

Trabajo de Diploma



Universidad de las Ciencias Informáticas *Facultad 9*

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

Título: Sistema automatizado para la gestión de la información
generada en el proceso de evaluación a los proyectos de Software de
la Universidad de las Ciencias Informáticas (SGIGPE).

Autores: Eleanne Magnolia Guilbeaux Hernández.

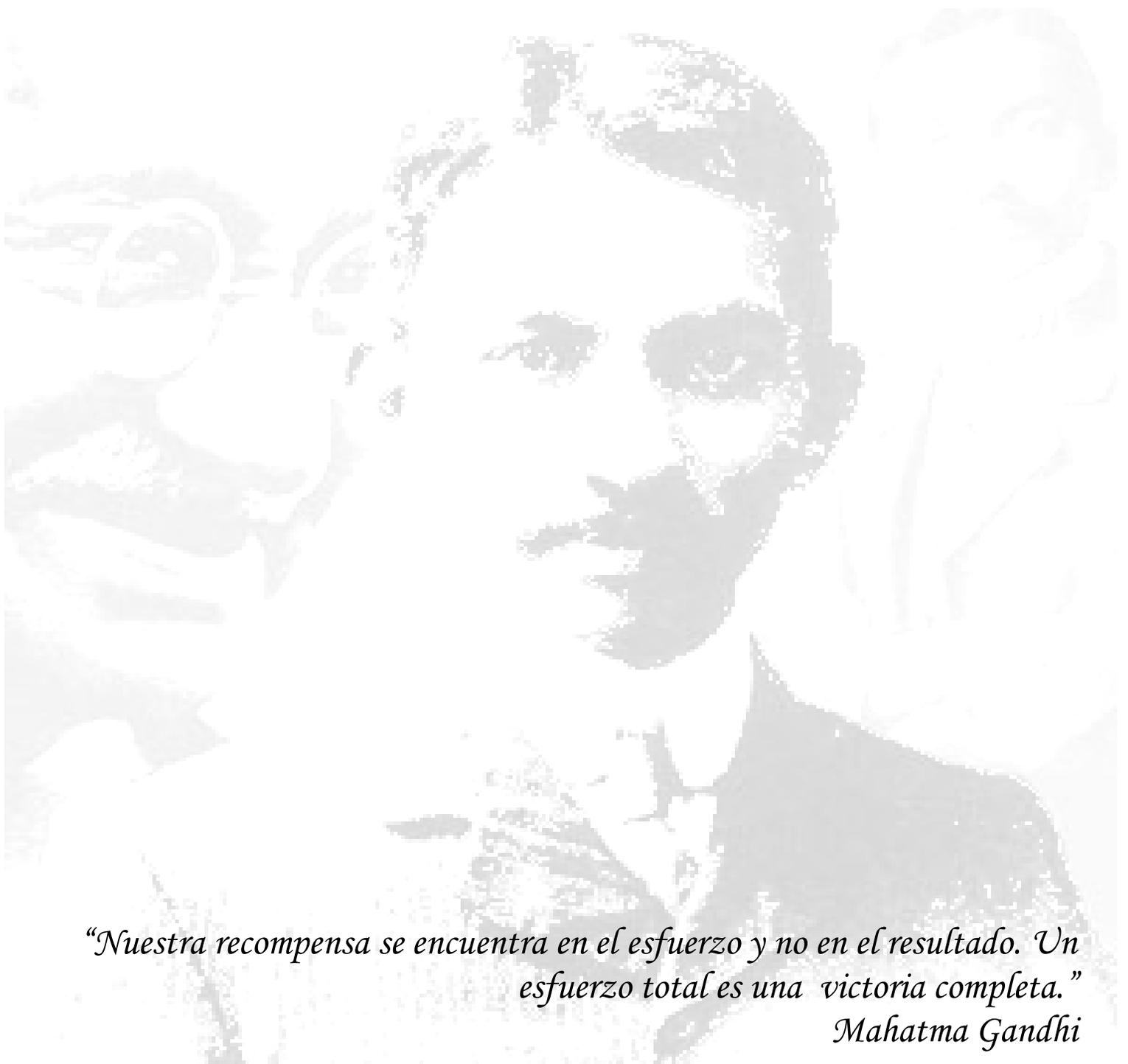
Reinier Villavicencio de la Rosa.

Tutora: Ing. Indira Lilled Laurencio Fuentes.

Co-Tutora: Ing. Olga Lidia Rodríguez Martínez.

Ciudad de la Habana, julio 2009.

"Año del 50 Aniversario del Triunfo de la Revolución."



*“Nuestra recompensa se encuentra en el esfuerzo y no en el resultado. Un
esfuerzo total es una victoria completa.”*

Mahatma Gandhi

Dedicatoria

Le dedico el presente trabajo de diploma primeramente a mi mamá, por haber sido madre, padre, hermana, amiga, confidente y por el apoyo y confianza que siempre me ha brindado.

A mi tía, que es como mi segunda madre, a mi prima Susana que es la hermana que nunca tuve, a mi primo Raulito y a la personita más especial que tengo en el mundo y que es el amor de mi vida, Sotiris.

Eleanne.

A mis Padres, por su amor y confianza.

A mi madre, porque sencillamente se lo debo todo en esta vida, la mejor del mundo.

A mi padre, por sus consejos, por apoyarme y comprenderme siempre.

A mi padrastro, por siempre confiar en mí y entenderme en cada momento de mi vida.

A mis abuelos y en especial a Yaya, alguien que dio lo mejor de sí para lograr que yo fuera lo que hoy soy, para ti va este regalo, mi título es tuyo también.

A mis Tías y Tío

A mi familia sin excepción de nadie, todos son magníficos, orgullo estoy de todos ustedes.

A todos aquellos que sienten amor hacia mí, y me estiman, gracias por haber confiado en mí durante todos estos años, aquí tienen mi primer regalo, todo fue hecho pensando en ustedes.

Reinier.

Agradecimientos

Quiero agradecerle a Dios en primer lugar por haberme dado la oportunidad de ver este sueño realidad, a Reinier por el sacrificio y por ser el mejor compañero de tesis, a la cotutora Olquita por estar siempre presente, por la ayuda, por los consejos, por guiarnos en todo el desarrollo del trabajo, a nuestra tutora Indira por las críticas constructivas y el apoyo que nos brindó, a Yelenys por la ayuda incondicional, por habernos dado siempre la oportunidad de mejorar, por asistirnos en todo momento. A mi familia, a mi gran amiga Yari y a Glendys (la negrita) gracias por soportarme y por estar siempre a mi lado aunque sepa que es difícil, a mis compañeros de cinco años, Maggie, Ivis, Mayro, Anibal, Franki, Yordanys, Angelito, Gálvez, Pimi, Ale, Eriot, que hicieron amena mi estancia en la universidad y que me ayudaron en todo momento de la carrera. Al teacher (Jorge Emilio) por la constante preocupación que tiene por nosotros. A todos mis nuevos compañeros de grupo y a mis amistades, Inés, Yeslenys, Henry, Cinolquis, el Yuma, Jean, a todos en general gracias.

Élleanne.

Agradecer es querer abarcar con palabras las inmensidades del corazón...Es un acto de nobleza, de reconocimiento sempiterno, de alabanza al amigo entrañable, a la madre abnegada, a la pareja entregada, a la primera escuela, a la universidad forjadora del futuro... A la Universidad de las Ciencias Informáticas por habernos arraigado en su seno para convertirnos en lo que somos.

Este trabajo no es el producto de la labor individual, sino más bien el resultado del esfuerzo de muchas personas durante los años de formación; para ellos nuestro recuerdo y agradecimiento, en especial:

A mi mamá, mi papá, mi hermano, mis abuelos Yaya, El Chino, Silvia, Verto y el pelegrino (mi padrastro) como siempre le digo, a todos les agradezco por darme en todo momento su apoyo incondicional durante mi vida, mi carrera, y en la confección de este trabajo.

Agradecimientos

A mi amigo, y hermano que siempre me ha apoyado y me ha ayudado desde mis primeros pasos por la universidad hasta lograr lo que hoy en día soy, ese que nunca me dijo que no cuando más lo necesitaba, ese que siempre me dio su confianza y siempre me quiso ayudar, ese es Rolando Toledo Fernández, siempre estaré agradecido de haber compartido estos cinco años de mi carrera sabiendo que siempre pude contar con alguien en las buenas o en las malas.

*A mi amiga o mejor dicho a mi hermanita ele quien me acompañó durante todo el desarrollo del trabajo y que dio lo mejor de sí, para lograr hacer mucho en tan poco tiempo, a pesar de ponerse bravita porque increíblemente se me había olvidado ponerla en la parte más linda del trabajo, para ti mis más gratos sentimientos, y siempre ten presente que me acordaré de ti en todo momento, suerte en tu vida y te deseo lo mejor del mundo, cuídate chuli, sin más **EL VILL@**.*

Al figuri (Yudiel) por el tiempo que me dedicó para poder realizar la confección de este trabajo y los consejos que me brindó durante la realización del mismo.

A mis amigos que han compartido todos estos duros años, Maikel Alejo, Leandris, Helimay, Héctor, El Yusde, Raúl, Arturo, Aíran, El Yuti (Yusel) y todos aquellos que en algún momento de su vida por la UCI hemos compartido momentos fáciles y difíciles.

A mi casi tutora Olguita, por su constante dedicación, apoyo y preocupación durante el desarrollo de la tesis y a mi tutora Indira por su ayuda y existencia de forma incondicional.

Agradezco a todas las personas que me han brindado su apoyo, confianza, cariño, comprensión y han aportado su modesto grano de arena durante el desarrollo de este trabajo, a todas aquellas que en alguna ocasión se preocuparon por mí y preguntaron ¿Cómo te va en la tesis?

A todos, simplemente gracias por ser mi apoyo.

Reinier.

El trabajo de diploma Sistema automatizado para la gestión de la información generada en el proceso de evaluación a los proyectos de Software de la Universidad de las Ciencias Informáticas de los diplomantes Eleanne Magnolia Guilbeaux Hernández y Reinier Villavicencio de la Rosa significa sin dudas un paso de avance y mejora en el ámbito del manejo de información en la Dirección de Calidad de Software. El mismo será integrado a otros dos módulos que se han realizado por el Grupo de Auditoría y Revisiones y facilitará la recogida de datos resultante en el proceso de evaluación a los proyectos productivos dentro de la UCI. Además permitirá que información estadística necesaria esté disponible de en todo momento. Igualmente este sistema humanizará todo el proceso de recogida de datos así como la generación de reportes que se traduce en ahorro de tiempo y aumento de la eficiencia de los equipos de trabajo así como la disminución sustancial de errores introducidos por las personas. Por tales motivos se considera de gran relevancia la implementación de este sistema informático.

Olga Lidia Rodríguez Martínez

Especialista de la Dirección de Calidad de Software

Resumen

Actualmente la informática constituye un factor muy importante para el desarrollo económico y social de todos los países. La Universidad de las Ciencias Informáticas (en lo adelante UCI), es la institución para la producción de software más joven de Cuba, donde se aplican las técnicas más avanzadas en la construcción de los mismos.

La presente investigación se realizó a partir de las Auditorías, Revisiones y Diagnósticos, que conforman el proceso de evaluación que se realiza en la universidad. También se estudiaron las metodologías, herramientas y lenguajes existentes posibles a utilizar, se mencionaron los conceptos relacionados al proceso, se modela y diseña el sistema y se dejan algunas recomendaciones para el mejoramiento futuro del mismo, todo esto con el objetivo de implementar un sistema eficiente y con la mejor calidad.

En la Dirección de Calidad de Software de la universidad se afrontan problemas con el manejo de la información resultante del proceso de evaluación que se lleva a cabo en cada uno de los proyectos productivos. El Sistema de Gestión de la Información Generada en el Proceso de Evaluación a los proyectos productivos de la universidad surgió a partir de la necesidad de facilitar y ayudar en la gestión de dicha información relacionada con las evaluaciones realizadas y con los resultados obtenidos en las mismas, que serían las no conformidades.

Es de especial interés para los especialistas de Calidad UCI que se lleve a cabo la implementación del sistema, porque lograría humanizar las labores de planificación y la obtención de partes y resúmenes sobre las actividades de auditorías y revisiones, además de que permitiría una mejor gestión de la información con lo cual, se podrá tener un mejor control de los procesos que se llevan a cabo en la universidad.

Palabras claves:

Procesos, Evaluación, Calidad, Sistema, Gestión, Información.

Introducción.....	1
Capítulo 1: Fundamentación teórica	5
1.1 Introducción	5
1.2 Conceptos asociados al problema.	5
Sistema informático:.....	5
Calidad.....	5
Auditoría:	6
Revisión:	6
Diagnóstico:	7
Proceso de evaluación General:	7
Aseguramiento de la Calidad de Software:	7
No conformidades:	7
1.3 Proceso de evaluación en general	8
1.3.1 Auditorías:.....	8
Objetivos Específicos y el Alcance.....	9
Principios:	9
Fases:.....	10
Tipos:.....	11
Auditorías Externas	11
Auditorías Internas	11
1.3.2 Revisiones	12
Objetivos y Alcance:	12
Principios:	13
Etapas:	13
Fases:.....	13
Tipos:.....	14
1.3.3 Diagnósticos	15
Etapas:	16
Fases:.....	16
1.4 Descripción de la Situación Problemática	19

1.5 Análisis de soluciones existentes.....	22
1.6 Conclusiones Parciales.....	23
Capítulo 2. Arquitectura del Sistema.....	25
2.1 Introducción.....	25
2.2 Metodología RUP.....	25
2.3 Lenguaje de Modelado UML.....	26
2.4 Herramienta CASE Visual Paradigm 3.4.....	26
2.5 Framework para el desarrollo Symfony 1.1.7.....	26
2.6 Gestor de base de datos PostgreSQL 8.2.....	27
2.7 Lenguaje de programación del lado del servidor PHP 5.....	27
2.8 Servidor HTTP Apache 2.2.....	28
2.9 Patrones.....	28
2.9.1 Patrones de Casos de uso:.....	28
2.9.2 Patrones de Diseño:.....	29
2.9.3 Patrones GOF (Gang of Four, en inglés).....	29
Patrones GOF que implementa el framework Symfony.....	30
Patrón Modelo Vista Controlador.....	30
2.10 Conclusiones Parciales.....	31
Capítulo 3: Características del sistema.....	32
3.1 Introducción.....	32
3.2 Modelado del negocio.....	32
Reglas del negocio.....	32
Actores del negocio.....	33
Trabajadores del negocio.....	33
Diagrama de Casos de Uso del Negocio.....	34
Casos de Uso del Negocio.....	34
3.3 Requerimientos del Sistema.....	34
Requisitos funcionales.....	34
Requisitos no funcionales.....	35
3.4 Modelado del Sistema.....	37

Descripción de la solución propuesta.....	37
Actores del sistema.....	38
Diagrama de Casos de Uso del sistema.....	38
Descripción de los Casos de uso del sistema.....	39
3.5 Conclusiones Parciales.....	43
Capítulo 4: Diseño del Sistema.....	44
4.1 Introducción.....	44
4.2 Diseño de Sistema.....	44
Extensiones para diseño web.....	44
Diagramas de clases del Diseño.....	45
Vista Lógica del Diseño.....	45
4.3 Diagrama Entidad Relación de la BD.....	46
Módulo Evaluaciones.....	49
Módulo Administración.....	50
4.4 Aplicación Patrones de Diseño.....	50
4.5 Implementación.....	51
4.5.1 Diagrama de Despliegue.....	51
4.5.2 Diagrama de Componentes.....	52
4.6 Pruebas.....	52
4.6.1 Prueba de Caja negra.....	53
4.7 Conclusiones.....	54
Conclusiones Generales.....	55
Recomendaciones.....	56
Anexos.....	59
Anexo 1: Entrevista a realizar por los especialistas de Calidad UCI.....	59
Anexo 2: Descripción del CUN Solicitar Datos de Evaluación.....	60
Anexo 3: Descripción del CUN Planificar Evaluaciones.....	62
Anexo 4: Descripción del CUN Ejecutar Evaluaciones.....	63
Anexo 5: Diagrama de actividades del CUN Solicitar Datos de Evaluación.....	65
Anexo 6: Modelo de objeto del CUN Solicitar Datos de Evaluación.....	66

Anexo 7: Diagrama de actividades del CUN Planificar Evaluaciones	66
Anexo 8: Modelo de Objeto del CUN Planificar Evaluaciones	67
Anexo 10: Modelo de Objeto del CUN Ejecutar Evaluaciones	69
Anexo 11: Flujo Normal de Eventos del CUS Autenticar Usuario.	69
Anexo 12: Flujo Normal de Eventos del CUS_ Gestionar Usuario.	70
Anexo 13: Flujo Normal de Eventos del CUS Gestionar Evaluación	72
Anexo 14: Flujo Normal de Eventos del CUS Gestionar No Conformidad.....	76
Anexo 15: Flujo Normal de Eventos del CUS Gestionar evaluador.	79
Anexo 18: Diagrama de clases del diseño CU Gestionar Evaluación.....	84
Anexo 19: Diagrama de clases del diseño CU Gestionar No conformidades.	85
Anexo 20: Diagrama de clases del diseño CU Gestionar Evaluador.	86
Anexo 21: Diagrama de clases del diseño CU Reportes de No conformidades.	87
Anexo 22: Diagrama de clases del diseño CU Reportes Generales.....	88
Anexo 23: Descripción de las clases de la Base de Datos.	89
Anexo 24: Relación de tablas de la Base de Datos.....	90
Anexo 25: Descripción de la Base de Datos.	92
Anexo 26: Caso de Prueba del CU_Autenticar usuario.....	103
Anexo 27: Caso de Prueba del CU_Gestionar usuario.	105
Anexo 28: Caso de Prueba del CU_Gestionar evaluación.	106
Anexo 29: Caso de Prueba del CU_Gestionar evaluador.....	110
Anexo 30: Caso de Prueba del CU_Gestionar no conformidad.....	112

Índice de Tablas y Figuras

Figuras

Figura 1: Tipos de auditorías (InformáticasUCI, 2008).....	11
Figura 2: Tipos de revisiones (InformáticasUCI, 2008).....	15
Figura 3: Fase 1 del proceso de diagnóstico (Galafet & Martínez Amador, 2008).....	17
Figura 4: Fase 2 del proceso de diagnóstico (Galafet & Martínez Amador, 2008).....	18
Figura 5: Proceso de diagnóstico. (Galafet & Martínez Amador, 2008).....	19
Figura 6: Representación de la situación problemática.....	20
Figura 7: Funcionamiento del patrón MVC.....	31
Figura 8: Diagrama de CUN.	34
Figura 9: Diagrama de Casos de Usos del sistema	39
Figura 10: Vista Lógica del diseño	46
Figura 12: Módulo de Información al Usuario.....	48
Figura 13: Módulo de Evaluaciones.....	49
Figura 14: Módulo de Administración.....	50
Figura 15: Diagrama General de Despliegue del sistema.	51
Figura 16: Diagrama de Componentes	52

Tablas

Tabla 1: Actores del negocio	33
Tabla 2: Trabajadores del Negocio	33
Tabla 3: Actores del sistema.....	38
Tabla 4: Descripción del CUS_Autenticar Usuario.	40
Tabla 5: Descripción del CUS_Gestionar Usuario.....	40
Tabla 6: Descripción del CUS_Gestionar Evaluaciones.....	40
Tabla 7: Descripción del CUS_Gestionar No Conformidad.....	41
Tabla 8: Descripción del CUS_Gestionar Evaluador.....	41
Tabla 9: Descripción del CUS_Consultar No Conformidades.	42
Tabla 10: Descripción del CUS_Consultar auditorías respecto al plan anual.	42
Tabla 11: Tablas de la Base de Datos	47
Tabla 12: Resumen de la Prueba de Caja Negra.....	54

Introducción

El desarrollo científico-tecnológico está enmarcado principalmente en el avance de la informática en los últimos años, jugando un papel relevante las grandes empresas productoras de software y el desempeño de las grandes industrias en función de la Ciencia Computacional.

En la actualidad, la sociedad se ha adentrado en el mundo de las tecnologías a pasos agigantados, con un nivel de transformación considerable en la esfera científica y técnica. Cuba en su empeño por alcanzar un nivel reconocido mundialmente en el marco de la informática y a pesar de su situación económica, ha tratado de elevar su desarrollo en esta rama, llegando a incluirse en el mundo de la producción de software.

Esto se debe al correcto aprovechamiento del capital humano con que cuenta la isla, en función del objetivo estratégico de la Industria Cubana del Software, que ha sido llamada a convertirse en una significativa fuente de ingresos para el país. Todas las empresas cubanas vinculadas a la rama juegan un papel importante en el desarrollo de esta industria y en la materialización de los proyectos asociados al programa cubano de informatización de la sociedad.

Desde finales del siglo XX, los sistemas informáticos se han convertido en las herramientas más poderosas para materializar e implementar uno de los conceptos vitales y necesarios en cualquier organización empresarial. Los Sistemas de Gestión de la Información al igual que los sistemas informáticos, se han tornado imprescindibles porque garantizan un mayor éxito a la hora de analizar un cúmulo importante de datos y contribuyen a elevar la eficiencia de la organización en el momento de tomar decisiones referentes a los procesos que se llevan a cabo dentro de la misma.

Hoy día la Universidad de las Ciencias Informáticas, creada al calor de la batalla de ideas, está ligada al proceso de desarrollo de productos de software no solo para la exportación, sino que además son empleados en la informatización de la sociedad cubana. Su principal objetivo es lograr un desarrollo científico y tecnológico en todas las esferas, con el fin de enriquecer los conocimientos sobre el avance de la tecnología a nivel mundial y sobre todas las cosas, poder aplicar y difundir dichos conocimientos en todo el país.

Los proyectos productivos que existen en la universidad, son sometidos a una serie de evaluaciones por la Dirección de Calidad de Software con el fin de mejorar la calidad del software, calidad se garantiza buscando la satisfacción del cliente, sus necesidades y expectativas. Estas evaluaciones pueden realizarse a través de revisiones, diagnósticos y auditorías, de las que se obtienen cierta cantidad de información importante relacionada con los proyectos, como por ejemplo, las no conformidades.

Introducción

En estos momentos el proceso de gestión de la información generada, es gestionada manualmente y no se cuentan con registros históricos actualizados de las evaluaciones realizadas, sólo con los informes finales no siempre disponible a todos los interesados, lo que trae como consecuencia que exista demora en la generación de la documentación y que sea deficiente el estudio estadístico de esta. Calidad UCI quiere desarrollar un sistema para automatizar el proceso de evaluación, por tanto el sistema de gestión de la información generada en dicho proceso, que se quiere implementar, tributaría a ese sistema. Esto se debe a que dicha información es necesaria para luego hacer una correcta toma de decisiones respecto a las no conformidades encontradas.

Al analizarse la anterior situación problemática se determinó que el **Problema** al que se enfrenta en esta investigación es: ¿Cómo mejorar la gestión de la información generada en el proceso de evaluación a los proyectos de Software de la Universidad de las Ciencias Informáticas?

Se define como **Objeto de estudio** el proceso de evaluaciones a la actividad productiva, enfocándose el **Campo de acción** en la informatización de toda la gestión de información generada en el proceso de evaluación a los proyectos productivos de software en la Universidad de las Ciencias Informáticas.

Este trabajo tiene como **Objetivo general** para dar solución al problema planteado implementar un sistema que permita gestionar la información generada en el proceso de evaluación de los proyectos productivos de software en la Universidad de las Ciencias Informáticas.

Idea a defender: Si se implementa un sistema que gestione la información obtenida en el proceso de evaluación a los proyectos productivos en la Universidad de las Ciencias Informáticas se garantizará de manera más eficiente el acceso a los datos estadísticos que se generen en este proceso.

Para dar cumplimiento al Objetivo General planteado se realizarán los siguientes **Objetivos Específicos:**

- Analizar y caracterizar el procedimiento de las evaluaciones a los proyectos productivos en la Universidad de las Ciencias Informáticas.
- Definir y modelar un sistema que permita gestionar toda la información resultante del proceso de evaluaciones en la UCI.
- Implementar un sistema que gestione la información obtenida en el proceso de evaluaciones.

Para llevar a cabo la investigación se emplean los siguientes **Métodos Científicos:**

Métodos Teóricos:

- **Analítico–Sintético:** se realiza un análisis de los diferentes procedimientos de evaluación existentes, permitiendo descubrir características generales del objeto de estudio (parte Analítica), y recopilar la información más importante (parte Sintética).
- **Análisis Histórico-Lógico:** se analizan los procedimientos de evaluación, permitiendo ver su comportamiento y evolución (parte Histórica).

Métodos Empíricos:

- **La Entrevista:** se aplica a los especialistas del equipo de Dirección de Calidad del software, con el objetivo de conocer mejor como se realizan los distintos procedimientos en la universidad (**ver anexo 1**)

Para la realización de dicha entrevista se utilizará una de las Técnicas de Muestreo No Probabilísticas, llamada Muestreo Intencional, con él se puede escoger los integrantes de la muestra, basados en las características que deben cumplir, para que aporten la mayor información posible.

La Población escogida para la aplicación de la técnica de muestreo va a ser de 5 especialistas.

Debido a que la población es muy pequeña, la muestra va a estar formada por los mismos especialistas escogidos anteriormente, que son personas que dominan el tema. Esta muestra constituye el 100% de la población.

Para realizar de manera satisfactoria el presente trabajo de diploma, se estructuró en cinco capítulos, a continuación se hace una breve descripción de los aspectos que tratarán cada uno de ellos:

CAPÍTULO 1: Fundamentación Teórica

En este capítulo se abordarán todos los elementos teóricos que ayudarán al entendimiento del problema científico y los objetivos del trabajo. Se hace una descripción general del objeto de estudio.

CAPÍTULO 2: Arquitectura del sistema.

En este capítulo se realizará un análisis y se evaluarán las herramientas, metodologías y lenguajes posibles a utilizar, seleccionando al final la más adecuada para el sistema.

CAPÍTULO 3: Características del sistema.

En este capítulo se abordará todo lo relacionado con la parte del negocio, la captura de requisitos, tanto funcionales como no funcionales que condicionan la organización por casos de uso y sus respectivas

especificaciones. El objetivo fundamental de este capítulo es garantizar la mayor comprensión, por parte de los desarrolladores y clientes, de las características del sistema a proponer.

CAPÍTULO 4: Diseño, implementación y prueba del sistema.

Este capítulo tratará todo lo relacionado con los artefactos resultantes del flujo de trabajo de diseño, así como la modelación de los diferentes casos de uso con sus respectivos diagramas de clases de diseño y las descripciones de las mismas. Además explicará toda la parte que tiene que ver con la implementación de la aplicación, es decir, todos los componentes utilizados así como los dispositivos involucrados y las pruebas de caja negra realizadas a los casos de uso más significativos.

Capítulo 1: Fundamentación teórica

1.1 Introducción

El desarrollo de las tecnologías y la aparición de la informática, ha modificado todas las ramas de la sociedad, principalmente las que generan cantidad de información, haciéndose necesario la automatización de los procesos que la gestionan, con el fin de hacerlo más rápido e independiente, menos costoso y más funcional. Este primer capítulo representa la fundamentación del objeto de estudio, definido anteriormente. Tiene como objetivo principal, abordar diferentes elementos que servirán de base teórica para el entendimiento del sistema que se propone. Donde se definirán conceptos básicos y fundamentales de los procedimientos que conforman el proceso de evaluación a los proyectos de software y se hará una breve reseña de los mismos, destacando sus características, ventajas y desventajas.

1.2 Conceptos asociados al problema.

Sistema informático:

“Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de computadoras electrónicas”. (RAE, 2006)

"Sistema es un todo integrado, aunque compuesto de estructuras diversas, interactuantes y especializadas. Cualquier sistema tiene un número de objetivos, y los pesos asignados a cada uno de ellos pueden variar ampliamente de un sistema a otro. Un sistema ejecuta una función imposible de realizar por una cualquiera de las partes individuales. La complejidad de la combinación está implícita". (IEEE, 1997)

"Sistema es una totalidad organizada, hecha de elementos solidarios que no pueden ser definidos más que los unos con relación a los otros en función de su lugar en esa totalidad". (Saussure, 1931)

De los conceptos anteriores, se considera de mayor importancia el elaborado por La Real Academia Española, porque es el que más se acerca a lo que se quiere realizar con el sistema de gestión de información SGIGPE.

Calidad

Capítulo 1: *Fundamentación teórica*

La calidad de un producto o servicio es la percepción que el cliente tiene del mismo, es una fijación mental del consumidor que asume conformidad con dicho producto o servicio y la capacidad del mismo para satisfacer sus necesidades. **(Aguilar, 1997)**

“Calidad es cumplimiento de requisitos” **(Crosby, 2001)**

“Calidad es satisfacción del cliente”. **(Deming, 1940)**

“Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” **(Pressman, 1992)**

Mencionados los conceptos, se puede decir que cada uno de los autores una manera muy particular de decir lo que es calidad, pero todos, quieren decir lo mismo y concuerdan con el concepto que tiene la universidad de lo que es la calidad.

Auditoría:

Según la Norma ISO 9000, la auditoría no es más que un:

“Proceso sistemático, independiente y documentado para obtener evidencias de las auditorías y evaluarlas de manera objetiva con el fin de determinar la extensión en que se cumplen los criterios de auditoría”.

