



**Universidad de las Ciencias Informáticas  
Facultad 9**

## **Análisis y Diseño del portal "Calidad Facultad 9"**

---

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS**

**Autor:**

**Dayana Benitez Betancourt**

**Tutor:**

**Ing. Yesleny Becerra Torreira**

**Ciudad de La Habana, mayo de 2010  
"Año del 51 Aniversario del Triunfo de la Revolución"**

# Opiniones y avales

---

## GENERALES DEL TUTOR/COTUTOR

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Dayana Benitez Betancourt

\_\_\_\_\_

Yesleny Becerra Torreira

\_\_\_\_\_



## RESUMEN

La Universidad de las Ciencias Informáticas tiene la misión de impulsar el desarrollo de la producción de software. Para lograrlo es necesario que los productos tengan la calidad requerida.

En la facultad 9 existe un Grupo encargado de llevar el control de la calidad de cada uno de los proyectos. Este grupo para realizar correctamente sus funciones está dividido en áreas de trabajo, resumidas en: auditorías y revisiones, estrategia de prueba, recursos humanos y procedimiento de inicio de proyecto. En la actualidad el grupo está presentando dificultades a la hora de gestionar y acceder a la documentación, por lo que se hace necesario desarrollar un sistema para eliminar dichas deficiencias.

El presente trabajo de diploma que lleva por título: Análisis y Diseño del Portal “Calidad Facultad 9”, tiene como objetivo desarrollar el análisis y diseño de un sistema para la gestión de la información del grupo Calidad de la Facultad 9. Para lograr el objetivo principal se realizó un estudio de los procesos de gestión de información del Grupo Calidad de la Facultad 9. Además fue necesario determinar la metodología de desarrollo, la herramienta CASE (Ingeniería de Software Asistida por Computadora) y el lenguaje de modelado que permitieron desarrollar el proceso ingenieril del sistema. Una vez culminada la solución propuesta se hizo preciso aplicar métricas que permitieron eliminar errores y ambigüedades que en futuro hubieran provocado grandes trastornos.

## PALABRAS CLAVES

Modelamiento, ingeniería, herramienta CASE, metodología.



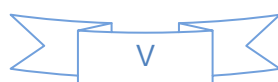
## ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1.    Introducción.....	5
1.2.    Conceptos Principales.....	5
1.2.1.    Calidad de software.....	5
1.2.2.    Proceso.....	5
1.2.3.    Gestión de la Información.....	5
1.2.4.    Proceso de Desarrollo de Software.....	6
1.3.    Procesos de Gestión de la Información del Grupo Calidad Facultad 9.....	6
1.3.1.    Descripción General.....	6
1.3.2.    Descripción de los procesos del Grupo Calidad.....	6
1.4.    ¿Por qué portal y no aplicación de escritorio?.....	13
1.4.1.    Aplicación de escritorio.....	13
1.4.2.    Aplicación Web.....	14
1.4.3.    Portal.....	16
1.5.    El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.....	17
1.5.1.    ¿Qué es UML?.....	17
1.6.    El Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución. 19	
1.6.1.    RUP.....	19
1.6.2.    XP.....	20
1.7.    Herramienta Case.....	21
1.7.1.    Visual Paradigm.....	21

# Índice

---

1.7.2. Rational Rose .....	22
1.7.3. ¿Por qué Visual Paradigm? .....	23
1.8. Conclusiones .....	23
CAPÍTULO 2: Negocio, Requisitos, Sistema, Análisis y Diseño. ....	24
2.1. Introducción.....	24
2.2. Modelo de Negocio.....	24
2.2.1. Procesos de negocio.....	24
2.2.2. Actores y trabajadores del negocio .....	24
2.2.3. Diagrama de Casos de Uso del Negocio.....	26
2.2.4. Diagrama de actividades para cada Caso de Uso del Negocio .....	27
2.2.5. Modelo de objeto .....	29
2.2.6. Descripción de los casos de uso del Negocio .....	29
2.3 Requisitos Funcionales .....	32
2.4 Requisitos no funcionales (RNF) .....	34
2.5 Descripción del sistema propuesto .....	35
2.5.1 Descripción de los actores .....	35
2.5.2 Diagrama de Caso de Uso del Sistema.....	35
2.5.3 Describir cada Caso de Uso del Sistema .....	36
2.6. Análisis.....	61
2.7. Diseño .....	62
2.8. Conclusiones.....	62
CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA .....	63
3.1. Introducción.....	63
3.2. Método de validación de los resultados .....	63



# Índice

---

3.3. Aplicación de la métrica de la calidad de especificación de los requisitos.....	63
3.4. Métricas de casos de uso del sistema .....	65
3.5. Conclusiones .....	67
Conclusiones generales.....	68
Recomendaciones .....	69
Bibliografía referenciada .....	70
Bibliografía consultada.....	72

# Introducción

---

## INTRODUCCIÓN

En la actualidad, el acelerado desarrollo de las tecnologías de la información, ha traído como consecuencia que la producción de software a nivel mundial constituya un importante renglón en el ámbito económico y social. Según Manuel Henríquez Lagarde “Cuba ha tomado una serie de medidas para informatizar al pueblo. Parte de ese esfuerzo se ve en la instalación de equipos de computación en todas las escuelas del país y la producción de gran cantidad de software educativo, que cubren una amplia gama de necesidades en la educación primaria, secundaria, preuniversitaria y tecnológica. Se incluye la ampliación de los estudios de formación en esta área, en especial con el incremento del número de institutos tecnológicos de informática y la creación de la Universidad de las Ciencias Informáticas (UCI)”. (1)

La UCI como eslabón fundamental en la informatización del país, tiene como misión la producción de productos altamente calificados, lo cual se realiza con la vinculación estudio trabajo. Este centro de enseñanza no solo se creó por el beneficio que aporta al producir un sistema informático para uso interno, sino también con el objetivo de insertarse en el mercado de software mundial por la ayuda económica que esto aportaría.

En todas las esferas de la vida se ambiciona lo más cercano a la perfección, que no es más que el requisito fundamental que exige el cliente cuando solicita un producto de software. La UCI para alcanzar un alto nivel de calidad y obtener la preferencia de los clientes se trazó como estrategia incorporar a cada una de las facultades un grupo de trabajadores y estudiantes organizados, que se encargan de controlar la calidad de cada software desarrollado en los proyectos productivos de la universidad. El grupo Calidad de la Facultad 9 para realizar correctamente sus funciones está dividido por áreas de trabajo que se resumen en: Auditorías y Revisiones, Recursos Humanos, Procedimiento de Inicio de Proyecto y Estrategia de Prueba.

En la actualidad el Grupo está presentando algunas deficiencias que no permiten un funcionamiento óptimo, algunos de estos problemas son:

- La gestión de la información referente a cada una de las áreas del Grupo Calidad Facultad 9 se lleva a cabo de forma manual por la inexistencia de un sistema que sea capaz de gestionar la documentación de manera efectiva.

# Introducción

---

- El trabajo que se realiza en el Grupo requiere gran cantidad de planillas, estas se manejan a través de correos o medios de almacenamiento lo que puede provocar pérdida o duplicidad de la información.
- El jefe de Recursos Humanos recoge la información de los profesores, estudiantes y computadoras de forma manual.
- Un integrante de un área determinada no tiene permisos para acceder a toda la documentación, por lo que no pueden informarse del estado actual del proyecto que está siendo inspeccionado.
- La dependencia de un repositorio hace que exista la necesidad de instalar en las máquinas del Grupo Calidad una herramienta que cumpla como única función permitir la conexión al repositorio.

A partir de las reflexiones hechas anteriormente, se identificó el siguiente **Problema a Resolver**: La gestión de forma manual de la información referente al Procedimiento de Inicio de Proyecto, Estrategia de Prueba, Recursos Humanos y Auditorías y Revisiones dificulta el acceso a la documentación que arrojan estos procesos del grupo calidad de la Facultad 9.

Realizando un análisis del problema se plantea el siguiente **objetivo general**: Desarrollar el análisis y diseño de un sistema para la gestión de la información del grupo Calidad de la Facultad 9.

Para el cumplimiento del objetivo general se definieron los siguientes **objetivos específicos**:

1. Elaborar los artefactos derivados del proceso ingenieril de análisis y diseño de un sistema para la gestión de la información del Grupo Calidad de la Facultad 9.
2. Evaluar la calidad de los artefactos obtenidos.

Analizando el problema a resolver se define como **objeto de estudio**: Los procesos de Gestión de la Información del Grupo Calidad de la Facultad 9. Para el desarrollo de este trabajo es necesario enmarcar la investigación en el análisis y diseño de un sistema para la gestión de la información que constituye el **campo de acción**.

Se propone como **idea a defender**: La elaboración de la documentación técnica correspondiente al análisis y diseño del entorno de negocio, evaluada a partir de las métricas



# Introducción

---

de calidad seleccionadas, garantizará al equipo de desarrollo trabajar de manera organizada en la implementación del sistema.

Para dar cumplimiento a los objetivos específicos planteados se formulan las **tareas** que a continuación se detallan:

1. Describir el estado actual de los procesos de gestión de la información del Grupo Calidad de la Facultad 9.
2. Describir el estado actual del desarrollo de los sistemas de gestión de información.
3. Argumentar el uso de la metodología de desarrollo a utilizar, lenguaje de modelado y herramienta CASE.
4. Desarrollar la documentación técnica correspondiente al modelo del negocio, requisitos funcionales y no funcionales y análisis y diseño de los procesos de gestión de la información del Grupo Calidad facultad 9.
5. Aplicar métricas para la evaluación de la calidad de la documentación técnica generada.

Durante el desarrollo de la investigación se emplearon **métodos científicos**:

## **Métodos teóricos:**

Histórico lógico: Se utilizó con el objetivo de caracterizar los procesos de gestión de la información que se realizan actualmente en el Grupo Calidad de la Facultad 9.

Modelación: “La modelación es el método mediante el cual se crean abstracciones con el objetivo de explicar la realidad”. (2)

Se utilizó para modelar los procesos del Grupo Calidad Facultad 9.

## **Métodos empíricos:**

Observación: Se utilizó con el objetivo de observar las actividades que se desarrollan en el Grupo Calidad Facultad 9 y para determinar el flujo de los procesos además de los requerimientos funcionales y no funcionales.

Entrevista: Se realizaron entrevistas a todos los profesores del Grupo Calidad Facultad 9 con el objetivo de comprender con más detalle cómo funcionan los procesos de Procedimiento de Inicio de Proyecto, Estrategia de Prueba, Recursos Humanos y Auditorías y Revisiones.

# Introducción

---

Se utilizó la técnica no probabilista con muestreo intencional, la población son los profesores del grupo calidad que a su vez se convierte en la muestra ya que es muy pequeña.

Luego de darle cumplimiento a los objetivos propuestos, se esperan como **posibles resultados**: Los artefactos ingenieriles obtenidos en el análisis y diseño del proceso de desarrollo del portal “Calidad Facultad 9”.

- Modelo del Negocio.
- Requerimientos (Funcionales, No Funcionales).
- Modelo del Sistema.
- Modelo de Análisis.
- Modelo de Diseño.

El trabajo de diploma consta de 3 capítulos:

**Capítulo 1:** “Fundamentación teórica”: Este capítulo contiene la base teórica para entender el problema planteado, donde se abordan algunos conceptos asociados al dominio del problema para facilitar su entendimiento. Se describen detalladamente todos los procesos del grupo calidad que se quieren automatizar, presentando el análisis realizado para seleccionar la metodología, el lenguaje de modelado y la herramienta case que se utiliza para el desarrollo ingenieril del Portal Calidad Facultad 9.

**Capítulo 2:** “Negocio, Requisitos, Sistema, Análisis y Diseño”: En este capítulo se plasma todo lo relacionado con la modelación del negocio que incluye el Diagrama de Caso de Uso del Negocio, así como una descripción de cada actor y trabajador, también están los diagramas de actividades y de objeto para cada Caso de Uso del Negocio. Se encuentra la especificación de requisitos funcionales y no funcionales así como el Diagrama de Caso de Uso del Sistema, la descripción de cada Caso de Uso del Sistema y para finalizar están todos los diagramas correspondientes al análisis y diseño.

