

Tipo de artículo: Artículo original
Temática: Soluciones informáticas
Recibido: 20/12/16 | Aceptado: 25/02/17 | Publicado: 27/03/17

Módulo para el diseño de modelos entidad relación en la plataforma RDB-learning

Designing module of entity-relationship models in the platform RDB-learning

Carlos Yordan González Herrera^{1*}, Yaniel Lázaro Aragón Barreda¹, Eylín Hernández Luque¹, Yarileidis Barcena Calzado¹

¹Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Boyeros, Ciudad de La Habana, Cuba. { [cygonzalez](mailto:cygonzalez@estudiantes.uci.cu), [yaniell](mailto:yaniell@estudiantes.uci.cu), [ehernandezl](mailto:ehernandezl@estudiantes.uci.cu), [ybarcena](mailto:ybarcena@estudiantes.uci.cu) } @estudiantes.uci.cu

* Autor para correspondencia: cygonzalez@estudiantes.uci.cu

Resumen

La Universidad de las Ciencias Informáticas (UCI), es un centro de altos estudios para formar profesionales de la informática, dotado con tecnología y caracterizado por un uso intensivo de las Tecnologías de la Información y las Comunicaciones (TIC) en el proceso docente-educativo (PDE), incorporándose a este proceso el uso de la tecnología e-learning o aprendizaje electrónico. Dentro del Plan de estudio que emplea la UCI, se encuentra la asignatura Sistemas de Bases de Datos I (SBD I), la cual cuenta con dos habilidades genéricas necesarias para cumplir con el objetivo de la misma. Una de ellas es diseñar sistemas correctos para el almacenamiento de información. Los estudiantes continúan presentando problemas con la adquisición de esta habilidad, siendo una de las principales causas, la falta de tecnologías que faciliten la práctica. La herramienta utilizada en la asignatura, así como la plataforma educativa RDB-learning, no permiten el diseño de modelos entidad relación siguiendo las formas de representación impartidas en clases. La presente investigación presenta un módulo web de diseño para la plataforma RDB-learning, el cual permite la resolución de ejercicios de diseño a través de la representación de modelos entidad relación. Dicho módulo basa su funcionamiento en la tecnología e-learning, permitiendo la retroalimentación entre sus usuarios. Las pruebas de validación realizadas al MDBDR rigieron el desarrollo del mismo, lo cual permitió la

entrega al cliente de una solución libre y confiable, que pueda ser utilizada como apoyo al PDE de la asignatura SBD I en la UCI.

Palabras clave: diseño, e-learning, modelo entidad relación, retroalimentación

Abstract

The Computer Sciences University (CSU), is a center of advanced studies to instruct informatics-related professionals. It is endowed with technology and characterized by an intensive use of the Information and Communications Technologies (ICT) in the Educational-Teaching Process (ETP), in which the electronic learning use is sitting up. Within the Study Plan that the CSU uses, the subject Data Bases System I (DBS I) is founded, which counts with two generic and necessary skills to fulfill the objectives it has. One of them is to design correct systems for the information storage. The students keep having problems with the acquisition of this skill, being this one, one the mean causes, the lack of technologies that makes the practice-way easy. Both the tool used in the subject, as well as the educational platform RDB-learning, they do not permit the design of entity-relationship models following the representation ways given in classes. The current investigation presents a web module of design for the platform RDB-learning, which permits the designing exercises solution through the entity relationship models representation. The aforementioned module works under the e-learning technology, allowing the feedback among its users. The validation tests executed to the MRDBD addressed its own development, which allowed a free and reliable solution delivery to the customer that could be used as support to the subject DBS I's ETP in the CSU.

Keywords: design, e-learning, entity-relationship model, feedback

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) son el conjunto de herramientas aplicadas principalmente a la gestión de la información, que abarcan un abanico muy amplio de soluciones. Según la UNESCO (2017), estas desempeñan un papel cada vez más importante en la forma de comunicarnos, aprender y vivir. En la actualidad, las TIC se han desarrollado aceleradamente, lo que ha traído consigo un elevado auge en el desarrollo de la sociedad y de los procesos que en ella se suceden.

Según Granados y otros (2014), el entorno educativo es uno de los que mayores transformaciones ha sufrido con este acelerado desarrollo de las TIC. Es allí donde aparecen los conceptos de tecnologías del aprendizaje y el conocimiento (TAC), implicando la tecnología *e-learning* como recurso virtual de aprendizaje.