Un examen independiente de un producto de software, proceso del software, o sistema de procesos del software para determinar conformidad con especificaciones, estándares, acuerdos contractuales, u otros criterios. **(IEEE, 1997)**

De los diferentes conceptos expuestos anteriormente, se considera que el más acertado con la investigación que se realiza es el planteado por la Norma ISO 9000, que es el que más semejanza tiene con el modo en que se realiza la auditoría en la universidad y que coincide con el objetivo que tiene esta.

Revisión:

“Las revisiones del software son un “filtro” para el proceso de ingeniería del software. Se aplican en varios momentos del desarrollo del software y sirven para detectar errores y defectos que puedan así ser eliminados. Las revisiones del software sirven para “purificar” las actividades de ingeniería del software que suceden como resultado del análisis, el diseño y la codificación.” **(Pressman, 2001)**

Un proceso o reunión durante la que un producto de software se presenta a personal del proyecto, administradores, usuarios, clientes u otras partes interesadas para comentario o aprobación. **(IEEE, 1997)**

Capítulo 1: Fundamentación teórica

Proceso o una reunión durante la cual un artefacto es presentado al equipo revisor para formular sus no conformidades o aprobación. **(Flores, Martínez & Trujillo, 2008)**

Analizados los conceptos expuestos anteriormente, se decidió tomar el elaborado por Roger Pressman, porque describe el proceso detalladamente, resaltando la importancia y las ventajas que tienen las revisiones en el proceso de evaluación a los proyectos de software.

Diagnóstico:

Proceso analítico que permite conocer la situación real de la organización en un momento dado para descubrir problemas y áreas de oportunidad.

Proceso de evaluación General:

Conjunto de una o más auditorías, revisiones y diagnósticos planificados o solicitados para un período de tiempo determinado, dirigidas hacia un propósito y una organización productiva específica. **(Calidad UCI, 2008)**

Aseguramiento de la Calidad de Software:

El aseguramiento de calidad del software es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto (software) satisfará los requisitos dados de calidad. **(Lovellette, 1999)**

No conformidades:

Son errores o defectos que presenta el producto o su equipo de realización en el proceso de desarrollo del mismo. **(CalidadUCI, 2008)**

No concordancia con normas internacionales que deben ser cumplidas por el artefacto. Incumplimiento de un requisito pactado con el cliente. **(Flores, Martínez & Trujillo, 2008)**

A partir de las definiciones vistas anteriormente, se va explicar detalladamente en qué consisten cada uno de los procedimientos que conforman el proceso de evaluación, con el fin de conocer y entender más este proceso.

1.3 Proceso de evaluación en general

Desde la década del 70, el tema de la calidad del software, ha sido motivo de preocupación para especialistas, ingenieros, investigadores y comercializadores del mismo, los cuales han realizado gran cantidad de investigaciones al respecto con dos objetivos fundamentales:

1. ¿Cómo obtener un software con calidad?
2. ¿Cómo evaluar la calidad del software?

La creciente demanda que tiene la rama de la informática, sumada a las décadas de incumplimiento en cuanto a la calidad del software y al aumento de los costos en el desarrollo de los mismos, condujeron a la creación de procedimientos con el objetivo de evaluar el proceso de desarrollo del software. Todo esto se puede percibir en los proyectos de desarrollo de software en la UCI, donde se hace necesario identificar, dar seguimiento y controlar los errores, a medida que se encuentren.

Para poder identificar y tratar los errores, como se explicó anteriormente, la universidad realiza un proceso de evaluación, el cual consiste en la realización de auditorías, revisiones y diagnósticos a cada proyecto productivo de la UCI. Este proceso controla y guía el proceso de desarrollo del software, en cada una de sus fases y mide el estado actual de los procesos de desarrollo para conocer las fortalezas, riesgos y debilidades. Es importante su realización porque así, los resultados de este, harán posible que se genere un plan de mejoramiento adecuado que permitirá aplicar las acciones correctivas donde serán más efectivas.

Este proceso tiene como principales objetivos, identificar los puntos fuertes y las debilidades para iniciar un programa de mejoramiento, constatar el progreso alcanzado por las iniciativas de mejoramiento de procesos en curso (habitualmente iniciadas como producto de una evaluación anterior), demostrar a una tercera parte (clientes, socios potenciales, corporación) la madurez de procesos de la organización (como medio para aumentar las posibilidades de nuevos negocios) y más que nada verificar la implementación de los procesos de desarrollo de software con respecto a un estándar de la industria, tal como el Modelo de Madurez de Capacidades (SW-CMM) del SEI, el ISO 9001, el ISO 15504, u otro similar. **(Guerrero, 1999- 2000)**

A continuación se explican detalladamente cada una de las evaluaciones que conforman el proceso de evaluación.

1.3.1 Auditorías:

Capítulo 1: Fundamentación teórica

Para revisar e inspeccionar estos proyectos, se utiliza la auditoría esta palabra proviene del latín auditorius, y de esta proviene la palabra auditor, que se refiere a todo aquel que tiene la virtud de oír. **(InformáticasUCI, 2008)**

En la UCI esta palabra tiene un significado muy diferente al expuesto anteriormente. Se trata de un método que tiene como principales objetivos, acumular y evaluar la evidencia, además de determinar e informar sobre el grado de correspondencia existente entre la información con que cuenta el proyecto y los criterios de evaluación establecidos en el mismo.

A partir del análisis de los resultados obtenidos en la entrevista realizada los especialistas de Calidad UCI, se dio a conocer, que actualmente esta evaluación se realizan por solicitud de la dirección de producción para evaluar el estado de un proyecto, donde puede ser auditado una vez al año. Además se verifica la conformidad con el expediente del proyecto, los Lineamientos de Calidad y el ambiente de trabajo de estos.

Además de los objetivos mencionados anteriormente, se encuentran estos:

Objetivos Específicos y el Alcance

➤ Objetivos

- Evaluar el cumplimiento de los lineamientos mínimos de Calidad.
- Verificar la conformidad con el expediente del proyecto.
- Definición y cumplimiento del cronograma y del plan de resultado.
- Contribuir a mejorar la aplicación de los elementos claves de la ingeniería de software, soporte, gestión y ambiente del proyecto.

➤ Alcance:

- Proceso. Actividad.
- Área de proceso.
- Equipo de desarrollo. **(InformáticasUCI, 2008)**

Principios:

La auditoría se caracteriza por depender de varios principios. Estos hacen de esta una herramienta eficaz y fiable, capaz de proporcionar información sobre la cual una organización puede actuar para mejorar su desempeño.

Capítulo 1: Fundamentación teórica

La auditoría sigue los siguientes principios:

- **Constructiva:** La evaluación debe estar dirigida además, a la mejora del desempeño del auditado, a partir de las no conformidades detectadas y posibles áreas de mejora identificadas.
- **Enfoque basado en la evidencia:** La evidencia de la auditoría es verificable y está basada en muestras de la información disponible. El uso apropiado del muestreo está estrechamente relacionado con la confianza que se deposita en las conclusiones de la auditoría.
- **Independencia:** Los auditores son independientes de la entidad que es auditada y están libres de riesgo y conflicto de intereses. Los auditores mantienen una actitud objetiva a lo largo del proceso de auditoría para asegurarse de que los hallazgos y conclusiones de la auditoría estén basados sólo en la evidencia de la auditoría. **(Calidad UCI, 2008)**

Además de conocer los principios por los cuales se rigen las auditorías es importante conocer las fases por las que atraviesa este procedimiento. Sus fases son:

Fases:

La auditoría para su realización consta de tres fases:

- **Fase 1 Planificación de la auditoría:** Se desarrolla el Plan trimestral de auditorías de los próximos meses, a partir de las solicitudes recibidas y tomando de referencia el programa de evaluaciones.
- **Fase 2 Inicio de la auditoría:** Se le entrega al equipo auditor los informes de auditorías previas realizadas. Se realizan las siguientes actividades:
 - Determinar objetivos de la auditoría.
 - Definir alcance de la auditoría.
 - Definir criterios de la auditoría.
 - Evaluar los riesgos de la auditoría.
- **Fase 3 Ejecución de la auditoría:** El equipo auditor realiza un encuentro preliminar para preparar la reunión de apertura donde se analizan y precisan los aspectos importantes a abordar en la reunión.
- **Fase 4 Finalización de la auditoría:** Se elabora el informe final de auditoría, haciendo una revisión exhaustiva del cumplimiento del Plan de auditoría y las conclusiones de esta.

Capítulo 1: Fundamentación teórica

- **Fase 5 Seguimiento:** Se verifica si se resolvió la no conformidad y si se implementó la acción correctiva y su eficacia. Esta verificación puede ser parte de una auditoría posterior. (Trujillo, Acosta, & Díaz, 2008)

Tipos:

Para poder realizarle una auditoría a un proyecto, es importante los tipos de auditorías para saber cuál es la más adecuada dependiendo de las características de este. A continuación se muestran los tipos que existen:

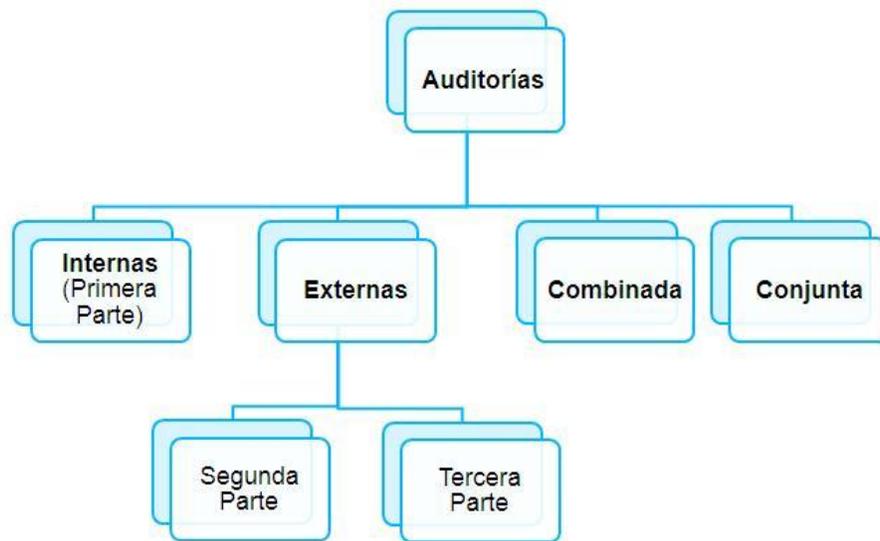


Figura 1: Tipos de auditorías (InformáticasUCI, 2008)

Auditorías Externas

- Auditorías de segundas partes: Son las auditorías realizadas por los clientes a sus proveedores.
- Auditorías de terceras partes: Son las auditorías realizadas por los organismos autorizados e independientes.

Las auditorías externas tienen como propósitos certificar y lograr el mejoramiento del Sistema de Gestión implantado y proporcionar información en el proceso de evaluación, aprobación y selección de proveedores evaluados.

Auditorías Internas

Capítulo 1: Fundamentación teórica

Auditorías de primeras partes: Las realizan la propia entidad.

Las auditorías internas tienen como propósitos establecer las normas de la Calidad, porque la auditoría constituye una herramienta para desarrollar y mejorar el Sistema de Gestión, identificar y corregir las deficiencias antes de que sean detectadas por auditores externos y medir la efectividad del sistema, proporcionando información de la marcha del proceso y corregir las deficiencias detectadas.

(InformáticasUCI, 2008)

Su aplicación trae consigo una serie de desventajas, una de ellas es genera extensa documentación, conlleva un amplio programa de capacitación, formación y conciencia de los trabajadores, necesita recursos y ejerce mucha presión sobre la persona o entidad auditada. **(CalidadUCI, 2008)**

1.3.2 Revisiones

La revisión al software es el principal procedimiento para validar la calidad de un proceso o producto, donde un grupo examina parte o todo un proceso o sistema y su documentación para encontrar problemas. También se realiza mediante reuniones, en las cuales los requisitos del software, el diseño, código y otros elementos son presentados al usuario, para ser comentados y aprobados. Esta actividad es muy importante al final de cada fase del ciclo de vida del producto desarrollado, porque gracias a ella se puede saber si el producto cumple con los requerimientos del usuario y si este se está desarrollando bien.

El procedimiento visto anteriormente tiene como principales objetivos, descubrir los defectos e inconsistencias en el sistema, indicar la necesidad de mejoras en un producto y las partes que no es necesario mejorar y por último evaluar la conformidad con estándares y especificaciones técnicas.

Y como objetivos específicos y alcance lo siguiente:

Objetivos y Alcance:

➤ **Objetivos:**

- Examinar parte o todo de un proceso o sistema y su documentación para encontrar problemas potenciales.
- Verificar la conformidad con el expediente del proyecto.
- Actuar como filtro

➤ **Alcance:**

Capítulo 1: Fundamentación teórica

- Procesos, Actividades
- Producto **(InformáticasUCI, 2008)**

Principios:

La revisión se caracteriza por depender de varios principios. Éstos hacen de la revisión una herramienta para evaluar y aprobar los artefactos resultantes de los procesos realizados en la actividad productiva sobre la cual una organización puede actuar para mejorar su desempeño. La revisión sigue los siguientes principios:

- **Constructiva:** La evaluación debe estar dirigida además, a la mejora la calidad de los artefactos y de los procesos, a partir de las no conformidades detectadas.
- **Enfoque basado en la evidencia:** La evidencia de la revisión es verificable y está basada en la información disponible.
- **Independencia:** Los revisores son independientes de la actividad que es objeto de revisión y están libres de sesgo y conflicto de intereses. Los revisores mantienen una actitud objetiva a lo largo del proceso de revisión para asegurarse de que las no conformidades estén basadas sólo en la evidencia. **(Flores, Martínez & Trujillo, 2008)**

Etapas:

En la etapa inicial se elabora y aprueba el plan mensual de revisiones, se notifica a los involucrados y se conforma el plan de la revisión, la agenda de la reunión y los materiales necesarios para llevarlo a cabo, en la segunda etapa es en la que se realiza la reunión de revisión, se determinan las no conformidades y se realizan las conclusiones y ya en la tercera y última etapa hace referencia a la elaboración del informe final de la revisión, el almacenamiento del expediente de la revisión y la evaluación del desempeño de los revisores. **(Calidad UCI, 2008)**

Fases:

Las revisiones cuentan con diferentes fases.

- **Fase 1 Inicio de la revisión:** Al inicio de cada mes, se comienza a desarrollar el Plan mensual de revisiones del próximo mes, a partir de las solicitudes recibidas y tomando de referencia los hitos de las entidades objetos de revisión.

Capítulo 1: Fundamentación teórica

- **Fase 2 Ejecución de la revisión:** Se realiza la reunión con la presencia de los miembros de la entidad objeto de revisión implicada en el elemento de revisión, responsable del revisado y cuando sea apropiado, con aquellos responsables para las funciones o procesos que se van a revisar.
- **Fase 3 Cierre de la revisión:** Se le envía el Informe final de revisión al jefe de la entidad objeto de revisión. (Flores, Martínez & Trujillo, 2008)

Tipos:

Existen diferentes tipos de revisiones y se pueden clasificar de acuerdo a los objetivos, a la forma de trabajo y a su estado.

- **De acuerdo a los objetivos se encuentran:**
 - Inspecciones para eliminar defectos (producto).
 - Revisiones para evaluar el progreso (producto y proceso).
 - Revisiones de calidad (producto y estándares).
- **De acuerdo a la forma de trabajo están:**
 - Una revisión técnica: debe centrarse sólo en el producto (no en sus desarrolladores).
 - Una revisión formal: proporciona información fiable sobre cuestiones técnicas.
- **Y por último, De acuerdo a su estado están las:**
 - Dinámicas: son las que detectan los defectos ejecutando el software.
 - Estáticas: son visuales y se realizan sin necesidad de que el software esté ejecutando.



Figura 2: Tipos de revisiones (InformáticasUCI, 2008)

A partir de la entrevista realizada los especialistas de Calidad UCI se pudo conocer que las revisiones se realizan de manera sistemática en concordancia con los planes de desarrollo de cada proyecto. Se planifican a partir de las fechas de finalización de cada fase de los proyectos, teniendo en cuenta que si el proyecto tiene una duración de un mes se le realizará una revisión, si es de tres a cuatro meses se le realiza una revisión al finalizar las fases de Inicio y de Transición, mientras que los proyectos grandes se les realiza una revisión al finalizar cada una de sus fases (Inicio, Elaboración, Construcción y Transición).

La realización de revisiones a los proyectos de la universidad, algunas veces trae como consecuencia desventajas como, que se requiere entrenamiento formal de los evaluadores, además de que cuando se realiza en forma de reunión, estas pueden ser muy largas.

1.3.3 Diagnósticos

El diagnóstico en comparación con los procedimientos expuestos anteriormente es utilizado con menos frecuencia, una o dos veces al año como máximo a los proyectos para medir el estado de la producción de la UCI. Sólo se revisan documentos rectores como el Plan de Desarrollo, Cronograma y otros.

Se caracteriza por depender de un solo principio que hace de este una herramienta eficaz y fiable, proporcionando información sobre la cual una organización puede actuar para mejorar su desempeño.

Capítulo 1: Fundamentación teórica

Este principio es el de la imparcialidad la cual es la base para lograr la objetividad del diagnóstico. Los expertos y ejecutores mantienen una actitud objetiva a lo largo del proceso de diagnóstico para asegurarse de que los resultados estarán basados sólo en la evidencia del diagnóstico.

Este procedimiento consta de tres etapas:

Etapas:

- **Etapla inicial:** se crean las bases para el desarrollo del diagnóstico y se determina si es viable su ejecución o no, para ello se define la misión, los objetivos y el alcance del mismo, se estiman los riesgos y los recursos necesarios.
- **Etapla de desarrollo:** se conforman los grupos que preparan y ejecutan las actividades necesarias para la obtención de la información que permitirá alcanzar los resultados esperados.
- **Etapla final:** se desarrolla y almacena el informe y el expediente del diagnóstico y se evalúa el desempeño del Equipo de Diagnóstico.

Fases:

- ✓ **Fase I: Inicio del diagnóstico:** Su propósito es determinar la misión, los objetivos y el alcance del diagnóstico actual y se enuncia el jefe del equipo de diagnóstico. **Ver figura 3**

Capítulo 1: Fundamentación teórica

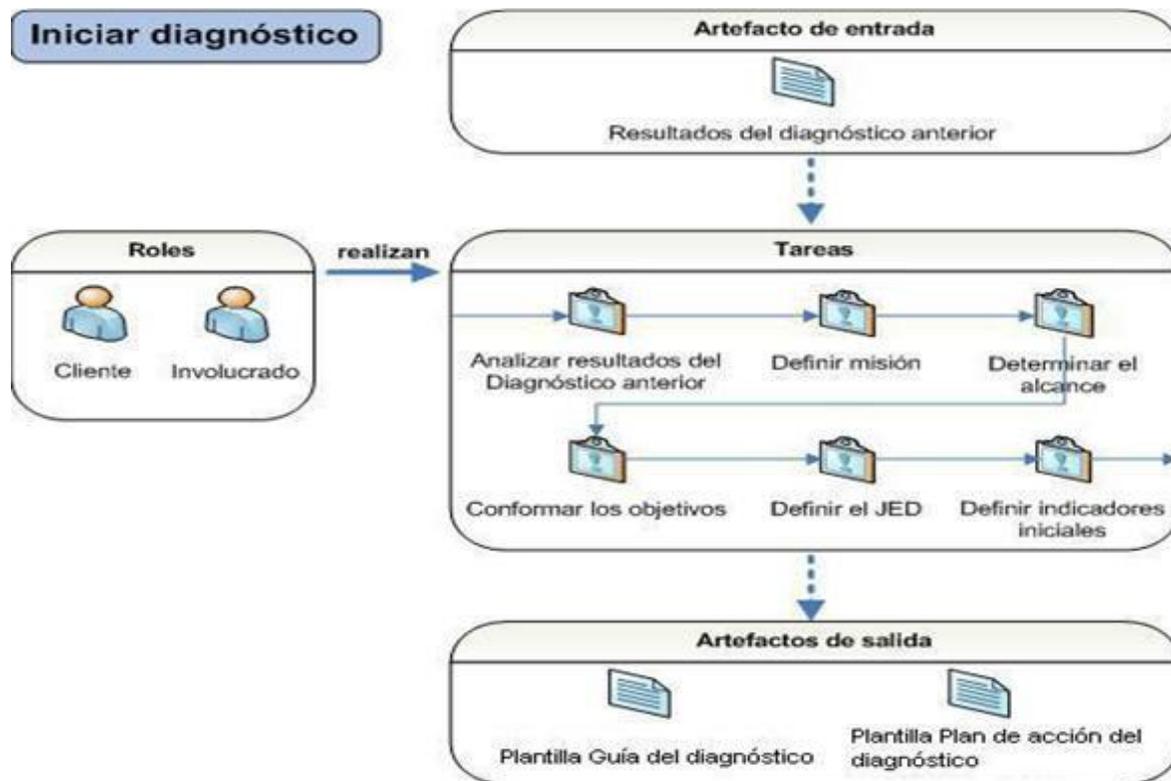


Figura 3: Fase 1 del proceso de diagnóstico (Galafet & Martínez Amador, 2008)

- ✓ **Fase II: Desarrollo del Diagnóstico:** La actividad tiene como propósito seleccionar el personal que va a formar parte del Grupo de Trabajo Técnico. Para ello se describen las competencias a considerar en la evaluación de los especialistas, esto garantiza que el personal seleccionado sea el indicado para cumplir con las tareas que deben desarrollar. **Ver figura 4**

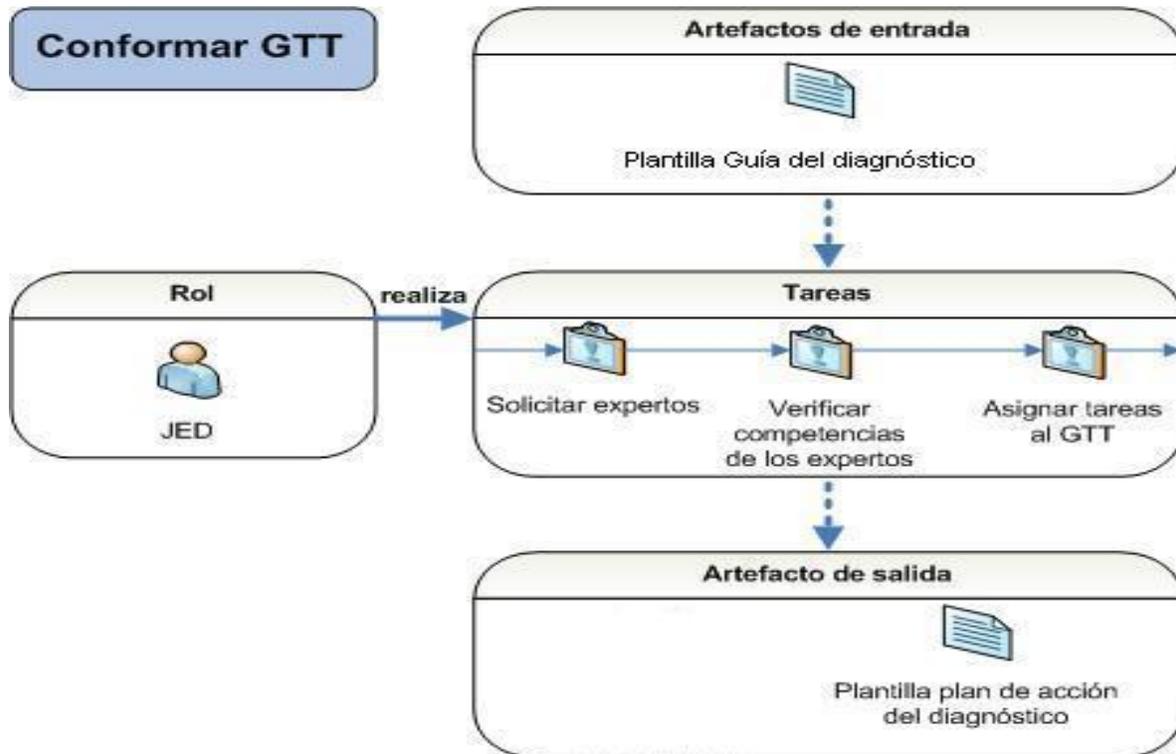


Figura 4: Fase 2 del proceso de diagnóstico (Galafet & Martínez Amador, 2008)

Los diagnósticos de acuerdo al final que tengan, son clasificados en abortados, si no hay recursos necesarios o si se determina que el diagnóstico no es viable, o ejecutados, si se ejecutan totalmente. (Galafet & Martínez Amador, 2008)

A continuación se muestra como se realiza el proceso de diagnóstico completo:

Capítulo 1: Fundamentación teórica

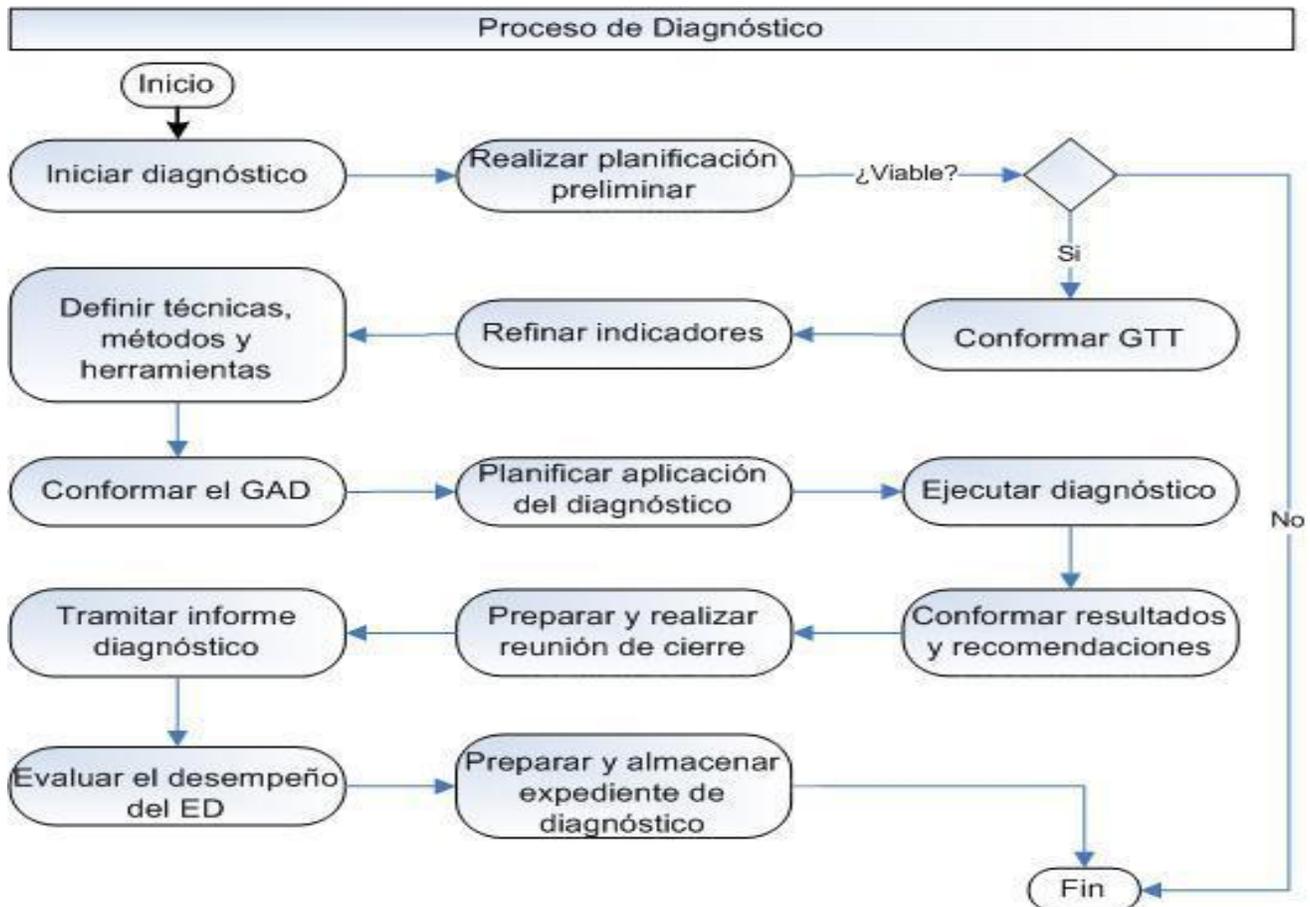


Figura 5: Proceso de diagnóstico. (Galafet & Martínez Amador, 2008)

La aplicación de este procedimiento a los proyectos es de vital importancia porque proporciona información relevante para la toma de decisiones, con respecto a los resultados obtenidos y la planeación estratégica a corto, mediano y largo plazo de las acciones correctivas. Además permite conocer la situación real de la organización en un momento dado para descubrir problemas y áreas de oportunidad.

1.4 Descripción de la Situación Problemática

Durante la etapa de desarrollo de los proyectos productivos de la Universidad de las Ciencias Informáticas, estos son sometidos a un proceso de evaluación, el cual es ejecutado por el equipo especializado de Calidad UCI. Este proceso tiene el propósito de contribuir a la mejora de los procesos que se llevan a cabo en el desarrollo del software, aumentar el rendimiento y sobre todo asegurar la calidad del software. El proceso de

Capítulo 1: Fundamentación teórica

aseguramiento de la calidad del software depende de las tareas realizadas durante todo el proceso y del control que se tengan sobre estas, ya que detectar errores en forma temprana, ahorra esfuerzos, tiempo y recursos.