**Capítulo 3:** “Validación de la solución propuesta”: Se hace uso de métricas para evaluar los artefactos generados en el transcurso de la investigación, lo cual permite demostrar que la solución propuesta es correcta.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### 1.1. Introducción

En el presente capítulo se abordan algunos conceptos asociados al dominio del problema, con el objetivo de facilitar una mayor comprensión del tema. Se describen detalladamente todos los procesos del grupo calidad que se quieren automatizar. Se presenta el análisis realizado para seleccionar la metodología, el lenguaje de modelado y la herramienta case que se utiliza en el desarrollo ingenieril del Portal Calidad Facultad 9.

### 1.2. Conceptos Principales

#### 1.2.1. Calidad de software

“Es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente.” (3)

”La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, portabilidad, usabilidad, seguridad e integridad.” (16)

Un software con calidad es el que cumple con todas las necesidades del cliente, es el software que nos proporciona seguridad y facilidad de uso.

#### 1.2.2. Proceso

“El concepto hace referencia a la acción de ir hacia adelante, al transcurso del tiempo, al conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial”. (4)

”Conjunto de las fases sucesivas de un fenómeno natural o de operación artificial”. (17)

Un proceso no es más que los pasos que se llevan a cabo para lograr un objetivo.

#### 1.2.3. Gestión de la Información

“Comprende las actividades relacionadas con la obtención de la información adecuada, a un precio adecuado, en el tiempo y lugar adecuado, para tomar la decisión adecuada”. (5)

# Capítulo 1: Fundamentación teórica

---

La gestión de la información se puede definir como el conjunto de actividades realizadas con el fin de controlar, almacenar y posteriormente recuperar adecuadamente la información producida, recibida o retenida por cualquier organización en el desarrollo de sus actividades.

## 1.2.4. Proceso de Desarrollo de Software

“Es el conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software”. (6)

Es el conjunto de pasos que se siguen para lograr un producto de software.

## 1.3. Procesos de Gestión de la Información del Grupo Calidad Facultad 9

### 1.3.1. Descripción General

Un software con calidad es un producto que cumple adecuadamente con los requisitos que demanda el cliente. Este concepto se viene trabajando desde hace algunos años por la necesidad de obtener un mejor producto, con el objetivo de satisfacer al cliente y asegurar que exista la posibilidad de que una empresa o persona vuelvan a solicitar los servicios de la universidad. En la Facultad 9 de la UCI se controla la calidad de cada producto con la ayuda del Grupo Calidad, esta tarea es cumplida mediante procesos de Gestión de la Información. Estos procesos consisten en la inserción, actualización o eliminación de documentación referente a cada una de las áreas de trabajo del Grupo Calidad.

### 1.3.2. Descripción de los procesos del Grupo Calidad

El Grupo de Calidad de la Facultad 9 para realizar correctamente sus funciones está dividido por áreas de trabajo:

El área de trabajo **Procedimiento de Inicio de proyecto** tiene como objetivo: Garantizar un nivel de calidad básico homogéneo en todos los proyectos de la facultad 9, como requerimiento para su certificación.

Para que un proyecto pueda ser certificado, primeramente debe tener bien definidas las Condiciones y Normativas y además realizar todas las actividades que se encuentran en el documento Procedimiento de Inicio de Proyecto (**Ver Anexo 1**). Cuando el proyecto finaliza las tareas anteriormente mencionadas, el líder podrá solicitar la certificación, esto se realiza de la siguiente forma:

# Capítulo 1: Fundamentación teórica

---

1. La solicitud de certificación se lleva a cabo mediante un correo electrónico por parte del líder de proyecto al vicedecano de producción con copia al asesor de calidad de la facultad. Se conservará dicho correo como registro del hecho.
2. Una vez enviada la solicitud de certificación, el grupo de calidad dispondrá de 7 días hábiles para verificar que el proyecto cumpla con los requisitos para la certificación. Culminada la revisión, el Grupo de Calidad notificará al vicedecano de producción los resultados de la misma.
3. La notificación del resultado de la revisión para la certificación se realizará mediante un correo electrónico de parte del responsable de la revisión designado por el grupo de calidad de la facultad al vicedecano de producción. Se conservará dicho correo como registro del hecho.

Será responsabilidad del líder de proyecto supervisar que se cumpla lo establecido en los puntos anteriores.

El significado de que un proyecto sea certificado es que obtiene la autorización para comenzar a desarrollar el producto que tenían previsto.

**Proceso de Auditorías y Revisiones:** una auditoría o revisión se realiza con el objetivo de detectar errores o faltantes que existan en el expediente de proyecto y una vez detectados los errores asegurarse de que se le den solución. El área de trabajo de Auditorías y Revisiones del Grupo Calidad Facultad 9 para efectuar una auditoría primeramente junto con el cliente hacen una reunión de apertura donde el proyecto debe facilitarle al jefe de Auditorías y Revisiones el expediente de proyecto o permitirle el acceso al repositorio donde se encuentre la información, después los auditores serán los encargados de verificar si el expediente de proyecto cumple con los Lineamientos de Calidad (**Ver Anexo 2**), posteriormente se hace la reunión de cierre en la que se le informa al proyecto en un documento final (**Ver Anexo 4**) las no conformidades encontradas, con el objetivo de que sean erradicados todos los problemas encontrados. Después de terminada la auditoría se hará un seguimiento para verificar que las no conformidades sean resueltas.

**Proceso de Recursos Humanos:** es el encargado de tener registrada información concreta de cada uno de los profesores, estudiantes y computadoras del laboratorio del Grupo Calidad Facultad 9. Esta información se desglosa en los siguientes puntos:

<b>Estudiante:</b>	<b>Profesor</b>	<b>Computadora</b>
--------------------	-----------------	--------------------

## Capítulo 1: Fundamentación teórica

---

<ul style="list-style-type: none"><li>• Nombre</li><li>• Apellidos</li><li>• Año que cursa.</li><li>• Grupo.</li><li>• Apartamento.</li><li>• Teléfono.</li><li>• Tema de Investigación.</li><li>• Tutor.</li><li>• Evaluación de cortes.</li><li>• Trabajos realizados.</li><li>• Trabajos publicados.</li><li>• Nota final de PP.</li></ul>	<ul style="list-style-type: none"><li>• Nombre</li><li>• Apellidos</li><li>• Apartamento.</li><li>• Teléfono.</li><li>• Responsabilidad (líder del grupo, auditor líder, arquitecto líder, probador líder, líder de procedimiento de inicio de proyecto).</li></ul>	<ul style="list-style-type: none"><li>• Nombre de personas que se sientan en la computadora</li><li>• Número de computadora.</li></ul>
---	---	--

**Tabla 1 Recursos del Grupo Calidad (Elaboración propia)**

**Proceso de Estrategia de prueba:** El grupo de prueba en estos momentos está trabajando de la siguiente forma, realiza un levantamiento de los tipos de pruebas que se llevan a cabo en cada uno de los proyectos de la facultad, capacita al personal que está involucrado con las pruebas en cada proyecto y cuando un producto está listo para liberarse se le realizan las pruebas de liberación.

A la hora de confeccionar los casos de prueba es necesario que el proyecto facilite la documentación correspondiente a las descripciones de los casos de uso, el jefe de pruebas es el encargado de recibir la información y entregársela a los probadores del Grupo Calidad Facultad 9, ellos se encargan de verificar la documentación y a su vez desarrollar los casos de prueba una vez culminados los casos de prueba los probadores se lo envían al jefe de pruebas y él es el responsable de comunicarle al jefe del proyecto cuáles fueron los resultados de las pruebas.

La Estrategia de Prueba de la Facultad 9 (**Ver Anexo 3**) tiene como:

**Objetivos:**

Definir un procedimiento para homogenizar las pruebas en los proyectos productivos de la facultad 9.

**Alcance:**

# Capítulo 1: Fundamentación teórica

---

Esta estrategia de prueba define cómo se efectuará el esfuerzo de prueba contra uno o más aspectos de los sistemas que se desarrollan en los diferentes proyectos productivos de la Facultad 9.

## **Roles de pruebas de software:**

- Diseñador de pruebas
- Gestor de prueba
- Analista de pruebas
- Verificador

Estos roles pueden ser desempeñados tanto por profesores como estudiantes que estén a cargo de las pruebas de software dentro del proyecto.

**Diseñador de pruebas:** Este rol dirige la identificación de las técnicas, herramientas y directrices apropiadas para implementar las pruebas necesarias y para proporcionar orientación al esfuerzo de prueba sobre los requisitos de recursos correspondientes.

**Gestor de prueba:** Este rol dirige el apoyo de calidad y prueba, la planificación y gestión de recursos y la resolución de cuestiones que impiden el esfuerzo de prueba.

**Analista de pruebas:** Este rol identifica y define las pruebas necesarias, supervisa el proceso y los resultados de cada ciclo de prueba y evalúa la calidad global. El rol también representa a los interesados que no tienen una representación directa o regular en el proyecto.

**Verificador:** Este rol realiza y registra los resultados de las pruebas.

## **Casos de prueba**

“La plantilla utilizada para los casos de prueba fue la propuesta a principio del año 2009. Dicha plantilla está compuesta por varias secciones, las que se describirán brevemente a continuación para su mejor entendimiento.

- **Control de versiones:** los casos de pruebas se pueden ir refinando, y en esta tabla se registran las versiones por la que pasa la plantilla.

Fecha	Versión	Descripción	Autor
-------	---------	-------------	-------

# Capítulo 1: Fundamentación teórica

---



**Tabla 2 Control de versiones (7)**

- **Descripción General:** en esta sección se describe brevemente el caso de uso correspondiente al caso de prueba.
- **Condiciones de Ejecución:** en esta sección se muestran las condiciones necesarias para la ejecución del caso de uso referente al caso de prueba.
- **Secciones a probar en el caso de uso:** en esta tabla se describe el caso de uso correspondiente al caso de prueba, a través de sus columnas.

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

**Tabla 3 Secciones a probar en el caso de uso (7)**

- Nombre de la sección: en esta columna se nombra la sección o las secciones en las que está dividido el caso de uso. Se enumeran de la siguiente manera: SC1, SC2,..., SCX.
  - Escenarios de la sección: en esta columna se muestran los escenarios posibles dentro de una sección determinada. Se enumeran de la siguiente manera: EC1.1, EC2.1,..., ECX.Y.
  - Descripción de la funcionalidad: aquí se describe muy brevemente la funcionalidad del escenario.
  - Flujo Central: esta columna pertenece al flujo central del escenario, descrito detalladamente en el caso de prueba.
- **Descripción de las variables:** en esta tabla se describen las variables a utilizar en el caso de prueba.

No	Nombre de campo	Clasificación	Puede ser nulo	Descripción

**Tabla 4 Descripción de las variables (7)**

- No: es el número que identificará la variable.
- Nombre del campo: es el nombre de la variable.



## Capítulo 1: Fundamentación teórica

---

- Clasificación: especifica el tipo de dato que puede tomar la variable.
- Puede ser nulo: especifica si la variable puede ser nula en algún momento.
- Descripción: breve descripción de la variable.
- **Secciones:** esta tabla se hace para cada sección, mostrándose de ella las clases válidas e inválidas que se probarán.

Id del escenario	Escenario	Variables		Respuesta del Sistema	Resultado de la Prueba

**Tabla 5 Secciones (7)**

- Id del escenario: identificador del escenario que se probará.
- Escenario: nombre del escenario que se probará.
- Variables: se enumeran las variables que participan en esta sección específica y los valores que toman para formar las clases. Las variables pueden ser: válidas (V), inválidas (I) o no aplica (N/A) para las variables que no participan en determinado escenario.
- Respuesta del Sistema: se describe lo que el sistema debe hacer frente a determinada clase (V o I).
- **Registro de defecto y dificultades encontradas:** en esta sección se registran los fallos encontrados específicamente en el caso de uso que se está probando.

