

Universidad de las Ciencias Informáticas

Facultad 4



Trabajo de diploma para optar por el Título de
Ingeniero en Ciencias Informáticas.

**Título: Desarrollo del Módulo de Capital Humano para el
Sistema de Gestión de Evaluaciones de Software de la
UCI.**

Autor(es): Santiago Marrero Giró

Tutor(es): Ing. Ana Margarita Abreu Sáez

Ing. Maikel Muñoz Roja

Ciudad de La Habana, Junio de 2011

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los ____ días del mes de Junio del año 2011.

Santiago Marrero Giró

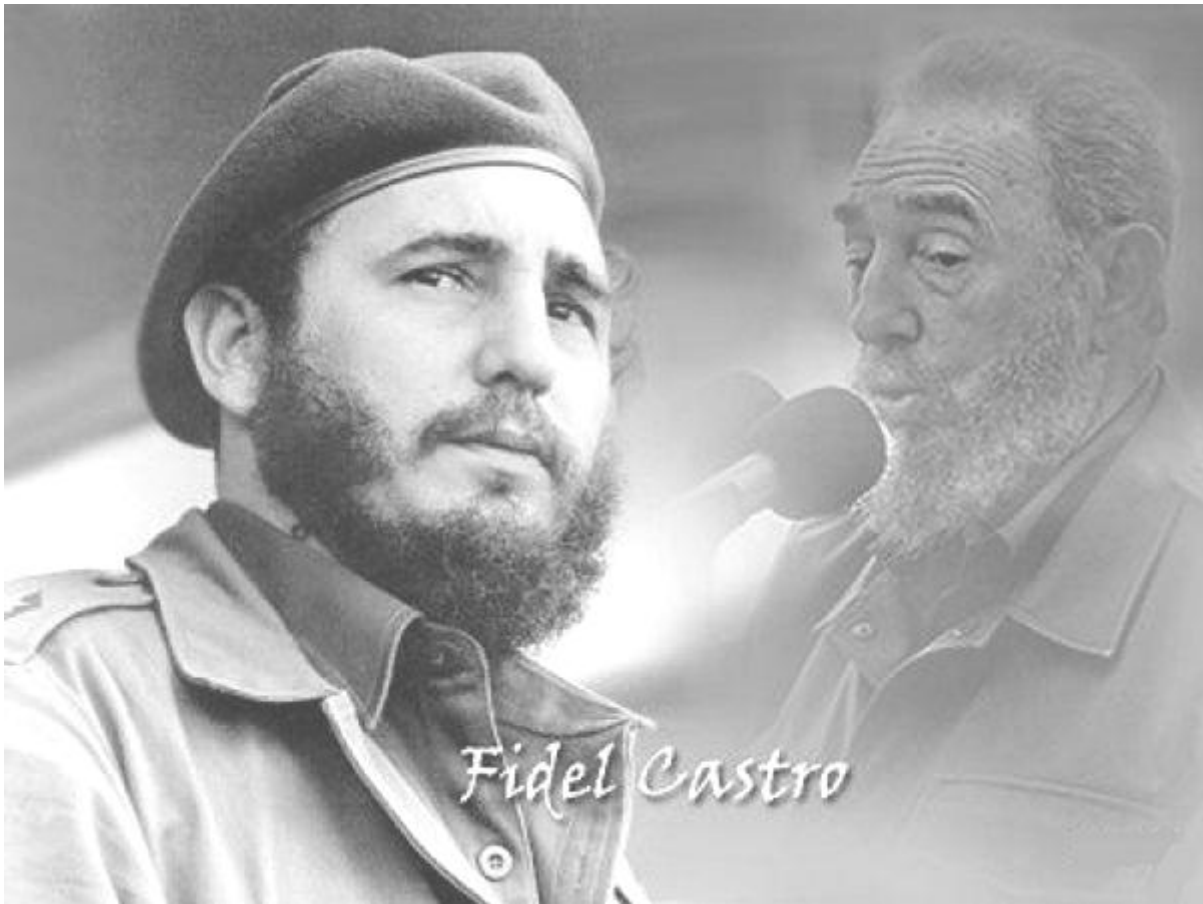
Firma del Autor

Ing. Ana Margarita Abreu Sáez

Firma del Tutor

Ing. Maikel Muñoz Roja

Firma del Tutor



“Universidad de excelencia es una idea de hacer lo mejor que pueda hacerse, es desarrollar la mejor universidad y podemos hacerlo buscando lo óptimo, lo más perfecto posible dentro de las cosas humanas, lo más nuevo, lo más creativo, algo que no sólo sirva a los intereses de nuestro país”.

Agradecimientos:

Agradecer ante todo a mi querida mamá Ana, mis abuelitos Iris y Eloy por sus consejos, por su sacrificio, por todo el apoyo y amor que me han brindado en toda mi vida, por haber sabido guiarme por el camino correcto, en fin por hacer de mí el hombre que hoy soy. A mi papá por sus consejos y su confianza en mí. Este trabajo es la recompensa de sus años de esfuerzo y dedicación, gracias por creer en mí, los quiero mucho.

A mi hermanita Arianne por formar parte de mí en todo momento.

A mi novia Yadira por haberme apoyado mucho y por el amor y cariño que me brinda. A su madre Augusta por estar siempre preocupada por mí y apoyarme en todo.

A mis tíos: Tamara, Jorge Luis, Eloy, Ramón, en fin a todos por estar al tanto siempre que en mis estudios todo marchara bien.

A mis amistades de la UCI en especial a: mi hermano Alejandro Alvarez, por su incondicional apoyo durante la realización de este trabajo.

A mis tutores Ana Margarita y Maiquel Muñoz por su paciencia y ayuda oportuna siempre que los necesité.

A mis amigos que compartieron estos cinco hermosos años de la carrera.

A todos los profesores que he tenido en el transcurso de toda mi vida, que han confiado en mí.

A todos los que de una forma u otra contribuyeron a mi formación como ingeniero informático.

A la Revolución por haber creado este proyecto.

A todos muchas gracias.

Dedicatoria:

A mi mamá, mis abuelos, mis tíos por su dedicación, por sus consejos, por su cariño y sobre todo por creer en mí.

A mi hermana por ser la luz de mi vida.

A mi novia por darme su amor y estar junto a mí brindándome su apoyo y confianza.

Resumen

En la sociedad actual, autodenominada “sociedad de la información y las tecnologías”, la gestión del Capital Humano constituye la disciplina que se ocupa de la identificación, almacenamiento, recuperación, compartimiento y evaluación del conocimiento organizacional. El término Capital Humano no es más que una forma de capital que indica el aprovechamiento al máximo del aporte humano al que como activo intangible se le asigna un valor y se considera como capital. El rendimiento actual del empleado ceñido a un trabajo específico ha sido siempre la herramienta estándar que mide la productividad de un trabajador en una determinada empresa. La Gestión del Capital Humano busca enfocarse en el potencial del empleado, lo que implica su desempeño futuro si se fomenta el desarrollo apropiado de habilidades. El presente trabajo de diploma tiene como objetivo fundamental desarrollar el Módulo de Gestión de Capital Humano para el Sistema de la Gestión de Evaluaciones de Software del Centro de Calidad para Soluciones Informáticas (CALISOFT). A partir del software implementado se permitirá gestionar el Capital Humano que allí labora, evaluar el desempeño del personal, gestionar las habilidades de cada revisor auditor, los cursos de capacitación que se imparten por la entidad, generar reportes de evaluaciones, entre otras funcionalidades específicas.

Para cumplir con los objetivos del presente Trabajo de Diploma se describió la solución partiendo del diseño de la aplicación y se detalla la implementación. Finalmente, se implementó el diseño definido y se realizaron pruebas a las funcionalidades exigidas, arrojando resultados satisfactorios que validaron las tecnologías utilizadas y el modelado propuesto.

Palabras claves: Capital Humano, Evaluación de software, Sistema de gestión, Software

Índice

Introducción.....	1
CAPÍTULO I.....	5
FUNDAMENTACIÓN TEÓRICA.....	5
Introducción.....	5
1.1 Definición del Capital Humano.....	5
1.2 Componentes del Capital Humano.....	6
1.3 Gestión del Capital Humano.....	7
1.4 Software de Gestión.....	8
1.5 Medición de los Resultados de las Iniciativas del Capital Humano.....	8
1.6 Caracterización del Sistema de Evaluaciones de Software (GESOFT).....	9
1.7 Herramientas y Tecnologías para la Construcción del Software.....	10
1.8 Dependencias y Restricciones del Cliente.....	10
Lenguaje de Programación PHP 5.0.....	10
Framework Symfony.....	11
EXT.JS 3.0.....	12
Sistema Gestor de Base de Datos PostgreSQL 3.8.....	12
Proceso Unificado de Desarrollo (RUP).....	13
Lenguaje de Modelado UML.....	14
1.9 Herramientas para el Modelado de la Aplicación Visual Paradigm.....	14
1.10 Entorno de Desarrollo Integrado NetBeans 6.9.....	14
Conclusiones.....	15
CAPÍTULO II.....	16
Introducción.....	16
2.1 Modelo de Negocio.....	16
2.1.1 Procesos del Negocio.....	16
2.1.2 Entidades del Negocio.....	17
2.1.3 Actores del Negocio.....	18
2.1.4 Trabajadores del Negocio.....	18
2.1.5 Diagrama de Caso de Uso del Negocio.....	19
2.1.6 Modelo de Objetos del Negocio.....	21
2.1.7 Reglas del Negocio.....	22
2.2 Captura de Requisitos.....	22
2.2.1 Técnicas Empleadas para la Captura de Requisitos.....	22

2.2.2	Requerimientos Funcionales.....	23
2.2.3	Descripción Detallada de los Requerimientos Funcionales. Ver: Anexo I.....	23
2.2.4	Requisitos No Funcionales (RNF)	23
2.2.5	Descripción del Sistema Propuesto.....	25
2.3	Actores del Sistema.....	26
2.3.1	Diagrama de Casos de Uso del Sistema	26
2.3.2	Descripción de los Casos de Uso del Sistema.	27
	Conclusiones.....	29
CAPÍTULO III		30
Introducción		30
3.1	Análisis	30
	Modelo de Análisis	30
3.1.1	Diagramas de Clases de Análisis.....	31
3.1.2	Diagramas de Interacción.....	32
3.2	Diseño	34
	Modelo del Diseño	34
3.2.2	Diagramas de Clases del Diseño.	35
3.3	Diseño de la Base de Datos.....	36
3.3.1	Descripción de las Tablas.	37
3.4	Patrones de Diseño y de Arquitectura.....	40
	Conclusiones.....	43
CAPÍTULO IV.....		44
Introducción		44
4.1	Diagrama de Componentes	44
4.2	Diagrama de Despliegue	45
4.3	Modelo de Prueba	45
4.3.1	Pruebas de Caja Negra.....	45
4.3.2	Casos de Prueba (CP)	46
4.4	Pruebas de Aceptación.....	54
4.5	Resultados de las Pruebas	54
4.6	Impacto del Módulo Desarrollado en la Eficiencia de los Procesos de Auditorías y Revisiones de CALISOFT	55
	Conclusiones.....	57
Conclusiones Generales		57
Recomendaciones		58

Glosario de Términos.....	59
Referencias Bibliográficas.....	60

Introducción.

La Industria de las Tecnologías de la Informática y las Comunicaciones están fuertemente sujetas al desarrollo económico de un país. Bajo estas mismas condiciones se encuentra la Industria del Software. El mundo de hoy está inmerso en una nueva revolución tecnológica, en la cual existen notables desigualdades. Cuba constituye un ejemplo singular, país que aunque se encuentra bloqueado financieramente aspira a convertirse en una potencia productora de software.

Para elaborar un producto que cumpla con las necesidades del cliente, diversas empresas definen sus propias metodologías y procesos, la gestión del Capital Humano es uno de ellos, la cual tiene como objetivo primordial mantener, desarrollar y gestionar el personal necesario para cumplir de manera eficaz y eficiente los objetivos de la institución.

El conocimiento y las habilidades del Capital Humano han cobrado una gran relevancia, siendo cada día más importantes en comparación con otros activos tangibles. De allí que los recursos humanos se hayan convertido en una parte de vital significación para el éxito de las organizaciones.

La gestión del capital humano de una empresa tiene como objetivo fundamental proveer a la empresa de personal suficiente tanto en calidad y con la motivación suficiente.

Las instituciones en Cuba, se ven sometidas a retos, desafíos y presiones, a los cuales tienen que responder con alto grado de creatividad y realismo. Una de las tareas que se ha trazado el país como estrategia de desarrollo económico, teniendo como base el alto nivel de su capital intelectual, ha sido perfeccionar la industria del software y convertirla en uno de los principales renglones de la economía, con el objetivo de incurrir en menos costos, sustituyendo de esta manera las importaciones de países extranjeros.

En el país son varias las empresas que se dedican profesionalmente a la producción de software y soluciones informáticas, ya sea para la propia entidad o con fines comerciales o de exportación; las universidades se incluyen en este grupo. El aseguramiento de la calidad del proceso y el producto de estos sistemas informáticos es un aspecto esencial.

La Universidad de las Ciencias Informáticas pretende ser la vanguardia en el desarrollo de la industria del software en Cuba. La misma se encuentra envuelta en proyectos de producción de software para diferentes empresas, tanto nacionales como extranjeras. Para ello cuenta con diferentes centros productivos que participan en el desarrollo de dichos proyectos.

Actualmente en la institución se comenzó a trabajar en la mejora continua de la calidad, así como en satisfacer las necesidades y expectativas de sus clientes, ya que en su misión se

declara como “Universidad innovadora de excelencia científica, académica y productiva que forma de manera continua profesionales integrales comprometidos con la patria, soporte de la informatización del país y la competitividad internacional de la industria cubana del software [1].”; cual no podría lograrse, si no se estuviera plenamente convencido de que la gestión de la calidad en toda su dimensión es argumento y herramienta imprescindible y que la actividad de servicios está implícita en su contenido.

El Centro de Calidad para Soluciones Informáticas (CALISOFT) de la Universidad de las Ciencias Informáticas es una organización enfocada a contribuir al desarrollo de la industria cubana de software facilitando la implementación de las mejores prácticas en el proceso de desarrollo y/o mantenimiento de software. Es la entidad responsable de la evaluación y certificación de productos, procesos y organizaciones según normas nacionales e internacionales y de la asesoría, adiestramiento y formación continua de los especialistas en el país en los temas de Calidad e Ingeniería de Software. Promueve además, el uso de buenas prácticas para el desarrollo de software en la Industria Cubana del Software (InCuSoft.)^[2]

En dicho centro existe un sistema para la gestión de evaluaciones de software que actualmente no cuenta con la capacidad de gestionar los recursos humanos que allí laboran, los cursos de superación que se imparten, evaluar el desempeño del personal que participan en la revisión de los proyectos, por lo que el proceso de selección y evaluación del personal se realiza de forma manual con la consiguiente pérdida de tiempo durante el proceso.

A raíz de esta problemática surge como **problema científico**: ¿Cómo lograr que en el Centro de Calidad para Soluciones Informáticas (CALISOFT) el proceso de selección del Capital Humano encargado de realizar las revisiones de proyectos se efectúe de forma eficiente?

Por lo antes planteado el presente trabajo de diploma tiene como **objetivo general**:

Desarrollar el Módulo de Gestión del Capital Humano para el Sistema de Gestión de Evaluaciones de Software de la UCI.

La presente investigación tiene como **objeto de estudio** Los Sistemas Informáticos de Gestión, enmarcado en la gestión del capital humano como **campo de acción**. De acuerdo con el problema de investigación planteado anteriormente, la **idea a defender** es la siguiente:

Si se desarrolla el Módulo de Gestión del Capital Humano para el Sistema de Gestión de Evaluaciones de Software de la UCI, se garantizarán mejores niveles de eficiencia en las

actividades de selección, seguimiento y evaluación del personal perteneciente al Grupo de Auditorías y Revisiones de Software de CALISOFT.

Para dar cumplimiento al objetivo general se trazaron las siguientes **tareas de investigación**:

- Realizar búsquedas bibliográficas sobre sistemas de gestión del capital humano.
- Estudio de los sistemas de gestión del capital humano propuesto por autores y organizaciones especializadas.
- Realizar búsquedas bibliográficas sobre la aplicación de la gestión del capital humano en empresas, organizaciones e institutos que desarrollan software en el mundo.
- Levantamiento de los requisitos funcionales y no funcionales del subsistema a desarrollar.
- Definición de las entidades, actores y casos de usos del sistema.
- Definición del modelo de datos.
- Especificación de los casos de uso del sistema.
- Realización del modelo de análisis.
- Realización del modelo de diseño.
- Implementación de las funcionalidades.
- Validación y verificación de las funcionalidades.
- Realizar las pruebas para evaluar los requisitos de calidad.

Para dar cumplimiento a las tareas de la investigación se tuvieron en cuenta varios **Métodos de Investigación Científicos**.

Métodos Teóricos:

- **Método analítico-sintético:** Para el estudio de las concepciones y los conceptos empleados en los sistemas de gestión del capital humano, analizando todos los documentos elaborados por desarrolladores, para la extracción de los elementos más importantes.
- **Método histórico-lógico:** Para la comprensión de los antecedentes y las tendencias actuales referidas a la evolución en el mundo de los Sistemas de Gestión del Capital Humano.