Las evaluaciones se pueden realizar mediante revisiones, diagnósticos o auditorías y las tres se realizan de igual manera a todos los proyectos y tienen como objetivo en común, verificar que se cumpla con lo establecido por Calidad UCI, ya sean los lineamientos de calidad o la conformidad que tengan estos con el expediente de proyecto establecido para su uso en la producción dentro de la UCI. De este proceso de evaluación se obtiene como resultado, no conformidades además de cierta cantidad de información relacionada con los proyectos productivos y que es importante y necesaria para posteriormente hacer una toma de decisiones efectiva.



Figura 6: Representación de la situación problemática

Para cada no conformidad encontrada el líder del proyecto evaluado emite acciones correctivas, con una prioridad y con un tiempo específico de resolución. Estos aspectos permitirán a los especialistas de Calidad UCI evaluar tendencias, obtener resultados concretos y tomar decisiones que tributen a una mejora eficiente del proceso productivo en la UCI.

Capítulo 1: Fundamentación teórica

Actualmente la gestión de la información generada por las Auditorías, Revisiones y Diagnósticos efectuados a los proyectos productivos se realiza de forma manual, específicamente en informes con formato Excel, PDF y Word, no siempre disponible a todos los interesados. Esta se divide según las partes interesadas, al proyecto se le da la información que le es útil para resolver las no conformidades detectadas, a la gerencia se le informa de manera general como está el proyecto y el grupo de auditoría y revisiones es el encargado de administrar toda la información generada. Por todos los problemas expuestos anteriormente es que resulta muy difícil realizar el análisis estadístico (que no es más que la interpretación de la información la cual contribuye al estudio científico de los problemas planteados en el ámbito de la organización y que tributa a una posterior toma de decisiones y mejora de los procesos internos de la misma), el estudio de tendencia de la calidad (se realiza cuando se está haciendo el cierre trimestral, se comparan las no conformidades encontradas en el trimestre pasado con las del actual, con el objetivo de verificar si el proyecto mejoró o empeoró) y llevar un registro histórico eficiente y actualizado de los procedimientos realizados. Además cuando se va a realizar una recopilación o un resumen de las dificultades detectadas hay que ir informe por informe lo cual hace muy engorrosa esta labor, unido a esto los proyectos no pueden mantener actualizado el registro de no conformidades por lo que los jefes superiores al líder de proyecto, dígase Jefe de Polo, De Área Temática y hasta el propio Vicedecano de Producción, desconocen los resultados de la evaluación del proyecto. (Este problema ocurre principalmente con las revisiones).

Sin un sistema automatizado, el proceso de recogida de datos así como la generación de reportes y plantillas, sería difícil de realizar. Por tanto se puede decir que resulta de especial interés para la Dirección de Calidad de Software la automatización del registro de la información procedente de estos procesos porque primeramente, se ahorraría tiempo y aumentaría la eficiencia del grupo de trabajo de Auditorías y Revisiones a la hora de gestionar toda la información que se genere de este proceso de evaluación, evitando así retrasos en entrega de información, y esta se tendría organizada y controlada de una mejor forma. En segundo lugar se humanizarían las labores de planificación y la obtención de partes y resúmenes sobre las actividades de auditorías y revisiones y el intercambio de información a través del correo sería mínima, puesto que la gestión y seguimiento de los resultados de las evaluaciones se realizarían a través de la interfaz de un sistema centralizado con recursos compartidos. Por último existirá una mejor gestión de la información y un mejor control del proceso y la planificación se realizará de una mejor forma, porque hoy representa una tarea engorrosa por la inconsistencia que existe en los datos de los proyectos. Esta información se obtuvo como

resultado de la entrevista realizada a los especialistas de la Dirección de Calidad de la universidad. **(Ver anexo 1)**

1.5 Análisis de soluciones existentes.

Actualmente en el mundo existe gran demanda de procedimientos que garanticen la calidad del desarrollo del software, esto se debe a los incumplimientos en cuanto a la calidad porque la mayoría no cumple con los requisitos planteados por los usuarios. El estudio realizado acerca de los sistemas similares, permitió la obtención de información relacionada con dos sistemas, estos son, el Sistema Automatizado para el Diagnóstico de la Gestión de Información y Conocimiento en la Empresa y el Sistema de Auditoría y Control (SAC).

A continuación se hace una breve descripción de cada uno de ellos.

El Sistema Automatizado para el Diagnóstico de la Gestión de Información y Conocimiento en la Empresa.

Este sistema es una herramienta automatizada para realizar el diagnóstico o auditoría de información y conocimiento, responde a las nuevas necesidades identificadas relativas a los requerimientos de los sistemas informativos para el sector empresarial, específicamente para la empresa en perfeccionamiento. Esta facilita el desarrollo de la actividad de auditoría de la información, o su similar, del diagnóstico de la gestión o sistema de información, con la particularidad de que ya se adentra en la evaluación de los procesos de la gestión del conocimiento. Además provee una herramienta de evaluación EVAL-IC-EP, que intenta comprobar la forma en que se manejan la información y el conocimiento en la organización y el análisis de los riesgos. Del levantamiento de la información se precisa un proceso de normalización de la información y es a partir de ahí que se procesa la información para diseñar los reportes de salida que ofrecerá el sistema. De acuerdo a los intereses de la empresa cliente y de los propios consultores, se obtendrán tantos informes como sean necesarios. **(Goñi, Artilles & Nieves, 2008)**

Sistema de Auditoría y Control (SAC)

Se considera una aplicación viable de migrar a cualquier ambiente sin importar su tamaño o recursos informáticos. Provoca la estandarización de los papeles de trabajo, ayuda a controlar y supervisar los avances de las Auditorías, permite contar con una Base de Datos que puede ser explotada de acuerdo a las necesidades del personal, optimiza los tiempos tanto en el proceso de Auditoría, así como en la elaboración de

Capítulo 1: Fundamentación teórica

informes y establece mayores canales de Comunicación Interna. La implantación del sistema SAC, entre otros, permitió una reducción de tiempos y trámites administrativos, el fortalecimiento del control en las revisiones, la eliminación de un nivel de supervisión directa, la implantación de formatos y reportes estándar, el desarrollo secuencial y ordenado de las actividades, la ejecución del proceso de auditoría de manera automatizada, el registro de los tiempos utilizados en cada una de las etapas, la definición de roles y responsabilidades, así como perfiles y descripciones de puestos y la implantación de esquemas más productivos para el desarrollo de reuniones de revisión. **(S.N.C, 2002)**

Después de realizado el análisis de cada uno de los sistemas, se llegó a la conclusión de que no cumplen con las características suficientes para dar solución a las necesidades planteadas en la situación problemática, porque ninguno trata el tema de las revisiones, que actualmente es la evaluación que más se realiza en la UCI. Esto se debe a que el proceso de evaluación que se realiza en la UCI, es diferente a los procesos que se realizan en el mundo, porque se puede realizar mediante tres evaluaciones fundamentales que son las auditorías, revisiones y los diagnósticos.

En el caso del Sistema Automatizado para el Diagnóstico de la Gestión de Información y Conocimiento en la Empresa, se puede decir que es el que más características en común tiene con SGIGPE, una de ellas es que muestra reportes sobre la información obtenida, pero además de no tratar el tema de las revisiones, su principal objetivo no es gestionar la información resultante como es el caso de SGIGPE, sino evaluar la gestión de esa información. En el caso de SAC, este es un sistema que se basa solamente en la automatización del proceso de auditoría y no en la gestión de información.

1.6 Conclusiones Parciales

En este primer capítulo se abordó todo lo referente a la fundamentación teórica del objeto de estudio definido anteriormente. Para ello, primeramente se mencionaron varios conceptos elaborados por diferentes autores de cada uno de los procedimientos que conforman el Proceso de Evaluación, con el objetivo de lograr un mayor entendimiento del tema a tratar. Posteriormente se realizó un análisis detallado de la aplicación de cada uno de los procedimientos en los proyectos productivos de la universidad, basado la entrevista que se le aplicó a los especialistas de Calidad UCI. Esta arrojó como resultado una serie de elementos básicos que servirán para la posterior implementación del sistema de gestión de información SGIGPE, como son las características, fases, etapas, etc., además de las ventajas y desventajas que presentan cada uno de los procedimientos.

Capítulo 1: Fundamentación teórica

Y por último se hizo una descripción minuciosa de la situación problemática que existe en la universidad con respecto a los problemas que se presentan a la hora de manejar el cúmulo de información resultante de la aplicación de Proceso de Evaluación a los proyectos productivos de la universidad, para un mejor entendimiento y esclarecimiento de la realización del proceso.

Capítulo 2. Arquitectura del Sistema

2.1 Introducción

Como se explicó al inicio, el Sistema de Gestión de Información que se quiere implementar, favorecerá de alguna manera la realización de los procedimientos que conforman el proceso de evaluación y tributará al sistema de informatización del proceso de evaluación, que está desarrollando la Dirección de Calidad de Software de la universidad, además de que va a ser utilizado por la misma. Por tanto se definió que el sistema de gestión de información se desarrollará utilizando la misma metodología, herramientas y lenguajes, es decir que fuera compatible totalmente, con el sistema que desarrollan.

2.2 Metodología RUP

Se define entonces que a pesar de existir otras metodologías más acordes para la implementación del sistema de gestión de información, por ejemplo la metodología XP, así como también asumiendo todos los posibles problemas que puedan surgir durante las etapas de desarrollo de la aplicación, que la metodología a usar para la implementación del sistema, será RUP, debido a tres características fundamentales:

1. El hecho de que RUP se basa en el modelado a través de casos de uso, hace que desde el modelado del negocio se garantice de manera eficaz la robustez del diseño de la aplicación, contando además con el hecho de que RUP, a criterio de muchos especialistas es la metodología ideal para aplicaciones orientadas a la gestión.
2. Por ser iterativa e incremental permite perfeccionar el sistema conforme se va desarrollando, pues cada una de sus fases constituye un mini proyecto, lo que hace que se garantice la calidad del producto de manera más eficiente que cualquier otra metodología.
3. Por ser un proceso centrado en la Arquitectura, permite que el trabajo de implementación sea más fluido y acertado.

Además es el proceso de desarrollo más general de los existentes actualmente, es orientado a objeto y proporciona guías, plantillas y ejemplos para todos los aspectos y etapas del desarrollo del software. Además verifica la calidad de este y tiene la característica de que permite ver una versión del producto al terminar cada fase. Las principales ventajas que tiene es que presenta una forma disciplinada de asignar tareas y se basa todo en las mejores prácticas que se han intentado y se han probado en el campo.

2.3 Lenguaje de Modelado UML

Se definió como lenguaje de modelación a utilizar para el diseño del sistema SGIGPE, UML, porque cuenta con una rica notación gráfica y un conjunto de diagramas y elementos, que permiten la visualización, especificación, construcción y documentación de los artefactos de un sistema de software, lo que permite un mejor entendimiento del proceso de desarrollo del software por parte del usuario. Ayuda a simplificar el proceso de diseño de software, permitiendo tener diferentes vistas de un modelo, uno de los grandes méritos que tiene, es la manera que ayuda a iniciar el proceso de desarrollo con los casos de uso. Es un lenguaje de alta precisión, lo cual lo hace un estándar industrial muy utilizado, es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

2.4 Herramienta CASE Visual Paradigm 3.4

Se optó por la herramienta CASE Visual Paradigm, porque permite la construcción de software de manera eficiente y presenta una interfaz visual amigable, que permite la construcción de diagramas en UML de forma rápida y sencilla. Soporta el ciclo de vida de desarrollo de software completo: análisis y diseño orientados a objetos, construcción, pruebas y despliegue, lo cual es vital para la implementación del sistema de gestión de información SGIGPE. Además es multiplataforma y posibilita la generación de código y documentación, beneficia una rápida construcción de aplicaciones de calidad a un costo menor y suministra abundantes tutoriales de UML, así como demostraciones interactivas y proyectos. Entre sus principales características está que cuenta con un soporte que facilita el trabajo simultáneo sobre un mismo diagrama entre dos desarrolladores en un tiempo real y admite además la importación y exportación de XML e imágenes, estas fueron utilizadas en la realización de las clases del diseño del sistema de gestión de información SGIGPE, además es una herramienta libre, aspecto que la hace más fuerte y beneficiosa para la construcción del sistema de gestión de información SGIGPE.

2.5 Framework para el desarrollo Symfony 1.1.7

Se escogió este framework porque es independiente del sistema gestor de bases de datos y fácil de entender, presenta una ayuda completa, que es la llamada "Symfony la guía definitiva" la cual fue de gran ayuda para la implementación del sistema, además de que es fácil de instalar y configurar en la mayoría de las

plataformas. Actualmente es el más famoso y utilizado en el mundo, por lo sencillo de usar en la mayoría de los casos y lo suficientemente flexible como para adaptarse a los casos más complejos, sigue la mayoría de las mejores prácticas y los patrones de diseño para la web. Entre las principales ventajas que lo caracterizan está que separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web, mediante el Modelo Vista Controlador, proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja y es compatible con el gestor de base de datos usado (PostgreSQL).

Además puede ser ejecutado en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows, ventajas que no presentan otros framework como por ejemplo Kumbia y Case, que fueron tenidos en cuenta para la selección de Symfony.

2.6 Gestor de base de datos PostgreSQL 8.2

Se eligió este gestor de base de datos por las facilidades que brinda, es un servidor de base de datos relacional libre (lo cual es muy importante para la universidad y el país, que en estos momentos está tratando de migrar para software libre con el objetivo liberarse de Microsoft). Tiene la característica de que se puede modificar de acuerdo a las necesidades y si es usado en Linux, BSD o *nix es inmune a los virus. Además se puede adquirir mediante un costo bajo o nulo, los requerimientos para su instalación son mínimos y presenta gran estabilidad y confiabilidad.

En cuanto a los diferentes gestores de base de datos que fueron analizados además del antes mencionado, My SQL y SQL Server, tienen algunas desventajas en comparación con PostgreSQL, como por ejemplo ambos gestores son propietarios y en el caso de My SQL, no tiene tantas capacidades como otros gestores profesionales, porque depende de la aplicación, no tiene integridad referencial, es lento con grandes bases de datos y no es tan robusto, ni soporta transacciones, ni muchos usuarios. En el caso de SQL Server el costo es muy elevado con respecto a PostgreSQL, el tiempo de respuesta es relativamente largo y poco eficaz, y tiene algunas incomodidades visuales para trabajar.

2.7 Lenguaje de programación del lado del servidor PHP 5

PHP es el lenguaje de programación más utilizado en la realización de páginas web avanzadas. El principal objetivo de PHP5 ha sido mejorar los mecanismos de Programación Orientada a Objeto para solucionar las carencias de las anteriores versiones. Una de las principales ventajas que presenta PHP, es

Capítulo 2: Arquitectura del Sistema

su soporte para una gran cantidad de bases de datos además de poder ser utilizado en los distintos sistemas operativos ya sea en Linux, muchas variantes Unix, Windows, Mac OS X, entre otros, además de soportar la mayoría de los servidores de hoy en día, incluyendo Apache y Microsoft Internet Information Server.

2.8 Servidor HTTP Apache 2.2

Apache es la solución usada para la mayoría de los sitios Web. La versión 2.0 es una profunda revisión del servidor Apache, las principales revisiones de código se han llevado a cabo para crear una arquitectura realmente escalable, hoy en día es considerada la plataforma Web más utilizada del mundo, pues aumentan día a día el número de usuarios que aceptan este código fuente abierto en su infraestructura. Entre sus principales características se encuentran presenta una organización por módulos, es multiplataforma y extensible y popular porque es fácil de conseguir y gratuito.

2.9 Patrones

Patrón es una pareja de problema/solución con un nombre y que es aplicable a otros contextos, con una sugerencia sobre la manera de usarlo en situaciones nuevas, o sea, un patrón es una descripción de un problema bien conocido que suele incluir: descripción, escenario de uso, solución concreta, consecuencias de utilizar el patrón, ejemplos de implementación y lista de patrones relacionados. (Larman, 1999)

2.9.1 Patrones de Casos de uso:

- **El nombre revela la intención:** Es empleado para tratar de levantar interés al lector, para ello se hace uso de nombres descriptivos para los casos de uso, es decir el nombre del caso de uso debe ser lo más descriptivo posible para lograra capturar la esencia del caso de uso.
- **Completar una única meta:** Consiste en escribir cada caso de uso dirigido hacia una completa y bien definida meta.
- **Alternativas exhaustivas, íntegras:** Trata de capturar todos los fallos y alternativas que deben ser manejados en el caso de uso. Una vez se tienen identificados todos los casos de uso y su flujo principal, se identifican y capturan todas las variaciones del flujo principal que se quiera que el sistema maneje.

Capítulo 2: Arquitectura del Sistema

- **Patrón Preciso y Legible:** Este patrón es el encargado de tratar de que los casos de usos estén lo suficientemente legibles para que los clientes los lean y sepan con precisión que es lo que se describe, y es de vital importancia para los implementadores para que sepan que es lo que están construyendo.
- **El patrón “CRUD”:** Completo”: se encarga de definir un solo caso de uso que englobe a otros, se pudiera hablar de un gestionar, que tenga asociado: eliminar, modificar, buscar. Este patrón es aplicado cuando todo el flujo tributa a un mismo valor dentro del negocio y este es corto y simple.
- **Patrón Claro Conjunto de Roles:** Es el encargado de identificar los actores y el rol que cada uno juega con respecto al sistema. Este patrón se encarga de describir claramente la interacción de los actores con el sistema y viceversa.

2.9.2 Patrones de Diseño:

Patrones GRASP (Patrones Generales de Software para Asignar Responsabilidades): Describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

Los primeros cinco patrones GRASP son:

- **Experto:** se encarga de asignar una responsabilidad al experto en información, o sea, la clase con la información suficiente para cubrir la necesidad.
- **Creador:** se encarga de darle a la clase A la responsabilidad de crear objetos de la clase B. En este caso A es creador de los objetos B.
- **Alta Cohesión:** se encarga de asignar responsabilidades a las clases de modo que las funcionalidades estén estrechamente relacionadas y que cada elemento del diseño tenga una labor única. Una clase con baja cohesión se hace difícil de entender, de conservar y de reutilizar.
- **Bajo Acoplamiento:** garantiza que una clase no dependa de muchas otras ya que se hace difícil de reutilizar.
- **Controlador:** propone asignarle la responsabilidad de manejar todos los eventos del sistema a una clase.

2.9.3 Patrones GOF (Gang of Four, en inglés)

- **Patrones de creación:** presentan la guía de cómo crear objetos cuando sus creaciones requieren tomar decisiones, las cuales serán resueltas dinámicamente decidiendo que clases instanciar o sobre que objetos delegar responsabilidades.

Capítulo 2: Arquitectura del Sistema

- **Patrones estructurales:** describen la forma en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros.
- **Patrones de comportamiento:** se utilizan para organizar, manejar y combinar comportamientos.

Patrones GOF que implementa el framework Symfony

- **Singleton:** es un patrón de creación que garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia.
- **Decorator:** es un patrón estructural que añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad.

Patrón Modelo Vista Controlador

La arquitectura del patrón Modelo-Vista-Controlador (MVC) es un paradigma de programación bien conocido para el desarrollo de aplicaciones con interfaz gráfica (GUI). Es una arquitectura preparada para los cambios, que separa los datos y la lógica de negocio de la lógica de presentación, permitiendo la actualización y desarrollo independiente de cada uno de los componentes.

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por 3 niveles:

- 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.

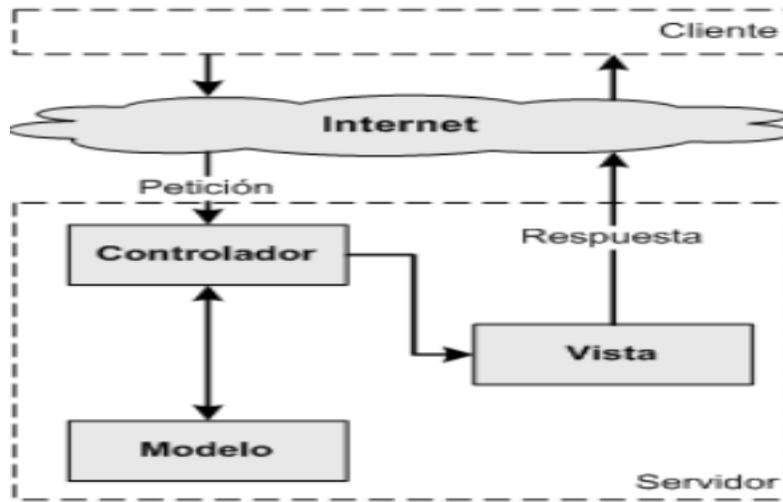


Figura 7: Funcionamiento del patrón MVC

La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones, porque cada paquete es independiente. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación. **(Potencier & Zaninotto, 2007)**

2.10 Conclusiones Parciales

En este segundo capítulo se abordó todo lo relacionado con la arquitectura del sistema, principalmente las herramientas, metodologías y lenguajes de programación posibles a utilizar para el desarrollo del sistema SGIGPE. Después de realizado este, se seleccionó de cada una, aquellas que por sus características daban solución a las necesidades del sistema que se quiere implementar. De las metodologías se seleccionó RUP, como lenguaje de modelado se definió UML y como herramienta CASE Visual Paradigm, además se realizó una breve descripción de los diferentes patrones que se utilizarán.

Capítulo 3: Características del sistema

3.1 Introducción

En este capítulo se realiza una descripción de la solución propuesta, que estará basada principalmente en el modelo de negocio, donde se expondrán los principales conceptos que se manejan en el negocio del sistema, ya sean objetos o eventos pertenecientes al entorno. También se describirán los requisitos funcionales y no funcionales que ha de satisfacer el sistema, se representará la modelación en términos de casos de usos del mismo y se detallará cada uno mediante sus especificaciones.

Mediante la metodología usada, se desarrollarán varias actividades y artefactos que permiten darle cumplimiento al problema planteado, y de esta forma, dominar el negocio como comienzo de un proceso que continuará con la especificación de los requisitos y con el diseño del sistema.

3.2 Modelado del negocio

¿Qué es un modelo de Negocio?

Es un modelo de casos de uso del negocio que describe los procesos de negocio de una empresa en términos de casos de uso del negocio y actores del negocio que se corresponden con los procesos de negocio y los clientes respectivamente. Al igual que el modelo de casos de uso para un sistema de software, el modelo de casos de uso del negocio presenta un sistema (en este caso, el negocio) desde la perspectiva de su uso, y esquematiza cómo proporciona valor a sus usuarios (en este caso, sus clientes y socios). **(Jacobson, 2000)**

El objetivo de este modelo del negocio es describir los procesos existentes y analizados, con el propósito de comprenderlos. Se especifican aquí qué procesos del negocio soportará el sistema. Además de identificar los objetos del dominio o del negocio, implicados, este modelo establece las competencias que se requieren de cada proceso: sus trabajadores, sus responsabilidades y las operaciones que llevan a cabo.

Reglas del negocio

1. Cada proyecto se podrá registrar solo por un área.
2. Cada proyecto debe tener al menos una persona y una computadora asignada.

Capítulo 3: Características del Sistema.

3. Cada persona debe estar registrada.
4. Las no conformidades se clasifican según su importancia en: alta, media o baja.
5. Por cada no conformidad se debe registrar una acción correctiva o preventiva, permiso o escalabilidad.
6. Cada no conformidad se le asigna un criterio, el responsable de la solución y la fecha de ejecución de la solución.
7. Las evaluaciones se clasifican según su finalidad en Ejecutada, Abortada y Aplazada.
8. Las evaluaciones pueden ser de Revisión, Auditoría o Diagnóstico.
9. Las evaluaciones de tipo auditorías se clasifican según su evaluación cualitativa en Satisfactoria, Aceptable o Deficiente.
10. No se puede realizar sobre un mismo proyecto dos evaluaciones al mismo tiempo.
11. Cada proyecto debe recibir al menos tres evaluaciones por año.

Actores del negocio

Actor	Descripción
Director general de la IP	El director general es el que solicita el levantamiento de información de las organizaciones productivas, aprueba la planificación de las evaluaciones, y se retroalimenta de estas por las estadísticas para la toma de decisiones.

Tabla 1: Actores del negocio

Trabajadores del negocio

Trabajador	Descripción
Especialista	Responsable del levantamiento de información, de hacer las planificaciones de las evaluaciones, archiva las no conformidades y realiza las estadísticas.
Evaluador líder	Encargado de dirigir las evaluaciones y entregar los resultados.
Evaluador	Miembro del equipo de evaluación.

Tabla 2: Trabajadores del Negocio

Diagrama de Casos de Uso del Negocio

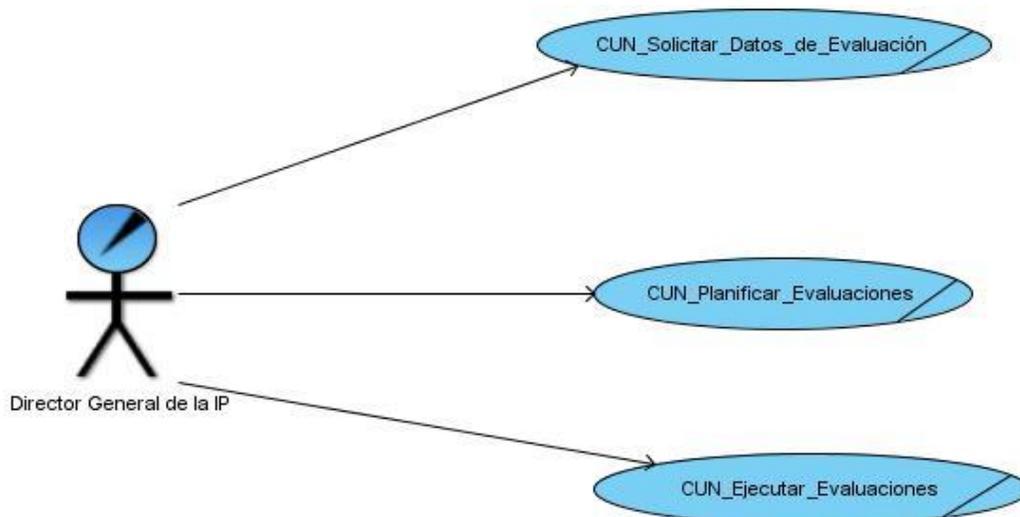


Figura 8: Diagrama de CUN.

Casos de Uso del Negocio

Un proceso de negocio es un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y manera y que emplean los recursos de la organización para dar resultados en apoyo a sus objetivos. Un caso de uso del negocio representa a un proceso de negocio, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores del negocio. Desde la perspectiva de un actor individual, define un flujo de trabajo completo que produce resultados deseables. (Colectivo, 2008). Para ver las descripciones de los Casos de Uso del Negocio ver a partir del anexo 2.

3.3 Requerimientos del Sistema

Los requisitos del sistema son las capacidades o condiciones que un sistema debe cumplir para lograr la satisfacción del cliente en cuanto a las funcionalidades requeridas o las restricciones que se imponen.

Requisitos funcionales

El sistema propuesto ha de satisfacer los siguientes requerimientos:

➤ RF 1 Autenticar usuario

- 1.1. Permitir al usuario introducir su usuario y contraseña del dominio UCI.

Capítulo 3: Características del Sistema.

- 1.2. Validar datos introducidos usando los directorios activos UCI.
- **RF 2 Gestionar usuario.**
 - 2.1 Asignar Roles.
 - 2.2 Eliminar rol a usuario.
- **RF 3 Gestionar evaluaciones.**
 - 3.1 Insertar evaluación.
 - 3.2 Modificar evaluación.
 - 3.3 Eliminar evaluación.
 - 3.4 Buscar evaluación.
- **RF 4 Gestionar no conformidad.**
 - 4.1 Insertar no conformidad.
 - 4.2 Modificar no conformidad.
 - 4.3 Eliminar no conformidad.
- **RF 5 Gestionar Evaluador.**
 - 5.1 Insertar evaluador.
 - 5.2 Modificar evaluador.
 - 5.3 Buscar evaluador.
- **RF 6 Consultar no conformidades**
 - 6.1 Mostrar criterios con mayor número de no conformidades.
 - 6.2 Mostrar no conformidades por áreas.
 - 6.3 Mostrar criterios con importancia alta.
- **RF 7 Consultas Generales.**
 - 7.1 Mostrar cantidad de evaluaciones.
 - 7.2 Mostrar relación de proyectos que su auditoría se ejecutó.
 - 7.3 Mostrar relación de proyectos que su auditoría abortó.
 - 7.4 Mostrar relación de proyectos que su auditoría se suspendió.

Requisitos no funcionales.

Capítulo 3: Características del Sistema.

Los requisitos no funcionales especifican propiedades o cualidades que el sistema debe tener. Representan las características del producto.

➤ **RNF 1 Requerimientos de Rendimiento**

Para el funcionamiento óptimo de la aplicación se siguieron las diferentes técnicas de elaboración en la Web, que facilitan el rápido acceso a sus páginas. La eficiencia del producto estará determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente/Servidor, y la velocidad de las consultas en la Base de Datos. La herramienta propuesta es rápida y el tiempo de respuesta es de alrededor de los 2.5 segundos.

➤ **RNF 3 Requerimientos de Portabilidad**

La herramienta propuesta podrá ser usada bajo cualquier sistema operativo; el Servidor Web Apache bajo sistema operativo Windows Server 2003 en adelante, y la Base de Datos bajo Windows Server 2003 en adelante. Los servidores Web y de Base de datos pueden estar en PC distintas o en una misma PC. Para su implementación se usará como herramienta de desarrollo Zend Studio for eclipse y Gestor de Base de Datos PostgreSQL 8.3.5.

