

Tipo de artículo: Artículo original
Temática: Soluciones informáticas
Recibido: 12/03/2019 | Aceptado: 14/04/2019 | Publicado: 20/06/2019

Sistema informático para la gestión de reportes de incidencias de mantenimiento en la Facultad de Ciencias y Tecnologías Computacionales

Computer system for the management of incidents of maintenance incidents in the Faculty of Sciences and Computational Technologies

Bárbara Bron Fonseca¹, Katy Mestre Kelly², William Simón Grass³

¹ Facultad de Ciencias y Tecnologías Computacionales, Universidad de las Ciencias Informáticas, bbron@uci.cu

² Facultad de Ciencias y Tecnologías Computacionales, Universidad de las Ciencias Informáticas, kmestre@estudiantes.uci.cu

³ Facultad de Ciencias y Tecnologías Computacionales, Universidad de las Ciencias Informáticas, wgrass@uci.cu

* Autor para correspondencia: bbron@uci.cu

Resumen

La Facultad de Ciencias y Tecnologías Computacionales de la Universidad de las Ciencias Informáticas, es un área que, desde el punto de vista administrativo, posee una infraestructura con un alto número de recursos materiales y constructivos. El Vicedecanato de Economía y Administración de esta facultad, decidió crear un equipo de mantenimiento que atienda problemas puntuales, que no requieran grandes recursos materiales. Este pequeño equipo, también cuenta con el apoyo del proyecto extensionista Aventuras de Paquito, integrado por estudiantes y trabajadores que se encargan de manera voluntaria, de realizar reparaciones a equipos e instalaciones en el edificio donde radica la facultad. El presente trabajo tiene como objetivo, desarrollar un sistema informático que gestione los reportes de incidencias de mantenimiento generados en la facultad, para que sean atendidos de manera interna. Para el análisis, diseño e implementación del sistema *web*, la autora optó por la utilización de la metodología AUP-UCI en su escenario número cuatro, se utilizó la herramienta *Visual-Paradigm* para el modelado. Como lenguajes de programación PHP y *Java Script*; *Symfony* como marco de trabajo, *NetBeans* como Entorno de Desarrollo Integrado, Apache como servidor *web* y *MySQL* como sistema gestor de base de datos. Como resultado, se obtuvo un sistema capaz de gestionar los reportes de incidencias, generados dentro de la facultad.

Palabras clave: Gestión de reportes de incidencias, mantenimiento, sistema informático.

Abstract

The Faculty of Computer Science and Technology of the University of Computer Sciences is an area that from the administrative point of view has an infrastructure with a high number of material and constructive resources. The vice-dean of economics and administration of this faculty decided to create a maintenance team that addresses specific problems of the faculty that do not require large material resources. This small team also has the support of the Aventuras de Paquito extension project integrated by students and workers who are in charge of voluntarily making repairs to equipment and facilities in the building where the faculty is located. The objective of this work is to develop a computer system that manages the reports of maintenance incidents generated in the faculty so that they can be attended internally. For the analysis, design and implementation of the web system, the author opted for the use of the AUP-UCI methodology in its scenario number four, using the Visual-Paradigm tool for modeling. As programming languages PHP and Java Script; Symfony as a framework, NetBeans as an Integrated Development Environment, Apache as a web server and MySQL as a database management system. As a result, a system capable of managing the incident reports generated within the faculty was obtained.