La introducción de la tecnología *e-learning* en el proceso de enseñanza-aprendizaje, si bien es considerada un reto para los docentes, desde el punto de vista de los estudiantes resulta más atractiva. En Cuba, con el uso de las TAC, se ha aplicado la tecnología *e-learning* en el proceso educativo. Esta vinculación, permite mejorar e innovar los métodos tradicionales de la enseñanza presencial, al brindar diversas modalidades para realizar estudios superiores, ya sea totalmente a distancia o mixto. La Educación Superior en Cuba, juega un destacado papel en la formación de profesionales y la superación permanente, y apuesta por el uso de la tecnología *e-learning* en el proceso educativo.

La Universidad de las Ciencias Informáticas (UCI), es un centro de estudios superiores dotado con tecnología y caracterizado por el uso de las TAC en el proceso docente-educativo. En la UCI se emplean diversas herramientas que mediante la práctica facilitan la adquisición de nuevos conocimientos, se pueden mencionar la plataforma educativa ZERA 2.0 y el Repositorio de Objetos de Aprendizaje RHODA.

En función de cumplir con la misión encomendada, la UCI cuenta con un Plan de Estudio de Ingeniería en Ciencias Informáticas (ICI), dentro del cual se incluye Sistemas de Bases de Datos I (SBD I) como una de las asignaturas esenciales en el segundo año de la carrera. Esta asignatura cuenta con dos habilidades genéricas necesarias para vencer los objetivos que tiene la misma. Una de ellas, es diseñar sistemas correctos para el almacenamiento de información, a través de la representación de modelos entidad relación (MER) basados en la teoría de los Sistemas de Bases de Datos Relacionales (SBDR), reflejada en el plan de estudios “D” de la carrera (Ministerio de Educación Superior, 2014).

Se debe tener en cuenta que con el avance de las TIC y el cúmulo de información circulante que estas generan, el uso de las bases de datos (BD) es cada vez mayor. En la actualidad, cualquier organización que se precie, por pequeña que sea, debe contar con una BD eficaz y bien definida desde sus inicios. Por esta razón, es de suma importancia realizar un correcto diseño de la misma, pues de ello dependerá que los datos estén correctamente actualizados, la información sea exacta y se puedan obtener reportes efectivos y eficientes que permitan a las instituciones tomar decisiones futuras.

En la UCI, dentro de los centros de desarrollo, está presente el rol de diseñador de base de datos y es SBD I, como se menciona con anterioridad, la asignatura encargada de garantizar estos conocimientos. Sin embargo, se evidencia que en el proceso de enseñanza-aprendizaje de la misma, los estudiantes presentan dificultades para diseñar MER, resultado que se muestra en los exámenes de la asignatura donde se mide este tema. Respecto a los tres últimos años académicos (2014-2017), en la Facultad 1 de la UCI, se tiene como resultado que aproximadamente el 62% de los estudiantes cometen errores al responder la pregunta de diseño en dichos exámenes. Entre las principales causas se encuentra la ausencia de tecnologías que faciliten la auto-preparación.

A pesar de existir un espacio para la asignatura en la plataforma ZERA, en ocasiones los profesores dejan a un lado la educación virtual y se enfocan más en la presencial. Ello se debe en gran medida, a que esta plataforma carece de herramientas que permitan conjugar los beneficios de ambas modalidades educativas, en pos de formar la habilidad diseñar MER desde un aprendizaje mezclado. Esto trae como consecuencia la necesidad de emplear una herramienta externa a la plataforma.

La herramienta *DB-Designer*, empleada en la asignatura durante una clase práctica de laboratorio, realiza el diseño de BD a través del modelo relacional (MR) y no a partir de un MER; por lo cual no contribuye al desarrollo de esta habilidad en los estudiantes. Otro de los inconvenientes de esta herramienta es que, al no ser una aplicación web, no hace uso de las potencialidades de la tecnología *e-learning*. Como consecuencia, no es posible la retroalimentación directa entre sus usuarios, la auto-preparación de los estudiantes guiada, controlada y evaluada por el profesor en espacios-tiempos diferentes, así como poder flexibilizar el ritmo de aprendizaje y el trabajo diferenciado con los estudiantes.

Como posible recurso, se encuentra en desarrollo hoy, en la Facultad 1 de la UCI, la plataforma educativa RDB-Learning, un soporte tecnológico para apoyar el proceso de enseñanza-aprendizaje de la asignatura SBD I en la Universidad. Basada en las TAC, la misma debe contribuir, mediante su uso, al desarrollo de las habilidades genéricas de la asignatura a través de las actividades de auto-preparación que realicen los estudiantes. En estos momentos, la plataforma no permite el diseño de MER, lo cual trae consigo, que los estudiantes no puedan usarla como herramienta de apoyo a la práctica de esta materia.