➤ **RNF 4 Requerimientos de Seguridad**

• **4.1 Sistema.**

Para la autenticación del usuario se verificará que esté registrado en el servidor de la universidad, por medio del Directorio Activo, brindándole acceso solamente a su página principal. El sistema brindará acceso al usuario a las funcionalidades específicas, en dependencia del rol que tenga asignado. Para evitar las inyecciones de SQL se utilizará el lenguaje de programación Java Script, el cual permitirá la validación de los campos que tendrán los formularios, a la hora de almacenar la información en la Base de Datos.

• **4.2 Protección de la Base de Datos**

La base de datos debe tener varios esquemas, dividiendo así de una forma lógica las funcionalidades, evitando la pérdida total de la información en caso de algún accidente o ataque, para ello se realizará una salva de la Base de Datos que podrá ser actualizada periódicamente.

➤ **RNF 5 Requerimientos de Software**

Se requiere de navegador para Internet o Intranet, bajo cualquier sistema operativo; en el Servidor Web, sistema operativo Windows Server 2003 en adelante, y en la Base de Datos, Windows Server 2003 en adelante. Para su implementación se usará como herramienta de desarrollo Zend Studio for eclipse y Gestor de Base de Datos PostgreSQL 8.3.5.

Capítulo 3: Características del Sistema.

➤ **RNF 6 Requerimientos de Hardware.**

Se requiere de una máquina de 128 MB de RAM, un servidor con 1 Giga de RAM y 80 GB de disco duro que se encargue del Servidor Web y el de BD, todas las computadoras implicadas en los servidores, la administración como las de los usuarios, deben estar conectadas a una red y tener al menos 100 MB de velocidad.

➤ **RNF 7 Restricciones en el diseño y la implementación.**

La herramienta permitirá dar respuesta en el menor tiempo posible, garantizando la calidad del sistema, la base de dato será desarrollada en PostgreSQL. Para garantizar el desarrollo de la aplicación se utilizó como guía el Proceso Unificado de Desarrollo, obteniendo como documentación los distintos artefactos propuestos. Se utilizó para realizar los modelos del sistema, UML (Unified Modelling Language) y como herramienta de apoyo a este Lenguaje de Modelación el Visual Paradigm, para la gestión de configuración se utilizará Subversión y Tortoise.

3.4 Modelado del Sistema

En esta etapa se desarrolla el modelo de casos de uso basado en los requisitos previamente capturados. Este artefacto posibilita llegar a un acuerdo entre desarrolladores y clientes. Además, es la entrada fundamental para el análisis, diseño y pruebas.

Descripción de la solución propuesta.

Dándole cumplimiento a los requerimientos antes expuestos se ha analizado la creación de un sistema que posibilite, a los profesionales vinculados al proceso de producción de software en la UCI, gestionar la información generada en el proceso de evaluación realizada a los proyectos productivos. Para ello, los usuarios del sistema han de poseer un conjunto de características, es decir deben ser registrados en la aplicación, para luego darle los privilegios que según su categoría le correspondan, una vez realizada esta operación serán entonces clientes del sistema de gestión propuesto para los proyectos productivos de la UCI.

Como parte de la gestión de la información generada en este proceso de evaluación los clientes del sistema podrán conocer una serie de reportes referentes a las evaluaciones que serán generados por la aplicación según las necesidades que estos presenten, además de poder consultar e interactuar con la información que se arroje del proceso de evaluación.

Capítulo 3: Características del Sistema.

Actores del sistema.

Los actores son terceros fuera del sistema que interactúan directamente con él.

Actor	Descripción
Especialista	Responsable de gestionar las evaluaciones y los datos de los evaluadores que intervendrán en las mismas. Además de realizar consultas sobre las no conformidades obtenidas en las evaluaciones y sobre el estado de las auditorías.
Evaluador líder	Se encarga de gestionar los resultados de la evaluación que no son más que las no conformidades encontradas.
Administrador	Persona que se encarga de gestionar los usuarios, asignándole o eliminándole un rol dentro del sistema.

Tabla 3: Actores del sistema.

Diagrama de Casos de Uso del sistema.

Un diagrama de casos de uso del sistema representa gráficamente a los procesos y su interacción con los actores.

Capítulo 3: Características del Sistema.

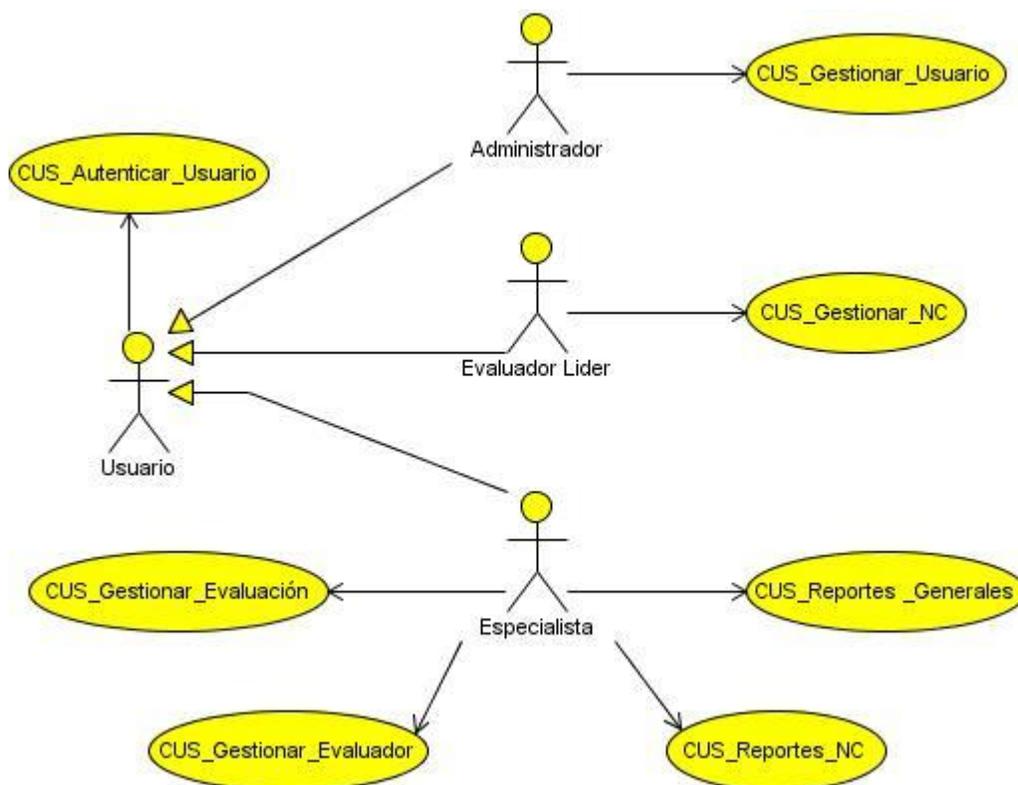


Figura 9: Diagrama de Casos de Usos del sistema

Descripción de los Casos de uso del sistema.

Los casos de uso del sistema son un fragmento de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores.

Caso de Uso:	Autenticar Usuario
Actores:	Usuario
Resumen:	El caso de uso comienza cuando un usuario desea acceder al sistema. Accede al sistema según su nivel de acceso. El caso de uso termina cuando el usuario accede satisfactoriamente al sistema.
Precondiciones:	La persona a autenticarse tiene que estar en el dominio de la UCI.
Referencias	Requisito funcional número 1.
Prioridad	Crítico

Capítulo 3: Características del Sistema.

Flujo Normal de Eventos: Ver Anexo 11.

Tabla 4: Descripción del CUS Autenticar Usuario.

Caso de Uso:	Gestionar Usuario.
Actores:	Administrador.
Resumen:	El caso de uso se inicia cuando el Administrador selecciona gestionar los diferentes usuarios del sistema. A partir de ahí el Administrador tiene la opción de asignarle o eliminarle un rol a un usuario.
Precondiciones:	El administrador tiene que estar registrado con anterioridad en la Base de Datos.
Referencias	Requerimiento Funcional número 2.
Prioridad	Crítico.
Flujo Normal de Eventos: Ver Anexo 12.	

Tabla 5: Descripción del CUS Gestionar Usuario.

Caso de Uso:	Gestionar Evaluación
Actores:	Especialista
Resumen:	Este caso de uso se inicia cuando el Especialista planifica una evaluación a un determinado proyecto. Este tiene la opción de Insertar, Modificar, Eliminar o Buscar una evaluación.
Precondiciones:	Debe estar guardado en la base de datos el proyecto posible a evaluar.
Referencias	Requerimiento Funcional número 3.
Prioridad	Crítico
Flujo Normal de Eventos: Ver Anexo 13	

Tabla 6: Descripción del CUS Gestionar Evaluaciones.

Caso de Uso:	Gestionar No Conformidad
Actores:	Evaluador Líder

Capítulo 3: Características del Sistema.

Resumen:	En este caso de uso el Evaluador Líder gestiona las no conformidades de la evaluación que se le hayan asignado en el período. Se inicia cuando el evaluador líder culmina la evaluación y registra los resultados de la misma. Al finalizar el caso de uso el evaluador líder puede Insertar, Modificar o Eliminar las no conformidades encontradas en la evaluación.
Precondiciones:	El Evaluador Líder sólo tendrá acceso a las evaluaciones que se le hayan asignado en el Plan de evaluaciones.
Referencias	Requerimiento Funcional número 4.
Prioridad	Crítico
Flujo Normal de Eventos: Ver Anexo 14	

Tabla 7: Descripción del CUS Gestionar No Conformidad.

Caso de Uso:	Gestionar Evaluador
Actores:	Especialista
Resumen:	El caso de uso se inicia cuando el Especialista selecciona la opción de gestionar los evaluadores. A partir de ahí este tiene la opción de Insertar, Modificar o Buscar los evaluadores.
Precondiciones:	El usuario debe estar autenticado y poseer el rol de Especialista.
Referencias	Requerimiento Funcional número 5.
Prioridad	Crítico
Flujo Normal de Eventos: Ver Anexo 15	

Tabla 8: Descripción del CUS Gestionar Evaluador.

Caso de Uso:	Reportes de No Conformidades
Actores:	Especialista
Resumen:	El caso de uso se inicia cuando el Especialista consulta las no conformidades que son arrojadas durante todo el proceso de evaluación a los proyectos productivos de la universidad. El sistema muestra los reportes sobre las no conformidad que pueden ser:

Capítulo 3: Características del Sistema.

	1.1 Criterios con mayor número de no conformidades. 1.2 No conformidades por áreas. 1.3 Criterios con importancia alta.
Precondiciones:	El usuario debe estar autenticado y poseer el rol de Especialista.
Referencias	Requerimiento Funcional número 6.
Prioridad	Secundario.
Flujo Normal de Eventos: Ver Anexo 16	

Tabla 9: Descripción del CUS Consultar No Conformidades.

Caso de Uso:	Reportes Generales
Actores:	Especialista
Resumen:	El caso de uso se inicia cuando el Especialista consulta las auditorías respecto al plan anual. El sistema muestra el resultado que arrojaron las auditorías en cuanto a los aspectos más relevantes para la universidad, estos pueden ser: 1.1 Cantidad de evaluaciones. 1.2 Relación de proyectos que su auditoría elaboró. 1.3 Relación de proyectos que su auditoría se suspendió. 1.4 Relación de proyectos que su auditoría se ejecutó.
Precondiciones:	El usuario debe estar autenticado y poseer el rol de Especialista.
Referencias	Requerimiento Funcional número 7.
Prioridad	Secundario.
Flujo Normal de Eventos: Ver Anexo 17	

Tabla 10: Descripción del CUS Consultar auditorías respecto al plan anual.

Es importante señalar que la gestión de los requisitos se tuvo en cuenta durante toda la etapa de levantamiento de requisitos, pues cuando se presentaron cambios, se realizaron tormentas de ideas para administrarlos, manejarlos y adecuarlos lo mejor posible, tratando de afectar lo menos posible el desarrollo del trabajo.

3.5 Conclusiones Parciales

En este capítulo se abordó todo lo referente a la descripción del sistema propuesto, para ello primeramente se desarrolló un modelo de negocio para un mejor entendimiento de los conceptos relacionados con la solución y el esclarecimiento de las funcionalidades que esta debe satisfacer. Tomando esto como base se llevó a cabo un sistema de modelación en términos de casos de usos que una vez especificados garantizan una comprensión exacta de lo que se desea desarrollar. Todos los artefactos generados servirán de entrada para la etapa posterior, el diseño.

Capítulo 4: Diseño del Sistema

4.1 Introducción

Este capítulo se abordará todo lo referente al diseño del sistema, que es de vital importancia porque es donde se modela el sistema y se le proporciona forma (la arquitectura), para que soporte todos los requisitos, incluyendo los no funcionales y las restricciones que se le suponen. Además se basa en los artefactos que genera este flujo y se refleja el uso patrones de diseño GRASP y el patrón arquitectónico Modelo Vista Controlador.

También tratará todo lo referente a la implementación del sistema y a la pruebas de Caja Negra que se le aplicaron. Además se verá reflejado los diagramas que corresponden al flujo de implementación, como es el caso del diagrama de despliegue que muestra mediante nodos como están distribuidos los recursos que intervienen e interactúan con el sistema y el diagrama de componentes que muestra la relación física entre estos.

4.2 Diseño de Sistema

Extensiones para diseño web

Debido a que la solución informática a realizar que se plantea es una aplicación web se decidió que la realización de este modelado sea a través de estereotipos web.

Estereotipos:



1. **Server Page:** representa la página Web que tiene código que se ejecuta en el servidor.



2. **Client Page:** una instancia de Página Cliente es una página Web, con formato HTML.

Capítulo 4: Diseño, Implementación y Prueba del Sistema.



3. **Form**: colección de elementos de entrada que son parte de una página cliente.
4. **<<Build>>**: representa una asociación especial que relaciona las páginas cliente con las páginas servidor.
5. **<<Link>>**: expresa las asociaciones más comunes entre las páginas, en este caso la del hipervínculo.
6. **<<Submit>>**: es la relación que se crea siempre entre una página servidor y un formulario.
7. **<<Redirect>>**: representa el redireccionamiento hacia otra página.
8. **<<Aggregated by>>**: representa la relación de agregación, como por ejemplo la de los formularios en las páginas clientes.

Diagramas de clases del Diseño

Como se mencionó anteriormente se aplicó el patrón Modelo-Vista-Controlador. El paquete Vista agrupa todas las clases de la presentación como las páginas servidoras, que son las encargadas de construir las páginas clientes, las que a su vez están compuestas por un formulario, aunque no todas.

En el paquete Modelo se encuentra la capa de acceso a datos. Sólo se representan cuatro clases que sintetizan las relaciones principales existentes entre ellas para minimizar la complejidad del diagrama. En el paquete Controlador están el **Controlador frontal ppqa** que se encarga de redireccionar todas las peticiones que se hacen desde de la vista y las clases **administracionActions** y **evaluacionActions** controlan todo el proceso y además de establecen la comunicación entre la vista y el modelo. También se observan el subsistema **Propel** que genera la capa de acceso a datos y el subsistema **Componentes de Symfony** que agrupa todas las clases y funcionalidades del framework Symfony. **(Ver los diagramas a partir del Anexo 18)**

Vista Lógica del Diseño

La vista lógica se encarga de describir el diseño el cual está muy cercano a la implementación, mostrando solo las clases más importantes así como la organización de éstas en capas. Esta vista permite observar en el interior del sistema como está diseñada su funcionalidad, tanto su comportamiento estático como su comportamiento dinámico.

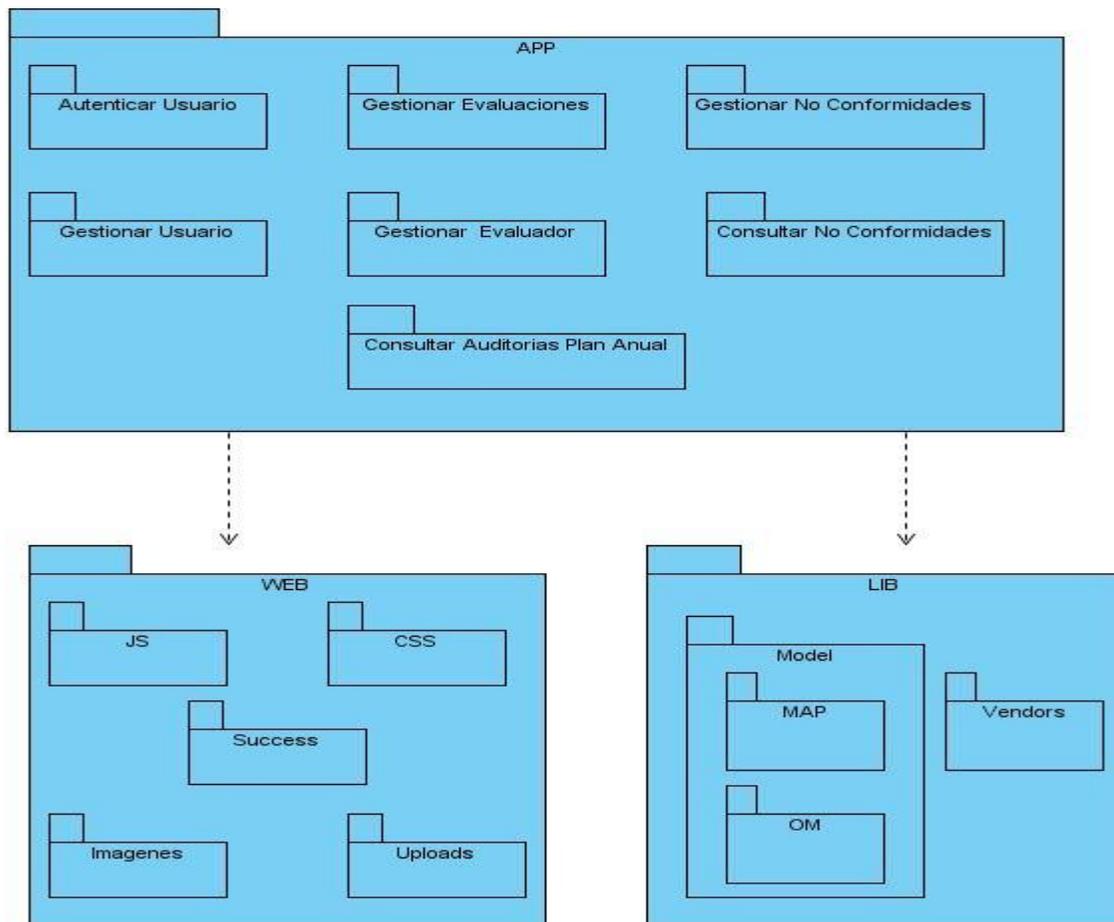


Figura 10: Vista Lógica del diseño

4.3 Diagrama Entidad Relación de la BD

A continuación se muestra el diseño de la Base de Datos, seguidamente a esta, la representación de la misma dividida en tres módulos fundamentales. El módulo de información al usuario, es donde se gestiona todo lo referente a recursos humanos, dígame proyectos o personas, donde se ve reflejado las relaciones existentes entre ellos. En el caso del módulo evaluación, en el cual se trabajará durante el desarrollo del sistema de gestión de información (SGIGPE), es donde se gestionan las evaluaciones, no conformidades, los evaluadores, además permite consultar los diferentes reportes con que cuenta la aplicación, basado en las no conformidades obtenidas en el proceso de evaluación y los estados de cada una de las evaluaciones. Por

Capítulo 4: Diseño, Implementación y Prueba del Sistema.

último el módulo de administración que es el encargado de controlar los usuarios que interactúan con el sistema y almacenar los roles que estos puedan tener en un momento determinado.

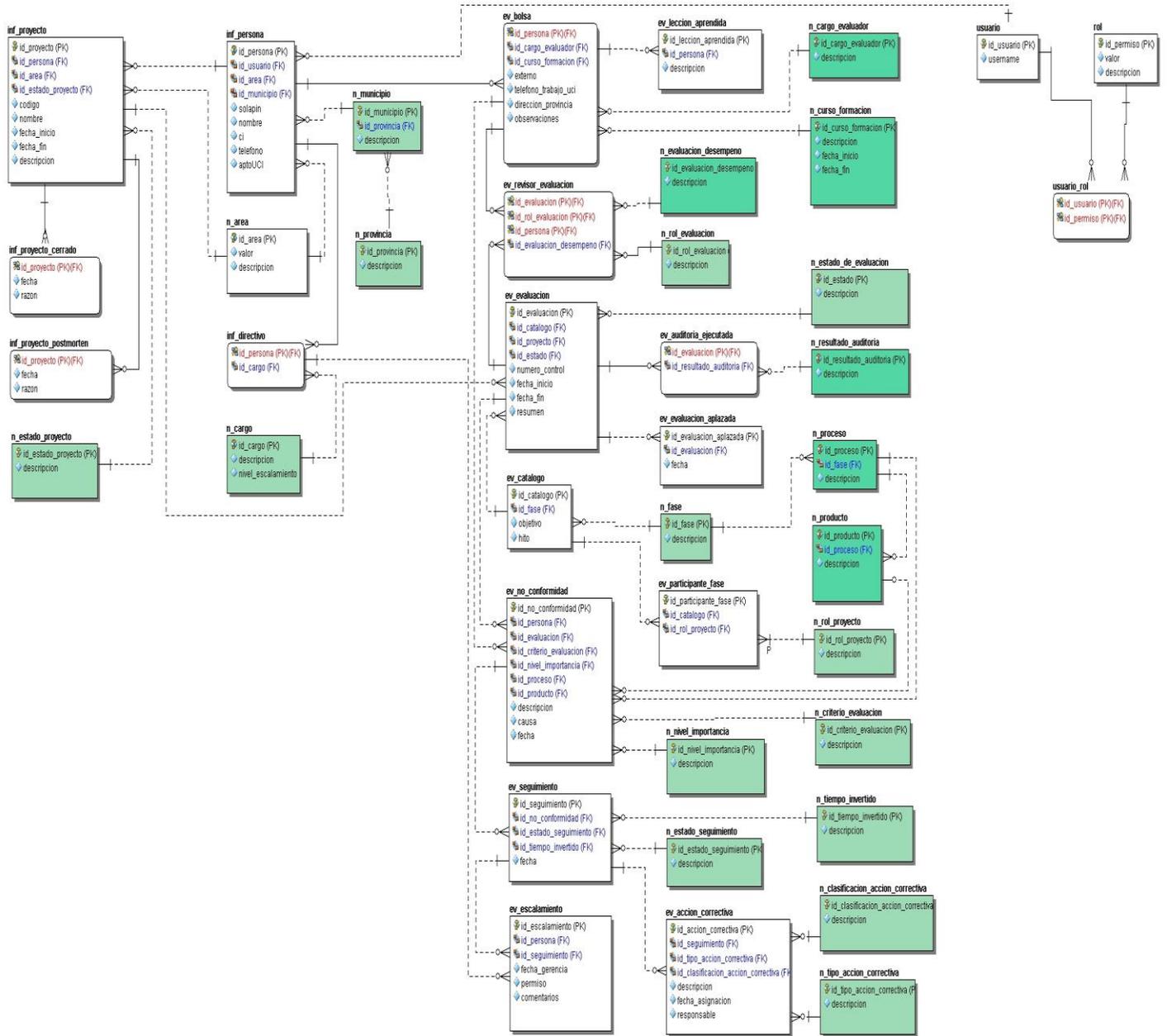


Tabla 11: Tablas de la Base de Datos

Módulo Información al Usuario.

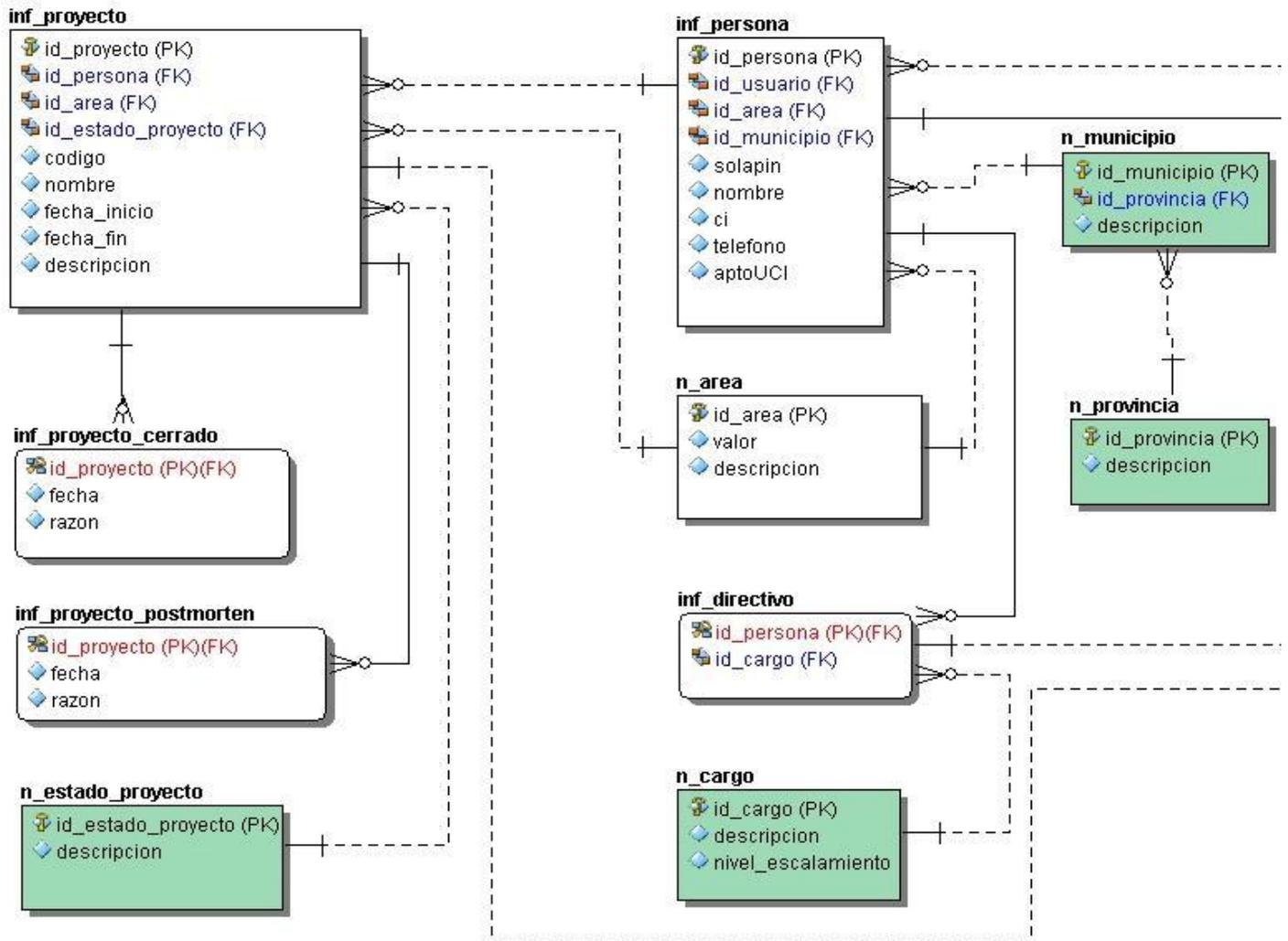


Figura 11: Módulo de Información al Usuario.

Capítulo 4: Diseño, Implementación y Prueba del Sistema.

Módulo Evaluaciones.

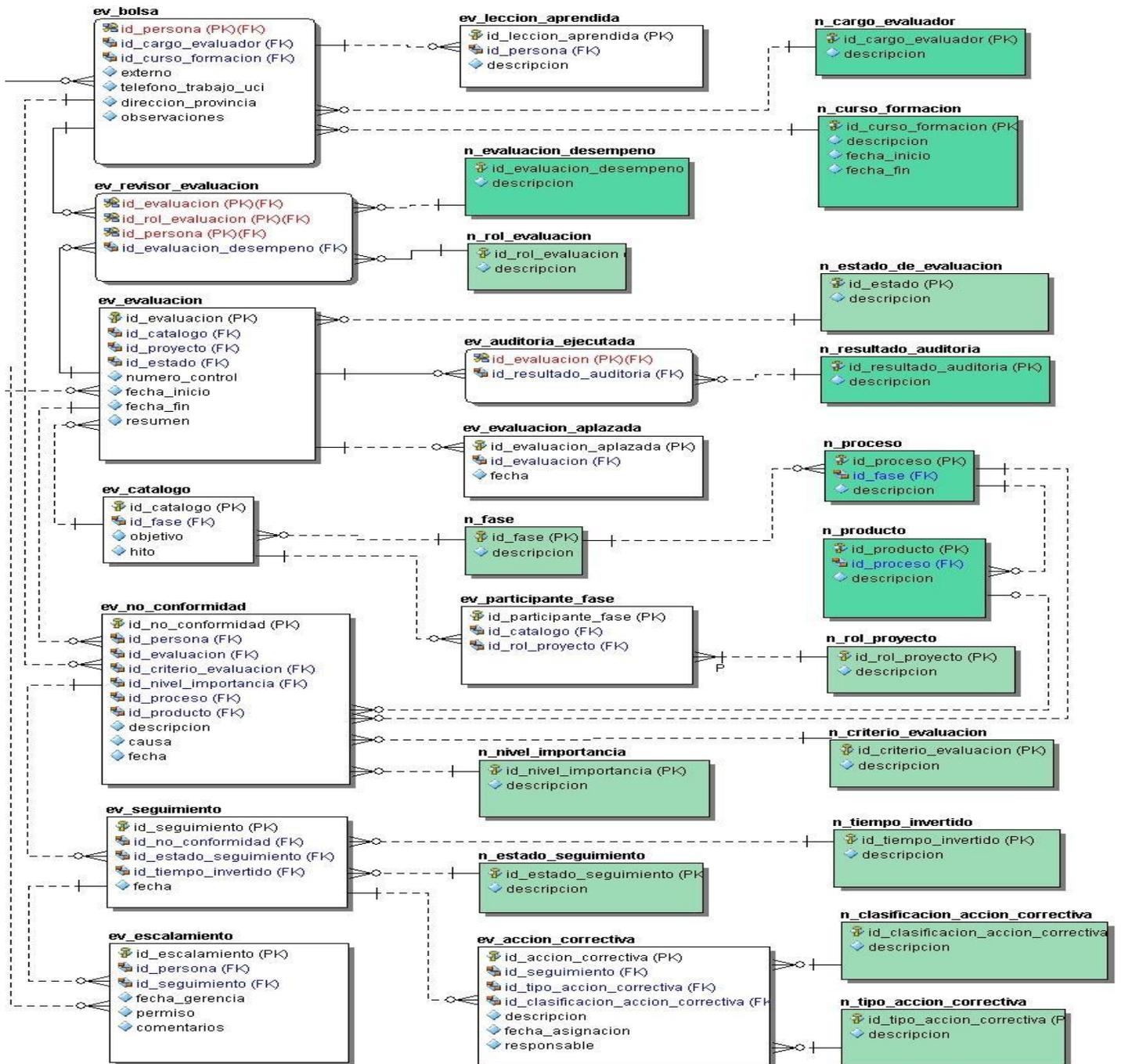


Figura 12: Módulo de Evaluaciones.

