

ARTÍCULO CIENTÍFICO
CIENCIAS DE LA EDUCACIÓN

Sistematización de experiencias en la integración de software libre educativo al proceso de enseñanza-aprendizaje

Systematization of experiences in the integration of educational free software in the teaching-learning process

Coca Bergolla, Yuniesky ^I; Pérez Pino, María Teresa ^{II}

I. ycoca@uci.cu. Dirección de Formación de Pregrado, Universidad de las Ciencias Informáticas, La Habana, Cuba

II. mariatpp@uci.cu. Centro de Innovación y Calidad en la Educación, Universidad de las Ciencias Informáticas, La Habana, Cuba

Recibido: 15/06/2020

Aprobado: 12/10/2020

Como citar en normas APA el artículo:

Coca Bergolla, Y., & Pérez Pino, M. T. (2020). Sistematización de experiencias en la integración de software libre educativo al proceso de enseñanza-aprendizaje. *Uniandes Episteme*, 8(1), 77-90.

RESUMEN

La integración de software educativo al proceso de enseñanza-aprendizaje en las carreras relacionadas con la informática ha sido ampliamente abordada. En asignaturas como la Inteligencia Artificial la interacción del estudiante con el código fuente del software juega un papel esencial. Al software educativo que aprovecha las libertades del software libre en el proceso de enseñanza-aprendizaje se le denomina software libre educativo. Su integración a este proceso se compone del diseño didáctico del software y su incorporación armoniosa al proceso de enseñanza-aprendizaje para el cual fue diseñado. Este trabajo tiene como objetivo presentar la sistematización de experiencias en el desarrollo y utilización de un software libre educativo a la asignatura Inteligencia Artificial II de la carrera Ingeniería en Ciencias Informáticas de la Universidad de las Ciencias Informáticas, en La Habana, Cuba. La sistematización se organizó en cuatro pasos, planificación, recuperación del proceso vivido, análisis crítico y cierre. Como resultado de la sistematización se obtiene un conjunto de lecciones aprendidas que enrumban el trabajo para futuras investigaciones.

PALABRAS CLAVE: Proceso de enseñanza-aprendizaje; sistematización de experiencias; software libre educativo

ABSTRACT

The integration of educational software into the teaching-learning process in computer-related careers has been widely addressed. In subjects such as Artificial Intelligence the interaction of the student with the software source code plays an essential role. Educational free software is the educational software that takes advantage of the freedoms of free software in the teaching-learning process. The objective of the present work is to present the results of the application of the experience systematization method applied in the development and incorporation of a free educational software to the subject Artificial Intelligence II at the University of Computer Sciences, in Havana, Cuba. We organize the systematization in four steps: planning, recovery of the lived process, critical analysis and closing. The systematization made it possible to obtain a set of lessons learned that define the work for future research.

KEYWORDS: Teaching-learning process; systematization of experiences; educational free software

INTRODUCCIÓN

El software educativo (SE) (Rodríguez, 2010) forma parte del conjunto de tecnologías de la información y la comunicación (TIC) que han irrumpido en los procesos de enseñanza-aprendizaje (PEA) con extraordinaria fuerza. Diseñar un software para ser utilizado como medio de enseñanza-aprendizaje y aprovechar sus potencialidades peculiares, constituye uno de los retos actuales de las ciencias pedagógicas (García & Rico, 2014).

En carreras de ingeniería relacionadas con la informática, se incluyen asignaturas que tratan la programación de computadoras. En ellas es tendencia en el mundo la utilización de software donde el estudiante modifica el código como parte de su proceso de aprendizaje (Sosnowski, Ernsberger, Cao, & Ray, 2013, Riedl, 2015). No todos cumplen con las especificaciones de la Fundación del Software Libre (FSF por sus siglas en inglés) para ser catalogados como software libre, pero todos tienen la característica de permitir el acceso a su código fuente de forma total o parcial. Esta característica reviste especial importancia, ya que le brinda al estudiante la posibilidad de la co-creación participativa del software (Romero & Patiño, 2018) y constituir una opción para el desarrollo de proyectos integradores (Velázquez, Velasteguí, & Arévalo, 2016).