Métodos Empíricos:

- **Entrevista:** Usado para obtener información acerca de la investigación y para realización del levantamiento de requisitos del sistema.
- **Encuesta:** Obtención de aspectos relevantes para determinar cómo se comporta la gestión del capital humano en la Universidad.
- **Modelación:** Se realizaron modelos y diagramas con el objetivo de lograr una mejor visibilidad y entendimiento de cómo se debe desarrollar el sistema.

Estructura Capítular:

La investigación está estructurada en cuatro capítulos donde se abordan temas particulares:

Capítulo I: Fundamentación Teórica: En este Capítulo se realiza el estudio del arte referente a la investigación. Se exponen los conceptos de capital humano, Sistema de Gestión de Capital Humano entre otras definiciones relacionadas con el tema. Se describen además los lenguajes, las herramientas y metodología a utilizar para el desarrollo de la aplicación.

Capítulo II: Características del Sistema: En este Capítulo se realiza una descripción de las características detalladas del sistema a implementar, se especifican detalladamente los requisitos funcionales y no funcionales y se plantea la propuesta del sistema teniendo en cuenta la especificación de los requisitos del software.

Capítulo III: Análisis y Diseño del Sistema: Se realiza el análisis y diseño del sistema y se exponen algunos diagramas relacionados con el Análisis y Diseño.

Capítulo IV: Implementación y Prueba: Se reflejan algunos modelos que dan paso a la implementación, se desarrolla la misma utilizando la metodología seleccionada y se realizarán las pruebas al producto obtenido.

Finalmente las conclusiones del trabajo, recomendaciones propuestas, bibliografía utilizada y términos importantes que brindan información relevante del trabajo realizado.

1

CAPÍTULO I FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se reflejan los temas fundamentales en los que se basa la investigación. El mismo hará referencia a los conceptos fundamentales tales como: capital humano, sistema de gestión de capital humano entre otros relacionados con el sistema a desarrollar. Se caracterizan además las metodologías y herramientas que se utilizarán para el desarrollo de la aplicación.

1.1 Definición del Capital Humano

Entendido tanto como término económico como sociológico, el concepto de Capital Humano hace referencia a la riqueza que se puede tener en una fábrica, empresa o institución en relación con la calificación del personal que allí trabaja. En ese sentido, el término capital humano representa el valor que el número de empleados (de todos los niveles) de una institución supone de acuerdo a sus estudios, conocimientos, capacidades y habilidades. El capital humano de una empresa es sin duda alguna, uno de los elementos más importantes a la hora de evaluar los rendimientos generales de la misma [3].

Este término surge en el siglo XVIII cuando teóricos de la economía tales como Adam Smith plantearon la necesidad de detenerse no sólo en factores de tipo técnicos si no también humanos a la hora de establecer las reglas de buen funcionamiento de una empresa. De tal modo, el capital humano apareció como uno de los elementos más importantes a tener en cuenta, ya que el mismo es el responsable de ejecutar las tareas y habilidades propias de cada área [3]. Así, mientras más valioso sea el capital humano de una empresa es decir, mientras mejor capacitado o preparado esté para las tareas específicas que le toquen, mejores serán los resultados de esa institución.

Esta explicación se basa en aspectos económicos y de rendimiento, pero sin embargo el concepto también puede relacionarse con elementos sociológicos tales como el acceso a los

medios de capacitación de un grupo de personas, la alfabetización en todas sus dimensiones, la proyección al futuro de determinadas carreras o labores, la posibilidad de triunfo de acuerdo al nivel de educación, etc. Todos ellos poseen relación especialmente con la noción de que el individuo no puede ser reducido a números o términos económicos o matemáticos si no que debe ser especialmente comprendido como un fenómeno social particular [3].

El comandante Fidel Castro enriquece esta definición, adaptándolas a nuestras realidades y nuestras convicciones como:

“Conjunto de conocimientos, experiencias, habilidades, sentimientos, actitudes, motivaciones, valores y capacidad para hacer, portados por los trabajadores para crear más riquezas con eficiencia, es además, conciencia, ética, solidaridad, espíritu de sacrificio y heroísmo” [4].

1.2 Componentes del Capital Humano

Cualquier empresa con mentalidad de avanzada, de éxito, debe contar con un proceso integral de desarrollo y aprovechamiento de las capacidades del Capital Humano, como base principal en el cumplimiento de los objetivos de la organización. Para que una empresa desarrolle un modelo de Capital Humano debe conocer bien los componentes del mismo [5]. Entre ellos se encuentran el comportamiento, el esfuerzo, las competencias y el tiempo, los cuales serán los cimientos que dirija el crecimiento de la organización.

Comportamiento:

Esto significa, que los medios observables de actuar contribuyen a la realización de una tarea. Los comportamientos combinan respuestas inherentes adquiridas con situaciones y estímulos del momento. Las maneras en que nos comportamos manifiestan nuestra ética, valores, creencias y reacciones ante el mundo en que vivimos.

Cuando los individuos revelan confianza en sí misma, forma un equipo con sus semejantes o denota una inclinación a actuar, exhiben conductas importantes para las organizaciones. La clave para los gestores estriba en la observabilidad: lo que usted vea será aquello con lo que tenga que tratar [5].

Esfuerzo:

Es la implementación consciente de unos recursos mentales y físicos a un fin específico. El esfuerzo es el núcleo de la ética laboral. Uno puede pedir perdón por la debilidad de su talento o la modestia de su capacidad, pero nunca por ahorrar esfuerzos. El esfuerzo físico y mental promueve habilidad, el conocimiento y encauza la conducta hacia el logro de una inversión de Capital Humano [5].

Competencia:

Es la capacidad de la persona de poner en juego e integrar conocimientos, habilidades y actitudes que hacen posible su desempeño en diversos contextos sociales. Se expresan en situaciones reales de trabajo mediante desempeños que respondan a requerimientos de calidad y eficiencia de cada área ocupacional [6].

Tiempo:

Es el recurso fundamental bajo el control del individuo, ya que el trabajador de mayor talento, conocedor y comprometido, nada logrará si no invierte tiempo en la tarea [5].

1.3 Gestión del Capital Humano

Mundialmente las organizaciones identifican a la Gestión del Capital Humano como el conocimiento, las habilidades, los valores y el potencial innovador de la organización, entre otros elementos. La gestión de dicho capital requiere de una atención muy especial, que supone la capacidad de los directivos de identificar, medir, desarrollar y renovar el activo intangible para el futuro éxito de la organización [7].

La Gestión del Capital Humano abarca un grupo de procesos que interactúan entre sí como son: Organización del Trabajo, Integración y Selección, Evolución del Desempeño, Capacitación y Desarrollo, Seguridad y Salud del Trabajo, Estimulación Moral y Material, Autocontrol, Comunicación Institucional.

Los aspectos principales de la Gestión del Capital Humano dentro de una organización deben siempre incluir:

- ✓ La gestión del desempeño.
- ✓ El desarrollo del liderazgo.
- ✓ La planificación de los recursos humanos/identificar las brechas de talento.
- ✓ El reclutamiento.

Las decisiones de la Gestión del Capital Humano se basan a menudo en un sistema de competencias organizacionales claves y en competencias inherentes al cargo. El sistema de competencias puede incluir conocimiento, habilidades, experiencia y rasgos personales (demostrados por comportamientos definidos). Los modelos más antiguos de competencias también incluían cualidades que raramente predicen el éxito (por ejemplo: la educación, la antigüedad, y factores de diversidad que hoy son considerados discriminatorios o poco ético dentro de las organizaciones) [8].

1.4 Software de Gestión

Las aplicaciones o software de gestión son aquellas diseñadas para sustituir uno o varios procedimientos, tanto comerciales como administrativos, que habitualmente realiza una persona en una empresa o institución de forma presencial, por un software, que permita realizar al cliente los mismos procedimientos de forma no presencial o disminuir el esfuerzo empleado para los mismos [9].

Se entiende por el término **gestionar** lo siguiente:

- **Planificar:** proceso de establecer objetivos con el fin de alcanzar determinados resultados. (Establecimientos de objetivos, elaboración de planes, etc.)
- **Organizar:** proceso de dividir el trabajo y de coordinar el logro de resultados que tienen un propósito común.
- **Dirigir:** proceso de conducir y coordinar esfuerzos laborales de las personas que integran una organización. Función mediante la cual se ponen en marcha las tareas programadas.
- **Controlar:** proceso de supervisar las actividades y resultados, comparándolos con los objetivos y tomando las acciones correctivas, si son necesarias [10].

Una vez dadas las definiciones de Capital Humano y Sistema de Gestión de Capital humano, se puede inferir que un Sistema de Gestión de Capital Humano debe estar orientado a satisfacer diferentes necesidades de las empresas con el fin de gestionar al potencial humano dentro de las mismas, haciendo especial énfasis en las actividades de selección del personal y las necesidades de capacitación.

1.5 Medición de los Resultados de las Iniciativas del Capital Humano

Cuando las organizaciones construyen y fortalecen sus capacidades, en ellas mejoran sus posibilidades de llevar a cabo una estrategia eficaz. El desarrollo de las capacidades organizativas exige a su vez la utilización de una serie de aspectos de importante aplicación.

Estos, hacen referencia a:

- ✓ El capital humano, que lo forman los recursos intangibles de la capacidad, el esfuerzo y el tiempo que los trabajadores emplean en su trabajo.
- ✓ La estructura de la organización que es el modelo de relaciones entre unidades e individuos en el seno de una empresa.
- ✓ Los procesos laborales, que son toda la serie de acciones y operaciones que proporcionan productos y servicios.

- ✓ La tecnología que supone el empleo de medios mecánicos, especialmente científicos y relacionados con la informática, para realizar tareas y gestionar la información.

Con el fin de llevar a cabo una estrategia, una organización debe decidir cuáles de estos aspectos y en qué modo, lograrán capacidades que aportarán a la organización una ventaja competitiva.

Mediante la gestión de estos cuatro elementos, las compañías pueden crear y explotar otras formas de su capital tangible e intangible [11].

1.6 Caracterización del Sistema de Evaluaciones de Software (GESOFT)

El Sistema para la Gestión de Evaluaciones de Software (GESOFT) surge a partir de la necesidad del Grupo de Auditorías y Revisiones del centro CALISOFT de automatizar algunos de sus procesos, específicamente los de auditorías y revisiones de software, los cuales por la propia naturaleza de sus actividades hacen que sean procesos sumamente lentos y llenos de transacciones, con una gran cantidad de notificaciones desde y hacia los proyectos. Básicamente el trabajo que realiza este sistema se puede resumir en planificación y ejecución de auditorías/revisiones, seguimientos de no conformidades y la realización de reportes. El resto de las funcionalidades son de apoyo a estas funcionalidades básicas.

El sistema obtenido funciona en un entorno web. Está realizado a partir de una arquitectura cliente-servidor que implementa el patrón arquitectónico Modelo-Vista-Controlador que implementa el framework symfony utilizado para su desarrollo. El sistema gestor de base de datos está montado sobre PostgreSQL y la capa de presentación está implementada a través del ExtJS, que es un conjunto de librerías de javascript para el diseño de forma sencilla de interfaces visuales. Este sistema se encuentra en operación desde finales del 2009 y su desarrollo aún continúa.

Entre sus principales beneficios se pueden contar que elimina el trabajo manual de los auditores/revisores y lo sustituye por un trabajo automatizado. La información de las auditorías y revisiones se encuentra centralizada permitiendo en cualquier momento la generación de reportes estadísticos que necesita la Alta Gerencia.

Hasta el momento el sistema cuenta con tres módulos:

Administración: Permite configurar las variables del sistema, así como el registro de los usuarios, tipos de auditorías y revisiones y demás datos que son necesarios para el resto de las actividades.

Producción: Permite gestionar los proyectos y sus miembros, así como los roles y responsabilidades de los mismos, y el resto de la información relacionada con los mismos.

Evaluación: En este módulo se gestionan los planes de evaluaciones, las no conformidades, acciones correctivas, los seguimientos, las lecciones aprendidas, las encuestas y demás utilidades de apoyo para la realización de las auditorías y revisiones.

1.7 Herramientas y Tecnologías para la Construcción del Software

El presente epígrafe expone las características de las herramientas y tecnologías que serán utilizadas para llevar a cabo el desarrollo de la aplicación. La solución de la propuesta será desarrollada a partir de las herramientas que brinda el software libre, buscando la independencia tecnológica que estas brindan.

El hecho de desarrollar un sistema sobre plataforma libre es primeramente garantía de que el producto final pueda ser modificado por quien lo desee y de esta manera poder hacerle las adaptaciones o mejoras que éste estime conveniente.

1.8 Dependencias y Restricciones del Cliente

Lenguaje de Programación PHP 5.0

Para la implementación del sistema se usará PHP versión 5.0, pues posee una comunidad de desarrollo muy grande, la cual implementa constantemente mejoras en su código, en la Universidad de las Ciencias Informáticas este lenguaje es utilizado por muchos programadores, pues cuenta con muchas ventajas que le son de gran beneficio a los mismos. PHP es un lenguaje "del lado del servidor" (esto significa que PHP funciona en un servidor remoto que procesa la página web antes de que sea abierta por el navegador del usuario) especialmente creado para el desarrollo de páginas web dinámicas. Puede ser incluido con facilidad dentro del código HTML, y permite una serie de funcionalidades tan extraordinarias que se ha convertido en el favorito de millones de programadores en todo el mundo.

Características:

- ✓ Gratuito.
- ✓ De gran popularidad.
- ✓ Posee una enorme eficiencia.
- ✓ Se integra de forma sencilla con múltiples bases de datos.
- ✓ Es versátil.
- ✓ Tiene un gran número de funciones predefinidas.

Framework Symfony

El Framework Symfony presenta como objetivo fundamental la automatización de los patrones más utilizados en la elaboración de aplicaciones web; también se destaca por orientar a los programadores hacia la clasificación de código, establece un estándar de código legible, encapsula operaciones complejas en simples líneas de código, lo que trae consigo un ahorro de tiempo a la hora de mostrar datos directamente de la base de datos (BD). Este framework esta implementado bajo el lenguaje PHP 5, es compatible con la mayoría de los SGBD existentes como MySQL, PostgreSQL, Oracle y el SQL Server de Microsoft; otorga facilidades para diferentes plataformas; está basado en el patrón Modelo-Vista-Controlador el cual en Symfony proporciona una abstracción de la base de datos permitiendo una mayor facilidad en la obtención de datos, haciendo a la vista y las acciones independientes al SGBD; en la vista posee helpers que facilitan el trabajo para los diseñadores en el código HTML de la aplicación; aunque es necesario crear una nueva vista, mantiene al modelo y al controlador original; encargándose de mantener alejados a la vista y al modelo de los detalles del protocolo utilizado. A continuación se hará referencia a las características más detalladas de este framework.

Características de Symfony:

Symfony se diseñó para que se ajustara a los siguientes requisitos:

- ✓ Fácil de instalar y configurar en la mayoría de las plataformas.
- ✓ Independiente del sistema gestor de bases de datos.
- ✓ Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- ✓ Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- ✓ Preparado para aplicaciones empresariales, adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- ✓ Código fácil de leer que incluye comentarios de PHP Documentor y que permite un mantenimiento muy sencillo.
- ✓ Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros ^[12].

EXT.JS 3.0

Es un nuevo marco de trabajo de javascript del lado del cliente para el desarrollo de aplicaciones web. Ofrece una gran cantidad de widgets para crear interfaces de usuario complejas. Ofrece una serie de librerías (compatibles con PHP 4 y 5) para integrar Ext JS en un sistema. Funciona como un mapeado en clases de la librería JS. Entre las posibilidades que brinda se encuentra con la creación de formularios, combos, grids o menús. Además, ayuda a la comunicación entre el cliente y el servidor mediante JSON y XML.

Ventajas:

- ✓ Código reutilizable.
- ✓ Independiente o adaptable a marco de trabajo diferentes.
- ✓ Orientada a la programación de interfaces tipo escritorio en la web.
- ✓ El API es homogeneizado independientemente del adaptador usado. Los controles siempre se verán igual.
- ✓ Soporte comercial.
- ✓ Una extensa comunidad de usuarios.
- ✓ El diseño está completamente separado de la funcionalidad.
- ✓ Funciones comunes como validación, combobox editables, ventanas deslizables (con minimizar y maximizar) y grillas editables, son muy fáciles de implementar.
- ✓ Utilización de AJAX y JSON como mecanismos de comunicación con el servidor.
- ✓ Implementación basada en patrones de diseño.
- ✓ Amplia librería de componentes gráficos fácilmente extensibles.
- ✓ Buena y amplia documentación, así como también su comunidad.

Sistema Gestor de Base de Datos PostgreSQL 3.8

Está orientado principalmente a bases de datos grandes, tiene muy buen soporte de administración y auditoría. Es distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales. PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando. A continuación se describen las principales ventajas de PostgreSQL ^[13].

- ✓ Su Acceso Concurrente Multiversión (MVCC) permite que mientras un proceso escribe en una tabla otros accedan a ella sin necesidad de bloqueos, puesto que cada usuario obtiene una “fotografía” de la base de datos.
- ✓ Mantiene la integridad transaccional.
- ✓ Soporta amplia variedad de tipos nativos, tales como números de diversa precisión, figuras geométricas, direcciones IPv4 e IPv6, direcciones MAC.
- ✓ Es multiplataforma y de libre copia.
- ✓ Como desventaja, se puede mencionar que es más lento que otros SGBD, como SQL/Server u Oracle ^[13].

Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. Como se expresaba anteriormente, esta metodología concibió desde sus inicios el uso de UML como lenguaje de modelado.

Es un proceso dirigido por casos de uso, este avanza a través de una serie de flujos de trabajo (requisitos, análisis, diseño, implementación, prueba) que parten de los casos de uso; está centrado en la arquitectura y es iterativo e incremental.

Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software como son:

- ✓ Desarrollo de software en forma iterativa.
- ✓ Manejo de requerimientos.
- ✓ Utiliza arquitectura basada en componentes.
- ✓ Modela el software visualmente.
- ✓ Verifica la calidad del software.
- ✓ Controla los cambios.

Lenguaje de Modelado UML

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es importante resaltar que UML es un "lenguaje" para especificar y no para describir métodos o procesos. Se utiliza para definir un sistema de software, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo. Se puede aplicar en una gran variedad de formas para dar soporte a una metodología de desarrollo de software, pero no especifica en sí mismo qué metodología o proceso usar. UML no puede compararse con la programación estructurada, pues UML significa (Lengua de Modelación Unificada), no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la orientación a objetos viene siendo un complemento perfecto de UML.

1.9 Herramientas para el Modelado de la Aplicación Visual Paradigm.

Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo de desarrollo de software: análisis y diseño orientado a objetos, construcción, prueba y despliegue. El software de modelado ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta CASE UML también proporciona abundantes tutoriales de UML, demostraciones interactivas y proyectos UML [14].

Entre los lenguajes conocidos por Visual Paradigm para la ingeniería inversa, se encuentran los siguientes:

- ✓ Java Source.
- ✓ C++ Source.
- ✓ PHP 5.0 Source.
- ✓ XML.

1.10 Entorno de Desarrollo Integrado NetBeans 6.9

Esta versión tiene nuevos soportes para lenguajes tanto web como desktop. Importante recalcar que es un software libre, multiplataforma, esta versión es más rápida a la hora de compilar, permite un completamiento de código más extenso al estar utilizando el Symfony como framework, ofrece ventajas técnicas y económicas para todo aquel que desee usarlo,

presenta un plugin para el desarrollo de la tecnología AJAX, presenta una interfaz amigable y sobre todo permite depurar el código realizado.

Conclusiones

En este capítulo se abordaron los conceptos fundamentales del objeto de estudio, lo que constituye además la base del presente Trabajo de Diploma. En él se realizó una caracterización del entorno de desarrollo donde se integrará la solución, además se expusieron las suposiciones y restricciones establecidas por el cliente. Quedaron especificadas además las herramientas y metodologías que se utilizarán a lo largo de todo el trabajo, entre las que se encuentran PHP como lenguaje de programación, UML como lenguaje de modelado, la Programación Orientada a Objetos como paradigma de programación, el Proceso Unificado de Desarrollo como metodología de desarrollo, Visual Paradigm como herramientas CASE, así como PostgreSQL como sistema gestor de base de datos y Symfony como Framework de desarrollo.

2

CAPÍTULO II CARACTERÍSTICAS DEL SISTEMA

Introducción

Este capítulo abarcará una descripción detallada de los procesos del negocio, donde serán identificados: actores, casos de uso, trabajadores y reglas del negocio; serán elaborados: el diagrama de casos de uso del negocio y el modelo de objetos. Esto sentará las bases para la definición de los requerimientos funcionales y no funcionales del sistema, indispensables para el posterior desarrollo de los casos de uso y el modelo de casos de uso del sistema.

2.1 Modelo de Negocio

El Modelamiento del Negocio constituye el primer flujo de trabajo a desarrollar según la metodología de desarrollo de software (RUP). Este describe los procesos de negocio de una empresa en términos de casos de uso y actores del negocio que se corresponden con los procesos del negocio y los clientes respectivamente.

2.1.1 Procesos del Negocio

“Un proceso de desarrollo de software es una definición del conjunto completo de actividades necesarias para convertir los requisitos de usuario en un conjunto consistente de artefactos que conforman un producto software, y para convertir los cambios sobre esos requisitos en un nuevo conjunto consistente de artefactos ^[15].”

Proceso	Descripción
Generar Reportes.	Este proceso es realizado por el Revisor Líder o el Coordinador de PPQA. Se confecciona a partir de la información del estado de los recursos, así como información sobre la evaluación de desempeño de cada revisor. Contienen datos sobre la disponibilidad del personal, evaluaciones realizadas, recursos humanos disponibles según el cronograma de cada persona.

Gestionar Usuarios.	Proporciona la posibilidad de adicionar, modificar los datos de las personas registradas en el sistema. De estas personas se conocen datos como nombre y apellidos, correo electrónico, nombre de usuario (que se deduce del correo electrónico), área a la que pertenece entre otros datos de contacto que pueden resultar de interés.
Gestionar Habilidades.	Proporciona la posibilidad de adicionar, editar o eliminar habilidades asociadas al personal. Las habilidades están asociadas a un rol en el sistema y son heredadas. Por ejemplo un revisor debe tener la habilidad de la observación, asimismo el revisor líder debe tener la misma habilidad puesto que debe tener todas las habilidades del revisor más otras inherentes a su propio rol.
Gestionar Cursos.	Proporciona la posibilidad de adicionar, editar o eliminar los datos de un determinado curso registrado en el sistema. De los cursos se conoce: entidad que los imparte, el título del curso, las habilidades que cubre y si además otorga certificación o no.

Tabla 1. Procesos del Negocio.

2.1.2 Entidades del Negocio

Las entidades de negocio representan a los objetos que los trabajadores del negocio toman, inspeccionan, manipulan, producen o utilizan durante la realización de los casos de uso de negocio [16].

Entidades del Negocio	Descripción
Reporte de la Evaluación de Desempeño	Es el documento que recoge la evaluación de desempeño del evaluado una vez terminado el proceso de revisión de un proyecto. Dichas evaluaciones se realiza utilizando criterios predefinidos: resumen de los datos del evaluado, criterios de evaluación (atributos personales, conocimientos mostrados, desenvolvimiento en su desempeño, entrega de la información con calidad, etc.).
Reporte de estado de los recursos.	Contiene datos sobre el estado del Capital Humano

Entidades del Negocio	Descripción
	(disponibles o no, evaluaciones realizadas entre otros).
Reporte sobre recursos disponibles	Contiene los datos del Capital Humano disponible dado un rango de fecha específico a partir de la disponibilidad de estos según el cronograma de cada persona.

Tabla 2. Entidades del Negocio

2.1.3 Actores del Negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa [16].

Actor del Negocio	Descripción
Alta Gerencia	La Alta Gerencia de la producción en la UCI es la que solicita periódicamente la evaluación del desempeño de los recursos humanos que participan en las revisiones de los proyectos. Además requieren que se mantenga vigilancia permanente sobre la capacitación de estos recursos.

Tabla 3. Actor del Negocio

2.1.4 Trabajadores del Negocio

Un trabajador del negocio es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol [17].

Trabajador del Negocio	Descripción
Coordinador de PPQA	Es el encargado de Gestionar los cursos que se imparten en la entidad, el Capital Humano existente y las habilidades de cada revisor/auditor.
Revisor Líder	Es el encargado de realizar la evaluación de desempeño de cada revisor/auditor según criterios predefinidos.

Tabla 4. Trabajadores del Negocio

2.1.5 Diagrama de Caso de Uso del Negocio

Un diagrama de casos de uso del negocio representa gráficamente los procesos del negocio y su interacción con los actores del negocio. A continuación se muestran los casos de uso que fueron identificados en el negocio.



Figura.1. Diagrama de Caso de Uso del Negocio.

Caso de Uso del Negocio	
Caso de Uso	Realizar Solicitud de Evaluación de Desempeño
Actor	Alta Gerencia de la UCI
Trabajador	Coordinador de PPQA, Revisor Líder
Resumen	El caso de uso se inicia cuando la Alta Gerencia de la UCI solicita la evaluación del desempeño de los recursos humanos que participan en las revisiones de los proyectos. El Revisor Líder es el encargado de realizar la evaluación de desempeño de cada revisor/auditor. Este caso de uso finaliza cuando la Alta Gerencia obtiene la evaluación de desempeño.
Precondiciones	No procede.
Flujo Normal de Eventos	
Sección 1	
Acción del Actor	Respuesta del Negocio
1. La Alta Gerencia de la UCI solicita la evaluación de desempeño de los revisores.	2. El Revisor Líder confecciona la evaluación de desempeño de los revisores. Teniendo en cuenta aspectos tales como: resumen de los datos del evaluado, criterios de evaluación (atributos personales, conocimientos

4. La Alta Gerencia de la UCI recibe la evaluación de desempeño.	mostrados, desenvolvimiento en su desempeño, entrega de la información con calidad, etc.). 3. Este modelo es enviado a la Alta Gerencia por el Revisor Líder, periódicamente cada tres meses.
Poscondiciones	No procede.

Tabla 5. CUN_ Realizar Solicitud de Evaluación de Desempeño

Caso de Uso del Negocio	
Caso de Uso	Realizar control de capacitación de RRHH
Actor	Alta Gerencia de la UCI
Trabajador	Coordinador de PPQA, Revisor Líder
Resumen	El caso de uso se inicia cuando la Alta Gerencia de la UCI solicita el control de la capacitación de los recursos humanos que participan en las revisiones de los proyectos. El Coordinador de PPQA es el encargado de gestionar los Cursos que se imparten por las diferentes entidades. Este caso de uso finaliza cuando el Coordinador de PPQA finaliza el proceso de gestión de cursos según el conjunto de habilidades de los recursos humanos y la disponibilidad de los mismos.
Precondiciones	No procede
Flujo Normal de Eventos	
Sección 1	
Acción del Actor	Respuesta del Negocio
1. La Alta Gerencia de la UCI realiza el control de la Capacitación de los Recursos Humanos.	3 El Coordinador de PPQA gestiona los cursos que se imparten por las diferentes entidades, los RRHH existentes y las habilidades de cada

	revisor/auditor.
Poscondiciones	No procede.

Tabla 6. CUN_ Controlar Capacitación de Recursos Humanos.

A continuación se muestra el diagrama de actividades correspondiente a las descripciones de los casos de uso del negocio descritos anteriormente:

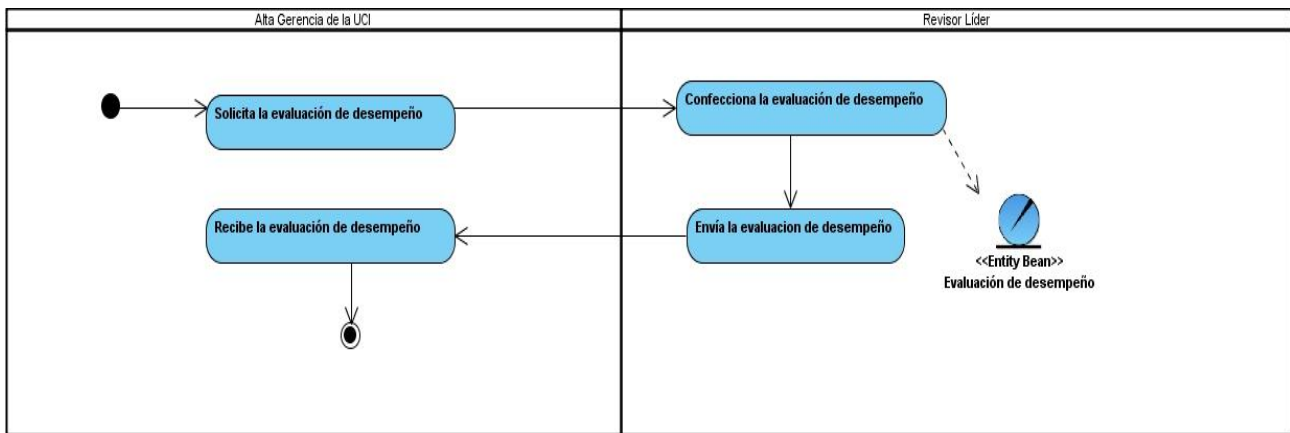


Figura 2. Diagrama de actividades CU: Realizar Solicitud de Evaluación de Desempeño.

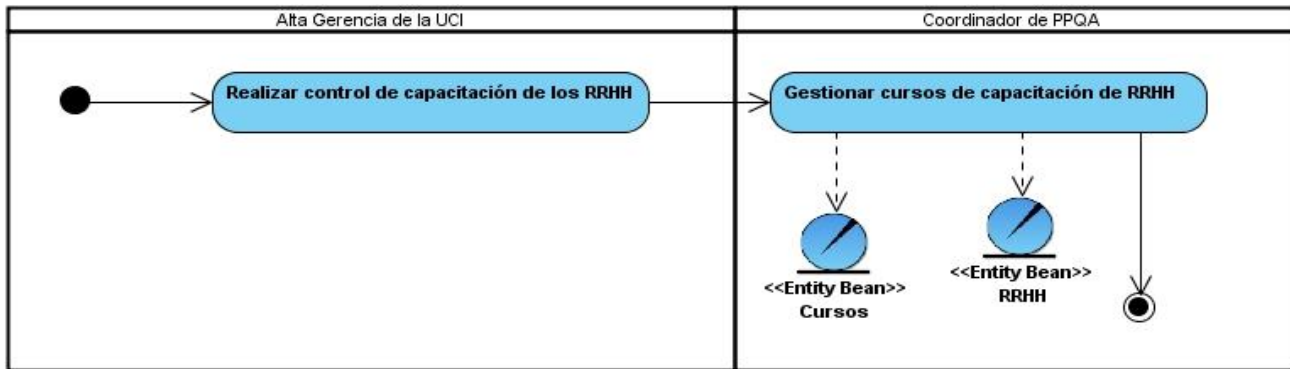


Figura 3. Diagrama de actividades CU: Realizar control de capacitación de RRHH.

2.1.6 Modelo de Objetos del Negocio

El modelo de objetos del negocio es un modelo interno del negocio. Describe cómo cada caso de uso de negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades y unidades de trabajo [18].

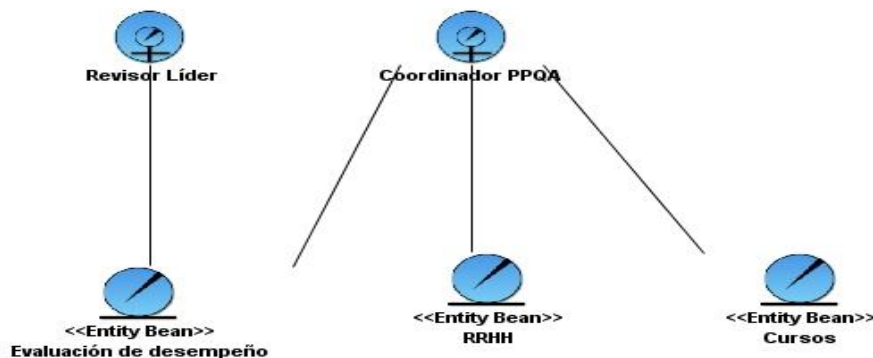


Figura 4. Modelo de objeto del negocio.

2.1.7 Reglas del Negocio

Las reglas del negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. En el Módulo de Gestión del Capital Humano se consideraran las siguientes:

- ✓ Para realizar cualquier acción en el sistema el usuario debe estar autenticado.
- ✓ No pueden existir dos personas con el mismo nombre de usuario en el sistema.
- ✓ La autenticación se realiza a través del directorio activo de la UCI. De fallar la conexión se realiza de forma local contra los datos de la base de datos.
- ✓ Un evaluador no puede realizar dos evaluaciones a la misma vez.
- ✓ Para realizar auditorías y revisiones los auditores/revisores deben haber recibido el curso de capacitación correspondiente.
- ✓ El Coordinador de PPQA no puede participar como revisor/revisor líder en ninguna revisión.

2.2 Captura de Requisitos

2.2.1 Técnicas Empleadas para la Captura de Requisitos

La captura de requisitos es la actividad mediante la que el equipo de desarrollo de un sistema de software extrae de cualquier fuente de información disponible, las necesidades que debe cubrir dicho sistema. El proceso de captura de requisitos puede resultar complejo, principalmente si el entorno de trabajo es desconocido para el equipo de analistas, y depende mucho de las personas que participen en él. Por la complejidad que todo esto puede implicar, la ingeniería de requisitos ha trabajado desde hace años en desarrollar técnicas que permitan hacer este proceso de una forma más eficiente y precisa [19].

A continuación se presentan un grupo de técnicas que de forma clásica han sido utilizadas para esta actividad en el proceso de desarrollo de todo tipo de software.

La Entrevista es una técnica para recopilar información a partir de un intercambio directo entre personas o grupos [20]. Esta fue la primera técnica empleada para la captura de los requisitos con los que debía cumplir el módulo de gestión del Capital Humano. La misma estuvo dirigida al Grupo de Calidad y Soluciones Informáticas de la Universidad.