Teniendo en cuenta que el diseño de MER es considerado como la base para el entendimiento de los SBD a partir de su modelación inicial y la importancia que tiene un correcto diseño de una BD, surge el siguiente problema de investigación: ¿Cómo contribuir, mediante el uso de las tecnologías del aprendizaje y del conocimiento, al diseño de modelos entidad relación en la asignatura Sistemas de Bases de Datos I en la Facultad 1 de la Universidad de las Ciencias Informáticas?

El objeto de estudio de la presente investigación va dirigido a las herramientas informáticas para el diseño de modelos entidad relación, y el campo de acción a las tecnologías del aprendizaje y el conocimiento para el diseño de modelos entidad relación en la asignatura Sistemas de Bases de Datos I en la Facultad 1 de la Universidad de las Ciencias Informáticas.

Para dar solución al problema mencionado se plantea como objetivo general: desarrollar un módulo como tecnología del aprendizaje y el conocimiento para la plataforma RDB-Learning, que permita el diseño de modelos entidad relación en la asignatura Sistemas de Bases de Datos I en la Facultad 1 de la Universidad de las Ciencias

Informáticas.

Materiales y métodos

Teniendo en cuenta que el sistema a desarrollar es un módulo, este debe ser implementado en el mismo entorno de desarrollo del sistema al cual se debe integrar. Luego de analizar la documentación relacionada a la plataforma RDB-Learning, la cual documenta las herramientas, así como la metodología de desarrollo empleada, se especifica el uso del siguiente entorno de desarrollo:

Lenguaje de modelado

El lenguaje de modelado es cualquier lenguaje informático gráfico o textual que provee el diseño y construcción de estructuras y modelos siguiendo un conjunto sistemático de reglas y marcos (Techopedia, 2017). Como lenguaje de modelado se utiliza el Lenguaje Unificado de Modelado (*UML* por sus siglas en inglés) en su versión 2.1, que consiste en un lenguaje diseñado para visualizar, especificar, construir y documentar software orientados a objetos (Martínez, y otros, 2014).

Herramienta de modelado

Dentro de las principales herramientas para el modelado se encuentran herramientas *CASE* definidas por Araujo y otros (2013) como herramientas informáticas que asisten al diseñador en algunas de las actividades relacionadas con el desarrollo de un sistema (requerimientos, análisis, diseño, codificación y pruebas).

Visual Paradigm para *UML* es una herramienta para desarrollo de aplicaciones utilizando modelado *UML*, ideal para ingenieros de software, analistas de sistemas y arquitectos de sistemas, que están interesados en la construcción de sistemas a gran escala y necesitan confiabilidad y estabilidad en el desarrollo orientado a objetos. Para la presente investigación, se utiliza esta herramienta en su versión 8.0.

Herramienta para control de versiones

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante (Saldaña, 2017). Para garantizar el control de versiones en el sistema, y a su vez, la continua integración del módulo a la plataforma RDB-Learning, se emplea el repositorio GitLab, disponible en la UCI en la dirección <http://codecomunidades.prod.uci>, para esto, se propone el uso del cliente Git en su versión 2.8, el cual, según Saldaña (2017):

- Es un sistema de control de versiones distribuido
- No depende de acceso a la red o un repositorio central

- Está enfocado a la velocidad, uso práctico y manejo de proyectos grandes

Lenguaje de programación (lado del servidor)

Un lenguaje de programación está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones (Peralta, y otros, 2014). Para el desarrollo del módulo se emplea el lenguaje *Python* en su versión 2.7.11, el cual es un lenguaje de programación poderoso y fácil de aprender. Cuenta con estructuras de datos eficientes y de alto nivel y un enfoque simple pero efectivo a la programación orientada a objetos (Python, 2017).

Marco de trabajo

Un marco de trabajo o *framework* puede definirse, según TeachTerms (2013) como una plataforma para desarrollar aplicaciones de software. Proporciona una base en la que los desarrolladores de software pueden crear programas para una plataforma específica. Esto agiliza el proceso de desarrollo ya que los programadores no necesitan reinventar la rueda cada vez que desarrollan una nueva aplicación.

Para el desarrollo del módulo se utiliza el marco de trabajo *Django* en su versión 1.9.5, que es un marco de trabajo de desarrollo web totalmente implementado sobre *Python*, con el que se pueden crear y mantener aplicaciones de alta calidad e incluye un servidor web ligero que se puede usar mientras se desarrolla (Django, 2014).

Lenguaje de programación (lado del cliente)

El lenguaje para el lado del cliente, empleado en el desarrollo de la plataforma RDB-Learning, y por tanto propuesto para el desarrollo del módulo, es *JavaScript*, en su versión 1.8.5. Este, es un lenguaje interpretado que permite incluir macros en páginas web. Estas macros se ejecutan en el ordenador del visitante de dichas páginas, y no en el servidor, algo muy interesante, pues los servidores web suelen estar sobrecargados, mientras que las computadoras de los usuarios no suelen estarlo (Gutiérrez, 2009).