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa	No Significativa	Recomendación	Estado NC	Respuesta del Equipo de Desarrollo

**Tabla 6 Registro de defecto y dificultades encontradas (7)**

# Capítulo 1: Fundamentación teórica

---

La tabla es la sección más importante dentro de la plantilla de No Conformidades (NC), porque es precisamente donde se recogen los defectos encontrados durante el proceso de pruebas. Por este motivo se decidió describirla detalladamente.

- Elemento: especifica el nombre del elemento que se está probando.
- No.: especifica el número de la no conformidad correspondiente, es el identificador de ésta.
- No conformidad: descripción clara y detallada de la no conformidad.
- Aspecto correspondiente: especifica el lugar donde ocurrió la no conformidad. Debe estar bien especificado y puede hacerse uso de imágenes para mostrar exactamente el error y donde ocurrió.
- Etapa de detección: especifica en que etapa de las pruebas se descubrió el error.
- Clasificación: especifica el tipo de no conformidad.
- Estado NC: especifica el estado en que se encuentra la no conformidad detectada, una vez entregadas al grupo de probadores las respuestas a todas las no conformidades. Puede estar: respondida (RA) cuando el equipo de desarrollo dio respuesta a la no conformidad, pendiente (PD) cuando la no conformidad aún no ha sido respondida, o no procede (NP) cuando el equipo de desarrollo no considera la no conformidad como un problema a resolver.
- Respuesta del equipo de desarrollo: especifica la respuesta a cada una de las no conformidades por parte del equipo de desarrollo.

Las no conformidades se clasificaron en Significativas, No Significativas y Recomendaciones, para priorizar los errores encontrados, según la gravedad de estos. Es válido aclarar que el hecho de ser significativas o no, no quiere decir que se le de menos o más importancia, ambos constituyen errores que hay que reparar.

- No Significativas: son los errores detectados que afectan solamente la funcionalidad que se está probando.
- Significativas: son aquellos errores que pueden afectar más allá de la funcionalidad que se está probando.

- Recomendaciones: son sugerencias dadas por los probadores al equipo de desarrollo para mejorar funcionalidades o para evitar un posible error en el futuro. “(7)

## 1.4. ¿Por qué portal y no aplicación de escritorio?

### 1.4.1. Aplicación de escritorio

Una aplicación de escritorio es desarrollada para ser ejecutada en una plataforma específica, ya sea Windows o Linux. El desarrollo sobre una plataforma, normalmente, implica que la aplicación no pueda ser ejecutada en otras. Una aplicación de escritorio posee varias características dentro de las cuales se tiene que requiere de su instalación en el disco duro de la computadora, son independientes de internet por lo que su ejecución no depende de si se posee conexión o no en determinado momento, poseen gran velocidad de acceso aunque esto es relativo según el sistema operativo utilizado y la potencia del procesador de la máquina, con una computadora de potencia media la aplicación debe ejecutarse de manera óptima, debe disponer de gran cantidad de opciones que acompañan el objetivo de la herramienta dentro de los cuales se puede tener las opciones de: creación, edición y almacenamiento de información.

La utilización y creación de aplicaciones de escritorio tiene varias ventajas con relación a las aplicaciones web, de las que se pueden mencionar:

- Mayor capacidad gráfica visual.
- Menor tiempo de respuesta (aplicación más rápida).
- Mayor personalización.
- No requiere conectividad para su funcionamiento.

No todas las características de las aplicaciones de escritorio traen consigo beneficios, estas también poseen desventajas dentro de las que se encuentran:

- Tiene que existir una versión para cada plataforma.
- Se limitan a manejar datos en el sistema de archivos local.
- Las aplicaciones más completas o complejas requieren de gran espacio en disco duro.
- Duplicidad de datos por la falta de unificación de los mismos.
- Diseminación de la información y lógica en muchas partes, en cada computadora que la use.

- Traumas a la hora de realizar actualizaciones o correcciones al programa ya que las instalaciones están diseminadas.
- La administración de la seguridad, controlando el acceso de los usuarios a información no relevante o privada es un caos.
- Dificultad para configurar cada una de las instalaciones, dependiendo de las necesidades de cada usuario.

## 1.4.2. Aplicación Web

Una aplicación Web es un sistema informático que los usuarios utilizan accediendo a un servidor Web a través de Internet o de una intranet mediante un navegador. Es decir, es una aplicación de software que se codifica en un lenguaje soportado por los navegadores Web en la que se confía la ejecución al navegador. Las aplicaciones Web son populares debido a que el navegador Web como cliente ligero es muy práctico y no hay que instalar ningún programa con CD, descargar o actualizar el software para su utilización.

El uso y creación de aplicaciones Web tiene varias ventajas con relación a las aplicaciones de escritorio, de las que se pueden mencionar:

- Sus actualizaciones se hacen de una manera muy sencilla, sin necesidad de hacer descargas, instalaciones o comprar físicamente el producto.
- Para su ejecución simplemente basta con teclear su dirección URL en cualquier navegador Web.
- No dependen de ningún sistema operativo ni configuración de hardware específica.
- Se puede acceder desde cualquier tipo de computadora sin la instalación de software.
- Facilidad del control de acceso.
- Naturalmente multiusuario.
- Solamente una instalación/versión de software para mantener.
- La información se encuentra toda en un mismo servidor al que se puede acceder desde cualquier lugar.

# Capítulo 1: Fundamentación teórica

---

- Son aplicaciones multiplataforma por lo que los usuarios no tienen que usar un sistema operativo en específico.
- Ahorran costes de hardware y software.
- Fácil uso, sin necesidad de conocimientos de informática.
- Facilitan el trabajo colaborativo y a distancia.
- Rápida actualización.
- Los datos son más seguros.

No todas las características de las aplicaciones Web traen consigo beneficios, estas también poseen desventajas dentro de las que se encuentran:

- El tiempo de respuesta es mayor comparado con las aplicaciones de escritorio.
- La capacidad gráfica es más limitada comparadas con las aplicaciones de escritorio.
- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones Web (mediante formularios principalmente) son muy escasas.

Las aplicaciones Web se dividen en tres grupos fundamentales:

- Aplicaciones Web informativas: Son aplicaciones en las que sólo se muestra información de un tema específico, teniendo la página principal mayor importancia a la hora de su elaboración porque es la que introduce el contenido sobre el cual tratará la aplicación.
- Aplicaciones Web de negocio: Incluyen las aplicaciones informativas además de otras opciones donde el cliente o usuario selecciona lo que necesita de un producto o servicio, en esta aplicación el intercambio de información es más complejo, después de la selección de los elementos que el usuario necesita, el sistema le pide los datos y los almacena en la base de datos con toda la información introducida por el usuario.
- Portales Web: Son el grupo más complejo de las aplicaciones Web. Un portal Web es conocido como “Puerta o Puerto”. Es un sitio Web que sirve como punto de partida para la navegación en la red posibilitando de forma fácil y sencilla, el acceso a una serie de recursos y de servicios, entre

## Capítulo 1: Fundamentación teórica

---

los que suelen encontrarse facilidades para la localización de sitios como son los buscadores, noticias, buzones de correo, directorios telefónicos y foros de discusión con temas de interés general.

Internet se caracteriza por la enorme cantidad de información que maneja, es donde se pueden hallar portales que han sido desarrollados en el transcurso de los años en todos los países del mundo.

A nivel mundial se puede encontrar Portales médicos, este portal está diseñado principalmente para personas que estudien en cualquier campo de la medicina, tiene muchas opciones como fórum de debate, buscador, artículos, cursos online, últimas noticias y publicaciones, diccionario médico, fotografías e imágenes, consejos para elegir un seguro médico, etc.

A nivel nacional se tiene el Portal Cuba que recoge información general organizada por categorías que agrupan las más importantes temáticas del país como son ciencia, cultura, educación, economía, leyes, política, medios de comunicación, salud, turismo y tiene buscador.

A nivel UCI se tiene el Portal UCI en este se pueden encontrar informaciones con relación a la universidad, tiene una galería donde aparecen fotos de cada evento y actividad en la que participa la UCI, un buscador donde aparecen datos relacionados al tema que desea investigar, un contacto donde podrá enviar sugerencias y opiniones, un mapa de la universidad, etc.

### 1.4.3. Portal

“Un portal puede definirse como la evolución del concepto de "Sitio Web”, en donde la Web se ha convertido en el punto de entrada a un conjunto de servicios e información, a los que se accede de formas sencilla, unificada y segura”. (8)

“Los portales ofrecen una gran diversidad de servicios: listado de sitios Web, buscador, noticias, e-mail, información meteorológica, chat, grupos de discusión y comercio electrónico. En muchos casos el usuario puede personalizar la presentación del portal. Algunos de los más conocidos son AltaVista, Yahoo!, Netscape y Microsoft”. (9)

Desde el punto de vista funcional, un portal se diferencia de un sitio Web convencional debido a que incluye muchas más secciones, como pueden ser noticias, eventos, directorios, que requieren estar siempre actualizados. Por otro lado, normalmente se caracterizan por administrar claves de acceso, destinadas a restringir la visita a determinadas páginas con información de alto valor. Realmente el portal

# Capítulo 1: Fundamentación teórica

---

es la evolución de un sitio Web, más en el ámbito conceptual que en el tecnológico, se ha ido evolucionando en contenidos y servicios.

Existen varias clasificaciones de portales como pueden ser:

- Horizontales: Tienen un propósito general y una audiencia amplia por lo que son más masivos, tratando de llegar a toda la gente con muchas cosas, teniendo como ejemplo el portal de Yahoo y AltaVista.
- Verticales: Se dirigen a usuarios para ofrecer contenido dentro de un tema específico como puede ser un portal de música, empleo, inmobiliario, un portal de finanzas personales, arte o de deportes.
  - ✓ Portal Intranet: Comunicación corporativa para los empleados.
  - ✓ Portal Extranet: Comunicación corporativa para los proveedores.
  - ✓ Portal Vertical: Comunicación corporativa con clientes.
- Diagonales: Son una mezcla entre el portal horizontal y el vertical. Se trata de portales que utilizan redes sociales o aplicaciones generalistas como Facebook.

Para darle solución a la problemática planteada se realizará un portal Web de gestión de información porque debe dar la posibilidad de crear, modificar y eliminar los datos almacenados. Un portal permitirá el acceso de varios usuarios al mismo tiempo a través de un vínculo y no tendrá que ser instalado en cada computadora que se necesite. No existe la posibilidad de que los datos estén duplicados porque son controlados en un único servidor. El portal a implementar es de tipo vertical pues se analizará, diseñará e implementará para un tema en específico que es la Calidad de Software.

## **1.5.El Lenguaje Unificado de Modelado (UML) como soporte de la modelación de la solución propuesta.**

### **1.5.1. ¿Qué es UML?**

UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y unas reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema. Este lenguaje nos indica cómo crear y leer los modelos.

# Capítulo 1: Fundamentación teórica

---

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones:

- **Visualizar:** UML permite expresar de una forma gráfica un sistema, de forma que otro lo puede entender.
- **Especificar:** UML permite especificar cuáles son las características de un sistema antes de su construcción.
- **Construir:** A partir de los modelos especificados se pueden construir los sistemas diseñados.
- **Documentar:** Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Un modelo UML está compuesto por tres clases de bloques de construcción:

- **Elementos:** Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.).
- **Relaciones:** relacionan los elementos entre sí.
- **Diagramas:** Son colecciones de elementos con sus relaciones.

## Diagramas UML

Un diagrama es la representación gráfica de un conjunto de elementos con sus relaciones. En concreto, un diagrama ofrece una vista del sistema a modelar.

Para poder representar correctamente un sistema, UML ofrece una amplia variedad de diagramas que permiten visualizar el sistema desde varias perspectivas.

- Diagrama de casos de uso.
- Diagrama de clases.
- Diagrama de objetos.
- Diagrama de secuencia.
- Diagrama de colaboración.
- Diagrama de estados.
- Diagrama de actividades.
- Diagrama de componentes.



- Diagrama de despliegue. (10)

## 1.6.El Proceso Unificado de Desarrollo de Software (RUP) como base en el desarrollo de la solución.

En las últimas décadas se ha incrementado notablemente el interés en la realización de productos de software, y para el desarrollo de estos existen dos tipos de metodologías: las ágiles y las tradicionales o robustas, dentro del gran grupo de metodologías se encuentran RUP y Programación extrema (XP), de las que se abordarán algunas características para poder argumentar así la selección realizada.

