

Universidad de las Ciencias Informáticas

Facultad 6



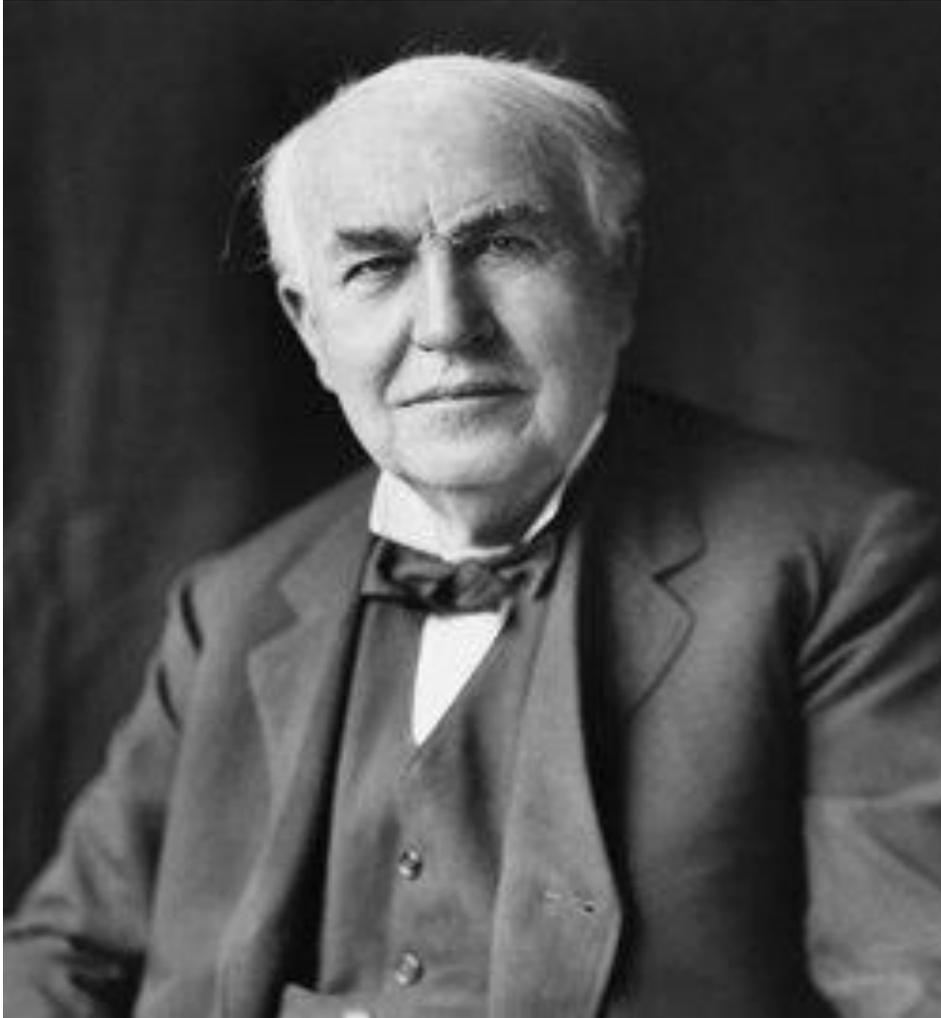
Título: Módulo de gestión de la información del departamento de docencia y capacitación del Centro Nacional de Electromedicina

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autor: Roberto Cabello Landeaux

Tutores: Ing. Denys J. Hernández Peña
MsC. Dayany Díaz Corona

La Habana, julio 2016



“Nuestra gran debilidad es siempre la renuncia. El único camino cierto para lograrlo es volver a intentarlo una vez más...”

Thomas Alva Edison

DECLARACIÓN DE AUTORÍA

Declaro ser autor del presente trabajo de diploma que tiene por título: Módulo de gestión de la información del departamento de docencia y capacitación del Centro Nacional de Electromedicina y se reconoce a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Roberto Cabello Landeaux

Firma del Autor

Ing. Denys J. Hernández Peña

Tutor

MsC. Dayany Díaz Corona

Tutora

DATOS DE CONTACTO

Ing. Denys Hernández Peña: Graduado del curso 2004-2005. CUJAE. Profesor Asistente. Profesor adjunto de la universidad medica de la habana. Jefe del departamento de informática del centro nacional de electromedicina.

Correo Electrónico: dhernandezp@cne.sld.cu

MsC.Ing. Dayany Díaz Corona: Graduado de Ingeniería en Ciencias Informáticas en el 2007 en la Universidad de la Ciencia Informáticas, profesora asistente y master en Ingeniería de Software e Inteligencia Artificial por la Universidad de Málaga, España. Actualmente se desempeña como jefa del Departamento de Práctica Profesional del Centro de Informática Industrial (CEDIN) de la Universidad de las Ciencias Informáticas.

Correo Electrónico: ddiazc@uci.cu

AGRADECIMIENTOS

A todas las personas que me han alentado y acompañado a lo largo de estos años, quizás sus nombres no estén hoy aquí, pero si están en mi mente y en mi corazón.

A la Universidad de las Ciencias Informáticas y a la Revolución Cubana por darme la oportunidad gratuita e incondicional de prepararme como ingeniero durante todo el transcurso de la carrera.

A mis tutores Denys y su esposa Dayany por darme todo el apoyo que necesitaba, por saber guiarme, alentarme y darme las herramientas para encontrar la solución a mis problemas, por ser pacientes y amigos, gracias.

A todos los profesores del C.P.E, en especial a nuestra carismática profesora de cálculo Yanelis.

A mi tía Elia con su carácter fuerte siempre empujando para que salga adelante.

A mi hermano Rey que ha estado siempre presente en todas las etapas de mi vida.

A mis amigos Jorgito, Osvaldito, Iosmany y su mujer Haydee que dedicaron muchas hora, noches y madrugadas a ayudarme con mi trabajo.

DEDICATORIA

Quiero dedicar este Trabajo de Diploma a mi madre Milagro L.H. por haberme dado la vida, por guiar mis primeros pasos, por estar siempre a mi lado, gracias a tus consejos mamá hoy he logrado parte de mis sueños, gracias por todo el sacrificio que tuviste que hacer para que hoy tu hijo pueda estar donde está, gracias por ser madre y padre a la vez, por ser la base de mi existencia, por no darte por vencida y por servirme siempre de ejemplo. Sin la ayuda de nadie logramos grandes cosas en nuestras vidas. Aunque no estés presente te llevo siempre conmigo en mi mente y mi corazón. Te quiero mami, este logro es por ti.

RESUMEN

El presente trabajo tiene como objetivo informatizar la actividad de docencia y Capacitación en el Centro nacional de Electromedicina, esta es realizada por los trabajadores del sector con el propósito de perfeccionar su accionar científico y social dentro del Ministerio de Salud Pública de Cuba.

El diseño del software que se ha creado para la propuesta, proyecta como cualidades: la flexibilidad en su uso, la claridad y facilidad para su navegación, la asequibilidad de su comprensión, así como la apreciación visual de sus entornos. La aplicación se desarrolla sobre tecnologías web, corriendo sobre un servidor LAMP (Linux, Apache, MySQL, PHP), utilizando herramientas de software libre.

A la aplicación, se accede desde las instituciones clínico-docentes del país. Se concibió un sistema que tiene como objetivo lograr que toda la información referente a esta actividad se almacene en una base de datos central. La misma podrá ser consultada desde cualquier centro docente. Además, permite, gestionar la información docente para la actividad posgraduada en el sistema de salud cubano, específicamente en las áreas de Capacitación y Superación Profesional.

Palabras Claves: Capacitación, Lamp, Superación Profesional.

SUMMARY

This work aims to computerize the activity of teaching and training at the National Center Electromedicina, this is done by workers in the sector in order to improve its scientific and social actions within the Ministry of Public Health of Cuba.

Software design that has been created for the proposal, projected as qualities: flexibility in use, clarity and ease of browsing, the affordability of their understanding and appreciation of their visual environments. The application is developed on web technology, running on a LAMP (Linux, Apache, MySQL, PHP) server, using free software tools.

The application is accessed from the country-clinical teaching institutions. A system that aims to ensure all information related to this activity is stored in a central database. It can be accessed from any educational institution. It also allows to manage information for teaching postgraduate activity in the Cuban health system, specifically in the areas of training and professional improvement.

Keywords: Training, Lamp, Professional Advancement.

ÍNDICE

INTRODUCCIÓN	2
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
INTRODUCCIÓN	5
1.1 CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA.....	5
Capacitación.....	5
Docencia.....	5
1.2 ÁMBITO INTERNACIONAL	6
Altamira Employees.....	6
Dataprocess Soluciones Registradas (DSR)	6
1.3 ÁMBITO NACIONAL	7
Akademos.....	7
Sistema de Gestión Universitaria.....	7
1.4 ESTUDIO DE LAS HERRAMIENTAS Y TECNOLOGÍAS A UTILIZAR EN EL DESARROLLO DE LA SOLUCIÓN.....	8
Metodología y Lenguaje de Modelado	9
Proceso Unificado Ágil (AUP)	9
Herramienta CASE	10
Lenguaje Unificado de Modelado 2.1	11
Entorno integrado de desarrollo (IDE).....	11
Netbeans 7	11
Framenworks.....	12
ExtJs 2.2	12
Plataforma de desarrollo LAMP	13
Sistema Operativo	13
CONCLUSIONES PARCIALES	14
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	15
INTRODUCCIÓN	15
2.1 DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN	15

Modelo del dominio.....	15
2.2 REQUISITOS FUNCIONALES.....	16
2.3 REQUISITOS NO FUNCIONALES.....	18
CONCLUSIONES PARCIALES	21
CAPÍTULO 3. DESARROLLO DE LA SOLUCIÓN PROPUESTA.....	22
INTRODUCCIÓN.....	22
3.1 ARQUITECTURA DEL SISTEMA.....	22
3.3 DIAGRAMA DE CLASES	27
3.4 MODELO DE DATOS.....	29
3.5 DIAGRAMAS DE COMPONENTES.....	30
3.6 DIAGRAMA DE DESPLIEGUE	32
Pruebas de software	33
Pruebas de aceptación	33
Diseño de casos de prueba	33
CONCLUSIONES PARCIALES	35
CONCLUSIONES GENERALES.....	37
RECOMENDACIONES.....	38
BIBLIOGRAFÍA	39
ANEXOS	41
GLOSARIO DE TÉRMINOS.....	61

ÍNDICE DE FIGURAS

<i>Figura 1. Modelo de dominio</i>	16
<i>Figura 2. Arquitectura del SIGICEM</i>	23
<i>Figura 3. Arquitectura del módulo</i>	24
<i>Figura 4. Comportamiento del patrón Experto en el modelado del diseño</i>	25
<i>Figura 5. Comportamiento del patrón Alta Cohesión en el modelado del diseño</i>	26
<i>Figura 6. Comportamiento del patrón Bajo Acoplamiento en el modelado del diseño</i>	26
<i>Figura 7. Comportamiento del patrón Controlador en el modelado del diseño</i>	27
<i>Figura 8. Comportamiento del patrón Decorador en el modelado del diseño</i>	27
<i>Figura 9. Diagrama de clases insertar curso</i>	28
<i>Figura 10. Diagrama de clases visualizar edición.</i>	29
<i>Figura 11. Modelo de datos</i>	30
<i>Figura 12. Diagrama de componentes</i>	31
<i>Figura 13. Diagrama de despliegue</i>	32
<i>Figura 14. Incidencias detectadas en el sistema</i>	35

ÍNDICE DE TABLAS

<i>Tabla 1. Comparación entre herramientas y tecnología.....</i>	<i>8</i>
<i>Tabla 2. HU Insertar especialista</i>	<i>20</i>
<i>Tabla 3. Caso de prueba Insertar edición.....</i>	<i>34</i>
<i>Tabla 4. Resultado de las iteraciones.....</i>	<i>34</i>

INTRODUCCIÓN

La revolución cubana, ha dedicado una parte importante de sus recursos a la informatización de diferentes esferas de la sociedad, siendo una de ellas el sector de la salud, controlado a la vez por el Ministerio de Salud Pública (MINSAP).

Debido a la participación activa del Ministerio de Informática y las Comunicaciones (MINCOM) ha existido un incremento vertiginoso de la tecnología médica, surgiendo la necesidad de crear el Centro Nacional de Electromedicina (CNE), orientado a gestionar y controlar los equipos de todas las unidades de salud del país. En la actualidad el centro cuenta con un sistema automatizado conocido como Sistema de Gestión para la Ingeniería Clínica y Electromedicina (SIGICEM), Este es un proyecto desarrollado y distribuido por (CNE) de conjunto con ingenieros de la Universidad de las Ciencias Informáticas (UCI). Que pretende cubrir el proceso de gestión tecnológica llevado a cabo por los especialistas del CNE institución que tiene como objeto social garantizar la gestión tecnológica de los equipos médicos instalados en todas unidades de salud del país. Lo que contribuye a la fiabilidad y actualización de la información referente al equipamiento médico instalado en el país, y a elevar la calidad y eficiencia de los servicios técnicos dentro del CNE mediante la organización, planificación, control y análisis de las acciones que se ejecuten.

Dentro del proceso a cubrir por el sistema se encuentra la organización, ejecución y control de la capacitación técnica de los especialistas, así como su participación en cursos internacionales y misiones técnicas en el exterior del país. En este sentido la versión actual del sistema comprende un grupo de funcionalidades básicas que garantizan la gestión de información complementaria de los especialistas de cada uno de los centros de atención tecnológica en función de la especialidad técnica en la cual se desempeñan los mismos dentro del Sistema Nacional de Electromedicina.

Sin embargo, en el área de docencia y capacitación dicho sistema no abarca totalmente lo concerniente al proceso de capacitación, participación en entrenamiento y misiones técnicas, que realizan los especialistas, trayendo consigo que se dificulte la toma de decisiones, ya que no es capaz de guardar la información relacionada con la planificación de los cursos, ni con las ediciones de cursos que imparten o reciben los mismos, tanto dentro como fuera de la institución, como entrenamiento para la atención de una tecnología específica. Tampoco se gestionan los datos de contacto de los profesores y participantes externos. Esto trae como consecuencia que la información de capacitación no se maneje correctamente, ya que la misma puede extraviarse, dañarse, y afectar la planificación de los especialistas y de los recursos asociados, llegando incluso a cometer pérdidas en la economía del país.

Como consecuencia de la situación planteada anteriormente se define como **problema a resolver**: ¿Cómo contribuir a la gestión de la información del departamento de docencia y capacitación en el Centro Nacional de Electromedicina para controlar las actividades de formación y superación de los especialistas?, definiéndose como **objeto de estudio**: Los sistemas para la gestión de la información de docencia y capacitación, enmarcado en el **campo de acción**: El sistemas para la gestión de la información de docencia y capacitación en el CNE.

Para dar solución al problema planteado se trazó como **objetivo general**: Desarrollar el módulo de gestión de la información del departamento de docencia y capacitación del Centro Nacional de Electromedicina para el SIGICEM.

Idea a defender

Con el desarrollo de un módulo que se integre al SIGICEM, se contribuirá a la gestión de la información del departamento de docencia y capacitación en el Centro Nacional de Electromedicina.

Para dar cumplimiento al objetivo planteado en la investigación se han definido las siguientes tareas de investigación:

1. Estudio del arte referente al proceso de gestión de información de capacitación y colaboración del personal electro médico a nivel nacional e internacional.
2. Definición de los procesos dentro SIGICEM con los cuales interactuarán los nuevos procesos de gestión de información de capacitación.
3. Identificación en el sistema los eventos que van a estar asociados a los procesos de gestión de información de capacitación.
4. Evaluación y aplicación de la arquitectura definida para el desarrollo del SIGICEM.
5. Realización del diseño de los artefactos del flujo de trabajo, Modelamiento del Negocio, Requisitos, Análisis, Diseño e Implementación para las funcionalidades definidas en el: Módulo para la gestión del departamento de docencia y capacitación del SIGICEM.
6. Realización del modelo de datos existentes y descripción de las nuevas entidades.
7. Realización de la implementación de las funcionalidades identificadas.

Una vez desarrollada la aplicación se espera obtener los siguientes resultados:

1. Entrega de la versión funcional módulo de la gestión información complementaria de los especialistas para el SIGICEM.
2. Integración de las funcionalidades implementadas en el SIGICEM.
3. Entrega de toda la documentación de acuerdo con los lineamientos mínimos de calidad establecidos por la Universidad y los artefactos generados a partir de la metodología empleada.

Para la realización del documento se dividió la información de la siguiente forma:

Capítulo 1: Fundamentación Teórica. En este capítulo se realiza un estudio de los sistemas existentes en el mundo que abordan el tema de docencia y capacitación. Se describen brevemente los conceptos fundamentales relacionados con el objeto de estudio, la metodología, herramientas y las tecnologías a utilizar para el desarrollo de la aplicación.

Capítulo 2: Características del sistema. En este capítulo se ofrece una visión práctica del sistema, la cual incluye los requisitos funcionales y no funcionales, así como las historias de usuario, el modelo de dominio y sus descripciones.

Capítulo 3: Diseño e implementación del sistema. Se presenta una vista interna del sistema, el patrón de arquitectura, diagramas de componentes y diagrama de despliegue, diagrama de clases del diseño, patrones de diseño. Se especifican las pruebas realizadas al sistema, incluyendo los resultados obtenidos al ejecutar las mismas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

INTRODUCCIÓN

En el presente capítulo se plantean los conceptos fundamentales relacionados con el objeto de estudio del trabajo, que servirán para un mejor entendimiento de la solución propuesta. Se realizará un estudio de las herramientas encontrada en la bibliografía consultada relacionadas con el objeto de estudio.

Por último, se realiza una caracterización de las tecnologías, herramientas y lenguajes para definir y seleccionar las que se utilizarán en el desarrollo de este trabajo de diploma.

1.1 CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA

Capacitación

La Norma Cubana 3000: 2007 define la capacitación como: “Conjunto de acciones de preparación, continuas y planificadas, concebido como una inversión, que desarrollan las organizaciones dirigidas a mejorar las competencias y calificaciones de los trabajadores, para cumplir con calidad las funciones del cargo, asegurar su desempeño exitoso y alcanzar los máximos resultados productivos o de servicios” [1].

La capacitación es una herramienta substancial para el crecimiento profesional en un mundo competitivo y exigente. Permite acceder a mejores ubicaciones en la empresa, incrementar conocimientos, aplicar nuevas destrezas en la producción laboral, entre otras ventajas. Es un proceso de desenvolvimiento de las aptitudes que perfecciona el desempeño del trabajador y la idoneidad de la organización [2].

Docencia

Como se plantea en la ley (Ley de Educación Nacional, artículo 71) [3]. La docencia es una profesión cuya especificidad se centra en la enseñanza, entendida como acción intencional y socialmente mediada para la transmisión de la cultura y el conocimiento en las escuelas, como uno de los contextos privilegiados para dicha transmisión, y para el desarrollo de potencialidades y capacidades de los alumnos. Los Institutos Superiores de Formación Docente son las instituciones formadoras.

La formación docente es un proceso permanente, que acompaña todo el desarrollo de la vida profesional. La formación docente inicial, tiene la finalidad de preparar profesionales capaces de enseñar, generar y transmitir los conocimientos y valores necesarios para la formación integral de las personas, el desarrollo nacional y la construcción de una sociedad más justa y promoverá la construcción de una identidad docente basada en la autonomía profesional, el vínculo con las culturas y las sociedades

contemporáneas, el trabajo en equipo, el compromiso con la igualdad y la confianza en las posibilidades de aprendizaje de sus alumnos.