Librerías

Una librería es un conjunto de recursos (algoritmos) prefabricados, que pueden ser utilizados por el programador para realizar determinadas operaciones. Las declaraciones de las funciones utilizadas en estas librerías, junto con algunas macros y constantes predefinidas que facilitan su utilización, se agrupan en ficheros de nombres conocidos que suelen tener las extensiones lib, bpl, a, dll, js, etc. Las librerías utilizadas en el desarrollo del módulo son:

JQuery

JQuery es una biblioteca multiplataforma de *JavaScript*, que permite simplificar la manera de interactuar con los documentos *HTML*, manipular el árbol *DOM*, manejar eventos, desarrollar animaciones y agregar interacción con la

técnica *AJAX* a páginas web. Es además un software libre y de código abierto, permitiendo su uso en proyectos tanto libres como privados. *JQuery*, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en *JavaScript* que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio (Gutiérrez, 2009). Para la implementación del módulo se emplea la librería *JQuery* en su versión 3.1.1.

GoJS

También es empleada la librería *GoJS* en su versión 1.7.5. Esta, es una biblioteca de *JavaScript* con múltiples funciones para implementar diagramas interactivos personalizados y visualizaciones complejas en navegadores y plataformas web modernos. Facilita la construcción de diagramas con nodos, enlaces y grupos complejos con plantillas y diseños personalizables. Ofrece, además, muchas características avanzadas para la interactividad del usuario, tales como arrastrar y soltar, copiar y pegar, edición de texto en el lugar, información sobre herramientas, menús contextuales, diseños automáticos, plantillas, vinculación y modelos de datos, administración de estados transaccionales y deshacer, paletas, vistas generales, controladores de eventos, comandos y un sistema de herramientas extensible para operaciones personalizadas (GoJS, 2017).

Formato de intercambio de datos

JSON, acrónimo de *JavaScript Object Notation*, es un formato de texto ligero para el intercambio de datos. Es un subconjunto de la notación literal de objetos de *JavaScript*, aunque hoy, debido a su amplia adopción como alternativa a *XML*, se considera un formato de lenguaje independiente. Una de las ventajas de *JSON* como formato de intercambio de datos es la sencillez para escribir un analizador sintáctico (*parser*) de *JSON*. En *JavaScript*, un texto *JSON* se puede analizar fácilmente usando la función `eval()`, lo cual ha sido fundamental para que *JSON* haya sido aceptado por parte de la comunidad de desarrolladores *AJAX*, debido a la ubicuidad de *JavaScript* en casi cualquier navegador web (Gutiérrez, 2009). Por lo antes expuesto, se propone el uso del formato *JSON 3* para el desarrollo del módulo.

Entorno integrado de desarrollo (IDE por sus siglas en inglés)

JetBrains PyCharm es un entorno de desarrollo integrado de código abierto y multiplataforma desarrollado por *JetBrains*, que se utiliza para programar en *Python*. Proporciona un análisis de código, depuración gráfica, probador de unidad integrada y apoya el desarrollo web con el marco de trabajo Django. El mismo incluye heurísticas sofisticadas para determinar lo que cada tipo de variable es, y proporcionar sugerencias de autocompletado para dichas variables. También se puede aprovechar la información en tiempo de ejecución cuando se ejecuta la aplicación con el depurador integrado (*PyCharm*, 2015). Para la implementación del módulo se utiliza el *IDE PyCharm* en su

versión 2016.2.3.

Servidor de bases de datos

PostgreSQL es un servidor de bases de datos que incorpora el modelo relacional para sus BD y es compatible con el lenguaje de consulta *Structured Query Language (SQL)*, que significa lenguaje de consulta estructurado. Es ampliamente considerado como una de las alternativas de sistemas de BD de código abierto (PostgreSQL, 2014). Para la gestión de la BD del sistema se emplea *PostgreSQL* en su versión 9.4.

Servidor web

Para el alojamiento del sistema se emplea el servidor web *Apache 2.2*. Según Mifsuf (2013), Apache es un servidor web de software libre desarrollado por *Apache Software Foundation (ASF)*. Desde 1996 es el servidor más utilizado en Internet y es el utilizado en sistemas *GNU/Linux*. Es un servidor robusto y con un ciclo de desarrollo muy rápido, gracias a la gran cantidad de colaboradores con los que dispone.