La posibilidad de estudiar, modificar y reutilizar el código fuente del software puede ser de gran utilidad en asignaturas relacionadas con la programación de computadoras. El término software libre educativo se reconoce como toda aplicación informática diseñada con la intencionalidad de aprovechar las libertades del movimiento de software libre en un proceso de enseñanza-aprendizaje (Coca & Pérez, 2020a). Su integración al proceso de enseñanza-aprendizaje en carreras de informática se reconoce como una variable compleja con dos áreas

bien delimitadas y estudiadas: el diseño didáctico del software libre educativo y su incorporación armoniosa al PEA. Estas se constituyen en dimensiones de la variable y están compuestas a su vez por subdimensiones para su estudio y evaluación (Coca & Pérez, 2020a). Las subdimensiones para la dimensión diseño didáctico del software libre educativo son: Didáctica, tecnológica, espacial, de gestión y práctica. Mientras para la dimensión incorporación armoniosa del software libre educativo al proceso de enseñanza-aprendizaje: Contextualizada, sistémica, continua, reflexiva y planificada.

En Coca y Pérez (2020b) se presenta un modelo para llevar a cabo dicho proceso de integración en asignaturas de Inteligencia Artificial. El modelo incluye un componente instrumental, con tres etapas generales, fases y acciones concretas. La segunda etapa (Ejecución) incluye dos fases que se corresponden con las dimensiones de la variable. Estas fases incluyen un conjunto de acciones específicas.

Para la fase de diseño didáctico del software libre educativo las acciones son:

- Descripción de las relaciones entre el software libre educativo y cada uno de los componentes de la didáctica.
- Elaboración de las tareas a realizar en los diversos espacios disponibles.
- Especificación de las características del software para garantizar su distribución y utilización en el proceso de enseñanza-aprendizaje.
- Elaboración de tareas que aprovechen las libertades del software libre.
- Análisis, diseño y desarrollo computacional del software libre educativo.

Para la fase de Incorporación del software libre educativo al proceso de enseñanza-aprendizaje de la asignatura de IA las acciones son:

- Diagnóstico inicial a los estudiantes.
- Presentación del software libre educativo.
- Familiarización de los estudiantes con el software.
- Asignación de tareas personalizadas a realizar con el software.
- Seguimiento a las tareas personalizadas a realizar con el software.
- Evaluación final al estudiante.

En investigaciones relacionadas con el tema se muestran resultados en el desarrollo (Coca, Rosell, & Velazquez, 2017) y la incorporación (Coca & Pérez, 2018) de estos medios en asignaturas de Inteligencia Artificial de la carrera Ingeniería en Ciencias Informáticas. Sin embargo, en los trabajos publicados se aprecian carencias en la exposición de las acciones llevadas a cabo y de los resultados específicos alcanzados en cada una de las subdimensiones y dimensiones de la variable de estudio. Estas carencias conducen a plantear el problema: ¿Cómo exponer las experiencias adquiridas en la integración de software libre

educativo al proceso de enseñanza-aprendizaje de la Inteligencia Artificial en la carrera Ingeniería en Ciencias Informáticas?

La sistematización de experiencias, en educación, examina la reconstrucción de experiencias, la interpretación crítica como proceso de reflexión que ordena y relaciona datos, hechos, modos de actuar para construir aprendizajes que conllevan a mejores tomas de decisiones en el ámbito educativo (Pavón, 2017). Este método refiere a aquellas acciones o experiencias que se derivan de intervenciones intencionadas y con objetivos claros, en la que los actores involucrados participan del resultado esperado. En el contexto universitario se reconocen varias fuentes de experiencias, dentro de las que resaltan las estrategias de enseñanza y la producción de material didáctico (Pérez de Maza, 2016).

El presente trabajo tiene como objetivo presentar la sistematización de experiencias en la integración de software libre educativo a la asignatura Inteligencia Artificial II (IA-II) en la carrera Ingeniería en Ciencias Informáticas (ICI) de la Universidad de las Ciencias Informáticas (UCI) de La Habana, Cuba.

MÉTODOS

Varios autores han propuesto pasos o momentos para llevar a cabo la sistematización de experiencias (Berrutti, Cabo, & Dabezies, sf.; Pérez de Maza, 2016; Jara, 2018). A partir de los puntos coincidentes de estos autores y las características de la experiencia a sistematizar, se organizó el proceso de la siguiente manera:

1. **Planificación de la sistematización:** Incluyó la especificación de los participantes, el objetivo, el objeto, los ejes de sistematización y los pasos a seguir.
2. **Recuperación del proceso vivido:** Se realizó una reconstrucción histórica y se ordenaron fuentes de información.
3. **Reflexión crítica:** Se realizó una interpretación crítica y se compararon los resultados con resultados anteriores.
4. **Cierre:** Se especificaron conclusiones a partir de las experiencias y se elaboraron productos de comunicación de resultados.

A continuación se muestran los elementos generales llevados a cabo en cada uno de los pasos definidos.