1.2 ÁMBITO INTERNACIONAL

Altamira Employees

Altamira Employees es un software de personal altamente personalizable desarrollado para la administración de recursos humanos, dando la flexibilidad de diseñar la plataforma HRM2, según los requerimientos de cada organización.

Toda la información importante sobre un empleado del personal (datos personales, posiciones, beneficios, retribuciones, etc.) puede ser rastreada históricamente a través de una base de datos online que permite rastrear el historial de toda la información del empleado y así se pueden conocer las diferentes versiones realizadas a dicho empleado [4].

Este sistema es una potente solución al problema de la gestión de la información del personal, pero presenta algunas violaciones de las reglas del negocio dentro del SIGICEM, porque los empleados de Altamira Employees pueden acceder a la plataforma a través de sus cuentas particulares, visualizar y actualizar sus datos.

Para el desarrollo de la investigación estos servicios son perjudiciales porque el personal no debe ser capaz de actualizar sus datos personales, ya que la información versionada debe ser única en el sistema y solamente actualizada por el personal correspondiente.

2HRM (Human Resource Management, o sus siglas en español Gestión de Recursos Humanos).

Dataproces Soluciones Registradas (DSR)

Es un software desarrollado por Dataproces Soluciones Registradas (DSR) que proporciona medios fáciles y flexibles para organizar y controlar los datos del empleado.

En su base de datos la información Almacenada en cada una de las fichas de empleado contiene, tanto los datos personales del empleado como varios códigos de estado administrativos. Entre estos códigos existe uno para contratos de empleo, que puede utilizar para asignar niveles o contratos de empleo estandarizados.

La información confidencial de un empleado se controla a través de una tabla asociada a la ficha del empleado. Esto le permite dar un acceso relativamente amplio a la información básica del archivo del

empleado y poner unas restricciones de seguridad más rigurosas en los datos confidenciales [5]. Este sistema soluciona el problema de la gestión de la información personal, dividiendo la misma en confidencial y básica, pero tiene la contradicción que es un software propietario, y para adquirirlo es necesario pagar elevados costos.

Además, es una solución general para el sector empresarial que no puede ser adaptada a las características del SIGICEM con la facilidad y eficiencia de un software desarrollado específicamente para el mismo.

1.3 ÁMBITO NACIONAL

Akademios

Akademios es un sistema Web desarrollado en la Universidad de las Ciencias Informáticas (UCI). Brinda una interfaz común para todos sus usuarios. Además, ofrece la información actualizada en tiempo real de las facultades de la universidad, es una herramienta que mantiene informado a los estudiantes sobre su desempeño académico, además de llevar el control del desarrollo del proceso docente, notas y asistencia. En este sitio tanto profesores como estudiantes se mantienen informados sobre la vida docente del estudiantado de la universidad [6].

Sistema de Gestión Universitaria

El Sistema de Gestión Universitaria es una herramienta para la gestión de los procesos de formación docente de la Universidad de las Ciencias Informáticas, las funcionalidades que incluye permiten realizar las siguientes tareas:

- ✓ Gestión de estudiantes y estructuras.
- ✓ Control y desarrollo del proceso docente, lo que incluye asistencia y evaluaciones
- ✓ Registro de las asignaturas por cada disciplina dentro de una carrera.
- ✓ Sistema de apoyo a los procesos del área de Formación Docente del Centro ISEC.
- ✓ Planificación de profesores y grupos por cada asignatura.

URL: <https://gestionuniversitaria.uci.cu>

En este apartado se analizaron diferentes herramientas que son utilizadas con el objetivo de llevar a cabo la gestión de docencia y capacitación. Para la descripción de las mismas se tomaron en cuenta los siguientes parámetros.

En la tabla 1. se muestra un resumen de las características de las herramientas analizadas.

Tabla 1. Comparación entre herramientas y tecnología

	Base de Datos	Servidor WEB	Marco de Trabajo o Framenworks	Lenguaje de Programación
Sistema de Gestión Universitaria	Postgres SQL 8.4.1	Apache 2.2	CodeIgniter 1.7.3	Hypertext Preprocessor PHP 5.2
AKADEMOS	Postgres SQL	Apache 2.2	ExtJS Zend	PHP 5.0
SIGICEM	MySQL 5.1	Apache 5.1	Synfony 1.4	PHP 2.2

En la investigación sobre las herramientas de gestión de la información en el área de docencia y capacitación en el ámbito internacional y nacional, se ha llegado a la conclusión que el uso de las mismas no es el más idóneo para la aplicación. Debido a que la información que se necesita registrar tiene particularidades (los Framenworks de desarrollo Zend y CodeIgniter, tienen una arquitectura diferente al de Symfony 1.4, igualmente sucede con la Base de Datos Postgres y MySQL) que no se adaptan a las previstas por SIGICEM. Otra cuestión fundamental, es que ya posee una arquitectura definida que no coincide con la precisada en los sistemas antes mencionados, las mismas crean dependencias innecesarias e incompatibilidad en los servidores desplegados y no se cuenta en las áreas de salud con los elementos esenciales para el mínimo funcionamiento de las tareas a cumplir.

1.4 ESTUDIO DE LAS HERRAMIENTAS Y TECNOLOGÍAS A UTILIZAR EN EL DESARROLLO DE LA SOLUCIÓN

Para el desarrollo del módulo de gestión de la información del departamento de docencia y capacitación del Centro Nacional de Electromedicina para SIGICEM, se hace necesaria la utilización de las tecnologías, metodología y herramientas que han sido definidas por la dirección del proyecto para lograr la estandarización con todos los módulos del sistema.

Metodología y Lenguaje de Modelado

Los estándares o metodologías definen un conjunto de procedimientos, herramientas, técnicas y soporte documental que guían la forma en que se aplica la ingeniería del software. Si a lo largo del camino no se sigue metodología alguna no se sabrá con certeza si lo que se está desarrollando conduce a donde realmente se desea llegar [7].

Proceso Unificado Ágil (AUP) no tiene referencia

AUP (Agile Unified Process), es una versión simplificada del Proceso Unificado de Rational (RUP), donde se describe de manera fácil la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos válidos en RUP [8]. Entre las técnicas que utiliza AUP se encuentran el Desarrollo Dirigido por Pruebas (TDD), Modelado Ágil, Gestión de Cambios Ágil y Refactorización de Base de Datos para mejorar la productividad.

El Modelo establecido por AUP es más simple que el que aparece en RUP, pues reúne en una las disciplinas de Modelado de Negocio, Requisitos y Análisis y Diseño. El resto de las disciplinas (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión y Entorno) coinciden con RUP. El ciclo de vida de AUP al igual que de RUP consta de cuatro fases fundamentales:

- **Concepción:** se establece la comunicación entre el cliente y el equipo de desarrollo para que ambos lleguen a comprender el alcance del nuevo sistema.
- **Elaboración:** el equipo de desarrollo profundiza en la comprensión de los requisitos del sistema y en validar la arquitectura.
- **Construcción:** se desarrolla el sistema y se prueba por completo en el ambiente de desarrollo.
- **Transición:** se lleva el sistema a los entornos de preproducción donde se somete a pruebas de validación, aceptación y finalmente se despliega en los sistemas de producción [9].

AUP es la adopción de muchas de las técnicas ágiles de XP y otros procesos ágiles de RUP, incorporando UML como Lenguaje Unificado de Modelado e incluyendo artefactos como de la definición de los requisitos y la estimación de recursos en la fase de Concepción. Otros de los artefactos generados en la fase de Elaboración, mediante el uso de esta metodología, son el Modelo del negocio o de procesos, la descripción de la arquitectura y el modelo funcional de alto nivel. Estos artefactos se generan luego de haber realizado el análisis del dominio y de los riesgos del problema, además de la definición de la

arquitectura básica y la planificación del proyecto. De esta forma quedan documentados los elementos más necesarios para una posterior mejora del sistema por un equipo de trabajo diferente.

Debido a la necesidad de utilizar una metodología no tan resumida como XP, pero tampoco tan extensa como RUP, se decidió utilizar la metodología AUP, dado el hecho de que XP no muestra de forma explícita cómo crear algunas de las herramientas que la administración necesita y RUP es el gestor que más cantidad de artefactos genera. AUP permitirá al desarrollador de la aplicación obtener un sistema funcional en un corto o mediano plazo y a su vez generar la documentación mínima necesaria para continuar con el segundo ciclo de desarrollo.

Herramienta CASE

Ingeniería de Software Asistida por Ordenador, es “una combinación de herramientas de software y metodologías de desarrollo”, las cuales ayudan a los ingenieros de software y analistas durante una o más fases del desarrollo de software. Aceleran el desarrollo de los sistemas y aumentan la calidad del software.

Visual Paradigma 6.4

Visual Paradigma es una herramienta CASE que ha sido concebida para el diseño de los artefactos obtenidos durante el ciclo de vida completo del proceso de desarrollo del software, a través de la representación de todo tipo de diagramas. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación [10]. Es una herramienta muy potente, gratuita, fácil de instalar, utilizar y actualizar. Permite dibujar todo tipo de diagramas UML, revertir código fuente a modelos UML, generar código fuente desde los diagramas UML. Incluye los objetos más recientes de UML además de diagramas de casos de uso, diagramas de clase, diagramas de componentes, reversa instantánea para Java, C++, DotNet Exe/DLL, XML, XML Schema, y Corba IDL, ofrece soporte para Rational Rose, integración con Microsoft Visio, además de generar reportes y documentación en HTML/PDF [11].

La versión Visual Paradigm UML 6.4, fue diseñada para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque Orientado a Objetos. Permitiendo mejoras de disponibilidad, diseño centrado en casos de uso y enfocado al negocio

que generan un software de mayor calidad y el uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.

Lenguaje Unificado de Modelado 2.1

Unified Modeling Language o UML, por sus siglas en inglés, fue desarrollado por Grady Booch y Jim Rumbaugh en 1995, con el objetivo de crear un lenguaje para el modelado. Mediante la modelación se logran hacer más entendibles distintos puntos de la realidad que sean de interés, todo esto a partir de la complementación de varias técnicas de modelado en una sola. Integrado por un conjunto de diagramas, los cuales se encuentran agrupados en dos categorías fundamentales: diagramas de estructura y diagramas de comportamiento.

Este lenguaje de modelado permite crear un nivel de comprensión y entendimiento entre los analistas, desarrolladores o cualquier personal involucrado, haciendo más soluble la comunicación existente entre ellos; permitiendo visualizar, especificar, construir y documentar la información con la que se cuente [10].

Entorno integrado de desarrollo (IDE)

Un entorno integrado de Desarrollo o IDE (por sus siglas en inglés) es un entorno de programación que ha sido empaquetado como un programa de aplicación, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los mismos pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Los IDE ofrecen un marco de trabajo amigable para la mayoría de los lenguajes de programación, donde pueden funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto. Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación mediante los plugins que le añaden soporte de lenguajes adicionales.

Netbeans 7

Para realizar el desarrollo del módulo se utilizará como IDE la herramienta NetBeans en su versión 7, la cual permite realizar el trabajo bajo las últimas tendencias de PHP, integrando gran variedad de herramientas modernas para mejorar la extensibilidad con el apoyo de los principales marcos de PHP.

Cuenta con una enorme cantidad de extensiones y plugins que pueden incorporarse al sistema para facilitar las tareas de programación, también posee soporte para el framework de Symfony a utilizar en la aplicación, también cuenta con experiencia de las versiones anteriores en la legibilidad y completamiento de sentencias de código. Se destaca su intuitiva interfaz de usuario, su sistema de depuración de programas y las demás herramientas de ayuda, que facilitarán el desarrollo del módulo, la cual mediante su dinámica interacción permite al usuario mismo instruirse sobre las nuevas ventajas y funcionalidades de la herramienta [12].

Framenworks

El framework es una estructura de soporte definido para resolver los problemas básicos que presentan el software o las especificidades de los módulos. Trabaja utilizando una llamada automatización del código fuente, la cual exige al desarrollador la creación de un código simple, es el esqueleto sobre el cual varios objetos son integrados para facilitar una solución dada.

Symfony 1.4

Es un framework completo que permite la optimización del patrón Modelo Vista Controlador, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

Symfony puede ser completamente personalizado para cumplir con los requisitos de las empresas que disponen de sus propias políticas y reglas para la gestión de proyectos y la programación de aplicaciones. Por defecto incorpora varios entornos de desarrollo diferentes e incluye varias herramientas que permiten automatizar las tareas más comunes de la ingeniería del software [13].

ExtJs 2.2

Es una biblioteca de JavaScript del lado del cliente para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX, DHTML y DOM, dispone de un conjunto de componentes para incluir en las aplicaciones web, las cuales están capacitados para comunicarse con el servidor usando AJAX. También contiene numerosas funcionalidades que permiten añadir interactividad a las páginas que utiliza.

ExtJs también trae un rico paquete de datos que permite a los desarrolladores utilizar un controlador de modelo-vista (MVC) en la construcción de su aplicación. El MVC aprovecha características como Big Data Grids permiten un nuevo nivel de interactividad en las aplicaciones web [14].

Permite la existencia de un balance entre Cliente – Servidor, donde la carga de procesamiento se distribuye, permitiendo que el servidor, al tener menor carga, pueda manejar más clientes al mismo tiempo. También aporta una comunicación asíncrona, en este tipo de aplicación el motor de render puede comunicarse con el servidor sin necesidad de estar sujeta a un clic o una acción del usuario, dándole la libertad de cargar información sin que el cliente se dé cuenta [15].

Plataforma de desarrollo LAMP

LAMP está considerada como una de las mejores herramientas disponibles para que cualquier organización o individuo pueda emplear un servidor web versátil y potente. Aunque creados por separado, cada una de las tecnologías que lo forman disponen de una serie de características comunes (Linux, Apache, MySQL y uno de los siguientes lenguajes: Perl, Python o PHP).

Especialmente interesante es el hecho que estos cuatro productos pueden funcionar en una amplia gama de hardware, con requerimientos relativamente pequeños sin perder estabilidad. Esto ha convertido a LAMP en la alternativa más adecuada para pequeñas y medianas empresas [15, 16].

La plataforma está integrada por:

- Linux con su distribución Ubuntu 12.04 como Sistema Operativo.
- Apache como servidor web en su versión 2.2.
- MySQL como gestor de base de datos en su versión 5.1.
- PHP como lenguaje de programación en su versión 5.1.

Sistema Operativo

Fue seleccionado Ubuntu en su versión 10.04 para el desarrollo de la aplicación por ser el sistema operativo definido en la arquitectura del proyecto SIGICEM y del departamento SES. Ubuntu es una distribución Linux que ofrece un sistema operativo enfocado a ordenadores de escritorio, aunque también proporciona soporte para servidores. Basada en Debian GNU/Linux, concentra su objetivo en la libertad

de uso y los lanzamientos regulares (cada 6 meses)¹. Ubuntu está totalmente basado en los principios del Software Libre.

CONCLUSIONES PARCIALES

Luego de analizados los sistemas a nivel nacional e internacional que permiten la gestión de inventarios, se ratificó la necesidad de desarrollar un sistema que cumpla con las características del sistema precedente. Fueron definidas la metodología de desarrollo, tecnologías y herramientas, seleccionadas por el equipo de trabajo para este tipo de aplicaciones complementarias con el SIGICEM, que soportarán el sistema a desarrollar.

¹Canonical Ltd. *ubuntu-es*. [En línea] *ubuntu-es*, 4 de 6 de 2011. http://doc.ubuntu-es.org/Sobre_Ubuntu.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

INTRODUCCIÓN

En el presente capítulo se describen las principales características que poseerá el sistema a desarrollar, enumerando los Requisitos funcionales y no funcionales que debe presentar, lo que permite hacer una concepción general del mismo. Además, se abordará lo referente al negocio a estudiar mostrando en detalles los procesos que serán objeto de automatización, se obtiene el modelo del dominio, las historias de usuario y un glosario de términos para identificar los conceptos del modelo de dominio.

2.1 DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN

Modelo del dominio

Teniendo en cuenta el bajo nivel de estructuración que presenta el negocio que se está estudiando y que está altamente centrado en tecnologías informáticas, se propone un modelo del dominio, ya que de manera visual permite mostrar al usuario los principales conceptos que se manejan en el dominio del sistema en desarrollo. Esto ayuda a los usuarios, clientes y desarrolladores e interesados, a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema. Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

Para identificar todos los conceptos que se emplearán en el diagrama de modelo de dominio se utilizará un glosario de términos sobre los nombres de las entidades que lo conforman.

Especialista: Es un actor del negocio encargado de manipular la información que se gestiona a través del sistema según su rol.

Profesor interno o externo: Tanto el profesor interno (trabajador de la entidad), como el externo (no es trabajador de la entidad). Es el encargado de impartir las ediciones de los cursos previamente planificados a los alumnos y controlar el desempeño de los mismos durante el curso y sus evaluaciones.

Curso: Entidad donde se gestiona todos los datos relacionados con los cursos de capacitación que se planifican.

Edición: Entidad donde se gestiona todos los datos relacionados con cada una de las ediciones de los cursos que se realizan.

Periodicidad: Tiempo que transcurre entre las ediciones de los cursos que se imparten (diaria, semanal, quincenal, mensual).

Curso presencial: Es aquella modalidad donde el proceso de formación tiene lugar a partir de la presencia de los estudiantes y sus profesores en el mismo lugar.

Curso semipresencial: Es la combinación del aprendizaje con la ayuda de materiales online y del aprendizaje tradicional en un grupo con un profesor.

Curso a distancia: Es poca o nula la frecuencia con que se encuentran estudiantes y profesores para desarrollar el proceso docente.

El modelo del dominio se describe mediante diagramas UML, específicamente con un diagrama de clases conceptuales significativas en el dominio del problema.

