

Universidad de las Ciencias Informáticas

Facultad 3



*“Análisis del Sistema de Gestión por
competencias del subsistema Capital Humano
del centro CEIGE.”*

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

Autor: Carlos Abel González González.

Tutor: Ing. Henry Raúl González Brito.

La Habana, Cuba

Junio 2011

A close-up portrait of Che Guevara, wearing his iconic green cap with a gold star and a green jacket. He has a full, grey beard and is looking slightly to the left. His right hand is raised, with the index finger pointing upwards. The background is a blurred red and white. A handwritten signature in black ink is overlaid on the left side of the image.

Billharts

"Las ideas no necesitan ni de las armas, en la medida en que sean capaces de conquistar a las grandes masas"

DECLARACIÓN DE AUTORÍA



DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores del trabajo titulado: *Análisis del Sistema de Gestión por competencias del subsistema capital humano del centro CEIGE* y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Carlos Abel González

Henry Raúl González Brito

Firma del Autor

Firma del Tutor



AGRADECIMIENTOS

Quiero agradecer a todos los que de una forma u otra en estos 5 inolvidables años de mi vida han estado junto a mí, a los que llegamos hasta el final, a los que les falta un poquitín para llegar y a los que no pudieron, a todos ustedes gracias.

A mi amigo de la infancia Pupito:

- ✓ *Que mas que amigo, mi hermano, tu músico y ingeniero, esta buena la combinación. Gracias por todo.*

A mis hermanos de la UCI Janio, Alexander y Damián:

- ✓ *A ustedes que desde que nos conocimos siempre hemos sido amigos, confidentes, siempre ayudándonos los unos a los otros. Sus consejos, sus acciones para conmigo me han ayudado mucho. Gracias por estar ahí todos estos años.*

A mis hermanas de la UCI Odelkis, Maydel y Lili.

- ✓ *A ustedes les debo mucho, gracias por toda la paciencia que gastaron conmigo, gracias por todas las horas de sueño perdidas estudiando conmigo, gracias por sus consejos, por su cariño, por su comprensión, por llegarme a querer como un hermano. Nunca las olvidaré.*

AGRADECIMIENTOS



✓ *A ti Odelkis que aunque no estés aquí hoy no puedo dejar de agradecerte todo lo que hiciste por mí, creo que si no hubiera sido por ti hoy quizás no estaría aquí leyendo mis agradecimientos, eso nunca lo olvidaré, aquí tienes un hermano para siempre.*

A todos mis compañeros de grupo:

✓ *Gustavo, Yasmany, Perdomo, Leo, Isamil, Gilberto, Ramsés, El ruso, Wilber, Adnerys, Anaibis, Gretter, Danaris, Yurixay, Mariem, Doralis, Elizabeth, Yusney, Daneisy. A todos ustedes muchas gracias por estar día a día compartiendo juntos, aunque fue el destino quien nos unión siempre los recordaré como mis compañeros de la UCI.*

A mi piquete inolvidable, primero los RANGER, después LA CALIENTE:

✓ *Richard, El bosch, El león, Charles, Junito, El Perdo, Osiele, Figueredo, El mena, El wilber, Javier, El alex. A todos ustedes gracias. Cada momento que pasamos juntos en fiestas, fiestas y más fiestas nunca lo olvidaré, para mí son más que amigos son hermanos que durante estos 5 años hemos estado juntos en las buenas y en las malas. Siempre será así, no creo que el tiempo logre borrar todo lo que hicimos juntos.*

AGRADECIMIENTOS



Agradecer además a todos los profesores que durante estos 5 años contribuyeron con mi preparación como ingeniero, en especial a los que me ayudaron en la realización de este trabajo. A todos muchas gracias.



DEDICATORIA

Dedico este trabajo a toda mi familia, en especial a mis padres Carlos y Adalina:

- ✓ *A ti mamá que has estado junto a mí desde que por primera vez abrí mis ojitos en este mundo.*
- ✓ *A ti papá que siempre has estado ahí cuando te he necesitado y cuando no.*

Creo que si tuviera que volver nacer solo lo haría si ustedes dos estuvieran conmigo nuevamente.

A mis abuelos queridos Ada, Orlando, Minerva y Adalberto:

- ✓ *A ti abuela Ada que siempre has estado ahí para malcriarme y darme todos los gustos.*
- ✓ *A ti abuelo Orlando por darme desde que nací todo tu amor y cariño.*
- ✓ *A ti abuela Minerva por siempre saber aconsejarme y estimularme, ya tienes un nieto ingeniero.*
- ✓ *A ti abuelo guarapo que aunque no estás junto a nosotros se que nos quieres mucho y siempre nos tienes presentes.*

A mis abuelitos muchas gracias creo que si no hubiese podido contar con todos sus conocimientos, consejos y amor hoy no sería el hombre que soy.



A mis hermanas Adalis y Adamarys:

Que este ejemplo les llegue con todo mi cariño y amor para que puedan ustedes dos mis hermanitas queridas en un futuro ser mucho mejor que yo, aquí tienen un hermano mayor con el que siempre podrán contar.

A mis tías, primas y primos:

A todos ustedes que siempre han formado parte importante de mi vida y que día a día me han regalado su amor y cariño.

A mi otra mamá Yadira (Bella):

A ti bella por siempre preocuparte por mí, por quererme, por estar siempre que te he necesitado junto a mi papá, por criarme como si fuera tu hijo, eso nunca lo olvidaré.

A mi novia Loreni:

Gracias por estar junto a mí todo este tiempo, quiero que sepas que siempre estés donde estés tendrás tú huequito en mi corazón.

Y a todos aquellos que de una forma u otra durante estos 5 años me han ayudado a convertirme en lo que hoy soy.



RESUMEN

En el presente trabajo de Diploma se exponen los resultados en la realización del análisis de un sistema informático para la gestión por competencias, el cual tiene como objetivo fundamental contribuir a erradicar las deficiencias en el proceso de gestión por competencias de los profesionales del centro CEIGE, mejorando además su nivel científico-técnico. Se presentan el diseño de los artefactos Descripción de los requisitos del sistema Informático, la Descripción de los Casos de Uso, los Diagramas de Clases de Análisis y su validación. Se realiza un estudio de los sistemas informáticos para la gestión por competencias a nivel internacional y nacional.

PALABRAS CLAVE

Competencias, Gestión por competencia, Deficiencias, Gestión.



TABLA DE CONTENIDO

| | |
|--|-----------|
| INTRODUCCIÓN..... | 14 |
| 1.1. Introducción..... | 17 |
| 1.1.1. Ingeniería de Software..... | 17 |
| 1.1.2. Ingeniería de Requerimientos (IR) | 18 |
| 1.1.2.1. Elicitación de Requisitos (ER) | 19 |
| 1.1.2.2. Especificación de Requisitos (ER) | 20 |
| 1.1.2.3. Validación de Requisitos (VRE)..... | 21 |
| 1.1.3. Metodologías de Desarrollo de Software | 22 |
| 1.1.3.1. Rational Unified Process (RUP) | 22 |
| 1.1.4. Herramienta CASE | 23 |
| 1.1.4.1. Visual Paradigm 6.1..... | 23 |
| 1.1.4.2. Axure RP Pro..... | 24 |
| 1.1.5. Lenguaje de modelado | 24 |
| 1.1.5.1. Unified Modeling Language (UML) | 25 |
| 1.1.6. Patrones de Casos de Uso | 25 |
| 1.2. El capital humano y la gestión por competencias..... | 26 |
| 1.2.1. Definición de algunos términos utilizados | 26 |
| 1.2.2. Estudio de sistemas para la gestión por competencias en el mundo | 30 |
| 1.2.3. Estudio de sistemas para la gestión por competencias en Cuba | 31 |
| 1.2.4. Sistemas informáticos para la gestión por competencias en la Universidad de las Ciencias Informáticas. | 32 |
| 1.2.5. Desventajas de los Sistemas estudiados..... | 33 |
| 1.3. Conclusiones Parciales..... | 33 |
| CAPÍTULO 2: CARACTERIZACIÓN DEL SISTEMA | 35 |
| 2.1. Introducción. | 35 |
| 2.2. Problema..... | 35 |
| 2.3. Objeto de Automatización | 35 |
| 2.4. Información que se maneja | 35 |
| 2.5. Modelo de dominio | 36 |
| 2.6. Propuesta de Sistema | 36 |
| 2.7. Modelo del dominio | 37 |
| 2.7.1. Descripción del Diagrama de Clases del Dominio..... | 37 |
| 2.7.2. Diagrama de Clases del Dominio..... | 38 |
| 2.7.3. Descripción de las entidades del dominio. | 39 |

TABLA DE CONTENIDOS



| | |
|--|-----------|
| 2.8. Especificación de los Requisitos del Software | 40 |
| 2.8.1. Requerimientos Funcionales..... | 40 |
| 2.8.2. Requerimientos No Funcionales | 47 |
| 2.9. Definición de los Casos de Uso | 49 |
| 2.9.1 Actores del sistema..... | 49 |
| 2.9.2. Diagrama de casos de uso del sistema..... | 49 |
| 2.9.3. Descripción de los casos de uso del sistema. | 50 |
| 2.10. Conclusiones Parciales | 58 |
| CAPÍTULO 3: ANÁLISIS DEL SISTEMA. | 59 |
| 3.1. Introducción. | 59 |
| 3.2. Análisis..... | 59 |
| 3.2.1. Modelo del análisis | 59 |
| 3.2.2. Clases del análisis | 59 |
| 3.2.3. Diagramas de clases del análisis | 59 |
| 3.3. Validación de los requisitos | 67 |
| 3.3.1. Métrica de software | 67 |
| 3.3.2. Métrica para la calidad de especificación de requisitos de software. | 67 |
| 3.3.3. Técnica de prototipado | 68 |
| CONCLUSIONES..... | 70 |
| RECOMENDACIONES | 71 |
| ANEXOS..... | 74 |
| Anexo 1. Encuestas | 74 |
| Anexo 2. La Ingeniería de Software | 74 |
| Anexo 3 Casos de uso del sistema..... | 75 |



ÍNDICE DE FIGURAS

| | |
|--|----|
| <i>Figura 1. Proceso de Maduración de las competencias, búsqueda y elección. (2010 10th IEEE International)(Giovanni Acampora, Matteo Gaeta, Francesco Orciuoli, y Pierluigi Ritrovato 2010)</i> | 29 |
| <i>Figura 2. Red social e integración con la capa semántica. (2010 10th IEEE International Conferencie)</i> | 30 |
| <i>Figura 3. Diagrama de Clases del Dominio</i> | 38 |
| <i>Figura 4. Diagrama de casos de uso del sistema.</i> | 49 |
| <i>Figura 5. CU- 1 Aprobar Solicitud de matrículas.</i> | 60 |
| <i>Figura 6. CU- 2 Chequear estado de matrícula.</i> | 60 |
| <i>Figura 7. CU- 3 Consultar portafolio.</i> | 60 |
| <i>Figura 8. CU- 4 Consultar reporte administración.</i> | 61 |
| <i>Figura 9. CU- 5 Consultar reporte.</i> | 61 |
| <i>Figura 10. CU- 6 Gestionar Encuesta.</i> | 62 |
| <i>Figura 11. CU- 7 Gestionar matrícula.</i> | 62 |
| <i>Figura 12. CU- 8 Gestionar Portafolio.</i> | 63 |
| <i>Figura 13. CU- 10 Gestionar Roles por departamento.</i> | 63 |
| <i>Figura 14. CU- 11 Gestionar competencias por Rol.</i> | 64 |
| <i>Figura 15. CU- 12 Realizar encuestas.</i> | 64 |
| <i>Figura 16. CU- 13 Realizar petición de nuevo curso.</i> | 64 |
| <i>Figura 17. CU - Gestionar curso.</i> | 65 |
| <i>Figura 18. CU- Gestionar departamento.</i> | 65 |
| <i>Figura 19. Realizar matrícula.</i> | 66 |
| <i>Figura 20. Realizar plan de capacitación.</i> | 66 |
| <i>Figura 21. Autenticar usuario. (Universidad de las Ciencias Informáticas 2011).</i> | 66 |
| <i>Figura 22. La Ingeniería de Software es una tecnología multicapa.</i> | 74 |
| <i>Figura 23. Metodología Rational Unified Process (RUP).</i> | 75 |

TABLA DE CONTENIDOS



ÍNDICE DE TABLAS

| | |
|--|----|
| <i>Tabla 1. Definición de los actores del sistema.</i> | 49 |
| <i>Tabla 2. CU Realizar Solicitud de matrícula</i> | 52 |
| <i>Tabla 3. CU Realizar petición de nuevo curso</i> | 55 |
| <i>Tabla 4. CU Chequear estado de matrícula</i> | 57 |
| <i>Tabla 5. Interpretaciones de requisitos funcionales</i> | 68 |
| <i>Tabla 6. CU Solicitud de matrícula.</i> | 75 |
| <i>Tabla 7. CU Petición nuevo curso.</i> | 76 |
| <i>Tabla 8. CU Chequear estado de la matrícula.</i> | 76 |
| <i>Tabla 9. CU Realizar encuesta.</i> | 76 |



INTRODUCCIÓN

En la actualidad es indiscutible la importancia que tiene la Gestión del Conocimiento en la industria informática. Es un proceso que ha evolucionado en la medida que las entidades se han visto obligadas a enfrentarse a las demandas crecientes del mercado y los constantes cambios tecnológicos de la informática y las comunicaciones (Ikujiro Nonaka 1993).

La Universidad de las Ciencias Informáticas tiene como misión formar profesionales, comprometidos con su patria, calificados en la rama de la Informática, a partir de un modelo pedagógico flexible, que vincula dinámica y coherentemente el estudio con la producción y la investigación, acorde con las necesidades sociales del país y de otros pueblos hermanos (Morell 2003).

La producción se realiza en los centros productivos adscriptos a las facultades de la universidad, es allí donde constantemente se crean nuevos conocimientos y convergen la formación y la investigación con la producción. Uno de estos es el Centro de Informatización de la Gestión de Entidades CEIGE.

Desde su creación el CEIGE se ha caracterizado por mantener un crecimiento constante de la producción, tanto en el marco teórico como de innovación y en la superación profesional (Investigaciones 2009). No obstante persisten un grupo de deficiencias que indican que dicha gestión no se desarrolla con el nivel esperado para este tipo de industria.

- La falta de enfoque en las competencias laborales dentro del centro limitan en gran medida los procesos de selección, contratación y capacitación de los recursos humanos, lo que dificulta el mejoramiento del entorno de trabajo y por tanto el incremento de la productividad y la competitividad.
- Las deficiencias en el modelo de gestión por competencias dificulta las actividades de valoración del conocimiento (captación, selección, organización, presentación) para resolver problemas y aumentar la capacidad de innovación y adaptabilidad de sus trabajadores.
- Las deficiencias en la gestión integrada de los recursos humanos no permiten la correcta integración interna de dicho proceso y la facilitación de un desempeño laboral superior.



- La escasa gestión en el desarrollo de competencias supone un mayor esfuerzo en el desempeño en el puesto de trabajo. Lo que ha imposibilitado que el capital humano se convierta en una ventaja competitiva para el centro.
- La deficiente calidad de los sistemas de información para la toma de decisiones en la gestión por competencias.

La presencia de estos problemas no es casual si a ello se le añade la poca definición existente en los perfiles profesional y competencial que no favorecen a la productividad, el ínfimo desarrollo de los equipos humanos con las competencias necesarias en cada área específica de trabajo, la deficiente evaluación del desempeño que se realiza sobre la base de los objetivos medibles y cuantificables y la ausencia de concienciación de los equipos a que asuman la corresponsabilidad de su propio desarrollo. Por lo antes expuesto no se ha logrado alcanzar el nivel deseado en los resultados de la gestión del conocimiento, lo que a su vez impide trazarse objetivos más intencionados que incrementarían la calidad y competencia de los productos del centro a partir del capital organizacional disponible.

Dada la situación problemática anterior, se puede identificar el siguiente **problema a resolver** Las deficiencias en el proceso de gestión por competencias de los roles de los profesionales del centro CEIGE dificulta la superación profesional de los mismos y su preparación para acometer las actividades de los proyectos en lo que participan. El **objeto de estudio** del presente trabajo se centrará en el Proceso de Gestión por Competencias. El **campo de acción** se enfoca en Sistemas para informatizar el proceso de Gestión por Competencias.