Keywords: *Incident report management, internal maintenance.*

Introducción

La sociedad de la información, es aquella que basa su estructura de relaciones económicas, políticas, sociales y de ocio, en los sistemas y redes de telecomunicaciones. Cambios que en un principio afectaban a aquellos sectores más implicados directamente en el campo de la información, se han hecho extensibles a cualquiera de las actividades y procesos profesionales de las organizaciones. Los gobiernos de todos los países y a todas las escalas, han tenido que promover y propiciar la constitución de redes de información y su gestión. Esta nueva situación está propiciando un cambio, desde una sociedad que basaba su riqueza en la producción de bienes de consumo, a otra, cuya base de riqueza es la gestión de la información y el conocimiento. Se requiere, por tanto, de la utilización de nuevos enfoques en la solución de problemas y procesos, que sirvan para mejorar la vida cotidiana; tratándose fundamentalmente de un nuevo paradigma de desarrollo, cuya base de riqueza son los Sistemas de Información (MARTÍN-PENA *et al.* 2018). Por otra parte, las Tecnologías de Información (TI), se han convertido en elementos de gran relevancia y creciente influencia, tanto en la sociedad, como en la economía actual, dejando de ser simples herramientas y constituyéndose en factores claves para el correcto desempeño del giro del negocio, en las nuevas sociedades económicas y sus procesos (PEDRO 2009), (DOÑA 2008). La incorporación de las TI en las organizaciones, ayuda a reducir costos, recursos, entre otros. Las TI han cambiado el curso de la historia, porque han traído consigo diferentes formas de trabajar, de pensar, de relacionarse con las personas y otros. La automatización de su gestión se ha convertido en una herramienta imprescindible y clave para las empresas u organizaciones, generando ventajas competitivas (ORTIGUEIRA and GÓMEZ 2016).

En el momento actual, parece indiscutible que el éxito de la empresa no dependerá únicamente de cómo maneje sus activos materiales, sino también de la gestión de los recursos de información. La importancia de este recurso, es tal, que algunos autores estiman, que las organizaciones deben ser consideradas como sistemas de información.

Sin embargo, se debe considerar que las TI requieren tener una administración adecuada que se componga de métricas definidas y así poder medir el verdadero desempeño y valor que están aportando a los fines del negocio. Existen diferentes estándares para implementar la gestión de las TI y la elección de uno de ellos depende de las necesidades de cada organización. Con la introducción, desde hace ya muchos años de la informática en la gestión empresarial, se han multiplicado enormemente las posibilidades de interacción, entre los integrantes de una organización, acercando la información y haciendo más fácil la comunicación (GARCÍA and MONTERO 2013), (GÓMEZ *et al.* 2013).

Existen distintas soluciones informáticas para facilitar la comunicación, la organización de los mensajes según su naturaleza y la generación y adecuación de respuestas. En este caso, se recomienda el uso de una herramienta informática que comunique desde cualquier punto y en cualquier momento toda la información que se genere, clasificándola y guardándola en una base de datos. Los servicios de tecnología han dejado de ser procesos y funciones complementarias de una organización o empresa, para formar parte de sus activos estratégicos.

La Facultad de Ciencias y Tecnologías Computacionales de la Universidad de las Ciencias Informáticas, es un área que, desde el punto de vista administrativo, posee una infraestructura con un alto número de recursos materiales y constructivos (HERNÁNDEZ GONZÁLEZ *et al.* 2019). El Vicedecanato de Economía y Administración de esta facultad, decidió crear un equipo de mantenimiento que atienda problemas puntuales, que no requieran grandes recursos materiales. También cuenta con el apoyo del proyecto extensionista “Aventuras de Paquito”, integrado por estudiantes y trabajadores, que se encargan de manera voluntaria de realizar reparaciones a equipos e instalaciones en el edificio donde radica la facultad. Debido al deterioro de sus locales, se ha hecho necesario el desarrollo de nuevas formas para elevar el control interno en el área administrativa del docente, lo que respecta al trabajo de mantenimiento y reportes de incidencias en la facultad.

Una de las situaciones que más comúnmente se presenta está relacionada con el control de los reportes:

- ✓ Existe saturación de la información, y es difícil conocer en qué estado se encuentra el reporte.

Las personas que trabajan en esta área deben atender gran número de llamadas, lo que provoca que exista el riesgo de que se pierdan datos específicos del problema que informan.