La investigación documental consiste en el estudio de documentos escritos sobre un objeto determinado, es decir son todos aquellos documentos registrados en diferentes dispositivos físicos a los que podemos tener acceso en forma directa o indirecta para su consulta [20]. Esta técnica se utilizó para estudiar y analizar las descripciones del negocio hechas por los especialistas, tanto como los Manuales de usuario y otros documentos relativos al Módulo de Gestión del Capital Humano, lo que permitió además tener conocimiento de otros requisitos necesarios que no fueran capturados durante la entrevista.

Sistemas existentes

Esta técnica consiste en analizar distintos sistemas ya desarrollados que están relacionados con el módulo de gestión a desarrollar. Se analizaron las interfaces de usuario, observando el tipo de información manejada, se analizaron las distintas salidas que se producen, porque siempre pueden surgir nuevas ideas sobre la base de estas [21].

2.2.2 Requerimientos Funcionales.

RF_1. Gestionar cursos.

RF_2. Gestionar habilidades.

RF_3. Gestionar recursos humanos.

RF_4. Gestionar evaluaciones de desempeño.

RF_4.1. Exportar a PDF.

RF_5. Generar Reportes.

RF_5.1. Reportar sobre el desempeño de los revisores.

RF_5.2. Reportar sobre el estado de los recursos.

RF_5.3. Reportar sobre recursos disponibles.

RF_6. Gestionar agenda.

2.2.3 Descripción Detallada de los Requerimientos Funcionales. Ver: [Anexo I.](#)

2.2.4 Requisitos No Funcionales (RNF)

Son las propiedades emergentes del sistema como la fiabilidad, el tiempo de respuesta, la capacidad de almacenamiento, la capacidad de los dispositivos de entrada/salida, y la representación de datos que se utiliza en las interfaces del sistema.

RNF_1: Usabilidad

Comprende todo lo relacionado con las facilidades de uso que debe brindar el sistema a los usuarios, las cuales incluyen: una interfaz amigable, una vía rápida y efectiva para realizar búsquedas de información, una rápida adaptación a usuarios con poca experiencia. Incluye el empleo de los perfiles de usuarios, para la diferenciación de opciones e interfaces de acceso al sistema mediante diferentes roles.

RNF_2: Seguridad

- **Confiabilidad:** La información manejada por el sistema debe estar protegida de acceso no autorizado y divulgación.
- **Integridad:** La información manejada por el sistema debe ser objeto de una cuidadosa protección contra la corrupción y estados de inconsistencia.
- **Disponibilidad:** La aplicación deberá estar disponible en todo momento para aquellas personas que tienen acceso a su información. Los mecanismos utilizados para lograr la seguridad no deben ser un obstáculo para los usuarios a la hora de obtener los datos deseados en un momento dado.

RNF_3: Portabilidad

Las herramientas utilizadas podrán ser usadas bajo cualquier sistema operativo Windows NT en adelante o cualquier distribución de Linux. El servidor web y el servidor de base de datos pueden estar en la misma computadora sin ocasionar problema alguno.

RNF_4: Rendimiento

Para el funcionamiento óptimo de la aplicación se seguirán diferentes técnicas de elaboración de sitios Web, que faciliten un acceso rápido 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 la consultas de la base de datos. La aplicación propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición.

RNF_5: Apariencia o interfaz externa e interna

- ✓ **Interfaz Web:** la interfaz deberá ser sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo de su empleo.
- ✓ **Interfaz interna:** la interfaz interna estará determinada por el desarrollador, construyendo así una vista escalable de las clases o agrupaciones de clases que

permitirán un mejor encapsulamiento de las funcionalidades y una mayor abstracción modular del sistema.

RNF_6: Soporte

El sistema debe permitir posteriores modificaciones y actualizaciones a fin de alcanzar un mayor número de funcionalidades, en caso de cambios en el negocio.

RNF_7: Diseño e implementación

Para desarrollar el análisis y diseño del sistema se utilizará: como metodología de desarrollo RUP, como lenguaje de modelado UML y como herramienta de modelado Visual Paradigm. En la implementación del sistema se utilizará como lenguaje de programación PHP 5.0, como herramienta NetBeans 6.9 y como Framework de desarrollo Symfony.

RNF_8: Requerimientos de Software

En el servidor de aplicaciones estará instalado Apache 2.0 y en el servidor de base de datos, PostgreSQL 3.8 o superior. Para que funcione la aplicación, se utilizará el navegador Mozilla Firefox.

RNF_9: Requerimientos de Hardware

En el cliente se requiere una máquina con 256 MB de RAM como mínimo, un microprocesador Pentium 4 a más de 1.6 GHz y un disco duro de más de 10 GB, el servidor web junto con el servidor de base de datos debe tener al menos 512 MB de RAM y 10 GB de disco duro mínimo, todas las máquinas implicadas en la funcionalidad de la aplicación deben estar conectadas a la red de al menos 1.0 Mbps de velocidad.

2.2.5 Descripción del Sistema Propuesto

Con el propósito de automatizar los procesos de gestión del Capital Humano que realiza las revisiones de proyectos, así como controlar y gestionar su capacitación, en el Centro de Calidad para Soluciones Informáticas perteneciente a la UCI, se propone la realización del análisis, diseño e implementación del Módulo de Capital Humano para el Sistema de Gestión de Evaluaciones de Software de la UCI, para agilizar y elevar la calidad de los procesos. Dicho software presentará entre sus principales funcionalidades permitir: adicionar, modificar, eliminar (cursos de capacitación, usuarios, habilidades), mostrar diferentes tipos de listados, exportar documentos a formato PDF, imprimir documentos, etc.

2.3 Actores del Sistema

Cada trabajador del negocio que tiene actividades a automatizar es considerado un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema.

Actor del Sistema	Descripción
Coordinador de PPQA	Es el encargado de Gestionar los Cursos que se imparten por las diferentes entidades, el Capital Humano existente y las habilidades de cada revisor/auditor. Además es el encargado de realizar las evaluaciones a los revisores líderes.
Revisor Líder	Es el encargado de realizar la evaluación de desempeño de cada revisor/auditor según criterios predefinidos.

Tabla 7. CUN_ Actores del Sistema.

2.3.1 Diagrama de Casos de Uso del Sistema

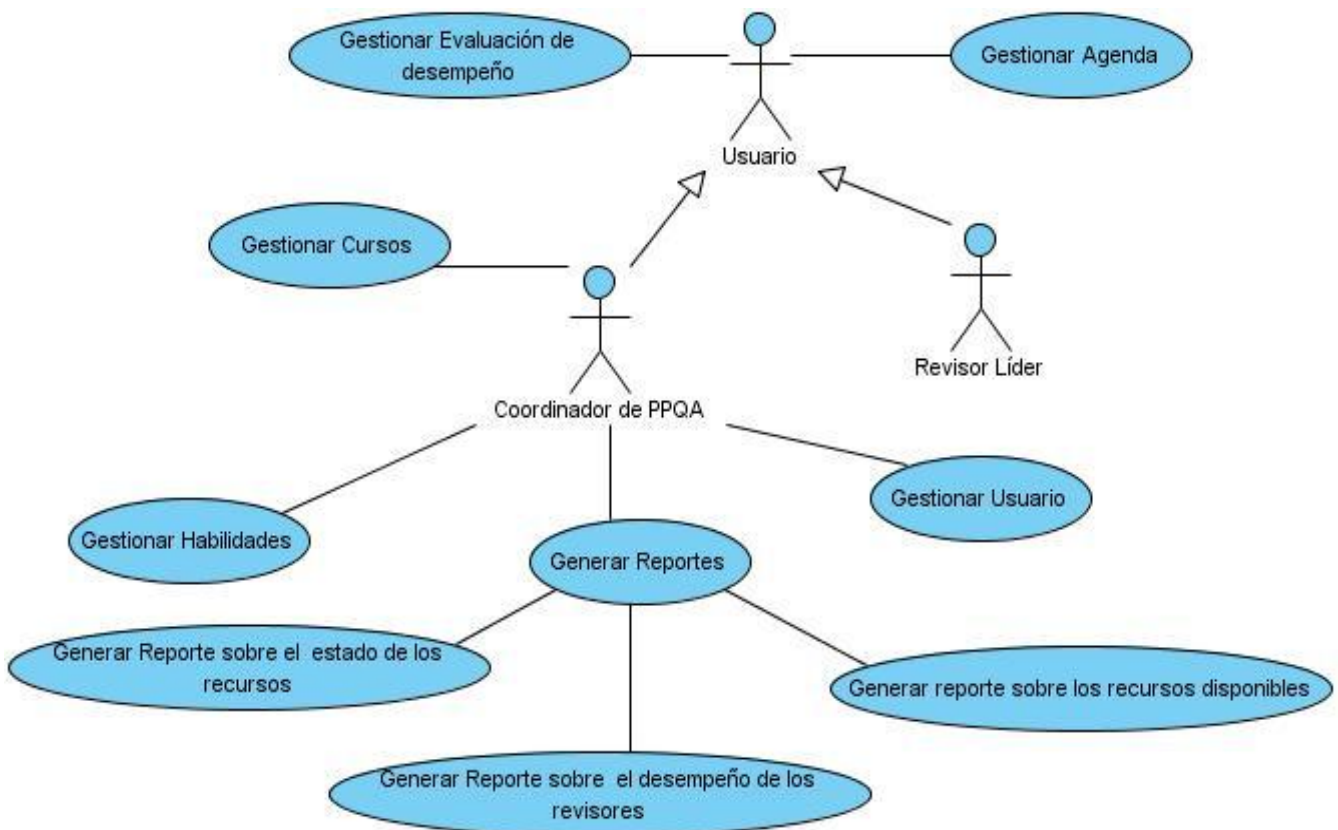


Figura 5. Diagrama de Casos de Uso del Sistema.

2.3.2 Descripción de los Casos de Uso del Sistema.

Caso de Uso del Sistema	
CUS_1	Gestionar evaluación de desempeño.
Actor	Coordinador de PPQA, Revisor líder.
Descripción	El caso de uso se inicia cuando el actor selecciona la opción Gestionar evaluación de desempeño, inmediatamente se le muestra un modelo con los diferentes criterios a evaluar, brindando la opción de salvar la evaluación de desempeño de la persona según criterios de evaluación predefinidos, también permite exportarla en formato .PDF para su futura impresión. El Coordinador de PPQA evalúan a los revisores líderes, y estos a su vez evalúan a los revisores.
Referencia	RF_4, 4.1, 4.2,5.1

Tabla 8. Descripción del Caso de Uso del Sistema Gestionar evaluación de desempeño.

Caso de Uso del Sistema	
CUS_2	Gestionar habilidades.
Actor	Coordinador de PPQA.
Descripción	El caso de uso se inicia cuando selecciona la opción Gestionar habilidades, inmediatamente se le muestra un formulario con la opción de insertar, modificar o eliminar una habilidad determinada.
Referencia	RF_2

Tabla 9. Descripción del Caso de Uso del Sistema Gestionar habilidades.

Caso de Uso del Sistema	
CUS_3	Gestionar cursos.
Actor	Coordinador de PPQA.
Descripción	El caso de uso se inicia cuando el actor ejecuta la aplicación y selecciona la opción Gestionar cursos. Luego se muestra un formulario donde el actor tiene la posibilidad de Adicionar, Editar o Eliminar un curso en el sistema. De los cursos se conoce entidad que los imparte, el título del curso, fecha de inicio, fecha de fin, las habilidades que cubre y si además otorga certificación o no.
Referencia	RF_1

Tabla 10. Descripción del Caso de Uso del Sistema Gestionar cursos.

Caso de Uso del Sistema	
CUS_4	Gestionar Usuarios
Actor	Coordinador de PPQA.
Descripción	El caso de uso se inicia cuando el actor ejecuta la aplicación y selecciona la opción Gestionar usuarios. Proporciona la posibilidad de Adicionar o Editar datos de una determinada persona en el sistema. De estas personas se conocen datos como nombre y apellidos, correo electrónico, nombre de usuario, área a la que pertenece entre otros datos de contacto que pueden resultar de interés.
Referencia	RF_3

Tabla 11. Descripción del Caso de Uso del Sistema Gestionar usuarios.

Caso de Uso del Sistema	
CUS_5	Generar Reportes
Actor	Coordinador de PPQA.
Descripción	El caso de uso se inicia cuando el actor ejecuta la aplicación y selecciona la opción Generar reportes. Esta permite tres opciones diferentes. (1) Reporte sobre el desempeño de los revisores. Proporciona la posibilidad de realizar reportes sobre el desempeño de los revisores/auditores. Estos reportes deben resumir los resultados de las evaluaciones realizadas a estos revisores/auditores. (2) Reporte sobre el estado de los recursos. Proporciona la posibilidad de mostrar reportes sobre el estado del capital humano (disponibles o no en la fecha solicitada, evaluaciones realizadas entre otros). (3) Reporte sobre recursos disponibles. Proporciona la posibilidad de realizar reportes sobre el capital humano disponible según el cronograma de cada persona.
Referencia	RF_5,5.1,5.2,5.3

Tabla 12. Descripción del Caso de Uso del Sistema Generar Reportes.

CUS_6	Gestionar Agenda
Actor	Coordinador de PPQA, Revisor Líder, Usuarios comunes
Descripción	El caso de uso se inicia cuando el actor selecciona la opción Gestionar Agenda. El sistema proporciona la posibilidad de seleccionar en el calendario el día o la

	cantidad de días en que cada persona no estará disponible para realizar las revisiones de los proyectos. Esta información es almacenada en la base de datos, para así generar los diferentes tipos de reportes.
Referencia	RF_6

Tabla 13. Descripción del Caso de Uso del Sistema Gestionar Agenda.

Conclusiones

En este capítulo se abordaron las características fundamentales del sistema a implementar. Se realizó el estudio de los objetivos y los procesos del negocio que lo soportan, así como un análisis crítico de todos los procesos vinculados con el campo de acción. Se identificaron los trabajadores y los actores que intervienen en el mismo así como sus responsabilidades. El modelado del negocio realizado, así como los diagramas de actividades por cada caso de uso, el levantamiento de requisitos funcionales y no funcionales, los diagramas de casos de uso y diagramas de clases, permitieron obtener un mayor entendimiento del problema y la solución propuesta que se materializará a través del diseño e implementación en el próximo capítulo.

3

CAPÍTULO III ANÁLISIS Y DISEÑO

Introducción

En el presente capítulo se realizará el diseño del sistema donde se traducirán los requisitos a una especificación que describe cómo implementar el sistema. Se transforman los requisitos al diseño del futuro sistema utilizando una arquitectura correcta, logrando adaptar el diseño para que sea consistente con el entorno de implementación.

3.1 Análisis

En el análisis se utiliza una combinación de textos y diagramas para describir los requisitos funcionales del software, estructurándolos en función de hacer más fácil su comprensión, modificación y en general su mantenimiento. A través del mismo se presenta una vista interna del sistema refinando los requisitos y agrupándolos en base a clases y paquetes.

Modelo de Análisis

El modelo de análisis debe lograr tres objetivos primarios: describir lo que requiere el cliente, establecer una base para la creación del diseño del software, y definir un conjunto de requisitos que se puedan validar una vez que se construya el software [19].




Clases del Análisis		
Nombre	Características	Representación
Interfaz	Modelan la interacción entre el sistema y los actores.	 Clase Interfaz
Controladora	Representan coordinación, secuencia, y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto.	 Clase Control
Entidad	Modelan la información que posee una vida larga y que a menudo es persistente.	 Clase Entidad

Tabla 14. Clases del Análisis

3.1.1 Diagramas de Clases de Análisis

Módulo de Gestión del Capital Humano

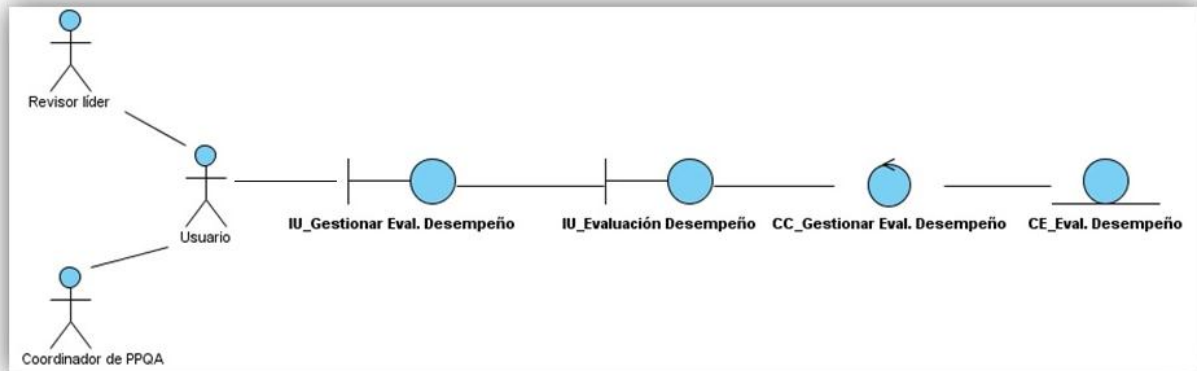


Figura 6. Diagrama de clases de Análisis CUS: Gestionar Evaluación de Desempeño.