Para responder al problema anteriormente planteado, se tiene como **objetivo general**: Análisis del Sistema de Gestión de Competencias para el centro CEIGE. Como **objetivos específicos** se definieron:

- Establecer el marco teórico relativo a la gestión por competencias del capital humano definido en el objetivo de la investigación para establecer un punto de partida hacia la solución del problema planteado.
- Realizar el análisis del Sistema de Gestión por competencias del subsistema Capital Humano del centro CEIGE para establecer un punto de partida hacia la implementación de la solución.



- Realizar pruebas para validar los resultados de la investigación.

Como posibles resultados:

- Modelación de los artefactos de requisitos y análisis de un sistema informático para la gestión por competencias.
- Definición del Alcance y esfuerzo necesario para el desarrollo de la solución informática.
- Estudio de los sistemas informáticos más representativos en la gestión por competencias.



CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En este capítulo se abordan los aspectos y conceptos relacionados con la gestión por competencias. Se realiza un análisis de las aplicaciones informáticas existentes, tecnologías y tendencias actuales relacionado con la gestión por competencias. Se hace un estudio de las metodologías de desarrollo de sistemas informáticos actuales con el objetivo de establecer las bases teóricas para la realización del proceso de Análisis del sistema. Se explicarán las herramientas escogidas para dar solución a la problemática identificada.

1.1.1. Ingeniería de Software

Durante los primeros años de la informática, no existía una disciplina en el desarrollo del software, la programación era considerada un "arte", y no se controlaba que los ingenieros utilizaran de forma consistente los métodos. El proceso de desarrollo de un sistema se realizaba sin planificación alguna (Menéndez Barzanallana, 2008). La evolución y perspectivas de la Ingeniería del Software posteriormente, desde mediados de la década de los 60 hasta finales de los 70 se caracterizaron por el establecimiento del software como un producto que se desarrollaba para una distribución general. En la actualidad, en la Industria de Software hay tendencia al crecimiento del volumen y complejidad de los productos, los proyectos están excesivamente tarde, se exige mayor calidad y productividad en menos tiempo y hay insuficiente personal calificado (Fernández Pérez, 2007).

Para enfrentar esta situación las empresas requieren desarrollar o adquirir una disciplina en el desarrollo del software y controlar que los ingenieros usen de forma consistente los nuevos métodos. Según (Roger S. Pressman 2007), la Ingeniería de Software es una tecnología multicapa en la que se pueden identificar: los métodos (indican cómo construir técnicamente el software), el proceso (es el fundamento de la Ingeniería de Software, es la unión que mantiene juntas las capas de la tecnología) y las herramientas (soporte automático o semiautomático para el proceso y los métodos). **Ver anexo 2.**



1.1.2. Ingeniería de Requerimientos (IR)

La Ingeniería de Requerimientos se define, según Ortas, como un "conjunto de actividades en las cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución (a veces más de una)." (Ortas 1997)

Ingeniería de Requerimientos, se utiliza para definir todas las actividades involucradas en el descubrimiento, documentación y mantenimiento de los requerimientos para un producto determinado. El uso del término "ingeniería" implica que se deben utilizar técnicas sistemáticas y repetibles para asegurar que los requerimientos del sistema estén completos y sean consistentes y relevantes.

La más notable de las definiciones pertenece a la IEEE Standard Glossary of Software Engineering Terminology (IEEE90). Un requisito es:

1. Condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
2. Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.
3. Una representación documentada de una condición o capacidad como en 1 ó 2.

La determinación de los requisitos de software constituye la base para las etapas futuras de desarrollo, todos los integrantes del equipo a partir de aquí conocen cuales son las funcionalidades que el sistema deberá tener y no solo las funcionalidades, porque en la determinación de los requisitos se tienen en cuenta además las propiedades que el cliente desea para su sistema, de ahí que se separen los requisitos en dos grandes grupos: los funcionales y los no funcionales. Los requisitos funcionales se definen como las capacidades o condiciones que el sistema deberá cumplir o los servicios que deberá brindar y los no funcionales, se definen como las propiedades o cualidades, que deberá tener el software o de qué manera se brindarán esos servicios.

Muchos autores han propuesto estrategias para llevar a cabo la IR, en esta investigación el desarrollo de la IR se basará en el desarrollo de tres de las cuatro actividades propuestas por Pressman: Elicitación, Especificación y Validación de los Requisitos.(Ing. Denise Fernández Miranda y Ing. Yaniuska García Romero 2010)



1.1.2.1. Elicitación de Requisitos (ER)

El proceso mediante el cual se identifica la información que determinan las características deseadas y las restricciones que deberá satisfacer el sistema software es también conocido como Elicitación de requisitos. (Roger S. Pressman 2007)

La elicitación de requisitos proporciona la interacción con los clientes y expertos del negocio con el objetivo de crear las primeras relaciones entre ellos y el equipo de desarrollo, y así obtener la información necesaria para determinar las necesidades reales del cliente, tanto explícitas como implícitas y el dominio del problema.

Antes de identificar los requisitos es necesario conocer el ambiente, incluyendo los sistemas actuales tanto manuales como informatizados, sus aspectos negativos y positivos.

Técnicas de la Elicitación de Requisitos (TER)

Producto de la experiencia obtenida en el desarrollo de la actividad Elicitación de Requisitos los desarrolladores de esta rama han definido un conjunto de técnicas que ayudan a los analistas a llevarla a cabo:

1. Estudio de la documentación: depende de la información que exista en las entidades sobre los procesos y la terminología que se maneja en la misma. Esta técnica es utilizada para capturar requisitos, que luego deben ser validados por otras técnicas pues no se considera efectiva por sí sola.
2. Glosario de términos: Se almacenan los términos y significados relativos al problema, esta técnica intenta registrar el conocimiento obtenido del dominio del problema y compartirlo con todos los participantes. El glosario puede ser tan autocontenido como sea posible, en donde los términos estén relacionados entre sí, a esta técnica se le conoce como Principio de Circularidad.
3. Modelado de Negocio: En el modelo de negocio se describen los procesos del negocio, los trabajadores del negocio, las entidades, sus clases y relaciones existentes entre ellas.
4. Entrevistas: Es la más utilizada, porque es la forma natural de entendimiento de los humanos. Es importante conocer el vocabulario del dominio del problema para lograr un buen entendimiento con el cliente.



5. Desarrollo Conjunto de Aplicaciones ó Join Application Development (JAD): Es una alternativa a las entrevistas, se desarrolla durante varios días y el cliente participa junto al equipo de desarrollo dando opiniones y posibles soluciones. Esta técnica intenta atraer más al cliente para que no se sienta fuera del equipo, se apoya en el uso de técnicas audiovisuales, diagramas y herramientas CASE (**Ver 1.1.4**). Tiene como ventaja que los clientes pueden revisar la documentación que se genera, sin embargo, es muy compleja de organizar y depende de los horarios de los clientes y usuarios.

A pesar de existir muchas técnicas para desarrollar la ER, por la naturaleza del proyecto se han decidido combinar en el presente trabajo las técnicas: estudio de la documentación, entrevistas y JAD.

1.1.2.2. Especificación de Requisitos (ER)

También conocida como Definición de los Requisitos, es la manera habitual de registrar los requisitos identificados. La especificación describe las funciones y características de un sistema y las restricciones que gobiernan su desarrollo (Roger S. Pressman 2007). Para las ER se han definido un grupo de características, dentro de las cuales se tienen las siguientes:

1. Correcta: Todo requisito en la ER contribuye a satisfacer una necesidad real.
2. No ambigua: Todo requisito posee una sola interpretación.
3. Completa: Todo lo que se supone que el software debe hacer está incluido en la ER.
4. Internamente Consistente: No existen subconjuntos de requisitos contradictorios.
5. Verificable: Para cada requisito expresado en la ER existe un procedimiento de prueba finito y no costoso para demostrar que el futuro sistema lo satisface.
6. Modificable: Su estructura y estilo permiten hacer cambios a los requisitos fácilmente.
7. Precisa: La ER hace uso de valores numéricos para precisar las características del sistema.
8. Realizable: Si con los recursos del proyecto pueden implementarse los requisitos de la ER.
9. Concisa: La ER debe ser lo más breve posible sin afectar a los atributos de calidad.

Técnicas de la Especificación de Requisitos (TER)

Para desarrollar la ER se proponen un conjunto de técnicas, dentro de ellas las más significativas son las siguientes:



1. Plantillas: Se describen los requisitos mediante el lenguaje natural pero de forma estructurada. Esta se conforma por una tabla con una serie de campos y una estructura predefinida que el equipo de desarrollo va a dar cumplimiento. Las plantillas eliminan parte de la ambigüedad del lenguaje natural al estructurar la información, cuanto más estructurada sea esta, menos ambigüedad ofrece.
2. Escenarios: La técnica de los escenarios consiste en describir las características del sistema informático a desarrollar mediante una secuencia de pasos. La representación del escenario puede ser casi textual o ir encaminada hacia una representación gráfica en forma de diagramas de flujo. El análisis de los escenarios, hechos de una forma u otra, pueden ofrecer información importante sobre las necesidades funcionales del sistema.
3. Casos de uso: Se propone dentro de la metodología de desarrollo RUP, sin embargo, varios autores defienden que pueden resultar ambiguos a la hora de definir los requisitos, por lo que hay propuestas adicionan descripciones basadas en plantillas o de diccionarios de datos que eliminen su ambigüedad.

1.1.2.3. Validación de Requisitos (VRE)

En esta actividad se comprueba que las ER se ajustan a las necesidades de clientes/usuarios y otros implicados, se valida por los usuarios que sus necesidades fueron adecuadamente interpretadas, se desarrollan actividades de verificación de requisitos con el fin de comprobar que las ER se construyeron de acuerdo con los estándares establecidos, sin ambigüedad, sin inconsistencias, sin omisiones, que hayan sido corregidos los errores detectados y que el resultado del trabajo se ajuste a los estándares establecidos para el proceso, el proyecto y el producto (Roger S. Pressman 2007)

Técnicas de la Validación de Requisitos (TVRE)

Existen un conjunto de técnicas que se utilizan para llevar a cabo la Validación de los Requisitos.

a) Revisiones: Está técnica consiste en la lectura y corrección de la completa documentación o modelado de la definición de requisitos. Con ello solamente se puede validar la correcta interpretación de la información transmitida.



b) Auditorías: Esta técnica consiste en la revisión de la documentación, lo que permite controlar los resultados contra una lista de chequeo predefinida o definida a comienzos del proceso, es decir, sólo una muestra es revisada.

c) Matrices de trazabilidad: Esta técnica consiste en marcar los objetivos del sistema y chequearlos contra los requisitos del mismo. Es necesario ir viendo qué objetivos cubre cada requisito, de esta forma, se podrán detectar inconsistencias u objetivos no cubiertos.

d) Prototipos: Permite al usuario hacerse una idea de la interfaz del sistema. Esta técnica tiene como problema que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final. Con la VRE se pretende descubrir los problemas que la ER pueda tener antes de comprometer recursos en su implementación. En esta investigación se seleccionaron las revisiones y los prototipos como las técnicas a desarrollar en la validación de los requisitos (Ing. Denise Fernandez Miranda e Ing. Yaniuska García Romero 2010)

1.1.3. Metodologías de Desarrollo de Software

Las metodologías se desarrollan con el objetivo de dar solución a los problemas existentes en la producción de software, que cada vez son más complejos. Estas engloban procedimientos, técnicas, documentación y herramientas que se utilizan en la creación de un producto de software.

Dentro de las metodologías más utilizadas hoy día se pueden citar: RUP, XP y MSF, de ellas Rational Unified Process (RUP) se adapta mejor a los proyectos de largo plazo, al dividir el desarrollo de software en cuatro fases desarrolladas iterativamente, lo que permite abarcar seis disciplinas ingenieriles y tres de apoyo; Extreme Programming (XP) por su parte se recomienda para proyectos cortos y tiene por particularidad contar con el usuario final como parte del equipo; Microsoft Solution Framework (MSF) se adapta por su parte a cualquier tipo de proyecto y se centra en los modelos de procesos y de equipo (Mendoza Sánchez, 2004).

1.1.3.1. Rational Unified Process (RUP)

El Proceso Unificado de Desarrollo (RUP-Rational Unified Process) está pensado en dos dimensiones. El mismo se aplica a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo concluye con una versión del producto para los clientes. Cada ciclo consta de cuatro fases. Cada fase termina con un hito. RUP, en su modelación define como sus principales elementos:



- Trabajadores (“quién”): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto como un equipo. Ellos realizan las actividades y son propietarios de elementos.
- Actividades (“cómo”): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos
- Artefactos (“qué”): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- Flujo de actividades (“Cuándo”): Secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

El ciclo de vida de RUP se caracteriza por:

- Dirigido por casos de uso.
- Centrado en la arquitectura.
- Iterativo e Incremental.

Cuenta con 4 fases, 9 flujos de trabajos, los primeros seis son conocidos como flujos de ingeniería y los tres últimos como de apoyo.

Finalmente, en cuanto a su duración, se considera como sano una fase de elaboración de dos o tres iteraciones. Esto debido a la necesidad de realizar y probar el diseño, lo que genera una carga de documentación y de programación sensiblemente mayores a las que se tenían durante la fase de concepción. (Jacobson Ivar, Rumbaugh James, y and Booch, Grady 2000)

1.1.4. Herramienta CASE

Las siglas CASE en inglés significa Computer Aided Software Engineering y según su traducción en español es Ingeniería de Software Asistida por Computadoras.

1.1.4.1. Visual Paradigm 6.1

Visual Paradigm es una herramienta que ofrece un entorno de creación de diagramas que usa las notaciones UML 2.0, con un diseño centrado en casos de uso y enfocado además al negocio, lo cual



genera un software de alta calidad. Es importante destacar que la licencia de Visual Paradigm es muy restringida.

Debido a las funcionalidades brindadas por el Visual Paradigm al permitir crear diagramas UML y por ser una herramienta multiplataforma que se integra fácilmente con varios Entorno de desarrollo integrado (IDEs), se decidió por parte de la dirección del proyecto usarla como herramienta durante todo el proceso de modelado.

1.1.4.2. Axure RP Pro

Herramienta para la rápida creación de prototipos y especificaciones para aplicaciones y sitios web. Demuestra su grado de especialización en las anotaciones, en este punto, permite especificar el estado de cada elemento (Propuesto, Aceptado, Incorporado), el beneficio esperado (Crítico, Importante, Útil), el riesgo, la estabilidad, a quién va dirigido y a quién se le asignará la tarea.

Características:

- Fácil de usar.
- Permite realizar cambios muy rápidos en la aplicación a diseñar.
- La creación instantánea de prototipos funcionales.
- Creación de prototipos interactivos con anotaciones.
- Personalización.
- Versión de administración.
- Creación de un diseño efectivo.
- Experimentar con su propio diseño para lograr los máximos resultados.
- Automatización de la especificación para evitar la tediosa documentación.

1.1.5. Lenguaje de modelado

Un lenguaje de modelado de objetos se puede definir como el conjunto estandarizado de símbolos y de modos de disponerlos para modelar (parte de) un diseño de software orientado a objetos.



1.1.5.1. Unified Modeling Language (UML)

UML es un lenguaje, que permite modelar, analizar y diseñar sistemas orientados a objetos. Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad (Jacobson Ivar, Rumbaugh James, y and Booch, Grady 2000). Ofrece un estándar para describir un panorama del sistema modelo, incluyendo aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables y aspectos conceptuales como los procesos de negocios y funciones del sistema.

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. La finalidad de los diagramas es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. Debido a que el UML es un lenguaje, cuenta con reglas para combinar tales elementos.

1.1.6. Patrones de Casos de Uso

Los patrones de software describen un problema que ocurre repetidas veces en algún contexto determinado del proceso de desarrollo de software, y entregan una buena solución ya probada. Esto ayuda a diseñar correctamente en menos tiempo y a construir problemas reutilizables y extensibles, y facilita la documentación y la comunicación con otros miembros del equipo de desarrollo. (Gunnar Overgaard, Karin Palmvist, y Addison-Wesley 2005).