Materiales y métodos

En la presente sección se detallan las características del sistema, a partir de la descripción general de la propuesta de solución. Se identifican los requisitos funcionales y no funcionales del sistema, con la respectiva descripción de cada uno de ellos y se modelan los elementos del diseño del sistema (ZAMBRANO *et al.* 2019), (MAR, O *et al.* 2016).

A partir de las deficiencias identificadas anteriormente en la Facultad de Ciencias y Tecnologías Computaciones (CITEC) de la UCI en el proceso de gestión de incidencias y para dar solución al problema planteado en la investigación, se propone desarrollar un sistema que permita generar las incidencias de los trabajadores de todas las áreas de la facultad (SALAS and CERÓN 2014), (MAR CORNELIO *et al.* 2013),(MAR, OMAR *et al.* 2015) . El sistema debe ser administrado por los especialistas del Vicedecanato de Administración y Economía, permitiendo la generación de reportes con las incidencias generadas y el estado en que se encuentra, este sistema deberá estar compuesto por los siguientes módulos:

- Módulo Administración: Debe permitir agregar los usuarios en el sistema, la creación de las áreas de la facultad y la asignación de roles.
- Módulo Incidencias: Debe permitir crear las incidencias y seguir el estado en que estas se encuentran.
- Módulo Reporte: Debe permitir la emisión de informes de todas las incidencias que han sido registradas independientemente de su estado.

A continuación, se describen los requerimientos funcionales que deberá cumplir el software planteado determinando su comportamiento. Según la entrevista realizada a los usuarios, se han definido los siguientes requerimientos funcionales:

- RF 1. Autenticar usuario.
- RF 2. Adicionar usuario
- RF 3. Modificar usuario.
- RF 4. Eliminar usuario.
- RF 5. Buscar usuario por el nombre de usuario.
- RF 6. Asociar rol a usuario.
- RF 7. Adicionar área.
- RF 8. Modificar área.
- RF 9. Eliminar área.
- RF 10. Buscar área por nombre de área.
- RF 11. Crear incidencia.
- RF 12. Actualizar estado de la incidencia.

RF 13. Generar reporte: listado de todas las incidencias agregadas al sistema (Solo para el Rol Especialista y el Rol Vicedecano Administrativo).

RF 14. Eliminar incidencia (Solo para el Rol Técnico en administración y el Rol Vicedecano Administrativo).

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener, forman una parte significativa de la especificación. Tienen la característica de marcar la diferencia, pues una vez comprobado que el producto cumple con lo requerido por el cliente, las propiedades no funcionales; como cuán usable, seguro, conveniente y agradable, distinguen a un producto bien aceptado de uno con poca aceptación.

Para un correcto funcionamiento, el sistema debe tener como características principales para su funcionamiento los siguientes requisitos no funcionales:

Usabilidad:

RNF 1. El sistema debe cumplir con el estándar de usabilidad propuesto por la Dirección de Informatización de la UCI, donde las operaciones deben realizarse con la utilización de la menor cantidad de acciones posibles, los elementos visuales estén orientados hacia las áreas visibles de las interfaces, los íconos sean identificativos, entre otros propuestos, lo que permite que personas que no tenga conocimientos básicos en el manejo de la computadora, puedan operar el sistema.

RNF 2. El diseño será encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.

RNF 3. Para la realización del diseño se tendrá en cuenta el control y la transparencia como eje principal en torno al cual girará el sistema, además debe permitir el uso de colores claros y legibles.

Seguridad:

RNF 4. El acceso al sistema y sus funcionalidades debe ser mediante la asignación de permisos que tenga cada uno de los roles del sistema.

Software:

Para las PC Cliente

RNF 5. Navegadores *Internet Explorer* v8.0 o *Mozilla Firefox* v44.0.

Para la PC Servidor

RNF 6. Sistema Operativo *Nova* o *Windows*.

RNF 7. Servidor *Web Apache* 2.2.4 o superior.

RNF 8. Servidor de Bases de Datos *MySQL*.