1. Planificación de la sistematización

La experiencia seleccionada para llevar a cabo la sistematización incluyó como participantes un recién graduado en adiestramiento (RGA), dos estudiantes de 5to año de la carrera Ingeniería en Ciencias Informáticas y el investigador con más de 10 años de experiencia en la asignatura, con categoría docente de Profesor Auxiliar y Jefe de la Disciplina en la carrera. La sistematización fue llevada a cabo por el investigador con la colaboración de los demás participantes en la experiencia. El RGA participó como ayudante en la fase de diseño didáctico

del software libre educativo y en las clases de la asignatura. Los estudiantes llevaron a cabo el desarrollo computacional del software libre educativo, tutorados por el investigador, quien fue el profesor de la asignatura IA-II.

El objetivo de la sistematización fue analizar una práctica concreta de integración de SLE al PEA de una de las asignaturas de IA de la carrera Ingeniería en Ciencias Informáticas. El eje de la sistematización fue la variable de investigación: Integración de software libre educativo al proceso de enseñanza-aprendizaje de la Inteligencia Artificial. Esta variable es detallada en Coca & Pérez, (2020a), donde se especifican las características de cada subdimensión.

Para cada subdimensión de la dimensión diseño didáctico del software libre educativo se especifica lo siguiente:

- **Didáctica:** Determina las relaciones del software con los componentes del PEA: objetivos, contenidos, métodos, medios, evaluación y formas de organización.
- **Tecnológica:** Analiza las características del software, específicamente en las relacionadas con el movimiento de software libre que deben ser tomadas en cuenta en su desarrollo. Elementos de interés son la accesibilidad al código, la organización arquitectónica del software que permita su adecuado estudio y modificación y la disponibilidad de documentación de ayuda para utilizar el software en el PEA.
- **Espacial:** Se determinan las tareas para los distintos espacios en los cuales se utiliza el software dentro del PEA: individual, por equipo y grupal.
- **Gestión:** Incluye características de la licencia del software y las vías que permiten su correcta distribución. Así como las características tecnológicas que garanticen su utilización en el PEA, tomando en cuenta las condiciones existentes.
- **Práctica:** Incluye, como estrategia específica, el diseño de tareas donde los estudiantes tengan que estudiar, modificar y reutilizar el código fuente del software.
- Para las subdimensiones de la dimensión incorporación armoniosa del software libre educativo al proceso de enseñanza-aprendizaje:
- **Contextualizada:** Se refiere a la relación entre las tareas definidas para el estudiante y el contexto. Se incluye el trabajo de los estudiantes en grupos de investigación o centros donde lleven a cabo su práctica laboral, así como sus propios intereses enriquecidos con sus conocimientos culturales y profesionales.
- **Sistémica:** Esta subdimensión se evalúa integralmente, las demás dimensiones con sus indicadores tributan a evaluar el enfoque sistémico de todo el proceso. Específicamente se analiza el componente software libre educativo y su relación con la asignatura, así como su impacto en el PEA.
- **Continua:** Asume la incorporación armoniosa del SLE como un proceso que no termina con la evaluación de los resultados, sino que los mismos sirven de base para

próximos cursos. Se analiza si se toman en cuenta las soluciones de los estudiantes para mejorar el software. De igual forma es de interés la continuidad dentro de la propia asignatura. Se analiza si existe relación desde las primeras tareas del estudiante con el software hasta las más complejas al final del curso.

- **Reflexiva:** Se refiere a la percepción que tiene el profesor y los estudiantes acerca de la importancia de la utilización del software libre educativo. En el caso de los profesores se valora tanto su percepción sobre la importancia del SLE para su autosuperación, como para los estudiantes como futuros profesionales.
- **Planificada:** Vinculada a la organización y la creación de las condiciones para llevar a cabo la incorporación armoniosa del SLE al PEA. Incluye el análisis del tiempo utilizado por el profesor para el diseño didáctico del software, así como para su autosuperación. Se considera importante la cantidad de clases en que se planifican tareas con el software, así como el tiempo planificado y asignado para el trabajo independiente del estudiante con el SLE. Por último, las vías previstas y utilizadas para la retroalimentación de todo el proceso de integración del SLE al PEA de la IA.

La experiencia sistematizada se corresponde con el diseño didáctico del software: Sistema Multiagente para la Extracción de Conocimiento en Bases de Datos (SMEKDB) y su incorporación al PEA de la asignatura IA-II en el curso 2017-2018. Este software libre educativo fue diseñado para configurar los algoritmos de minería de datos estudiados en clase y aplicarlos a bases de datos. Utiliza la biblioteca de algoritmos de minería de datos Weka desarrollada con el lenguaje de programación Java.