Para el desarrollo del trabajo se utilizaron los siguientes patrones:

CRUD- Completo (Crear, Modificar, Eliminar, Mostrar): Este patrón se basa en la fusión de casos de uso simples para formar una unidad conceptual. Consta de un caso de uso, llamado Información CRUD o Gestionar información que modela todas las operaciones que pueden ser realizadas sobre una parte de la información de un tipo específico, tales como creación, lectura, actualización y eliminación. (Fowler, Martin 2007). Suele ser utilizado cuando todos los flujos contribuyen al mismo valor del negocio, y estos a su vez son cortos y simples. Existe un patrón alternativo llamado CRUD Parcial, que modela una de las vías de los casos de uso como un caso de uso separado. Es preferiblemente utilizado cuando una de las alternativas de los casos de uso es más significativa, larga o más compleja que las otras. (Jacobson Ivar, Rumbaugh James, y and Booch, Grady 2000)



Actores múltiples: Rol común: Actores múltiples: Rol común es un patrón de estructura que plantea que cuando dos actores juegan el mismo papel hacia un caso de uso se representa otro actor, del que heredan los actores que comparten este rol. Este patrón es aplicable cuando, desde el punto de vista de un caso de uso, hay solo una entidad externa interactuando con cada instancia del caso de uso. (Fowler, Martin 2007)

1.2. El capital humano y la gestión por competencias

Actualmente, el departamento de recursos humanos se ha convertido en un departamento estratégico para la empresa, su recurso más importante son, cada vez más, las personas que la integran. En esta nueva visión el departamento de recursos humanos debe ocuparse de seleccionar, formar, valorar e incentivar a los empleados para garantizar que la organización cuente con personas más capacitadas y comprometidas. Su papel es proactivo y se le pide que aporte valor añadido, además de resolver los problemas sociales de la empresa. (Norberto Pelegrín Entenza y Margarita Sosa Monzón 2004)

1.2.1. Definición de algunos términos utilizados

Capital Humano: Es el aumento en la capacidad de la producción del trabajo alcanzado con mejoras en las capacidades de los trabajadores. Estas capacidades realizadas se adquieren con el entrenamiento, la educación y la experiencia. Se refiere al conocimiento práctico, las habilidades adquiridas y las capacidades aprendidas de un individuo que lo hacen potencialmente.

Competencias: Existen diversos conceptos relacionados con el término competencia según diferentes autores; a continuación se muestran algunos de estos.

Para algunos como (Boyatzis 1982) la competencia es una característica personal subyacente que determina los comportamientos llevados a cabo. La concepción de Boyatzis implica que la competencia es una capacidad, un conocimiento o una habilidad que se traduce en un comportamiento.

Otros autores como (Pereda y Berrocal 2004) consideran las competencias como un conjunto de comportamientos observables que dan lugar a un desempeño bueno o excelente en un trabajo concreto y en una organización concreta. Las competencias son el resultado de combinar los conocimientos, las habilidades, actitudes e intereses y las motivaciones en un determinado contexto. Las competencias no son conocimientos, habilidades o actitudes aisladas, son comportamientos que



integran cinco componentes del desempeño laboral: conocimientos, habilidades y destrezas, actitudes, motivaciones y los medios y recursos necesarios para llevar a cabo los comportamientos incluidos en la competencia.

Maura González conceptualiza el término competencia como: “configuración psicológica compleja que integra en su estructura y funcionamiento formaciones, motivacionales, cognitivas y recursos psicológicos que se manifiestan en la calidad de la actuación profesional del sujeto, y que garantizan un desempeño profesional responsable y eficiente”. (González, 2002)

El Dr. Armando Cuesta Santos define competencias como: “características subyacentes en las personas, asociadas a la experiencia, que como tendencia están causalmente relacionadas con actuaciones exitosas en un puesto de trabajo, contextualizado en determinada cultura organizacional”. (Cuestas, 2005)

Competencia: Descripción de la conducta personal, calificación de rendimiento específico de un individuo o de una organización, se puede utilizar para clasificar a los empleados de conformidad con los requisitos del modelo de competencias establecido, estado que califica los cambios de comportamiento de un individuo, califica el propósito de lograr un alto rendimiento y por lo tanto permite a las empresas lograr un alto rendimiento, y en fin de obtener una ventaja competitiva sostenible. (Hong-Mei Jin y Shu-Ming Wang 2010).

Competencias laborales: Conjunto sinérgico de conocimientos, habilidades, experiencias, sentimientos, actitudes, motivaciones, características personales y valores, basado en la idoneidad demostrada, asociado a un desempeño superior del trabajador y de la organización, en correspondencias con las exigencias técnicas, productivas y de servicios. Es requerimiento esencial que esas competencias sean observables, medibles y que contribuyan al logro de los objetivos de la organización. (Oficina Nacional de Normalización (NC 3000) 2007)

Gestión por competencias: Actividades coordinadas para dirigir y controlar una organización con un enfoque basado en las competencias laborales y la capacidad de aprendizaje de los trabajadores. Su objetivo es una organización de calidad y la disposición del colectivo integrado para el logro de los objetivos de la organización. (Oficina Nacional de Normalización (NC 3000) 2007). Esencialmente, la gestión de competencia se refiere a la habilidad que las organizaciones han de encontrar, utilizar, mantener y desarrollar en su fuerza de trabajo para alcanzar sus metas. La gestión por competencias



se centra en las personas, en lo que puede ser mejorado y desarrollado en lugar de aquellas actividades que son transitorias y obsoletas. El objetivo principal de la gestión por competencias es definir y mantener constantemente las competencias de las personas de acuerdo con la metas de la organización por lo que es una parte importante de los recursos humanos. (Cristiane Casaca CSI - Control and Systems and Information Technologies 2010)

Modelo de competencias: Modelo de competencias se refiere a una serie de combinación de competencias que se pueden combinar para el éxito de los individuos en el trabajo o una organización en particular. Se trata de un grupo de conocimientos, habilidades y actitudes que pueden afectar a la directamente el rendimiento en el trabajo de una persona, puede ser medida por un estándar fiable y ser capaz de mejorar mediante la capacitación y el desarrollo. Representa a un grupo de características de comportamiento requerido por puestos de trabajo o un nivel específico que es capaz de identificar los motivos, cualidades, habilidades y como las habilidades entre los que tienen un rendimiento excepcional y la mediocre. Debido a las diferencias en definición de las personas de la competencia y las diferencias en los elementos que constituyen un modelo de competencias, el modelo de competencias también tienen grandes diferencias. La construcción del modelo de competencias suele ser adaptada a trabajos específicos y varía tanto como los resultados de la organización(Hong-Mei Jin y Shu-Ming Wang 2010)

Estructura de un Modelo de Gestión por Competencias: Confección del catálogo de competencias: a partir de la descomposición de las tareas que es necesario hacer para un proceso, se identifica lo que se necesita saber hacer para ejecutar eficientemente dichas tareas. Estructuración de conocimiento dentro de un esquema predefinido: una vez identificadas las competencias se debe trabajar en la descripción clara de cada uno de sus niveles, para poder realizar las calificaciones en forma eficiente y objetiva. Identificar los requerimientos de competencias para un puesto o equipo de trabajo: de esta forma, se puede individualizar el grado de adecuación y la forma de cubrir las posibles brechas, así se podrá identificar a los colaboradores que cuentan con conocimientos críticos dentro de la empresa y su potencial uso, lo que posibilita evitar que el ejecutivo en una división dependa exclusivamente de los recursos que tenga a su alcance, lo que permitirá asignar siempre a la persona más idónea para el puesto, y solo en el evento que no exista internamente se podrá recurrir a la contratación de empresas externas.(Gary Stanley Becker 1993)



Figura 2. Red social e integración con la capa semántica. (2010 10th IEEE International Conference)

Restricciones Del Modelo: Resistencia al cambio: Es una conducta natural del ser humano ante cada situación de cambio, ante cada propuesta diferente, ante todo aquello que dista de alguna medida de nuestro esquema de pensamiento y acción vigente. (Gary Stanley Becker 1993)

1.2.2. Estudio de sistemas para la gestión por competencias en el mundo

En la actualidad existen en el mundo diversos sistemas informáticos con el objetivo de gestionar el capital humano que serán presentadas a continuación:

Cezanne Software: Provee soluciones Administración de Capital Humano para contribuir en el desarrollo y retención del Capital Humano Incluyen aplicaciones para la gestión del rendimiento del empleado, planes de sucesión y carreras, formación y desarrollo, gestión de las personas, selección, análisis salarial, planificación retributiva, revisión salarial, encuestas y diseño de organigramas(<http://www.cezannesw.com/es/> 2011).

ION: Software de Gestión Estratégica de Recursos Humanos

ION le proporciona una forma de medir, controlar, administrar y mejorar la Inteligencia Organizacional de su empresa. Permite gestionar y realizar el seguimiento de los objetivos organizacionales, agilizar los procesos y planificar el desarrollo del Capital Humano. Con la implementación de ION puede



mejorar los procesos de selección y evaluación de desempeño para obtener información en cualquier momento, con el fin de realizar medidas correctivas.

Gestiones por competencias

- ION incorpora el modelo de Gestión por Competencias que permite un análisis inteligente de sus procesos de negocio.
- Desarrolla capacidades y competencias de su personal y basándose en los objetivos empresariales, monitorea el desempeño de cada uno de sus colaboradores.
- Esta herramienta es completamente adaptable y flexible a las necesidades organizacionales,
- Carácter dinámico
- Posibilita la gestión gráfica detallada, que abarca, desde una competencia raíz, hasta las diferentes evidencias observables (con sus correspondientes grados de aparición).(ION company 2010)

INFOR HCM (Human Capital Management). Administración del capital humano

- INFOR HCM: es un sistema de gestión por competencias del capital humano habilitado en el Internet utilizado por compañías emprendedoras alrededor del mundo para convertir los recursos humanos en una ventaja competitiva. Este software de RH tiene cabida para varios idiomas, tipos de moneda, bancos, definiciones fiscales, prestaciones, configuraciones de seguridad y plataformas al tiempo que incluye los siguientes componentes.(INFOR company 2011)

1.2.3. Estudio de sistemas para la gestión por competencias en Cuba

SIMAPRO en Cuba: sector azucarero

Antecedentes: En el año 2000 el Ministerio de Trabajo y Seguridad Social de Cuba (MTSS), a través del Instituto de Estudios e Investigaciones del Trabajo (IEIT) adscrito a él, y con la colaboración del Sr. Leonard Mertens, Consultor de la OIT Regional para México y Cuba, comenzó la aplicación de la metodología SIMAPRO en la Empresa Azucarera "30 de Noviembre", perteneciente al Grupo Empresarial Agroindustrial de Pinar del Río, en las áreas del Basculador y Molino, para el período de



Zafra, experiencia que, por su impacto en los indicadores productivos y sociales, se extendió al año siguiente a toda la fábrica.

A través de un proceso de colaboración, trabajo conjunto y al evaluar sistemáticamente las experiencias observadas y del trabajo desarrollado, se ha logrado incorporar gradualmente esta herramienta como método de trabajo destinado fundamentalmente al fortalecimiento de los métodos y estilos de dirección, la participación de los trabajadores en la planeación y solución de sus problemas, la evaluación del desempeño de los colectivos de trabajo, y el pago por los resultados del trabajo. (Centro Interamericano para el Desarrollo del Conocimiento en la Formación Profesional 2010)

TEAMSOFT: Este software fue una propuesta llevada a cabo en el Instituto Superior Politécnico “José Antonio Echeverría” (CUJAE) para dar solución a la falta de una herramienta que sirviera de apoyo a la asignación del personal a los equipos de desarrollo de software, el mismo que posee una interfaz flexible y amigable. El objetivo de este sistema es controlar el desempeño de los miembros de un equipo en la elaboración de un software, al aplicar las prácticas de *Personal Software Process* (PSP) y *Team Software Process* (TSP). Para ello a partir de la definición de un proyecto, los roles, las tareas, las actividades, las competencias necesarias y las métricas se puede asignar el personal adecuado y crear las bases para evaluar el desempeño individual y colectivo en el desarrollo del proyecto. El sistema cuenta con dos módulos; uno para el control del desempeño de los equipos que aplica prácticas de TSP y otro para el control del desempeño individual de cada miembro que constituye una ampliación del *Personal Process Dashboard* (herramienta que implementa las prácticas de PSP). (Margarita André Ampuero y María Gulnara Baldoquín de la Peña 2010)

1.2.4. Sistemas informáticos para la gestión por competencias en la Universidad de las Ciencias Informáticas.

En la universidad se han realizado diversos estudios y algunos sistemas con el objetivo de gestionar los recursos humanos, para conformar el objeto de estudio del presente trabajo con aquellas aplicaciones que automaticen procesos como gestión por competencias y gestión de los recursos humanos.

- Sistema informático para la identificación de las competencias asociadas a los profesiogramas o perfiles de cargo.



- Sistema de Gestión por Competencias (SGC).Módulo de Tratamiento de Datos
- Sistema de Gestión por Competencias (SGC). Módulo de Reportes.
- Software para la Selección del Personal por Gestión de Competencias utilizando Técnicas Matemáticas Multicriteriales.

1.2.5. Desventajas de los Sistemas estudiados.

A partir del estudio realizado no existe a nivel internacional o nacional un sistema que pueda adaptarse totalmente a las necesidades que existen y tomarse como punto de partida para realizar la gestión por competencias del centro CEIGE. Quedando demostrado al exponer las siguientes desventajas:

Software privativo: Algunos sistemas informáticos estudiados son privativos por lo cual se debe incurrir en cuantiosos gastos por concepto de pago de licencias de autorización para el uso de los mismos.

Basado en otros modelos económicos: Algunos sistemas informáticos estudiados en la presente investigación responden a modelos de negocios capitalistas, totalmente diferentes a los que se aplican en el país.

Necesidad de transformar los procesos de negocio donde se aplique: La aplicación de un software con alguna de las características que se mencionan anteriormente en el centro traería consigo la necesidad de modificar todos los paradigmas implantados en el mismo, así como romper con los modelos económicos que se aplican en el país.

1.3. Conclusiones Parciales

A partir del estudio realizado no existe a nivel internacional o nacional un sistema que pueda adaptarse totalmente a las necesidades que existen y tomarse como punto de partida para realizar la gestión por competencias del centro CEIGE. En busca de la solución se estudiaron las características de las principales tecnologías existentes a nivel internacional para el desarrollo de los procesos de software. Se decide utilizar en este trabajo RUP como guía de desarrollo para obtener mejores resultados pues es una metodología completa y bien documentada que se utiliza como una interesante fuente de ideas y herramientas. Se usará el UML como lenguaje de modelado. Para el uso de este se empleará como herramienta el Visual Paradigm por las ventajas que posibilita ser multiplataforma, además es una



herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software, también permite dibujar todos los tipos de diagramas de clases y generar documentación con calidad. Para modelar prototipos se decide utilizar la herramienta Axure RP Pro, ya que muestra a los prototipos de una forma dinámica, permite al usuario poder interactuar con formularios y de esta manera tener una idea más clara de cómo será la aplicación en un futuro. Como técnicas para la captura y validación de los requerimientos de software y patrones de casos de uso se empleará en la elicitación de requisitos por la naturaleza del proyecto a realizar, la combinación de las técnicas: estudio de la documentación, entrevistas y JAD. Para la especificación de requisitos se decidieron utilizar las técnicas: glosario de términos y plantillas porque resultan las actividades más prácticas, frecuentes y a la vez sumamente demostrativas. Para la validación de los requisitos en esta investigación se seleccionaron las revisiones, los prototipos y métricas de software como las técnicas a utilizar.



CAPÍTULO 2: CARACTERIZACIÓN DEL SISTEMA

2.1. Introducción.

En el capítulo se abordan las características que el sistema debe tener, se describe el problema, se hace un análisis del proceso que será objeto de automatización, la información que se maneja, se presenta la propuesta del sistema, se detallan los requisitos funcionales y no funcionales que debe tener el mismo, así como la realización del modelo conceptual. Se definirán además los actores y casos de uso del sistema, las relaciones que existen entre ambos representados en el diagrama de casos de uso del sistema y finalmente se describirán los casos de uso.

2.2. Problema

El CEIGE es uno de los centros de gestión y desarrollo de software más grandes de nuestra universidad. Desde sus inicios las deficiencias existentes en el proceso de gestión por competencias de los roles de los profesionales del mismo ha dificultado la superación profesional de estos y su preparación para acometer las actividades de los proyectos en los que participan. Todas estas deficiencias descritas anteriormente traen consigo la insatisfacción de los propios trabajadores haciéndose estas notorias y fáciles de solucionar con una correcta gestión y puesta en marcha de un proceso que ayude a socavar dichos inconvenientes.

2.3. Objeto de Automatización

El proceso que será objeto de automatización es Gestión por competencias del subsistema Capital Humano del centro CEIGE.