Hardware:

Para los clientes del sistema

- RNF 9. Se necesitan como requerimientos mínimos una Computadora Personal (*Personal Computer*) PC con procesador *Pentium IV*.
- RNF 10. El microprocesador debe poseer una velocidad a 1.0 GHz.
- RNF 11. La memoria RAM de 256 MB.
- RNF 12. Debe poseer tarjeta de red.

Para servidor de aplicaciones

- RNF 13. Se necesitan como requerimientos mínimos una PC con procesador *Pentium IV*. Microprocesador con una velocidad de 3.0 GHz.
- RNF 14. La memoria RAM debe ser de 1GB.
- RNF 15. El Disco duro debe ser de 80 GB.
- RNF 16. Debe poseer tarjeta de red.

Rendimiento:

- RNF 17. El sistema debe responder a las peticiones del usuario en un tiempo promedio no superior a 1200 milisegundos.

Resultados y discusión

Las historias de usuarios son una explicación breve de la funcionalidad del sistema tal y como lo desee el cliente, describen lo que se quiere implementar y se escriben con una o dos frases, utilizando el lenguaje común del usuario. Estas historias son aprovechadas en las metodologías ágiles para la precisión de los requisitos del sistema. A continuación, se muestran las historias de usuarios para los requisitos: crear incidencia, actualizar estado de la incidencia y adicionar usuario. Ver tablas 1, 2 y 3:

Tabla 1. Historia de usuario: crear incidencia.

HU: “crear incidencia”	
Número: 1	Nombre del Requisito: crear incidencia
Programador: Katy Mestre Kelly	Iteración asignada: 1

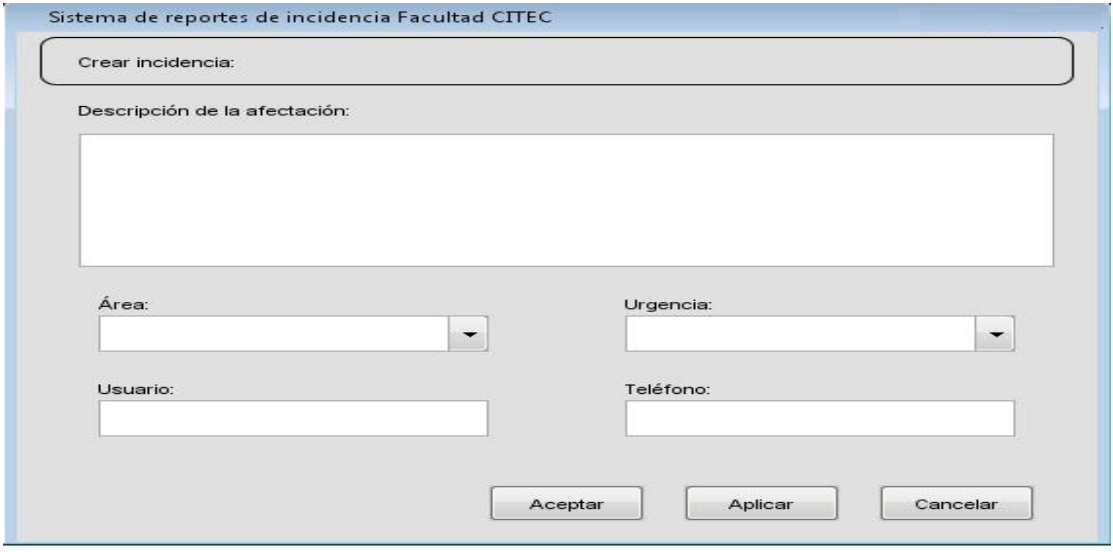
Prioridad: alta	Tiempo estimado: 4 días
Riesgo en desarrollo: ✓ Interrupción del servicio eléctrico. ✓ Planificación irreal.	Tiempo real: 3 días
Precondiciones: debe estar creada, al menos un Área en el sistema.	
Descripción: Debe permitir que el usuario con rol Trabajador adicione una incidencia al sistema. Se debe ofrecer un campo de texto por cada dato que requiera la incidencia. Las incidencias deben poseer los siguientes datos: <ul style="list-style-type: none"> ✓ Servicio ✓ Afectación ✓ Área ✓ Descripción ✓ Estado: nuevo, en espera, rechazado, resuelto, cerrado. 	
Prototipo de Interfaz 	

Tabla 2. Historia de usuario: actualizar estado de la incidencia.