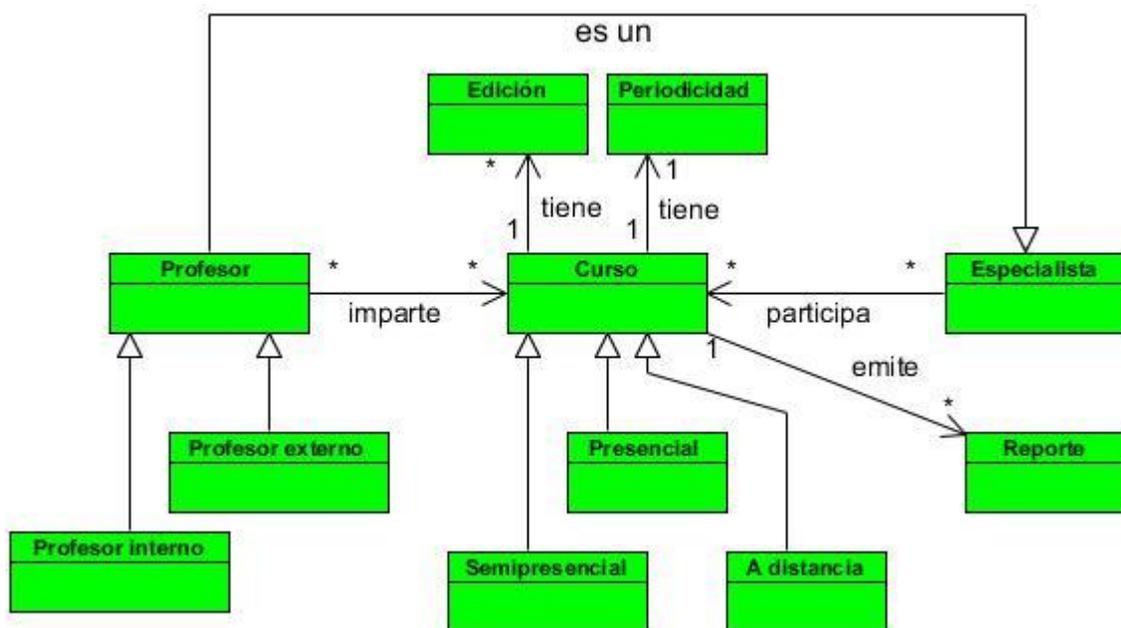


Figura 1. Modelo de dominio

2.2 REQUISITOS FUNCIONALES

Una vez conocidos los conceptos que rodean al objeto de estudio, se puede comenzar a analizar ¿Qué debe hacer el sistema para que se cumplan los objetivos planteados al inicio de este trabajo?, para ello se

enumera a través de requerimientos funcionales las funciones que el sistema deberá ser capaz de realizar. Dentro de ellos se incluyen las acciones que podrán ser ejecutadas por el usuario, las acciones ocultas que debe realizar el sistema, y las condiciones extremas a determinar por el sistema. De acuerdo con los objetivos planteados el sistema debe ser capaz de:

RF1 Insertar Especialista para ello debe de recoger la siguiente información: Graduado de, categoría salarial que esta va desde la escala I hasta la XIII, Grupo escala a la que pertenece (Administrativas, Común, Técnicos, Genéricos MINSAP, Operarios, Seguridad y Protección, Servicios, Dirigentes y Otras), cargo que ocupa e idiomas que domina.

RF2 Actualizar Especialista para ellos el sistema permitirá actualizar cualquier información registrada del especialista. Ya sea Graduado de, Categoría salarial que esta va desde la escala I hasta la XIII, Grupo escala a la que pertenece (Administrativas, Común Técnicos, Genéricos MINSAP, Operarios, Seguridad y Protección, Servicios, Dirigentes y Otras), Cargo que ocupa e idiomas que domina.

RF3 Eliminar Especialista el sistema debe de permitir eliminar un especialista.

RF4 Insertar Profesor se debe especificar si es un profesor externo o interno en caso de ser externo se debe de registrar los siguientes datos: (nombre y apellidos, sexo, email, teléfono, centro de estudio, fecha de nacimiento, fecha de graduación, graduado de, categoría docente, grado científico, editar, eliminar) en caso de ser interno se registra categoría docente y grado científico.

RF5 Eliminar Profesor el sistema debe de permitir eliminar un profesor.

RF6 Actualizar Profesor para ellos el sistema permitirá actualizar cualquier información registrada del profesor. Ya sea un profesor externo o interno en caso de ser externo se debe de actualizar los siguientes datos: (nombre y apellidos, sexo, email, teléfono, centro de estudio, fecha de nacimiento, fecha de graduación, graduado de, categoría docente, grado científico, editar, eliminar) en caso de ser interno se actualizará categoría docente y grado científico.

RF7 Insertar Curso se debe de especificar nombre del curso, Modalidad (presencial, semipresencial o a distancia), Periodicidad (diaria, semanal, quincenal, mensual).

RF8 Eliminar Curso el sistema debe de permitir eliminar un curso.

RF9 Actualizar Curso para ellos el sistema permitirá actualizar cualquier información registrada del curso. Ya sea (nombre del curso, Modalidad (presencial, semipresencial o a distancia), Periodicidad (diaria, semanal, quincenal, mensual).

RF10 Insertar Edición para ello debe de recoger la siguiente información:(título, profesor, fecha de inicio, fecha final, matricula, estado, cambiar de estado, editar, eliminar).

RF11 Actualizar Edición para ellos el sistema permitirá actualizar cualquier información registrada de la edición del curso.

RF12 Eliminar Edición el sistema debe de permitir eliminar la edición de un curso.

RF13 Mostrar reporte Edición de un curso

RF14 Mostrar reporte listado de participantes por curso.

RF15 Mostrar reporte de Profesores.

RF16 Mostrar reporte de Curso por especialidad.

RF17 Autenticar permite el acceso al sistema y la asignación de privilegios, según el rol.

RF18 Cerrara Sesión el sistema debe de permitir cerrar sesión de forma definida y automáticamente.

2.3 REQUISITOS NO FUNCIONALES

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable.

1. Usabilidad

RnF1_ Facilidad de uso por parte de los usuarios: el sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación al mismo.

RnF2_ Especificación de la terminología utilizada: el sistema debe adaptarse al lenguaje y términos utilizados por los clientes en la rama abordada con vista a una mayor comprensión por parte del cliente de la herramienta de trabajo.

RnF3_ Menús: la solución propuesta debe presentar una serie de menús tanto laterales como en forma de barra de iconos flotantes e internos que permitan el acceso rápido a la información por parte de los usuarios, aprovechando así las potencialidades de estas estructuras.

2. Seguridad

RnF4_ Políticas de seguridad por usuarios y roles: la solución propuesta debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del sistema, en dependencia del nivel de acceso que está basado fundamentalmente en roles.

RnF5_ Se garantiza el acceso controlado a la información. Esto influye en la forma en que se presentarán las interfaces a cada usuario en dependencia del nivel de acceso que le será conferido.

3. Portabilidad

RnF6_ El sistema será multiplataforma, podrá ser visualizado en sistemas operativos tales como versiones de Windows (XP o superiores) y Linux.

4. Confiabilidad

RnF7_ La información manejada y generada por el sistema estará protegida de accesos no autorizados y a su vez de ataques informáticos.

RnF8_ Los usuarios deben tener acceso (según sus permisos y roles otorgados) en todo momento a la información solicitada.

RnF9_ Al ocurrir un error el sistema muestra un mensaje indicando dónde se ha producido el mismo.

5. Disponibilidad

RnF10_ Se utiliza el servicio de autenticación para dar acceso a los usuarios que interactúan con el sistema. Solo los usuarios autorizados tendrán acceso a la información en todo momento.

6. Restricciones de Diseño

RnF11_ El sistema estará desarrollado en base a las políticas del software libre, que fueron ajustadas al sistema nacional de salud como: uso de servidores GNU/Linux.

RnF12_ Se utilizará como gestor de base de datos MySQL 5.1 y Symfony 1.4 como framework de desarrollo de aplicaciones web, el cual utiliza PHP como lenguaje de programación del lado del servidor.

RnF13_ Se utilizará Netbeans como IDE de desarrollo 7.0.

RnF14_ Para el diseño de las interfaces se utilizará Ext JS, librería JavaScript para el desarrollo de aplicaciones RIA (Rich Internet Application).

RnF15_ Se utilizará Visual Paradigm como herramienta para el modelado de los artefactos generados por la metodología AUP.

7. Soporte

RnF16_ El Sistema debe ser generado en tecnología web para ser accesible a través de la red nacional

RnF17_ Debe ser compatible como mínimo con los siguientes navegadores:

- Mozilla Firefox 4.0 o superior.
- Chrome 6.0 o superior

RnF18_ La aplicación Web contará con una ayuda donde el usuario podrá suplir las dudas que se le puedan presentar durante la utilización de la misma.

8. Mantenimiento

RnF19_ El sistema debe estar bien documentado de forma tal que el tiempo de mantenimiento sea mínimo en caso de necesitarse.

2.4 HISTORIAS DE USUARIO.

La Historia de Usuario es la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las HU es dinámico y flexible. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas. Estas deben proporcionar sólo el detalle suficiente como para poder hacer razonable la estimación de cuánto tiempo requiere su implementación. También se les asigna un número identificativo, una prioridad en el negocio (alta, media, baja) y la iteración en la que se implementará.

A continuación, se muestra una de las HU de prioridad media diseñada para el desarrollo del sistema:

Tabla 2. HU Insertar especialista

Historia de usuario	
Número: HU 1	Nombre del requisito: Insertar especialista
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días

Descripción:

Se visualiza una ventana donde se introducen los datos del especialista a registrar en el sistema para ello se debe de recoger la siguiente información (Graduado de, Categoría Salarial que esta va desde la escala I hasta la XIII, Grupo escala a la que pertenece (Administrativas, Común Técnicos, Genéricos MINSAP, Operarios, Seguridad y Protección, Servicios, Dirigentes y Otras), Cargo que ocupa e Idiomas que domina). El sistema válida la información registrada y se guarda de manera satisfactoria en la base de datos.

Observaciones:

El resto de las HU se pueden encontrar en los anexos de este documento (Anexos del 1 al 18).

CONCLUSIONES PARCIALES

Al finalizar el presente capítulo se arriba a las siguientes conclusiones:

- Después de realizar el análisis del sistema, quedaron definidas las Historias de Usuario, proporcionando una comprensión detallada de las funcionalidades de la aplicación.
- La obtención de los requisitos funcionales y no funcionales, permitió definir el comportamiento y restricciones del sistema para su implementación.

CAPÍTULO 3. DESARROLLO DE LA SOLUCIÓN PROPUESTA

INTRODUCCIÓN.

En este capítulo se muestran elementos relacionados con el diseño del sistema, tales como los diagramas de clases de diseño con extensiones web. Se añade al mismo tiempo, una descripción del diseño de la base de datos mediante el modelo de datos. También se abordarán brevemente los patrones de diseño y arquitectura que se utilizarán, los principios del diseño y finalizando se presentarán el modelo de implementación mediante el diagrama de despliegue que resultó del diseño realizado de cada una de las historias de usuario del sistema y los casos de pruebas realizados a las historias de usuario.

3.1 ARQUITECTURA DEL SISTEMA.

Para la estructuración del diseño se hace uso del patrón Modelo-Vista-Controlador que permite la escalabilidad y flexibilidad del sistema, además de ser el patrón definido por el equipo de desarrollo del SIGICEM. Este patrón de arquitectura de software, separa los datos de una aplicación, la interfaz de usuario y la lógica de control, en tres capas distintas [16].

- **El Modelo** representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- **La Vista** transforma el modelo en una página web que permite al usuario interactuar con ella.
- **El Controlador** se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

El MVC, al separar la presentación de la programación, consigue que la aplicación en desarrollo sea más escalable y pueda mantenerse mejor. Permite, que el resultado sea más claro y que la repartición de tareas dentro del equipo de trabajo sea fácil. Además, la depuración de la aplicación es más sencilla y, finalmente, puede utilizarse un marco de trabajo (o framework) bien testeado [17]. A continuación, se muestra la estructura arquitectónica del SIGICEM (Ver Figura 2).

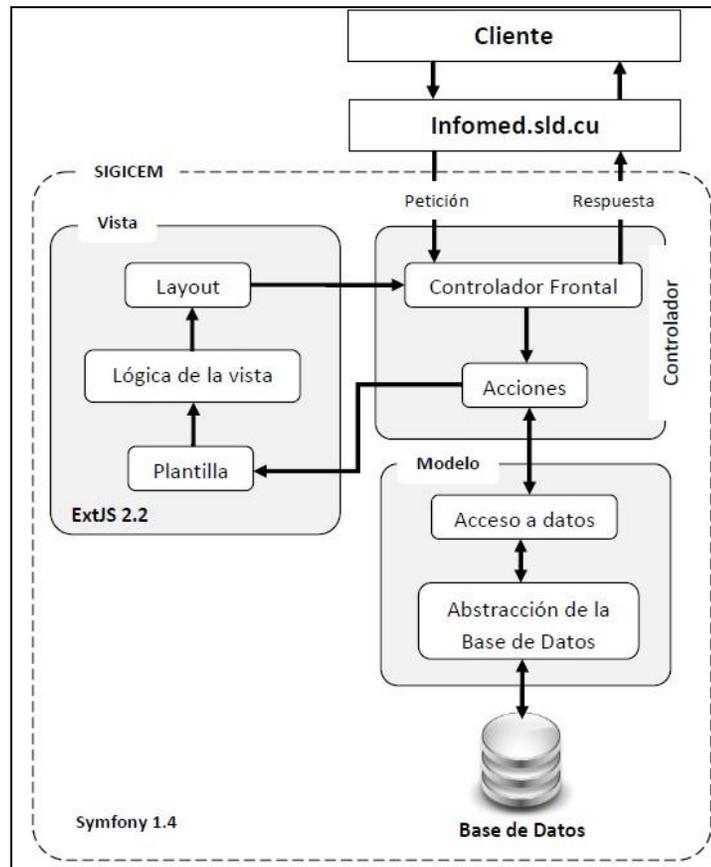


Figura 2. Arquitectura del SIGICEM

A continuación, presentamos un ejemplo de cómo se evidencia el patrón de arquitectura MVC en nuestra aplicación con la funcionalidad insertar edición (Ver Figura 3).

Patrón arquitectónico MVC

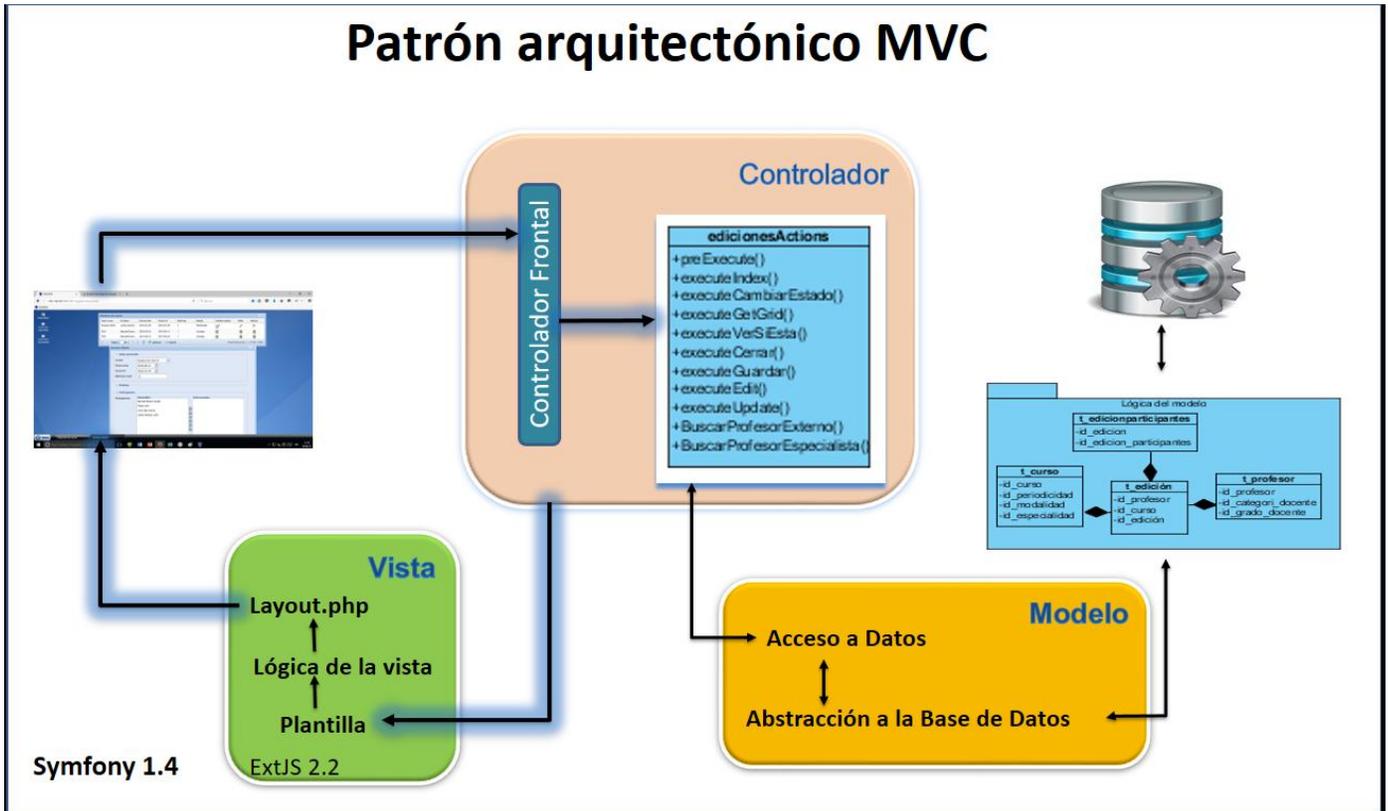


Figura 3. Arquitectura del módulo

Cuando el usuario decide insertar la edición de un curso, se muestra en pantalla un formulario, con los datos necesarios, llenado el formulario y pulsado el botón aceptar el sistema envía la petición al componente controlador donde es recibido por el controlador frontal de Symfony que re direcciona la información a la clase `edicionesActions` que tiene la responsabilidad de procesar todo lo concerniente a las ediciones, este proceso pasa al modelo donde se realiza el acceso a datos y la abstracción a la base de datos con las clases correspondientes. La respuesta es devuelta al controlador y posteriormente a la vista donde ExtJS, a través de la plantilla, la lógica de la vista y la plantilla global layout construye una nueva página con los resultados de la petición del usuario.

3.2 DISEÑO DEL SISTEMA.

Los patrones de diseño brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares” [18]. Su utilización hace posible identificar: clases, instancias, roles, colaboraciones y la distribución de responsabilidades, además de ayudar a construir clases y a estructurar sistemas de clases [19]. Para el desarrollo de la solución propuesta se utilizó el framework de desarrollo Symfony, el cual se encuentra concebido de manera tal que puede afirmarse que obliga al uso de varios patrones. A continuación, se describen y ejemplifican aquellos que han sido considerados a utilizar: GRASP: Patrones de Principios Generales para Asignar Responsabilidades

Describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones [20].

Experto: Al incluir Propel como ORM, este genera las clases para la gestión de las entidades con las responsabilidades debidamente asignadas. Esta es precisamente la solución que propone el patrón Experto, pues cada una de estas clases cuenta con un conjunto de funcionalidades relacionadas directamente con la entidad que representan.

```

TEdicionPeer.php
-DATABASE_NAME = 'propel'
-OM_CLASS = 'TEdicionTProfesor'
-ID_EDICION = 't_edicion_t_profesor.ID_EDICIO...
-ID_PROFESOR = 't_edicion_t_profesor.ID_PR...

-getMixerPreSelectHook(method)
+getUniqueColumnNames()
+getOMClass(withPrefix = true) : string
+getTableMap() : TableMap
+doSelectJoinAll(criteria : Criteria, con = null, joi...
+doSelectJoinTProfesor(criteria : Criteria, con = ...
+getInstanceFromPool(key)
+getFieldNames(type = BasePeer::TYPE_PHPN...
+doSelectOne(criteria : Criteria, con : PropelPD...
+doSelect(criteria : Criteria, con : PropelPDO = n...
+doSelectStmt(criteria : Criteria, con : PropelPD...
+doDeleteAll(con = null) : int
+doDelete(values, con : PropelPDO = null) : int
    
```

Figura 4. Comportamiento del patrón Experto en el modelado del diseño

Alta Cohesión: Symfony permite la organización del trabajo en cuanto a la estructura del proyecto y la asignación de responsabilidades con una alta cohesión. Un ejemplo de ello es la clase Actions, la cual está formada por varias funcionalidades que están estrechamente relacionadas, siendo la misma la responsable de definir las acciones para las plantillas y colaborar con otras para realizar diferentes operaciones, instanciar objetos y acceder a las properties.