Para la recuperación del proceso vivido se analizaron los artefactos generados por la metodología de desarrollo de software Ingenia (Botía, González, Gómez, & Pavón, 2008). Esta es una metodología de desarrollo específica para sistemas multiagentes. Se consideró necesario especificar el diseño didáctico del software en un guion de contenido (Ramos, 2016), este fue de gran utilidad en el proceso de sistematización de la experiencia. Adicionalmente se analizaron los documentos presentados en los talleres de trabajo de diploma de los estudiantes, así como los planes de clases del profesor.

2. Recuperación del proceso vivido:

Para llevar a cabo la fase de diseño didáctico del software libre educativo, se elaboró un perfil de trabajo de diploma donde se especificaron los objetivos generales del software a desarrollar. Se seleccionaron dos estudiantes de 5to año de la Facultad 4, con buenos resultados en las asignaturas de Inteligencia Artificial y con interés de trabajar en la investigación.

En los primeros encuentros se explicó el trabajo, se aclararon las dudas y se elaboró un plan de formación con el cronograma general de las tareas a realizar. El investigador, con la ayuda

de los estudiantes, elaboró el guion de contenido necesario en el proceso de diseño del software y de incorporación del mismo al PEA.

Para llevar a cabo el diseño didáctico y elaborar el guion de contenido, se analizó el programa analítico y el plan calendario de la asignatura IA-II. Se determinaron los objetivos y contenidos a trabajar con el SLE, se especificaron métodos para incorporar el software en cada forma organizativa del PEA. Se elaboraron indicaciones para la utilización de los medios disponibles en el aula y en los laboratorios, la forma de trabajar en la plataforma de enseñanza-aprendizaje y cómo aprovechar los medios de los propios estudiantes. Por último se declararon las vías para la evaluación sistemática, parcial y final a los estudiantes, tomando en cuenta el trabajo con el software.

Se elaboraron tareas generales para trabajar con el software en el grupo, por equipos y de forma individual por los estudiantes. De igual forma, se elaboraron tareas generales para estudiar el código fuente, para modificarlo y reutilizarlo. En la Tabla 1 se muestran ejemplos de tareas definidas para el trabajo con el software.

Tabla 1. Ejemplos de tareas definidas para el trabajo con el software libre educativo.

Tarea a diseñar	Ejemplo de tarea
Tareas para el trabajo individual con el software	Comparar los resultados al aplicar varios algoritmos a una base de datos
Tareas para el trabajo en equipo con el software	Modificar los parámetros de los distintos algoritmos de clasificación
Tareas para el trabajo grupal con el software	Analizar la estructura arquitectónica del SLE
Tareas para el estudio del código fuente	Estudiar cómo el software libre educativo realiza la llamada a los métodos implementados en la biblioteca Weka.
Tareas para la modificación del código fuente	Modificar los parámetros de los algoritmos y visualizar los resultados a través de la interfaz para interpretarlos.
Tareas para la reutilización del código fuente	Desarrollar una aplicación nueva que reutilice métodos o agentes del SLE para resolver una tarea específica.

Para garantizar la utilización y distribución del software se especificaron los requerimientos tecnológicos del mismo para ser utilizado según las condiciones de aulas y laboratorios. Se llevó a cabo la adecuación del curso en la plataforma de enseñanza-aprendizaje Zera (Manso, Cañizares, & Febles, 2016) para incluir el software, los documentos de ayuda y otras orientaciones necesarias.

Posteriormente, se definieron los elementos computacionales para el desarrollo computacional del software, de manera que se tomaron en cuenta las características ya diseñadas. Para llevar a cabo el desarrollo del sistema, se siguió la metodología Ingenias (Botía et al., 2008), en la cual se definen metamodelos que guían y ayudan en la estructuración arquitectónica del sistema multiagente. Durante la fase de análisis, los metamodelos desarrollados fueron el de agentes, el de organización, el de objetivos y tareas y el de entorno,

posteriormente en la fase de diseño se refinaron dichos metamodelos y se agregó la especificación de las tareas mediante el metamodelo de interacción.

Adicionalmente, se especificaron los requisitos no funcionales, funcionales y demás artefactos según la metodología. Los elementos de mayor interés en el desarrollo fueron la utilización de la biblioteca libre JADE para el desarrollo de los agentes y la utilización de Weka como biblioteca de algoritmos de minería de datos. La inclusión de estas bibliotecas exigió especificar la forma de vincularlas al entorno de desarrollo en lenguaje Java NetBeans, utilizado para crear el proyecto computacional. La utilización de bases de datos en el gestor libre PostgreSQL requirió hacer una transformación de las tablas del formato SQL, al formato arff utilizado por Weka.