HU: “actualizar estado de la incidencia”	
Número: 2	Nombre del Requisito: actualizar estado de la incidencia
Programador: Katy Mestre Kelly	Iteración asignada: 1
Prioridad: Alta	Tiempo estimado: 2 días
Riesgo en desarrollo: ✓ Interrupción del servicio eléctrico. ✓ Planificación irreal.	Tiempo real: 1 día
Precondiciones: Debe estar creada al menos una incidencia en el sistema	
Descripción: Debe permitir que el usuario con rol Especialista o Vicedecano Administrativo, actualice el estado de la incidencia. Se debe ofrecer un campo de texto por cada estado que requiera la incidencia. Los estados de incidencias deben poseer los siguientes datos: <ul style="list-style-type: none"> ✓ Nuevo ✓ En espera ✓ Rechazado ✓ Resuelto ✓ Cerrado 	
Prototipo de Interfaz	



Tabla 3. Historia de usuario: adicionar usuario.

HU: “adicionar usuario”	
Número: 3	Nombre del Requisito: adicionar usuario
Programador: Katy Mestre Kelly	Iteración asignada: 1
Prioridad: Alta	Tiempo estimado: 3 días
Riesgo en desarrollo: ✓ Interrupción del servicio eléctrico. ✓ Planificación irreal.	Tiempo real: 3 días
Precondiciones:	
Descripción: Debe permitir que el usuario con rol Vicedecano Administrativo adicione un usuario al sistema. Se debe ofrecer un campo de texto por cada dato que requiera el usuario. Los usuarios deben poseer los siguientes datos: ✓ Nombre y Apellidos ✓ Número de solapín ✓ Cargo ✓ Categoría	

✓ Teléfono
✓ Usuario UCI

Prototipo de Interfaz

Sistema de reportes de incidencia Facultad CITEC

Adicionar usuario:

Nombre y Apellidos:	Número de solapín:
<input type="text"/>	<input type="text"/>
Cargo:	Categoría:
<input type="text"/>	<input type="text"/>
Área:	Teléfono:
<input type="text"/>	<input type="text"/>
Usuario:	Contraseña:
<input type="text"/>	<input type="text"/>

Aceptar Aplicar Cancelar

Jean Cartejon (2014) indica que: “se establece que el patrón fundamental para el diseño del sistema *web* es el Modelo–Vista–Controlador (MVC), porque propone la separación en distintos componentes de la interfaz del usuario (vistas), el modelo del negocio y la lógica de control”.

La estrategia de MVC, se basa en la separación del código en tres capas diferentes, acotadas por su responsabilidad, llamadas Modelo, Vista y Controlador. Fue creada hace varias décadas, incluso antes de la aparición de la *web*. No obstante, en los últimos años ha ganado mucha fuerza y seguidores, gracias a la aparición de numerosos marcos de trabajo para el desarrollo *web*, que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones.

- **Modelo:** representa las reglas de negocio de la aplicación (y el acceso a datos subyacente). Representa toda la lógica de negocio y el acceso a datos. Consiste en el conjunto de objetos que modelan los procesos de negocio, que se realizan a través del sistema.
- **Controlador:** el controlador o capa de negocio implementa las reglas de negocio a respetar en la aplicación. Se comunica directamente con el Modelo o Capa de Datos (a quien hace peticiones de datos) y con la Vista o Capa de Presentación (con quién interactúa para buscar y mostrar datos al usuario).
- **Vistas:** representan la presentación de la aplicación. Las vistas contienen el código enviado al navegador, es decir el código HTML y código de servidor asociado, siempre y cuando implique tareas de presentación, no de lógica de negocio. Denominada como una “fotografía” del modelo o parte del mismo en un momento determinado.