### 1.6.1. RUP

”La metodología RUP divide en 4 fases el desarrollo del software:

- **Inicio:** Se determina la visión del proyecto.
- **Elaboración:** Se determina la arquitectura óptima.
- **Construcción:** Se obtiene la capacidad operacional inicial.
- **Transición:** Se obtiene el release del proyecto.

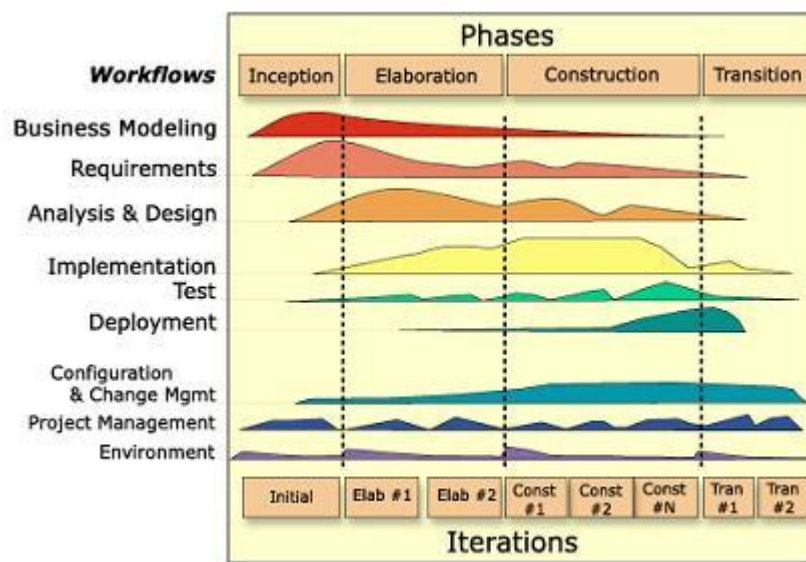


Figura 1 Fases e iteraciones de la metodología RUP (11)

Los elementos de RUP son:

# Capítulo 1: Fundamentación teórica

---

- **Actividades:** Son los procesos que se llegan a determinar en cada iteración.
- **Trabajadores:** Son las personas involucradas en cada proceso.
- **Artefactos:** Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

## 1.6.2. XP

Es una de las metodologías de desarrollo de software más exitosas en la actualidad utilizadas para proyectos de corto plazo, corto equipo y cuyo plazo de entrega era ayer. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto.

¿Qué es lo que propone XP?

Derechos del Cliente	Derechos del Desarrollador
<ul style="list-style-type: none"><li>• Decidir que se implementa.</li><li>• Saber el estado real y el progreso del proyecto.</li><li>• Añadir, cambiar o quitar requerimientos en cualquier momento.</li><li>• Obtener lo máximo de cada semana de trabajo.</li><li>• Obtener un sistema funcionando cada 3 ó 4 meses.</li></ul>	<ul style="list-style-type: none"><li>• Decidir cómo se implementan los procesos.</li><li>• Crear el sistema con la mejor calidad posible.</li><li>• Pedir al cliente en cualquier momento aclaraciones de los requerimientos.</li><li>• Estimar el esfuerzo para implementar el sistema.</li><li>• Cambiar los requerimientos en base a nuevos descubrimientos.</li></ul>

**Tabla 7 Derechos del cliente y desarrollador en XP (Elaboración propia)**

Lo fundamental en este tipo de metodología es:

# Capítulo 1: Fundamentación teórica

---

- La comunicación, entre los usuarios y los desarrolladores.
- La simplicidad, al desarrollar y codificar los módulos del sistema .
- La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales
- El cliente o el usuario se convierte en miembro del equipo”. (11)

Después de realizado el análisis de las metodologías RUP y XP se determinó que la más apropiada para modelar el sistema es RUP, por las siguientes razones:

- RUP es una metodología que tiene la particularidad de que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.
- RUP no necesita que el cliente forme parte del equipo de desarrollo al contrario de XP.
- RUP realiza las pruebas al final del producto al contrario de XP que esta constantemente realizando pruebas.
- La gran cantidad de artefactos que se generan en RUP contribuyen a un mejor entendimiento del problema.
- En el grupo calidad no se tiene mucha experiencia en la construcción de sistemas por lo que se necesita una metodología que genere documentación útil sobre el negocio y la más indicada para esto es RUP.
- Es una metodología con alta adaptabilidad a las condiciones reales del desarrollo del sistema. Es decir que se puede hacer más ágil según se necesite.

## 1.7. Herramienta Case.

### 1.7.1. Visual Paradigm

Visual Paradigm es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y

# Capítulo 1: Fundamentación teórica

---

proyectos UML. Presenta licencia gratuita y comercial. Es fácil de instalar, actualizar y compatible entre ediciones.

Características principales:

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
- Ingeniería inversa - Código a modelo, código a diagrama.
- Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Ingeniería inversa de bases de datos - Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
- Editor de figuras. (12)

## 1.7.2. Rational Rose

Rational Rose es una herramienta de producción y comercialización establecidas por Rational Software Corporation (actualmente parte de IBM). Rose es un instrumento operativo conjunto que utiliza el Lenguaje Unificado (UML) como medio para facilitar la captura de dominio de la semántica, la arquitectura y el diseño.

Sus características principales:

- No es gratuito, se debe hacer un previo pago para poder adquirir el producto.
- La ingeniería de código (directa e inversa) es posible para ANSI C++, Visual C++, Visual Basic 6, Java, J2EE/EJB, CORBA, Ada 83, Ada 95, Bases de datos: DB2, Oracle, SQL 92, SQL Server, Sybase, Aplicaciones WEB.
- Posee limitantes que la hacen débil en comparación a otras herramientas como Visual Paradigm, estas debilidades radican en la dependencia de la plataforma Windows y la integración solo con herramientas que estén en el mismo grupo de software propietario.

### 1.7.3. ¿Por qué Visual Paradigm?

Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Soporta la importación de modelos de Rational Rose. Cuenta con gran cantidad de tutoriales, es fácil de instalar, tiene características gráficas muy cómodas que facilitan la realización de los diagramas de modelado que sigue el estándar de UML y posee gran ventaja al ser libre, que es la estrategia que se está utilizando en el desarrollo del Portal Calidad, esto trae como beneficio que el producto final esté enteramente desarrollado sobre software libre. Además es una herramienta multiplataforma.

### 1.8. Conclusiones

Con el estudio realizado en este capítulo, se pudo describir de forma detallada cómo funcionan los procesos del Grupo Calidad Facultad 9, comprendiendo la necesidad que hay de automatizar estos procesos. Se seleccionó la metodología de desarrollo, lenguaje de modelado, la herramienta case más conveniente para el desarrollo de la ingeniería del software. Se definió el tipo de aplicación a desarrollar según la necesidad del grupo calidad facultad 9.

## **CAPÍTULO 2: Negocio, Requisitos, Sistema, Análisis y Diseño.**

### **2.1. Introducción**

En el presente capítulo se describen los procesos del negocio que están comprendidos en el objeto de estudio. Se identifican los actores, trabajadores y casos de usos que intervienen en dicho negocio. Además se enumeraron los requisitos funcionales y no funcionales que debe tener el sistema que se propone. También se identifican mediante los diagramas de casos de uso las relaciones de los actores que interactúan con el sistema y se confeccionó la descripción de los casos de uso del sistema.

### **2.2. Modelo de Negocio**

El modelamiento del negocio brinda una visión general de los procesos que existen en un determinado negocio, permitiendo comprender a qué se dedica el mismo, así como establecer una comunicación entre cliente y equipo de desarrollo.

#### **2.2.1. Procesos de negocio**

El Grupo Calidad de la Facultad 9 necesita gestionar la información que se maneja en cada una de las áreas Procedimiento de Inicio de Proyecto, Estrategia de Prueba, Recursos Humanos y Auditorías y Revisiones.

#### **2.2.2. Actores y trabajadores del negocio**

##### **Actores**

Según la metodología seleccionada un actor del negocio es cualquier individuo, grupo o entidad, organización, máquina o sistema de información externos, con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando interactúa con el negocio para beneficiarse de sus resultados. A continuación se presentan los actores que intervienen en el negocio.

<b>Actor</b>	<b>Justificación</b>
Integrante	Estudiante o profesor que va a entrar en el Grupo Calidad o ya era parte del mismo y sus datos son gestionados por el jefe de Recursos Humanos.

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

Cliente	Es el proyecto al que se le van a realizar las auditorías y revisiones, pruebas y procedimiento de inicio de proyecto.
Líder de proyecto	Es la persona que se encarga de dirigir y llevar el control sobre todo lo que sucede en su proyecto. En este caso es el que manda una solicitud para que su proyecto sea certificado.

**Tabla 8 Actores del negocio (Elaboración propia)**

### Trabajadores

Según RUP un trabajador del negocio es una abstracción de una persona o grupo de personas, una máquina o un sistema automatizado, que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio. Representa un rol.

Trabajador	Justificación
Jefe Recursos Humanos	Es el encargado de registrar los datos de un integrante (estudiante, profesor) o computadora.  Elimina datos de un nuevo integrante o una nueva computadora.  Modifica y adiciona datos de integrantes y computadoras.
Jefe Auditorías y Revisiones	Es el que dirige el área de auditorías y revisiones.
Jefe de Procedimiento de Inicio	Es el que dirige el área de procedimiento de inicio de proyecto.
Jefe de Pruebas	Es el que dirige el área de pruebas.
Probador	Todos los integrantes del grupo calidad que trabajan en el área de pruebas.
Auditor	Todos los integrantes del grupo calidad que trabajan en el área de auditoría y revisiones.

Iniciador	Todos los integrantes del grupo calidad que trabajan en el área de procedimiento de inicio de proyecto.
Vicedecano de producción	Recibe la solicitud de certificación por parte del líder de proyecto.
Asesor de calidad	Recibe la solicitud de certificación por parte del líder de proyecto y designa al responsable que realizará la revisión al proyecto que hace la solicitud.

Tabla 9 Trabajadores del negocio (Elaboración propia)

### 2.2.3. Diagrama de Casos de Uso del Negocio

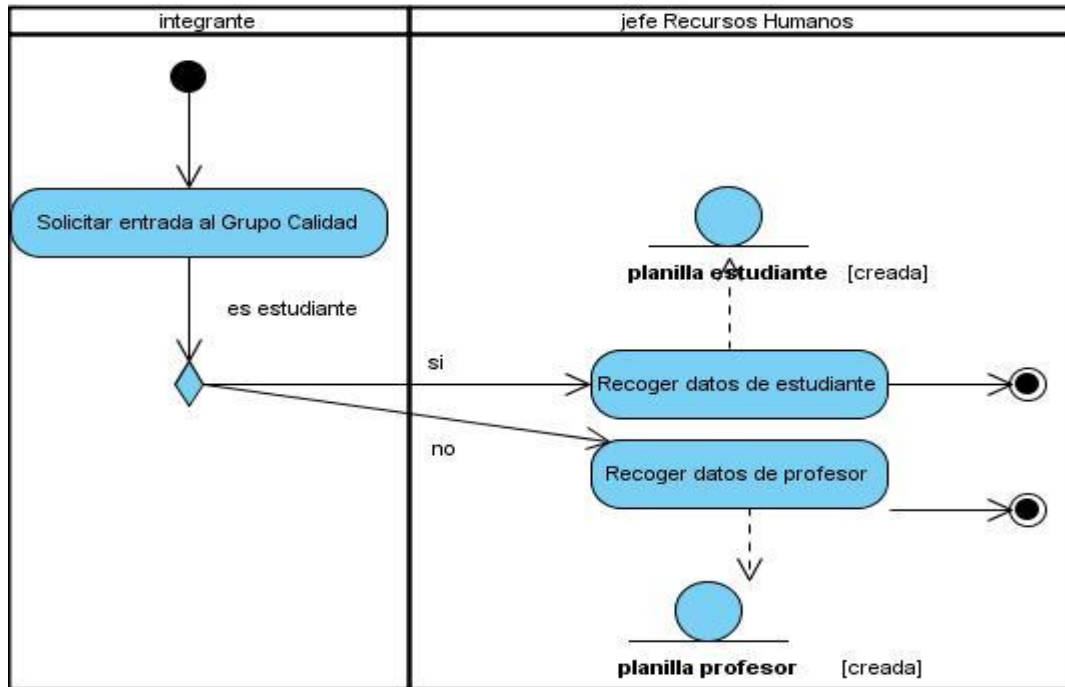
Un diagrama de casos de uso del negocio representa gráficamente los procesos de negocio y su interacción con los actores del negocio. A continuación se muestran los casos de uso que fueron identificados en el negocio.