2.4. Información que se maneja

La información manejada está enmarcada en: La gestión por competencias del conocimiento de los trabajadores del centro, datos relacionados con las competencias y habilidades de cada trabajador, que competencias posee, cuáles necesita para poder desempeñarse en un determinado cargo o departamento, cómo matricularse en un curso y cuantos cursos tiene.



2.5. Modelo de dominio

Un actor del negocio es cualquier individuo, grupo, organización o máquina que interactúa con el negocio. Pueden ser: Clientes o Potenciales Clientes, Socios, Proveedores, Autoridades, Propietarios, Sistemas de Información externos al negocio, Otras partes de la organización, si la organización es grande. El actor del negocio no es más que algo o alguien que interactúa con el negocio al desempeñar un rol determinado en el mismo. Si consideramos que el negocio que se estudia tiene muy bajo nivel de estructuración, es decir que no presenta un proceso bien definido **Ver Anexo 1**.

Se decidió utilizar un modelo del dominio porque permite mostrar los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema. Esto ayuda a los usuarios, clientes y desarrolladores e interesados a utilizar un vocabulario común para poder entender el contexto en que se ubica el sistema. Para capturar correctamente los requisitos y poder construir un sistema correcto se necesita tener un firme conocimiento del funcionamiento del objeto de estudio. Este modelo va a contribuir posteriormente a identificar algunas clases que se utilizarán en el sistema.

2.6. Propuesta de Sistema

Con el fin de cumplir con los objetivos y requerimientos planteados en este trabajo, el sistema a proponer debe estar compuesto de 3 módulos, un módulo administrativo donde se realizarán las actualizaciones de todo lo referente a los cursos, matrícula de los mismos y reportes que la aplicación debe brindar, otro módulo que será el punto de acceso a toda la información para los trabajadores, este debe ser capaz de permitir a los mismos los procesos de matrícula de un curso y petición de un nuevo curso. Un tercer módulo que debe ser capaz de permitir al jefe de departamento, realizar las matrículas a los cursos de sus trabajadores. Todos los trabajadores del centro tendrán acceso a dicha aplicación, los mismos podrán interactuar con las funcionalidades que brinde según su nivel de privilegios. Mientras que la administración de los contenidos se realizará solamente por el administrador o alguna persona autorizada por este que será el encargado de mantener el sistema disponible y actualizado.



2.7. Modelo del dominio

El modelo de dominio es la vista estática del dominio de la aplicación, así como los conceptos internos inventados como parte de la implementación de la aplicación. Esta visión es estática porque no describe el comportamiento del sistema dependiente del tiempo, que se describe en otras vistas.

2.7.1. Descripción del Diagrama de Clases del Dominio

La universidad de las ciencias informáticas (UCI) es la institución encargada de la formación de ingenieros informáticos en el país. La misma está estructurada en polos productivos que a su vez poseen diferentes centros o proyectos de desarrollo e investigación. Uno de estos centros es el CEIGE o Centro de informatización de gestión de entidades, dicho centro trabaja en la solución integral de un sistema de gestión por competencias del conocimiento para sus trabajadores. El mismo posee un asesor de posgrado que se encarga entre otras cosas de definir las competencias por roles, además está encargado de gestionar todo lo referente a los curso, matrículas, capacitación, la realización de encuestas para conocer el estado de opinión y el portafolio individual de competencias. Dichas competencias tienen diferentes clasificaciones que ayudan a dar una valoración psicológica y de idoneidad de un trabajador para un puesto. También dichas competencias poseen toda la información cognitiva referente a un trabajador y su desempeño laboral dentro del centro que es utilizada como referencia por el asesor de posgrado a la hora de tomar decisiones sobre la supuesta superación o no de un trabajador. Además dicha información es proporcionada por el CEIGE a otras instancias en caso de ser necesario. El jefe de departamento juega otro papel fundamental pues se encarga de aprobar todo el proceso de matrícula que desarrollan sus trabajadores, o sea los que pertenecen a su departamento. Los trabajadores del centro son los principales protagonistas por así decirlo de dicho modelo pues estos están repartidos por departamentos y a cada uno de ellos se le asigna un rol dentro del mismo según la acumulación de competencias que posee e idoneidad para el puesto. Los trabajadores son los que realizan matrículas, encuestas, poseen un portafolio individual de competencias y cursan diferentes asignaturas para su superación personal a través de un plan de capacitación definido.



2.7.2. Diagrama de Clases del Dominio

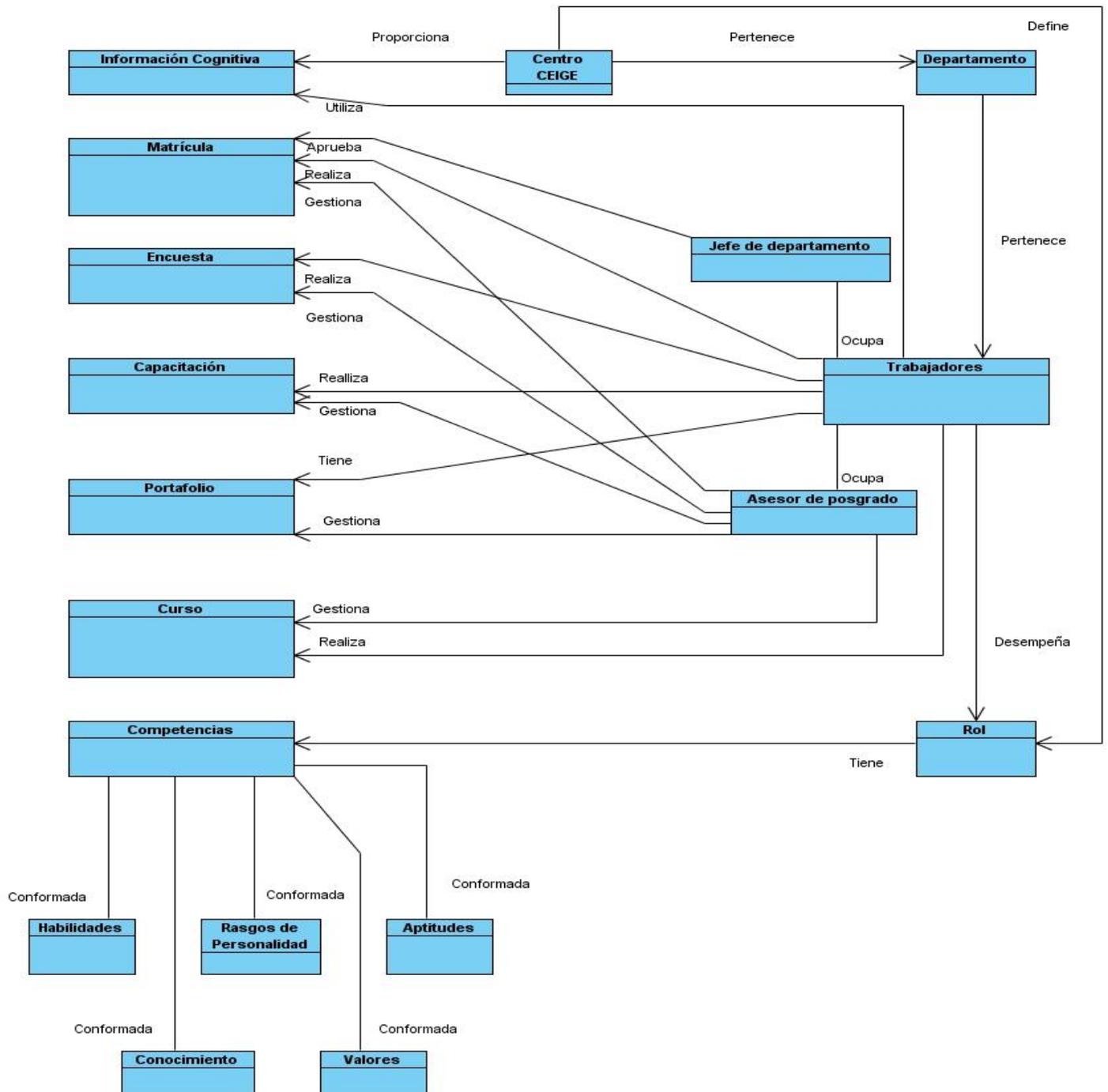


Figura 3. Diagrama de Clases del Dominio



2.7.3. Descripción de las entidades del dominio.

A continuación se presentan una serie de los principales conceptos que se tratan en el modelo de dominio.

CEIGE: Centro de informatización de gestión de entidades.

Trabajador: Persona que trabaja en el centro y ocupa un rol determinado dentro de su departamento según las competencias que posee.

Rol: Papel de un trabajador dentro de una organización, función que desempeña alguien dentro de un empresa u organización determinada.

Información Cognitiva: Información que se almacena y que puede ser consultada después para la toma de decisiones.

Competencias: Descripción de la conducta personal, calificación de rendimiento específico de un individuo o de una organización, se puede utilizar para clasificar a los empleados de conformidad con los requisitos del modelo de competencias establecido, estado que califica los cambios de comportamiento de un individuo, califica el propósito de lograr un alto rendimiento y por lo tanto permite a las empresas lograr un alto rendimiento, y en fin de obtener una ventaja competitiva sostenible.

Matrícula: Lista o catálogo de los nombres de las personas que se inscriben para un fin determinado por las leyes o reglamentos.

Curso: En un centro de enseñanza, tiempo señalado en cada año para asistir a oír las lecciones.

Encuesta: Conjunto de preguntas tipificadas dirigidas a una muestra representativa, para averiguar estados de opinión o diversas cuestiones de hecho.

Portafolio: Lugar virtual donde se guardaran todos las competencias o artefactos que genere un trabajador.



2.8. Especificación de los Requisitos del Software

El flujo de trabajo de requerimientos es uno de los más importantes. En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se especifiquen. Se dividen en dos grupos: los requisitos funcionales y los requisitos no funcionales.

2.8.1. Requerimientos Funcionales

Una vez descrito el modelo de dominio, para poder identificar que debe hacer el sistema y entender su funcionamiento, es fundamental conocer los requisitos funcionales que el sistema debe cumplir. Los requerimientos funcionales deben comprenderlo tanto los desarrolladores como los usuarios, a continuación se relacionan los que debe cumplir la aplicación a desarrollar:

RF 1. Realizar Solicitud de matrícula

El sistema debe mostrar una lista de cursos disponibles a matricular, además debe mostrar una lista de cursos seleccionados a matricular, debe mostrar los datos del curso deseado, también mostrará los datos del curso permitiendo al usuario conocer más sobre el curso seleccionado:

- Datos del curso (nombre, tipo, horas clases, responsable).
- Bibliografía del curso.

RF 2. Realizar petición de nuevo curso

El sistema debe mostrar un listado de cursos disponibles a matricular, además permitir entrar los datos del curso que se desea crear.

- Datos del curso (nombre, tipo, horas clases, responsable, breve descripción).

RF 3. Chequear estado de matrícula

El sistema debe ser capaz de mostrar lista de matrículas realizadas, además muestra un combo box con la lista de cursos previamente pedidos a matricular y debe permitir también seleccionar el curso para ver el estado de la matrícula.



RF 4. Consultar portafolio.

El sistema debe mostrar la lista de todas las competencias, en este caso debe permitir consultar las mismas.

RF 5. Realizar encuesta.

El sistema debe mostrar la lista de todas las encuestas disponibles, además permitir seleccionar la encuesta que se desee realizar.

RF 6. Aprobar solicitudes de matrícula.

El sistema debe mostrar el listado de trabajadores solicitantes con sus datos personales.

- Datos personales (nombre, 2 apellidos).

Además mostrará la lista de cursos a matricular del trabajador seleccionado y permite consultar los datos del curso seleccionado dentro de la lista.

- Datos del curso (tipo, horas clases, responsable).

RF 7. Gestionar curso.

El sistema debe ser capaz de Adicionar curso

Mostrará un conjunto de datos a llenar para crear el nuevo curso.

- Datos del nuevo curso (nombre, tipo, horas clases, responsable).
- Permitir adjuntar bibliografía sobre el curso a crear.

El sistema debe modificar curso.

Mostrará la lista de cursos existentes, además de un conjunto de datos a modificar del curso seleccionado.

- Datos del curso a modificar (tipo, horas clases, responsable).



- Permitirá modificar bibliografía sobre el curso seleccionado.

El sistema debe eliminar curso.

Muestra listado con todos los cursos existentes y debe permitir eliminar un curso.

RF 8. Gestionar Encuesta.

El sistema debe adicionar encuesta.

Mostrará un conjunto de datos a llenar para crear la encuesta.

- Datos de la nueva encuesta (nombre, No preguntas, Dpto. dirigidos).
- Debe permitir realizar la encuesta.

El sistema debe permitir modificar encuesta.

Muestra un conjunto de datos a llenar para crear la encuesta.

- Datos de la nueva encuesta (nombre, No preguntas, Dpto. dirigidos).
- Debe permitir modificar la encuesta.

El sistema debe eliminar una encuesta

Debe mostrar listado con todas las encuestas existentes y eliminar encuesta seleccionada.

RF 9: Gestionar Roles por departamento

El sistema debe Adicionar rol

Mostrará el listado con departamentos del centro, además debe mostrar:

- Datos de los departamentos (nombre).

Mostrará además el listado de roles por departamento y los datos asociados

- Datos del rol (nombre).



Debe permitir introducir nuevos roles al departamento.

El sistema debe Eliminar rol

Debe mostrar el listado con departamentos del centro, además de los datos del departamento. (Nombre), además mostrará el listado de roles por departamento.

- Datos del rol (nombre).

RF10: Gestionar Competencias por Roles

El sistema debe Adicionar competencia

El sistema mostrará el listado con departamentos del centro.

- Datos de los departamentos (nombre).

Mostrará además el listado con roles por departamentos.

- Datos de rol (nombre).

El sistema debe mostrar un listado de competencias por roles.

- Datos de competencias (nombre).

Debe permitir introducir nuevas competencias por rol y crear nuevas competencia.

El sistema debe Eliminar competencias

Mostrará el listado con departamentos del centro.

- Datos del departamento (nombre).

Mostrará además el listado de roles por departamento.

- Datos del rol (nombre).

También debe mostrar listado de competencias por rol



- Datos de competencias (nombre).

Finalmente debe eliminar una competencia.

RF 11. Gestionar Matrícula

El sistema debe Adicionar grupo

Mostrará el listado de cursos existentes.

- Datos de curso (nombre).

Mostrará además el listado de trabajadores por curso seleccionado

- Datos de listado de trabajadores (nombre).

Debe permitir crear grupos para curso, mover trabajadores existentes en el listado de trabajadores por curso hacia un nuevo listado denominado (crear un grupo de matriculados). Además mostrará un listado nuevo con un número de trabajadores finito que será el grupo de estudio creado. Permitirá también seleccionar datos específicos para cada curso.

- Datos a seleccionar (# grupo, capacidad del grupo, profesor).

El sistema debe Modificar grupo

Mostrará el listado de cursos existentes.

- Datos de curso (nombre).

Mostrará el listado de grupos existentes por curso.

- Datos de curso (# grupo, capacidad del grupo, profesor).

Debe seleccionar un grupo específico del listado de grupos, además debe visualizar el listado de trabajadores del grupo seleccionado.

- Mostrar Datos del grupo seleccionado (# grupo, capacidad del grupo, profesor).



Permitirá quitar uno o varios miembros del listado de trabajadores del grupo seleccionado y modificar datos del grupo seleccionado (# grupo, capacidad del grupo, profesor).

El sistema debe Eliminar grupo

Mostrará el listado de cursos existentes.

- Datos de curso (nombre).

Mostrará además el listado de grupos existentes por curso.

- Datos de curso (# grupo, capacidad del grupo, profesor).

Debe permitir seleccionar un grupo específico del listado de grupos.

RF 12. Consultar reporte de cursos planificados

El sistema debe mostrar el listado de cursos planificados en el centro.

- Datos del listado de cursos (nombre, tipo, horas clase, responsable).

RF 13. Consultar reporte de matrícula de trabajadores por cursos planificados

El sistema debe mostrar el listado de trabajadores según el curso deseado, grupo escogido.

- Datos del listado (nombre, 2 apellidos).

RF 14. Consultar reporte de plan de capacitación por trabajador.

El sistema debe mostrar el listado de cursos matriculados por un trabajador

- Datos del listado (nombre, tipo, horas clases, responsable).

Además el listado de grupos asignados para un trabajador.

- Datos del listado (# grupo, nombre del curso a que pertenece).