Módulo Administración.

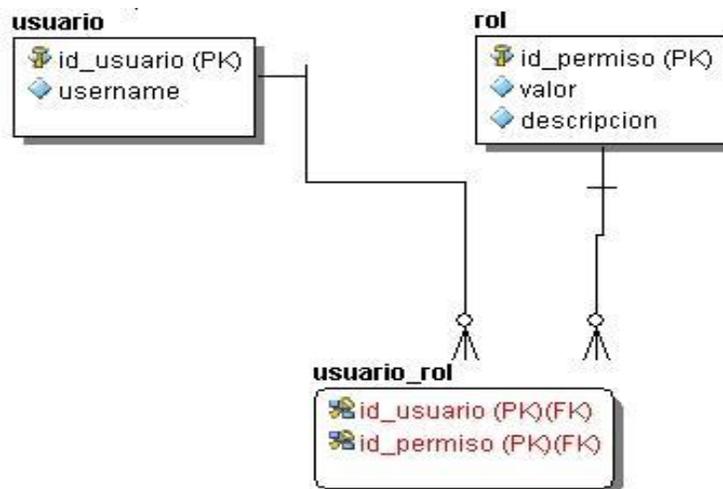


Figura 13: Módulo de Administración.

4.4 Aplicación Patrones de Diseño

Patrón Creador: La clase **AdministradorActions** es la que contiene las acciones (los métodos) y es la encargada de ejecutarlas. En dichas acciones se crean los objetos de las clases, que representan las tablas de la base de datos, por lo cual la clase **AdministradorActions** es la “creadora” de dichas clases.

Patrón Experto: es uno de los patrones que se emplean al trabajar con el Framework Symfony, se evidencia en la inclusión de **Propel** para mapear la base de datos. Este último, genera las clases para la gestión de la información de las tablas de dicha base de datos con las responsabilidades debidamente asignadas según el patrón Experto. Cada una de estas clases cuenta con un conjunto de funcionalidades que las convierte en expertas de la información de la tabla a la que representa.

Patrón Alta Cohesión: Una de las características principales del Framework Symfony es la organización del trabajo en cuanto a la estructura del proyecto, lo cual permite crear y trabajar con clases con una alta cohesión. Por ejemplo, la clase **AdministracionActions** contiene responsabilidades estrechamente relacionadas, encargadas de controlar las acciones de las plantillas. Esto hace posible que el software sea flexible a cambios sustanciales con efecto mínimo.

Patrón Bajo Acoplamiento: En el modelo es donde único se pueden encontrar relaciones de herencia y asociación entre las clases, pero no es de gran una jerarquía.

Patrón Controlador: La clase **AdministracionActions** es la clase controladora del sistema y su función es servir de enlace entre la vista y el modelo. Precisamente es esta la evidencia de la existencia del patrón Controlador en el proyecto. Además, Symfony implementa el patrón Front Controller (Controlador Frontal), en el cual existen varias clases controladoras que forman un flujo para atender las peticiones del usuario y de otras clases.

4.5 Implementación

4.5.1 Diagrama de Despliegue

El modelo de despliegue describe la distribución física del sistema, es decir, muestra como están distribuidas las funcionalidades entre los nodos de computación sobre los que se va a instalar el sistema. La **Figura 15** muestra los nodos que intervienen en el despliegue del sistema SGIGPE, cada uno representa un recurso de computación que utilizará y complementará al sistema y las relaciones entre ellos representan los medios de comunicación.

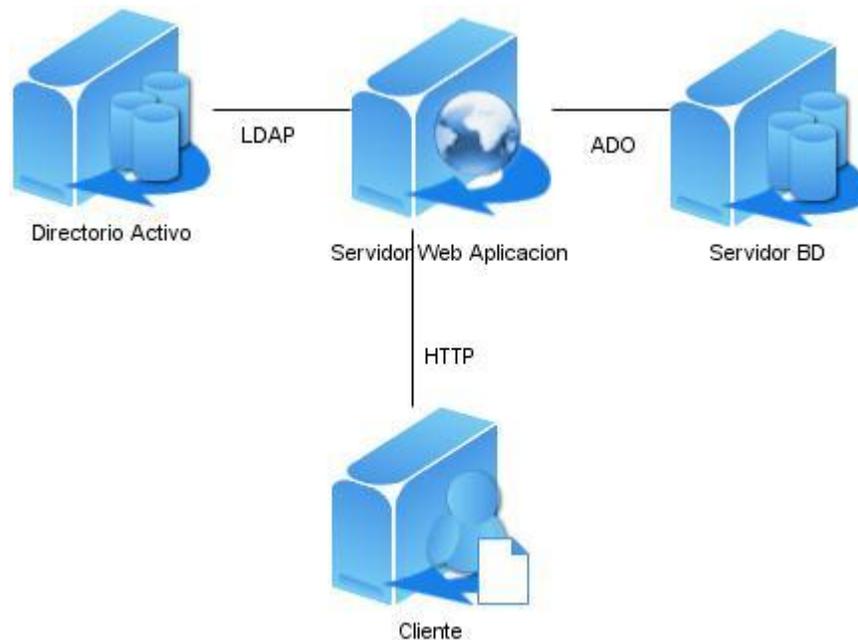


Figura 14: Diagrama General de Despliegue del sistema.

4.5.2 Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes software, sean éstos componentes de código fuente, binarios o ejecutables. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Los elementos de modelado dentro de un diagrama de componentes serán componentes y paquetes. **(Fernández, 2001)**

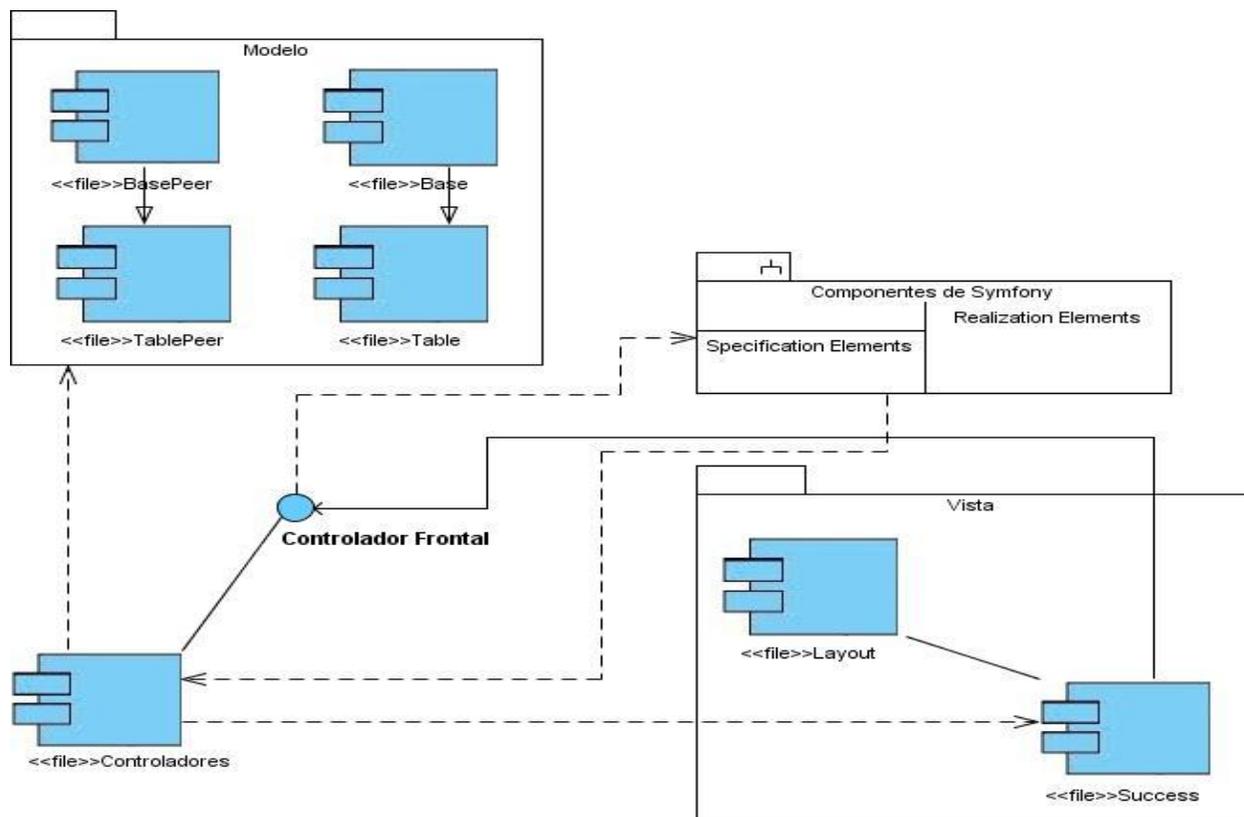


Figura 15: Diagrama de Componentes

4.6 Pruebas

El desarrollo del software ha de ir acompañado de alguna actividad que garantice la calidad del software, la prueba es un elemento crítico para ello.

Capítulo 4: Diseño, Implementación y Prueba del Sistema.

4.6.1 Prueba de Caja negra

Objetivo

El objetivo de realizarle este tipo de prueba al sistema, es para detectar el incorrecto o incompleto funcionamiento de este, así como los errores de interfaces, rendimiento y errores de inicialización y terminación.

Alcance

El proceso de pruebas de caja negra se va a centrar principalmente en los requisitos funcionales del software para verificar el comportamiento de la interfaz gráfica, su interacción con el usuario y la calidad funcional.

Descripción

Estas pruebas permiten evaluar todos los requisitos funcionales del programa.

Casos de Prueba

Para verificar que se cumplieran los requerimientos funcionales establecidos anteriormente, se le realizó la prueba a los casos de uso más críticos. Esta se realizó con el objetivo de evaluar la interacción del usuario con el sistema, dependiendo de las funcionalidades a las que tenga acceso el usuario y además se verificó que cada una de estas se correspondiera, con las descripciones de cada uno de los casos de uso del sistema realizadas anteriormente. **(Ver a partir del anexo 26)**

A continuación se muestra el resultado de las pruebas realizadas:

CU_Críticos:	Resultado del sistema	Resultados
Autenticar Usuario	El sistema permite acceder satisfactoriamente si el usuario está en el dominio UCI, detecta si el usuario dejó algún campo vacío y si los datos que introdujo están correctos.	Satisfactorio
Gestionar Usuario	El sistema da el acceso a la información según el rol que tenga el usuario, permite eliminar el rol del usuario seleccionado y detecta si hay algún campo vacío	Satisfactorio

Capítulo 4: Diseño, Implementación y Prueba del Sistema.

Gestionar Evaluación	El sistema permite insertar, eliminar y modificar los datos de la evaluación seleccionada por el Especialista correctamente, permite mostrar todas las evaluaciones que concuerden con los datos introducidos por el Especialista, de forma correcta y detecta si se dejó algún campo vacío.	Satisfactorio
Gestionar Evaluador	El sistema permite insertar, eliminar y modificar los datos del evaluador seleccionado por el Especialista correctamente, permite mostrar todos los evaluadores que concuerden con los datos introducidos por el Especialista de forma correcta y detecta si se dejó algún campo vacío.	Satisfactorio
Gestionar No conformidad	El sistema permite insertar, eliminar y modificar la no conformidad seleccionada por el Evaluador Líder y detecta si se dejó algún campo vacío.	Satisfactorio
Resumen: Se obtuvo un 100 % de resultados satisfactorios.		

Tabla 12: Resumen de la Prueba de Caja Negra.

4.7 Conclusiones

En el transcurso de este capítulo se obtuvieron los artefactos más significativos del diseño de la aplicación, estos servirán de entrada para el flujo de implementación, facilitando un mayor entendimiento de la aplicación a los desarrolladores. Como son los diagramas de clases del diseño, para lograr un mayor entendimiento de las funcionalidades a implementar, el diagrama Entidad Relación, entre otros, así como también la utilidad de los diferentes patrones de diseño en el desarrollo de la aplicación.

Ha mostrado la información referente a la etapa de implementación, empezando por el diagrama de despliegue, donde se detalló la relación Hardware y software del sistema y el de componentes. También se ve reflejado la prueba de Caja Negra que se le realizó al sistema con el objetivo de evaluar el cumplimiento de los requisitos funcionales definidos anteriormente. Todos los elementos mencionados anteriormente permitieron la construcción de un sistema eficiente y con mayor calidad.

Conclusiones Generales

Después de haber realizado un profundo estudio acerca de los procedimientos que conforman el proceso de Evaluación (Auditorías, Revisiones y Diagnósticos) y de la información resultante a automatizar y haber pasado por las distintas etapas que lleva la construcción de un software, como son Negocio, Requerimientos, Diseño e Implementación se arribó a la siguiente conclusión:

- Se investigaron cada una de las evaluaciones que conforman el proceso de evaluación, obteniendo así una descripción detallada sobre las etapas, fases, clasificaciones, ventajas y desventajas que presentan cada una de ellas.
- Se le realizó una entrevista a los especialistas de Calidad UCI, que permitió conocer como se lleva a cabo en la actualidad cada una de las evaluaciones antes mencionadas, las características que presentan, los problemas que afrontan a la hora de gestionar la información resultante de estas y la importancia que tiene para ellos la implementación del sistema SGIGPE.
- Se analizaron las metodologías, herramientas y lenguajes posibles a utilizar para la implementación del sistema SGIGPE, seleccionando las más adecuadas para este, en concordancia con las que utiliza la Dirección de Calidad de la UCI.
- En la etapa del diseño, se realizaron los diagramas de clases del diseño utilizando el patrón Modelo Vista Controlador, además se obtuvo el modelo Entidad Relación de la base de datos que se dividió por módulos para un mejor entendimiento del mismo.
- La etapa de implementación muestra a través de los diagramas de componentes y despliegue, como estarán organizados los componentes de software y los recursos de computación que utilizará el sistema, para evaluar el funcionamiento del mismo, se realizaron Pruebas de Caja Negra a los casos de uso más significativos.

Se diseñó e implementó una aplicación para la gestión de información resultante del proceso de evaluación a los proyectos productivos de la UCI, que tiene un ambiente fácil de entender y usar que cumple con los requerimientos definidos. Por todo lo antes expuesto, se puede asegurar que se culminó con toda la documentación del sistema y se dejó todo listo para comenzar la próxima etapa de refinamiento y mantenimiento de este.

Recomendaciones

Teniendo en cuenta la evolución de la informática, que todo está en constante desarrollo y que las herramientas de trabajo cada vez se hacen más robustas, es imposible que no se modifique esta aplicación. El sistema SGIGPE tiene como objetivo ayudar a los especialistas de la Dirección de Calidad de Software, que son los encargados de realizar las evaluaciones de auditorías y revisiones en los proyectos productivos de la universidad, a gestionar toda la información resultante de cada una de ellas.

Tomando como punto de partida los resultados obtenidos con la realización de este trabajo de diploma y el cumplimiento del Objetivo General, se recomienda:

- Integrar SGIGPE al sistema que se está llevando a cabo en la Dirección de Calidad de Software.
- Someter a prueba antes de una posible implantación para dejar atrás posibles fallas durante su aplicación.
- Después de integrada la aplicación SGIGPE con el sistema que se está desarrollando Dirección de Calidad de Software, que se le aplique a los proyectos productivos de la UCI.
- Proporcionar mayor funcionalidad al sistema atendiendo a las necesidades del equipo de la Dirección de Calidad de Software de la universidad.
- Seguir desarrollando la interfaz para responder a los diferentes requerimientos que puedan surgir a raíz de las necesidades crecientes de los usuarios.

Bibliografía Referenciada

- ✓ Aguilar Paulina, 1997. Calidad
- ✓ CalidadUCI, 2008. Infraestructura Productiva.Manual de Procedimiento.
- ✓ Crosby, Philip, 2001.
- ✓ Cuevas, Gonzalo, 2001.Gestión del Proceso Software
- ✓ Deming William E., 1940.
- ✓ Flores, Martínez y Trujillo, 2008. Revisiones a la actividad productiva.
- ✓ Galafet, Dailín, Martínez Amador José, 2008. Procedimiento para realizar Diagnósticos a las organizaciones Productivas en la UCI.
- ✓ Guerrero Luciano, 1999-2000. Métodos de evaluación de proyectos.
- ✓ IEEE, 1997.
- ✓ UCI, 2008. Universidad de las Ciencias Informáticas. Introducción a los conceptos y técnicas de las auditorías y revisiones.
- ✓ InformáticasUCI, 2008. Auditorías a la actividad productiva.
- ✓ Jacobson I.B, 2000.El proceso Unificado de Desarrollo de Software.
- ✓ Larman G, 1999. UML y Patrones.
- ✓ Pressman, 2001.Ingeniería de Software. Un enfoque Práctico
- ✓ Pressman Roger, 1992.
- ✓ RAE, 2006.Real Academia de la Lengua Española.
- ✓ S.N.C, 2002. Organo Interno de Control Nacional Financiero. Sistema de Auditoria y Control.
- ✓ Saussure de Ferdinand, 1931. Qué es un sistema?
- ✓ Solutions Intelligent, 2009.Intelligent Solutions
- ✓ Trujillo, Acosta, Díaz, 2008.Auditorías a la actividad productiva
- ✓ Velazquez, 2009. Auditoría:sistemas de gestión de calidad .

Bibliografía

- ✓ Buades, Gabriel. 2002. Auditorías Informáticas.
- ✓ Cuevas, Gonzalo. 2001. Gestión del Proceso Software.
- ✓ Delgado, Martha; Álvarez, Sofía; Rosete, Alejandro, 2005. Una propuesta de introducción
- ✓ *Fase de Inicio. Modelo del Negocio.* Ingeniería de Software, Colectivo de la asignatura. 2008.
- ✓ Fernández, Ana, 2001.
- ✓ Goñi, Ivis; Artilles, Sara; Nieves, Yadira, 2008. El Sistema Automatizado para el Diagnóstico de la Gestión de Información y Conocimiento en la Empresa. GECYT
- ✓ Inst. de Auditores internos, 1999. Proceso de Auditorías Internas.
- ✓ Lovelle, Juan M, 1999. Calidad del Software [Informe].
- ✓ Moya, Osiris; Pérez, Rislaidy, 2001. Propuesta de proceso de evaluación de la calidad del software educativo.
- ✓ Naranjo, Alice. 2008. Auditorías de Sistemas
- ✓ Oktaba, Hanna y otros. 2004. Método de Evaluación de procesos para la industria de software.
- ✓ Piriz, Nancy; Tedesco, Claudia, 2004. El desafío de implementar un proyecto de e-learning.
- ✓ Potencier, F; Zaninotto, F. 2007. Symphony la guía definitiva.
- ✓ Pressman, Roger, 2001. Ingeniería de Software. Un enfoque Práctico.
- ✓ Ingeniería de Software, 2008. Fase de Inicio. Modelo del Negocio.
- ✓ Ingeniería Software, 2003. Calidad del software [Informe]. Universidad Rey Juan Carlos.
- ✓ Váscquez, 2007. Evaluacion y mejoras de procesos.
- ✓ Velazquez , 2006. Auditoría:sistemas de gestión de calidad

Anexos

Anexo 1: Entrevista a realizar por los especialistas de Calidad UCI.

Preguntas:

1. ¿Haga una breve descripción de como se realiza el proceso de evaluación a los proyectos productivos de la Universidad de las Ciencias Informáticas?
2. ¿Es el mismo procedimiento para todos los proyectos?(Se realiza de la misma forma)
3. ¿Cómo se maneja la información que es generada en estos procesos?
4. ¿Mencione las inconformidades existentes a la hora de manipular la información generada en este proceso?
5. ¿Existe un historial de los procesos realizados y la documentación generada?
6. Se quiere realizar una aplicación que gestione la información generada en el proceso de evaluación a los proyectos de Software de la Universidad de las Ciencias Informáticas. El sistema trabajará solamente con la información generada por el proceso de evaluaciones de aseguramiento de la calidad (PPQA), realizada a los proyectos. Tendrá el objetivo de darle respuesta a los requerimientos que requiere para su funcionamiento, unido a esto permitirá mostrar los reportes de forma automática.

Reporte que mostrar el sistema:

➤ **Reportes de no conformidades**

Criterios con mayor número de no conformidades

No conformidades por áreas

Criterios de importancia alta con mayor número de no conformidades

➤ **Reporte de Auditorías en porcentos respecto al Plan Anual**

Cantidad de evaluaciones planificadas y ejecutadas por áreas

Relación de proyectos que su auditoría se ejecutó

Relación de proyectos que su auditoría abortó

Relación de proyectos que su auditoría se suspendió

¿Mencione que importancia y ventajas tendrá la implementación de dicho sistema para el equipo especializado de la Dirección de Calidad de Software en la universidad?

Descripción de los Casos de Uso del Negocio

Anexo 2: Descripción del CUN Solicitar Datos de Evaluación

Nombre del CU	CU_Solicitar_Datos_de_Evaluación
Actor	Director general de la IP
Resumen	El caso de uso inicia cuando el Director General de la IP solicita los datos de la evaluación de los proyectos existentes en la universidad.
Precondiciones	Que existan proyectos
Flujo Normal de Eventos	
Acción del actor	Respuesta del proceso de negocio
El Director General de la IP solicita realizar un levantamiento de información sobre las organizaciones productivas de la universidad	El Especialista conforma las plantillas para solicitar los datos.
	El Especialista envía por correo electrónico las plantillas al Director General de la IP para su aprobación.
El Director General de la IP revisa las plantillas	
El Director General de la IP envía la aprobación de las plantillas al Especialista. En caso de que no sean aprobadas las plantillas ver flujo alterno.	El Especialista envía por correo electrónico las plantillas al Evaluador.
	El Evaluador envía por correo electrónico las plantillas al Evaluador Líder.
	El Evaluador Líder llena las plantillas con los datos correspondientes
	El Evaluador Líder envía por correo electrónico las plantillas llenas al Especialista.

	El Especialista revisa la calidad de las plantillas. En caso de que los datos no estén completos o estén incorrectos (ver flujos alternos).
	El Especialista registra los datos para obtener las estadísticas.
	El Especialista genera las estadísticas de la Universidad y de cada Área.
	El Especialista desarrolla los informes con las estadísticas generadas.
	El Especialista envía por correo electrónico los informes con las estadísticas generadas al Director General de la IP.
El Director General de la IP revisa los informes En caso de que no lo apruebe (ver flujos alternos).	
El Director General de la IP registra la aprobación de los informes.	
El Director General de la IP envía los informes a Publicación.	
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
El Director General de la IP envía su inconformidad sobre las plantillas al Especialista.	Ir a la acción número 2 en el flujo normal de eventos.
El Especialista encontró errores en los datos de las plantillas que envió el Evaluador Líder.	Ir a la acción número 9 en el flujo normal de eventos.
El Especialista encontró errores en los datos de las plantillas que envió el Evaluador Líder.	Ir a la acción número 13 en el flujo normal de eventos.

Postcondiciones	Actualizado la información de los proyectos y los informes de las estadísticas de la producción en esta de publicación.
Ver Diagrama de Actividades en Anexo 5.	
Ver Modelo de Objeto en Anexo 6.	

Anexo 3: Descripción del CUN Planificar Evaluaciones

Nombre del CU	CU_Planificar_Evaluaciones
Actor	Director General de la IP-
Resumen	El caso de uso describe el proceso de elaboración del plan de evaluaciones.
Precondiciones	Existan proyectos
Flujo Normal de Eventos	
Acción del actor	Respuesta del proceso de negocio
El Director General manda el especialista a realizar el plan de evaluaciones.	El Especialista conforma el plan de evaluaciones.
	El Especialista envía por correo electrónico el plan de evaluaciones al Director General de la IP
El Director General de la IP revisa el plan de evaluaciones En caso de que no esté de acuerdo con el plan de evaluaciones (ver flujos alternos).	
El Director General de la IP envía la aprobación del plan de evaluaciones al Especialista.	El Especialista publica el plan de evaluaciones.
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
4.1. El Director General de la IP no está de acuerdo con el plan de evaluaciones.	4.2 Ir a la acción número 1 en el flujo normal de eventos.
Postcondiciones	
Ver Diagrama de Actividades en Anexo 7.	

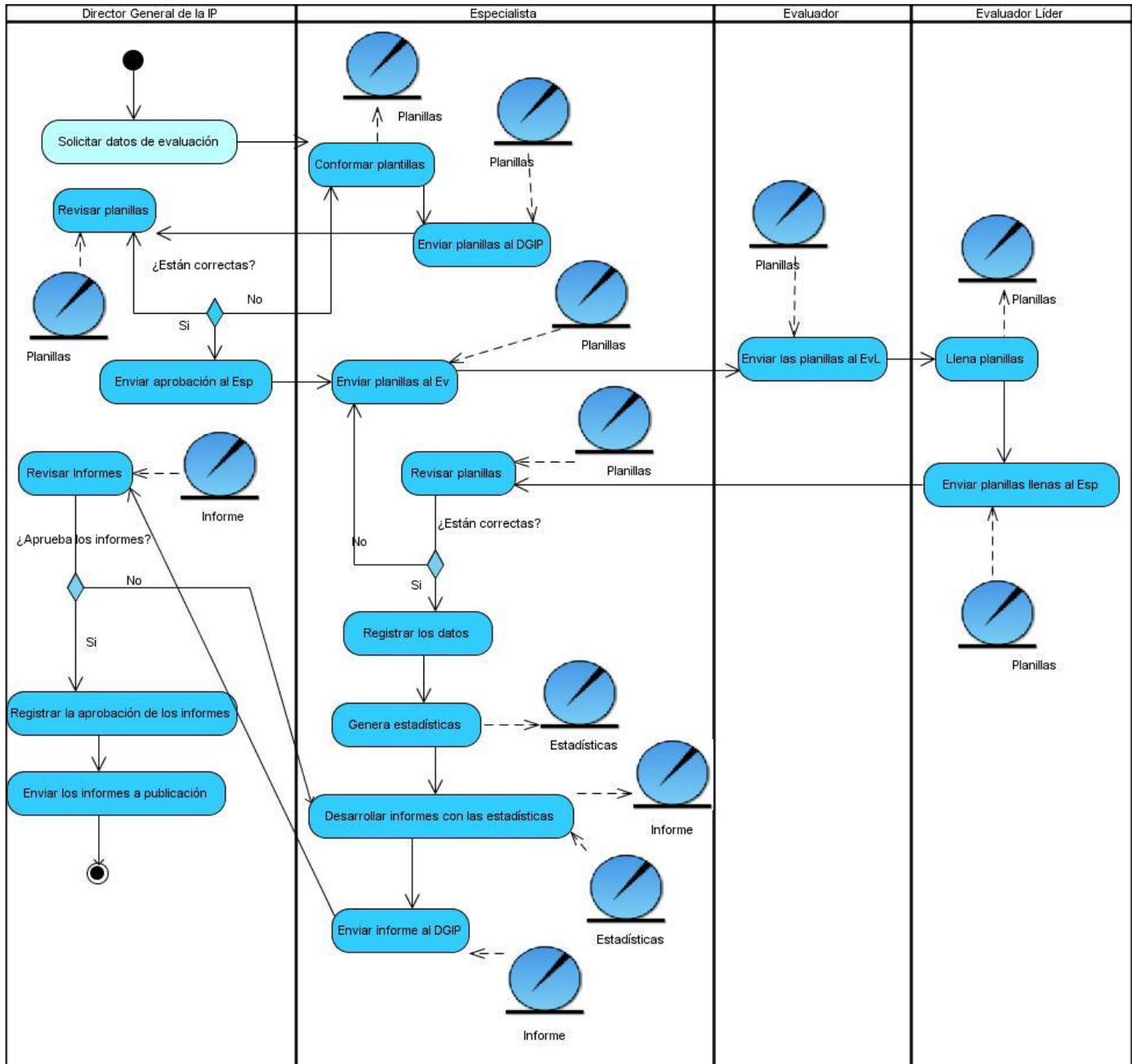
Ver Modelo de Objetos en Anexo 8.

