fundamentos de Programación.



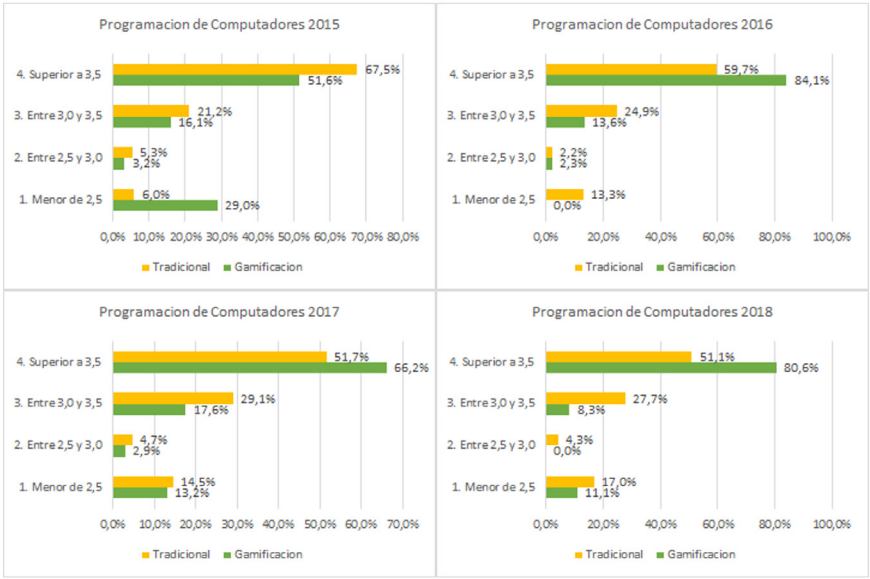
Fuente: sistema Banner Académico (febrero de 2019).

DOI: https://doi.org/10.15332/s1657-107X

ISSN: 1657-107X
e-ISSN: 2500-5421

Igualmente, para los cursos de Programación de Computadores podemos observar en los siguientes gráficos (figura 5) la distribución de las notas en los cursos que implementaron metodología tradicional o metodología basada en gamificación en el aula. Para el 2015 un total de 31 alumnos (17 % del total) recibieron clases con metodología basada en gamificación, así mismo en 2016 44 alumnos (19,6 %), en 2017 68 alumnos (28,3 %) y en 2018 36 alumnos (27,7 %).

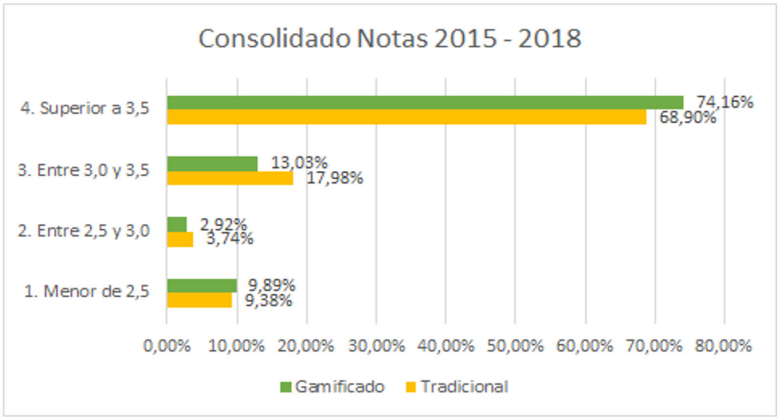
Figura 5. Distribución de notas finales en los curso de Programación de Computadores.



Fuente: sistema Banner Académico (febrero de 2019).

En términos generales se evidencia un aumento, en el grupo de las las notas más altas, de los estudiantes que recibieron cursos de Fundamentos de Programación y Programación de Computadores en los que se aplicaron estrategias gamificadas. Sin embargo, a causa del relativamente pequeño número de alumnos que han incorporado esta metodología, y de que los docentes que imparten estos cursos lo hacen de manera independiente con tan solo unos pocos acuerdos comunes sobre la estrategia a la hora de "gamificar", en general aún se desconoce el impacto detallado de la estrategia en el programa.

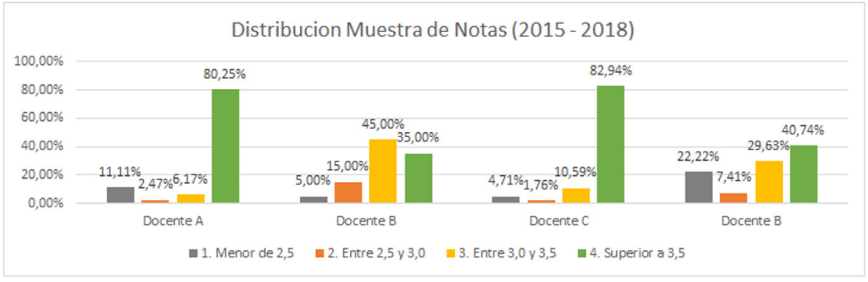
Figura 6. Consolidado de notas en cursos de programación.