Luego del diseño didáctico del software se llevó a cabo la fase de incorporación del software al proceso de enseñanza-aprendizaje. Primeramente se planificaron las tareas a desarrollar en el PEA. Se elaboraron los planes de clases siguiendo las indicaciones del diseño didáctico del software. El grupo asignado al profesor fue el 4505, integrado por 18 estudiantes, tres hembras y 15 varones.

En la primera conferencia de la asignatura se aplicó un diagnóstico a los estudiantes. La elaboración del mismo se realizó para complementar la información que dispone el profesor en la entrega pedagógica al inicio del curso y los resultados del diagnóstico integral realizado a los estudiantes al ingresar a la carrera.

Ya los estudiantes han recibido la asignatura Inteligencia Artificial I, por lo que tienen conocimientos generales de la temática. No obstante, se realizó un recordatorio general de los núcleos de conocimientos y problemas generales que resuelve. Con este punto de partida, se orientó la búsqueda de un problema profesional a ser resuelto con tareas de minería de datos. Este problema podía estar relacionado con el centro de desarrollo donde realizan su práctica profesional u otra inquietud o interés personal. Junto al problema debían entregar una base de datos en PostgreSQL a la cual se le aplicarían los algoritmos de minería de datos.

En la primera frecuencia de la semana seis, se recogieron los problemas redactados por los estudiantes. Dos estudiantes no entregaron problemas, alegando que no encontraron, ni supieron cómo redactar un problema de sus propios intereses. A todos los demás problemas entregados, se les hicieron adecuaciones para ajustarlos a los objetivos y contenidos de la asignatura. Para los dos estudiantes que no entregaron problemas se elaboraron tareas, a partir de la información recogida en el diagnóstico. En todos los casos se tuvo en cuenta las indicaciones del guion de contenido para la asignación de tareas de reutilizar el código fuente. La primera conferencia del tema Aprendizaje Automático se impartió en la semana ocho. En ella se presentó el SLE, su interfaz y las funcionalidades generales. Como trabajo independiente se orientaron las tareas personalizadas a cada estudiante.

En la siguiente clase práctica se realizaron tareas por equipo para el estudio del código fuente, según las indicaciones del guion de contenido. Estas tareas fueron presentadas y debatidas en el grupo. En la clase de laboratorio posterior, se asignaron las primeras tareas de modificación del código fuente, igualmente por equipos y tomando en cuenta las indicaciones del guion de contenido.

En las semanas 11 y 12, en las clases de laboratorio correspondientes, se realizaron tareas relacionadas con el contenido específico y se aclararon dudas a los estudiantes sobre el trabajo independiente asignado al inicio para reutilizar el código fuente. Siempre que fue posible se realizaron las preguntas y se aclararon las dudas de forma grupal, donde los propios compañeros de aula podían exponer sus experiencias y las formas de dar solución a los problemas enfrentados.

En la clase de laboratorio de la semana 14 se efectuó la presentación de los resultados por los estudiantes. Como elementos más significativos, tres estudiantes reutilizaron los agentes en JADE del SMA, cuando el resto solo reutilizó los métodos de acceso a la biblioteca Weka. Todos reutilizaron el agente de transformación de PostgreSQL a formato arff.

Para la nota final por el trabajo desarrollado se consideró que cuatro estudiantes vencieron los objetivos, por lo que fueron eximidos de la prueba final. A otros tres estudiantes se les subió la nota alcanzada en la prueba final, tomando como criterio fundamental su trabajo con el SLE.

Todas las soluciones de los estudiantes se valoraron para posibles modificaciones al software. Fueron incorporadas finalmente dos modificaciones al software, un nuevo algoritmo que incorporó la versión actualizada del Weka y un cambio en la interfaz que permitió un acceso más cómodo a la funcionalidad de comparación de algoritmos.

3. Reflexión crítica:

Para el análisis crítico del proceso, se llevó a cabo una valoración cualitativa de cada subdimensión. Adicionalmente, se expusieron lecciones aprendidas y aspectos de interés a tomar en cuenta en próximas investigaciones. Estos elementos son mostrados en el apartado de resultados del presente artículo.

4. Cierre:

Como parte del cierre del proceso de sistematización de experiencias, se resumieron los principales resultados. Se plantearon los aspectos de mayor interés y se procedió a la divulgación de los resultados.

RESULTADOS

Los resultados obtenidos en la etapa de valoración crítica se resumen por subdimensiones y dimensiones.