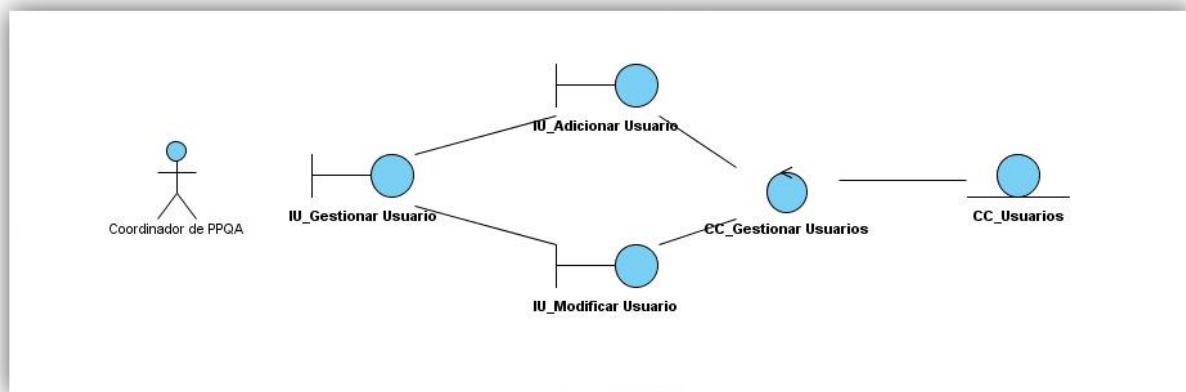


Figura 7. Diagrama de clases de Análisis CUS: Gestionar Usuarios.

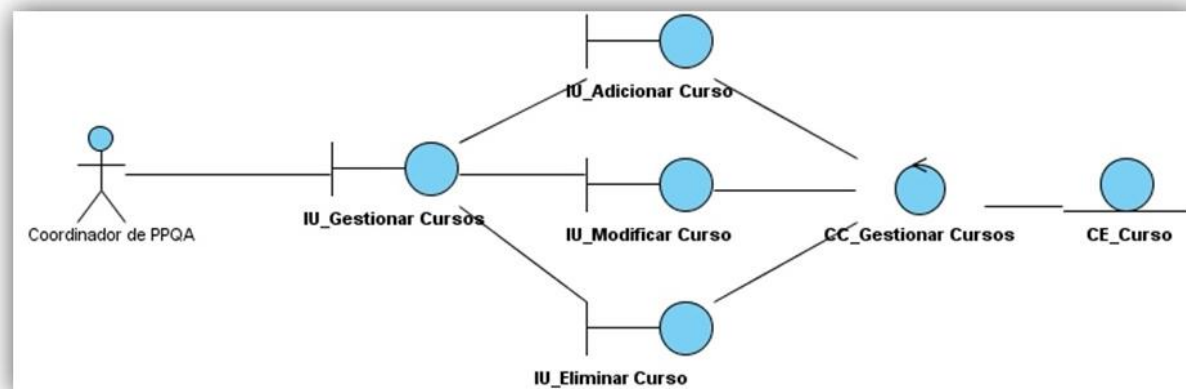


Figura 8. Diagrama de clases de Análisis CUS: Gestionar Cursos.

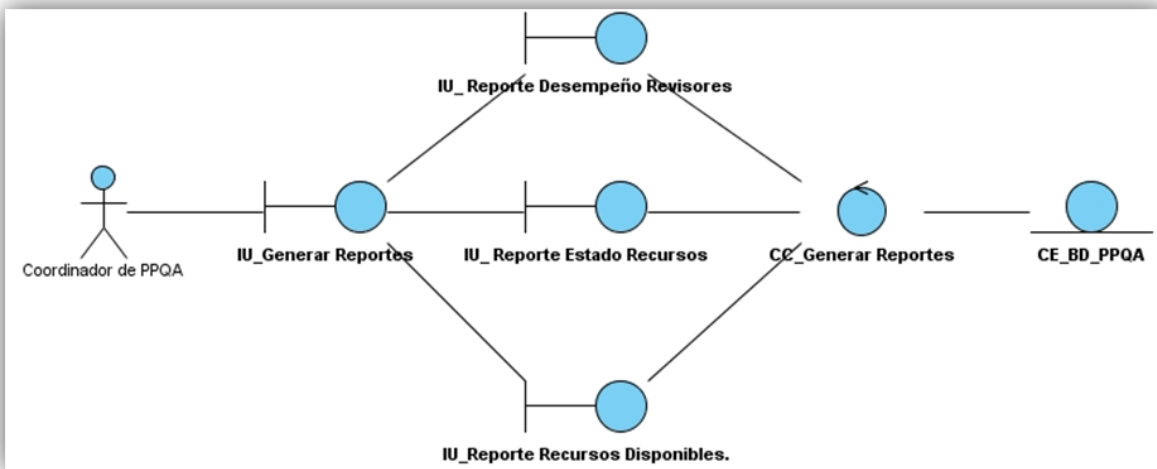


Figura 9. Diagrama de clases de Análisis CUS: Gestionar Reportes.

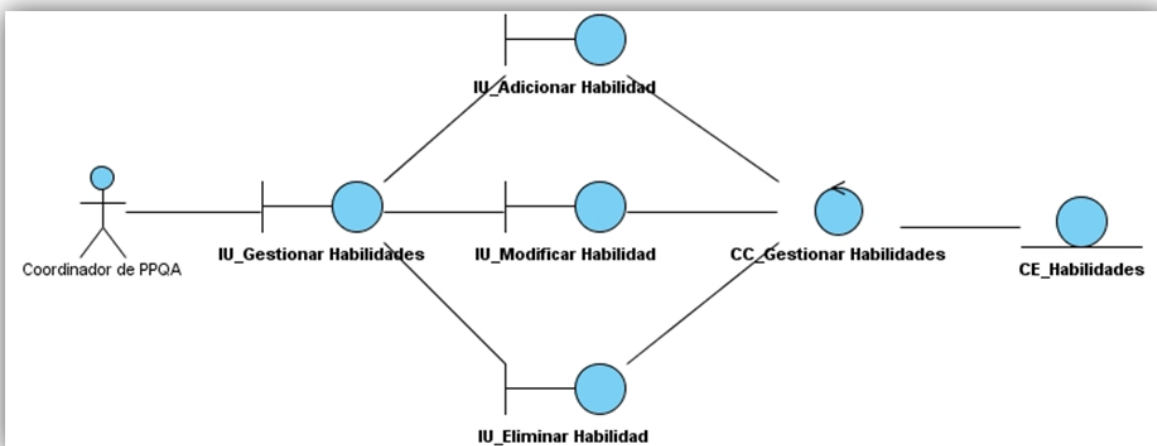


Figura 10. Diagrama de clases de Análisis CUS: Gestionar Habilidades.

3.1.2 Diagramas de Interacción.

Los diagramas constituyen la representación gráfica de un conjunto de elementos. Visualizan un sistema desde diferentes perspectivas. Un diagrama de interacción representa los objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos, dentro de esta clasificación se encuentran incluidos los diagramas de secuencia [24].

Como parte del desarrollo del presente trabajo de diploma se realizaron los diagramas de secuencias correspondientes a cada uno de los flujos de los requisitos del sistema los cuales darán soporte a la implementación del software.

A continuación, se muestran los diagramas elaborados para cada uno de los requisitos funcionales del sistema.

RF: Gestionar Cursos.

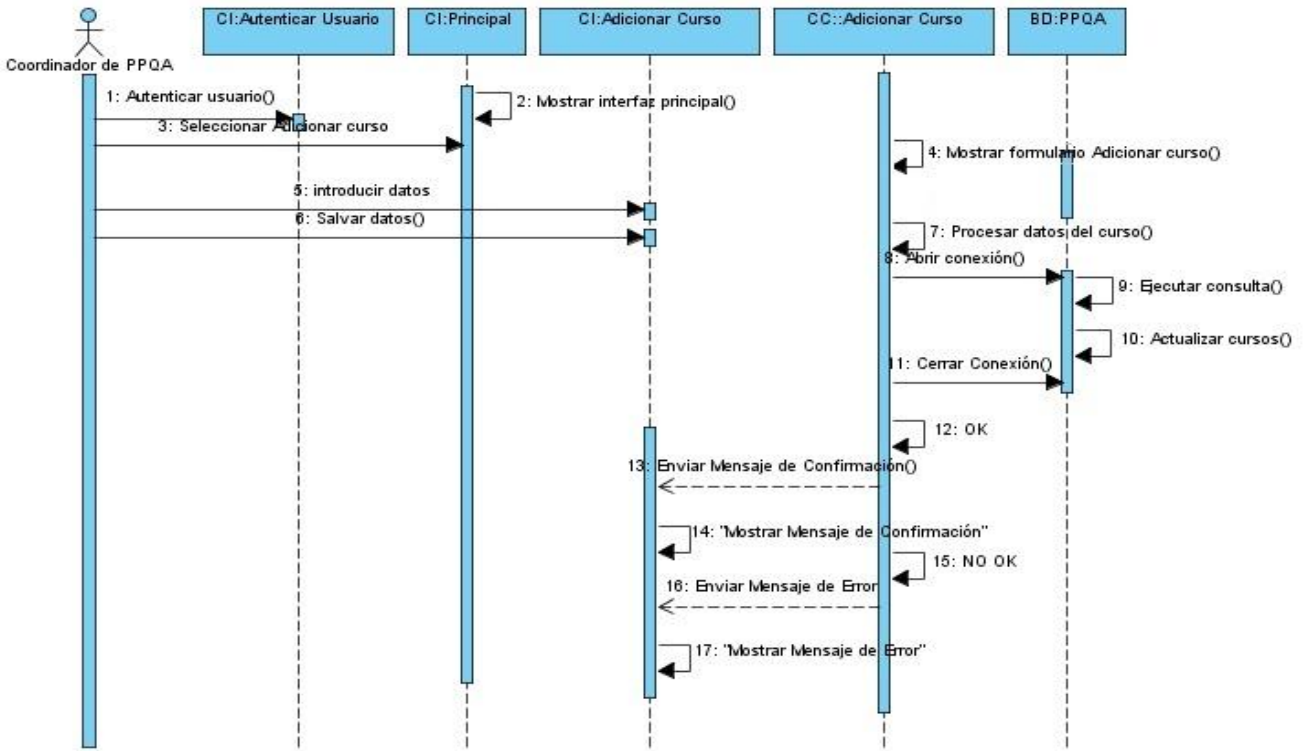


Figura 11. Diagrama de secuencia: Adicionar Curso.

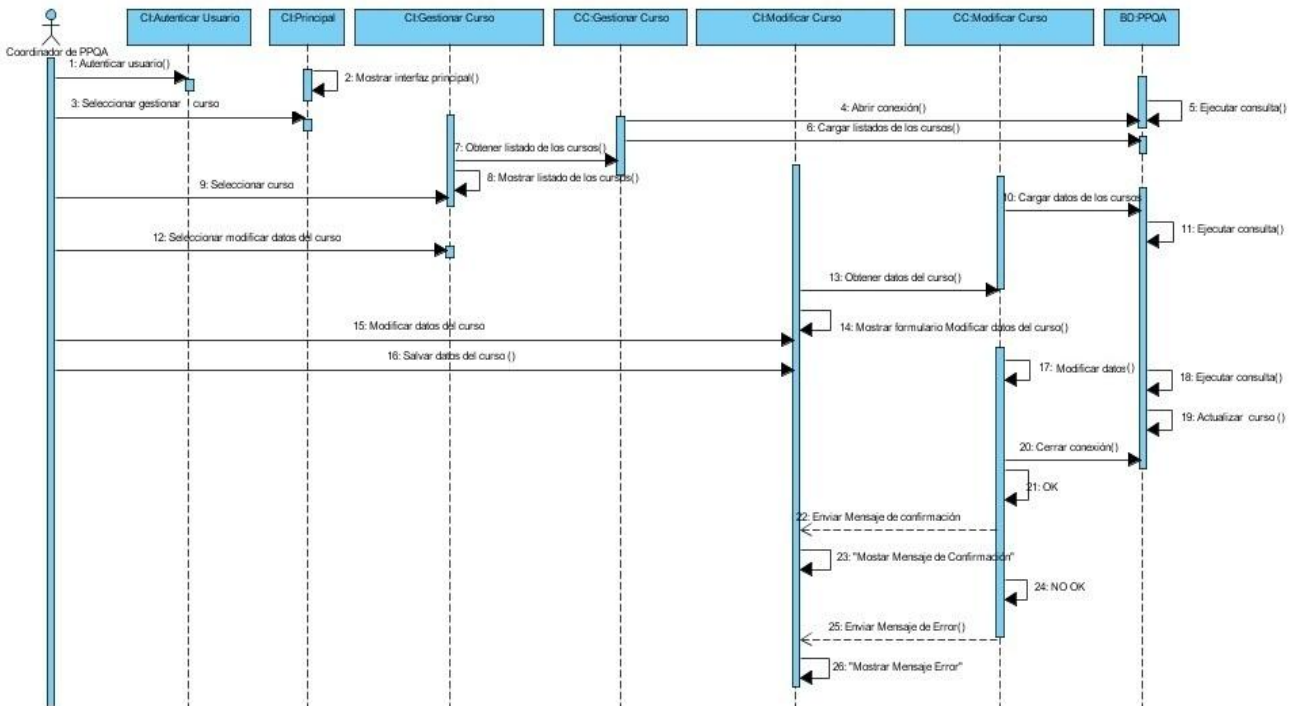


Figura 12. Diagrama de secuencia: Modificar Curso.

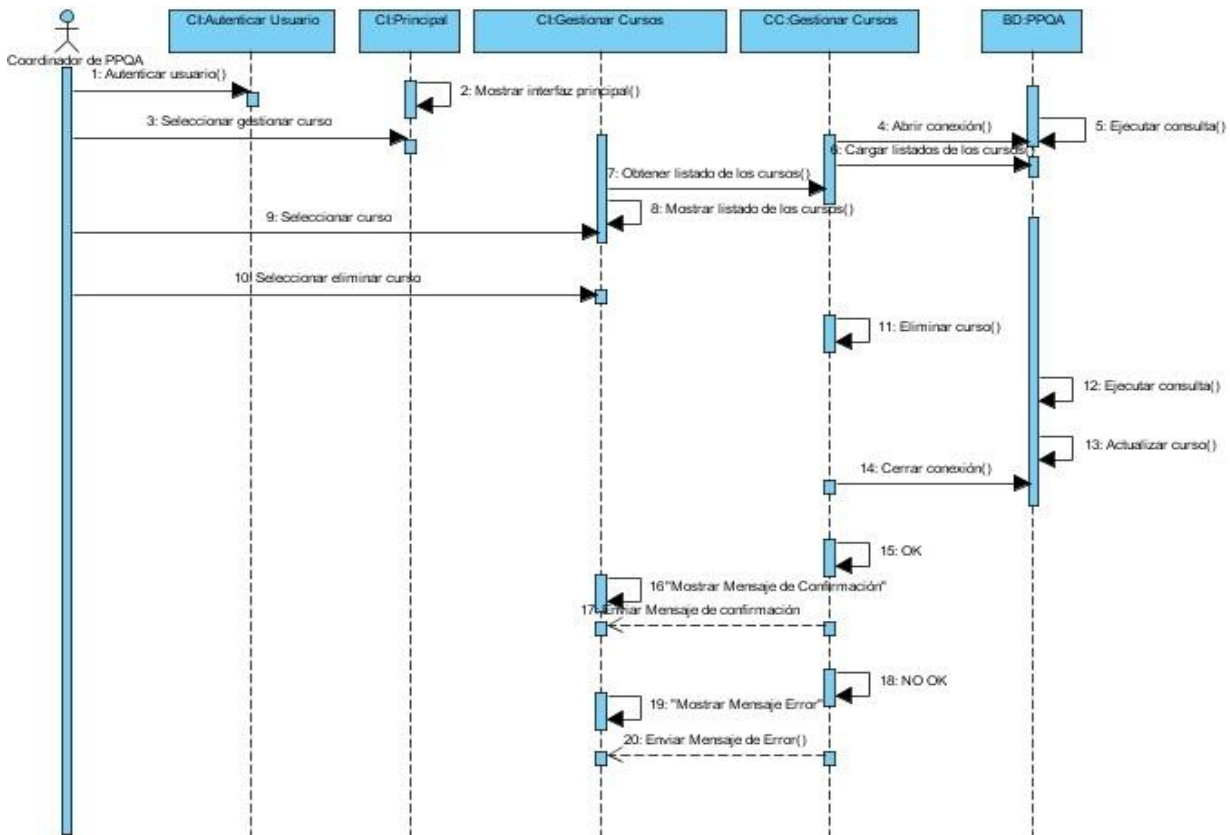


Figura 13. Diagrama de secuencia: Eliminar Curso.

En el Anexo II se encuentra la continuación de los diagramas de secuencias correspondientes a cada caso de uso del sistema. Ver: [Anexo II](#).

3.2 Diseño

El diseño es un proceso iterativo mediante el cual los requisitos se traducen en un plano para construir el software, el cual se encuentra en el núcleo técnico de la ingeniería de software. La importancia del diseño se puede traducir en una sola palabra, calidad, ya que sin este se corre el riesgo de construir un sistema inestable, que fallará cuando se lleven a cabo cambios; que pueden resultar difíciles de comprobar, y cuya calidad no puede evaluarse hasta muy avanzado el proceso [19].

Modelo del Diseño

El modelo del diseño es un modelo de objetos que describe la realización física de los casos de usos centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas en el entorno de implementación, tienen un impacto en el sistema a considerar y sirve de abstracción a la implementación y al código fuente del sistema [19].