Módulo de apache para python

Web Server Gateway Interface (WSGI)

WSGI es una interface simple y universal entre los servidores web y las aplicaciones web o *frameworks*. Es similar a la especificación *Java Servlet* o *ASP/ASP.NET*. En general, es mucho más simple que dichas especificaciones, y se basa en el estándar *CGI* con mejoras para hacerla reentrante y persistente (Python, 2017). Para la propuesta de solución se propone usar esta interfaz en su versión 4.5.10.

Herramientas para pruebas de software

Las pruebas de software, son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada (Pressman, 2010). En el caso de la propuesta de solución de la investigación, este será validado a través de varios tipos y técnicas de pruebas, algunas de ellas de forma manual y otras mediante el uso de herramientas que permiten la realización de dicha tarea. Entre las herramientas existentes, se pretende utilizar las herramientas *Acunetix Web Vulnerability Scanner* y *Apache JMeter*, para las pruebas de seguridad y de carga y estrés, respectivamente.

Acunetix realiza automáticamente auditorías a aplicaciones web comprobando vulnerabilidades de Inyección SQL, Cross site scripting y otras vulnerabilidades que puedan ser explotadas por hackers.

Apache JMeter se caracteriza por ser libre de uso y mostrar los resultados de las pruebas en una amplia variedad de informes y gráficas, con gran cantidad de variables que permiten interpretar los resultados desde diferentes puntos de vista.

Metodología de desarrollo

La metodología ágil *AUP*, es una versión simplificada del Proceso Unificado de *Rational (RUP)*. Esta describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en *RUP*. El *AUP* aplica técnicas ágiles incluyendo (Sánchez, 2015):

- Desarrollo Dirigido por Pruebas (*test driven development - TDD* en inglés)
- Modelado ágil
- Gestión de Cambios ágil
- Refactorización de Base de Datos para mejorar la productividad

Según Sánchez (2015), al no existir una metodología de software universal, ya que toda metodología debe ser adaptada a las características de cada proyecto (equipo de desarrollo, recursos, etc.), exigiéndose así que el proceso sea configurable, se decide hacer una variación de la metodología *AUP*, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI. Esta variación está unida al modelo *CMMI-DEV v1.3*, para garantizar las buenas prácticas en función de un software de calidad. La plataforma RDB-Learning, a la cual debe integrarse el módulo propuesto, fue desarrollada bajo esta metodología. Para el desarrollo de la propuesta de solución, se decide usar la metodología *AUP*, en su variación para la UCI, teniendo en cuenta todo lo antes expuesto.

Resultados y discusión

El módulo para el diseño de modelos entidad relación (MDMER) en la plataforma RDB-learning, se enmarca dentro de las aplicaciones web con fines educativos para el diseño de MER. Una de las características del módulo, es que el mismo permite a los usuarios (estudiantes) resolver ejercicios de diseño de MER.

Para lograr esto, el módulo estará integrado a una plataforma web educativa que permite la selección de los ejercicios disponibles, la visualización de los mismos y su posterior realización. Una vez creada la respuesta, los estudiantes tienen la posibilidad de mandar su respuesta de forma privada a ser revisada por el profesor o de enviarla de forma pública a un foro y que se emitan comentarios sobre la misma, siempre y cuando el ejercicio haya estado asociado a dicho foro. Una vez revisada la respuesta por el profesor, este puede emitir una evaluación y un criterio y enviarla al estudiante, también a través de la plataforma.

Validación del sistema

Durante la etapa de implementación, pueden cometerse algunos errores y pueden pasarse por alto algunos elementos que son importantes para el correcto funcionamiento del sistema. Por tal motivo, es esencial llevar a cabo la fase de

validación, en la cual, a través de varios tipos y técnicas de pruebas de software, se pretende comprobar el cumplimiento de las especificaciones del diseño y de la codificación, identificar los posibles errores cometidos y validar la solución propuesta. A continuación, se muestran los resultados de las distintas pruebas aplicadas al MDMER, en función de garantizar y validar su calidad.

Pruebas funcionales

Se denominan pruebas funcionales, a las pruebas de software que tienen por objetivo probar que los sistemas desarrollados, cumplan con las funciones específicas para los cuales han sido creados. A este tipo de pruebas se les denomina también pruebas de comportamiento o pruebas de caja negra, pues los probadores o analistas de pruebas, no enfocan su atención a cómo se generan las respuestas del sistema, básicamente el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y en los de salida, esto generalmente se define en los casos de prueba preparados antes del inicio de las pruebas (Oré, 2009).

Como resultado final de las pruebas funcionales, se obtuvo, en una primera iteración, un total de diez (10) no conformidades, de las cuales se resolvieron seis (6), y cuatro (4) quedaron pendientes. En una segunda iteración, se identificaron tres (3) nuevas no conformidades, para un total de siete (7), sumándole las que quedaban pendientes. De estas siete (7), solo una (1) quedó pendiente a ser resuelta. Para la tercera iteración, no se identificaron nuevas inconformidades y se resolvió la que estaba pendiente, obteniendo, de esta manera, resultados satisfactorios.