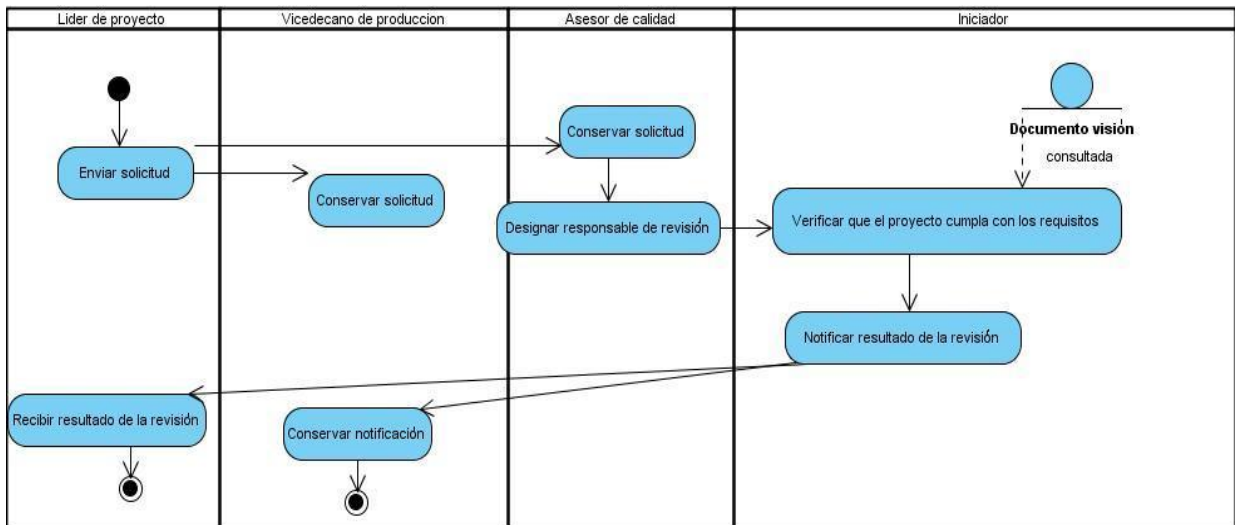
Figura 2 Modelo de Caso de Uso del Negocio (Elaboración propia)



### 2.2.4. Diagrama de actividades para cada Caso de Uso del Negocio



**Figura 3 Diagrama de actividades del CUN Insertar Integrante (Elaboración propia)**



**Figura 4 Diagrama de actividades del CUN Solicitar Certificación del Proyecto (Elaboración propia)**

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

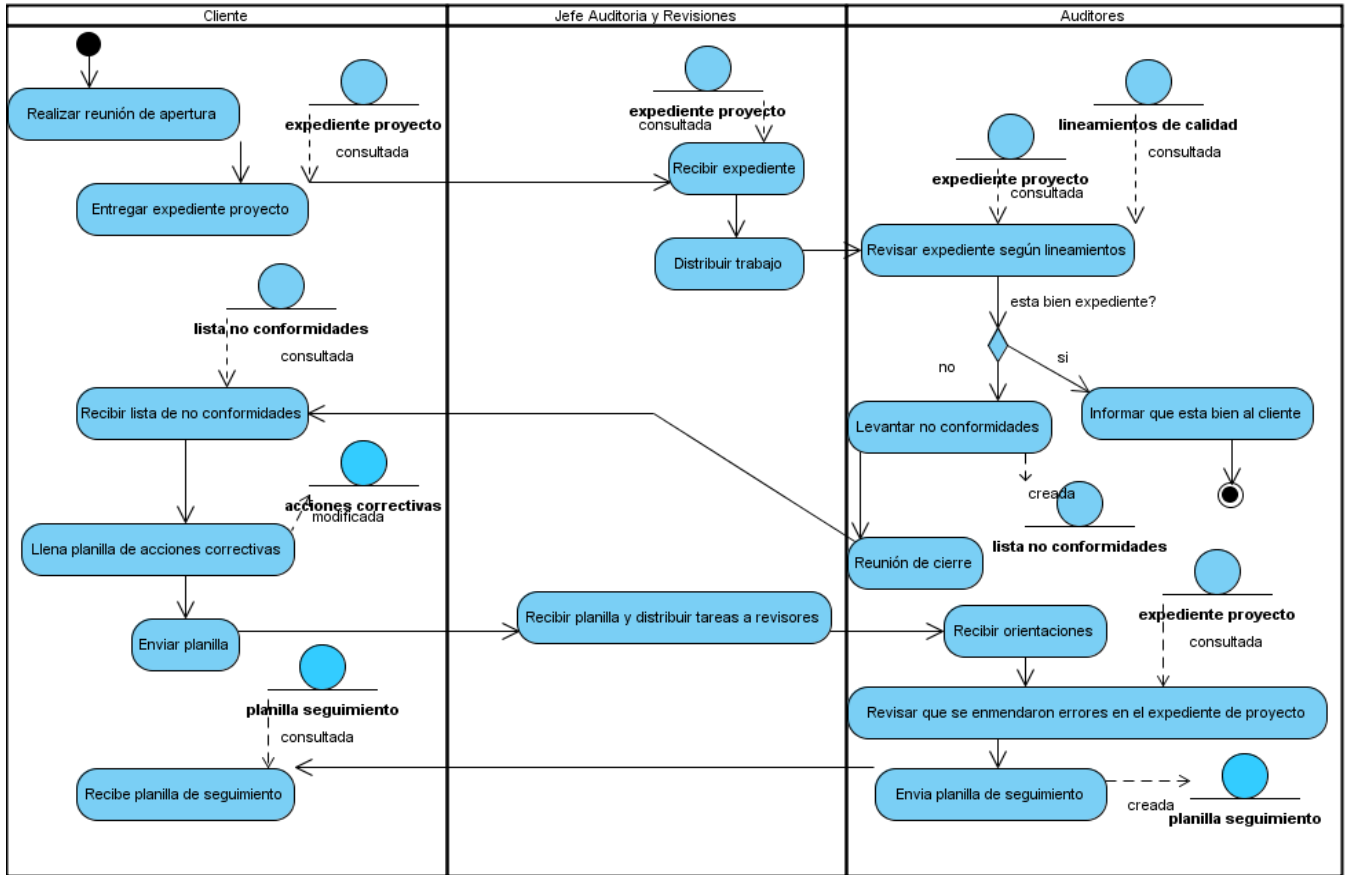


Figura 5 Diagrama de actividades del CUN Realizar Auditorías y Revisiones (Elaboración propia)

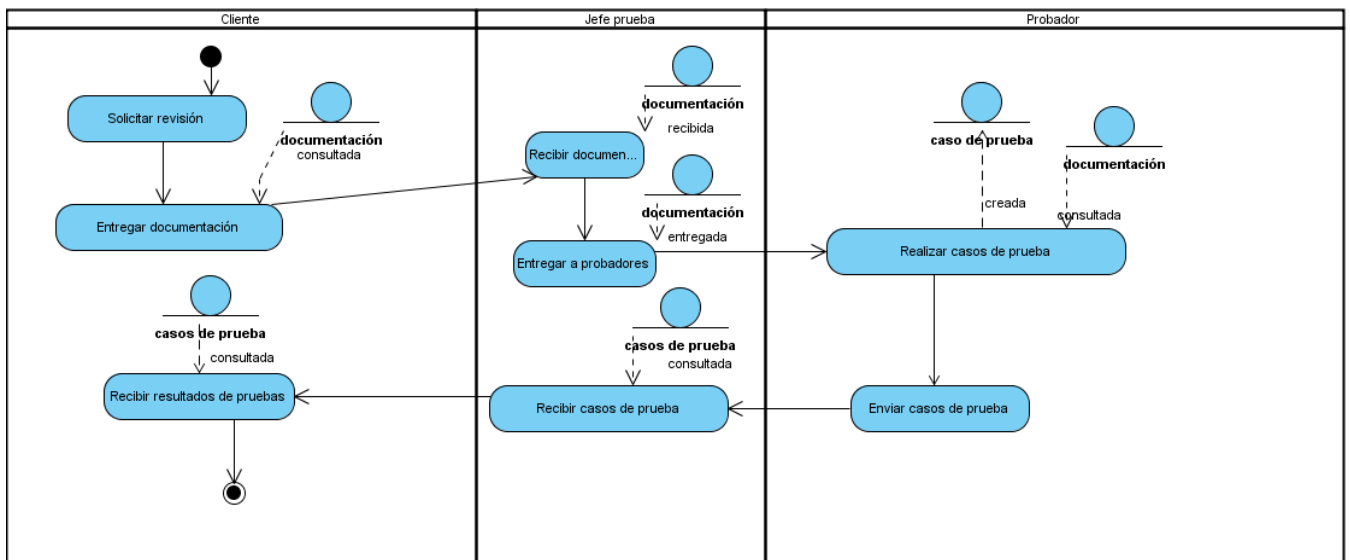


Figura 6 Diagrama de actividades del CUN Realizar Pruebas (Elaboración propia)

### 2.2.5. Modelo de objeto

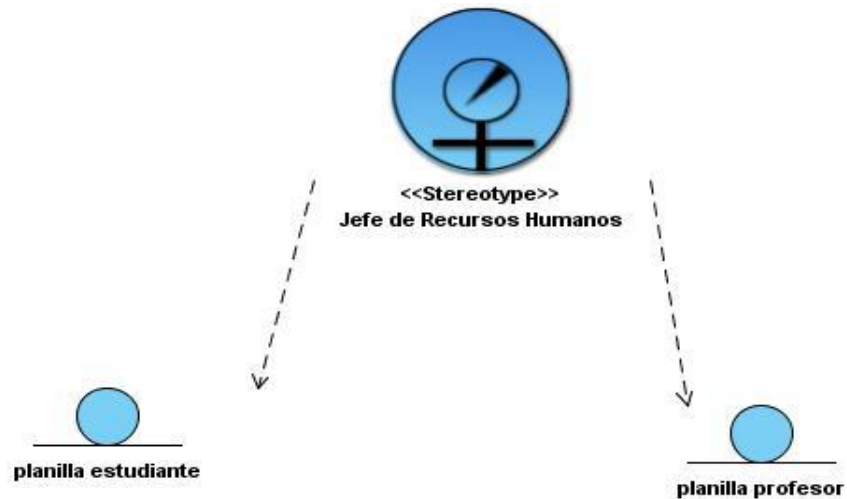


Figura 7 Insertar Integrante (Elaboración propia)

### 2.2.6. Descripción de los casos de uso del Negocio

<b>CUN</b>	<b>Insertar integrante</b>
<b>Actor</b>	<b>Integrante</b>
<b>Trabajador</b>	<b>Jefe de recursos humanos</b>
<b>Resumen</b>	Cuando llega un nuevo integrante al Grupo Calidad es preciso registrar sus datos en una planilla ya sea de profesor o estudiante.
<b>Precondiciones</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1. Solicita entrar al Grupo Calidad	2. Si es estudiante el jefe de recursos humanos llena una planilla de estudiante con los datos del nuevo integrante, en caso de ser profesor llena la planilla para profesores.