3.2.1 Clases del Diseño

Una clase de diseño es una construcción similar a una clase en la implementación del sistema.




Estereotipos Web		
Nombre	Descripción	Representación
Página Cliente	Es una instancia de Página Cliente es una página Web, con formato HTML.	 ««Client page»»
Formulario	Colección de elementos de entrada que son parte de una página cliente.	 ««Form»»
Página Servidora	Representa la página Web que tiene código, que se ejecuta en el servidor.	 ««Server page»»

Tabla 14. Clases del diseño.

3.2.2 Diagramas de Clases del Diseño.

En el diseño se modela el sistema y se encuentra su forma, incluyendo la arquitectura, para que soporte todos los requisitos y otras restricciones relacionadas con el entorno de la implementación, tiene impacto en el sistema a desarrollar. Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es utilizado como entrada fundamental de las actividades de implementación.

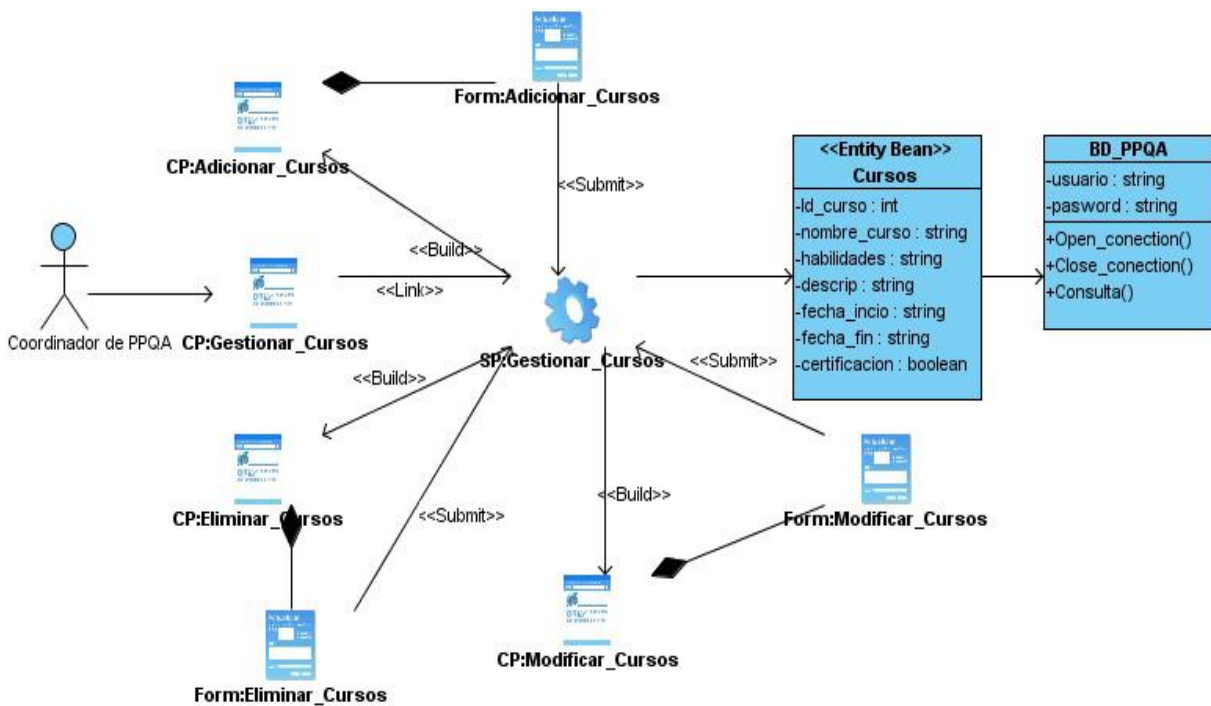


Figura 14. Diagrama de clases del diseño: Gestionar Cursos.

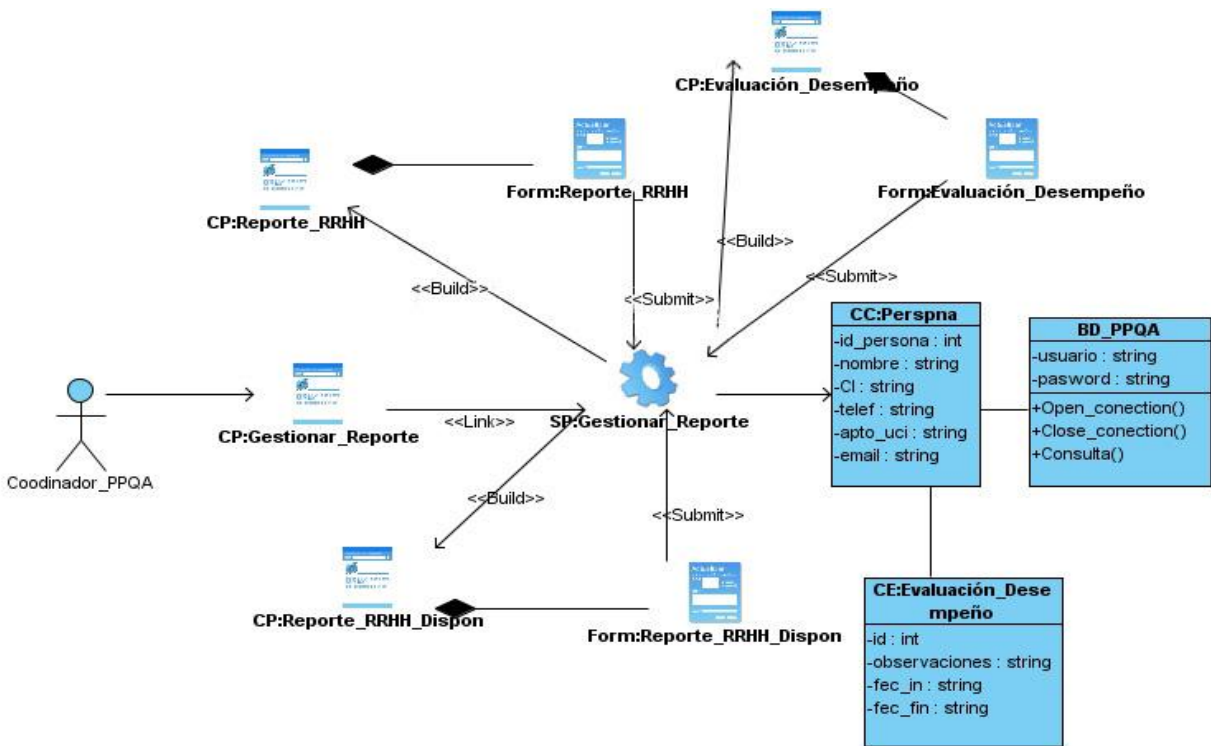


Figura 15. Diagrama de clases del diseño: Generar Reportes.

En el Anexo III se encuentra la continuación de los diagramas de clases del diseño correspondiente a cada caso de uso del sistema. Ver: [Anexo III](#).

3.3 Diseño de la Base de Datos.

Diagrama Entidad - Relación de la BD PPQA.

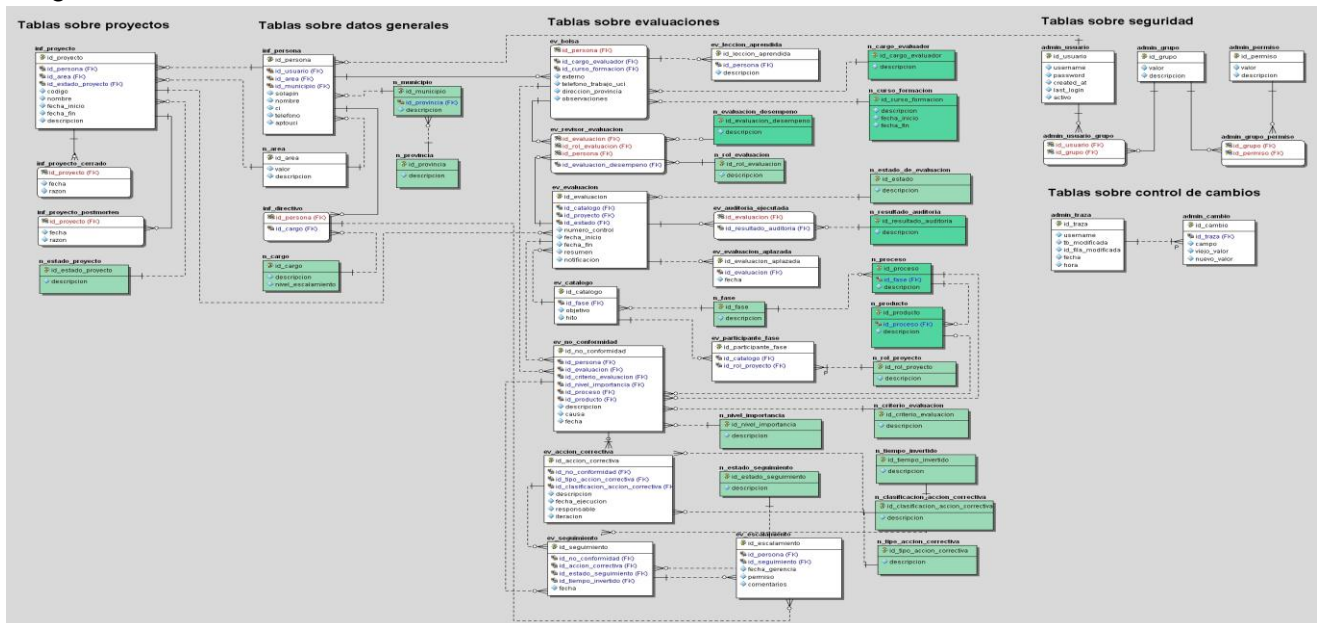


Figura 16. Diagrama Entidad - Relación BD PPQA.

Diagrama Entidad - Relación para el Módulo de Gestión de Capital Humano.

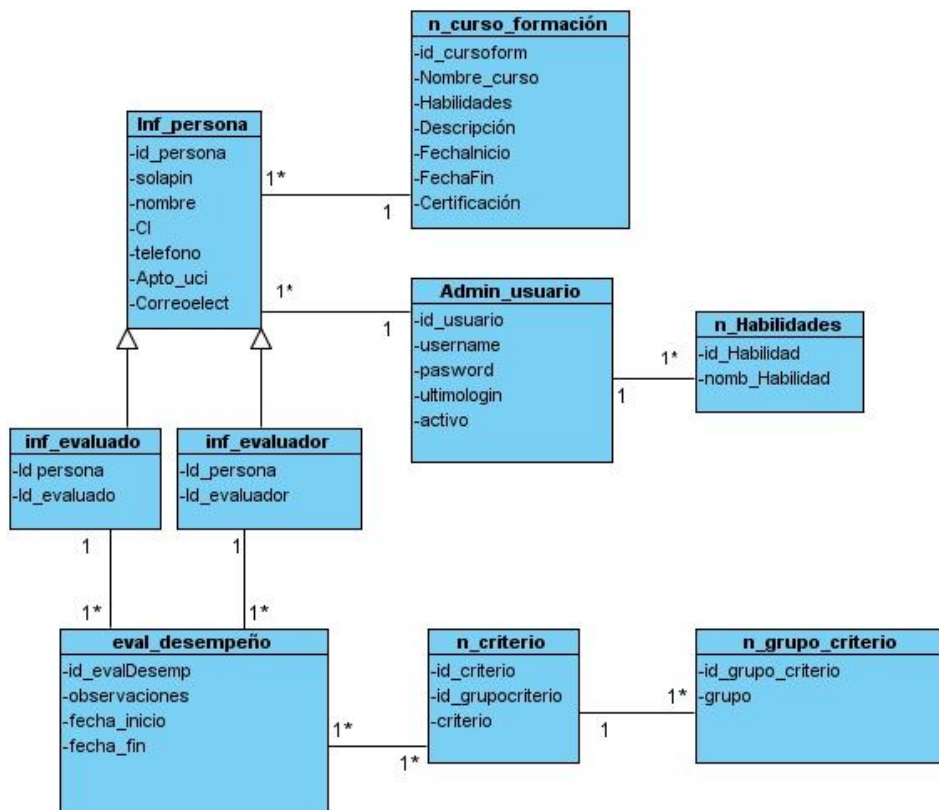


Figura 17. Diagrama Entidad Relación para el Módulo de Gestión del Capital Humano

3.3.1 Descripción de las Tablas.

Nombre: Inf_persona		
Descripción: La tabla almacena información general de las personas que harán uso de la aplicación.		
Atributo	Tipo	Descripción
Id_persona	INTEGER	Almacena el identificador. Es auto incrementable.
solapin	VARCHAR	Almacena el número de solapín.
nombre	VARCHAR	Almacena el nombre.
CI	INTEGER	Almacena el Carnet de identidad.
telefono	VARCHAR	Almacena el teléfono.
aptouci	VARCHAR	Almacena el apartamento donde vive en la uci.

correoelect	VARCHAR	Almacena el correo electrónico.
-------------	---------	---------------------------------

Tabla 15.Descripción de la clase Inf_persona.

Nombre: n_curso de formación		
Descripción: La tabla almacena información de los cursos de formación existentes en la entidad.		
Atributo	Tipo	Descripción
Id_cursoform	INTEGER	Almacena el identificador del curso. Es auto incrementable.
Nombre_Curso	VARCHAR	Almacena el nombre del curso.
Habilidades	VARCHAR	Almacena las habilidades que cubre el curso.
Descripcion	VARCHAR	Almacena una breve descripción del curso.
FechaInicio	DATE	Almacena la fecha que inicia el curso.
FechaFin	DATE	Almacena la fecha que finaliza el curso.
Certificacion	VARCHAR	Almacena si tiene o no certificación el curso.

Tabla 16.Descripción de la clase n_curso de formación.

Nombre: Admin_usuario		
Descripción: La tabla almacena información de los cursos de formación existentes en la entidad.		
Atributo	Tipo	Descripción
Id_usuario	INTEGER	Almacena el identificador que posee el usuario administrador. Es auto incrementable.
username	VARCHAR	Almacena el nombre del curso.
password	VARCHAR	Almacena su contraseña.
ultimologin	VARCHAR	Almacena último momento conectado a la aplicación.
activo	SMALLINT	Almacena si está o no activo. Guarda 0 si no tiene certificación, 1 si tiene certificación.

Tabla 17.Descripción de la clase Admin_usuario.

Nombre: n_Habilidades		
Descripción: La tabla almacena información de las habilidades.		
Atributo	Tipo	Descripción
Id_Habilidad	INTEGER	Almacena el identificador que posee la habilidad. Es auto incrementable.
Nombre_Habilidad	VARCHAR	Almacena el nombre de la habilidad.

Tabla 18.Descripción de la clase n_Habilidades.

Nombre: inf_evaluado		
Descripción: La tabla almacena información de la persona evaluada.		
Atributo	Tipo	Descripción
Id_persona	INTEGER	Almacena el identificador que posee la persona. Es auto incrementable.
Id_evaluado	INTEGER	Almacena el identificador que posee la persona evaluada. Es auto incrementable.

Tabla 19.Descripción de la clase inf_evaluado.

Nombre: inf_evaluador		
Descripción: La tabla almacena información del evaluador.		
Atributo	Tipo	Descripción
Id_persona	INTEGER	Almacena el identificador que posee la persona. Es auto incrementable.
Id_evaluado	INTEGER	Almacena el identificador que posee el evaluador. Es auto incrementable.

Tabla 20.Descripción de la clase inf_evaluador.

Nombre: eval_desempeño		
Descripción: La tabla almacena información de la evaluación de desempeño.		
Atributo	Tipo	Descripción
Id_evalDesemp	INTEGER	Almacena el identificador que posee la evaluación de desempeño. Es auto incrementable.
Observaciones	VARCHAR	Almacena las observaciones de la

		evaluación. Es auto incrementable.
FechaInicio	DATE	Almacena la fecha que inicia la evaluación.
FechaFin	DATE	Almacena la fecha que finaliza la evaluación.

Tabla 21.Descripción de la clase eval_desempeño.

Nombre: n_criterio		
Descripción: La tabla almacena información del criterio general de evaluación.		
Atributo	Tipo	Descripción
Id_grupocriterio	INTEGER	Almacena el identificador que posee el grupo del criterio a evaluar.
criterio	VARCHAR	Almacena el criterio de evaluación.

Tabla 22.Descripción de la clase n_criterio.

Nombre: n_grupo_criterio		
Descripción: La tabla almacena información del conjunto de criterios que poseen los criterios generales.		
Atributo	Tipo	Descripción
Id_grupocriterio	INTEGER	Almacena el identificador que posee el grupo del criterio a evaluar.
grupo	VARCHAR	Almacena el grupo de evaluación.

Tabla 23.Descripción de la clase n_grupo_criterio.

3.4 Patrones de Diseño y de Arquitectura

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias [21].

En el desarrollo del presente trabajo de diploma se evidencian algunos patrones de diseño utilizados y el patrón de arquitectura.