Pruebas de seguridad

Las pruebas de seguridad se realizan para comprobar que los mecanismos de protección integrados en el sistema realmente lo protejan de irrupciones inapropiadas (Pressman, 2010). Además, se encargan de certificar que los datos y las funciones del sistema solo son accesibles por los actores debidamente autorizados (Toll, y otros, 2007).

Para la realización de este tipo de prueba, se empleó la herramienta *Acunetix Web Vulnerability Scanner*, caracterizada en el epígrafe 1.5. En una primera iteración, se obtuvo un total de quince (15) no conformidades, divididas en seis (6) de nivel medio, seis (6) de nivel bajo y tres (3) de carácter informativo. De las de nivel medio, destacó el uso del protocolo no seguro para el envío de datos, así como los mensajes de error que se muestran en el modo DEBUG de Django para el desarrollo. Las de nivel bajo estuvieron relacionadas con problemas para la protección contra ataques de fuerza bruta a la página de autenticación, así como directorios que pueden ser accesibles directamente sin pasar la autenticación y la protección de las cookies y las sesiones en el navegador. De carácter informativo fueron detectadas una dirección de correo y una posible cuenta de usuario en un fichero. Todas estas deficiencias fueron corregidas en la primera iteración, y para una segunda, no se identificó ninguna nueva, por lo cual se obtuvo finalmente una herramienta que cumple con los requisitos de seguridad definidos para la misma.

Interfaces principales del MDBDR

Una vez desarrollado el software es posible visualizar las pantallas principales del mismo donde se observa el resultado obtenido después de la implementación.

La *figura 1* muestra la interfaz de autenticación donde los usuarios se autenticarán con dominio uci.cu.



Fig 1. Interfaz de autenticación

La *figura 2*, muestra la pantalla principal que se muestra una vez autenticado el usuario con el rol profesor, como se puede observar, el mismo tiene acceso a todas las secciones de la plataforma. Si el usuario autenticado fuese un estudiante, solo tendrá acceso a las secciones de foro, diseño e implementación, como se muestra en la *figura 3*.



Fig 2. Interfaz de la página principal para profesores

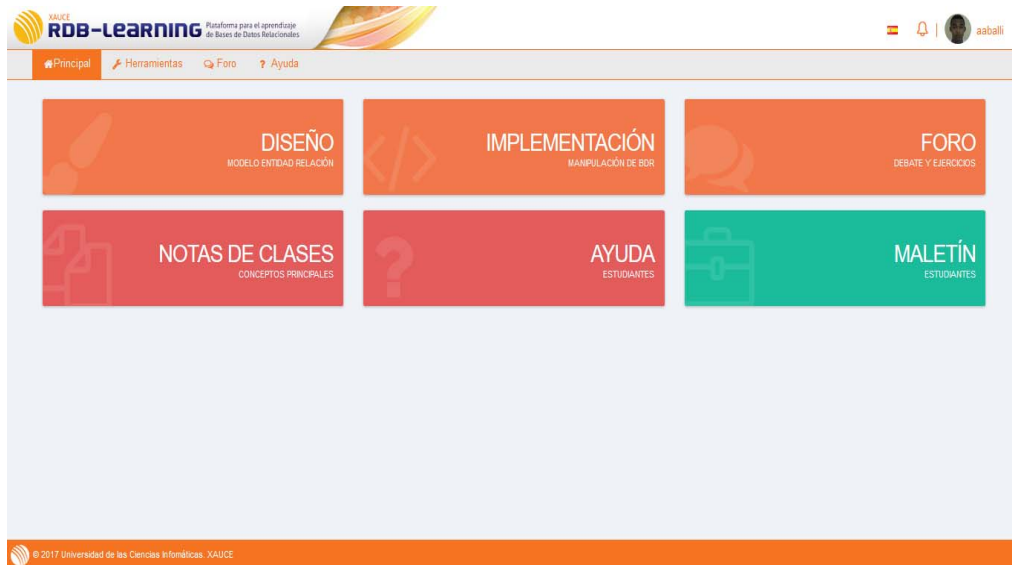


Fig 3. Interfaz de la página principal para estudiantes

Una vez que el estudiante accede al módulo de diseño, se mostrará el listado de los ejercicios disponibles (ver *figura 4*) en el sistema, de los cuales, el podrá seleccionar la acción de ver detalles del ejercicio, o resolver el ejercicio.