Dimensión diseño didáctico: El principal resultado en esta dimensión fue la obtención del guion de contenido. En este se recogen las especificaciones del software según las subdimensiones definidas. A continuación se muestran los principales resultados de cada subdimensión.

Didáctica: Se especifican indicaciones precisas para abordar varios objetivos y contenidos de la asignatura. Se especifican diversos métodos a utilizar según la tipología de clases y se relacionan un conjunto de medios a utilizar de manera que soporten el trabajo con el SLE.

Se realizan indicaciones para evaluar a los estudiantes de forma sistemática, parcial y para definir su evaluación final a partir de su trabajo con el software. Precisamente la evaluación dejó algunas insatisfacciones, ya que al estar definida una evaluación final en la asignatura, el trabajo con el software no siempre influyó en la nota final del estudiante.

Espacial: Se elaboraron tres tareas generales para el trabajo en grupo, tres para el trabajo en equipo y tres para el trabajo individual.

Práctica: Se elaboraron cuatro tareas para el estudio del código fuente, cuatro para la modificación del código fuente y dos para la reutilización del código fuente.

Gestión: El software está amparado en una licencia libre, es multiplataforma y se encuentra disponible en la plataforma de enseñanza-aprendizaje Zera. Se definieron las características del software para su correcto funcionamiento en los laboratorios y aulas. Sin embargo, en las aulas no siempre se logró utilizar, por problemas con los recursos tecnológicos.

Tecnológica: El código fuente es legible, se utiliza el paradigma de programación para agentes con una arquitectura organizada con facilidades para modificar y adicionar nuevas funcionalidades. Se cuenta con el guion de contenido y ayudas en el mismo software, todos disponibles en la plataforma de enseñanza-aprendizaje Zera.

De forma general se aprecia un diseño didáctico del software libre educativo que se corresponde con los elementos definidos en cada subdimensión y que tributan al perfeccionamiento de la variable de estudio.

Dimensión incorporación armoniosa del software libre educativo al proceso de enseñanza-aprendizaje: En general los resultados de incorporar el software fueron satisfactorios. Por subdimensión se muestran los resultados a continuación.

Contextualizada: Se utilizó un diagnóstico para conocer el trabajo en la práctica profesional de los estudiantes y otros intereses profesionales y personales. Se asignaron las tareas de búsqueda de problemas a los propios estudiantes, lo cual contribuye a su implicación en el proceso. Todas las tareas personalizadas asignadas a los estudiantes, tomaron en cuenta o su trabajo en la práctica profesional o sus intereses.

Sistémica: El software fue diseñado para utilizarse en uno de los dos temas generales de la asignatura, pero su alcance permite asignar tareas de investigación a los estudiantes, antes

de comenzar el tema para el cual fue diseñado. Se considera que la no utilización del software afectaría el cumplimiento de los objetivos relacionados con el tema de minería de datos.

La reutilización dentro de la misma asignatura no fue posible con el software porque fue dirigido su diseño a uno de los dos temas. Sí se hizo efectiva la reutilización de resultados para los siguientes cursos, en tanto se analizaron todas las soluciones de los estudiantes y se seleccionaron dos de ellas para mejorar el software.

Reflexiva: El trabajo con el software tributa al tema de investigación del profesor y del RGA que formó parte del colectivo de la asignatura. Se reconoce que a los estudiantes les aportará no solo al vencimiento de los objetivos y al desarrollo de las habilidades específicas de la asignatura, también contribuye al desarrollo de habilidades generales para su futura vida profesional. Los propios estudiantes reconocen la utilidad del software, incluso para estudiar y reutilizar el código en su futura vida profesional.

Planificada: El profesor dedicó más de 20 horas totales para el diseño didáctico del SLE y para su autosuperación. Se planificaron y cumplieron las tareas con el software en un 31% de clases y se asignan y controlan dos trabajos independientes que implican más de seis horas para el estudiante.

Como parte del cierre de la sistematización se plantearon aspectos de interés en la experiencia y que plantearán tareas y retos a tomar en cuenta en próximas investigaciones:

1. El software diseñado ofrece un entorno genérico que permite resolver problemas de minería de datos en diversos campos de aplicación. Facilita la visualización del resultado de aplicar los algoritmos que modifica el estudiante. Sin embargo, no cubre toda la asignatura por lo que la reutilización de resultados de un tema a otro no fue posible.
2. El trabajo de los estudiantes con el software no siempre influyó en la nota final, dadas las características de la asignatura. Se sugiere modificar el programa analítico de las asignaturas de manera que tomen en cuenta la integración del software libre educativo.
3. El software brinda facilidades para su incorporación armoniosa al PEA de la IA. Se cuenta con un diseño didáctico que incluye tareas generales a realizar por los estudiantes en el PEA aprovechando las libertades de estudio, modificación y reutilización del código fuente. En la asignación de tareas a los estudiantes se toman en cuenta sus intereses y su trabajo en la práctica profesional, logrando una mayor contextualización del trabajo con el software libre educativo.