Patrones GRASP

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. Constituyen un apoyo para la enseñanza que ayuda a entender el diseño de objeto esencial y aplica el razonamiento para el diseño de una forma sistemática, racional y aplicable. Los patrones básicos se refieren a cuestiones y aspectos fundamentales del diseño, algunos de estos patrones utilizados en este trabajo son: Experto: Este se encarga de asignar una responsabilidad al experto en información: la clase que cuenta con la información necesaria para cumplir la responsabilidad. Creador: Este patrón se encarga de asignarle a la clase B la responsabilidad de crear una instancia de clase A. B es un creador de los objetos A. Alta Cohesión: Este patrón se encarga de asignar una responsabilidad de modo que la cohesión siga siendo alta. Una clase de alta cohesión posee un número relativamente pequeño, con una importante funcionalidad relacionada y poco trabajo por hacer. Colabora con otros objetos para compartir el esfuerzo si la tarea es grande. Bajo Acoplamiento: Este patrón se encarga de asignar una responsabilidad para mantener bajo acoplamiento. Las clases deben comunicarse con un número pequeño de clases tanto como sea posible. Controlador: Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase [21].

Patrones GoF

Los patrones GoF se clasifican en 3 categorías: Patrones de creación: Muestran la guía de cómo crear objetos cuando sus creaciones requieren tomar decisiones. Estas decisiones normalmente serán resueltas dinámicamente decidiendo que clases instanciar o sobre qué objetos otro delegará responsabilidades. 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 [22].

Tipos de patrones GoF que implementa Symfony. El framework Symfony que se utiliza para la implementación de la solución que aquí se propone, utiliza una serie de patrones GoF: En la categoría de creacionales Symfony utiliza el patrón: Singleton (Instancia única, patrón de creación): Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia.

En la categoría de estructurales Symfony utiliza los patrones: Decorator (Envoltorio, patrones estructurales): Responde a la necesidad de añadir dinámicamente funcionalidad a un objeto. Command: Encapsula peticiones en forma de objetos permitiendo así parametrizar los clientes utilizando distintas peticiones, encolar las peticiones y ofrecer la posibilidad de

deshacer las operaciones. Permite solicitar operaciones sin tener que saber cómo o quien lleva a cabo esas operaciones.

Otros Patrones Registry: Es un medio simple y eficiente de compartir datos y objetos en la aplicación sin tener que preocuparse de mantener numerosos parámetros o hacer uso de variables globales [23].

Patrones de Arquitectura.

Los patrones de arquitectura ayudan a especificar la estructura fundamental de una aplicación, expresan el esquema fundamental de organización para sistemas de software, proveen un conjunto de subsistemas predefinidos, especifican sus responsabilidades e incluyen reglas y guías para organizar las relaciones entre ellos. Cada patrón de arquitectura ayuda a conseguir una propiedad específica en el sistema global; por ejemplo, la adaptabilidad de la interfaz de usuario. Dentro de los patrones de arquitectura se puede encontrar el patrón Modelo Vista Controlador (MVC).

Como ya se ha reflejado para el desarrollo de la aplicación informática que acompaña el presente trabajo de diploma se utilizará el framework Symfony, el cual está basado en el patrón MVC, que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el modelo es el sistema de gestión de base de datos y el controlador representa la lógica del negocio, 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.

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. 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 [24].

Conclusiones

En este capítulo se realizó el diseño e implementación de la solución propuesta. Para ello se expusieron en este documento los diagramas de clases del diseño y diagramas de secuencias correspondientes a cada uno de los requisitos del sistema. Además se realizó una exposición de los principales patrones de diseños utilizados, los cuales permitieron realizar una codificación ágil, sencilla y robusta. La funcionalidad del sistema implementado se pudo comprobar a través de la integración que se realizó con el Sistema de Gestión de Evaluaciones de Software.

4

CAPÍTULO IV IMPLEMENTACIÓN DEL SISTEMA

Introducción

El presente capítulo dará soporte a la implementación del software, en el mismo se incluyen como elementos relevantes el diagrama de componentes del sistema, y la descripción de los diseños de casos de prueba correspondientes a cada uno de las funcionalidades que brindará el Módulo de Gestión de Capital Humano.

4.1 Diagrama de Componentes

Los diagramas de componentes muestran la organización y las dependencias entre un conjunto de componentes. Los componentes físicos pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc. Estos diagramas describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Así como la representación de todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas [19].

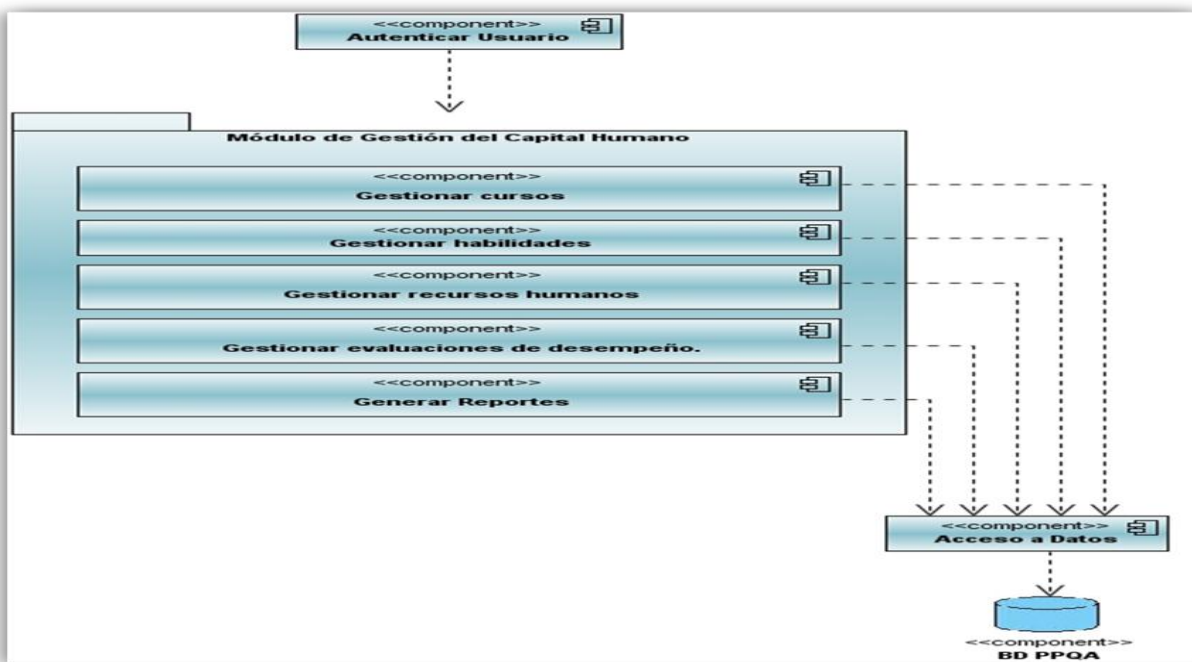


Figura.18: Diagrama de componentes del sistema de alto nivel.

4.2 Diagrama de Despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. Los estereotipos permiten precisar la naturaleza del equipo [19].

El siguiente diagrama muestra el diagrama de despliegue para la aplicación:

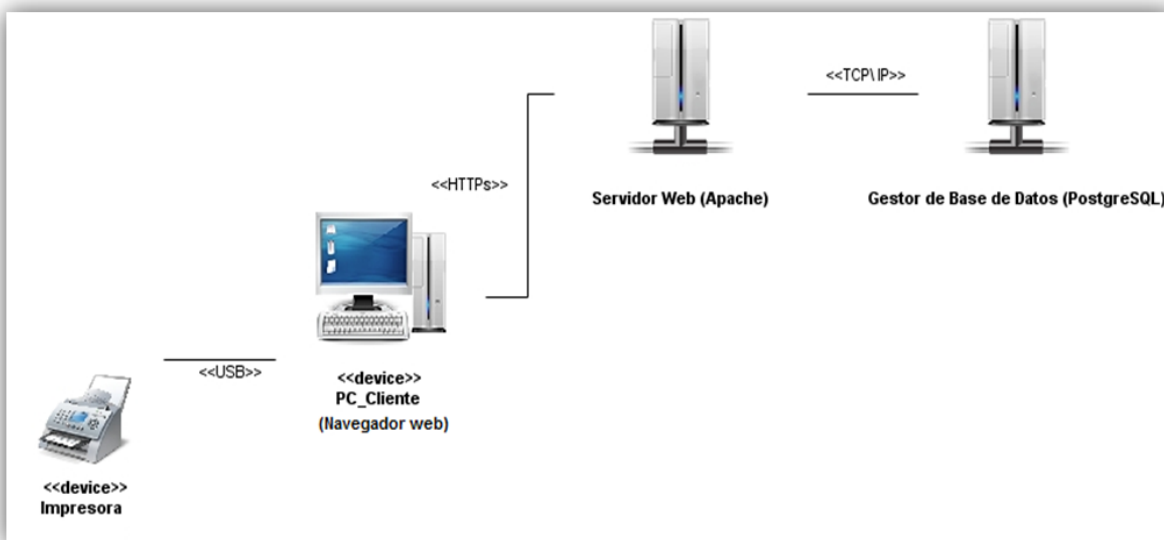


Figura. 19: Diagrama de despliegue.

4.3 Modelo de Prueba

Las pruebas de software es el proceso de ejercitar un programa con la intención específica de encontrar errores previos a la entrega del sistema al usuario final. Están encaminadas a garantizar la calidad del software en todo momento del desarrollo. Durante este flujo de trabajo se comprueba el resultado de la implementación, planificando, diseñando e implementando los casos de prueba.

4.3.1 Pruebas de Caja Negra

Las Pruebas de Caja Negra son llevadas a cabo sobre la interfaz del software, pretendiendo demostrar que las funciones del software son operativas, que la entrada se acepta de forma apropiada, que se produce una salida de forma eficiente y que la integridad de la información

externa se mantiene. Se derivan conjuntos de condiciones de entrada que ejerciten completamente todos los requerimientos funcionales del programa.

4.3.2 Casos de Prueba (CP)

Un caso de prueba es un conjunto de entradas, condiciones de ejecución y resultados esperados desarrollados para un objetivo particular. Los casos de pruebas se pueden derivar de los casos de usos del sistema o de la realización de estos en el modelo de diseño, permitiendo así validar los requerimientos funcionales del sistema.

Para realizar las pruebas del sistema se realizaron pruebas de cajas negras basado en un método de tres pasos para obtener un conjunto de casos de prueba del sistema a partir de los CU; los pasos son los siguientes:

Paso	Descripción	Resultado
1	Generar escenarios de uso.	Obtener los posibles caminos de ejecución de cada caso de uso. Cada camino es un escenario de uso.
2	Identificar casos de prueba.	Conjunto de casos de prueba a partir de los escenarios anteriores.
3	Identificar los valores a probar.	Valores de prueba asociados a cada caso de prueba anterior.

Para generar los casos de prueba se seguirán los siguientes pasos:

- ✓ Para cada CU, generar un sistema completo de los escenarios.
- ✓ Para cada escenario, identificar los casos de pruebas y las condiciones que hagan que se ejecute.
- ✓ Para cada caso de prueba, identificar los valores de los datos con los cuales se harán las pruebas.
- ✓ Las celdas de la tabla contienen V o I, siendo válido e inválido correspondientemente.

Modulo Capital Humano

CUS: Gestionar Evaluación de Desempeño.

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Generar evaluación	Flujo Normal	Flujo alternativo 1
2	Editar evaluación	Flujo Normal	Flujo alternativo 1
3	Exportar evaluación a formato PDF	Flujo Normal	

Escenario del CUS_ Gestionar Evaluación de Desempeño.			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	Se adiciona evaluación del trabajador en la base de datos del sistema. Se notifica al usuario que ha sido evaluado de forma correcta.
CP2	1	I	Se visualizan en el formulario los errores correspondientes, dándole la oportunidad al usuario de corregir los mismos.
CP3	2	V	Se modifica la evaluación general en la base de datos del sistema. Se notifica al usuario que los cambios fueron realizados correctamente.
CP4	3	V	Se exporta la evaluación de desempeño del evaluado en formato PDF.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	Se registran los datos correctamente.	Siendo los datos correctos se almacena en la base de datos la evaluación de desempeño del evaluado. Además se le notifica al usuario con un mensaje de éxito (Ha sido evaluado de forma correcta).
CP2	1	Se registran campos vacíos.	Es notificado al usuario los errores cometidos en el formulario de inserción, dando la posibilidad de corregirlos y

			proceder a la inserción de los datos.
CP3	2	Se registran los datos correctamente.	Siendo los nuevos datos correctos se procede a modificar la evaluación de evaluado correspondiente en la base de datos, notificando posteriormente al usuario con un mensaje de éxito ("Evaluación modificada correctamente").
CP4	3	Una vez creada la evaluación el usuario selecciona la opción de exportar dicha evaluación en formato PDF.	La evaluación es exportada en formato .PDF.

CUS: Gestionar Cursos

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Adicionar curso	Flujo Normal	Flujo alternativo 1
2	Editar curso	Flujo Normal	Flujo alternativo 1
3	Eliminar curso	Flujo Normal	

Escenario del CUS_ Gestionar Cursos			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	Se adiciona el curso a la base de datos del sistema. Se notifica al usuario ("El curso ha sido adicionado de forma correcta").
CP2	1	I	Se visualizan en el formulario los errores correspondientes, dándole la oportunidad al usuario de corregir los mismos.
CP3	2	V	Se modifican los datos curso en la base de datos del sistema. Se notifica al usuario ("El curso ha sido modificado correctamente").
CP4	2	I	Se visualizan en el formulario los errores correspondientes, dándole la oportunidad al usuario de

			corregir los mismos.
CP5	3	V	Se elimina el curso de forma satisfactoria en la base de datos.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	Se registran los datos correctamente.	Siendo los datos correctos se almacena en la base de datos el nuevo curso. Además se le notifica al usuario con un mensaje de éxito (El curso ha sido adicionado de forma correcta).
CP2	1	Se registran campos vacíos.	Es notificado el error cometido en el formulario de inserción, dando la posibilidad de corregirlos y proceder a la inserción.
CP3	2	Se registran los datos correctamente.	Siendo los nuevos datos correctos se procede a modificar el curso correspondiente en la base de datos, notificando posteriormente al usuario con un mensaje de éxito (El curso ha sido modificado correctamente).
CP4	2	Se registran campos vacíos.	Es notificado el error cometido en el formulario de inserción, dando la posibilidad de corregirlos y proceder a la inserción.
CP5	3	El usuario selecciona el curso a eliminar, posteriormente el botón de eliminar.	El curso es eliminado de la base de datos, notificando al usuario con un mensaje de éxito ("El curso ha sido eliminado correctamente").

CUS: Gestionar Habilidades

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Adicionar habilidad	Flujo Normal	Flujo alternativo 1

2	Editar habilidad	Flujo Normal	Flujo alternativo 1
3	Eliminar habilidad	Flujo Normal	

Escenario del CUS_ Gestionar Habilidades			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	Se adiciona la habilidad la base de datos del sistema. Se notifica al usuario que ha sido adicionado de forma correcta.
CP2	1	I	Se visualizan en el formulario los errores correspondientes, dándole la oportunidad al usuario de corregir los mismos.
CP3	2	V	Se modifican los datos correspondientes a la habilidad en la base de datos del sistema. Se notifica al usuario que los cambios fueron realizados correctamente.
CP4	3	V	Se elimina la habilidad de forma satisfactoria en la base datos.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	Se registra la habilidad correctamente.	Siendo los datos correctos se almacena en la base de datos la nueva habilidad. Además se le notifica al usuario con un mensaje de éxito ("La habilidad ha sido adicionada de forma correcta").
CP2	1	Se registra el campo vacío.	Es notificado al usuario los errores cometidos en el formulario de inserción, dando la posibilidad de corregirlos y proceder a la inserción.
CP3	2	Se registran los datos correctamente.	Siendo los nuevos datos correctos se procede a modificar la habilidad correspondiente en la base de datos, notificando posteriormente al usuario con un mensaje de éxito ("La habilidad ha sido

			modificada correctamente")
CP5	3	El usuario selecciona la habilidad a eliminar, posteriormente el botón de eliminar.	La habilidad correspondiente al capital humano seleccionado es eliminada de la base de datos, notificando al usuario con un mensaje de éxito ("La habilidad ha sido eliminada correctamente").

CUS: Gestionar Usuarios

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Adicionar usuario	Flujo Normal	Flujo alternativo 1
2	Editar datos del usuario	Flujo Normal	Flujo alternativo 1

Escenario del CUS_ Gestionar Usuarios			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	Se adiciona el personal en la base de datos del sistema. Se notifica al usuario que ha sido adicionado de forma correcta.
CP2	1	I	Se visualizan en el formulario los errores correspondientes, dándole la oportunidad al usuario de corregir los mismos.
CP3	2	V	Se modifican los datos correspondientes a la persona en la base de datos del sistema. Se notifica al usuario que los cambios fueron realizados correctamente.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	Se registran los datos correctamente.	Siendo los datos correctos se almacena en la base de datos correspondiente a la persona. Además se le notifica al usuario con un mensaje de éxito (Los datos han sido adicionados de forma correcta).
CP2	1	Se registra el campo vacío.	Es notificado al usuario los errores cometidos en el formulario de inserción, dando la posibilidad de corregirlos y proceder a la inserción nuevamente. Además se le notifica al usuario con un mensaje de error(Los datos del usuario no han podido ser guardados).
CP3	2	Se registran los datos correctamente.	Siendo los nuevos datos correctos se procede a modificar los datos de la persona correspondiente en la base de datos, notificando posteriormente al usuario con un mensaje de éxito ("Los datos de usuario han sido modificado correctamente").