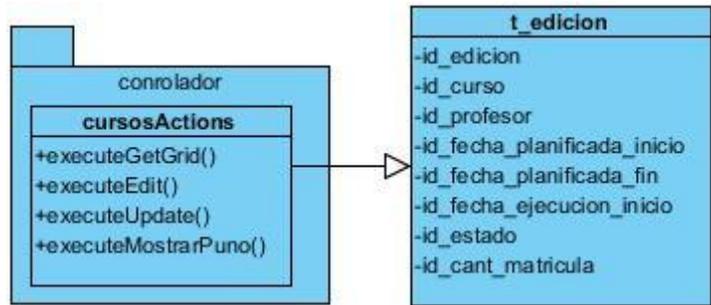


Figura 5. Comportamiento del patrón Alta Cohesión en el modelado del diseño

Bajo acoplamiento: En la capa del Modelo se encuentran las clases que implementan la lógica de negocio y de acceso a datos, estas clases tienen pocas asociaciones con otras de la Vista o el Controlador por lo que la dependencia en este caso es baja, poniéndose de manifiesto el patrón Bajo Acoplamiento [21].

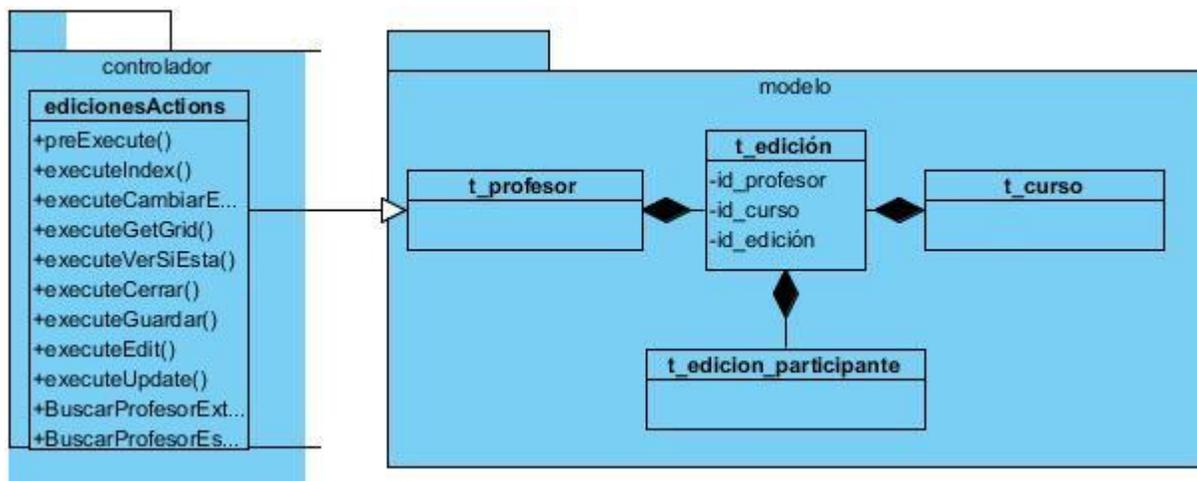


Figura 6. Comportamiento del patrón Bajo Acoplamiento en el modelado del diseño

Controlador: Este se basa en la existencia de un intermediario entre las páginas clientes y el algoritmo que responde a las peticiones realizadas por estas. La existencia del controlador frontal, es el ejemplo básico

que evidencia de forma clara su utilización en Symfony. Este maneja las peticiones del usuario, la seguridad, carga la configuración de la aplicación y otras tareas, es único para cada aplicación. En busca de aminorar un poco la carga que este posee se cuentan con las acciones que contienen las especificaciones de cada página.

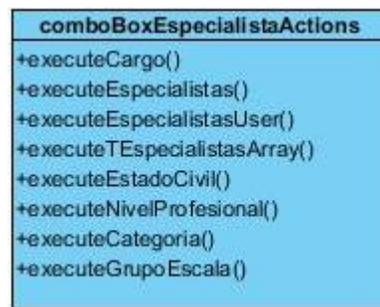


Figura 7. Comportamiento del patrón Controlador en el modelado del diseño

Decorador: Symfony presenta el denominado archivo layout.php o también conocido como plantilla global, en la que convergen todos los elementos comunes a cada una de las páginas del sistema en construcción. Este fichero se complementa con las plantillas, decorándolas y obteniéndose la interfaz final que será mostrada al usuario.

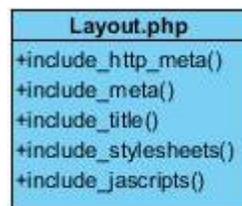


Figura 8. Comportamiento del patrón Decorador en el modelado del diseño

Aplicar estos patrones posibilita la reutilización del diseño y logra identificar aspectos claves de la estructura de un diseño que puede ser aplicado en varias situaciones. De esta manera se reducen los esfuerzos de desarrollo y mantenimiento, mejora la seguridad, eficiencia y consistencia del diseño. Además de elevar la flexibilidad, modularidad y extensibilidad del software que se encuentre en desarrollo.

3.3 DIAGRAMA DE CLASES

Los diagramas de clases (DC) facilitan las representaciones a partir de las cuales los desarrolladores podrían trabajar y colaborar en lo referente al análisis y diseño. Es un tipo de diagrama de estructura

estática que describe la estructura de un sistema mostrando las clases del sistema, sus atributos, métodos, y las relaciones entre los objetos (Ver Figura 8 y 9).

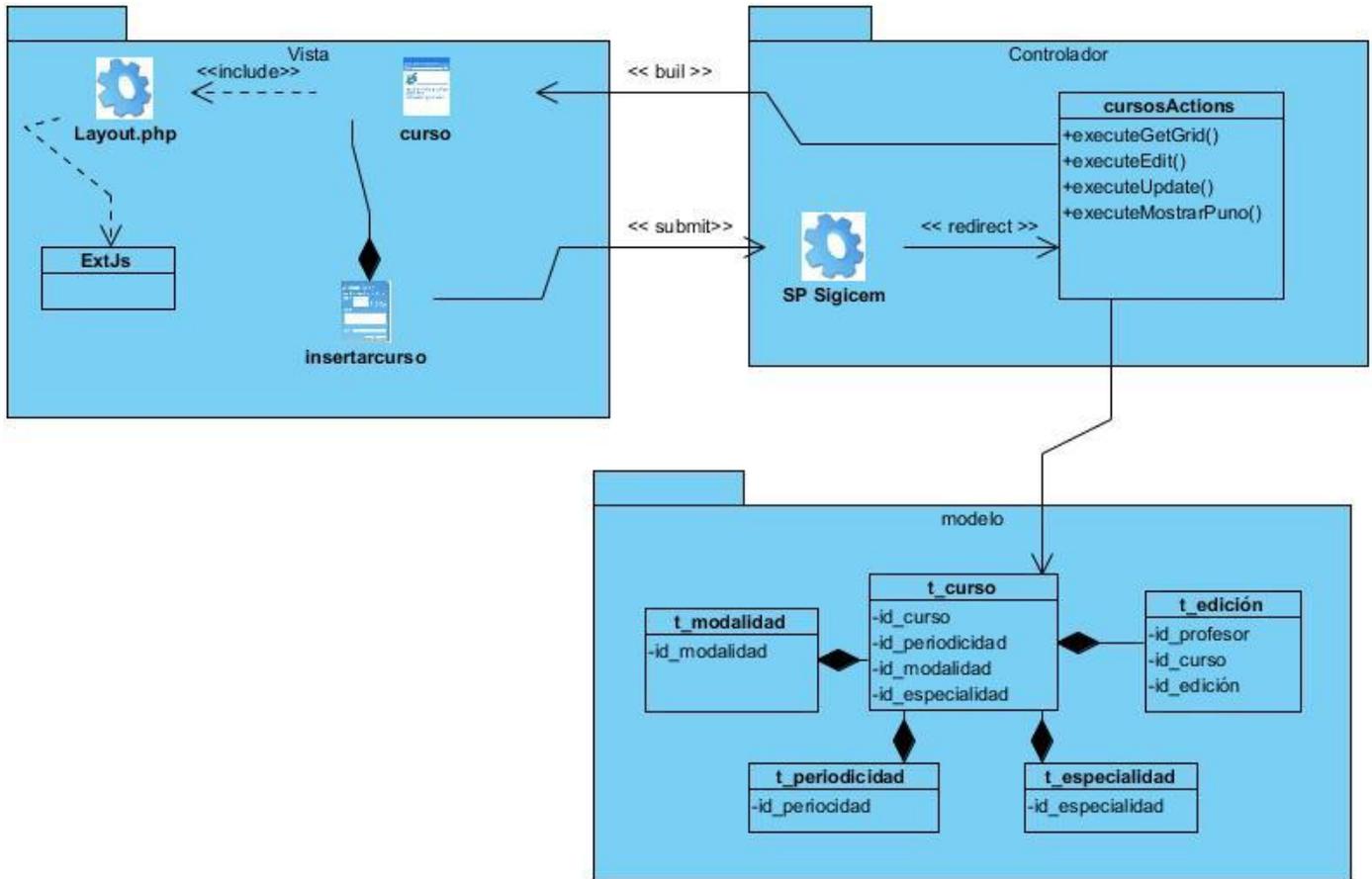


Figura 9. Diagrama de clases insertar curso

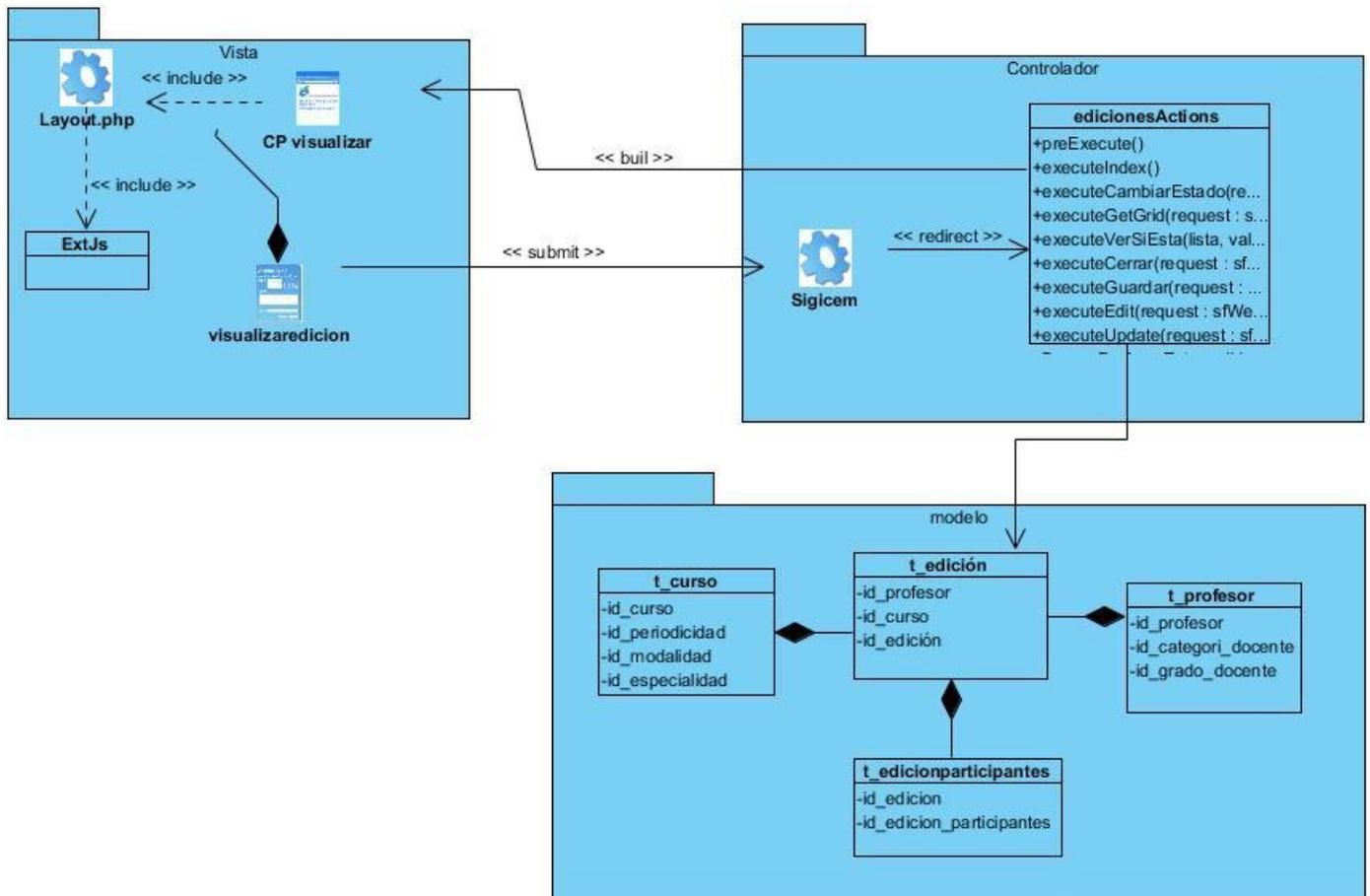


Figura 10. Diagrama de clases visualizar edición.

El resto de las figuras de los diagramas de clases se pueden encontrar en los anexos de este documento (Anexos 19 y 20).

3.4 MODELO DE DATOS

En el desarrollo de un sistema informático, el diseño de la Base de Datos (BD) es de suma importancia, debido que permite almacenar todos los datos que son necesarios en la modelación del problema que se desea resolver, además es la fuente de obtención de toda la información que se quiera recuperar del sistema. La base de datos necesita la definición de una estructura que permita almacenar datos, así como recuperar la información deseada. El Modelo de datos representa todas las tablas de la base de datos utilizadas en la implementación del subsistema, permite describir los datos contenidos en las mismas y las

muestran los componentes de software que constituyen una parte reusable, sus interfaces, y sus interrelaciones, en muchos aspectos se puede considerar que un diagrama de componentes es un diagrama de clases a gran escala. Cada componente en el diagrama debe ser documentado con un diagrama de componentes más detallado, un diagrama de clases o un diagrama de casos de uso [23].

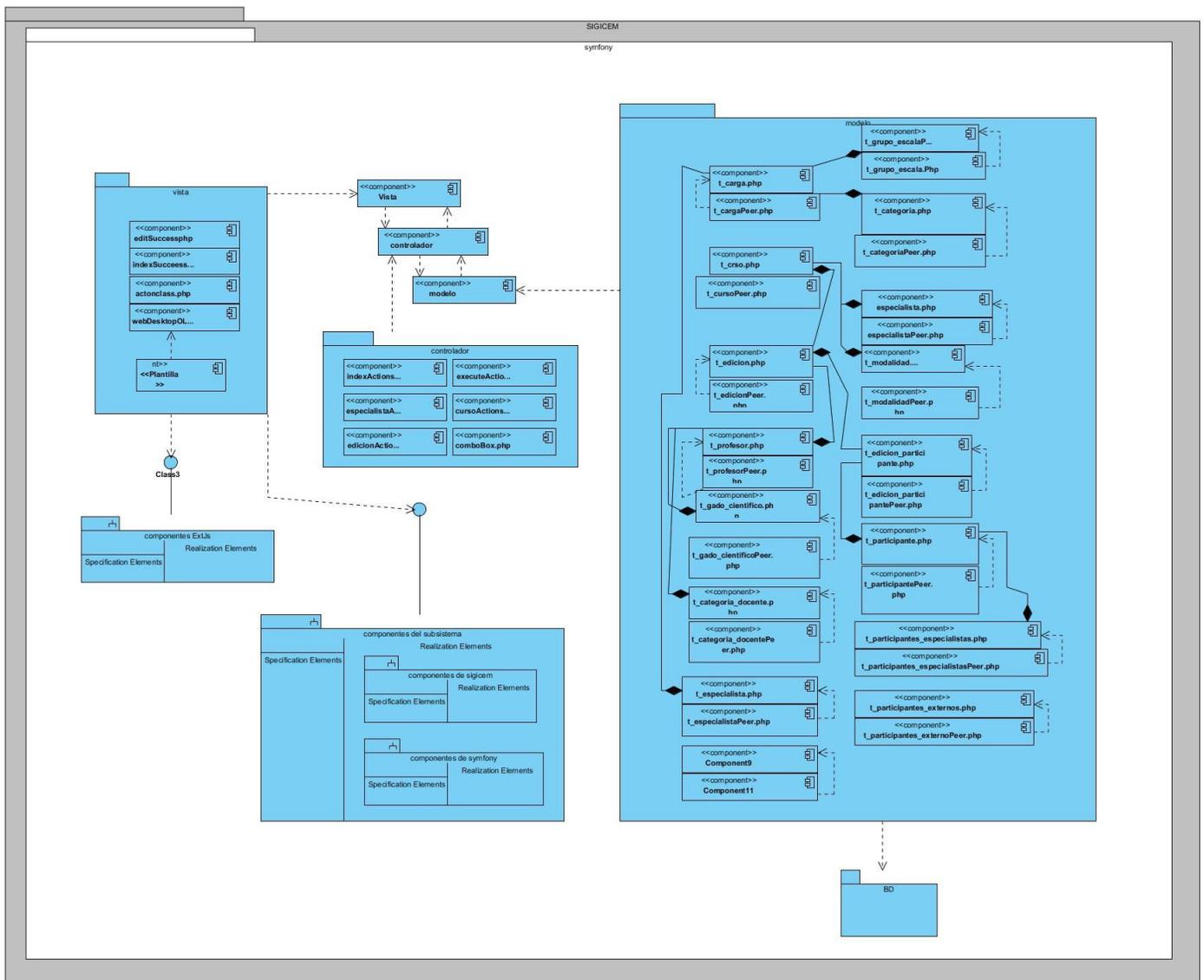


Figura 12. Diagrama de componentes

3.6 DIAGRAMA DE DESPLIEGUE.

El diagrama de despliegue como se muestra en la Figura 13, es un recurso que se utiliza para emular las condiciones físicas en lo que a hardware se refiere en las que va a estar desplegado el sistema.

La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación [24]. Físicamente, el sistema será distribuido por toda la red de centros de ingeniería clínica y electromedicina pasando por todos los niveles, desde la estructura básica, hasta los municipios, provincias y en el centro nacional. Para la estructura de la distribución física del subsistema se utilizó un nodo PC_Cliente que representa la computadora del usuario. Esta se conecta mediante el protocolo de Transferencia de Hipertexto (HTTPS, Hypertext Transfer Protocol) al nodo Servidor Web. Desde el servidor se puede acceder al nodo Servidor de BD mediante el protocolo diseñado para facilitar la reutilización de código de bases de datos (TCP/IP). Además, desde el nodo PC_Cliente se accede a la impresora mediante el puerto USB (Ver Figura 13).