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

	3.Guarda planilla
<b>Poscondiciones</b>	
<b>Mejoras propuestas</b>	Automatizar este proceso creando una aplicación.
<b>Prioridad</b>	Crítico

**Tabla 10 Descripción del CUN Insertar integrante (Elaboración propia)**

<b>CUN</b>	<b>Realizar auditorías y revisiones</b>
<b>Actor</b>	<b>Cliente</b>
<b>Trabajador</b>	<b>Jefe de auditorías y revisiones, auditores</b>
<b>Resumen</b>	Se le realiza una auditoría a un proyecto, para esto lo primero es hacer una reunión de apertura para coordinar la auditoría y entregarle al jefe de auditorías y revisiones el expediente de proyecto, los auditores revisan todo según los lineamientos de calidad y en la reunión de cierre se le da al proyecto el informe final con la lista de no conformidades.
<b>Precondiciones</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1. Realizar reunión de apertura.	
2. Entregar expediente de proyecto o darle acceso al repositorio al jefe de auditorías y revisiones.	3. El jefe de auditorías y revisiones consulta la información y distribuye las tareas.
	4. Los auditores verifican que el expediente de proyecto cumpla con los lineamientos de

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

	calidad.
	5. Se hace un informe final con las no conformidades encontradas en la auditoría.
	6. Se hace una reunión de cierre donde se entrega el informe final con el objetivo de que sean erradicados.
7. Recibe y consulta el informe final.	
<b>Poscondiciones</b>	
<b>Mejoras propuestas</b>	Automatizar este proceso creando una aplicación.
<b>Prioridad</b>	Crítico

**Tabla 11 Descripción del CUN Realizar auditorías y revisiones (Elaboración propia)**

<b>CUN</b>	<b>Realizar pruebas</b>
<b>Actor</b>	<b>Cliente</b>
<b>Trabajador</b>	<b>Jefe de prueba, probador</b>
<b>Resumen</b>	Se realiza cuando un proyecto solicita que le realicen pruebas. El proyecto en cuestión entrega al jefe de prueba la documentación referente a la descripción de los casos de uso y los probadores realizan los casos de prueba.
<b>Precondiciones</b>	
<b>Acción del actor</b>	<b>Respuesta del negocio</b>
1. Solicita pruebas.	

2. Entrega documentación.	3.El jefe de Pruebas consulta la información y distribuye las tareas.
	4.Los probadores realizan los casos de prueba y los envían al jefe de pruebas.
	5.El jefe de pruebas recibe la información y le comunica al jefe del proyecto los resultados de las pruebas.
6. Recibe resultados.	
<b>Poscondiciones</b>	
<b>Mejoras propuestas</b>	Automatizar este proceso creando una aplicación.
<b>Prioridad</b>	Crítico

Tabla 12 Descripción del CUN Realizar pruebas (Elaboración propia)

### 2.3 Requisitos Funcionales

Los requisitos funcionales no son más que las condiciones o capacidades que el sistema debe cumplir.

#### RF1. Autenticar usuario

#### RF2. Gestionar estudiante

RF2.1 Adicionar nuevo estudiante.

RF2.2 Modificar datos de un estudiante

RF2.3 Eliminar estudiante

RF2.4 Mostrar estudiante

#### RF3. Gestionar profesor

RF3.1 Adicionar nuevo profesor

RF3.2 Modificar datos de un profesor

**RF3.3** Eliminar profesor

**RF3.4** Mostrar datos de profesor

**RF4. Gestionar computadora**

**RF4.1** Adicionar nuevo computadora

**RF4.2** Modificar datos de una computadora

**RF4.3** Eliminar computadora

**RF4.4** Mostrar computadora

**RF5. Mostrar reporte:** El sistema debe mostrar un reporte de todos los estudiantes, profesores y computadoras del Grupo Calidad.

**RF6 Gestionar planilla de evaluación de procedimiento de inicio**

**RF6.1** Adicionar proyecto certificado (si el proyecto cumple con todo lo verificado y obtiene evaluación de certificado, el sistema dará la opción de adicionar el proyecto a la lista de los proyectos certificados).

**RF7** Buscador

**RF8** Gestionar caso de prueba.

**RF10** Gestionar paquete de revisiones.

**RF10.1** Modificar planilla Acciones correctivas.

**RF10.2** Modificar acta de reunión de apertura.

**RF10.3** Modificar acta de reunión de cierre.

**RF10.4** Gestionar Evaluación de desempeño de revisores (tiene la opción de adicionar más de una planilla de este tipo porque es una para cada revisor que participó en la revisión).

**RF10.5** Modificar Informe final.

**RF10.6** Mostrar lista de chequeo 1.0.

**RF10.7** Mostrar lista de chequeo 2.0.

**RF10.8** Gestionar paquete de seguimiento.

**RF10.8.1** Modificar registro de evaluaciones del proyecto.

**RF10.8.2** Modificar registro de no conformidades.

**RF10.8.3** Modificar seguimiento a las no conformidades.

**RF10.8.4** Gestionar plan de evaluaciones del proyecto (modificar y eliminar).

**RF10.8.5** Modificar registro de evaluaciones.

### 2.4 Requisitos no funcionales (RNF)

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener.

#### RNF de Software

Las computadoras que utilizan el software deben tener instalado:

- Un sistema operativo Windows 2000 o superior.
- Como navegador el Internet Explorer o Mozilla Firefox.

#### RNF de Hardware

- Las computadoras que utilizarán el software a desarrollar deberán tener 256 MB de memoria RAM como mínimo.

#### RNF Requerimientos de apariencia o interfaz externa

El sistema debe tener diseño perfectamente encuadrado para resoluciones de 800 x 600, pero preparado para verse en otras resoluciones.

#### RNF Requerimientos de Seguridad

Al sistema se accederá a través de la autenticación convencional: usuario y contraseña.

Cada usuario debe tener solo los permisos necesarios para realizar las operaciones que le sean permitidas.

Disponibilidad: El sistema deberá tener un 100% de disponibilidad por lo que podrá ser usado las 24 horas del día.



### RNF Requerimientos de Usabilidad

El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y de un ambiente Web en sentido general.

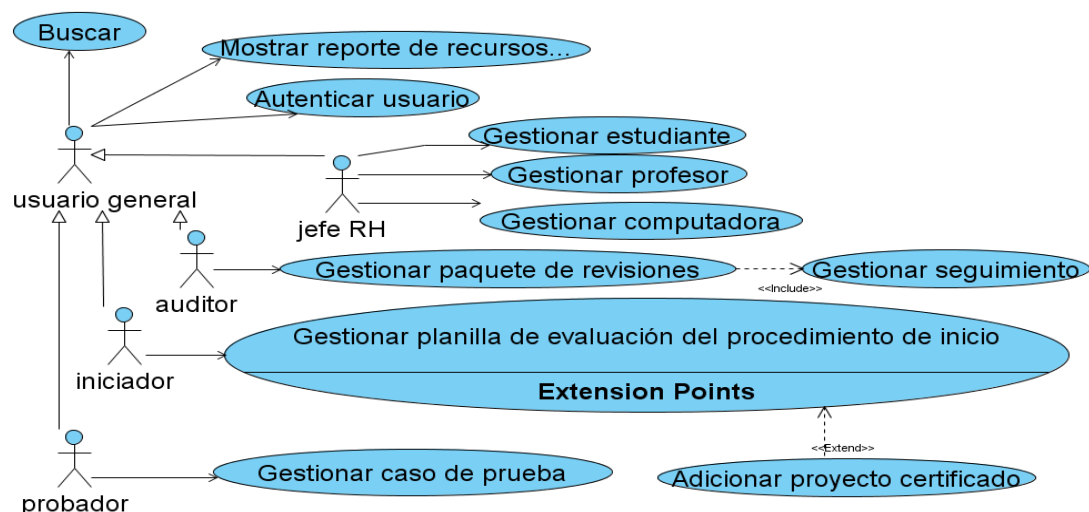
## 2.5 Descripción del sistema propuesto

### 2.5.1 Descripción de los actores

Actor	Descripción
Usuario general	Es todo usuario que independientemente de su responsabilidad después de autenticarse podrá ver información de cualquiera de los módulos del portal y visitar la sección, reporte de recursos humanos.
Auditor	Es el encargado de gestionar una lista de no conformidades en dependencia de los errores que encuentre cuando revise el expediente de un proyecto determinado.
Probador	Es el actor encargado de gestionar la planilla de prueba y diseñar casos de prueba.
Iniciador	Es el actor encargado de verificar que un proyecto puede ser certificado o no y en caso de que al final de la evaluación resulte certificado el actor podrá adicionar el proyecto a la lista de proyectos certificados.
Jefe de Recursos Humanos	Es el encargado de gestionar la información referente a los estudiantes, profesores y computadoras del Grupo Calidad Facultad 9.

**Tabla 13 Actores del sistema (Elaboración propia)**

### 2.5.2 Diagrama de Caso de Uso del Sistema



### 2.5.3 Describir cada Caso de Uso del Sistema

<b>Caso de Uso:</b>	Gestionar estudiante.	
<b>Actores:</b>	Jefe de Recursos Humanos	
<b>Resumen:</b>	El caso consiste en que el jefe de recursos humanos puede adicionar un nuevo estudiante, eliminar uno ya existente o modificar datos de un estudiante ya existente.	
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.	
<b>Referencias</b>	RF2, RF2.1, RF2.2, RF2.3, RF2.3	
<b>Prioridad</b>	Alta	
<b>Nivel</b>		
<b>Flujo Normal de Eventos</b>		
1. El caso de uso se inicia cuando el usuario accede al módulo de <b>Recursos Humanos</b> .	2. Se muestra una ventana que permite gestionar estudiante, profesor o computadora	
3. Selecciona la opción “Gestionar estudiante”.	4. El sistema muestra las opciones insertar, eliminar, modificar y mostrar estudiante.	
5. El usuario selecciona la opción: Insertar(Ir a la sección “Insertar nuevo estudiante”) Eliminar(Ir a la sección “Eliminar estudiante”) Modificar(Ir a la sección “Modificar datos de estudiante”) Mostrar(Ir a la sección “Mostrar datos de estudiante”)		

Sección "Insertar nuevo estudiante"	
Acción del Actor	Respuesta del Sistema
	<p>1. El sistema muestra una nueva ventana con una lista de campos que debe llenar, que son:</p> <ul style="list-style-type: none"><li>• Año que cursa.</li><li>• Grupo.</li><li>• Apartamento.</li><li>• Teléfono.</li><li>• Tema de Investigación.</li><li>• Tutor.</li><li>• Evaluación de cortes.</li><li>• Trabajos realizados.</li><li>• Trabajos publicados.</li><li>• Nota final de PP.</li><li>• Área de trabajo.</li></ul>
<p>2. El usuario llena los campos y hace clic en el botón "Adicionar".</p>	<p>3. El sistema verifica que los datos estén correctos.</p>
	<p>4. El sistema guarda los datos.</p>

### Prototipo de interfaz

#### Módulo de Recursos Humanos.

+ **Formulario Insertar Estudiante.**

Nombre	<input type="text" value="nombre"/>	1er Apellido	<input type="text" value="apellido"/>
2do Apellido	<input type="text" value="apellido"/>	Solapín	<input type="text" value="solapin"/>
Carnet de Identidad	<input type="text" value="carnet"/>	Grupo	<input type="text" value="grupo"/>
Apto	<input type="text" value="apto"/>	Año	<input type="text" value="año"/>
Teléfono	<input type="text" value="teléfono"/>	Correo	<input type="text" value="correo"/>

### Flujos Alternos

Acción del Actor	Respuesta del sistema
	3.1. Si los datos no están correctos el sistema muestra un mensaje de error y vuelve a la acción 1 del flujo normal de los eventos.
Sección “Eliminar estudiante”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una nueva ventana con una lista de todos los estudiantes.
2. El usuario selecciona al estudiante que desea eliminar y presiona el botón “ <b>Aceptar</b> ”.	3. El sistema verifica que se seleccionó al menos un estudiante.
	4. Elimina al estudiante de la lista.