Anexo 4: Descripción del CUN Ejecutar Evaluaciones.

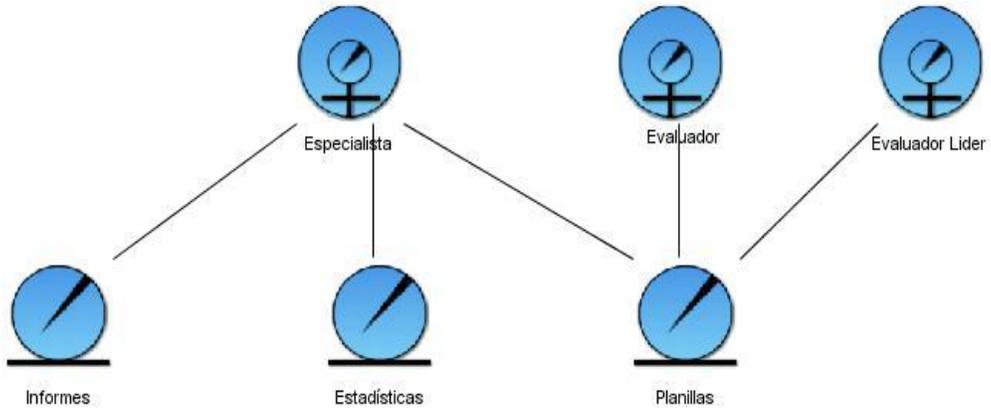
Nombre del CU	CU_Ejecutar_Evaluaciones
Actor	Director General de la IP
Resumen	El proceso de realizar evaluaciones a proyectos productivos
Precondiciones	Exista una evaluación a realizar
Flujo Normal de Eventos	
Acción del actor	Respuesta del proceso de negocio
El Director General le pide al equipo de evaluadores que reúnan para empezar con las evaluaciones.	El Evaluador Líder, los Evaluadores, y el Especialista realizan la reunión de inicio.
	El Evaluador Líder y los Evaluadores aplican la lista de verificación a los elementos del proyecto
	El Evaluador Líder y los Evaluadores elaboran los resultados y realizan los informes.
	El Evaluador Líder comunica los resultados al Director General de la IP.
	El Evaluador Líder elabora las acciones correctivas o preventivas.
	El Evaluador Líder envía los informes al Especialista.
	El Especialista revisa la calidad de los informes. En caso que no estén correctos (ver flujos alternos)
	El Especialista registra los datos de la evaluación.
	El Especialista genera las estadísticas de la Universidad y de cada Área.
	El Especialista desarrolla los informes con las estadísticas generadas.
	El Especialista envía por correo electrónico los informes con las estadísticas generadas al Director General de la IP

El Director General de la IP revisa los informes. En caso de que no lo apruebe (ver flujos alternos)	
El Director General de la IP envía la aprobación de los informes.	
El Director General envía los informes a Publicación.	
Flujos Alternos	
Acción del actor	Respuesta del proceso de negocio
12.1 Los informes están incorrectos	8.1 Ir a la acción número 3 en el flujo normal de eventos
12.2 El Director General de la IP no aprobó los informes.	13.1 Ir a la acción número 11 en el flujo normal de eventos
Postcondiciones	
Ver Diagrama de Actividades en Anexo 9. Ver Modelo de Objetos en Anexo10.	

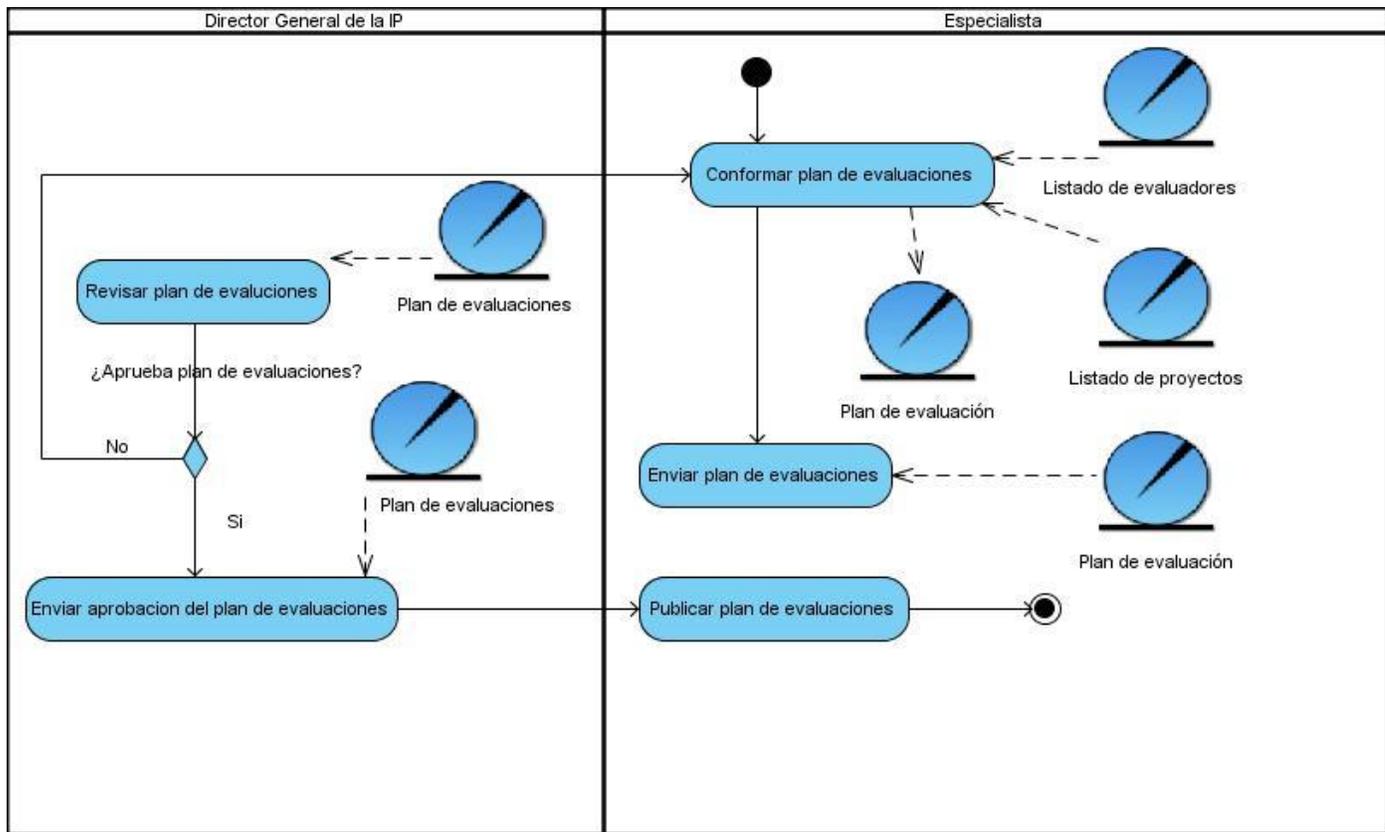
Anexo 5: Diagrama de actividades del CUN Solicitar Datos de Evaluación



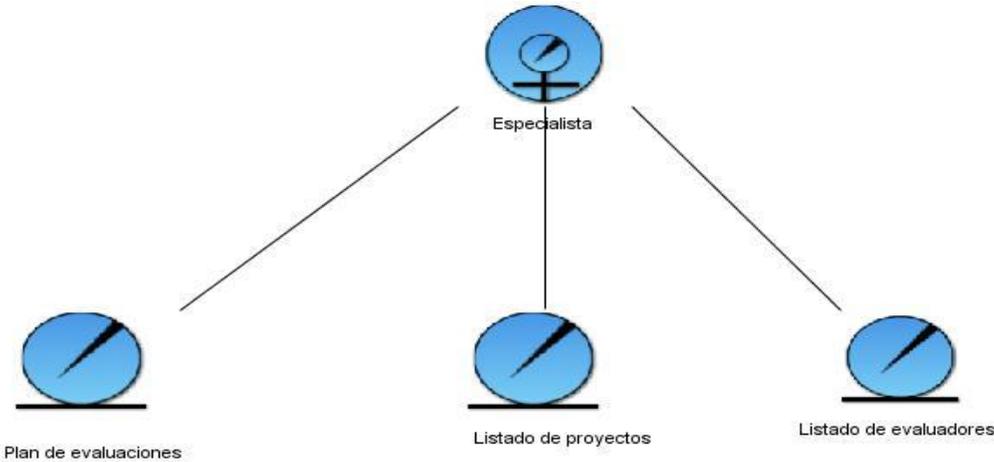
Anexo 6: Modelo de objeto del CUN Solicitar Datos de Evaluación



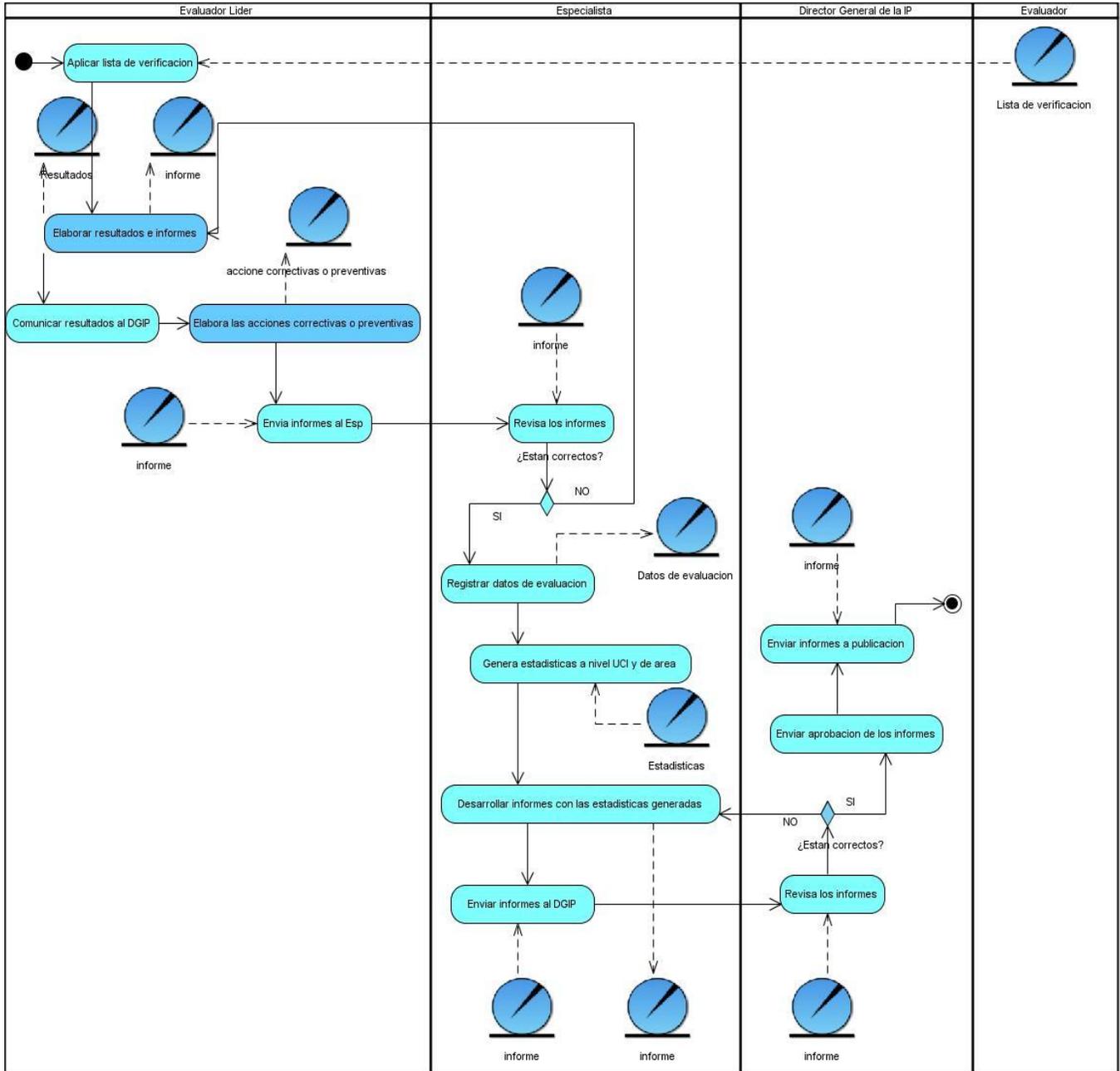
Anexo 7: Diagrama de actividades del CUN Planificar Evaluaciones



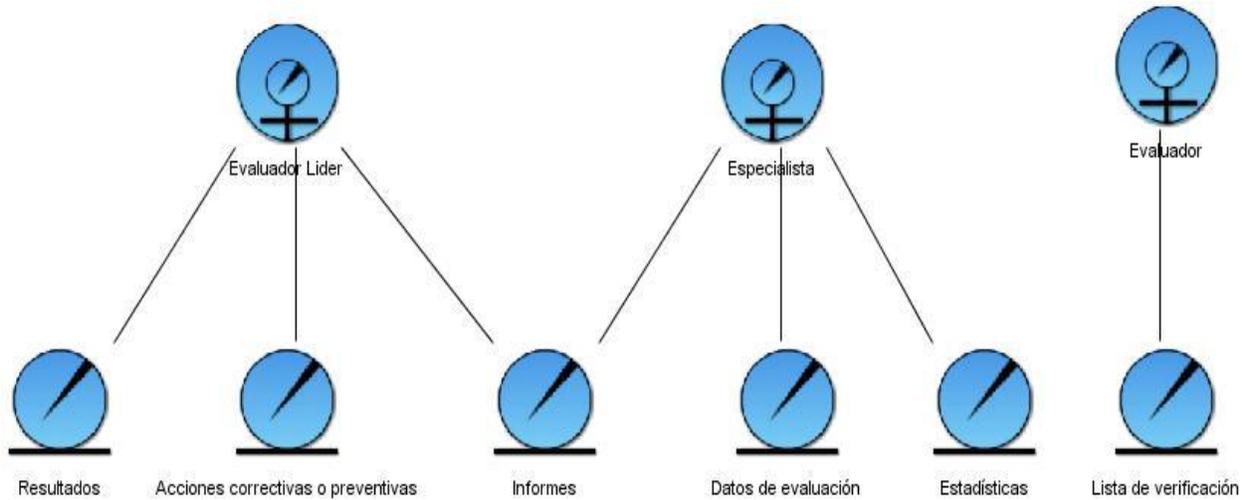
Anexo 8: Modelo de Objeto del CUN Planificar Evaluaciones



Anexo 9. Diagrama de Actividades del CUN Ejecutar Evaluaciones



Anexo 10: Modelo de Objeto del CUN Ejecutar Evaluaciones



Anexo 11: Flujo Normal de Eventos del CUS Autenticar Usuario.

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1_ Se inicia el caso de uso cuando el usuario solicita acceder al Sistema.	2_ El sistema muestra un formulario para introducir usuario y contraseña. Usuario Contraseña
3_ El usuario introduce los datos y pulsa el botón Aceptar.	4_ El sistema verifica que los datos estén correctos contra los datos del dominio de la UCI.
5_ El usuario accede correctamente al sistema.	6_ El sistema se configura para que el usuario sólo tenga acceso a la parte que según su rol le permite.

Prototipo de Interfaz de Usuario

Flujos Alternos

Acción del Actor	Respuesta del Sistema
3_ Si el Usuario pulsa el botón Salir.	3.1_ Se termina el caso de uso y el sistema muestra la página principal.

4_En caso de que los datos estén incorrectos	4.1_ El sistema muestra el mensaje “Usuario o contraseña incorrectos”
Prototipo de Interfaz	
Postcondiciones	El Usuario queda Autenticado y puede acceder al sistema según el rol definido.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1_ Se inicia el caso de uso cuando el usuario solicita acceder al Sistema.	2_ El sistema muestra un formulario para introducir usuario y contraseña. Usuario Contraseña
3_ El usuario introduce los datos y pulsa el botón Aceptar.	4_ El sistema verifica que los datos estén correctos contra los datos del dominio de la UCI.
5_ El usuario accede correctamente al sistema.	6_ El sistema se configura para que el usuario sólo tenga acceso a la navegación básica del sistema.
Prototipo de Interfaz de Usuario	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
3_ Si el Usuario pulsa el botón Salir.	3.1_ Se termina el caso de uso y el sistema muestra la página principal.
4_ En caso de que los datos estén incorrectos	4.1_ El sistema muestra el mensaje “Los datos son incorrectos, intente de nuevo”
4.2_ El usuario acepta el mensaje	4.3_ El sistema pasa a la acción 2
Prototipo de Interfaz	
Post condiciones	El Usuario queda Autenticado y puede acceder al sistema según el nivel de acceso definido.

Anexo 12: Flujo Normal de Eventos del CUS_ Gestionar Usuario.

Flujo Normal de Eventos	
1_ El caso de uso se inicia cuando el	2_ El sistema muestra un formulario con la opción de

Administrador, selecciona la opción de gestionar usuario.	asignarle o eliminarle el rol a un usuario.
3_El Administrador selecciona una de las opciones: 3.1_Solicita Asignar Rol, Ver Sección Asignar Rol. 3.2_Solicita Eliminar Rol usuario, Ver Sección Eliminar Rol usuario	
Flujo Normal de Eventos	
Sección “Asignar Rol”	
4_El caso de uso se inicia cuando el Administrador, selecciona la opción de asignar roles.	5_ El sistema muestra un formulario para introducir el nombre del usuario y una lista de todos los roles que puede tener este. La primera vez que los usuarios accedan al sistema, tiene que esperar a que el Administrador le asigne un rol para acceder a determinada información.
6_El Administrador introduce el nombre del usuario y un rol determinado para asignárselo pulsando el botón Asignar.	7_El sistema muestra un mensaje “¿Está seguro que desea asignar este rol?”.
8_ El Administrador acepta el mensaje.	9_El sistema muestra un mensaje “El rol ha sido asignado satisfactoriamente”
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
6_Si no es entrado a la aplicación un usuario para asignarle un rol.	6.1 _El sistema muestra el mensaje “Debe introducir el nombre de usuario”.
6.2_El Administrador acepta el mensaje.	6.3_El sistema termina el caso de uso.
6_Si el usuario introducido ya tiene asignado un rol y se le quiere asignar el mismo por segunda vez.	6.1 _El sistema muestra el mensaje “Este usuario ya tiene este rol”.
6_Si el usuario es incorrecto.	6.1_ El sistema muestra el mensaje “Ese usuario no existe”
Flujo Normal de Eventos	
Sección “Eliminar Rol Usuario”	

Acción del Actor		Respuesta del Sistema
10_ El caso de uso se inicia cuando el Administrador, selecciona la opción de Eliminar rol usuario.		11_ El sistema muestra un formulario para que el Administrador introduzca el usuario.
12_ El Administrador introduce el usuario y pulsa el botón Eliminar.		13_ El sistema verifica que haya introducido el usuario para eliminar el rol o unos de los roles que tenga asociado.
		14_ El sistema muestra el mensaje “Está seguro que desea eliminar este rol”.
15_ El Administrador acepta el mensaje.		16_ El sistema elimina el rol al usuario seleccionado.
		17_ El sistema muestra el mensaje “El rol ha sido eliminado satisfactoriamente”.
18_ El Administrador acepta el mensaje.		19_ El sistema pasa a la acción 2.
Flujos Alternos		
Sección “Eliminar Rol Usuario”		
Acción del Actor		Respuesta del Sistema
12_ Si el Administrador no introduce un usuario para eliminar su rol.		12.1_ El sistema muestra un mensaje “Debe introducir el usuario para eliminarle un rol”.
12.2_ El Administrador acepta el mensaje.		12.3_ El sistema pasa a la acción 2.
15_ Si el Administrador selecciona la opción de cancelar la solicitud de confirmación para eliminar el rol que tenga asignado el usuario.		15.1_ El sistema pasa a la acción 2.
Prototipo de Interfaz		
Postcondiciones	Los datos de los usuarios al sistema fueron almacenados, o eliminados en la Base de Datos. Cada usuario estará asociado a un rol en dependencia de las funciones que desempeñe.	

Anexo 13: Flujo Normal de Eventos del CUS Gestionar Evaluación

Flujo Normal de Eventos	
1_ El caso de uso se inicia cuando el Especialista selecciona la opción de Gestionar	2_ El sistema muestra un formulario con las opciones insertar, modificar, buscar y eliminar evaluación.

Evaluación para insertar las posibles evaluaciones a realizar	
3_El Especialista selecciona una de las opciones: Solicita Insertar Evaluación, Ver Sección Insertar Evaluación. Solicita Modificar Evaluación, Ver Sección Modificar Evaluación Solicita Buscar Evaluación, Ver Sección Buscar Evaluación Solicita Eliminar Evaluación, Ver Sección Eliminar Evaluación.	
Flujo Normal de Eventos Sección “Insertar Evaluación”	
Acción del Actor	Respuesta del Sistema
	4_El sistema muestra el formulario para entrar los datos de la Evaluación a realizar.
5_El Especialista introduce los datos solicitados y pulsa el botón Insertar.	6_El sistema muestra el mensaje” ¿Está seguro que desea insertar esta evaluación?”
7_El Especialista acepta el mensaje	8_El sistema valida que la información entrada fue la necesaria para completar la operación.
	9_El sistema muestra el mensaje ” La evaluación ha sido insertada satisfactoriamente”
10_El Especialista acepta el mensaje	
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
5_ Si se deja algún campo vacío o con datos incorrectos.	5.1_ El sistema muestra el mensaje “Error en los datos introducidos”.
5.2_ El Especialista acepta el mensaje.	5.3_ Se pasa a la acción 2.

9_Si el Especialista selecciona la opción de Cancelar para la solicitud de guardar los datos.	9.1_Vuelve a la acción 4.
Flujo Normal de Eventos	
Sección “Modificar Evaluación”	
Acción del Actor	Respuesta del Sistema
14_El Especialista selecciona la Evaluación que desee modificar y pulsa el botón Modificar.	
	15_El sistema valida que esté seleccionada una Evaluación para modificar.
	16_ El sistema muestra el mensaje ¿Está seguro que desea modificar esta evaluación
17_ El Especialista acepta el mensaje	18_El sistema muestra los datos actuales de la Evaluación seleccionada.
18_ El Especialista modifica los campos que desee de la Evaluación seleccionada y pulsa el botón Modificar.	19_El sistema muestra el mensaje "Desea modificar la evaluación"
20_ El Especialista acepta el mensaje	21_ El sistema valida que la información entrada fue la necesaria para completar la operación.
	22_ El sistema guarda la nueva información.
	23_El sistema muestra el mensaje "Los datos de la Evaluación han sido modificados satisfactoriamente"
24_El Especialista acepta el mensaje	25_Se pasa a la acción 2.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
14_Si el Especialista no selecciona la Evaluación que desea modificar.	14.1_ El sistema muestra el mensaje: "Debe seleccionar una Evaluación a modificar".
14.2_El Especialista acepta el mensaje	14.3_Se pasa a la acción 14.
20_Si el Especialista selecciona la opción de cancelar la acción de modificar los datos.	20.1_Se pasa a la acción 14.

17_Si no son válidos los valores de entrada para los datos modificados.	17.1_ El sistema muestra el mensaje “Entrada de datos no válida”.
17.2_El Especialista acepta el mensaje.	17.3_Se pasa a la Acción 15.
Flujo Normal de Eventos	
Sección “Eliminar Evaluación”	
Acción del Actor	Respuesta del Sistema
22_El Especialista selecciona la Evaluación a eliminar y pulsa el botón Eliminar.	23_El sistema verifica que esté seleccionada al menos una Evaluación para eliminar.
	24_ El sistema muestra el mensaje “Está seguro que desea eliminar esta Evaluación”
25_El Especialista acepta el mensaje.	26_El sistema elimina la Evaluación
	27_ El sistema muestra el mensaje “La evaluación fue eliminada satisfactoriamente”
28_El Especialista acepta el mensaje.	29_Se pasa a la acción 2.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
23_ Si el Especialista no selecciona la Evaluación que desea eliminar.	23.1_ El sistema muestra el mensaje: “Debe seleccionar una Evaluación para eliminar”.
23.2_El Especialista acepta el mensaje.	23.3_Se pasa a la acción 2.
26_Si el Especialista selecciona la opción de cancelar la solicitud de confirmación para eliminar la Evaluación seleccionada.	26.1_Se pasa a la acción 2.
Flujo Normal de Eventos	
Sección “Buscar Evaluación”	
Acción del Actor	Respuesta del Sistema
	31_El sistema muestra las opciones para buscar una evaluación que puede ser por: Fecha de inicio Número de control

32_El Especialista introduce las palabras claves para realizar la búsqueda en dependencia del parámetro seleccionado en la acción 31.	33_El sistema muestra la lista de evaluaciones encontradas según los criterios seleccionados.
34_El Especialista selecciona la evaluación que desea obtener o gestionar información.	35_El sistema muestra los datos de la evaluación.
36_El Especialista consulta los datos	37_El sistema pasa a la acción 2.
Flujos Alternos	
Acción del Actor	Acción del Actor
33_Si el Especialista no introduce ninguna evaluación para gestionar sus datos.	33.1_Se termina el caso de uso
34_En caso de que el sistema no encuentre evaluaciones que cumplan con el criterio solicitado.	34.1_El sistema no muestra ningún dato de la evaluación entrada por el Especialista
Prototipo de Interfaz	
Postcondiciones	Queda almacenado en la Base de datos el Plan de Evaluaciones del período señalado.

Anexo 14: Flujo Normal de Eventos del CUS Gestionar No Conformidad.

Flujo Normal de Eventos	
1_El caso de uso se inicia cuando el Evaluador Líder, selecciona la opción de Gestionar las no conformidades.	2_El sistema muestra un formulario con las opciones insertar y modificar las no conformidades.
3_El Evaluador Líder selecciona una de las opciones: Solicita Insertar las no conformidades, Ver Sección Insertar no conformidades. Solicita Modificar las no conformidades Ver Sección las Modificar no conformidades. Solicita Eliminar las no conformidades Ver Sección Eliminar no conformidades.	

Flujo Normal de Eventos

Sección “Insertar no conformidades”

Acción del Actor	Respuesta del Sistema
	4_ El sistema muestra el formulario para entrar la no conformidad.
5_ El Evaluador Líder introduce los datos de la no conformidad y pulsa el botón Insertar.	6_ El sistema muestra el mensaje “Desea insertar la no conformidad.”
7_ El Evaluador Líder acepta el mensaje	8_ El sistema valida que la información entrada fue la necesaria para completar la operación.
	9_ El sistema muestra el mensaje ” La no conformidad ha sido insertada satisfactoriamente”
10_ El Evaluador Líder acepta el mensaje	11_ Se pasa a la acción 2.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
5_ Si se deja algún campo vacío o con datos incorrectos.	5.1_ El sistema muestra el mensaje ”Debe llenar todos los campos”
5.2_ El Evaluador Líder acepta el mensaje.	5.3_ Se pasa a la acción 4.
8_ Si el Evaluador Líder selecciona la opción de cancelar para la solicitud de guardar los datos.	8.1_ Se pasa a la acción 2.

Flujo Normal de Eventos

Sección “Modificar no conformidades”

Acción del Actor	Respuesta del Sistema
	14_ El sistema valida que esté seleccionada una no conformidad de las que se le asignaron al Evaluador Líder para modificar.
15_ El Evaluador Líder solicita modificar.	16_ El sistema muestra el mensaje” ¿Está seguro que desea modificar esta no conformidad?”
17_ El Evaluador Líder acepta.	18_ El sistema muestra los datos actuales de la no conformidad.

19_ El Evaluador Líder modifica los campos que desee y pulsa el botón Modificar.	20_ El sistema muestra el mensaje” ¿Está seguro que desea modificar esta no conformidad? ”
21_ El Evaluador Líder acepta	22_ El sistema valida que la información entrada fue la necesaria para completar la operación.
	23_ El sistema muestra el mensaje “La no conformidad ha sido modificadas satisfactoriamente”.
24_ El Evaluador Líder acepta el mensaje	25_ Se pasa a la acción 2.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
17_ Si no selecciona la no conformidad.	17.1_ El sistema muestra el mensaje “Debe seleccionar una no conformidad”
17.2_ El Evaluador Líder acepta el mensaje	17.3_ Se pasa a la Acción 14
22_ Si el Evaluador Líder selecciona la opción de cancelar la acción de guardar los datos modificados.	22.1_ Se pasa a la acción 2

Flujo Normal de Eventos

Sección “ Eliminar no conformidades”

Acción del Actor	Respuesta del Sistema
26_ El Evaluador Líder selecciona la no conformidad a eliminar y pulsa el botón Eliminar.	27_ El sistema verifica que esté seleccionada al menos una no conformidad para eliminar.
	28_ El sistema muestra el mensaje “Está seguro que desea eliminar esta no conformidad”
29_ El Evaluador Líder acepta el mensaje.	30_ El sistema elimina la no conformidad
	31_ El sistema muestra el mensaje “La no conformidad fue eliminada satisfactoriamente”
	32_ Se pasa a la acción 26.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
26_ Si el Evaluador Líder no selecciona la no	26.1_ El sistema muestra el mensaje: “Debe seleccionar una

conformidad que desea eliminar.	no conformidad para eliminar”.
26.2_ El Evaluador Líder acepta el mensaje.	26.3_ Se pasa a la acción 2.
27_ Si el Evaluador Líder selecciona la opción de cancelar la solicitud de confirmación para eliminar la no conformidad seleccionada.	27.1_ Se pasa a la acción 2.