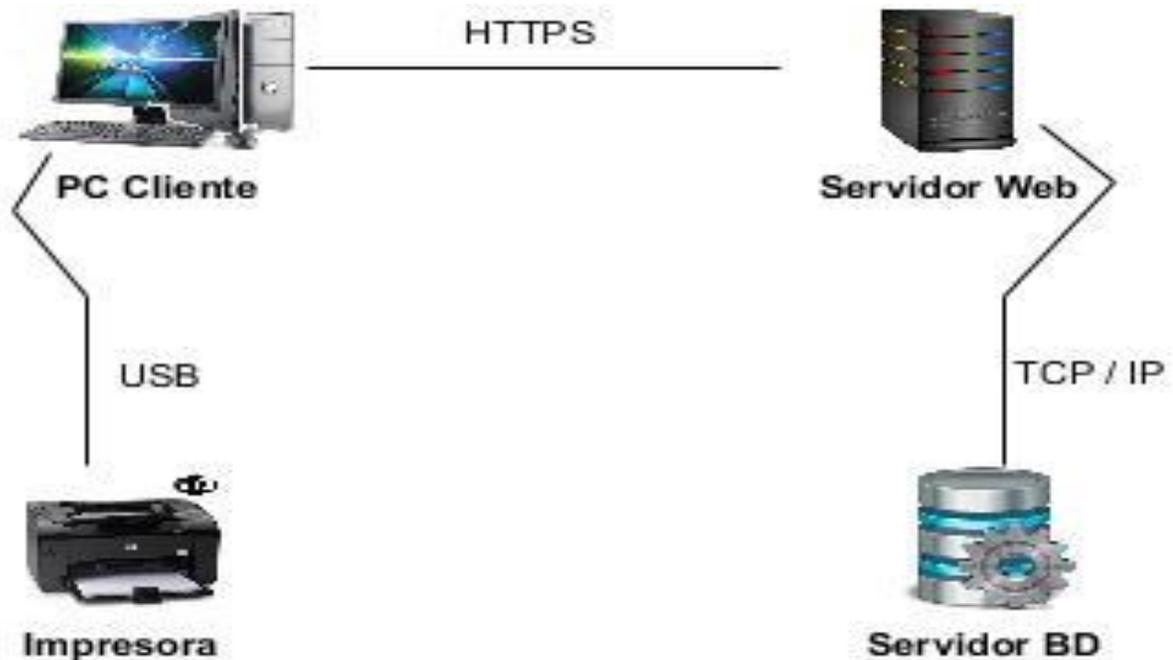


Figura 13. Diagrama de despliegue

Sintetizando lo desarrollado, se concluye que a lo largo de este capítulo se identificó el patrón de arquitectura a emplearse en la solución, los patrones de diseño. También se obtuvieron los diagramas de clases. De la misma manera se elaboró el modelo de datos del sistema, el diagrama de componente y diagrama de despliegue.

Pruebas de software

El único instrumento adecuado para determinar el grado de calidad de un producto de software, es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el nivel en que el software cumple con los requerimientos [25].

De los tipos de pruebas de software existentes se selecciona para comprobar la calidad de la solución desarrollada las pruebas de aceptación, ya que este es el tipo de pruebas recomendado por la metodología de desarrollo empleada. Es válido aclarar que estas pruebas las realiza el propio cliente acompañado del equipo de desarrollo y se orientan a las funcionalidades del sistema. Este tipo de pruebas genera como artefacto los Casos de Pruebas (CP).

Pruebas de aceptación

Las pruebas de aceptación son realizadas por el cliente a partir de las historias de usuario. Básicamente aplicadas al sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, son aplicadas al producto terminado e integrado [26].

Una prueba de aceptación puede ir desde un informal caso de prueba hasta la ejecución sistemática de una serie de pruebas bien planificadas. Pueden tener lugar a lo largo de semanas o meses, descubriendo así errores latentes o escondidos que pueden ir degradando el funcionamiento del sistema. Estas pruebas son muy importantes, ya que definen el paso nuevas fases del proyecto como el despliegue y mantenimiento.

Diseño de casos de prueba

Un caso de prueba cubre el software más a fondo y con más detalle que un caso de uso. Los mismos incluyen todas las funciones que el programa es capaz de realizar. Estos deben tener en cuenta el uso de

todo tipo de datos de entrada/salida, cada comportamiento esperado, todos los elementos de diseño, y cada clase de defecto. Todos los requisitos deberán ser cubiertos por los casos de prueba [27].

A continuación, se muestra un CP correspondiente a las pruebas de aceptación realizadas.

Tabla 3. Caso de prueba Insertar edición

Prueba de Aceptación	
Número: 1	Historia de usuario: 11
Nombre: Insertar Edición.	
Descripción: Comprobar que la opción adicionar edición funciona correctamente.	
Condiciones de Ejecución: El usuario con el rol de profesor debe estar autenticado.	
Entradas/ Pasos de Ejecución: <ul style="list-style-type: none"> • Seleccionar inicio / capacitación / edición de curso / adicionar. • Llenar el formulario. • Marcar opción guardar . 	
Resultado esperado: Se muestra una ventana con los datos de la edición insertada.	
Evaluación de la prueba: Satisfactoria.	

Siendo realizadas 3 iteraciones de pruebas, los resultados obtenidos son los siguientes:

Tabla 4. Resultado de las iteraciones.

Sistema	HU	Iteración	NC	Cerradas
Módulo de gestión de la información del departamento de docencia y capacitación del CNE para el SIGICEM	18	1ra	7	7
		2da	4	4
		3ra	0	0

Durante el plan de iteraciones realizado, para llevar a cabo la fase de pruebas con el objetivo de validar el correcto funcionamiento de la aplicación desarrollada, se detectaron 11 no conformidades las cuales fueron resueltas una vez realizadas tres iteraciones. Las no conformidades encontradas pueden clasificarse según el tipo de falta:

De presentación: No se visualiza toda la información que se desea mostrar y fueron localizados varios errores ortográficos siendo estos generalmente de acentuación.

De ejecución: Durante el despliegue del sistema fueron detectadas diferentes fallas, siendo estas ocasionadas porque el sistema no realizaba todos los procedimientos definidos.

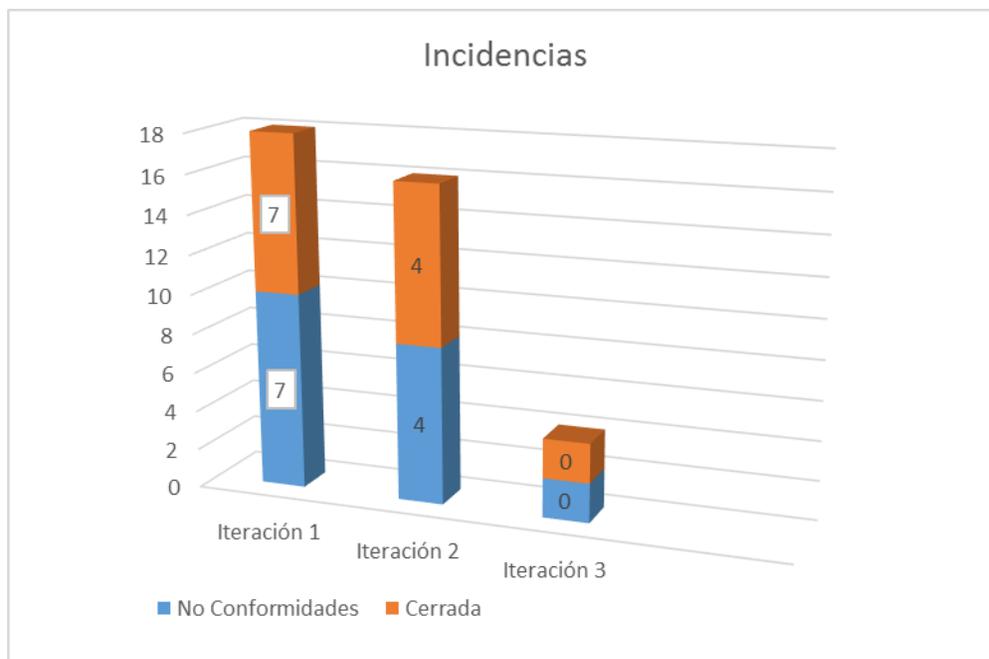


Figura 14. Incidencias detectadas en el sistema

CONCLUSIONES PARCIALES

Al finalizar el presente capítulo se arriba a las siguientes conclusiones:

- La propuesta de arquitectura del sistema se sustenta en un conjunto de componentes reutilizables que tienen como base el patrón arquitectónico MVC, lo que conforma un sistema robusto y flexible a cambios.

- El empleo de patrones de diseño garantizó una solución que tiene como premisa la reutilización de código durante la fase de implementación del software.
- Se ejecutaron las pruebas de aceptación de manera satisfactoria, donde a partir de los casos de prueba se pudo verificar que el software desarrollado está listo y que puede ser usado por los usuarios finales, para ejecutar aquellas funciones y tareas para las cuales fue diseñado y construido.

CONCLUSIONES GENERALES

Con la culminación de la presente investigación se concluye que:

- El estudio de los referentes teóricos y de los sistemas existentes evidenció la necesidad de llevar a cabo el desarrollo del módulo de gestión de información del departamento docencia y capacitación del Centro Nacional de Electromedicina para el SIGICEM.
- Con la utilización de patrones de diseño y patrones arquitectónicos, se logró implementar un sistema de acuerdo a los estándares y modelos utilizados en el desarrollo de software que responden a las necesidades de gestión de la información del departamento de docencia y capacitación en el CNE.
- Con los resultados obtenidos en la validación de los requerimientos, se comprobó el correcto funcionamiento del sistema, la estabilidad y la calidad necesaria de la solución, por tanto, las funcionalidades descritas satisfacen las necesidades del cliente.

RECOMENDACIONES

Luego de haber concluido el presente trabajo se recomienda:

- Integrar el modulo al sistema.
- Evaluar por el grupo de desarrollo del sistema, la posibilidad de migrar al gestor de base de dato Postgres
- Brindar capacitación al personal que utilizará el subsistema, sobre su funcionamiento y prestaciones.

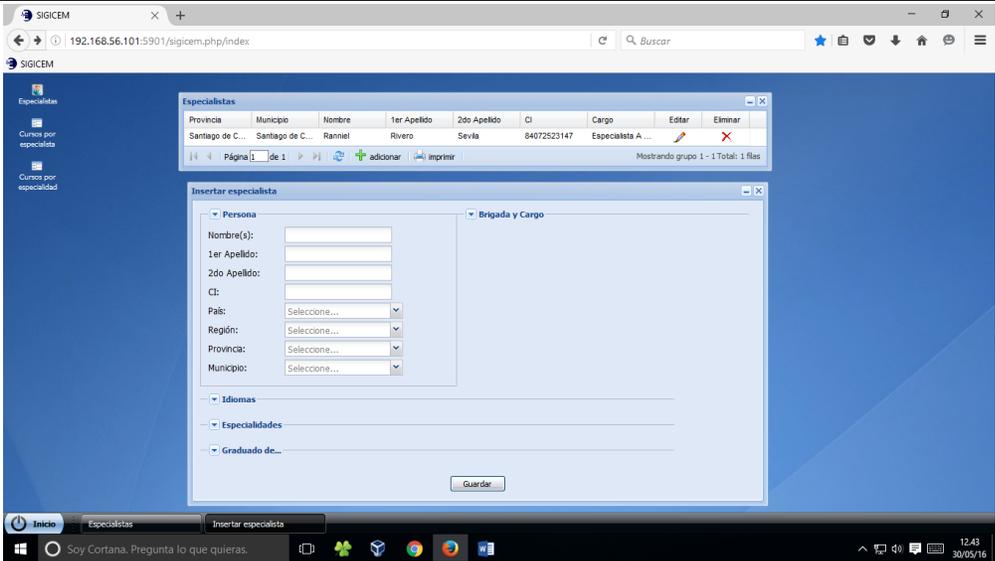
BIBLIOGRAFÍA

1. Hernández, I., S. Fleitas, and D. Salazar, *La gestión del capital humano en empresas cubanas y sus particularidades*. Revista Venezolana de Análisis de Coyuntura, 2011. **Vol. XVII**: p. 137-148.
2. Ruiz, W.P. *Trascendencia de la capacitación en la empresa*. 2012; Available from: <http://wperezruiz.blogspot.com/2012/05/trascendencia-de-la-capacitacion-en-la>.
3. MES, A., *Ley de Educación Nacional*. 2006.
4. Rienzo, S.D. *Altamira Employees*
2011; Available from: <http://www.altamirahrm.com/es/software/software-rrhh>.
5. Carnegie, M. *Dataproces Soluciones Registradas (DSR)*
2011; Available from: https://es.over-blog.com/Software_recomendado_para_la_gestion_de_almacenes-122.
6. Akademos. *Sistema de Gestión Universitaria*. 2009; Available from: <http://akademos.uci.cu>.
7. Jacobson, I., G. Booch, and J. Runbaugh, *El Proceso Unificado de Desarrollo de Software*. 2000, Madrid.
8. Sánchez, T.R. *Metodología de Desarrollo para la Actividad Peroductiva de la UCI*
2015; Available from: <http://excriba.prod.uci.cu/page/context/shared/sharedfiles/Metodologiauci.pdf>.
9. Flores, E. and J.L. Cordero, *Metodologías ágiles "Proceso unificado ágil (AUP)"*. 2005.
10. Paradigm, V., *Visual paradigm for uml*. Visual Paradigm for UML-UML tool for software application development, 2013.
11. Pressman, R.S., *Mc Grah-Hill, 7a. edició, 2010*. Ingeniería del software, un enfoque práctico, 2010.
12. NetBeans. *NetBeans*. 2015 [cited 2015 11/11/2015]; Available from: https://netbeans.org/index_es.html.
13. LibrosWeb. *Symfony en pocas palabras*. 2014 [cited
Available from: http://librosweb.es/symfony_1_4/capitulo_1/symfony_en_pocas_palabras.html.
14. ExtJs. *ExtJs Comunidad en español*. 2013; Available from: <http://extjses.com/>.
15. Velmour, R.A.C.C.I., M. Casals, and L. del Valle Román. *SLD252 CAPA DE GESTIÓN DE SERVICIOS WEB PARA EL SISTEMA DE GESTIÓN PARA LA INGENIERÍA CLÍNICA Y ELECTROMEDICINA*. in *Informática Salud 2013*. 2013.
16. Ciberaula. *Master LAMP: La plataforma web libre*. 2014; Available from: http://www.ciberaula.com/curso/lamp/que_es/.
17. Argulo, I. *SGM. Diseño web y programación. Modelo Vista Controlador*. . 2013; Available from: <http://www.sgmweb.es/modelo.as>.
18. Tedeschi, N. *¿Qué es un Patrón de Diseño?* . 2013; Available from: <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.
19. Delgado, J.M. *Patrones de diseño*. 2009; Available from: <http://www.slideshare.net/jmruizforem/patrones-de-diseo-en-e-learning>.
20. Larman, C., *UML y Patrones*. Mexico: Down Speth White. 2000.
21. Machado Díaz, M. and Y.L. Coca Ribas, *SACCEM: Módulo de gestión de la información del Departamento de Supervisión del CCEEM 2011*, Universidad de las Ciencias Informáticas (UCI).

22. Acuña, K.B., *Metodología de desarrollo para aplicaciones web* 2009, Obtenido de www.eumed/libros/2009c/584.
23. LibrosWeb. *Métodos de validación y manejo de errores* 2011; Available from: http://www.librosweb.es/symfony/capitulo6/metodos_de_validacion_y_manejo_de_errores.html.
24. Marca Huallpara, H.M. and N.S. Quisbert Limachi. *Trabajo de Investigación y Exposición: Diagrama de Despliegue*. 2013; Available from: http://www.google.com/cu/url?sa=t&rct=j&q=modelo+de+despliegue&source=web&cd=6&ved=0CEEQFjAF&url=http%3A%2F%2Fvirtual.usalesiana.edu.bo%2Fweb%2Fpractica%2Farchiv%2Fdespliegue.doc&ei=FbNmUe3uJYLi4AP73IDwAQ&usg=AFQjCNGu7s__JDdYCEwEmbkYnxa2XI2Q2Q&bvm=bv.4510.
25. PRUEBASDESFTWARE. *Pruebas de software*. 2016; Available from: <http://www.pruebasdesoftware.com/laspruebasdesoftware.htm>.
26. 4rsoluciones. *Test de aceptación: el último paso para el aseguramiento de calidad de software*. 2012 [cited 2012; Available from: <http://www.4rsoluciones.com/test-de-aceptacion-el-ultimo-paso-para-el-aseguramiento-de-calidad-en-software/>.
27. TesteandoSoftware. *Casos de Uso vs. Casos de Prueba*. 2013; Available from: <http://testeandosoftware.com/casos-de-uso-vs-casos-de-prueba/>.

ANEXOS

ANEXO #1. HU Insertar especialista

Historia de usuario	
Número: HU 1	Nombre del requisito: Insertar especialista
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none">○ Problemas eléctricos.○ Problemas técnicos.	Tiempo real: 4 días
Descripción: <p>Se visualiza una ventana donde se introducen los datos del especialista a registrar en el sistema para ello se debe de recoger la siguiente información (Graduado de, Categoría Salarial que esta va desde la escala I hasta la XIII, Grupo escala a la que pertenece (Administrativas, Común Técnicos, Genéricos MINSAP, Operarios, Seguridad y Protección, Servicios, Dirigentes y Otras), Cargo que ocupa e Idiomas que domina). El sistema válido la información registrada y se guarda de manera satisfactoria en la base de datos.</p>	
Observaciones:	
Prototipo de interfaz:	 The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The page title is 'SIGICEM'. On the left, there is a sidebar with 'Especialistas', 'Cursos por especialista', and 'Cursos por especialidad'. The main content area features a table titled 'Especialistas' with columns: Provincia, Municipio, Nombre, 1er Apellido, 2do Apellido, CI, Cargo, Editar, and Eliminar. One row is visible with data: Santiago de C..., Santiago de C..., Ramiel, Rivero, Sevilla, 84072523147, Especialista A... Below the table is a form titled 'Insertar especialista' with two tabs: 'Persona' and 'Brigada y Cargo'. The 'Persona' tab is active, showing input fields for 'Nombre(s)', '1er Apellido', '2do Apellido', 'CI', 'País', 'Región', 'Provincia', and 'Municipio'. Each of these fields has a dropdown menu for selection. There are also sections for 'Idiomas', 'Especialidades', and 'Graduado de...'. A 'Guardar' button is at the bottom of the form. The Windows taskbar at the bottom shows the Start button, Cortana search, and system tray with the date 30/05/16 and time 12:43.