RF 15. Consultar reporte de cursos matriculados



El sistema debe mostrar el listado de trabajadores por cursos matriculados.

- Datos del listado (nombre, dos apellidos, curso matriculado).

RF 16. Consultar reporte plan de capacitación

El sistema debe ser capaz de mostrar el plan de capacitación por cursos matriculados.

- Datos del plan de capacitación (nombre del curso, fecha inicio, fecha fin, grupos por cursos, responsable o profesor del curso, local del curso).

RF 17. Realizar plan de capacitación

El sistema debe mostrar todos los cursos disponibles y permitir seleccionar el deseado, además una lista de grupos asociados al curso seleccionado, de los cuales también debe permitir seleccionarlos, También es necesario que el sistema muestre una vez seleccionado un grupo una serie de datos a llenar.

- Datos (fecha inicio, fecha fin, responsable, local).

RF 18. Gestionar departamentos del centro

El sistema debe Adicionar departamentos

Mostrará el listado de departamentos existentes en el centro.

- Datos de los departamentos (nombre).

Debe permitir adicionar nuevos departamentos a la lista, insertando una serie de datos sobre los mismos. (Nombre, # trabajadores, # del laboratorio, # PC).

El sistema debe Modificar departamentos.

Mostrará el listado de departamentos existentes en el centro.

- Datos de los departamentos (nombre).



Debe permitir seleccionar el departamento deseado y modificar los datos que se tienen almacenados sobre un curso.

- Datos guardados (Nombre, # trabajadores, # del laboratorio, # PC).

El sistema debe Eliminar departamento

Mostrará el listado de cursos existentes.

- Datos de curso (nombre).

Debe permitir seleccionar un departamento de los existentes en el listado de departamentos y eliminarlo.

2.8.2. Requerimientos No Funcionales

Una vez analizado las funcionalidades que el sistema debe cumplir se hace necesario analizar las propiedades que el producto software debe tener, de lo cual se deriva: las restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, facilidad de mantenimiento, entre otras. A esto se le conoce como requisitos no funcionales. Entre los requerimientos no funcionales del sistema propuesto se encuentran:

Requisitos de interfaz o apariencia

El sistema debe tener una interfaz con un diseño agradable y sencillo, lo que permite que sea amigable para los usuarios. La interfaz debe limitarse a presentar funcionalidades netamente necesarias, lo que ayuda a que sea de mayor interés para los usuarios de la comunidad.

Requisitos de usabilidad

Visibilidad del estado del sistema. La aplicación debe mantener siempre informado al usuario del estado del sistema, así como los caminos que este pueda tomar con una retroalimentación visual apropiada en un tiempo razonable.

Requisitos de rendimiento



Tiempo de respuesta: Las consultas a la base de datos no serán de un nivel de complejidad alto y los niveles de concurrencia de usuarios que accedan al servidor de aplicaciones no serán muy elevados, el tiempo de respuesta de la aplicación será rápido, es decir no debe de exceder los 3 segundos.

Requisitos de soporte

El sistema debe permitir ser usado desde cualquier Sistema Operativo.

Requisitos de seguridad El sistema deberá garantizar que las funcionalidades se muestren de acuerdo al nivel de usuario que esté activo. El sistema deberá estar protegido contra acciones que no estén autorizadas o que puedan afectar la integridad de los datos. El sistema deberá verificar acciones irreversibles (eliminaciones).

Requisitos de software

Para PC Servidor

Como Gestor de Base de Datos PostgreSQL 8.2.4.1. El lenguaje de programación a utilizar será Framework Symfony instalado. Se utilizara como Web Apache 2.2.6.

Para PC Cliente

Deberá tener instalado un Navegador Web (Moxila Firefox 4.0).

Requisitos de hardware

Para PC Servidor

El sistema necesita un servidor de base de datos en una PC o servidor con 512 MB de memoria RAM o superior y un disco duro con una capacidad de 120 GB o superior respectivamente, al tener en cuenta el tiempo de permanencia de la información en dicho servidor.

Para PC Cliente

El sistema necesita una PC cliente con 256 megabytes de memoria RAM o superior.



2.9. Definición de los Casos de Uso

2.9.1 Actores del sistema

| ACTOR | DESCRIPCION |
|-------------------------------|--|
| Usuario | Es el encargado de interactuar con la aplicación, el mismo solicita matrícula en cursos, realiza petición de nuevos cursos, chequea el estado de sus competencias, y puede consultar los reportes que brinda la información. |
| Jefe de departamento | Es el encargado de gestionar todo el proceso de matrícula de los cursos por departamento. |
| Asesor de posgrado del centro | Es el encargado de gestionar toda la información del sistema. |

Tabla 1. Definición de los actores del sistema.

2.9.2. Diagrama de casos de uso del sistema.

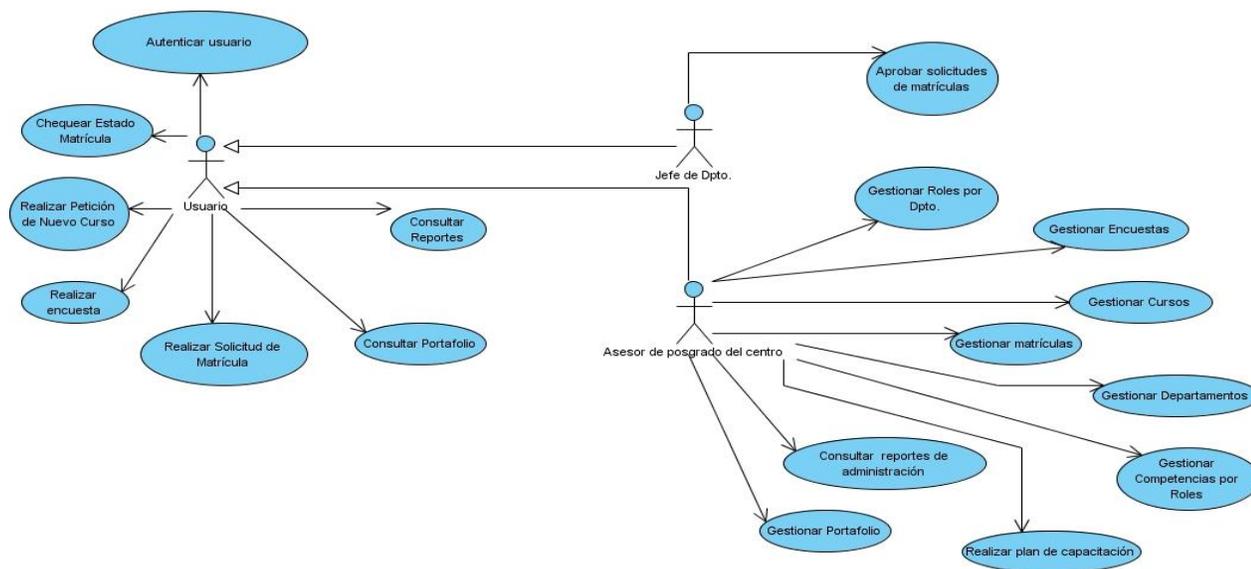


Figura 4. Diagrama de casos de uso del sistema.



2.9.3. Descripción de los casos de uso del sistema.

A continuación se muestran las descripciones detalladas de algunos de los casos de uso. Ver otros en (Anexos, Descripción de los casos de uso del sistema)

| | | |
|--|--|--|
| CU-1: | Realizar Solicitud de Matrícula | |
| Actor: | Usuario | |
| Resumen: | El CUS se inicia cuando el trabajador interactúa con la aplicación para realizar una petición de matrícula a un curso dado. El CUS finaliza cuando se ejecuta dicha acción. | |
| Referencia: | RF1.1, RF1.2, RF1.3, RF1.4 | |
| Precondición: | El usuario debe estar previamente autenticado. | |
| Poscondición: | | |
| Flujo Normal de eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El usuario accede a la aplicación y selecciona el módulo realizar solicitud de matrícula. | 2. El sistema muestra la funcionalidad realizar matrícula. | |
| 3. El usuario escoge la opción mostrada. | 4. El sistema muestra la siguiente información. <ul style="list-style-type: none"> - Listado de cursos disponibles a matricular. - Muestra información no modificable sobre el curso seleccionado. (nombre del curso, tipo, horas clases, responsable y bibliografía). | |



| | |
|--|--|
| <p>5. El usuario escoge uno o varios cursos en los que desea matricular.</p> <p>6. El usuario confirma la operación realizada.</p> | <p>7. Valida que los datos introducidos son correctos, que los campos obligatorios no están vacíos y guarda los datos.</p> <p>8. Informa al usuario de la(s) nuevas matrículas realizadas.</p> <p>9. Termina el caso de uso.</p> |
|--|--|

Prototipo de Interfaz



Flujo alternativo de eventos

| Acción del Actor | Respuesta del Sistema |
|------------------|--|
| | <p>10. El sistema debe comprobar la acción de validación y en caso negativo mostrar un mensaje de error: "Existen campos</p> |



| | |
|--|---|
| | vacios o datos incorrectos” y el sistema va al flujo 4. |
| Prototipo de Interfaz | |
|  | |

Tabla 2. CU Realizar Solicitud de matrícula

| | | |
|---|---|--|
| CU-2: | Realizar petición de nuevo curso | |
| Actor: | Usuario | |
| Resumen: | El CUS se inicia cuando el usuario interactúa con la aplicación para realizar una petición de creación de un nuevo curso. El CUS finaliza cuando se ejecuta dicha acción. | |
| Referencia: | RF2.1, RF.2.2, RF2.3, RF2.4 | |
| Precondición: | El usuario debe estar previamente autenticado. | |
| Poscondición: | | |
| Flujo Normal de eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El trabajador accede a la aplicación y selecciona el módulo realizar solicitud | 2. El sistema muestra la funcionalidad realizar matrícula. | |



| | |
|---|--|
| de matrícula. | |
| 3. El trabajador escoge la opción mostrada. | 4. El sistema muestra la siguiente información. <ul style="list-style-type: none">- Listado de nuevos cursos creados.- El sistema solicita los siguientes datos:<ul style="list-style-type: none">• Nombre del curso (Obligatorio).• Tipo de curso (Obligatorio), deberá escoger una opción de las brindadas.• Breve descripción, deberá explicar brevemente la necesidad de creación de dicho curso. |
| 5. El usuario introduce los datos. 6. Confirma la operación (guardar). | 7. Valida que los datos introducidos son correctos, que los campos obligatorios no están vacíos y guarda los datos. 8. Informa al usuario del nuevo curso creado. 9. Termina el caso de uso (aceptar). |
| Prototipo de Interfaz | |



Petición de nuevo curso

Cargo de Persona: Jefe de Depto.
 Nombre: Carlos Abel Gonzalez
 Estructura: CEIGE

Lista de nuevos cursos

Gestión empresarial:
 Administración de empresa

Datos nuevo curso

Nombre del curso:
 Tipo de curso:
 Breve Descripción:

Flujo alterno de eventos

| Acción del Actor | Respuesta del Sistema |
|------------------|--|
| | 10. El sistema debe comprobar la acción de validación y en caso negativo mostrar un mensaje de error: "Existen campos vacios o datos incorrectos" y regresa al flujo básico 4. |

Prototipo de Interfaz



Flujo alterno de eventos

| Acción del Actor | Respuesta del Sistema |
|--|---|
| 11. El usuario escoge la opción aceptar. | 12. El sistema muestra un mensaje de información y regresa al flujo básico 4. |

Prototipo de Interfaz

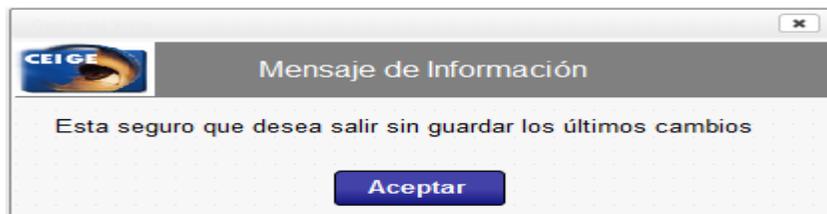


Tabla 3. CU Realizar petición de nuevo curso

| | |
|-----------------|--|
| CU-3: | Chequear estado de matrícula |
| Actor: | Usuario |
| Resumen: | El CUS se inicia cuando el usuario interactúa con la aplicación para chequear el estado de su matrícula. El CUS finaliza cuando se ejecuta dicha acción. |



| | | |
|---|--|--|
| Referencia: | RF 3.1, RF 3.2, RF 3.3. | |
| Precondición: | El usuario debe estar previamente autenticado. | |
| Poscondición: | | |
| Flujo Normal de eventos | | |
| Acción del Actor | Respuesta del Sistema | |
| 1. El trabajador accede a la aplicación y selecciona el módulo chequear estado de matrícula. | 2. El sistema muestra la funcionalidad estado de la matrícula. | |
| 3. El usuario escoge la matrícula deseada de las existentes. 4. Confirma la operación (Ver). | 5. El sistema muestra un mensaje de información con el estado de la matrícula. (aceptada o denegada) | |
| Prototipo de Interfaz | | |
| | | |
| Flujo alterno de eventos | | |
| Acción del Actor | Respuesta del Sistema | |



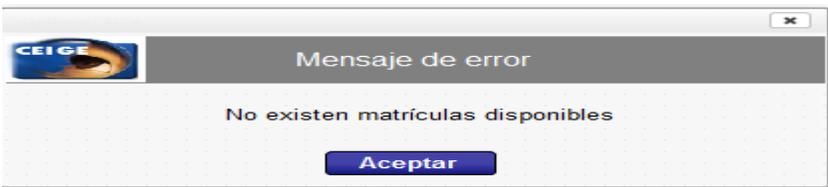
| | |
|--|---|
| 5. El usuario escoge la opción ver sin seleccionar una matrícula específica. | 6. El sistema muestra un mensaje de error. "Debe seleccionar una matrícula" y el sistema retorna al flujo básico 2. |
| Prototipo de Interfaz | |
|  | |
| Flujo alterno de eventos | |
| Acción del Actor | Respuesta del Sistema |
| 7. El usuario escoge la opción ver sin existir matrículas disponibles. | 8. El sistema muestra un mensaje de error. "No existen matrícula disponibles" y el sistema retorna al flujo básico 2. |
| Prototipo de Interfaz | |
|  | |

Tabla 4. CU Chequear estado de matrícula



2.10. Conclusiones Parciales

En este capítulo se describió el sistema, los requisitos funcionales y no funcionales que debía cumplir el mismo, los actores y casos de uso del sistema así como la relación que existe entre ambos representados gráficamente en un diagrama. Se detalló paso a paso cada caso de uso, la acción de los actores y la respuesta del sistema a cada acción. Además se aplicaron técnicas de validación de requisitos lo que garantiza la calidad y valía de los mismos. De esta forma quedan sentadas las bases para comenzar a construir el sistema si se tienen en cuenta los requerimientos especificados en el capítulo.



CAPÍTULO 3: ANÁLISIS DEL SISTEMA.

3.1. Introducción.

En el capítulo se describe el flujo de trabajo Análisis y se muestran los diagramas de clases del análisis. Además se describe el proceso de validación a los artefactos generados.

3.2. Análisis

En este se analizan los requisitos funcionales capturados y descritos anteriormente, de forma que se refinan y estructuran los mismos, con el propósito de lograr una comprensión más precisa de dichos requerimientos, para obtener una visión del sistema completo. A continuación se refleja el diagrama de clases de análisis correspondiente a los casos de usos descritos.(Cuevas Agustín, Gonzalo 2002)

3.2.1. Modelo del análisis

El lenguaje a utilizar se basa en un modelo de objetos conceptual, que se denomina modelo del análisis. El modelo de análisis contribuye a refinar los requisitos, así como a razonar sobre los aspectos internos del sistema. A pesar de que el modelo del análisis hay un refinamiento de los requisitos, no se toman en cuenta aspectos que afectan al sistema como el lenguaje de programación, la plataforma en la que se ejecutará la aplicación, los componentes prefabricados o reusables de otras aplicaciones, el objetivo del análisis es comprender perfectamente los requisitos del software y no precisar cómo se implementará la solución.(Cuevas Agustín, Gonzalo 2002)

3.2.2. Clases del análisis

Las clases del análisis se centran en los requisitos funcionales y son evidentes en el dominio del problema porque representan conceptos y relaciones del dominio. Tienen atributos y entre ellas se establecen relaciones de asociación, agregación / composición, generalización / especialización y tipos asociativos. RUP propone clasificar las clases en:

3.2.3. Diagramas de clases del análisis

Un Diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema. Representa las cosas del mundo real, no de la implementación automatizada de estas cosas.