### Prototipo de interfaz

**Módulo de Recursos Humanos.**

➔ **Formulario Eliminar Estudiante.**

Escoger estudiante

Yosniel Gongora Benitez ▼

Aceptar

### Flujos Alternos

Acción del Actor	Respuesta del Sistema
	3.1. Si no se seleccionó al menos un estudiante, el sistema muestra un mensaje de error y vuelve a la acción 2 del flujo normal de los eventos.
Sección “Modificar datos de estudiante”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una nueva ventana con una lista de todos los estudiantes.
2. El usuario selecciona el estudiante a modificar y da clic en el botón “ <b>Modificar</b> ”.	3. El sistema verifica que se seleccionó un estudiante.
	4. El sistema muestra una ventana con todos los campos del estudiante. <ul style="list-style-type: none"> <li>Año que cursa.</li> <li>Grupo.</li> <li>Apartamento.</li> <li>Teléfono.</li> </ul>

	<ul style="list-style-type: none"> <li>Tema de Investigación.</li> <li>Tutor.</li> <li>Evaluación de cortes.</li> <li>Trabajos realizados.</li> <li>Trabajos publicados.</li> <li>Nota final de PP.</li> <li>Área de trabajo.</li> </ul>
5. El usuario realiza los cambios y da clic en el botón “Aceptar”.	6. El sistema guarda los cambios.

### Prototipo de interfaz

#### Módulo de Recursos Humanos.

+ **Formulario Actualizar Estudiante.**

Nombre	<input type="text"/>	1er Apellido	<input type="text"/>
2do Apellido	<input type="text"/>	Solapín	<input type="text"/>
Carnet de Identidad	<input type="text"/>	Grupo	<input type="text"/>
Apto	<input type="text"/>	Año	<input type="text"/>
Teléfono	<input type="text"/>	Correo	<input type="text"/>

### Flujos Alternos

Acción del Actor	Respuesta del Sistema
	3.1. Si no se seleccionó un estudiante el sistema muestra un mensaje de error y vuelve

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

	a la acción 2 del flujo normal de los eventos.
<b>Sección “Mostrar datos de estudiante”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra una nueva ventana con una lista de todos los estudiantes.
2. Selecciona el estudiante que desea visitar y da clic en mostrar.	3. El sistema muestra una ventana con todos los datos del estudiante.

**Tabla 14 CUS Gestionar estudiante (Elaboración propia)**

<b>Caso de Uso:</b>	Mostrar reporte.
<b>Actores:</b>	Usuario (Inicia el caso de uso).
<b>Resumen:</b>	Cualquier usuario que se registre en la aplicación puede seleccionar la opción de ver reporte donde podrá escoger reporte de estudiantes del Grupo Calidad, reporte de profesores del Grupo Calidad o computadoras del laboratorio del Grupo Calidad.
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.
<b>Referencias</b>	RF2
<b>Prioridad</b>	Media
<b>Nivel</b>	
<b>Flujo Normal de Eventos</b>	
1. El caso de uso se inicia cuando el usuario accede al módulo de <b>Recursos Humanos</b> .	2. Se muestra una ventana con tres opciones de reporte de estudiantes, profesores y

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

	computadoras.
<p>3. Selecciona la opción:</p> <p>Reporte de estudiantes(Ir a sección <b>“Mostrar reporte de estudiantes”</b>)</p> <p>Reporte de profesores(Ir a sección <b>“Mostrar reporte de profesores”</b>)</p> <p>Reporte de computadoras(Ir a sección <b>“Mostrar reporte de computadoras”</b>)</p>	
<b>Sección “Mostrar reporte de estudiantes”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra una ventana con todos los estudiantes del proyecto con datos como nombre, apellido y área a la que pertenece.
<b>Sección “Mostrar reporte de profesores”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra una ventana con todos los profesores del proyecto con datos como nombre, apellido y responsabilidad.
<b>Sección “Mostrar reporte de computadoras”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. El sistema muestra una ventana con todas las computadoras del proyecto con datos como número e integrantes de la computadora.

**Tabla 15 CUS Mostrar reporte (Elaboración propia)**



## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

<b>Caso de Uso:</b>	Crear planilla de evaluación de procedimiento de inicio	
<b>Actores:</b>	Iniciador (son los que pertenecen al área de procedimiento de inicio de proyecto).	
<b>Resumen:</b>	El caso consiste en que el iniciador puede crear, eliminar o actualizar una planilla de evaluación de procedimiento de inicio.	
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.	
<b>Referencias</b>	RF6	
<b>Prioridad</b>	Alta	
<b>Nivel</b>		
<b>Flujo Normal de Eventos</b>		
1. El caso de uso se inicia cuando el iniciador accede al módulo de <b>Procedimiento de inicio de proyecto</b> .	2. Se muestra la opción de crear, eliminar o actualizar una <b>planilla de evaluación de procedimiento de inicio</b> .	
3. El usuario selecciona la opción Crear(Ir a la sección " <b>Crear planilla de evaluación de procedimiento de inicio</b> ") Eliminar(Ir a la sección " <b>Eliminar planilla</b> ") Actualizar(Ir a la sección " <b>Actualizar planilla</b> ")		
<b>Sección "Crear planilla de evaluación de procedimiento de inicio"</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
.	1. El sistema muestra una ventana pidiendo	

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

---

	datos como, fecha actual, versión, nombre del usuario y nombre del proyecto.
2. El usuario llena los campos y presiona el botón <b>siguiente</b> .	<p>3. El sistema muestra un conjunto de preguntas con su combobox correspondiente donde podrán seleccionar la opción si o no.</p> <p>Condiciones/Normativas.</p> <ul style="list-style-type: none"><li>• Metodología(si , no) si marco si dar la opción de marcar que metodología utilizó.</li><li>• Plan de capacitación.(si , no).</li></ul> <p>Artefactos</p> <ul style="list-style-type: none"><li>• Cronograma del proyecto. (si, no) en caso de si, saldrá la opción de marcar en que iteración se hizo.</li><li>• Lista de riesgos. (si, no).</li><li>• Plan de Mitigación de riesgos. (si, no).</li><li>• Plan de contingencia. (si, no).</li><li>• Documento visión. (si, no).</li><li>• Herramientas para la gestión de riesgos (dar la posibilidad de que el usuario entre las herramientas que se usan).</li><li>• Las actividades de evaluación de los riesgos y las de gestión de riesgos están incluidas en el cronograma de desarrollo con un nivel de prioridad alto (si, no).</li></ul>

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

	<ul style="list-style-type: none"><li>• El proyecto cumplió con el plazo de 1 mes para hacer la solicitud de certificación. (si, no)</li></ul> <p>Valoración económica</p> <ul style="list-style-type: none"><li>• ¿Está definido un rol económico?(si, no)</li><li>• Precio del producto(entrar precio estimado)</li><li>• ¿Se registraron los costos por concepto de materias primas y materiales que requiere el proyecto?(si, no)</li><li>• ¿Se definió la técnica de estimación a utilizar? (si, no)</li></ul> <p>Tiene como último combobox <b>Emitir evaluación</b> con las opciones de certificado o no certificado.</p> <p>Si la opción marcada es la de certificado la aplicación da la opción de Adicionar el proyecto a la lista de proyectos certificados.</p>
4. El usuario hace clic en el botón “ <b>Adicionar</b> ”.	5. El sistema verifica que los datos estén correctos.
	6. El sistema guarda los datos.
<b>Prototipo de interfaz</b>	

**Formulario crear planilla**

Nombre del autor

Nombre del proyecto

Versión

Fecha de asignados los recursos

Metodología

Plan de capacitación

Cronograma del proyecto

Lista de riesgos

Plan de mitigación de riesgos

Plan de contingencia

Documento visión

Herramientas para gestión de riesgos

Activ. de eval. y gestión de riesgos en cronograma (alto)

Cumplido el plazo

Definido un rol económico

Precio del producto

Registrados costos de mat. primas y materiales

Definida la técnica de estimación a utilizar

Certificado

### Flujos Alternos

Acción del Actor	Respuesta del Sistema
	5.1. Si los datos son incorrectos el sistema muestra un mensaje de error y vuelve a la acción 1 del flujo normal de los eventos.

### Sección “Eliminar planilla de evaluación de procedimiento de inicio”

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una ventana con una lista de las planillas.
2. Selecciona la planilla que desea eliminar y presiona el botón “Eliminar planilla”.	3. El sistema eliminar la planilla.

### Prototipo de interfaz


**Eliminar**

Nombre del proyecto  Calidad UCI

Calidad UCI

Sección “Actualizar planilla de evaluación de procedimiento de inicio”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una ventana con una lista de las planillas.
2. Selecciona la planilla que modificar y presiona el botón “Actualizar planilla”.	3. El sistema muestra los campos de la planilla.
4. El usuario realiza los cambios y presiona el botón guardar.	5. El sistema guarda los cambios.

**Prototipo de interfaz**



**Tabla 16 CUS Gestionar planilla de evaluación de procedimiento de inicio (Elaboración propia)**

<b>Caso de Uso:</b>	Gestionar caso de prueba
<b>Actores:</b>	Probador
<b>Resumen:</b>	Se inicia cuando el probador entra al módulo de prueba y selecciona la opción de gestionar caso de prueba.
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.
<b>Referencias</b>	RF9
<b>Prioridad</b>	Alta
<b>Nivel</b>	
<b>Flujo Normal de Eventos</b>	

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

<p>1. El caso de uso se inicia cuando el probador accede al módulo de <b>Prueba</b>.</p>	<p>2. Se muestra una ventana que permite gestionar casos de prueba y gestionar planilla de prueba.</p>
<p>3. Selecciona la opción Gestionar casos de prueba.</p>	<p>4. El sistema muestra las opciones de crear, eliminar, modificar y mostrar caso de prueba.</p>
<p>5. El usuario selecciona la opción:          Crear(Ir a la sección "<b>Crear caso de prueba</b>").          Eliminar(Ir a la sección "<b>Eliminar caso de prueba</b>")          Modificar(Ir a la sección "<b>Modificar caso de prueba</b>")          Mostrar(Ir a la sección "<b>Mostrar caso de prueba</b>")</p>	
<p><b>Sección "Crear caso de prueba"</b></p>	
<p><b>Acción del Actor</b></p>	<p><b>Respuesta del Sistema</b></p>
	<p>1. El sistema muestra una nueva ventana con gran cantidad de campos que deben ser llenados:</p> <p>Control de versiones</p> <ul style="list-style-type: none"> <li>• Fecha</li> <li>• Versión</li> <li>• Descripción</li> <li>• Autor</li> </ul> <p>Descripción General</p> <p>Condiciones de Ejecución</p>

	<p>Secciones a probar en el caso de uso</p> <ul style="list-style-type: none"><li>• Nombre de la sección</li><li>• Escenarios de la sección</li><li>• Descripción de la funcionalidad</li><li>• Flujo central</li></ul> <p>Descripción de las variables</p> <ul style="list-style-type: none"><li>• Número</li><li>• Nombre de campo</li><li>• Clasificación</li><li>• Puede ser nulo</li><li>• Descripción</li></ul> <p>Secciones</p> <ul style="list-style-type: none"><li>• Id del escenario</li><li>• Escenario</li><li>• Variables</li><li>• Respuesta del sistema</li><li>• Resultado de la prueba</li></ul> <p>Registro de defecto y dificultades encontradas</p> <ul style="list-style-type: none"><li>• Elemento</li><li>• Número</li><li>• No conformidad</li><li>• Aspecto correspondiente</li><li>• Etapa de detección</li><li>• Significativa o no significativa</li></ul>
--	---

	<ul style="list-style-type: none"><li>• Recomendación</li><li>• Estado NC</li><li>• Respuesta del equipo de desarrollo</li></ul>
2. El usuario llena los campos y hace clic en el botón “Adicionar”.	3. El sistema verifica que los datos estén completos y correctos.
	4. El sistema guarda los datos.

### Prototipo de interfaz

Crear caso de prueba.

Nombre Proyecto: Calidad

Versión:

Autor: 10001 - pepe LK

Descripción:

Fecha:

Descripción general:

Condiciones de ejecución:

Crear caso de prueba

Cancelar

Caso de prueba.

Caso de prueba versión 1.0 del proyecto Calidad.

Modificar caso de prueba

Características iniciales | Secciones a probar | Descripción variables | No conformidades

Control de versiones.

Fecha	Versión	Descripción	Autor
11/05/2010	1.0	qw	pepe LK

Descripción General.

qw

Condiciones de Ejecución.

qw

### Flujos Alternos



Acción del Actor	Respuesta del Sistema
	3.1 Si los datos son incorrectos el sistema muestra un mensaje de error y vuelve a la acción 1 del flujo normal de los eventos.