ANEXO #2. HU Actualizar especialista

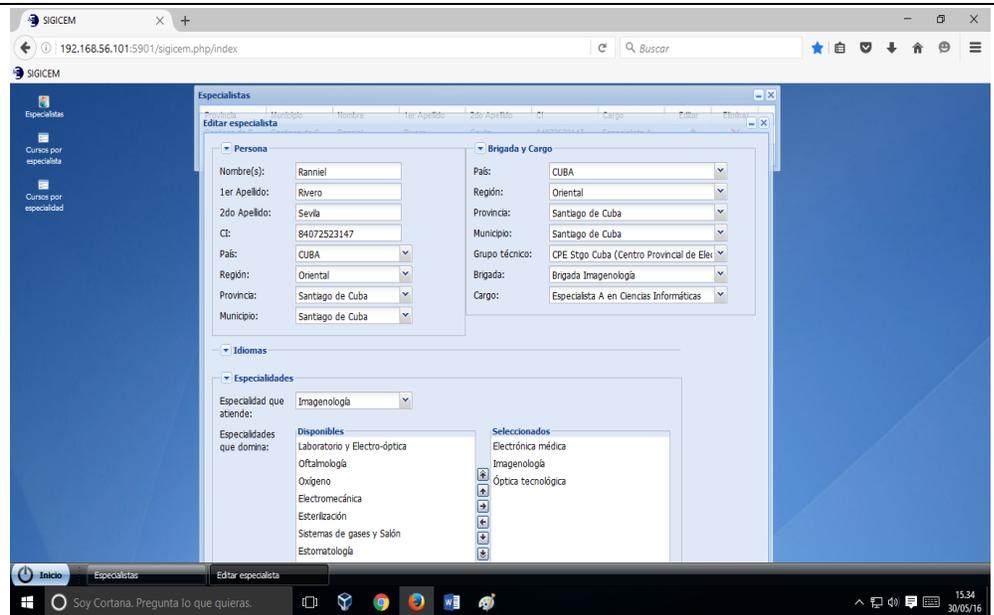
Historia de usuario	
Número: HU 2	Nombre del requisito: Actualizar especialista
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días

Descripción:

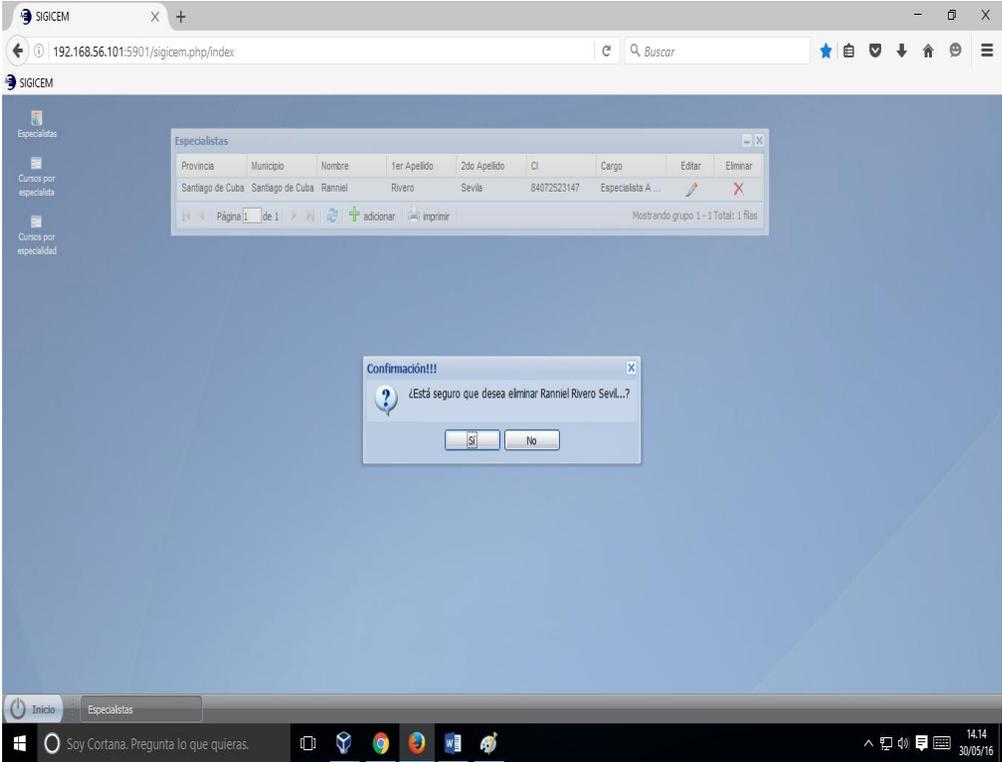
Se visualiza una ventana con los datos del especialista que pueden ser (Graduado de, Categoría Salarial que esta va desde la escala I hasta la XIII, Grupo escala a la que pertenece (Administrativas, Común Técnicos, Genéricos MINSAP, Operarios, Seguridad y Protección, Servicios, Dirigentes y Otras), Cargo que ocupa e Idiomas que domina. Se muestra un mensaje de confirmación de los cambios, el usuario confirma. El sistema, válida la información registrada y se guarda de manera satisfactoria en la base de datos.

Observaciones:

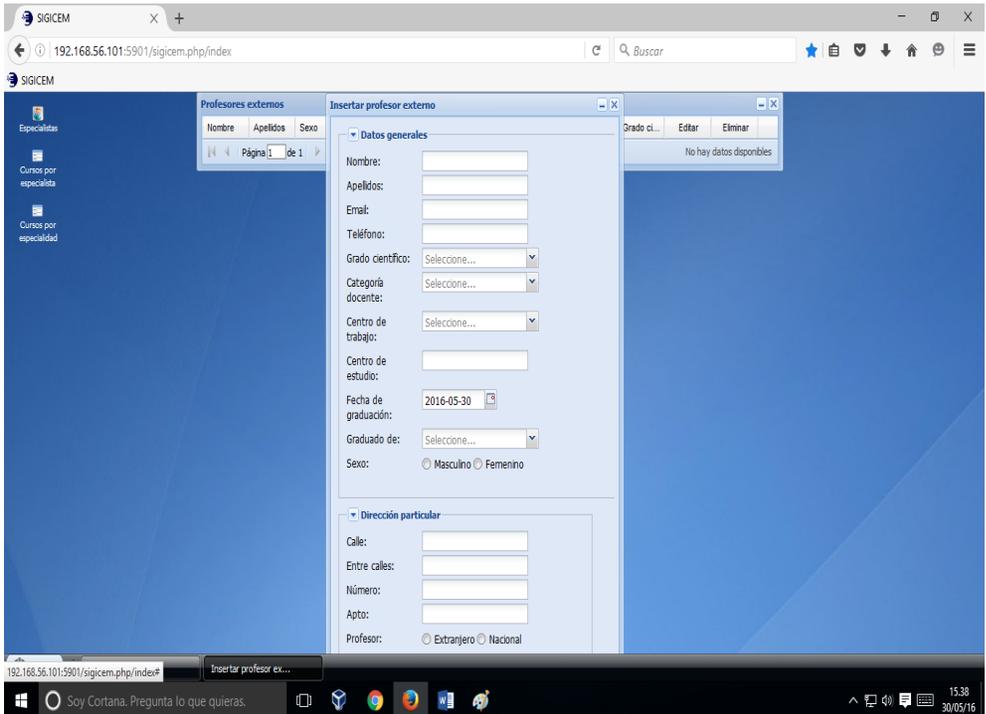
Prototipo de interfaz:



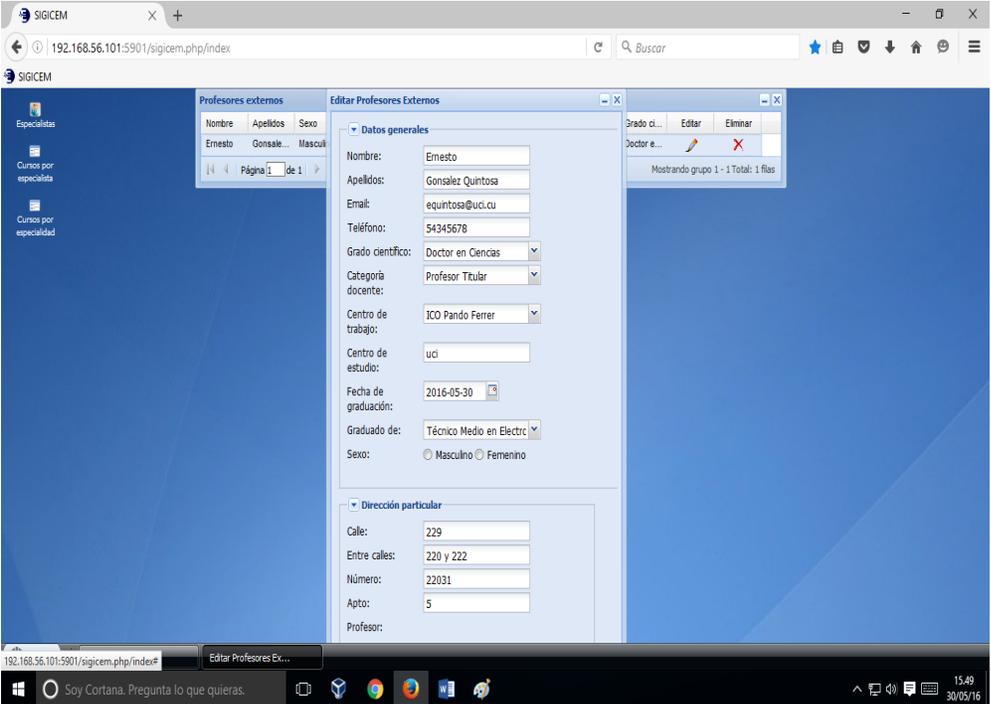
ANEXO #3. HU Eliminar especialista

Historia de usuario	
Número: HU 3	Nombre del requisito: Eliminar especialista
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con el listado de los especialistas, el usuario selecciona el especialista que desea eliminar, se muestra un mensaje de confirmación. El sistema realiza la acción seleccionada (Aceptar o Cancelar) y se guarda de manera satisfactoria los cambios en la base de datos.	
Observaciones:	
Prototipo de interfaz:  <p>The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The page displays a table of specialists with columns for Provincia, Municipio, Nombre, 1er Apellido, 2do Apellido, CI, and Cargo. A confirmation dialog box is overlaid on the table, asking: '¿Está seguro que desea eliminar Ranniel Rivero Sev...?' with 'Si' and 'No' buttons. The table contains one row: Santiago de Cuba, Santiago de Cuba, Ranniel, Rivero, Sevilla, 84072523147, Especialista A... The browser's taskbar shows the date 30/05/16 and time 14:14.</p>	

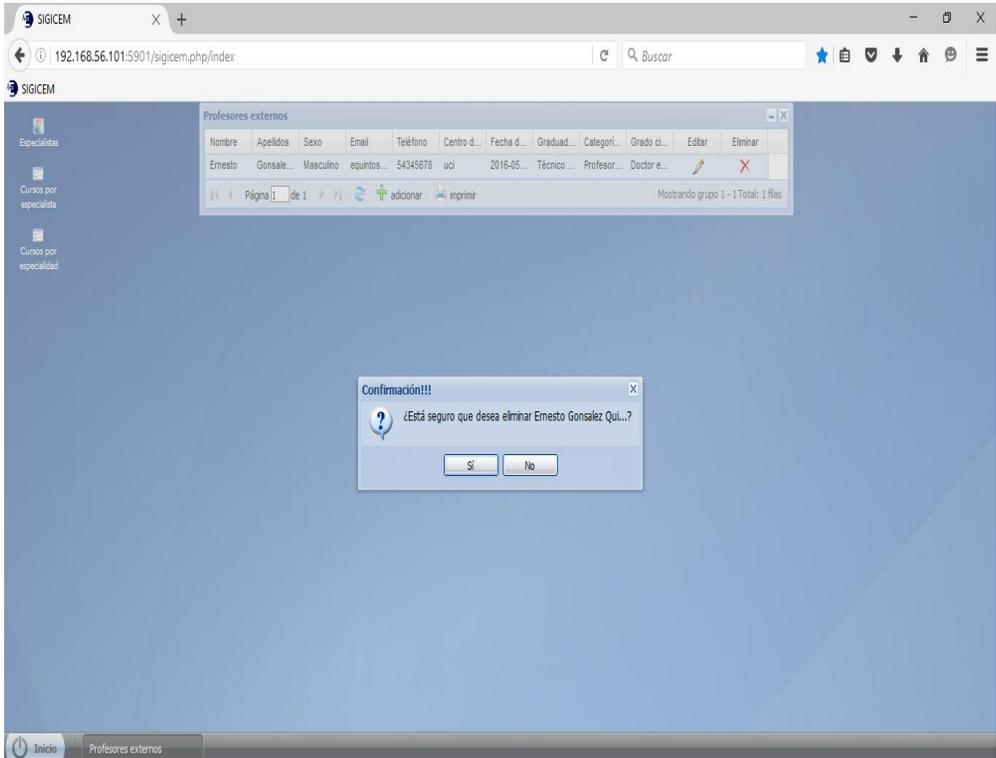
ANEXO #4. HU Insertar profesor

Historia de usuario	
Número: HU 4	Nombre del requisito: Insertar profesor
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana donde se introducen los datos del profesor (categoría docente, grado científico y si es interno o externo a la institución), si es un profesor externo se visualizan los datos que deben de ser completados. El sistema, válida la información registrada y se guarda de manera satisfactoria en la base de datos.	
Observaciones:	
Prototipo de interfaz: 	

ANEXO #5. HU Actualizar profesor

Historia de usuario	
Número: HU 5	Nombre del requisito: Actualizar profesor
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con los datos del profesor que pueden ser modificados. Se muestra un mensaje de confirmación de los cambios, el usuario confirma. El sistema, válida la información registrada y se guarda de manera satisfactoria en la base de datos.	
Observaciones:	
Prototipo de interfaz: 	

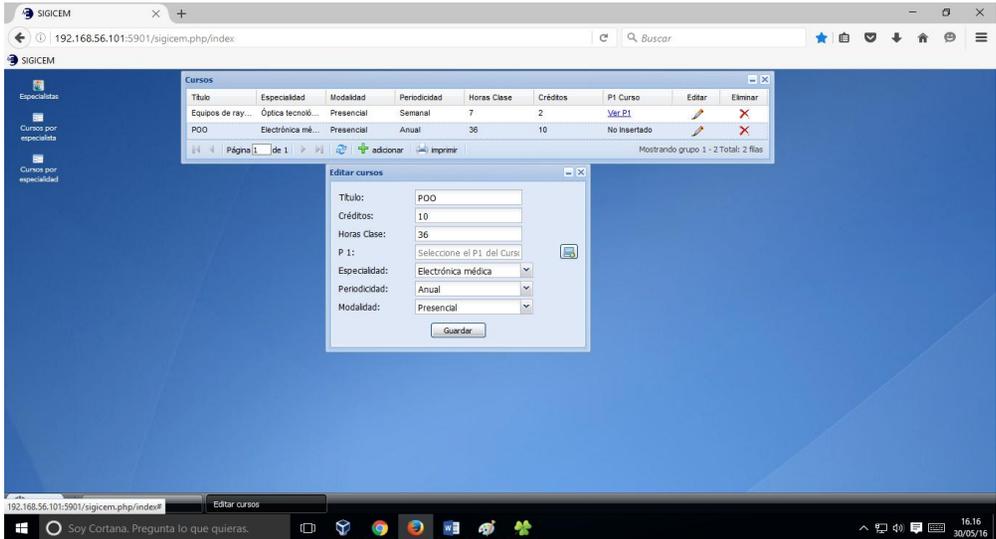
ANEXO #6. HU Eliminar profesor

Historia de usuario	
Número: HU 6	Nombre del requisito: Eliminar profesor
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con el listado de los profesores, el usuario selecciona el profesor que desea eliminar, se muestra un mensaje de confirmación. El sistema realiza la acción seleccionada (Aceptar o Cancelar) y se guarda de manera satisfactoria los cambios en la base de datos.	
Observaciones:	
Prototipo de interfaz:	 <p>The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The main content area displays a table titled 'Profesores externos' with columns: Nombre, Apellidos, Sexo, Email, Teléfono, Centro d..., Fecha d..., Gradua..., Categori..., Grado ci..., Editar, and Eliminar. A single row is visible for 'Ernesto Gonsale...'. A confirmation dialog box is overlaid on the table, asking '¿Está seguro que desea eliminar Ernesto Gonsalez Qui...?' with 'Sí' and 'No' buttons. The browser's taskbar at the bottom shows 'Inicio' and 'Profesores externos'.</p>

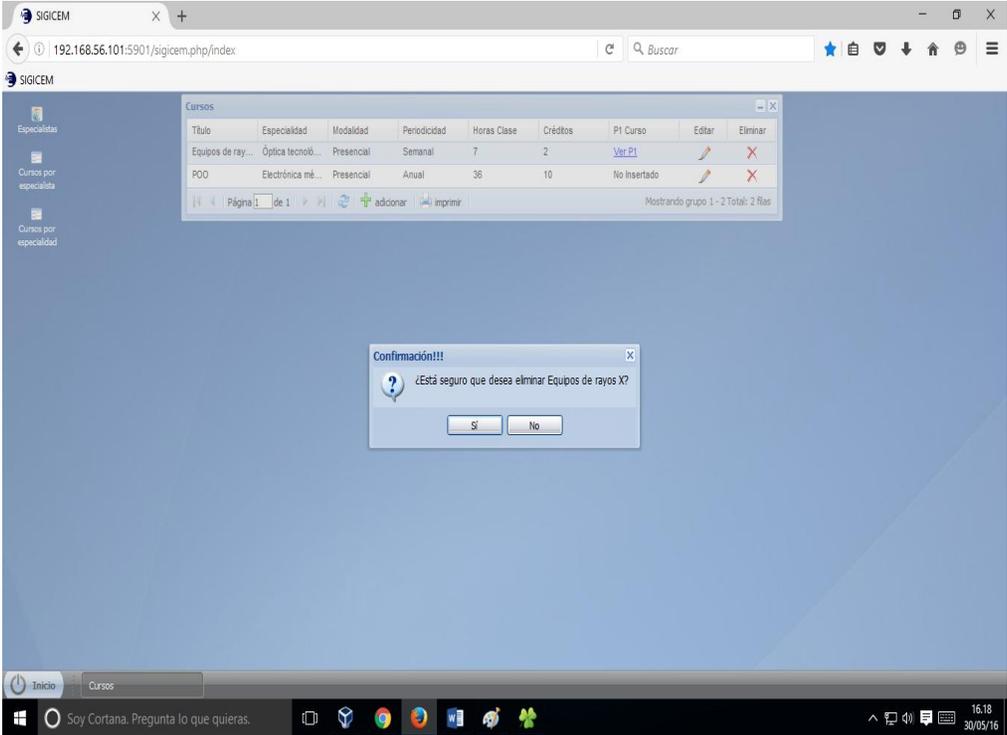
ANEXO #7. HU Insertar curso

Historia de usuario	
Número: HU 7	Nombre del requisito: Insertar curso
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con los datos que se deben registrar del curso (nombre del curso, Modalidad (presencial, semipresencial o a distancia), Periodicidad (diaria, semanal, quincenal, mensual). El sistema, válida la información registrada y se guarda de manera satisfactoria en la base de datos.	
Observaciones	
Prototipo de interfaz:	