El sistema de gestión de incidencias se desarrolló bajo el patrón arquitectónico MVC, que es una propuesta de diseño de *software*, utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear *software* más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos. A continuación, se muestra cómo sería su funcionamiento en la solución que se propone, con la implementación del patrón M-V-C con Symfony. Ver figura 1:

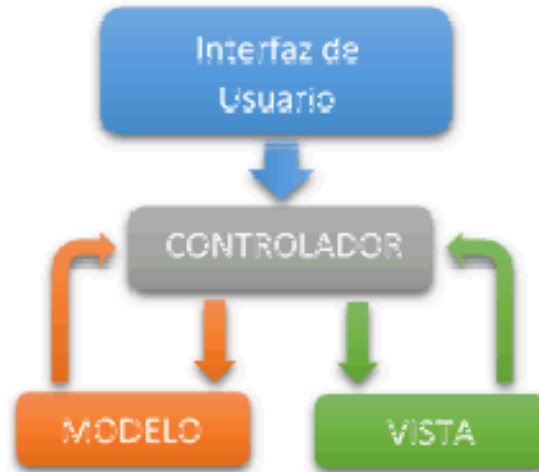


Figura 1. Funcionamiento del Patrón Modelo Vista Controlador.

El Diagrama de Clases del Diseño con estereotipos *web* (DCD), representa las clases que intervienen en el proceso de registro de incidencias, indicando las relaciones que tienen cada objeto instanciado, los atributos y métodos necesarios para su ejecución. A continuación, se muestra el DCD para el requisito “Actualizar estado de incidencia”. El resto de estos diagramas puede ser consultado junto con el resto de los artefactos ingenieriles en el expediente del trabajo.

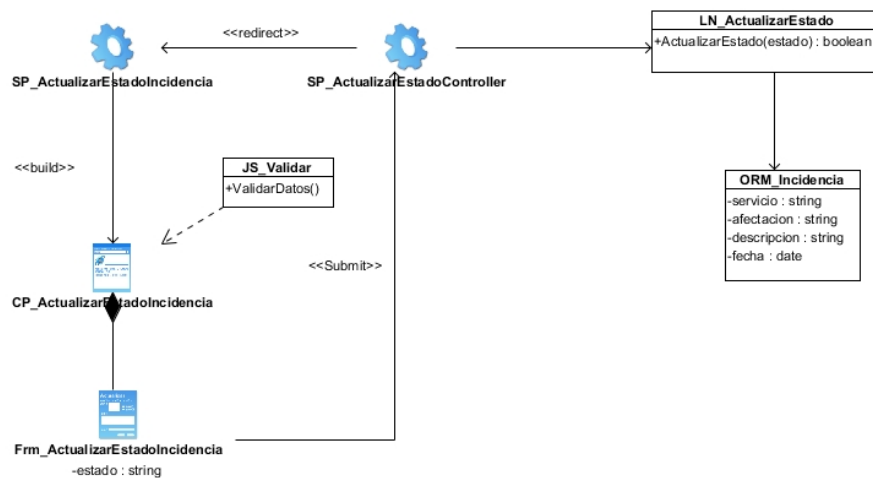


Figura 2. Diagrama de Clases del Diseño: actualizar estado de incidencia (Elaboración propia).

Patrones de diseño

Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de *software* o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación que resuelve un problema general de diseño en un contexto particular.