Prototipo de interfaz de Usuario

Postcondiciones	El sistema muestra los datos de la evaluación que el Evaluador Líder solicitó. Los datos de los resultados de las evaluaciones de los proyectos están insertados en la Base de Datos.
------------------------	---

Anexo 15: Flujo Normal de Eventos del CUS Gestionar evaluador.

Flujo Normal de Eventos

1_ El caso de uso se inicia cuando el Especialista, selecciona la opción de Gestionar evaluador.	2_ El sistema muestra un formulario con las opciones de insertar, modificar y buscar evaluador.
3_ El Especialista selecciona una de las opciones: Solicita Insertar evaluador, Ver Sección Insertar evaluador. Solicita Modificar evaluador, Ver Sección Modificar evaluador. Solicita Buscar evaluador, Ver Sección Buscar evaluador.	

Flujo normal de eventos

Sección “Insertar evaluador”

Acción del Actor	Respuesta del Sistema
	4_ El sistema muestra el formulario para entrar los datos de los evaluadores.
5_ El Especialista introduce los datos del evaluador y pulsa el botón Insertar.	6_ El sistema muestra el mensaje “Está seguro que desea insertar al evaluador”.
7_ El Especialista acepta el mensaje.	8_ El sistema valida que la información entrada fue la necesaria para completar la operación.

	9_ El sistema muestra el mensaje “El Evaluador ha sido insertado satisfactoriamente”.
10_ El Especialista acepta el mensaje.	11_ El sistema pasa a la acción 2.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
7_ Si el Especialista selecciona la opción de cancelar la acción de insertar los datos del evaluador.	7.1_ El sistema pasa a las acción 2.
5_ Si al llenar los datos del evaluador deja algún campo vacío.	5.1_ El sistema muestra el mensaje “Debe llenar todos los campos”.
5.2_ El Especialista acepta el mensaje.	5.3_ El sistema pasa a la acción 2.

Flujo Normal de Eventos

Sección “Modificar evaluador”

Acción del Actor	Respuesta del Sistema
11_ El Especialista selecciona un evaluador para modificar y pulsa el botón Modificar.	12_ El sistema valida que esté seleccionado un evaluador para modificar los datos.
	13_ El sistema muestra el mensaje “Está seguro que desea modificar el evaluador”.
14_ El Especialista acepta el mensaje.	15_ El sistema muestra los datos actuales del evaluador seleccionado.
16_ El Especialista modifica los campos que desee y solicita guardar los cambios.	17_ El sistema valida que la información entrada fue la necesaria para completar la operación.
	18_ El sistema muestra el mensaje “Los datos del evaluador han sido modificados satisfactoriamente”.
19_ El Especialista acepta el mensaje.	20_ El sistema pasa a la acción 15.

Flujos Alternos

Acción del Actor	Respuesta del Sistema
12_Si el Especialista no selecciona un evaluador para modificar sus datos.	12.1_El sistema muestra el mensaje: “Debe seleccionar un evaluador a modificar”.
12.2_El Especialista acepta el mensaje.	12.3_El sistema pasa a la acción 15.
17_ Si al llenar los datos del evaluador deja algún campo vacío.	17.1_El sistema muestra el mensaje “Debe llenar todos los campos”.
17.2_El Especialista acepta el mensaje.	17.3_El sistema pasa a la acción 2.

Flujo Normal de Eventos

Sección “Buscar evaluador”

Acción del Actor	Respuesta del Sistema
1_ El Especialista selecciona la opción de Buscar Evaluador.	2_El sistema muestra las opciones para buscar un Evaluador que puede ser por: Nombre. Carnet de Identidad. Solapín
3_El Especialista introduce las palabras claves para realizar la búsqueda en dependencia del parámetro seleccionado en la acción 2	4_El sistema muestra la lista de Evaluadores encontrados según los criterios seleccionados.
5_El Especialista consulta los datos del Evaluador.	

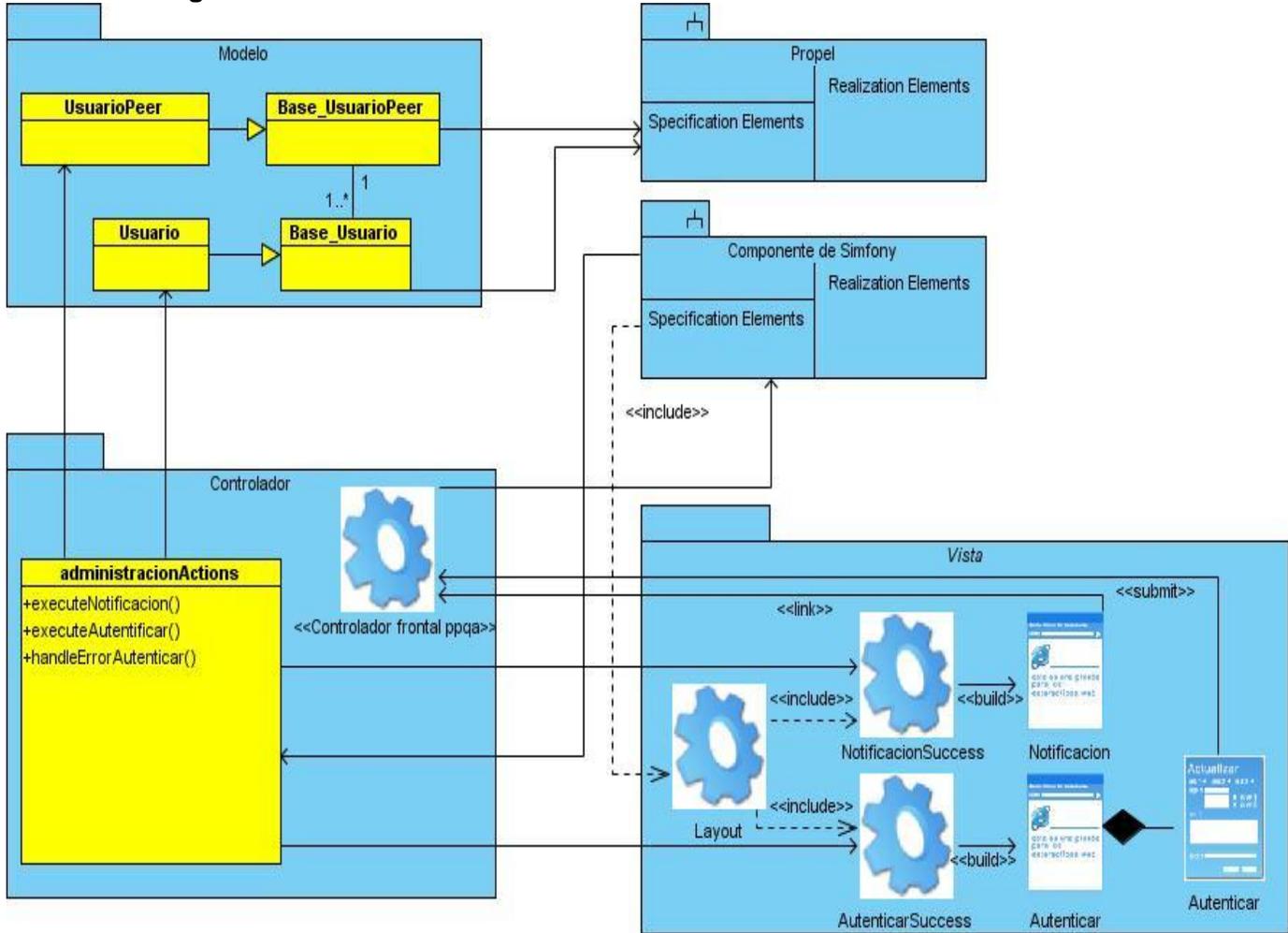
Flujos Alternos

Acción del Actor	Respuesta del Sistema
5_En caso de que el sistema no encuentre evaluadores que cumplan con el criterio solicitado.	5.1_El sistema no muestra ningún dato de algún evaluador entrado al sistema.

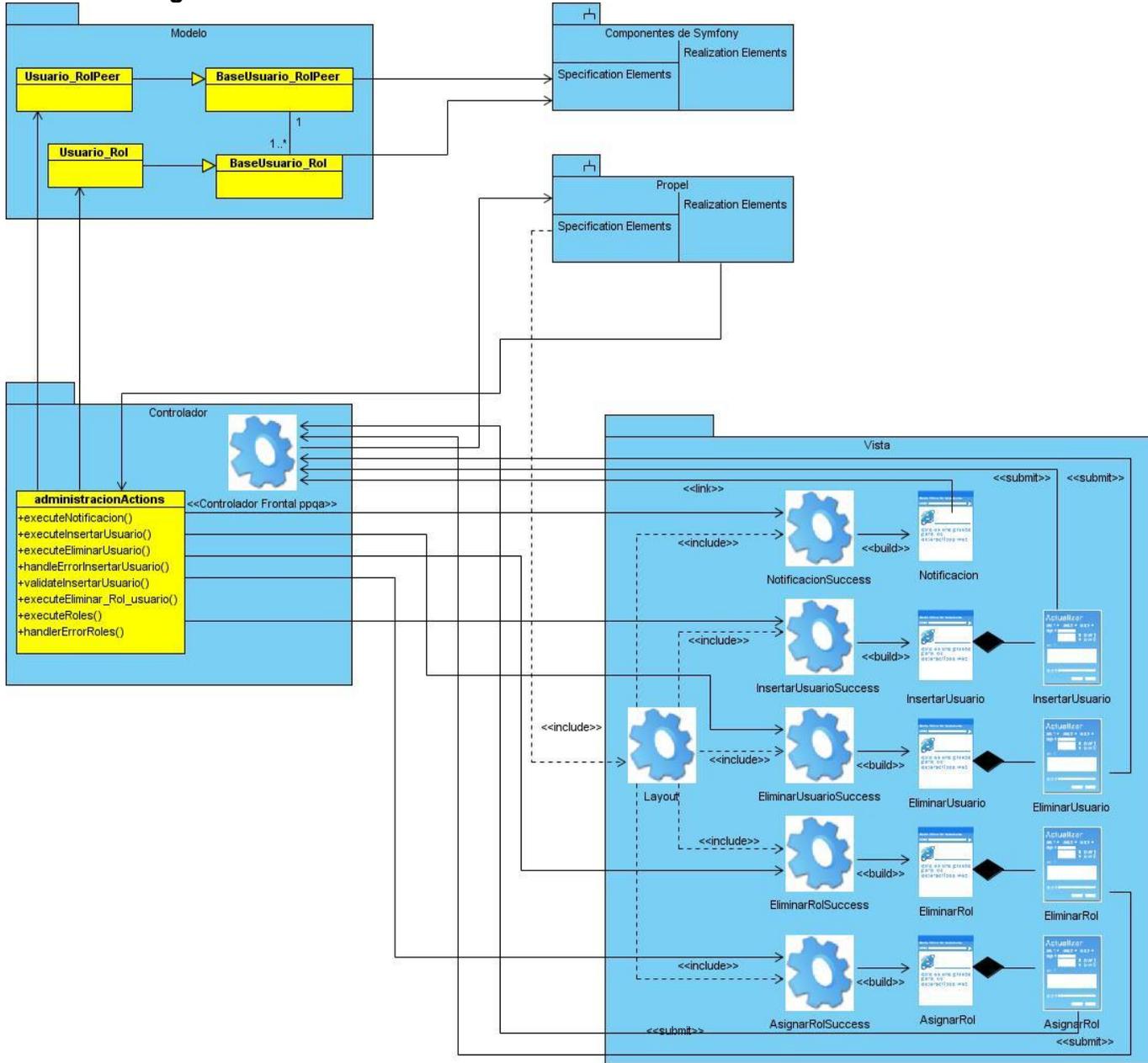
Prototipo de Interfaz

Postcondiciones	Los datos de los evaluadores fueron almacenados, modificados o buscados en la Base de Datos.
------------------------	--

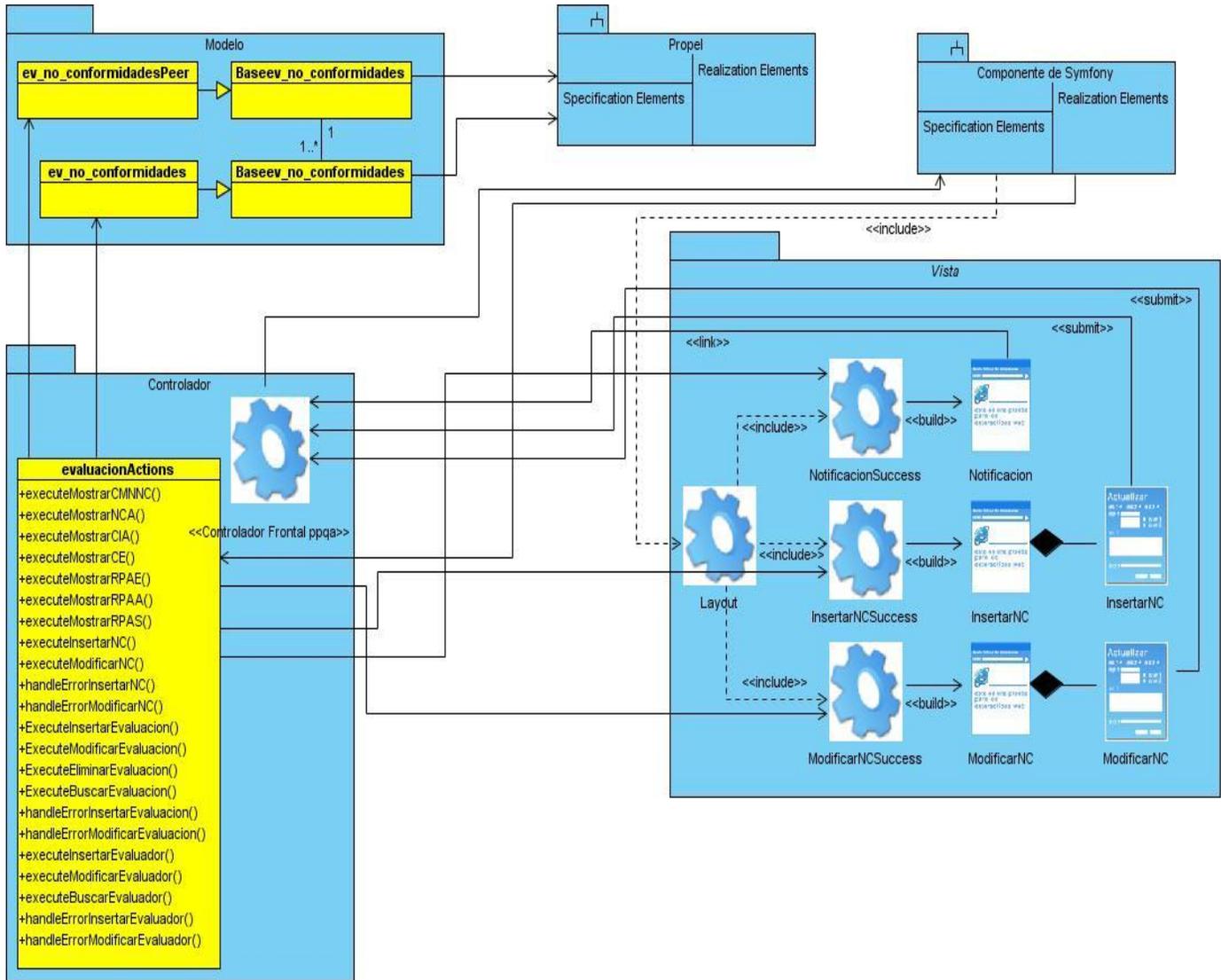
Anexo 16: Diagrama de clases del diseño CU Autenticar Usuario.



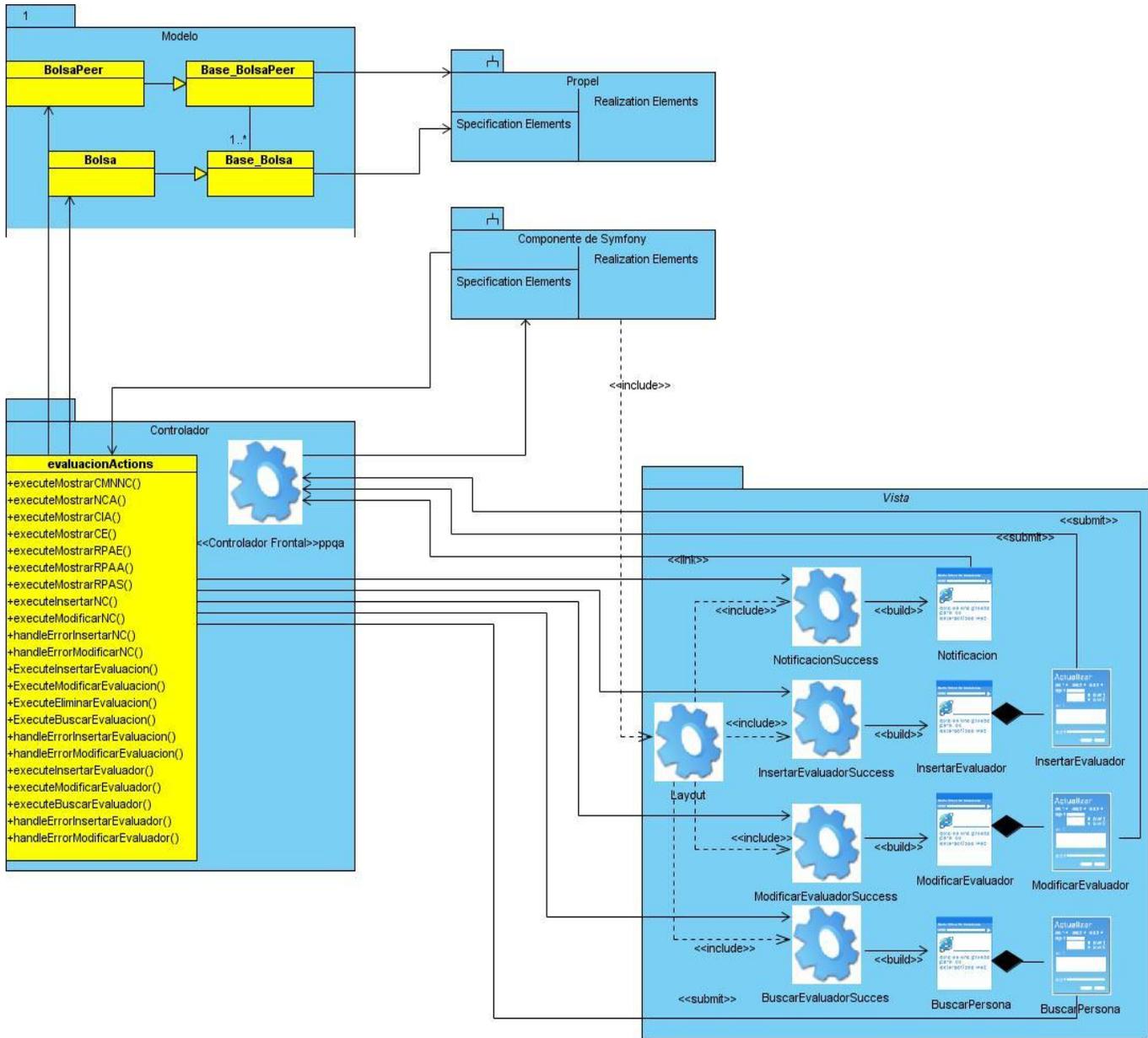
Anexo 17: Diagrama de clases del diseño CU Gestionar Usuario.



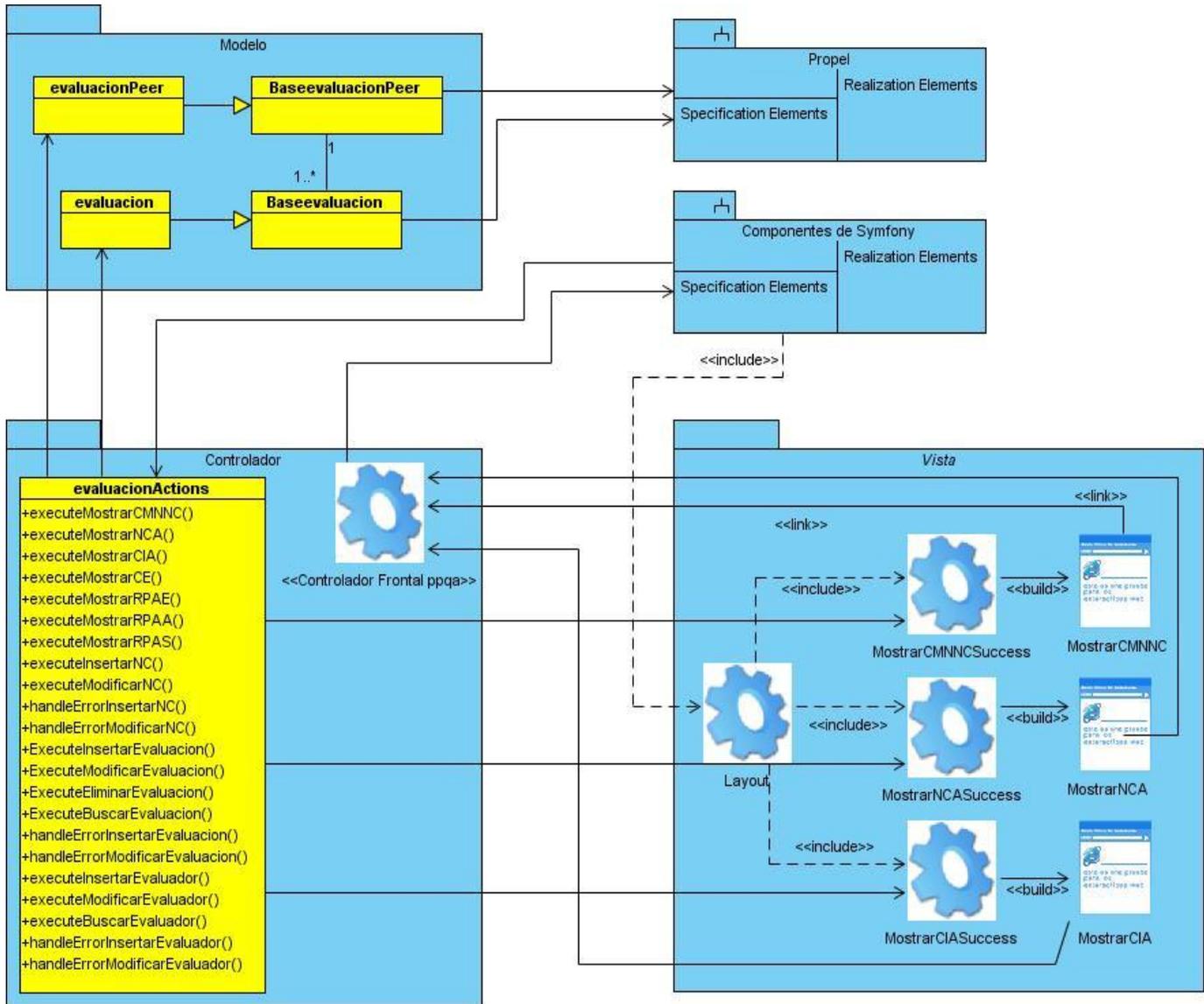
Anexo 19: Diagrama de clases del diseño CU Gestionar No conformidades.



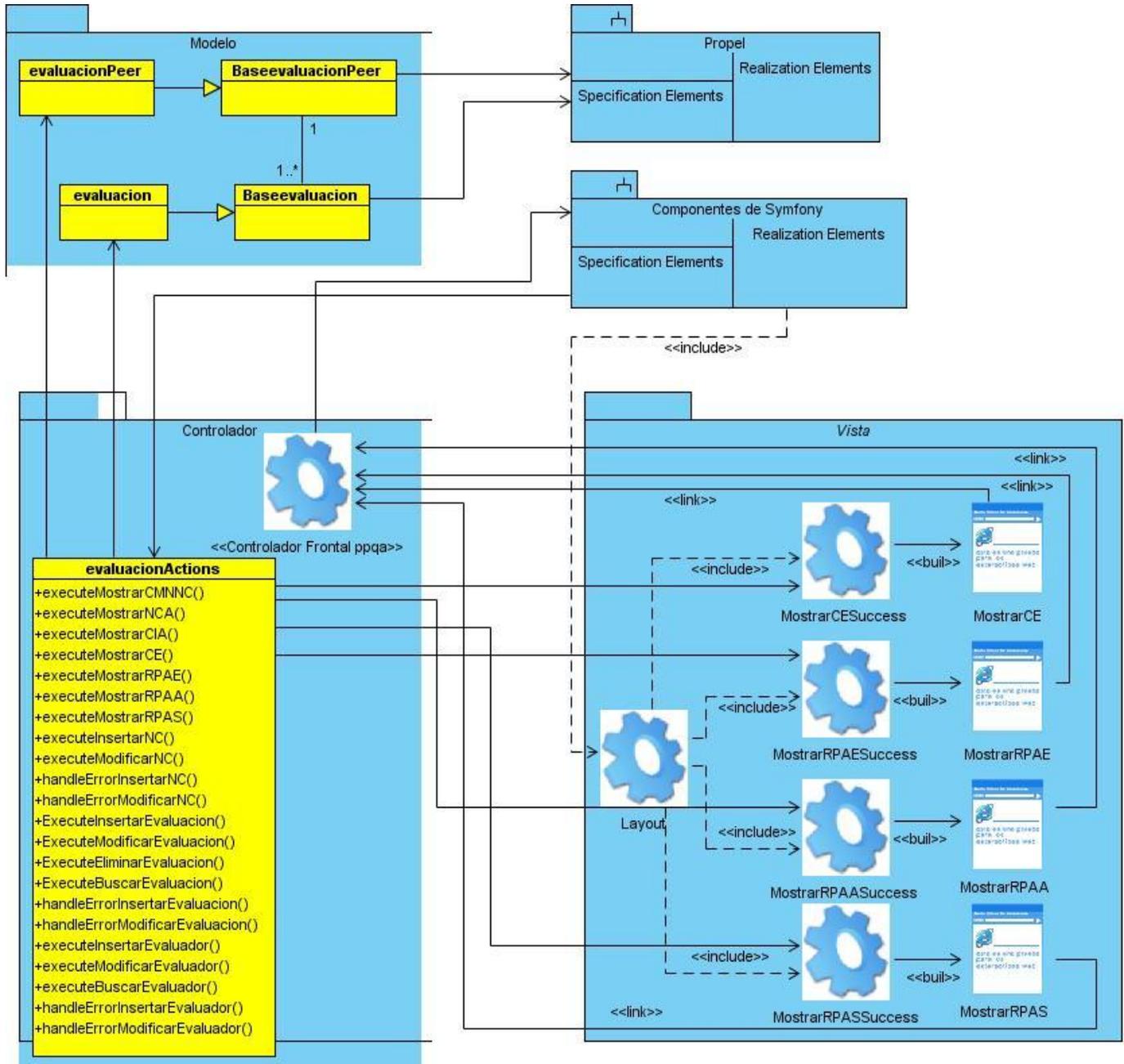
Anexo 20: Diagrama de clases del diseño CU Gestionar Evaluador.



Anexo 21: Diagrama de clases del diseño CU Reportes de No conformidades.



Anexo 22: Diagrama de clases del diseño CU Reportes Generales.



Anexo 23: Descripción de las clases de la Base de Datos.

Nombre: administracionActions Tipo de Clase: Controladora. Responsabilidades	
Nombre	executeNotificacion()
Descripción	Función que crea la página donde se muestran los errores ocurridos en las acciones de los diferentes módulos del proyecto.
Nombre	handleErrorAsignarRoles()
Descripción	Controla los errores de validación de la plantilla Asignar Roles.
Nombre	executeAsignarRoles()
Descripción	Función que se encarga de Asignar un rol a un usuario, es decir esto es lo que permite el nivel de navegabilidad por el sistema, según el rol que se le asigne será el acceso a la documentación que ésta almacenada en la aplicación.
Nombre	executeAutenticar()
Descripción	Función que se encarga se Autenticar un Usuario, si el usuario no se encuentra en la base de datos lo adicionada sino le da el acceso que ha este se le allá asignado anteriormente.
Nombre	handleErrorAutenticar()
Descripción	Controla los errores de validación de la plantilla Autenticar.
Nombre	executeEliminarRolUsuario()
Descripción	Función que se encarga de Eliminar un rol de un usuario en un momento determinado.
Nombre	executeInsertarUsuario()
Descripción	Función encargada de crear la página Insertar Usuario del sistema de gestión de la evaluación.
Nombre	executeEliminarUsuario()
Descripción	Función encargada de Eliminar un usuario dado su identificador.
Nombre	handleErrorInsertarUsuario()
Descripción	Controla los errores de validación de la plantilla Insertar Usuario.
Nombre: evaluacionActions Tipo de Clase: Controladora. Responsabilidades	
Nombre	executeInsertarEvaluacion()
Descripción	Función encargada de crear la página Evaluación del sistema de gestión de la evaluación. Además se encarga de insertar los datos de las evaluaciones en dicho sistema.
Nombre	executeModificarEvaluacion ()
Descripción	Función controladora de la acción Modificar Evaluación, encargada de modificar los datos de las evaluaciones.
Nombre	executeEliminarEvaluacion ()
Descripción	Función encargada de Eliminar una evaluación dado su identificador.
Nombre	executeBuscarEvaluador()
Descripción	Función controladora de la acción Buscar Evaluación, encargada de buscar una

	evaluación que puede ser por la fecha de inicio o por su número de control.
Nombre	handleErrorInsertarEvaluacion()
Descripción	Controla los errores de validación de la plantilla Insertar Evaluación.
Nombre	handleErrorModificarEvaluacion()
Descripción	Controla los errores de validación de la plantilla Modificar Evaluación.
Nombre	executeInsertarNC()
Descripción	Función encargada de crear la página de No Conformidades del sistema de gestión de la evaluación. Además se encarga de insertar los datos de las no conformidades en dicho sistema.
Nombre	executeModificarNC ()
Descripción	Función controladora de la acción Modificar NC, encargada de modificar los datos de las no conformidades.
Nombre	handleErrorInsertarNC()
Descripción	Controla los errores de validación de la plantilla Insertar NC.
Nombre	handleErrorModificarNC ()
Descripción	Controla los errores de validación de la plantilla Modificar NC.
Nombre	executeInsertarEvaluador()
Descripción	Función encargada de crear la página Evaluador del sistema de gestión de la evaluación. Además se encarga de insertar los datos de los evaluadores en dicho sistema.
Nombre	executeModificarEvaluador()
Descripción	Función controladora de la acción Modificar Evaluador, encargada de modificar los datos de los evaluadores.
Nombre	executeBuscarEvaluador()
Descripción	Función controladora de la acción Buscar Evaluador, encargada de buscar un evaluador dado su usuario.
Nombre	handleErrorInsertarEvaluador()
Descripción	Controla los errores de validación de la plantilla Insertar Evaluador.
Nombre	handleErrorModificarEvaluador()
Descripción	Controla los errores de validación de la plantilla Modificar Evaluador.
Nombre	handleErrorBuscarEvaluador()
Descripción	Controla los errores de validación de la plantilla Buscar Evaluador.