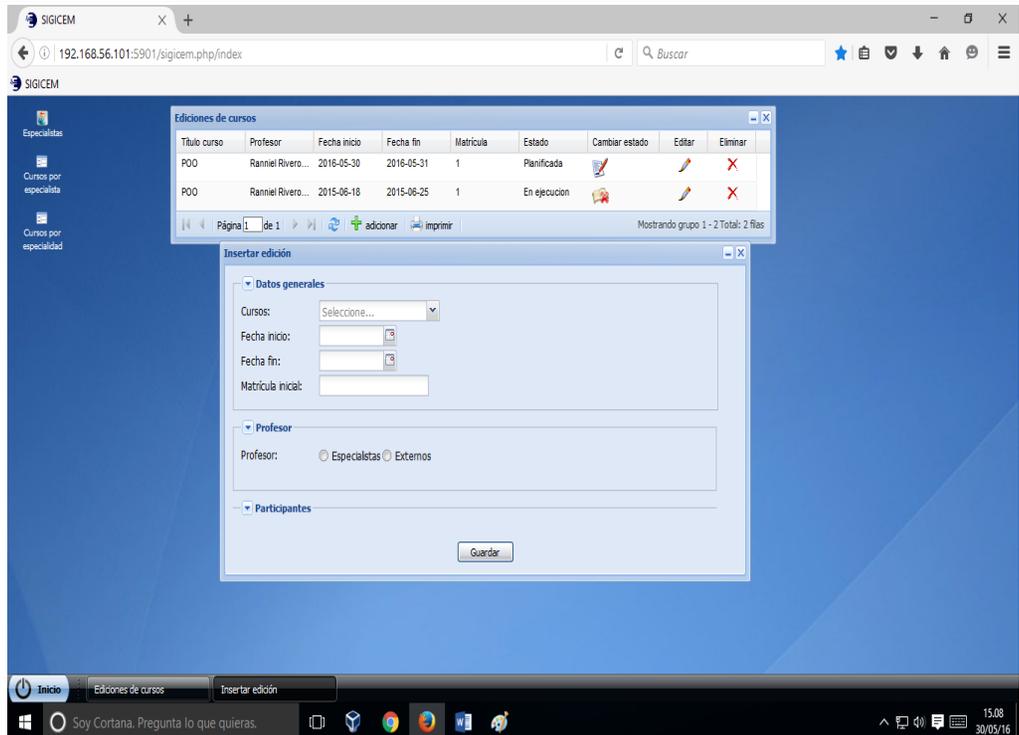
ANEXO #8. HU Actualizar curso

Historia de usuario	
Número: HU 8	Nombre del requisito: Actualizar curso
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con los datos que se pueden actualizar del curso (nombre del curso, Modalidad (presencial, semipresencial o a distancia), Periodicidad (diaria, semanal, quincenal, mensual). El sistema realiza la acción seleccionada (Aceptar o Cancelar) y se guarda de manera satisfactoria los cambios en la base de datos.	
Observaciones:	
Prototipo de interfaz: 	

ANEXO #9. HU Eliminar curso

Historia de usuario																												
Número: HU 9	Nombre del requisito: Eliminar curso																											
Programador: Roberto Cabello	Iteración: 1																											
Prioridad: Media	Tiempo estimado: 5 días																											
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días																											
Descripción: Se visualiza una ventana con el listado de los cursos, el usuario selecciona el curso que desea eliminar, se muestra un mensaje de confirmación. El sistema realiza la acción seleccionada (Aceptar o Cancelar) y se guarda de manera satisfactoria los cambios en la base de datos.																												
Observaciones:																												
Prototipo de interfaz:  <p>The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The main content area displays a table titled 'Cursos' with the following data:</p> <table border="1"> <thead> <tr> <th>Título</th> <th>Especialidad</th> <th>Modalidad</th> <th>Periodicidad</th> <th>Horas Clase</th> <th>Créditos</th> <th>PI Curso</th> <th>Editar</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>Equipos de ray...</td> <td>Óptica tecnoló...</td> <td>Presencial</td> <td>Semanal</td> <td>7</td> <td>2</td> <td>Ver PI</td> <td></td> <td></td> </tr> <tr> <td>POO</td> <td>Electrónica mé...</td> <td>Presencial</td> <td>Anual</td> <td>36</td> <td>10</td> <td>No Insertado</td> <td></td> <td></td> </tr> </tbody> </table> <p>Below the table, a confirmation dialog box is displayed with the title 'Confirmación!!!' and the message '¿Está seguro que desea eliminar Equipos de rayos X?'. The dialog has 'SI' and 'NO' buttons.</p>		Título	Especialidad	Modalidad	Periodicidad	Horas Clase	Créditos	PI Curso	Editar	Eliminar	Equipos de ray...	Óptica tecnoló...	Presencial	Semanal	7	2	Ver PI			POO	Electrónica mé...	Presencial	Anual	36	10	No Insertado		
Título	Especialidad	Modalidad	Periodicidad	Horas Clase	Créditos	PI Curso	Editar	Eliminar																				
Equipos de ray...	Óptica tecnoló...	Presencial	Semanal	7	2	Ver PI																						
POO	Electrónica mé...	Presencial	Anual	36	10	No Insertado																						

ANEXO #10. HU Insertar edición

Historia de usuario	
Número: HU 10	Nombre del requisito: Insertar edición
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días
Descripción: Se visualiza una ventana con los datos que se deben registrar de la edición del curso (título, profesor, fecha de inicio, fecha final, matricula, estado, cambiar de estado, editar, eliminar). El sistema válida la información registrada y se guarda de manera satisfactoria en la base de datos.	
Observaciones:	
Prototipo de interfaz:	 <p>The screenshot displays the SIGICEM web application. At the top, there is a navigation menu with 'Inicio', 'Ediciones de cursos', and 'Insertar edición'. The main content area features a table titled 'Ediciones de cursos' with columns for 'Titulo curso', 'Profesor', 'Fecha inicio', 'Fecha fin', 'Matricula', 'Estado', 'Cambiar estado', 'Editar', and 'Eliminar'. Two rows of data are visible, both for 'POO' courses by 'Ranniel Rivero...'. Below the table, there is a form titled 'Insertar edición' with sections for 'Datos generales' (including 'Cursos', 'Fecha inicio', 'Fecha fin', and 'Matricula inicial'), 'Profesor' (with radio buttons for 'Especialistas' and 'Externos'), and 'Participantes'. A 'Guardar' button is located at the bottom of the form. The browser's address bar shows '192.168.56.101:5901/sigicem.php/index'.</p>

ANEXO #11. HU Actualizar edición

Historia de usuario	
Número: HU 11	Nombre del requisito: Actualizar edición
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 5 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días

Descripción:
 Se visualiza una ventana para actualizar los datos de la edición del curso. El sistema realiza la acción seleccionada (Aceptar o Cancelar) válida la información registrada y se guarda de manera satisfactoria en la base de datos

Observaciones:

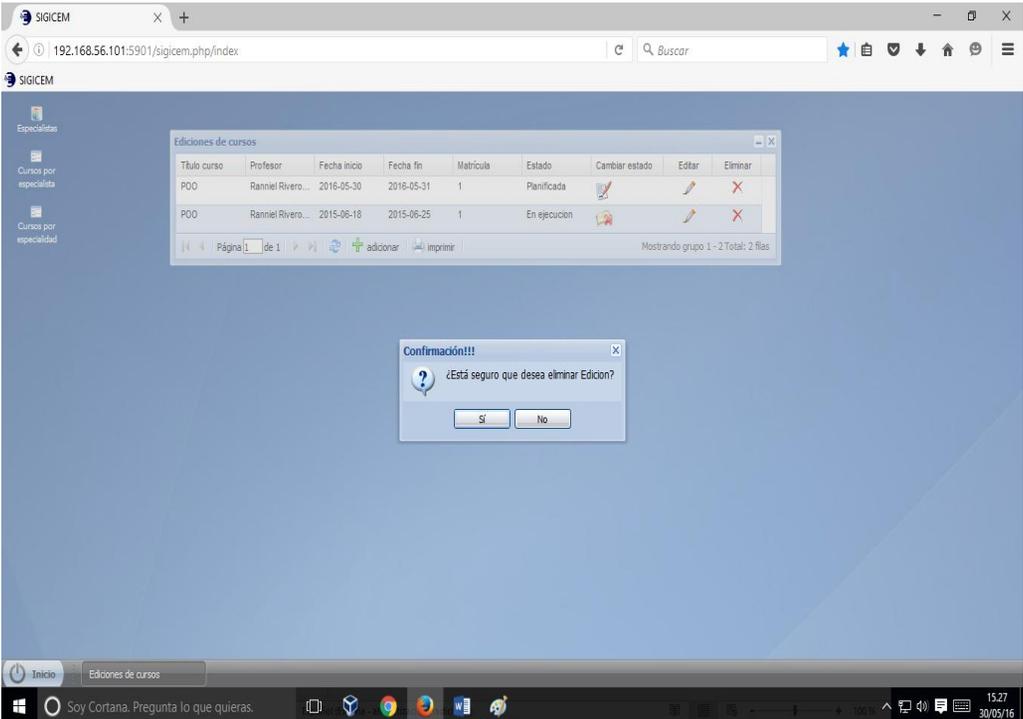
Prototipo de interfaz:

The screenshot displays a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The application interface is titled 'SIGICEM' and features a sidebar with navigation options: 'Especialistar', 'Cursos por especialista', and 'Cursos por especialidad'. The main content area shows a table titled 'Ediciones de cursos' with columns for 'Título curso', 'Profesor', 'Fecha inicio', 'Fecha fin', 'Matrícula', 'Estado', 'Cambiar estado', 'Editar', and 'Eliminar'. A row is visible for 'POO' with 'Danniel Divero' as the professor and '2016-05-30' as the start date. An 'Editar edición' modal window is overlaid on the table, containing the following fields and sections:

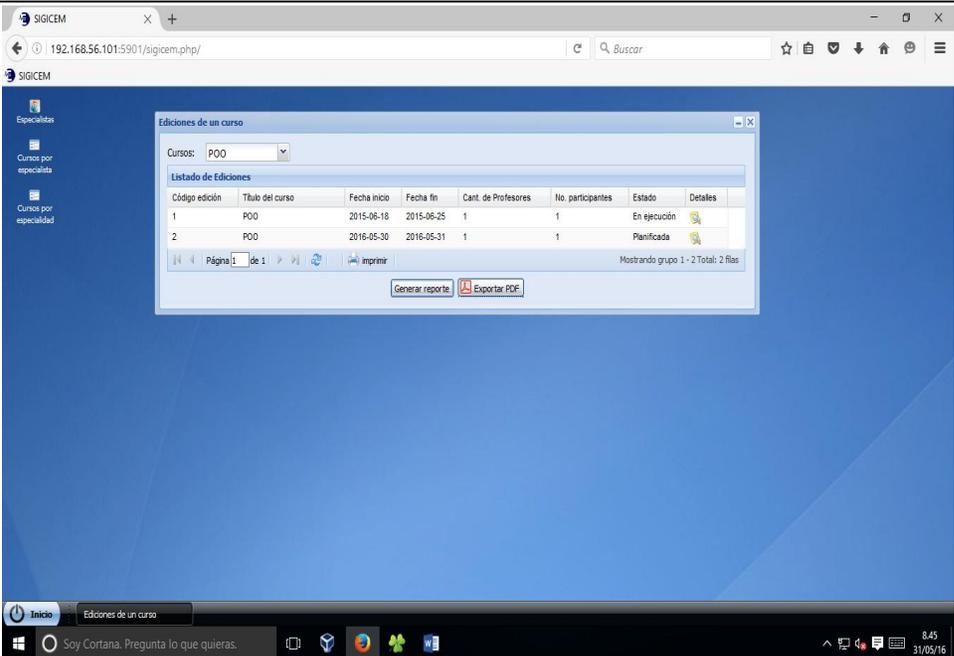
- Datos generales:**
 - Cursos: POO (dropdown)
 - Fecha inicio: 2016-05-30 (calendar icon)
 - Fecha fin: 2016-05-31 (calendar icon)
 - Matrícula inicial: 1 (text input)
- Profesor:** (empty field)
- Participantes:**
 - Disponibles: Raniel Rivero Sevilla
 - Seleccionados: Pepe Lolo

A 'Guardar' button is located at the bottom of the modal. The Windows taskbar at the bottom shows the 'Inicio' button, taskbar buttons for 'Ediciones de cursos' and 'Editar edición', and system tray icons including the time 15:31 and date 30/05/16.

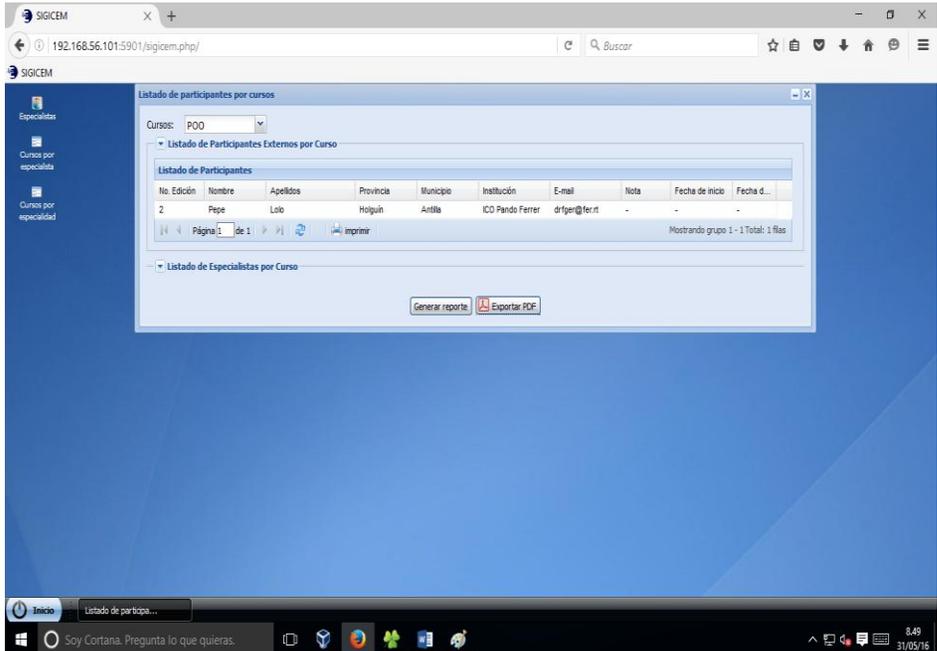
ANEXO #12. HU Eliminar edición

Historia de usuario																												
Número: HU 12	Nombre del requisito: Eliminar edición																											
Programador: Roberto Cabello	Iteración: 1																											
Prioridad: Media	Tiempo estimado: 5 días																											
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 4 días																											
Descripción: Se visualiza una ventana con el listado de las ediciones registradas de los cursos, el usuario selecciona la edición desea eliminar, se muestra un mensaje de confirmación. El sistema realiza la acción seleccionada (Aceptar o Cancelar) y se guarda de manera satisfactoria los cambios en la base de datos.																												
Observaciones:																												
Prototipo de interfaz:  <p>The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/index. The main content area displays a table titled 'Ediciones de cursos' with the following data:</p> <table border="1"> <thead> <tr> <th>Título curso</th> <th>Profesor</th> <th>Fecha inicio</th> <th>Fecha fin</th> <th>Matrícula</th> <th>Estado</th> <th>Cambiar estado</th> <th>Editar</th> <th>Eliminar</th> </tr> </thead> <tbody> <tr> <td>POO</td> <td>Raniel Rivero...</td> <td>2016-05-30</td> <td>2016-05-31</td> <td>1</td> <td>Planificada</td> <td></td> <td></td> <td></td> </tr> <tr> <td>POO</td> <td>Raniel Rivero...</td> <td>2015-08-18</td> <td>2015-08-25</td> <td>1</td> <td>En ejecución</td> <td></td> <td></td> <td></td> </tr> </tbody> </table> <p>Below the table, a confirmation dialog box is displayed with the title 'Confirmación!!!' and the message '¿Está seguro que desea eliminar Edición?'. It has two buttons: 'Sí' and 'No'.</p>		Título curso	Profesor	Fecha inicio	Fecha fin	Matrícula	Estado	Cambiar estado	Editar	Eliminar	POO	Raniel Rivero...	2016-05-30	2016-05-31	1	Planificada				POO	Raniel Rivero...	2015-08-18	2015-08-25	1	En ejecución			
Título curso	Profesor	Fecha inicio	Fecha fin	Matrícula	Estado	Cambiar estado	Editar	Eliminar																				
POO	Raniel Rivero...	2016-05-30	2016-05-31	1	Planificada																							
POO	Raniel Rivero...	2015-08-18	2015-08-25	1	En ejecución																							

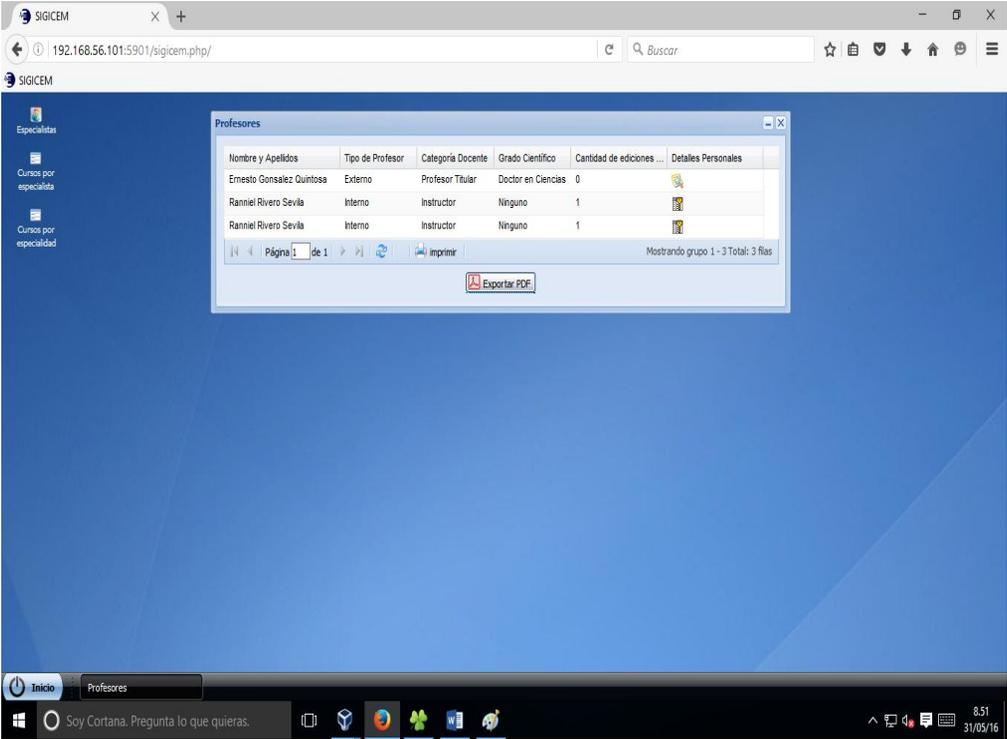
ANEXO #13. HU Mostrar reporte edición de un curso

Historia de usuario	
Número: HU 13	Nombre del requisito: Mostrar reporte edición de un curso
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 3 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 2 días
Descripción: Se visualiza una ventana con los distintos reportes que el usuario puede realizar, para conocer las ediciones que tiene un curso, selecciona este tipo de reporte y el curso del que requiere la información. El sistema devuelve la información solicitada, esta se puede exportar en los siguientes ficheros(pdf) y permite que se imprima.	
Observaciones:	
Prototipo de interfaz:	