Patrones GRASP:

Los Patrones de Principios Generales para Asignar Responsabilidades (*GRASP* por sus siglas en inglés), describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones:

- **Creador:** asignarle a una clase la responsabilidad de crear una instancia de otra. Dentro del sistema, este patrón se evidencia en las acciones de los controladores (LN_ActualizarEstado), las cuales crean objetos del modelo o los formularios que representan las entidades.
- **Bajo Acoplamiento:** determina el nivel de dependencia de una clase con respecto a otras. Una clase con bajo acoplamiento no depende de muchas otras. Es muy útil para clases que cambian constantemente y para obtener un grado alto de reutilización. Con este patrón, no se afectan por cambios a otros componentes, ya que están por separado.
- **Controlador:** es el encargado de asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase que represente una de las siguientes opciones. Se evidencia el uso de este patrón (SP_ActualizarEstadoController), ya que para cada petición o evento que se genere en el mismo, existe un

controlador con la responsabilidad de obtenerla y devolver una respuesta. La respuesta puede ser mostrar una vista, ejecutar un método, devolver un mensaje y otros.

Patrones Gof

Los patrones *GoF* (*Gang of Four* o “Pandilla de los Cuatro” en español), describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación entre clases, la combinación clases y la formación de estructuras de mayor complejidad. Nos permiten crear grupos de objetos para ayudarnos a realizar tareas complejas:

- **Observador** (*Observer*): patrón de comportamiento. Se utiliza para mantener “informados” a objetos desacoplados entre sí y que presentan dependencia de un objeto, notificando a los primeros ante cualquier cambio que ocurra en el último. Este patrón es utilizado tradicionalmente en la capa de presentación en el diseño de componentes para las interfaces de usuario de las aplicaciones.
- **Decorador** (*Decorator*): patrón de tipo estructura. A nivel de objetos, añade responsabilidades adicionales a un objeto dinámicamente. Se utiliza este patrón para la vista y el *layout* o plantilla global que decora el contenido de la misma.

En el siguiente epígrafe se muestra el modelo de datos. Es una representación abstracta de los datos de una organización, las relaciones entre ellos, su significado y sus restricciones de consistencia con el objetivo de describir mejor una organización. Entre las distintas representaciones de modelos de datos, se selecciona el modelo de Entidad-Relación, el cual consiste en un gráfico, por el cual se diseña con la finalidad de dar a conocer las entidades que intervienen dentro de una base de datos, junto con sus respectivas relaciones. Constituye una herramienta de modelado de datos, que fundamentalmente permite representar de forma gráfica una base de datos o parte de la misma. Ver figura 3:

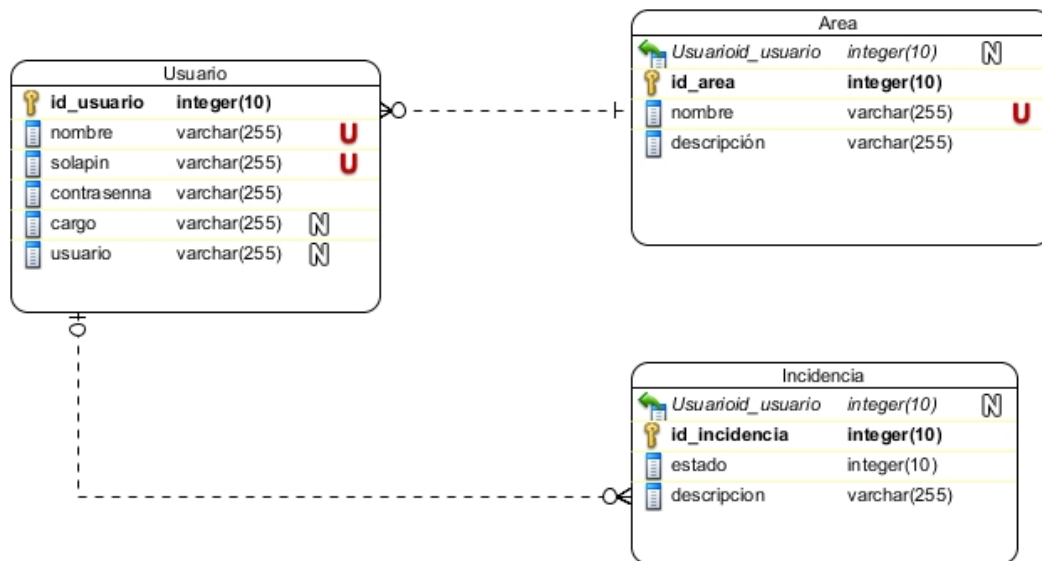


Figura 3. Modelo físico de la base de datos.