Fuente: sistema Banner Académico (febrero de 2019).

En la figura 7 se ilustran los resultados de notas en cursos dictados por 4 docentes; los docentes A y B implementaron metodología tradicional en el aula y los docentes C y D implementaron metodologías basadas en gamificación.

Figura 7. Distribución de notas entre docentes.



Fuente: sistema Banner Académico (febrero de 2019).

Conclusiones

Se observa entonces que el uso de estrategias lúdicas, por lo menos en el caso del estudio, no garantiza una mejora significativa en el desempeño general de los estudiantes, y se presume que el re-

DOI: https://doi.org/10.15332/s1657-107X
ISSN: 1657-107X
e-ISSN: 2500-5421

sultado definitivo está ligado a otros factores individuales del docente, como lo pueden ser: su carisma, sus habilidades comunicativas, su habilidad para resolver conflictos e inquietudes, la forma en que aplica su metodología, entre otras variables que se deberán indagar en estudios posteriores.

La metodología de gamificación, como cualquier otra, es una herramienta y un recurso más para el aula de clase; depende del docente la obtención de los resultados, pues no se trata de una receta de pasos a seguir con un único resultado infalible, sino que por el contrario depende también de factores muy personales relacionados con el docente que están por fuera del alcance de este estudio. Por otro lado, hemos evidenciado que la estrategia de gamificación en términos generales ha sido bien recibida por los estudiantes; los resultados de este estudio permitirán mejorar la implementación de la metodología en docentes que no la conozcan, lo que ayudará a mejorar sustancialmente los resultados.

Las orientaciones metodológicas son suceptibles de sistematizarse en una herramienta de software que servirá como modelo para los docentes que deseen implementarla presentando las posibilidades que ofrece la metodología respecto de las herramientas y recursos que se pueden utilizar en gamificacion; esta herramienta se está diseñando de manera independiente al estilo de cada docente para que pueda ser usada de manera general en las horas de contacto estudiante-docente así como en las horas de estudio independiente del estudiante.

Referencias bibliográficas

Bouki, V., Economou, D. y Kathrani, P. (2014). “Gamification” and Legal Education: A Game Based Application for Teaching University Law Students. En 2014 *International Conference on Interactive Mobile Communication Technologies and Learning* (IMCL 2014) (pp. 213-216).

Deterding, S., Dixon, D., Khaled, R. y Nacke, L. (2011). From Game Design Elements to Gamefulness: Defining "Gamification". En *MindTrek '11: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (pp. 9-15).

Lambruschini, B. B. y Pizarro, W. G. (2015). Tech-Gamification in University Engineering Education: Captivating Students, Generating Knowledge. En *2015 10th International Conference on Computer Science & Education (ICCSE)* (pp. 295-299).

Oliva, H. A. (2016). La gamificación como estrategia metodológica en el contexto educativo universitario. *Realidad y Reflexión, 16*(44), 29-47.

La República. (2010). *Ingenierías: mucha demanda y poca motivación para estudiarla.* Recuperado de http://www.mineducacion.gov.co/observatorio/1722/article-243384.html

Rojas, D., Cowan, B., Kapralos, B. y Dubrowski, A. (2014). Gamification and Health Professions Education. En 2014 *IEEE Games Media Entertainment (GEM)* (pp. 1-2).

Rojas, D., Kapralos, B. y Dubrowski, A. (2013). The Missing Piece in the Gamification Puzzle. En *Gamification '13: Proceedings of the First International Conference on Gameful Design, Research, and Applications* (pp. 135-138).

Rojas, D., Kapralos, B. y Dubrowsk, A. (2014). Gamification for Internet Based Learning in Health Professions Education. En *2014 IEEE 14th International Conference on Advanced Learning Technologies (ICALT)*, (pp. 281-282).

Swacha, J. y Baszuro, P. (2013, July). Gamification-based e-learning Platform for Computer Programming Education. En *X IFIP World Conference on Computers in Education* (pp. 122-130). Toruń: Universidad Nicolás Copérnico de Toruń.

DOI: https://doi.org/10.15332/s1657-107X

ISSN: 1657-107X
e-ISSN: 2500-5421

Tsalikidis, K. y Pavlidis, G. (2016). jLegends: Online Game to Train Programming Skills. En *2016 7th International Conference on Information, Intelligence, Systems & Applications (IISA)* (pp. 1-6).