CUS: Generar Reportes

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Generar Reporte sobre el desempeño de los revisores.	Flujo Normal	
2	Generar Reporte sobre el estado de los recursos.	Flujo Normal	
3	Generar Reporte sobre recursos disponibles.	Flujo Normal	

Escenario del CUS_ Generar Reportes			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	Se visualiza el reporte sobre el desempeño de los revisores.
CP2	2	V	Se visualiza el reporte sobre el estado de los recursos.
CP3	3	V	Se visualiza el reporte sobre recursos disponibles.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	Se visualiza el reporte sobre el desempeño de los revisores.	El sistema muestra una interfaz con los datos de las últimas evaluaciones realizadas al personal.
CP2	2	Se visualiza el reporte sobre el estado de los recursos.	El sistema muestra una interfaz informada al usuario el estado de personal. Especifica si la persona está disponible o no para realizar las pruebas de los proyectos.
CP3	3	Se visualiza el reporte sobre recursos disponibles.	El sistema muestra una interfaz informando cual es el personal que no posee afectaciones en la agenda para realizar las pruebas de los proyectos.

CUS: Gestionar agenda.

Escenario	Nombre	Flujo de comienzo	Alternativo
1	Actualizar agenda dinámica.	Flujo Normal	

Escenario del CUS_ Gestionar agenda			
ID	Escenario	Datos	Resultado esperado
CP1	1	V	El usuario selecciona el día o la cantidad de días en los que estará afectado para realizar las revisiones de los proyectos.

Casos de Prueba			
ID	Escenario	Datos	Resultado esperado
CP1	1	El usuario selecciona los días en los que estará afectado para realizar las revisiones de los proyectos.	La información es almacenada en la base de datos para la planificación de las tareas del personal y la conformación de los reportes de estado.

4.4 Pruebas de Aceptación

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo, pues sería impresentable al cliente; sino que se realizan sobre el producto terminado e integrado o pudiera ser una versión del producto o una iteración funcional pactada previamente con el cliente. La experiencia muestra que aún después del más cuidadoso proceso de pruebas por parte del desarrollador, quedan una serie de errores que sólo aparecen cuando el cliente comienza a usarlo [19].

4.5 Resultados de las Pruebas

Luego de la ejecución de los 20 casos de pruebas, contando estos con un total de 62 entradas se obtuvieron 56 entradas satisfactorias para 90.4 % contra 6 entradas no satisfactorias para un 9.6 %. Durante estas pruebas fueron detectados y solucionados un total de 6 defectos, para una media de 0.3 defectos por casos de pruebas. No se detectaron más defectos a partir de la segunda iteración.

Igualmente las pruebas de aceptación realizadas en conjunto con el cliente arrojaron conformidad con el módulo desarrollado, obteniéndose finalmente un acta de aceptación que garantiza el cumplimiento de los requisitos establecidos inicialmente.

Aceptación del cliente. Ver: [Anexo IV](#).

4.6 Impacto del Módulo Desarrollado en la Eficiencia de los Procesos de Auditorías y Revisiones de CALISOFT

La eficiencia de un proceso se puede establecer a partir del tiempo invertido en la realización de sus actividades. Un proceso será más eficiente según valla tendiendo este tiempo a disminuir mientras se obtienen los mismos resultados.

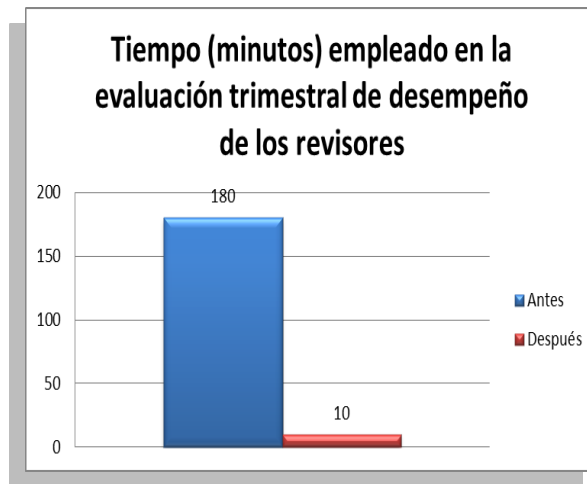
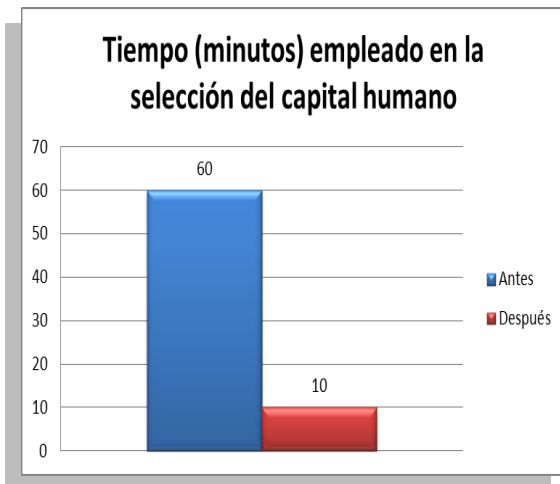
En la siguiente tabla se muestra el esfuerzo estimado en tiempo (minutos) para la realización de cada una de las actividades del proceso de Auditorías y Revisiones que utiliza CALISOFT. Los datos fueron cedidos por los propios especialistas que desarrollaron y utilizan actualmente estos procesos.

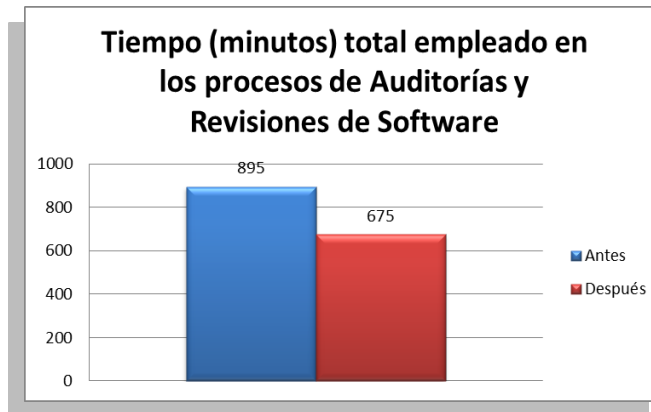
Actividades	Tiempo (Minutos)	
	Antes	Ahora
Planeación y selección del capital humano	60	10
Selección de procesos y productos a evaluar	20	20
Reunión de inicio	30	30
Aplicación de listas de verificación y entrevistas	180	180
Preparar resultados	60	60
Comunicar resultados	30	30
Analizar los resultados	30	30
Entregar Acciones correctivas	5	5
Plan de acción	30	30
Monitoreo de no conformidades	15	15
Escalar no conformidades	15	15
Documentar permiso	15	15
Recolectar información	120	120
Generar información de tendencia	60	60

Evaluación trimestral de desempeño	180	10
Comunicar resultados	30	30
Publicar informes	15	15
Tiempo total del proceso	895	675

Como se puede observar una vez implantada la solución propuesta disminuyeron significativamente tanto los tiempos de planeación y selección del capital humano para una reducción de un 86.4 % así como el tiempo utilizado para la evaluación del desempeño trimestral para un 95.5 %. El ahorro en tiempo total del proceso fue de 24.6 %.

Esto permite confirmar la idea a defender inicialmente donde se planteaba que el desarrollo de la solución permitiría mejorar los niveles de eficiencia (tiempo) en las actividades de selección, seguimiento y evaluación del personal perteneciente al Grupo de Auditorías y Revisiones de Software de CALISOFT.





Conclusiones

En este capítulo se presentó el diagrama de componentes, modelo de despliegue y la realización de las pruebas a la solución propuesta. Los resultados alcanzados en esta última etapa fueron satisfactorios, permitiendo concluir que el sistema desarrollado posee las características y los requerimientos especificados inicialmente por el cliente. Al finalizar el capítulo también se realizó una pequeña comprobación cuantitativa de la idea a defender planteada al inicio del presente Trabajo de Diploma concluyéndose que la misma era válida.

Conclusiones Generales

Durante el presente Trabajo de Diploma se alcanzó el objetivo propuesto inicialmente: Desarrollar un Módulo de Gestión de Capital Humano para el Sistema de Gestión de Evaluaciones de Software del Grupo de Auditorías y Revisiones de Software de CALISOFT. Al finalizar el desarrollo de dicho módulo se realizaron satisfactoriamente las pruebas del mismo.

Se realizó además la aceptación del módulo por el cliente y se integró con el Sistema de Gestión de Evaluaciones de Software de CALISOFT, estando el módulo actualmente en operación.

Los resultados satisfactorios obtenidos permiten considerar que tanto el lenguaje de programación, herramienta de modelado, framework de desarrollo, metodologías utilizadas y demás técnicas fueron las correctas y deben ser utilizadas en desarrollos similares en el futuro.

Recomendaciones

Luego de haber dado cumplimiento a los objetivos de este trabajo y teniendo en cuenta las experiencias adquiridas durante el desarrollo del mismo, se recomienda:

- ✓ Realizar la integración del módulo desarrollado y de todo el Sistema de Gestión de Evaluaciones de Software en general con el Sistema de Gestión de Proyectos (Redmine) que permita la interoperabilidad entre ambos.

- ✓ Implementar servicios web que permitan realizar consultas por terceros y faciliten la comunicación con otros sistemas.

Glosario de Términos

API: Interfaz de programación de aplicaciones (Applications Programming Interface): una serie de funciones que están disponibles para realizar programas para un cierto entorno.

Clase: Descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica.

Diagrama: Representación gráfica de un conjunto de elementos. Visualizan un sistema desde diferentes perspectivas.

Especificación de requisitos: Captura los requerimientos de software para el sistema completo o una porción del mismo.

Framework: se caracteriza por simplificar el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes; proporciona también una estructura al código fuente y facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

GNU: Nombre otorgado por la Free Software Fundación a todo software que se distribuye con su código fuente y que el usuario puede adecuarlo de acuerdo a sus necesidades.

MVC: Modelo Vista Controlador. Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos.

Software: Conjunto de programas y procedimientos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema.

SQL: (Structured Query Language) Lenguaje de Consulta Estructurado es un lenguaje declarativo de acceso a bases de datos relacionales.

TICs: (Tecnologías de la Informática y las Comunicaciones). Conjunto de servicios y tecnologías integradas a un sistema con el objetivo de mejorar los procesos y la calidad de vida.

Plugin: Módulo de hardware o software que añade una característica o un servicio específico a un sistema más grande.

Widgets: Componente con el cual un usuario interactúa en una interfaz gráfica.

Referencias Bibliográficas

- [1] [En línea] www.radioangulo.cu/.../12011-universidad-de-la-informatica-un-centro-superior-de-nuevo-tipo.html, 2010
- [2] Revista Española de Innovación, Calidad e Ingeniería de Software. [En línea] www.ati.es/IMG/pdf/Num2Vol5Sep09.pdf, 2009
- [3] Definición de Capital Humano. [En línea] <http://www.definicionabc.com/economia/capital-humano.php>, 2009
- [4] Fidel Castro, 2005.
- [5] **Cegarra Navarro, Juan Gabriel** .*Influencia de los Componentes del Capital Humano en el Proceso de Aprendizaje Relacional*. Universidad Politécnica de Cartagena Rodrigo Moya, Beatriz, [En línea] <http://www.aedem-virtual.com/articulos/iedee/v09/093159.pdf>, 2007.
- [6] Bunk, G.P., La transmisión de las competencias en la formación y perfeccionamiento profesionales en la RFA, ed. R.C. Nº1., 1994.
- [7] **Yohannis Lartí y Silva, Lic. Frank E. Hernández**. Biblioteca Virtual en Salud. [En Línea] http://bvs.sld.cu/revystas/aci/vol14_1_06/aci03106.htm, 2008
- [8] **Ponjuán, Gloria**. Gestión de la información en las organizaciones. *Principios, Conceptos y Aplicaciones*. "La Era de la Información", 2008.
- [9] **Pressman, Roger**. *Ingeniería del Software. Un Enfoque Práctico*. La Habana: Félix Varela, 2007.
- [10]. **Ponjuán, Gloria**. Gestión de la información en las organizaciones. Principios, Conceptos y Aplicaciones. "La Era de la Información". 2008.
- [11] Davenport, 2000.
- [12] **Potencier, Fabien**. *Symfony la guía definitiva.*, 2008.
- [13] **Rumbaugh, J.; Jacobson, I. y Booch, G**. *El Proceso Unificado de Desarrollo de Software*. 2000.
- [14] CASE: Herramientas CASE. Visual Paradigm for UML

[En línea:] <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5>, , 2007

[15] *Rumbaugh, 2000.*

[16][En línea] www.eumed.net/libros/2009c/587/Anexo.pdf, 2009.

[17] [En línea] www.ecured.cu/.../Flujo_de_Trabajo_Modelo_del_Negocio, 2010.

[18] [En línea] www.eumed.net › Libros, 2011.

[19] **Pressman.** Ingeniería de Software 1. Fase de Inicio. *Flujo de trabajo de requerimientos.* 2002.

[20] [En línea] bdigital.eafit.edu.co/bdigital/proyecto/P005.../marcoTeorico.pdf, 2010

[21]. **López, Sergio.** ComputerWorld. [En línea] [Citado el: 25 de Octubre de 2009.] <http://www.idg.es/computerworld/articulo.asp>.

[22] Patrones de diseño. [En línea] Colectivo de. 2007-2008. UCI.

[23] **Soriano, Ing. Fernando.** Diseño Orientado a Objetos Patrones GoF., 2008

[24] Conferencia 7 IS1 Patrones de diseño. [En línea]. Universidad de las Ciencias Informáticas. La Habana, Cuba., 2005-2006

Bibliografía

Definición de Capital Humano. [En línea] <http://www.definicionabc.com/economia/capital-humano.php>

Cegarra Navarro, Juan Gabriel. *Influencia de los Componentes del Capital Humano en el Proceso de Aprendizaje Relacional.* Universidad Politécnica de Cartagena Rodrigo Moya, Beatriz, 2007[En línea] <http://www.aedem-virtual.com/articulos/iedee/v09/093159.pdf>

Borrego, Daniel. *Gestión del Capital Humano: OrangeHRM*, 2008.

Culebro Juarez M., Gomez Herrera W... *Software libre vs software propietario.* 2006.
[En línea:] <http://www.cyta.com.ar/biblioteca/bddoc/bdlibros/proyectoinformatico/libro/c5/c5>.

Roberto Rodríguez Ríos. *Análisis y diseño del Sistema para la Gestión de la Calidad en la Facultad 8: Módulo Revisiones y Auditorías*, 2009. Universidad de las Ciencias Informáticas”.

José H. Canós, Patricio Letelier ,M^a Carmen Penadés. *Métodologías Ágiles en el Desarrollo de Software.* Valencia : DSIC -Universidad Politécnica de Valencia.

Solarte, Diego Samir Melo. *Bases de datos con software libre* Ingeniero de Sistemas, Facultad de Ingeniería. Universidad de Manizales : s.n.

Pressman, Roger. *Ingeniería del Software. Un Enfoque Práctico.* La Habana: Félix Varela, 2007.

Potencier, Fabien. *Symfony la guía definitiva.* 2008

Cartalla, Alfredo Morales. *Capital Humano, Hacia un sistema de gestión en la empresa cubana.* La Habana: Editora Política, 2009.

Ponjuán, Gloria. *Gestión de la información en las organizaciones. Principios, Conceptos y Aplicaciones.* "La Era de la Información" 2008.

Oktaba, Hanna. *Introducción a Patrones.* [En línea] Facultad de Ciencias, UNAM.



Acta de aceptación

ACTA DE ACEPTACIÓN

En cumplimiento del acuerdo de colaboración realizado entre la Facultad 4 a través del estudiante Santiago Marrero Giró y el Centro de Calidad para Soluciones Informáticas (CALISOFT), y en función de la ejecución del **Módulo de Capital Humano** para el **Sistema de Gestión de Evaluaciones de Software** de CALISOFT se hace entrega del producto que se relaciona a continuación.

Lista de productos que serán aceptados:

- Código fuente del **Módulo de Gestión de Capital Humano** para el Sistema de Gestión de evaluaciones de Software (GESOFT) con la satisfacción de los siguientes requisitos:
 - RF_1. Gestionar cursos.
 - RF_2. Gestionar habilidades.
 - RF_3. Gestionar recursos humanos.
 - RF_4. Gestionar evaluaciones de desempeño.
 - RF_4.1. Exportar a PDF.
 - RF_5. Generar Reportes.
 - RF_5.1. Reportar sobre el desempeño de los revisores.
 - RF_5.2. Reportar sobre el estado de los recursos.
 - RF_5.3. Reportar sobre recursos disponibles.
 - RF_6 Gestionar agenda.

Entrega

Recibe

Nombre y Apellidos: Santiago Marrero Giró

Nombre y Apellidos: Yaima Bety Suárez Gijón

Cargo: Desarrollador de la solución.

Cargo: Jefa del Grupo de Auditorías y Revisiones
de CALISOFT

Firma:

Firma:



Fecha: 31/05/2011