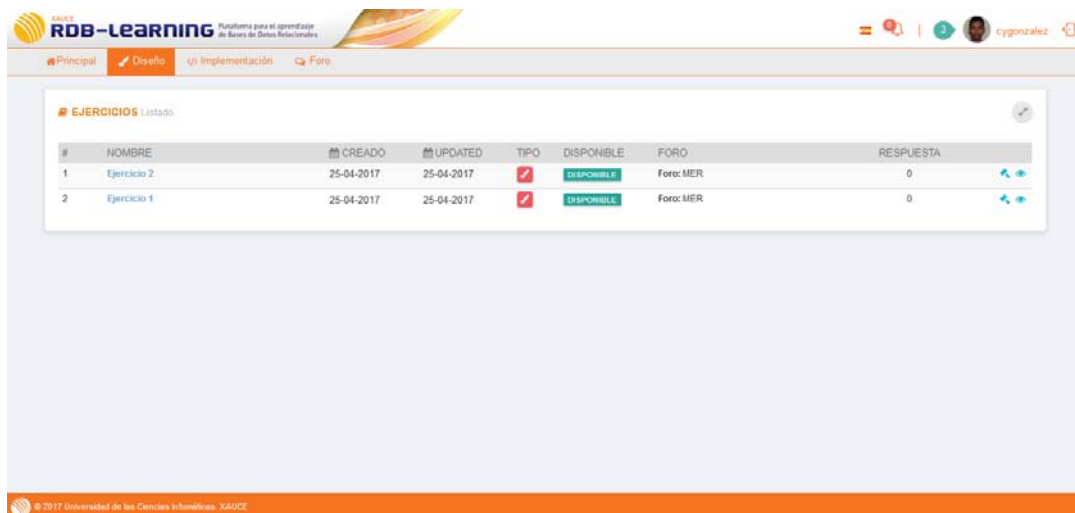


Fig 4. Interfaz del listado de ejercicios de diseño

Al seleccionar la opción resolver, se muestra la interfaz (ver figura 5) donde el estudiante dispondrá de la orden del ejercicio seleccionado y un área de trabajo donde podrá realizar el diseño del MER correspondiente a dicho ejercicio.

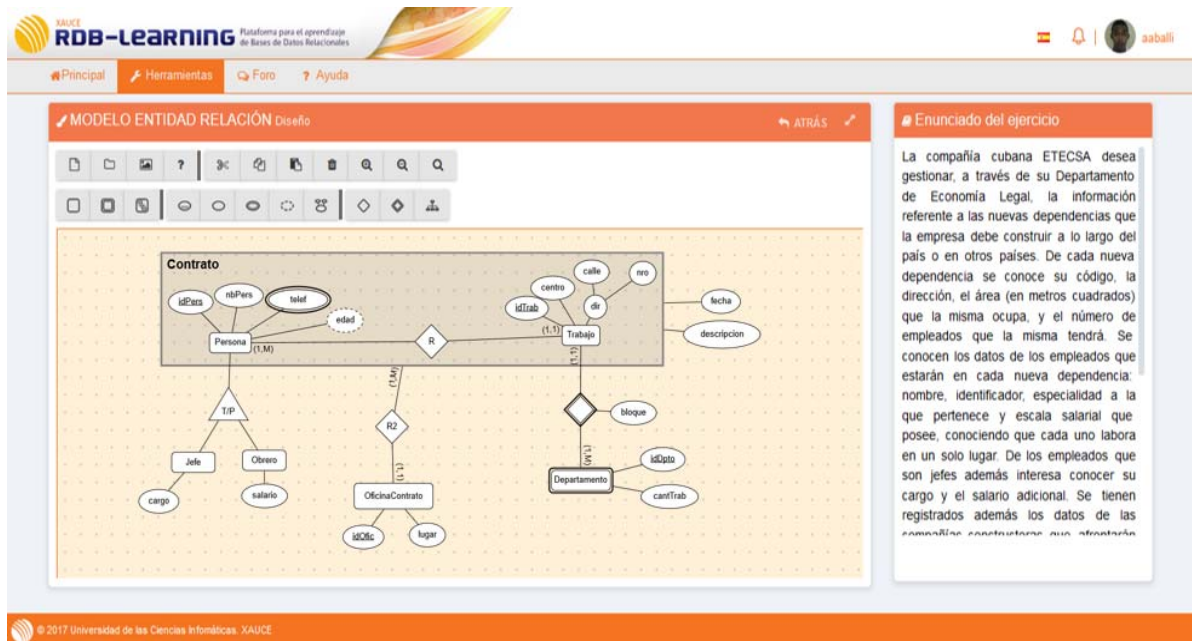


Fig 5. Interfaz para resolver ejercicios de diseño

Conclusiones

En la presente investigación se presentó el Módulo para el diseño de modelos entidad relación en la plataforma RDB-Learning, el cual sirve de apoyo al proceso de enseñanza – aprendizaje de la asignatura SBD I y contribuye al diseño de modelos entidad relación en dicha asignatura. Esto se debe a que el módulo fue desarrollado como una tecnología del aprendizaje y el conocimiento que facilita al estudiante sus actividades de auto-preparación, y al profesor, llevar el control de la realización de estas actividades.