Conclusiones

Con la obtención de los requisitos funcionales y no funcionales y el diseño del sistema, se propicia el entendimiento entre el equipo de desarrollo y el cliente en función de lo que el sistema debe realizar y las características que debe poseer.

Se obtuvieron, además, los artefactos correspondientes al flujo de trabajo de diseño, el cual sirve como material de referencia para futuras ampliaciones y modificaciones del sistema de gestión de incidencias de la facultad CITEC, como guía para comenzar su implementación.

Referencias

- DOÑA, J. *Modelado de los procesos de toma de decisión en entornos sociales mediante operadores de agregación OWA*, Universidad de Málaga, 2008. [Disponible en: <http://www.lcc.uma.es/~jmdona/Borrador%20TESIS%20040308.pdf>]
- GARCÍA, L. and J. MONTERO *Uso de Sistemas de Gestión de Contenidos de Aprendizaje para el desarrollo del Trabajo Independiente* *Referencia Pedagógica*, 2013, Vol.2(No.2): 152-165.
- GÓMEZ, D.; B. BLANCO, *et al.* *El Sistema de Control Interno para el Perfeccionamiento de la Gestión Empresarial en Cuba (Internal Control System for the Improvement of Corporate Governance in Cuba)* *GECONTEC: Revista Internacional de Gestión del Conocimiento y la Tecnología*, 2013, Vol.1(No.2).

- HERNÁNDEZ GONZÁLEZ, B.; T. RAMÍREZ RAMÍREZ, *et al.* Sistema para la auditoría y control de los activos fijos tangibles *Revista Universidad y Sociedad*, 2019, 11(1): 128-134.
- MAR CORNELIO, O.; J. VERDE ACOSTA, *et al.* Sistema para la reservación de tiempo de máquina en los laboratorios de la Universidad de las Ciencias Informáticas *Revista Cubana de Ciencias Informáticas*, 2013, 7(4): 124-133.
- MAR, O.; J. GULÍN, *et al.* Sistema de Laboratorios a Distancia para la práctica de Control Automático *Revista Cubana de Ciencias Informáticas*, 2016, 10(4): 171-183.
- MAR, O.; Y. Z. VÉLIZ, *et al.* Motor de inferencia decisional en sistema informático para la evaluación del desempeño *Revista Cubana de Ciencias Informáticas*, 2015, 9(4): 16-29.
- MARTÍN-PENA, D.; M. PAREJO-CUELLAR, *et al.* Las Tecnologías de la Información y la Comunicación en las radios universitarias españolas en el periodo 2012-2016 *Transinformação*, 2018, 30: 27-38.
- ORTIGUEIRA, L. and D. GÓMEZ Creación de habilidades y competencias a través del empleo de las nuevas tecnologías para el apoyo al proceso de aprendizaje *GECONTEC: Revista Internacional de Gestión del Conocimiento y la Tecnología*, 2016, 4(1).
- PEDRO, M. J. G.-S. El concepto de competencias y su adopción en el contexto universitario *REVISTA ALTERNATIVAS. CUADERNOS DE TRABAJO SOCIAL*, 2009, (No.16).
- SALAS, M. and C. CERÓN Sistema Web para Evaluar las Competencias mediante Pruebas Objetivas en Educación Superior *Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 2014, (12).
- ZAMBRANO, A.; T. GUARDA, *et al.* Técnicas de mitigación para principales vulnerabilidades de seguridad en aplicaciones web *Revista Ibérica de Sistemas e Tecnologias de Informação*, 2019, (E17): 299-308.