Los principales resultados de la sistematización fueron presentados en dos talleres metodológicos. El primero al colectivo de profesores de la disciplina IA en la carrera Ingeniería en Ciencias Informáticas. El segundo en el Departamento de Técnicas de Programación de la Facultad 4, integrado por profesores de las disciplinas Programación, Sistemas Digitales e Inteligencia Artificial. Adicionalmente, se diseñó un curso de postgrado con las principales

experiencias, tanto teóricas como metodológicas. Una primera edición del curso fue impartida como parte de la escuela de verano 2018 de la Universidad de las Ciencias Informáticas.

DISCUSIÓN

Se comparte con los autores consultados (Pavón, 2017; Pérez de Maza, 2016) que la sistematización de experiencias, es un proceso de reflexión crítica, donde se ordenan y relacionan datos, hechos, modos de actuar, para construir aprendizajes mediante su comunicación. Varios autores (Berrutti et al., sf; Pérez de Maza, 2016; Jara, 2018) han propuesto metodologías o pasos para llevarla a cabo. En el presente trabajo se proponen cuatro pasos generales, que simplifican, pero integran, los aspectos más relevantes del proceso. La sistematización se dirigió tanto a la producción de material didáctico, en este caso un software libre educativo, como una estrategia de enseñanza para su incorporación a una asignatura de Inteligencia Artificial. Estas son dos de las fuentes de experiencias más reconocidas en el contexto universitario (Pérez de Maza, 2016).

Con relación a las lecciones aprendidas y resultados alcanzados con la sistematización de experiencia, se consideraron elementos de interés para trabajos similares y futuros. Los software encontrados en la bibliografía consultada (López, Montaner, de la Rosa, & Rosa, 2001; Sosnowski et al., 2013; Riedl, 2015), por lo general, están basados en entornos de visualización gráfica. Varias experiencias van dirigidas a la utilización de entornos competitivos (Moreno Picot, Arevalillo Herráez, & Caveró Millán, 2010), otros incluso, utilizan robots reales (Arevalillo Herráez, Moreno Picot, & Caveró Millán, 2011). Esta característica se utiliza como contexto motivador, sin embargo, limita las posibilidades de práctica de los estudiantes en otros entornos de desarrollo. El software diseñado en la presente investigación es de propósito general, lo cual permite resolver problemas de minería de datos de una amplia gama de áreas.

Se comparte con los autores estudiados, la importancia de crear o diseñar el software a utilizar, según las necesidades de la asignatura. En la experiencia se brindan elementos que muestran la importancia del guion de contenido, tanto para el diseño del software como para su incorporación al proceso de enseñanza-aprendizaje.

En cuanto a resultados en la integración, se logra una asignatura más práctica, donde los estudiantes aplican los conocimientos. De manera particular se muestran en la experiencia tareas a los estudiantes, no solo de adicionar el código fuente. Las actividades de estudiar, modificar y reutilizar el código fuente permiten un trabajo gradual y más personalizado con el estudiante.

Como trabajos futuros se impone un estudio de los resultados académicos y motivacionales de los estudiantes. De igual forma se trabaja en el planteamiento de estrategias más integrales para formalizar y generalizar la experiencia para otras asignaturas de la carrera.

CONCLUSIONES

Para exponer las experiencias adquiridas en la integración de software libre educativo al proceso de enseñanza-aprendizaje de la Inteligencia Artificial en la carrera Ingeniería en Ciencias Informáticas se utilizó una sistematización de experiencias.

Los cuatro pasos generales (Planificación, recuperación del proceso vivido, reflexión crítica y cierre) permitieron presentar los elementos esenciales de la experiencia y los principales resultados alcanzados.

La sistematización de experiencias en el diseño didáctico del software libre educativo y su incorporación al proceso de enseñanza-aprendizaje de la Inteligencia Artificial permitió valorar las subdimensiones y dimensiones de la variable integración de software libre educativo al proceso de enseñanza-aprendizaje de la Inteligencia Artificial. Estas valoraciones permitieron consolidar un conjunto de lecciones aprendidas que encaminarán investigaciones posteriores.