A continuación los diagramas de clases del análisis:

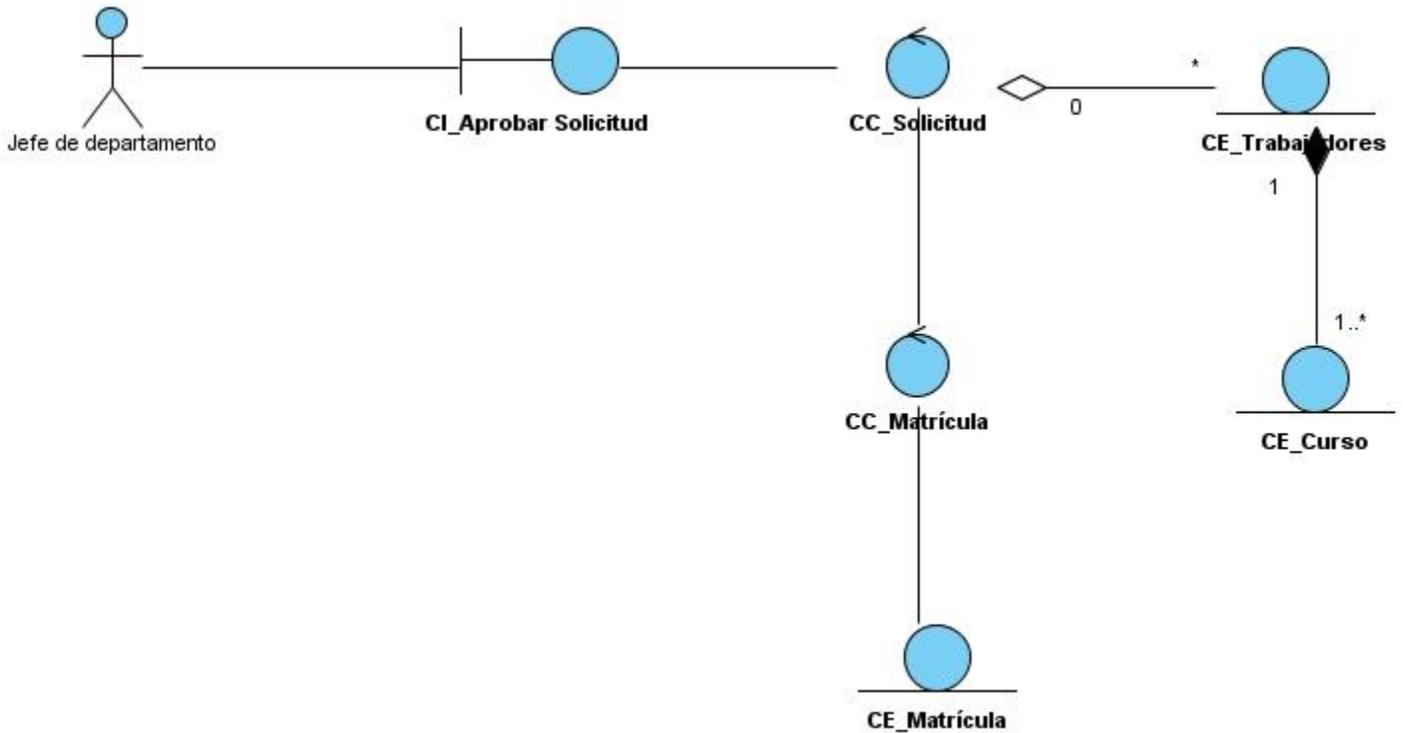


Figura 5. CU- 1 Aprobación Solicitud de matrículas.



Figura 6. CU- 2 Chequear estado de matrícula.

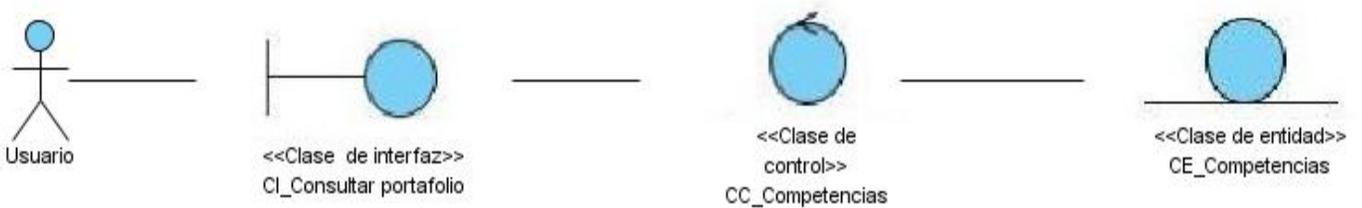


Figura 7. CU- 3 Consultar portafolio.

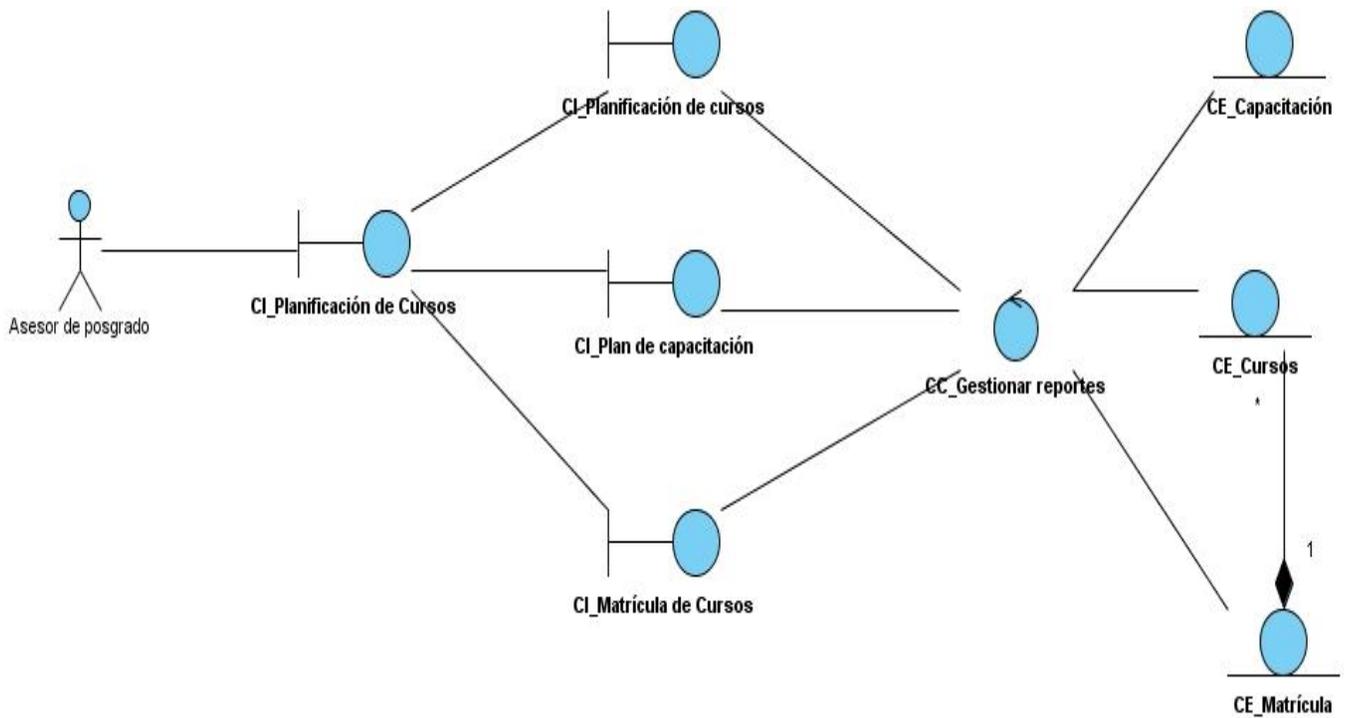


Figura 8. CU- 4 Consultar reporte administración.

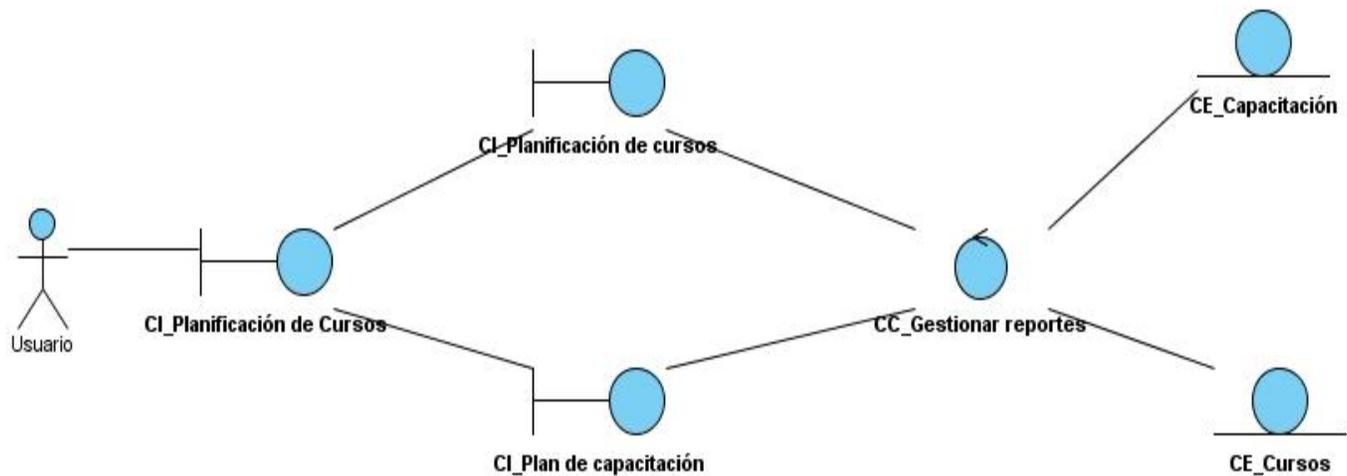


Figura 9. CU- 5 Consultar reporte.

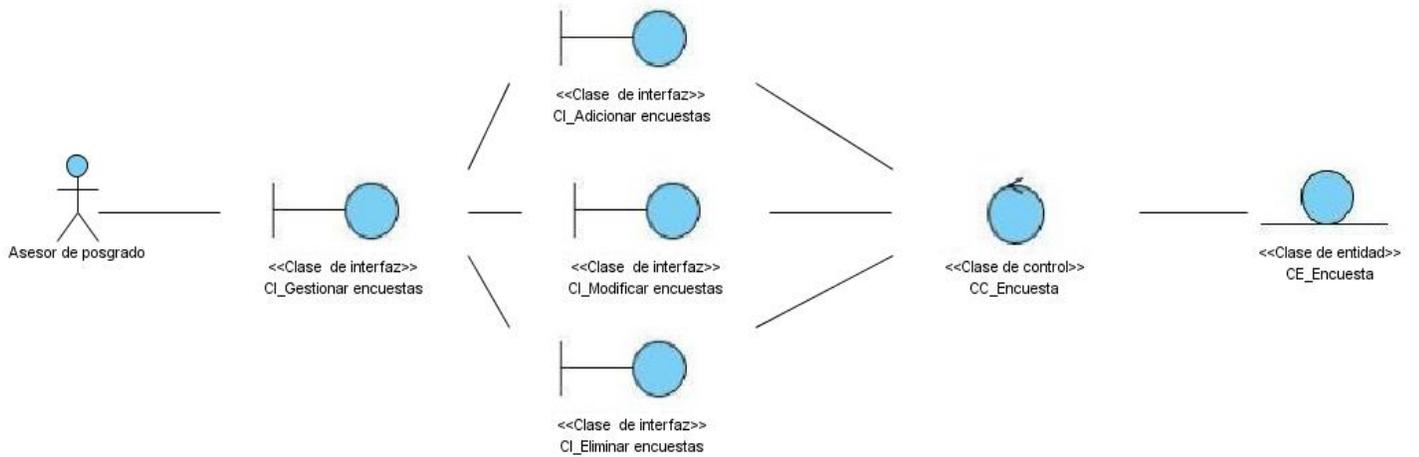


Figura 10. CU- 6 Gestionar Encuesta.

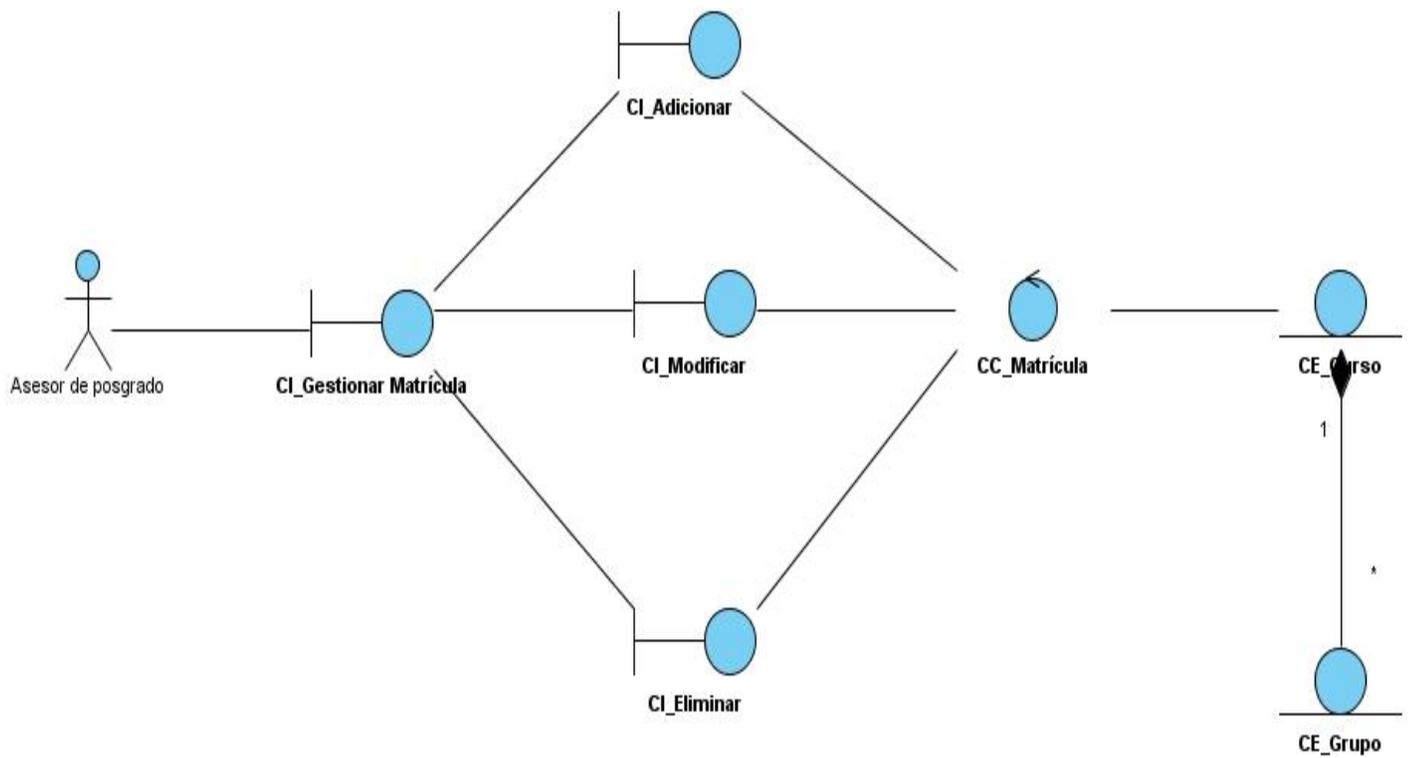


Figura 11. CU- 7 Gestionar matrícula.