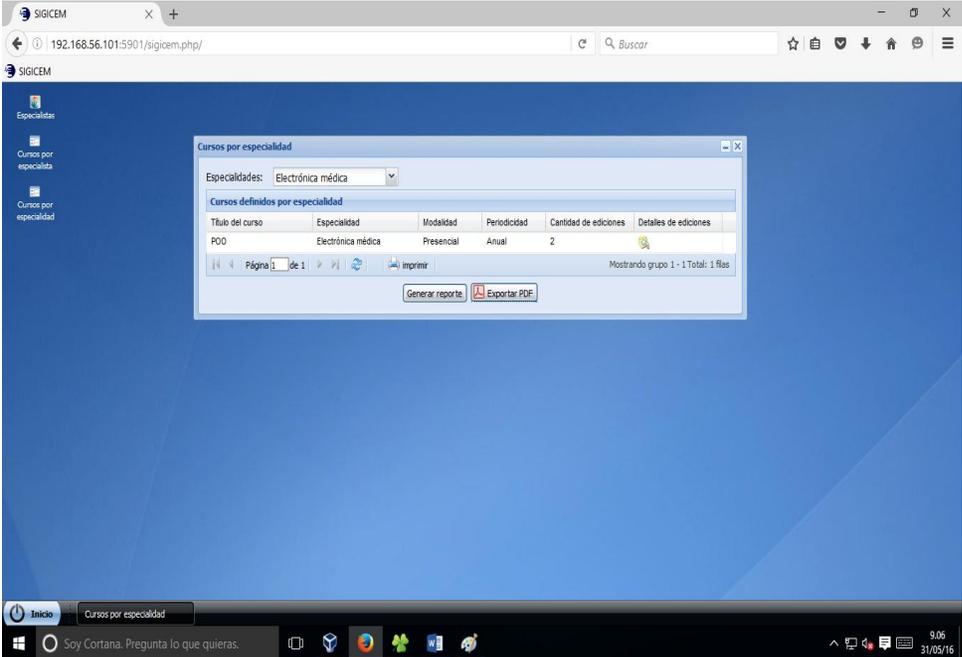
ANEXO #14. HU Mostrar reporte listado de participante por curso

Historia de usuario	
Número: HU 14	Nombre del requisito: Mostrar reporte listado de participantes por curso.
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 3 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 2 días
Descripción: Se visualiza una ventana con los distintos reportes que el usuario puede realizar, para conocer listado de participantes por curso., selecciona este tipo de reporte y el curso del que requiere la información. El sistema devuelve la información solicitada, esta se puede exportar en los siguientes ficheros(pdf) y permite que se imprima.	
Observaciones:	
Prototipo de interfaz:	

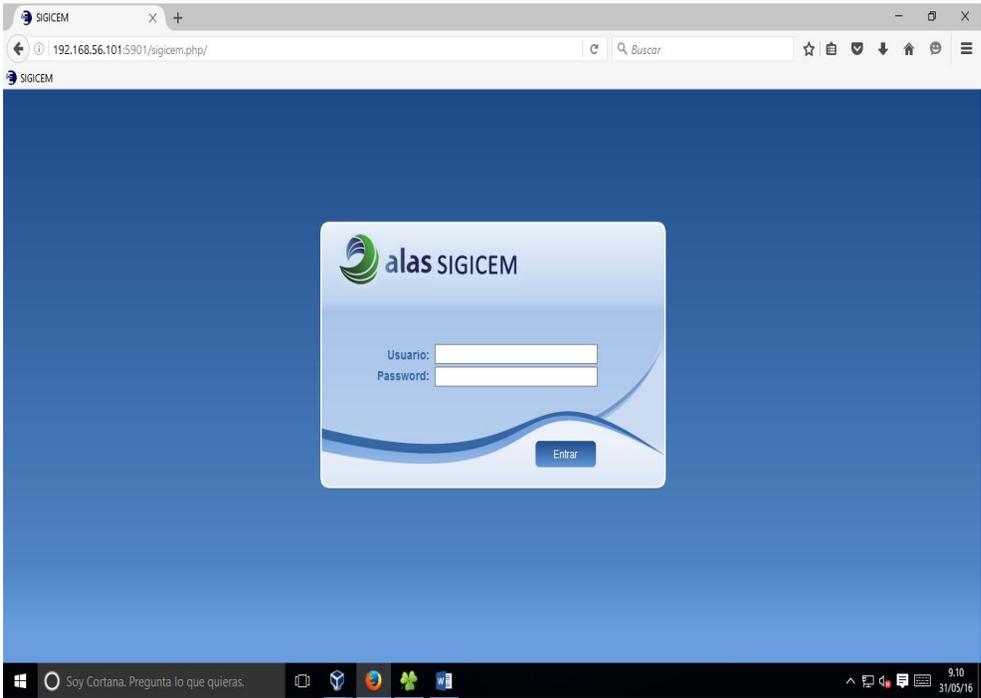
ANEXO #15. HU Mostrar reporte profesores

Historia de usuario	
Número: HU 15	Nombre del requisito: Mostrar reporte profesores
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 3 días
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 2 días
Descripción: Se visualiza una ventana con los distintos reportes que el usuario puede realizar, para conocer listado de Profesores, selecciona este tipo de reporte y el sistema devuelve la información solicitada, esta se puede exportar en los siguientes ficheros(pdf) y permite que se imprima.	
Observaciones:	
Prototipo de interfaz:	

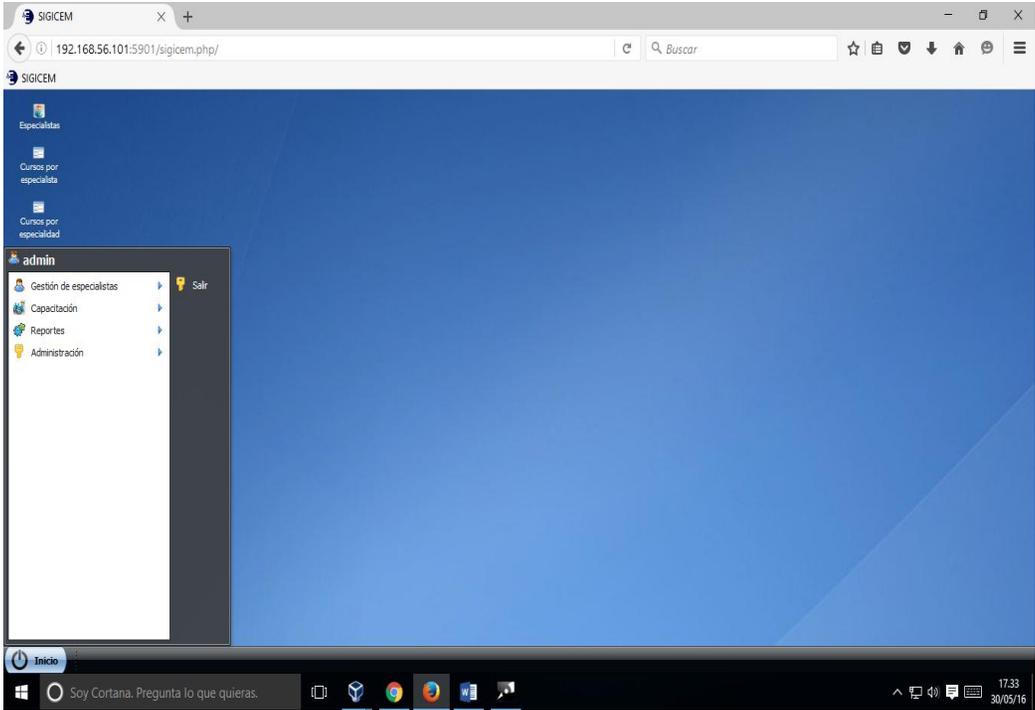
ANEXO #16. HU Mostrar reporte curso por especialidad

Historia de usuario													
Número: HU 16	Nombre del requisito: Mostrar reporte curso por especialidad.												
Programador: Roberto Cabello	Iteración: 1												
Prioridad: Media	Tiempo estimado: 3 días												
Riesgo en desarrollo: <ul style="list-style-type: none"> ○ Problemas eléctricos. ○ Problemas técnicos. 	Tiempo real: 2 días												
Descripción: Se visualiza una ventana con los distintos reportes que el usuario puede realizar, para conocer listado de Curso por especialidad, selecciona este tipo de reporte y la especialidad de la que se requiere la información. El sistema devuelve la información solicitada, esta se puede exportar en los siguientes ficheros(pdf) y permite que se imprima.													
Observaciones:													
Prototipo de interfaz:	 <p>The screenshot shows a web browser window with the URL 192.168.56.101:5901/sigicem.php/. The main content area displays a window titled 'Cursos por especialidad'. Inside this window, there is a dropdown menu for 'Especialidades' set to 'Electrónica médica'. Below this is a table titled 'Cursos definidos por especialidad' with the following data:</p> <table border="1"> <thead> <tr> <th>Título del curso</th> <th>Especialidad</th> <th>Modalidad</th> <th>Periodicidad</th> <th>Cantidad de ediciones</th> <th>Detalles de ediciones</th> </tr> </thead> <tbody> <tr> <td>POO</td> <td>Electrónica médica</td> <td>Presencial</td> <td>Anual</td> <td>2</td> <td></td> </tr> </tbody> </table> <p>At the bottom of the window, there are navigation controls (Page 1 of 1), an 'Imprimir' button, and buttons for 'Generar reporte' and 'Exportar PDF'. The browser's taskbar at the bottom shows the 'Inicio' button and the active window 'Cursos por especialidad'.</p>	Título del curso	Especialidad	Modalidad	Periodicidad	Cantidad de ediciones	Detalles de ediciones	POO	Electrónica médica	Presencial	Anual	2	
Título del curso	Especialidad	Modalidad	Periodicidad	Cantidad de ediciones	Detalles de ediciones								
POO	Electrónica médica	Presencial	Anual	2									

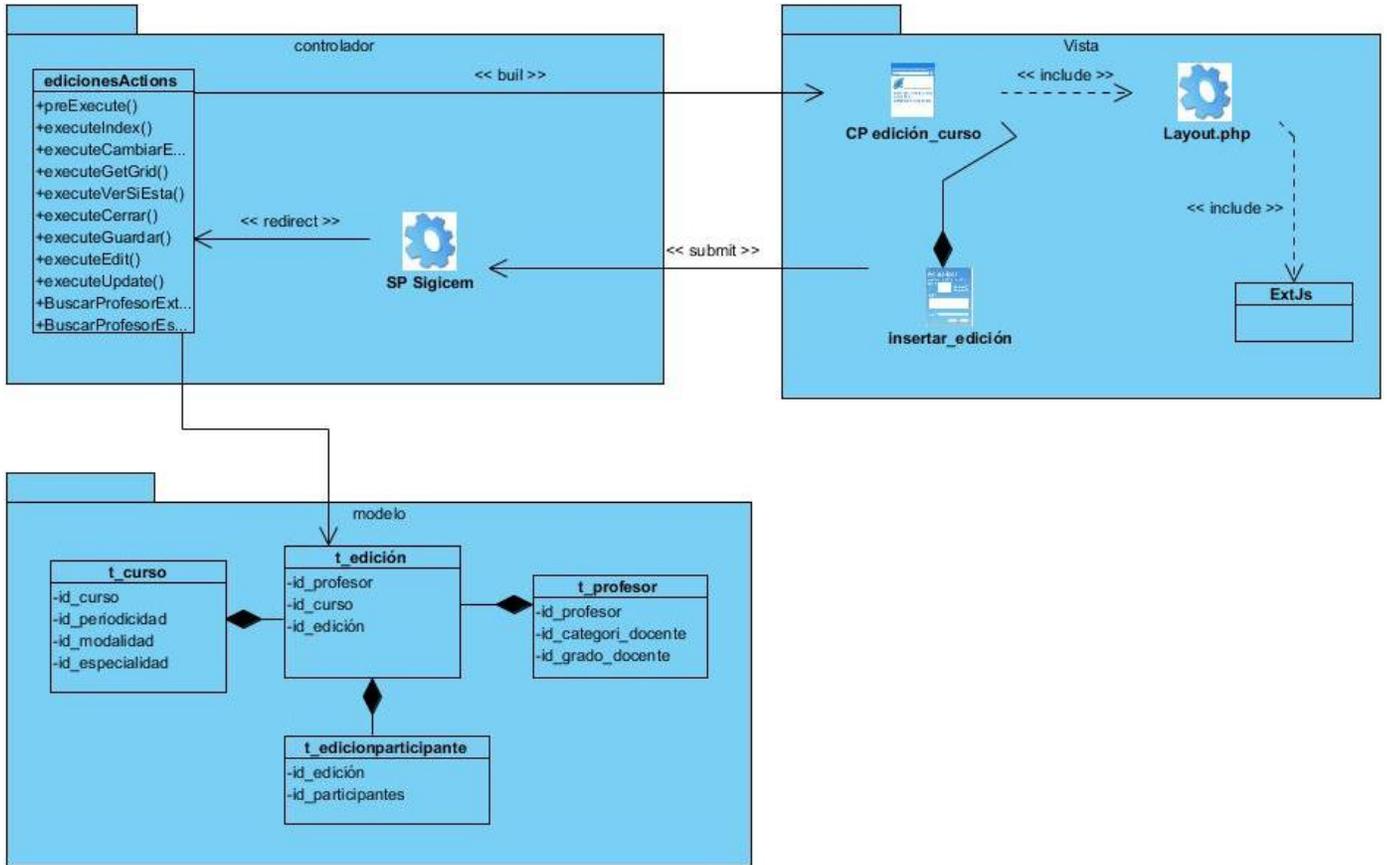
ANEXO #17. HU Autenticar

Historia de usuario	
Número: HU 17	Nombre del requisito: Autenticar
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 3días
Riesgo en desarrollo: <ul style="list-style-type: none">○ Problemas eléctricos.○ Problemas técnicos.	Tiempo real: 2 días
Descripción: Para autenticarse en el sistema es necesario usuario y contraseña, si los datos son correctos se visualiza una sección con privilegios asignados según el rol del usuario. Si no muestra un mensaje "Usuario o contraseña incorrecta".	
Observaciones:	
Prototipo de interfaz:	

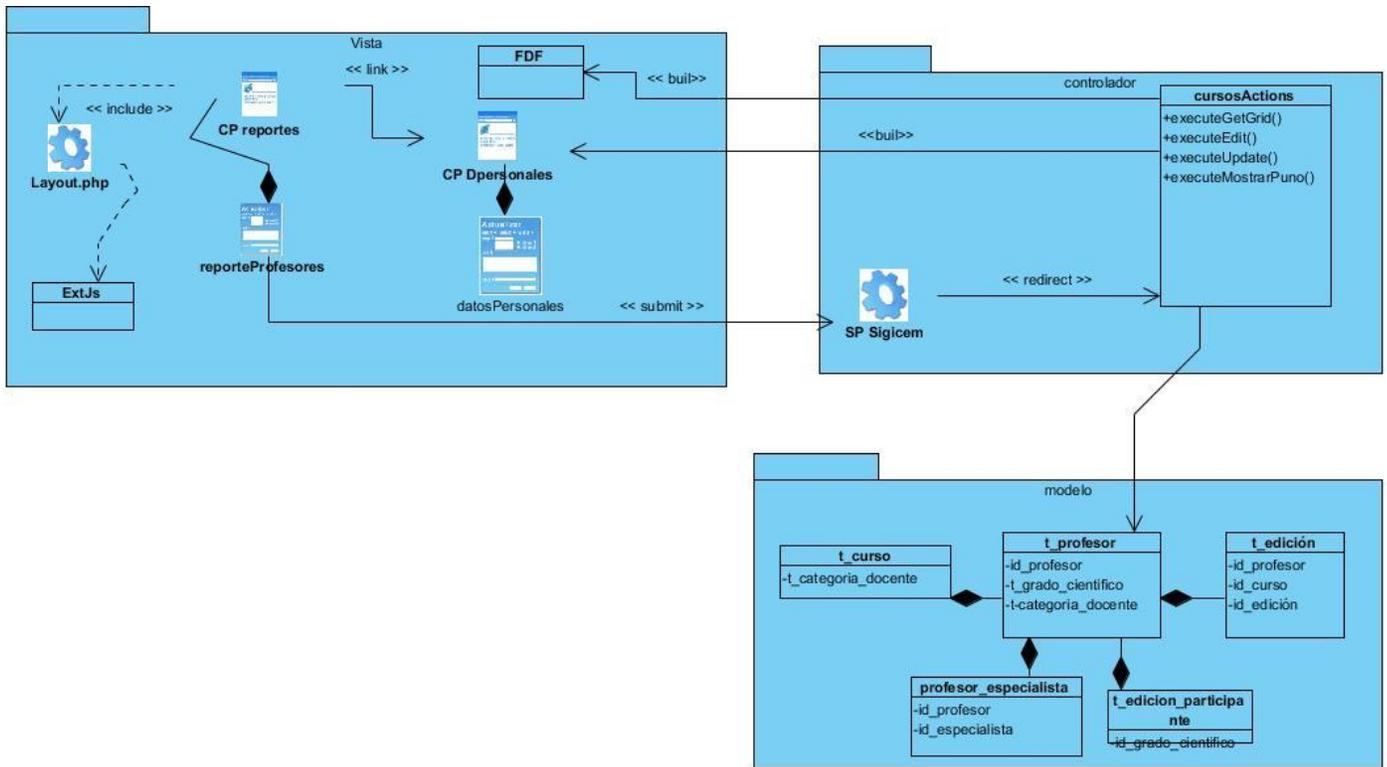
ANEXO #18. HU Cerrar sesión

Historia de usuario	
Número: HU 18	Nombre del requisito: Cerrar sesión
Programador: Roberto Cabello	Iteración: 1
Prioridad: Media	Tiempo estimado: 4 días
Riesgo en desarrollo: <ul style="list-style-type: none">○ Problemas eléctricos.○ Problemas técnicos.	Tiempo real: 3 días
Descripción: En la interfaz de cada usuario tendrá la opción de cerrar la sesión en el momento en que el usuario lo determine o cerrará automáticamente después de un tiempo previamente definido	
Observaciones:	
Prototipo de interfaz	
	

ANEXO #19 Diagrama de clases insertar edición



ANEXO #20. Diagrama de clases reporte de profesores



GLOSARIO DE TÉRMINOS

CASE (Computer Aided Software Engineering / Ingeniería de Software Asistida por Computadora): conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases.

W3C (Word Wide Web Consortium): un consorcio fundado en 1994 para dirigir a la Web hacia su pleno potencial mediante el desarrollo de protocolos comunes que promuevan su evolución y aseguren su interoperabilidad. Está dirigida por Tim Berners-Lee, el creador original de URL (Uniform Resource Locator, Localizador Uniforme de Recursos), HTTP (HyperText Transfer Protocol, Protocolo de Transferencia de Hipertexto) y HTML (Lenguaje de Marcado de Hipertexto) que son las principales tecnologías sobre las que se basa la Web.

UML (Lenguaje Unificado de Modelado): es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

SGBD: Sistema Gestor de Base de Datos.

PostgreSQL: es un SGBD objeto-relacional basado en el proyecto POSTGRES creado por el Departamento de Investigación de la Universidad de California en Berkeley, está distribuido bajo la licencia BSD y con su código fuente disponible libremente; es decir, es libre para utilizar, copiar, modificar y distribuir, sin importar para los fines que se utilice.

HTML (HyperText Markup Language o Lenguaje de marcado de hipertexto): Lenguaje de marcado utilizado para describir la estructura y el contenido de una página web.

IDE (Integrated Development Environment o Entorno integrado de desarrollo): Aplicación compuesta por un conjunto de herramientas útiles para un programador a la hora de desarrollar un software.

Módulo: Porción funcional del software, que compone a un sistema informático. Se le denomina al conjunto de elementos fundamentales que componen el producto.

PHP (Hypertext Pre-processor): lenguaje de programación, diseñado originalmente para la creación de páginas web dinámicas. Es utilizado fundamentalmente como lenguaje interpretado del lado del servidor.

AUP (Agile Unified Process o en español Proceso Unificado Ágil): es una metodología de desarrollo de software creada por Scott Ambler que representa una versión simplificada del Proceso Unificado de Rational (RUP).

TIC (Tecnologías de la Información y las Comunicaciones): agrupan los elementos y las técnicas utilizadas en el tratamiento y la transmisión de las informaciones, principalmente de informática, internet y telecomunicaciones