Anexo 24: Relación de tablas de la Base de Datos.

Módulo Administración

- Usuario
- Rol
- Usuario rol

Módulo Evaluaciones

- ev_accion_correctiva
- ev_auditoria_ejecutada.

- ev_bolsa
- ev_catalogo
- ev_escalamiento
- ev_evaluacion
- ev_evaluacion_aplazada
- ev_leccion_aprendida
- ev_no_conformidad
- ev_participante_fase
- ev_revisor_evaluacion
- ev_seguimiento

Módulo Información

- inf_directivo
- inf_persona
- inf_proyecto
- inf_proyecto_cerrado
- inf_proyecto_postmortem

Nomencladores

- n_area
- n_cargo
- n_cargo_evaluador
- n_clasificacion_accion_correctiva
- n_criterio_evaluacion
- n_curso_formacion
- n_estado_de_evaluacion
- n_estado_proyecto
- n_estado_seguimiento
- n_evaluacion_desempeno
- n_fase
- n_municipio
- n_nivel_importancia
- n_proceso

- n_producto
- n_provincia
- n_resultado_auditoría
- n_rol_evaluacion
- n_rol_proyecto
- n_tiempo_invertido
- n_tipo_accion_correctiva

Anexo 25: Descripción de la Base de Datos.

Módulo Administración

Nombre: Usuario				
Descripción: Registra los datos de los usuarios que se registran.				
No: 1				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_usuario	Serial/Integer			Llave primaria
Descripción	Text			Es el campo donde se almacena el nombre del usuario del dominio de la UCI.

Nombre: Rol				
Descripción: Registra los datos de los roles del sistema				
No: 2				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_permiso	Serial/Integer			Llave primaria
valor	Varchar(255)			Nombre del rol.
descripcion	Text	si		Breve descripción que aporta una idea de las responsabilidades del rol.

Nombre: Usuario_rol				
Descripción: Registra los datos de las relaciones que se establecen entre la tabla grupo y la tabla permiso.				
No: 3				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_usuario	Integer			Llave primaria
id_permiso	Integer			Llave primaria

Módulo Evaluaciones

Nombre: ev_accion_correctiva				
Descripción: Registra los datos de las acciones correctivas de cada no conformidad.				
No: 8				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_accion_correctiva	Serial/Interger			Llave primaria
id_seguimiento	Interger			Llave foránea
id_tipo_accion_correctiva	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea
id_clasificacion_accion_correctiva	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea
descripcion	Text	sí		Descripción de la AC para resolver la no conformidad.
fecha_ejecucion	Varchar(10)	sí		Fecha en la cual se ejecutará la AC.
responsable	Varchar(100)	sí		Máximo responsable de ejecutar la AC.

Nombre: ev_auditoría_ejecutada				
Descripción: Registra los datos de las auditorías que han sido ejecutadas.				
No: 9				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_evaluacion	Interger			Llave primaria
id_resultado_auditoría	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea

Nombre: ev_bolsa				
Descripción: Registra los datos de cada uno de los integrantes de las bolsas de revisores y expertos.				
No: 10				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_persona	Interger			Llave primaria
id_cargo_evaluador	Interger			Llave foránea
id_curso_formacion	Interger			Llave foránea
externo	TinyInt	sí	➤ 0 ➤ 1	Si vive en la UCI o no
telefono_trabajo_uci	Varchar(50)	sí		Teléfono del trabajo uci

direccion_provincia	Text	sí		Donde vive fuera de la UCI
observaciones	Varchar(100)	sí		Observaciones respecto al desempeño

Nombre: ev_catalogo				
Descripción: Registra los datos del catálogo de evaluaciones.				
No: 11				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_catalogo	Serial/Interger			Llave primaria
id_fase	Interger			Llave foránea
objetivo	Text	sí		Propósito o meta a alcanzar con la ejecución de la evaluación.
hito	Varchar(100)	sí		Descripción del período más conveniente para ejecutar la evaluación.

Nombre: ev_escalamiento				
Descripción: Registra los datos del escalamiento a las no conformidades.				
No: 12				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_escalamiento	Serial/Interger			Llave primaria
id_persona	Interger			Llave foránea
id_seguimiento	Interger			Llave foránea
fecha_gerencia	Varchar(20)	sí		Fecha en la cual una gerencia o nivel de escalabilidad resolverá la no conformidad, le dará un permiso o la escalará a un nivel superior
permiso	TinyInt	sí	> 0 > 1	Especifica si el nivel de escalabilidad otorgó un permiso para la no conformidad.
comentarios	Text	sí		Comentarios que aporten información que no se recogen en los demás elementos.

Nombre: ev_evaluacion				
Descripción: Registra los datos de las evaluaciones.				
No: 13				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_evaluacion	Serial/Interger			Llave primaria

id_catalogo	Interger			Llave foránea
id_proyecto	Interger			Llave foránea
id_estado	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea
numero_control	Varchar(10)			Identificador de la evaluación que especifica si es auditoría o revisión, el número consecutivo y el año
fecha_inicio	Varchar(10)	sí		Fecha en la que se ejecuta la reunión de inicio.
fecha_fin	Varchar(10)	sí		Fecha en la que se ejecuta la reunión de cierre.
resumen	Text	sí		Resumen descriptivo, experiencias, aclaraciones o comentarios que se deseen registrar.

Nombre: ev_evaluacion_aplazada				
Descripción: Registra los datos de las evaluaciones que estuvieron aplazadas y pasaron a otro estado.				
No: 14				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_evaluacion_aplazada	Serial/Interger			Llave primaria
id_evaluacion	Interger			Llave foránea
fecha	Varchar(10)			Fecha en la que se aplazó la evaluación.

Nombre: ev_leccion_aprendida				
Descripción: Registra los datos de las lecciones aprendidas en la ejecución de las evaluaciones.				
No: 15				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_leccion_aprendida	Serial/Interger			Llave primaria
id_persona	Interger			Llave foránea
descripcion	Varchar(100)			Describe experiencias o lecciones aprendidas en al ejecución de las evaluaciones.

Nombre: ev_no_conformidad				
Descripción: Registra los datos de las no conformidades.				
No: 16				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_no_conformidad	Serial/Interger			Llave primaria

id_persona	Interger			Llave foránea
id_evaluacion	Interger			Llave foránea
id_criterio_evaluacion	Interger			Llave foránea
id_nivel_importancia	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea
id_proceso	Interger			Llave foránea
id_producto	Interger	sí		Llave foránea
descripcion	Text	sí		Describe la no conformidad.
causa	Text	sí		Describe lo que dio lugar a la no conformidad.
fecha	Varchar(20)	sí		Fecha en la que fue identificada la no conformidad

Nombre: ev_participante_fase				
Descripción: Registra los datos de las fases del ciclo de vida de un proyecto.				
No: 17				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_participante_fase	Serial/Interger			Llave primaria
id_catalogo	Interger			Llave foránea
id_rol_proyecto	Interger			Llave foránea

Nombre: ev_revisor_evaluacion				
Descripción: Registra los datos del equipo que ejecutó una determinada evaluación.				
No: 18				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_evaluacion	Interger			Llave primaria
id_rol_evaluacion	Interger			Llave primaria
id_persona	Interger			Llave primaria
id_evaluacion_desempeño	Interger		➤ 1 ➤ 2 ➤ 3 ➤ 4	Llave foránea

Nombre: ev_seguimiento				
Descripción: Registra los datos del seguimiento a las no conformidades.				
No: 19				

Atributo	Tipo	Vacío	Restricciones	Descripción
id_seguimiento	Serial/Interger			Llave primaria
id_no_conformidad	Interger			Llave foránea
id_estado_seguimiento	Interger		➤ 1 ➤ 2	Llave foránea
id_tiempo_invertido	Interger		➤ 1 ➤ 2	Llave foránea
fecha	Varchar(10)	sí		Fecha en la que se realizó el seguimiento.

Módulo Información

Nombre: inf_directivo				
Descripción: Registra los datos de los miembros de la alta gerencia.				
No: 20				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_persona	Interger			Llave primaria
id_cargo	Interger			Llave foránea

Nombre: inf_persona				
Descripción: Registra los datos primarios de las personas.				
No: 21				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_persona	Serial/Interger			Llave primaria
id_usuario	Interger			Llave foránea
id_area	Interger			Llave foránea
id_municipio	Interger			Llave foránea
solapin	Varchar(50)	sí		Número de solapín.
nombre	Varchar(100)	sí		Nombre y Apellidos.
ci	Varchar(11)	sí		Carnet de Identidad.
telefono	Varchar(50)	sí		Número de teléfono UCI.
aptoUCI	Varchar(50)	sí		Número de apartamento UCI.

Nombre: inf_proyecto				
Descripción: Registra los datos primarios de los proyectos productivos.				
No: 22				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_proyecto	Serial/Interger			Llave primaria
id_persona	Interger			Llave foránea
id_area	Interger			Llave foránea

id_estado_proyecto	Interger		➤ 1 ➤ 2 ➤ 3	Llave foránea
codigo	Varchar(20)			Valor que determina si un proyecto es externo o no a la UCI.
nombre	Text			Nombre del proyecto.
fecha_inicio	Varchar(10)	sí		Fecha en la que dio inicio.
fecha_fin	Varchar(10)	sí		Fecha plan a terminar.
descripcion	Text	sí		Describe las principales características.

Nombre: inf_proyecto_cerrado					
Descripción: Registra los datos de los proyectos que han sido cerrados.					
No: 23					
Atributo	Tipo	Vacío	Restricciones	Descripción	
id_proyecto	Interger			Llave primaria	
fecha	Varchar(10)	sí		Fecha en la que cerró.	
razon	Text	sí		Motivos por los cuales cerró.	

Nombre: inf_proyecto_postmorten					
Descripción: Registra los datos de los proyectos que terminaron.					
No: 24					
Atributo	Tipo	Vacío	Restricciones	Descripción	
id_proyecto	Interger			Llave primaria	
fecha	Varchar(10)	sí		Fecha en la que pasó a postmorten.	
razon	Text	sí		Motivos por los cuales pasó a postmorten.	

Nomencladores

Nombre: n_area					
Descripción: Registra los datos de las áreas de la universidad, como facultades, direcciones, etc.					
No: 25					
Atributo	Tipo	Vacío	Restricciones	Descripción	
id_area	Serial/Interger			Llave primaria	
valor	Varchar(100)				

descripcion	Text	sí		
Nombre: n_cargo				
Descripción: Registra los datos de los cargos que puede ocupar la alta gerencia.				
No: 26				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_cargo	Serial/Interger			Llave primaria
descripcion	Varchar(100)			
nivel_escalamiento	Interger	sí	<ul style="list-style-type: none"> ➤ 1 ➤ 2 ➤ 3 ➤ 4 ➤ 5 	

Nombre: n_cargo_evaluador				
Descripción: Registra los datos de los compromisos que puede tener un evaluador. Ejemplo si es profesor, experto, etc.				
No: 27				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_cargo_evaluador	Serial/Interger			Llave primaria
descripcion	Varchar(100)			

Nombre: n_clasificacion_accion_correctiva				
Descripción: Registra los datos de las clasificaciones de las acciones correctivas.				
No: 28				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_clasificacion_accion_correctiva	Serial/Interger			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ correctiva ➤ preventiva ➤ de mejora 	

Nombre: n_criterio_evaluacion				
Descripción: Registra los datos de los criterios de la evaluación.				
No: 29				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_criterio_evaluacion	Serial/Interger			Llave primaria
descripcion	Text			

Nombre: n_curso_formacion				
Descripción: Registra los datos de los cursos de formación.				
No: 30				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_curso_formacion	Serial/Integer			Llave primaria
descripcion	Varchar(100)			
fecha_inicio	Varchar(10)	sí		
fecha_fin	Varchar(10)	sí		

Nombre: n_estado_de_evaluacion				
Descripción: Registra los datos de los estados en los que puede estar una evaluación.				
No: 31				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_estado	Serial/Integer			Llave primaria
descripcion	Varchar(50)	sí	<ul style="list-style-type: none"> ➤ ejecutada ➤ aplazada ➤ abortada 	

Nombre: n_estado_proyecto				
Descripción: Registra los datos de los estados en los que puede estar un proyecto.				
No: 32				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_estado_proyecto	Serial/Integer			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ abierto ➤ cerrado ➤ postmorten 	

Nombre: n_estado_seguimiento				
Descripción: Registra los datos de los estados en los que puede estar una no conformidad.				
No: 33				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_estado_seguimiento	Serial/Integer			Llave primaria
descripcion	Varchar(50)		<ul style="list-style-type: none"> ➤ abierta ➤ cerrada 	

Nombre: n_evaluacion_desempeno				
Descripción: Registra los datos de las evaluaciones que puede tener un desempeño.				
No: 34				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_evaluacion_desempeno	Serial/Interger			Llave primaria
descripcion	Varchar(200)		<ul style="list-style-type: none"> ➤ excelente ➤ bien ➤ regular ➤ mal 	

Nombre: n_fase				
Descripción: Registra los datos de las fases del ciclo de vida de un proyecto.				
No: 35				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_fase	Serial/Interger			Llave primaria
descripcion	Varchar(100)			

Nombre: n_municipio				
Descripción: Registra los datos de los municipios.				
No: 36				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_municipio	Serial/Interger			Llave primaria
id_provincia	Interger			
descripcion	Varchar(100)	sí		

Nombre: n_nivel_importancia				
Descripción: Registra los datos de los niveles de importancia que tiene una no conformidad.				
No: 37				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_nivel_importancia	Serial/Interger			Llave primaria
descripcion	Varchar(50)		<ul style="list-style-type: none"> ➤ alta ➤ media ➤ baja 	

Nombre: n_proceso				
Descripción: Registra los datos de los procesos a evaluar por cada fase.				

No: 38				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_proceso	Serial/Interger			Llave primaria
id_fase	Interger			
descripcion	Varchar(100)			

Nombre: n_producto				
Descripción: Registra los datos de los productos a evaluar por cada uno de los procesos y fases.				
No: 39				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_producto	Serial/Interger			Llave primaria
id_proceso	Interger			
descripcion	Varchar(100)			

Nombre: n_provincia				
Descripción: Registra los datos de las provincias.				
No: 40				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_provincia	Serial/Interger			Llave primaria
descripcion	Varchar(100)			

Nombre: n_resultado_auditoría				
Descripción: Registra los datos de las evaluaciones a las auditorías como resultado de la misma.				
No: 41				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_resultado_auditoria	Serial/Interger			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ satisfactorio ➤ aceptable ➤ deficiente 	

Nombre: n_rol_evaluacion				
Descripción: Registra los datos de los roles que se pueden desempeñar dentro de un equipo evaluador.				
No: 42				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_rol_evaluacion	Serial/Interger			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ revisor líder ➤ revisor 	

			<ul style="list-style-type: none"> ➤ auditor líder ➤ auditor 	
--	--	--	--	--

Nombre: n_rol_proyecto				
Descripción: Registra los datos de los roles que se pueden desempeñar dentro de un proyecto.				
No: 43				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_rol_proyecto	Serial/Interger			Llave primaria
descripcion	Varchar(100)			

Nombre: n_tiempo_invertido				
Descripción: Registra los datos del tiempo invertido para aplicar la acción correctiva.				
No: 44				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_tiempo_invertido	Serial/Interger			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ atrasada ➤ en fecha 	

Nombre: n_tipo_accion_correctiva				
Descripción: Registra los datos de los tipos de acciones correctivas.				
No: 45				
Atributo	Tipo	Vacío	Restricciones	Descripción
id_tipo_accion_correctiva	Serial/Interger			Llave primaria
descripcion	Varchar(100)		<ul style="list-style-type: none"> ➤ capacitación ➤ formación ➤ toma de medidas 	

Anexo 26: Caso de Prueba del CU_Autenticar usuario.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
----------------------	--------------------------	---------------------------------	---------------

SC 1: Autenticar Usuario	EC 1.1: Autenticar Usuario	El usuario ingresa usuario y contraseña en el dominio UCI y pulsa el botón Aceptar, el sistema debe ingresarlo satisfactoriamente.	El usuario introduce los datos, el sistema verifica que los datos estén en el dominio de la UCI para poder acceder satisfactoriamente.
	EC 1.2: Campo Vacío	El usuario ingresa usuario y contraseña en el dominio UCI, pero deja algún campo vacío y pulsa el botón Aceptar, el sistema debe ser capaz de detectarlo.	El usuario al ingresar usuario y contraseña en el dominio UCI deja algún campo vacío y pulsa el botón Aceptar, el sistema muestra un mensaje diciéndole que dejó algún campo vacío.
	EC 1.3: datos incorrectos	El usuario ingresa usuario y contraseña en el dominio UCI, pero lo realiza de forma incorrecta y pulsa el botón Aceptar, el sistema debe ser capaz de detectarlo.	El usuario ingresa usuario y contraseña en el dominio UCI, pero de forma incorrecta y pulsa el botón Aceptar, el sistema muestra un mensaje diciéndole que revise el usuario y la contraseña, que están incorrectos.

Id del escenario	Escenario	Variable 1 (Nombre de la variable)	Variable 2 (Nombre de la variable)	Variable N (Nombre de la variable)	Respuesta del Sistema	Resultado de la Prueba
1.1	Autenticar Usuario	N/A			El sistema debe permitir acceder satisfactoriamente si el usuario está en el dominio UCI.	Satisfactorio
1.2	Campo Vacío	N/A			El sistema debe detectar si el usuario dejó algún campo vacío.	Satisfactorio

1.3	Datos incorrectos	N/A			El sistema debe detectar si el usuario o la contraseña están incorrectos.	Satisfactorio
-----	-------------------	-----	--	--	---	---------------

Anexo 27: Caso de Prueba del CU_Gestionar usuario.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 2: Gestionar Usuario	EC 2.1: Asignar Roles	El Administrador selecciona la opción de asignar roles, el sistema debe configurarse según el rol asignado.	El Administrador selecciona la opción de asignar roles, el sistema muestra la lista de todos los roles que puede tener un usuario y los usuarios que hay registrados en el sistema y el Administrador selecciona un usuario y un rol para asignárselo al usuario escogido, el sistema le da el acceso según el rol que tenga.
	EC 2.2: Eliminar Rol Usuario	El Administrador selecciona la opción de Eliminar Rol Usuario y el sistema debe permitir eliminar el rol que tiene el usuario.	El Administrador, selecciona la opción de Eliminar Rol Usuario, el sistema muestra un formulario con los roles que tiene asignado el usuario seleccionado, el administrador selecciona el rol que le quiere eliminar al usuario y pulsa el botón Eliminar y el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.

	EC 2.3: Campo Vacío	Es cuando el Administrador desea asignar rol pero deja uno de los campos en blanco, donde debe escoger el usuario o el rol.	El Administrador selecciona la opción de asignar roles o eliminar rol usuario, el sistema muestra la lista de todos los roles que tiene o puede tener un usuario y los usuarios o roles que hay registrados en el, en dependencia de la funcionalidad que escogió, el Administrador deja uno de los campos en blanco, el sistema muestra un mensaje que especifica que tiene algún campo vacío.
--	---------------------	---	---

Id del escenario	Escenario	Variable 1 (Nombre de la variable)	Variable 2 (Nombre de la variable)	Variable N (Nombre de la variable)	Respuesta del Sistema	Resultado de la Prueba
EC 2.1	Asignar Roles.	N/A			El sistema le da el acceso a la información según el rol que tenga el usuario.	Satisfactorio
EC 2.2	Eliminar Rol Usuario	N/A			El sistema elimina el rol del usuario seleccionado	Satisfactorio
EC 2.3	Campo vacío	N/A			El sistema detecta si hay algún campo vacío	Satisfactorio

Anexo 28: Caso de Prueba del CU_Gestionar evaluación.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
----------------------	--------------------------	---------------------------------	---------------

SC 3: Gestionar Evaluación	EC 3.1: Insertar Evaluación.	El Especialista selecciona la opción de Insertar Evaluación, el sistema debe permitir insertar la evaluación correctamente.	El Especialista selecciona la opción de Insertar Evaluación, el sistema muestra un formulario para entrar los datos de la evaluación, el Especialista introduce los datos y pulsa el botón Insertar, el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.
	EC 3.2: Modificar Evaluación.	El Especialista selecciona la opción de Modificar Evaluación, el sistema debe permitir modificar la evaluación correctamente.	El Especialista selecciona la opción de Modificar Evaluación, el sistema muestra un formulario para seleccione la evaluación que desee, el Especialista selecciona la evaluación y pulsa el botón Modificar, el sistema muestra un formulario con todos los datos de la evaluación para que el Especialista modifique el que desee, este modifica los datos y pulsa el botón Modificar, el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.

EC 3.3: Eliminar Evaluación	El Especialista selecciona la opción de Eliminar Evaluación, el sistema debe permitir Eliminar la evaluación correctamente.	El Especialista selecciona la opción de Eliminar Evaluación, el sistema muestra un formulario para seleccione la evaluación que desee, el Especialista selecciona la evaluación y pulsa el botón Eliminar, el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.
EC 3.4: Buscar Evaluación	El Especialista selecciona la opción de Buscar Evaluación, el sistema debe permitir buscar las evaluaciones correctamente.	El Especialista selecciona la opción de Buscar Evaluación, el sistema muestra un formulario para introduzca el dato por el que desea buscar la evaluación (por fecha de inicio o número de control), el Especialista introduce los datos y a medida que esto sucede el sistema le muestra las evaluaciones que contiene esos datos.
EC 3.5: Campo vacío	El Especialista selecciona cualquiera de las opciones Insertar, Modificar, Eliminar o Buscar Evaluación, pero a la hora de ingresar los datos o buscar la evaluación deja algún campo vacío, el sistema debe detectarlo.	El Especialista selecciona cualquiera de las opciones Insertar, Modificar, Eliminar o Buscar Evaluación, El sistema le muestra el formulario correspondiente a la opción que escogió, el Especialista a la hora de llenar los formularios con los datos o buscar la evaluación deja algún campo vacío, el sistema debe detectarlo y mostrar un mensaje diciéndole que ha dejado algún campo vacío.

Id del escenario	Escenario	Variable 1 (Nombre de la variable)	Variable 2 (Nombre de la variable)	Variable N (Nombre de la variable)	Respuesta del Sistema	Resultado de la Prueba
EC 3.1	Insertar Evaluación.	N/A			El sistema permite insertar la evaluación correctamente.	Satisfactoria
EC 3.2	Modificar Evaluación	N/A			El sistema permite modificar los datos de la evaluación seleccionada por el Especialista de manera correcta.	Satisfactoria
EC 3.3	Eliminar Evaluación	N/A			El sistema permite eliminar la evaluación seleccionada por el Especialista, correctamente.	Satisfactoria
EC 3.4	Buscar Evaluación	N/A			El sistema permite mostrar todas las evaluaciones que concuerden con los datos introducidos por el Especialista, de forma correcta.	Satisfactoria
EC 3.5	Campo vacío	N/A			El sistema detecta si el Especialista dejó algún campo vacío al introducir los datos de la evaluación.	Satisfactoria

Anexo 29: Caso de Prueba del CU_Gestionar evaluador.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 4: Gestionar Evaluador	EC 4.1: Insertar Evaluador	El Especialista selecciona la opción de Insertar Evaluador y el sistema debe permitir realizar la acción correctamente.	El Especialista selecciona la opción de Insertar Evaluador, el sistema le muestra un formulario para entrar los datos del evaluador, el Especialista introduce los datos y pulsa el botón Insertar, el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.
	EC 4.2: Modificar Evaluador	El Especialista selecciona la opción de Modificar Evaluador	El Especialista selecciona la opción de Modificar Evaluador, el sistema muestra un formulario para seleccione al evaluador que desee, el Especialista selecciona el evaluador y pulsa el botón Modificar, el sistema muestra un formulario con todos los datos del evaluador para que el Especialista modifique lo que desee, el Especialista modifica los datos y pulsa el botón Modificar, el sistema muestra un mensaje diciéndole que se ha realizado la acción correctamente.

	EC 4.3: Buscar Evaluador.	El Especialista selecciona la opción de Buscar Evaluador	El Especialista selecciona la opción de Buscar Evaluador, el sistema muestra un formulario para introduzca el dato por el que desea buscar al evaluador (teléfono o dirección), el Especialista introduce los datos y a medida que esto sucede el sistema le muestra los evaluadores que presentan esos datos.
	EC 4.4: Campo vacío	El Especialista selecciona cualquiera de las opciones Insertar, Modificar o Buscar Evaluador, pero a la hora de ingresar los datos o buscar al evaluador deja algún campo vacío, el sistema debe detectarlo.	El Especialista selecciona cualquiera de las opciones Insertar, Modificar o Buscar Evaluador, el sistema le muestra el formulario correspondiente a la opción que escogió, el Especialista a la hora de llenar los formularios con los datos o buscar al evaluador deja algún campo vacío, el sistema debe detectarlo y mostrar un mensaje diciéndole que ha dejado algún campo vacío.

Id del escenario	Escenario	Variable 1 (Nombre de la variable)	Variable 2 (Nombre de la variable)	Variable N (Nombre de la variable)	Respuesta del Sistema	Resultado de la Prueba
EC 4.1	Insertar Evaluador	N/A			El sistema permite insertar al evaluador correctamente.	Satisfactorio

EC 4.2	Modificar Evaluador	N/A			El sistema permite modificar los datos del evaluador seleccionado por el Especialista de manera correcta.	Satisfactorio
EC 4.3	Buscar Evaluador.	N/A			El sistema permite mostrar todos los evaluadores que concuerden con los datos introducidos por el Especialista, de forma correcta.	Satisfactorio
EC 4.4	Campo vacio				El sistema detecta si el Especialista dejó algún campo vacio al introducir los datos del evaluador.	Satisfactorio

Anexo 30: Caso de Prueba del CU_Gestionar no conformidad.

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central
SC 5: Gestionar No Conformidad	EC 5.1: Insertar No Conformidad	El Evaluador Líder, selecciona la opción de Insertar No Conformidad del sistema debe permitir realizar la acción correctamente.	El Evaluador Líder, selecciona la opción de Insertar No Conformidad, el sistema muestra un formulario para que entre los datos de la no conformidad, el Evaluador Líder introduce los datos y pulsa el botón Insertar, el

	EC 5.2: Modificar No Conformidad	El Evaluador Líder, selecciona la opción de Modificar No Conformidad, el sistema debe permitir realizar la acción correctamente.	El Evaluador Líder, selecciona la opción de Modificar No Conformidad, el sistema muestra un formulario para que busque la no conformidad que desea modificar, el Evaluador Líder introduce el dato y pulsa el botón Modificar, el sistema le muestra un formulario con todos los datos
	EC 5.3: Campo vacío	El Evaluador Líder, selecciona cualquier opción ya sea Insertar o Modificar No Conformidad, el sistema debe detectar si dejó algún campo vacío al introducir o modificar los datos.	El Evaluador Líder, selecciona cualquier opción ya sea Insertar o Modificar No Conformidad, el sistema le muestra el formulario correspondiente a la opción seleccionada, el Evaluador Líder al introducir o modificar los datos deja algún campo vacío y pulsa el botón Modificar, el sistema le muestra un mensaje diciéndole que dejó algún campo vacío.

Id del escenario	Escenario	Variable 1 (Nombre de la variable)	Variable 2 (Nombre de la variable)	Variable N (Nombre de la variable)	Respuesta del Sistema	Resultado de la Prueba
EC 5.1	Insertar No Conformidad	N/A			El sistema permite insertar la no conformidad correctamente.	Satisfactorio

EC 5.2	Modificar No Conformidad	N/A			El sistema permite modificar los datos de la no conformidad seleccionada por el Evaluador Líder de manera correcta.	Satisfactorio
EC 5.3	Campo vacío	N/A			El sistema detecta si el Evaluador Líder dejó algún campo vacío al introducir los datos de la no conformidad.	Satisfactorio