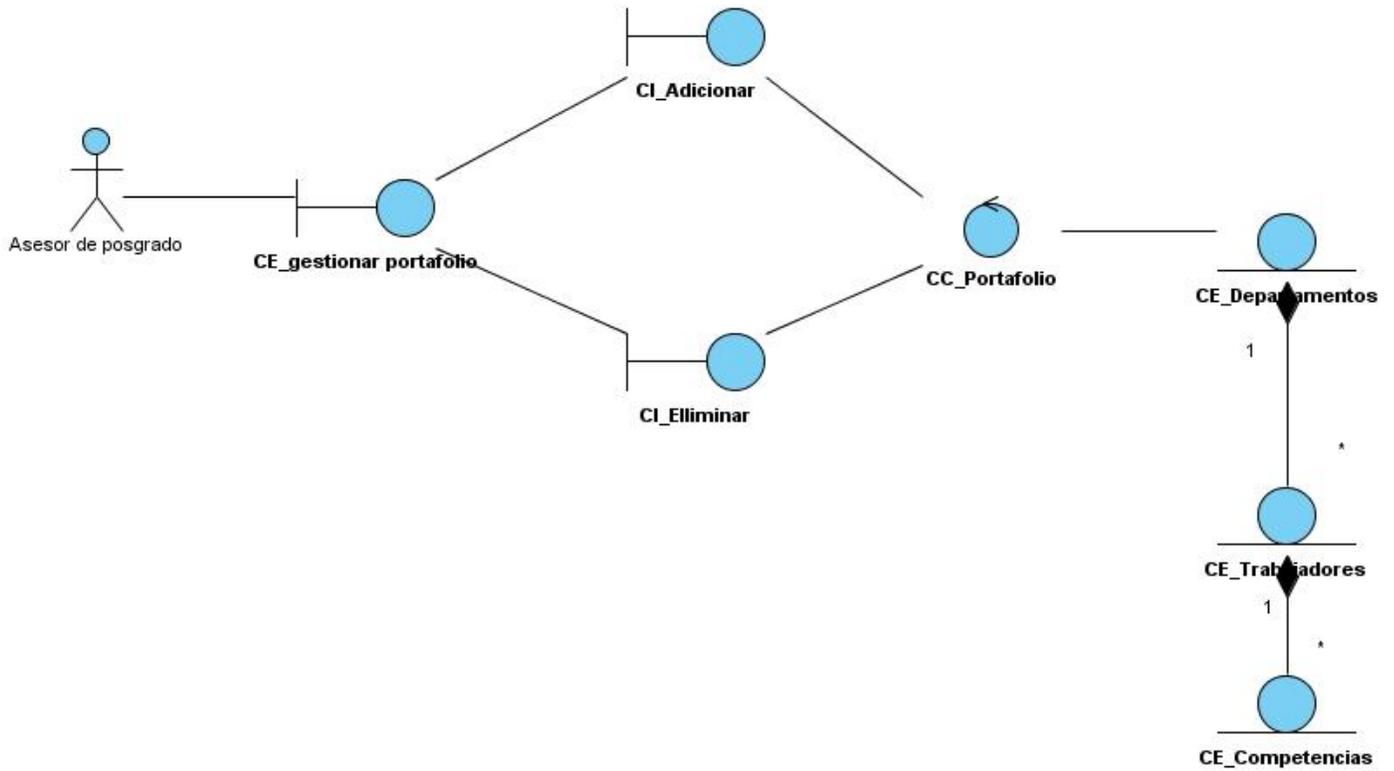


Figura 12. CU- 8 Gestionar Portafolio.

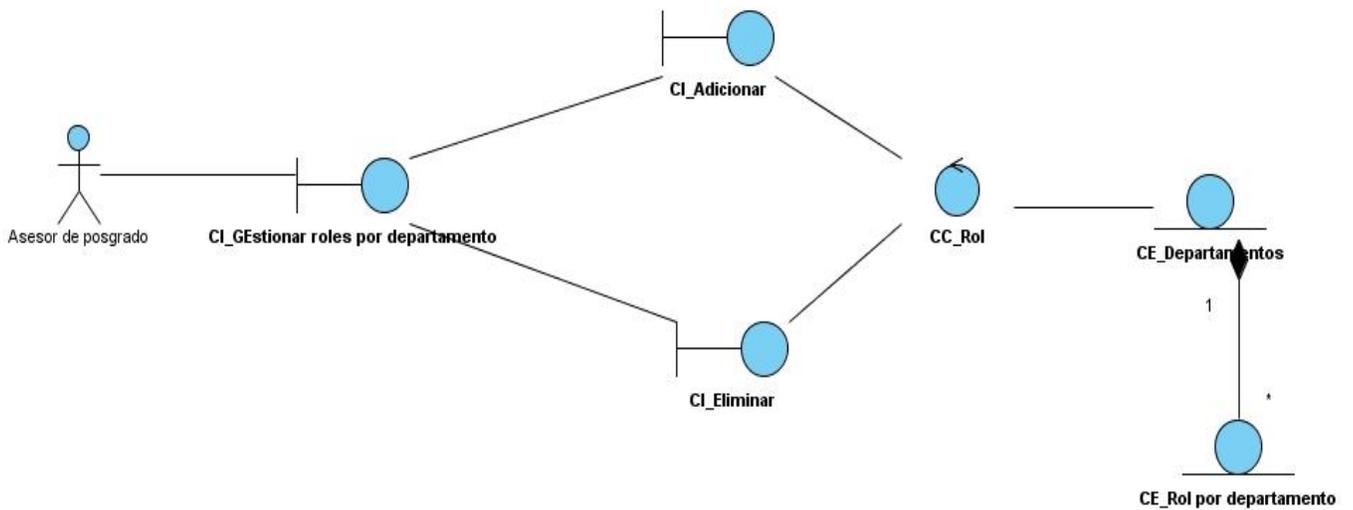


Figura 13. CU- 10 Gestionar Roles por departamento.

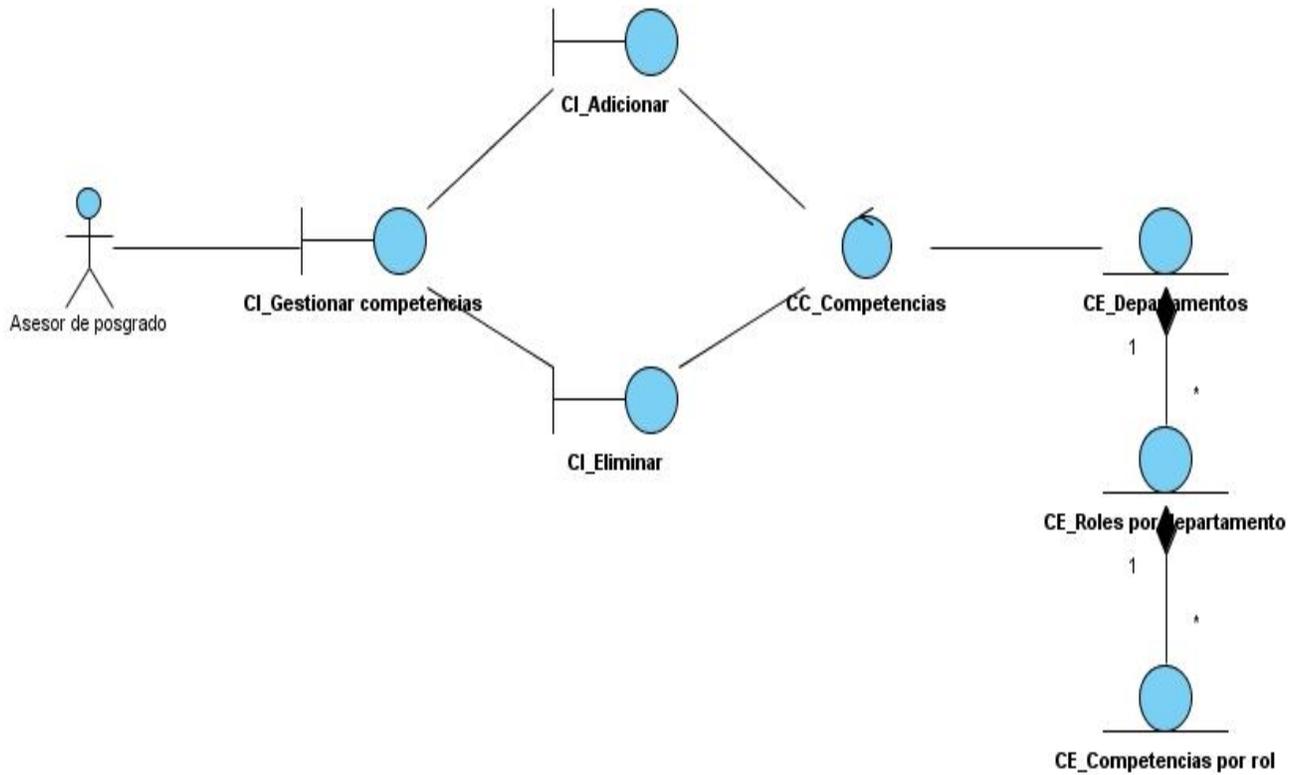


Figura 14. CU- 11 Gestionar competencias por Rol.



Figura 15. CU- 12 Realizar encuestas.

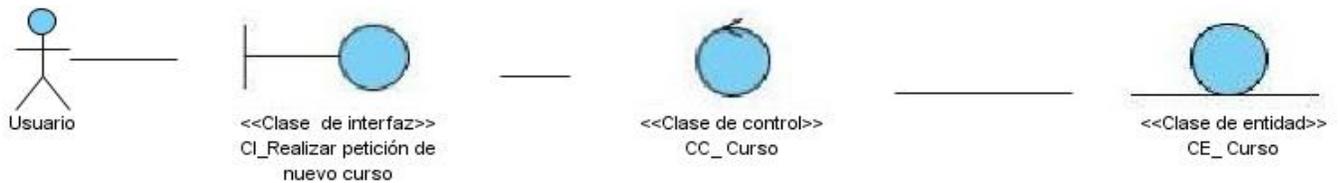


Figura 16. CU- 13 Realizar petición de nuevo curso.

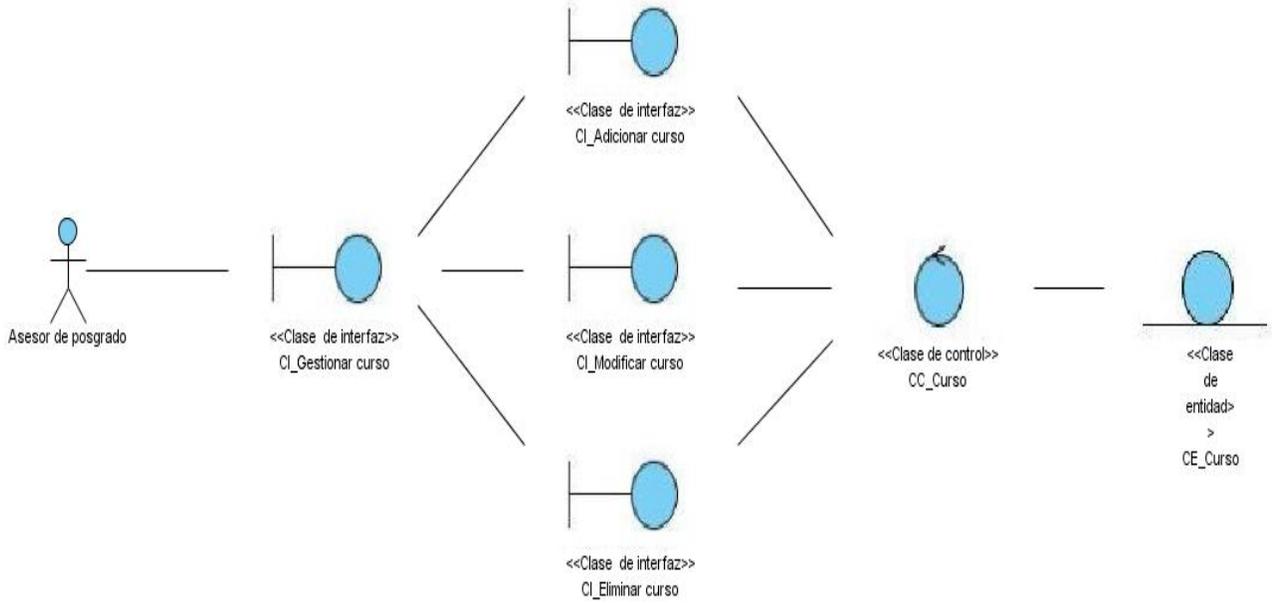


Figura 17. CU - Gestionar curso.

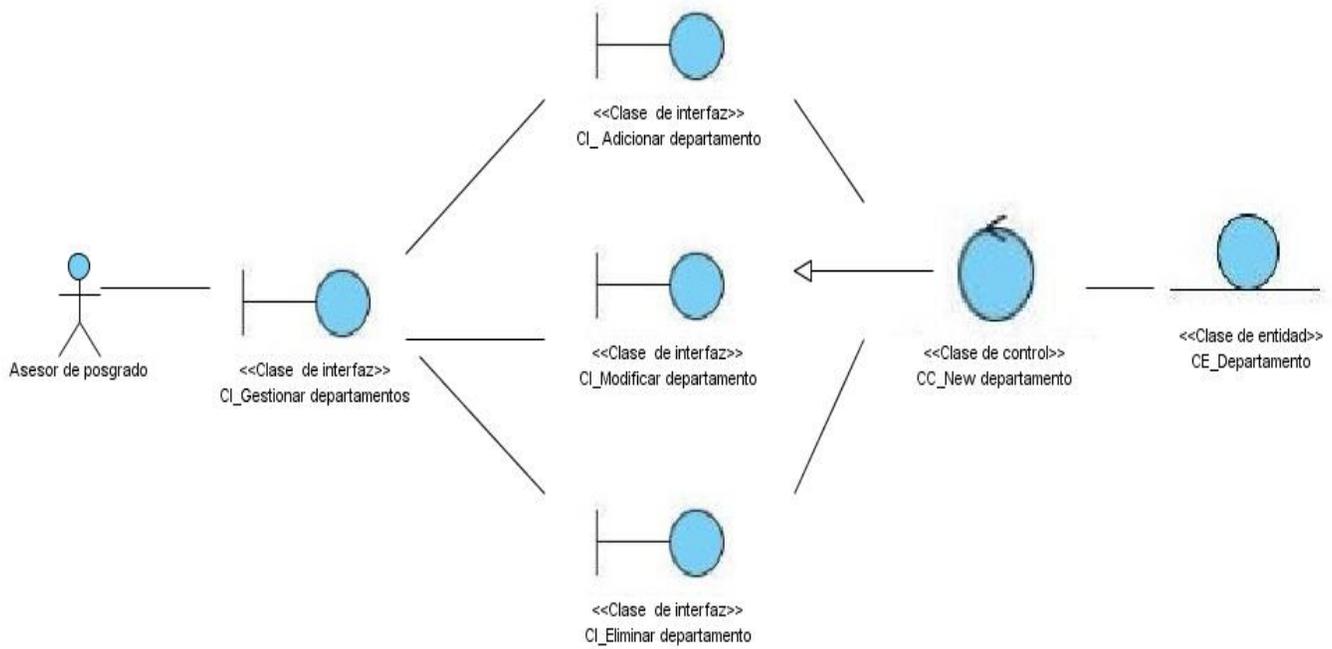


Figura 18. CU- Gestionar departamento.

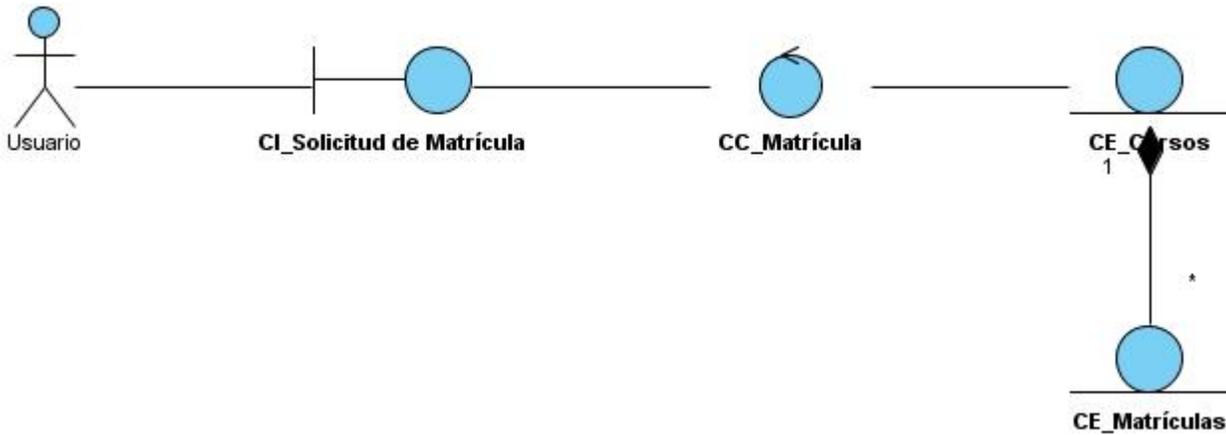


Figura 19. Realizar matrícula.