REFERENCIAS

- Arevalillo Herráez, M., Moreno Picot, S., & Cavero Millán, V. (2011). Arquitectura para utilizar robots aibo en la enseñanza de la inteligencia artificial. *Revista Iberoamericana de Tecnologías del Aprendizaje*, 6(1), 1–9.
- Berrutti, L., Cabo, M., & Dabezies, M. J. (n.d.). *Sistematización de experiencias de extensión*. Montevideo, Uruguay: Comisión Sectorial de Extensión y Actividades en el Medio.
- Botía, J. A., González, J. C., Gómez, J., & Pavón, J. (2008). IWPAAMS2007-07: The Ingenias Project: Methods And Tool For Developing Multiagent Systems. *IEEE Latin America Transactions*, 6(6), 529–534. <https://doi.org/10.1109/TLA.2008.4908186>
- Coca, Y., & Pérez, M. T. (2018). Estudio sobre utilización de software libre educativo en el proceso de enseñanza-aprendizaje de la Inteligencia Artificial del ingeniero en Ciencias Informáticas. *Ciencia e Innovación tecnológica*. Las Tunas, Cuba: Editorial Académica Universitaria.
- Coca, Y., & Pérez, M. T. (2020a). Integración de software libre educativo al proceso de enseñanza-aprendizaje en carreras de informática. *Referencia Pedagógica*, 8(1), 167–181.
- Coca, Y., & Pérez, M. T. (2020b). Model for educational free software integration into Artificial Intelligence teaching and learning. *Cross Reality and Data Science in Engineering: Proceedings of the 17th International Conference on Remote Engineering and Virtual Instrumentation* (Auer, M.E.). Springer International Publishing.
- Coca, Y., Rosell, L. B., & Velazquez, A. (2017). Modelo de agente lógico con inferencia basada en hechos. *Revista Cubana de Ciencias Informáticas*, 11(2), 29–45.
- García, S., & Rico, P. (2014). *Concepción desarrolladora para el diseño didáctico del software*

- educativo de la escuela primaria*. La Habana, Cuba: Editorial Universitaria.
- Jara, O. (2018). *La sistematización de experiencias: práctica y teoría para otros mundos posibles*. Bogotá, Colombia: CEP-Centro de Estudios y Publicaciones Alforja.
- López, B., Montaner, M., de la Rosa, J., & Rosa, J. L. De. (2001). Utilización de un simulador de fútbol para enseñar inteligencia artificial a ingenieros. *Actas de Las Jornadas de Enseñanza Universitaria de La Informática*.
- Manso, Y., Cañizares, R., & Febles, J. P. (2016). Diseño web adaptativo para la plataforma educativa ZERA. *Revista Cubana de Ciencias Informáticas*, 10(2), 100–115.
- Moreno Picot, S., Arevalillo Herráez, M., & Caveró Millán, V. (2010). Utilización de Robots AIBO en la enseñanza de la asignatura Inteligencia Artificial. *XVI Jornadas de Enseñanza Universitaria de La Informática*, 515–518.
- Pavón, R. (2017). *Metodología con aprovechamiento de las TIC para el desarrollo de la competencia comunicativa en inglés académico-profesional en la carrera de Ingeniería en Ciencias Informáticas*. Tesis doctoral. Universidad de La Habana, La Habana, Cuba.
- Pérez de Maza, T. (2016). *Guía didáctica para la sistematización de experiencias en contextos universitarios* (U. N. Abierta, Ed.). Caracas: Ediciones del Vicerrectorado Académico.
- Ramos, N. (2016). *Una metodología para el proceso pedagógico de desarrollo de software educativo de Química en la Educación General cubana*. Tesis doctoral. Instituto Central de Ciencias Pedagógicas, La Habana, Cuba.
- Riedl, M. O. (2015). *A Python Engine for Teaching Artificial Intelligence in Games*. arXiv:1511.07714v1
- Rodríguez, L. A. (2010). *Concepción didáctica del software educativo como instrumento mediador para un aprendizaje desarrollador*. Tesis doctoral. Universidad de Ciencias Pedagógicas "Félix Varela y Morales", La Habana, Cuba
- Romero, M., & Patiño, A. (2018). Usos pedagógicos de las TIC: del consumo a la co-creación participativa. *Referencia Pedagógica*, (1), 2–15.
- Sosnowski, S., Ernsberger, T., Cao, F., & Ray, S. (2013). SEPIA: A Scalable Game Environment for Artificial Intelligence Teaching and Learning. *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 1592–1597.
- Velázquez, R., Velasteguí, E., & Arévalo, M. (2016). Los proyectos integradores como tipo de investigación formativa y forma de evaluación en UNIANDES. *Uniandes Episteme*, 3(3), 269–289.