Referencias

- ARAUJO, PEDRO BERNABÉ Y RODRÍGUEZ, SEBASTIÁN ALBERTO. 2013. *Janeiro Studio*. 2013.
- DJANGO. 2014. Django Software Foundation. *The Django Book*. [En línea] 2014. <http://www.djangobook.com>.
- GOJS. 2017. GoJS. *Interactive JavaScript Diagrams in HTML*. [En línea] 2017. [Citado el: 30 de Abril de 2017.] <https://gojs.net>.

GRANADOS, JOHN; FERNÁNDEZ, RAÚL LÓPEZ; MARTÍNEZ, RAIDELL AVELLO; ÁLVAREZ, DIEGO LUNA; ÁLVAREZ, ENRIQUE LUNA; ÁLVAREZ, WALTER LUNA. 2014. Las tecnologías de la información y las comunicaciones, las del aprendizaje y del conocimiento y las tecnologías para el empoderamiento y la participación como instrumentos de apoyo al docente de la universidad del siglo XXI. Ecuador : Universidad de Guayaquil, 2014. ISSN 1727-897X.

GUTIÉRREZ, EMMANUEL. 2009. *JavaScript: Conceptos básicos y avanzados*. Barcelona : Informática Técnica, 2009. ISBN: 978-2-7460-4341-1.

MARTÍNEZ, ALEJANDRO Y MARTÍNEZ, RAÚL. 2014. Guía a Rational Unified Process. Albacete : Escuela Politécnica Superior de Albacete. Universidad de Castilla la Mancha, 2014.

MIFSUF, E. 2013. Apache. España : s.n., 2013.

MINISTERIO DE EDUCACIÓN SUPERIOR. 2014. Plan de estudios "D" Ingeniería en Ciencias Informáticas. La Habana : s.n., 2014.

ORÉ, ALEXANDER. 2009. FUNCTIONAL TESTING - PRUEBAS FUNCIONALES. *CalidadySoftware.com*. [En línea] 2009. [Citado el: 2017 de Abril de 10.] http://www.calidadyssoftware.com/testing/pruebas_funcionales.php.

PERALTA, CLAUDIA Y DURÁN, D. 2014. *Módulos de edición de plantillas y recepción de órdenes de impresión para el Sistema de Personalización de Documentos de Identidad basado en tecnologías libres. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas*. La Habana : Universidad de las Ciencias informáticas, 2014.

POSTGRESQL. 2014. PostgreSQL.org. [En línea] Diciembre de 2014. [Citado el: 20 de Noviembre de 2016.] <https://www.postgresql.org/about/press/presskit94/es/>.

PRESSMAN, ROGER S. 2010. *Ingeniería de Software, un enfoque práctico. Séptima Edición*. Madrid : s.n., 2010.

PYCHARM. 2015. JetBrains. [En línea] Diciembre de 2015. <http://www.jetbrains.com/pycharm/>.

PYTHON. 2017. Guía de estilo para el código Python – PEP 8 en Español. *www.recursopython.com*. [En línea] 2017. [Citado el: 12 de Abril de 2017.] www.recursopython.com/pep8es.pdf.

SALDAÑA, GABRIEL. 2017. Nethazard. [En línea] 2017. [Citado el: 30 de Abril de 2017.] <http://blog.nethazard.net>.

SÁNCHEZ, TAMARA RODRÍGUEZ. 2015. Metodología de desarrollo para la Actividad productiva de la UCI v1.2. La Habana, Cuba : s.n., 6 de Marzo de 2015.

TEACHERMS. 2013. TeachTerms. *Framework Definition*. [En línea] Marzo de 2013. <https://techterms.com/definition/framework>.

TECHOPEDIA. 2017. Modeling Lenguage. [En línea] 2017.
<https://www.techopedia.com/definition/20810/modeling-language>.

TOLL, YUNIET Y MENDOZA, YILENNIS. 2007. Propuesta de manual de procedimiento de Pruebas de Sistema y su aplicación en el Proyecto CICPC. [En línea] 2007. [Citado el: 20 de Abril de 2017.]
http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_0717_07/1/TD_0717_07.pdf.

UNESCO. 2017. UNESCO. *Building peace in the minds of men and women*. [En línea] 2017.
<http://en.unesco.org/themes/ict-education>.