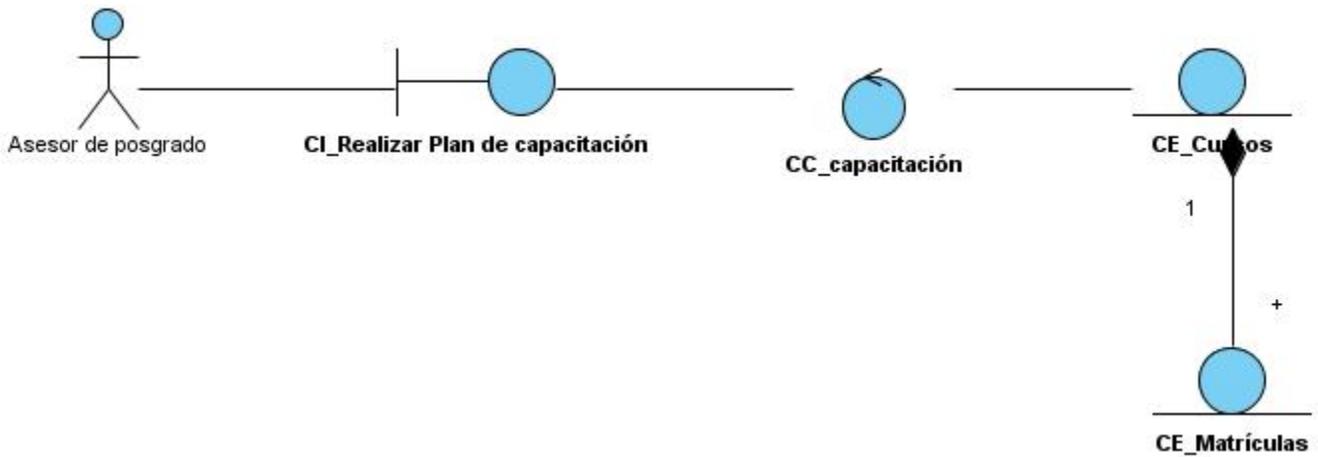


Figura 20. Realizar plan de capacitación.



Figura 21. Autenticar usuario. (Universidad de las Ciencias Informáticas 2011)



3.3. Validación de los requisitos

3.3.1. Métrica de software

Las métricas son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo software y los proyectos de mantenimiento. En general, la medición persigue tres objetivos fundamentales: ayudar a entender qué ocurre durante el desarrollo y el mantenimiento, permitir el control de lo que ocurre en nuestros proyectos y poder mejorar los procesos y productos. En un sistema de software están presentes los procesos que no son más que actividades que normalmente conllevan el factor tiempo (duración del proceso), el esfuerzo (asociado al proceso) y el número de incidentes de un tipo específico que se dan durante el proceso. (Roger S. Pressman 2007)

3.3.2. Métrica para la calidad de especificación de requisitos de software.

Para la medición de la calidad de la especificación de requisitos se tuvieron en cuenta una lista de características que pueden emplearse para valorar la calidad del modelo de análisis y la correspondiente especificación de requisitos: Especificidad, ambigüedad, corrección, compleción, comprensión, capacidad de verificación, consistencia externa e interna, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización. (Roger S. Pressman 2007) Además las especificaciones están empaquetadas por importancia y estabilidad relativas, con su versión correspondiente, organizadas, con referencias cruzadas y especificadas al nivel correcto de detalle. Aunque muchas de las características anteriores pueden ser de naturaleza cuantitativa, se sugiere que todas puedan representarse al usar una o más métricas.

$$NR = RF + RNF$$

Donde NR es el número de requisitos funcionales y RNF es el número de requisitos no funcionales.

Para determinar la especificidad de los requisitos, sugiere una métrica basada en la consistencia de la interpretación de los revisores para cada requisito:

$$Q1 = NRI / NR$$



Donde NRI es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas. Cuanto más cerca de 1 esté el valor de Q1 menor será la ambigüedad de la especificación.

Después de la realización de Diagrama de caso de uso del sistema y al seguir las pautas para que un Diagrama de caso de uso del sistema proceda o sea deseable, descritas en este epígrafe, se comienza a medir la relación de los requisitos según las interpretaciones de los miembros del equipo de inspección lo cual quedó reflejado en la siguiente tabla:

| Especificidad | | | | |
|-------------------|----|------------------|------------|-------|
| Tipo de requisito | | Interpretaciones | | |
| | | Iguales | Desiguales | Total |
| Funcionales | 15 | 14 | 1 | 15 |
| No funcionales | 7 | 7 | 0 | 7 |
| Total | 22 | 21 | 1 | 44 |

Tabla 5. Interpretaciones de requisitos funcionales

$$RF= 15 \quad RNF=7 \quad NR=RF+ RNF= 15 + 7 = 22$$

$$Q1=NRI/ NR =21 /22=0.95$$

El valor resultante de Q1 igual a 0.95, valor muy aproximado a 1 demuestra que el grado de ambigüedad en la especificación de los requisitos de software fue muy bajo y por consiguiente se comprobó calidad en la especificación. Con el margen de error existente, con valor igual a 0.05 se procedió a una revisión de los requisitos que presentaban dificultades. Después de la revisión se volvió a aplicar esta métrica, concluyendo que todos los requisitos tuvieron la misma interpretación y que los requisitos obtenidos se encuentran con un alto grado de especificidad.

3.3.3. Técnica de prototipado

Se conoce como prototipo a la implementación concreta de un sistema que se crea para explorar cuestiones sobre aspectos muy diversos durante el desarrollo de un sistema.(Roger S. Pressman



2007) Ellos permiten la comunicación y participación que puede haber entre el equipo y los clientes, al hacer de la calidad y las especificaciones una herramienta primordial para comprobar los requisitos del software.

Luego de la construcción de la interfaz principal, se hace necesaria la fabricación de otras con pequeñas modificaciones que complementen el producto del software final. (Ing. Mailen Edith Escobar Pompa y Ing. Leydis Andis Ortiz Azahares 2008) a aceptación

Se aplicó dicha técnica obteniéndose la total aceptación del cliente sobre el tema. Se logró concebir un prototipo no funcional de la posible interfaz del software a construir, además se obtuvo un mejor entendimiento del problema y se logró validar los requisitos que el usuario solicitó. Mediante los prototipos se pudo validar los requerimientos funcionales que fueron capturados durante la etapa de Requerimientos.

3.4. Conclusiones parciales

En el capítulo se mostraron los artefactos generados en el flujo de trabajo de análisis, además se incluyó una breve descripción de los mismos, con el objetivo de que se comprendan perfectamente los requisitos del software. Lo que posibilita la correcta transformación de los mismos a un diseño que indique como debe ser implementado el software.



CONCLUSIONES

Una vez concluido el estudio de sistemas para la gestión por competencias, se realizó el análisis del módulo Sistema de Gestión por competencias del subsistema Capital Humano del centro CEIGE, lo cual permitirá su posterior implementación. Se decidió realizar, al tener en cuenta las características del proyecto, las actividades Elicitación, Especificación y Validación de los requisitos. En la Elicitación se utilizaron diferentes técnicas, destacándose la técnica Desarrollo conjunto de aplicaciones o JAD que permitió definir los involucrados, trabajadores, artefactos, reglas y mejoras de los procesos del negocio. Como resultado de la Elicitación se obtuvo una primera versión de los requisitos funcionales del módulo, los cuales fueron refinados y especificados formalmente durante la Especificación de Requisitos. Por último, los requisitos se validaron con los clientes, además de aplicarse la métrica de calidad de especificación de requisitos del software.

Se definieron un total de 16 casos de uso como resultado de la Ingeniería de Requerimientos que representan la propuesta de funcionalidades. En el flujo de trabajo del análisis se obtuvieron los diagramas de clases del análisis necesario para desarrollar un software con mayor robustez.



RECOMENDACIONES

- Dar continuidad a los flujos de trabajo propuestos por RUP a partir de los casos de uso descritos en el presente trabajo, logrando de esta forma una solución informática que sea capaz de resolver los problemas reales para los cuales fue creada.
- Realizar la gestión de requisitos para tener un control sobre los cambios que puedan proponer los especialistas funcionales luego de la validación de los mismos; en función de evitar los riesgos que puedan atentar contra el buen desempeño del software.
- Se recomienda consultar la bibliografía de la investigación, por parte de todas aquellas personas que necesiten ampliar sus conocimientos sobre el tema.



REFERENCIA BIBLIOGRÁFICA

- Centro Interamericano para el Desarrollo del Conocimiento en la Formación Profesional. *SIMAPRO en Cuba: sector azucarero*. Centro Interamericano para el Desarrollo del Conocimiento en la Formación Profesional, 2010 Available from world wide web: <<http://www.cinterfor.org.uy/public/spanish/region/ampro/cinterfor/temas/prod/simapro/aplica/cuba/index.htm>>.
- Cristiane Casaca CSI - Control and Systems and Information Technologies. Recommender Systems for Human Resources TaskAssignment. 8, [portugal], 2010.
- Cuevas Agustín, Gonzalo. *GESTIÓN DEL PROCESO SOFTWARE (1ª)*. 2002 2002.
- Fowler, Martin. *Analysis Patterns*. 2007.
- Frank Armour, y Granville Miller. *Advanced Use Case Modeling*. 2001.
- Gary Stanley Becker. *Human capital*. 1993.
- Giovanni Acampora, Matteo Gaeta, Francesco Orciuoli, y Pierluigi Ritrovato. Exploiting Semantic and Social Technologies forCompetency Management. 5, 2010.
- Gunnar Overgaard, Karin Palmvist, y Addison-Wesley. *Use Cases- Patterns and Blueprints*. 2005.
- Hong-Mei Jin, y Shu-Ming Wang. A Theoretical Research on the Post Competence of University Administrators. 4, 2010.
- <http://www.cezannesw.com/es/>. Cezanne Software. 2011. Available from world wide web: <http://www.equipostrytalento.com/rrhh/software_recursos_humanos/cezanne-software>.
- INFOR company. *Infor HCM (Human Capital Management)*. *Administración del capital humano*. 2011 Available from world wide web: <<http://latinamerica.infor.com/soluciones/hcm/>>.
- Ing. Denise Fernández Miranda, y Ing. Yaniuska García Romero. Análisis y Diseño del módulo Análisis de Riesgo de Crédito del proyecto SAGEB. 2010.
- Ing. Mailen Edith Escobar Pompa, y Ing. Leydis Andis Ortiz Azahares. Análisis y Diseño de un Nodo Virtual de Procesos. 2008.
- ION company. ION:Software deGestión Estratégicade RecursosHumanos. 2010. Available from world wide web: <www.lempert.com.ar>.
- Jacobson Ivar, Rumbaugh James, y and Booch, Grady. *El Proceso Unificado del Desarrollo del Software*. 2000 Available from world wide web: <<http://bibliodoc.uci.cu/pdf/reg00060.pdf>>.
- Margarita André Ampuero, y María Gulnara Baldoquín de la Peña. UN SISTEMA DE SOPORTE A LA DECISIÓN PARA LAASIGNACIÓN DE RECURSOS HUMANOS A EQUIPOS DEPROYECTOS DE SOFTWARE. 2010, 9.

REFERENCIAS BIBLIOGRÁFICAS



Norberto Pelegrín Entenza, y Margarita Sosa Monzón. REFLEXIONES TEÓRICAS EN TORNO A LA GESTIÓN POR COMPETENCIAS. *PUBLICACIÓN ELECTRÓNICA DEL CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN, EHT DE TRINIDAD* Julio 2004, 24.

Oficina Nacional de Normalización (NC 3000). SISTEMA DE GESTIÓN INTEGRADA DE CAPITAL HUMANO—VOCABULARIO. 2007.

Ortas. *Ingeniería de Requerimientos*. 1997.

Roger S. Pressman. *Libro Ingeniería del Software: Un enfoque práctico*. 2007.

Universidad de las Ciencias Informáticas. Conferencia: Introducción a la Disciplina de Análisis y Diseño. 2011.



ANEXOS

Anexo 1. Encuestas

Encuesta Realizada a Ing. Yoansy Reyes: Subdirector de Formación del CEIGE.

¿Existe un proceso definido actualmente para la selección del personal para ocupar un rol determinado en un proyecto?

En la actualidad lo que se hace en el centro es tratar de no ubicar en algunos roles a personas que no sean estudiantes, estos roles son: Líder de Proyecto, Arquitecto. Esta es una de las principales condiciones que se orientaron tener en cuenta a nivel de universidad, pero no existe un proceso que defina como realizar esta selección.

¿Qué condiciones debe tener el sistema para realizar una selección adecuada de los RRHH?

La UCI no tiene un proceso definido para realizar esta selección, por lo que hay que guiarse por las formas en que lo realizan a nivel internacional y unirlos a las condiciones definidas por la universidad.

¿Se realiza en el centro la gestión por competencias de los recursos humanos?

En la actualidad no se realiza en el centro la gestión por competencias de los recursos humanos, siendo este uno de los principales problemas existentes dentro del mismo a la hora de realizar una asignación de roles según habilidades y capacidad que debe tener una persona para ocupar el mismo.

Anexo 2. La Ingeniería de Software



Figura 22. La Ingeniería de Software es una tecnología multicapa.

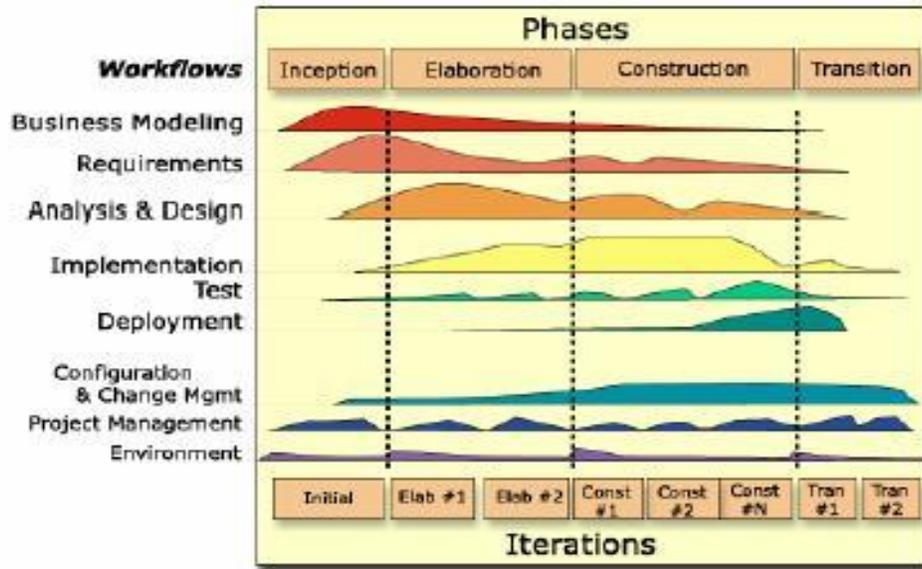


Figura 23. Metodología Rational Unified Process (RUP).

Anexo 3 Casos de uso del sistema

Los casos de uso del sistema que aparecen a continuación tienen como objetivo satisfacer los requisitos funcionales descritos.

| | |
|-------------|--|
| CU- 1 | Solicitud de matrícula |
| ACTOR | trabajador |
| DESCRIPCION | El trabajador debe solicitar la matrícula al curso al que desee matricularse |
| REFERENCIA | R1 |

Tabla 6. CU Solicitud de matrícula.

| | |
|-------------|---|
| CU- 2 | Petición de un nuevo curso |
| ACTOR | trabajador |
| DESCRIPCION | El trabajador debe realizar la petición de realización de un curso que no existe hasta el |



| | |
|------------|----------------------------------|
| | momento y que desee matricularse |
| REFERENCIA | R3 |

Tabla 7. CU Petición nuevo curso.

| | |
|-------------|---|
| CU- 3 | Chequear matrícula. |
| ACTOR | trabajador |
| DESCRIPCION | El trabajador debe constatar si la petición de matrícula de un curso fue aceptada o no. |
| REFERENCIA | R4 |

Tabla 8. CU Chequear estado de la matrícula.

| | |
|-------------|--|
| CU- 4 | Realizar encuesta |
| ACTOR | trabajador |
| DESCRIPCION | El trabajador deberá llenar todos los campos de preguntas que se le realicen en una encuesta sobre un tema dado. |
| REFERENCIA | R5 |

Tabla 9. CU Realizar encuesta.