### Sección “Eliminar caso de prueba”

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una nueva ventana con un listado de todos los casos de prueba.
2. El usuario selecciona el caso de prueba que desea eliminar y presiona el botón “ <b>Eliminar</b> ”.	3. El sistema verifica que exista al menos un caso de prueba seleccionado.
	4. El sistema elimina el caso de prueba.

### Prototipo de interfaz

El prototipo muestra una ventana con el título "Eliminar Caso de Prueba". Dentro de la ventana, hay un campo de texto etiquetado "Nombre Del Formulario". Debajo de esto, hay dos campos de entrada: "Nombre del proyecto" y "Versión de la prueba". El campo "Versión de la prueba" contiene el valor "1.0" y tiene un menú desplegable. Debajo de estos campos, hay un botón etiquetado "Eliminar".

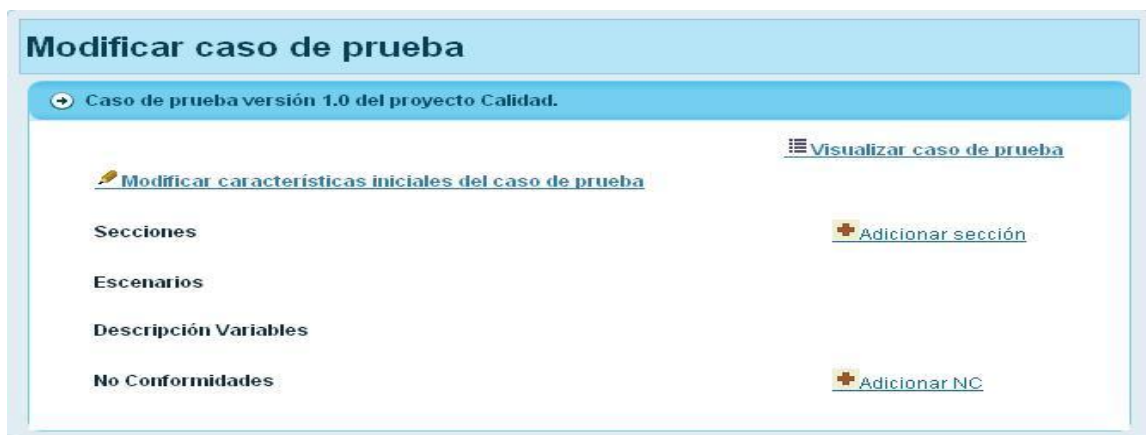
### Flujos Alternos

Acción del Actor	Respuesta del Sistema
	3.1 Si no se ha seleccionado ningún caso de prueba el sistema muestra un mensaje de error y vuelve a la acción 1 del flujo normal de los eventos.

### Sección “Modificar caso de prueba”

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una nueva ventana con un listado de todos los casos de prueba.
2. El usuario selecciona el caso de prueba que desea modificar.	3. El sistema muestra el caso de prueba en una ventana.
4. El usuario hace los cambios y presiona el botón <b>“Aceptar”</b> .	5. El sistema guarda los cambios.

### Prototipo de interfaz



### Sección “Mostrar caso de prueba”

Acción del Actor	Respuesta del Sistema
	1. El sistema muestra una lista con todos los casos de prueba.
2. Selecciona el caso de prueba que desea visitar y da clic en <b>“Mostrar”</b> .	3. El sistema muestra una ventana con los datos del caso de prueba.

**Tabla 17 CUS Gestionar caso de prueba (Elaboración propia)**

<b>Caso de Uso:</b>	Gestionar paquete de revisiones
---------------------	---------------------------------

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

<b>Actores:</b>	Auditor
<b>Resumen:</b>	El caso consiste en que un auditor tiene los permisos para insertar, eliminar y modificar un paquete de revisiones y a su vez puede modificar, adicionar o eliminar las planillas que se generan dentro del paquete.
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.
<b>Referencias</b>	RF10
<b>Prioridad</b>	Alta
<b>Nivel</b>	
<b>Flujo Normal de Eventos</b>	
1. El caso de uso se inicia cuando el usuario accede al módulo de <b>Revisiones</b> .	2. Se muestra una ventana con una lista de todos los paquetes de revisiones existentes hasta el momento, cada uno tiene la opción de ser eliminado, modificado o visualizado, además se muestra un botón que permite insertar un nuevo paquete.
3. Selecciona la opción:  Insertar(Ir a la sección “ <b>Insertar paquete de revisiones</b> ”)  Eliminar(Ir a la sección “ <b>Eliminar paquete de revisiones</b> ”)  Modificar(Ir a la sección “ <b>Modificar paquete de revisiones</b> ”)  Mostrar(Ir a la sección “ <b>Mostrar paquete de revisiones</b> ”)	

Sección "Insertar paquete de revisiones"	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción insertar paquete de revisiones.	2. Se muestra una ventana donde debes introducir el nombre del proyecto al que se le va a realizar la revisión.
3. El auditor llena los datos y da clic en el botón adicionar.	4. El sistema muestra una ventana para insertar todas las planillas pertenecientes a revisiones que se resumen en: <ul style="list-style-type: none"><li>• Acciones correctivas</li><li>• Acta de reunión de apertura</li><li>• Acta de reunión de cierre</li><li>• Evaluación de desempeño de revisores (tiene la opción de adicionar más de una planilla de este tipo porque es una para cada revisor que participó en la revisión).</li><li>• Informe final</li><li>• Lista de chequeo 1.0.</li><li>• Lista de chequeo 2.0.</li><li>• Paquete de planillas de seguimiento.</li></ul>

### Prototipo de interfaz

**Módulo de Auditoría y Revisiones.**

➤ **Formulario Insertar Paquete de Revisiones.**

Nombre del Proyecto.

Adicionar

+ **Formulario Insertar Planillas de Revisiones.**

Acciones Correctivas.	<input type="text" value="Acciones Correctivas"/>
Nombre del Proyecto	<input type="text"/>
	<input type="button" value="Aceptar"/>
Acta de reunión de apertura.	<input type="text" value="Acta de Reunión de Ape"/>
Nombre del Proyecto	<input type="text"/>
	<input type="button" value="Aceptar"/>
Acta de reunión de cierre.	<input type="text" value="Acta de Reunión de Cie"/>
Nombre del Proyecto	<input type="text"/>
	<input type="button" value="Aceptar"/>
Implementar para agregar una plantilla por revisor.	
Evaluación de Desempeño de revisores.	<input type="text" value="Evaluación de desemp"/>
Nombre del Proyecto	<input type="text"/>
	<input type="button" value="Aceptar"/>

### Flujos Alternos

Acción del Actor	Respuesta del Sistema
	2.1. Si los datos son incorrectos el sistema muestra un mensaje de error y vuelve a la acción 1 del flujo normal de los eventos.

### Sección “Eliminar paquete de revisiones”

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción eliminar que se encuentra al lado del paquete que desea eliminar.	2. Elimina el paquete.

Prototipo de interfaz

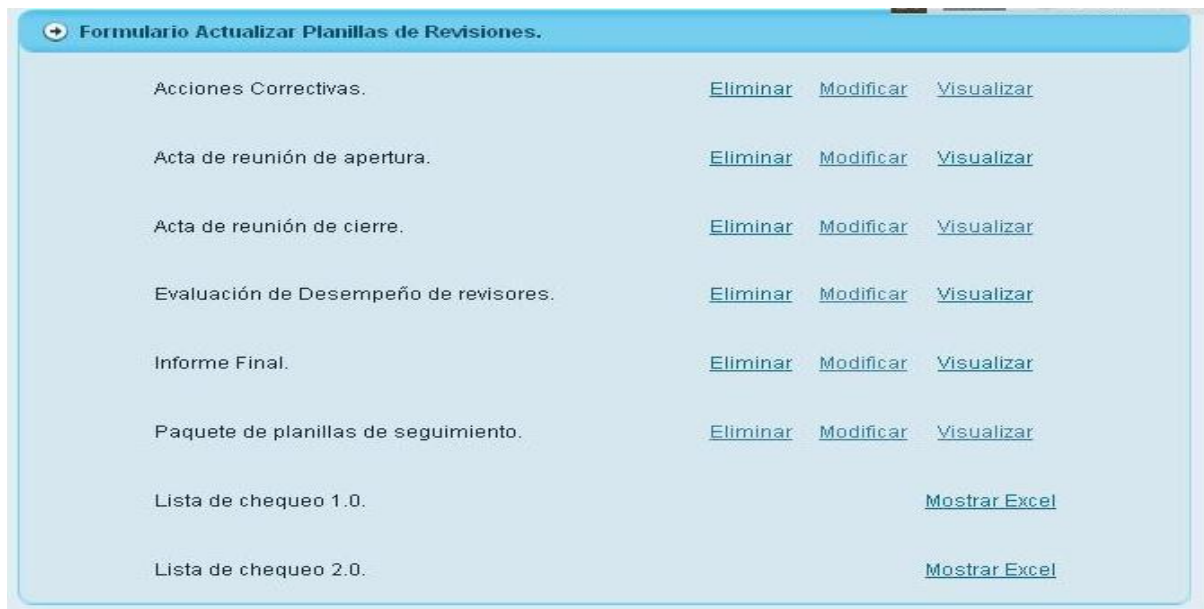


### Sección “Modificar paquete de revisiones”

Acción del Actor	Respuesta del Sistema
<p>1. Pulsa en el botón “modificar” del paquete deseado.</p>	<p>2. Se muestra una ventana con todas las planillas del paquete y las opciones que tiene cada planilla :</p> <ul style="list-style-type: none"> <li>Planilla acciones correctivas. Ver CU <b>Modificar Planilla acciones correctivas</b></li> <li>Acta de reunión de apertura. Ver CU <b>Modificar Acta de reunión de apertura</b></li> <li>Acta de reunión de cierre. Ver CU <b>Modificar Acta de reunión de cierre</b></li> <li>Evaluación de desempeño de revisores (tiene la opción de adicionar más de una planilla de este tipo porque es una para cada revisor que participó en la revisión). Ver CU <b>Gestionar Evaluación de desempeño de revisores</b></li> </ul>

- Informe final. Ver CU **Modificar Informe final**
- Lista de chequeo 1.0. Ver CU **Mostrar lista de chequeo 1.0**
- Lista de chequeo 2.0. Ver CU **Mostrar lista de chequeo 2.0**
- Paquete de planillas de seguimiento. Ver CU **Gestionar Paquete de planillas de seguimiento.**

### Prototipo de interfaz



### Sección “Mostrar paquete de revisiones”

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción visualizar que se encuentra al lado del paquete que desea ver.	2. El sistema muestra la lista de planillas con la opción de mostrar para cada planilla.

**Tabla 18 Gestionar paquete de revisiones (Elaboración propia)**

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

<b>Caso de Uso:</b>	<b>Gestionar paquete de seguimiento</b>
<b>Actores:</b>	Auditor
<b>Resumen:</b>	El caso consiste en que un auditor tiene los permisos para eliminar, visualizar y modificar las planillas que se encuentran en un paquete de seguimiento.
<b>Precondiciones:</b>	El sistema debe estar instalado y ejecutándose correctamente. El actor debe estar autenticado con los permisos necesarios.
<b>Referencias</b>	RF10.8
<b>Prioridad</b>	Alta
<b>Nivel</b>	
<b>Flujo Normal de Eventos</b>	
<b>Sección “Insertar paquete de seguimiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El caso de uso se inicia cuando el usuario adiciona un paquete de revisiones.	2. Se muestra una ventana que permite insertar cada planilla de revisiones y las planillas de seguimiento.
3. El usuario llena los datos que piden para cada planilla.	4. El sistema crea un nuevo paquete con el nombre del proyecto y lo pone al principio de la lista de paquetes. El paquete contiene un conjunto de planillas de auditoría y un paquete de planillas de seguimiento que contiene : <ul style="list-style-type: none"><li>• Registro de evaluaciones del proyecto.</li><li>• Registro de no conformidades</li><li>• Seguimiento a las no conformidades</li></ul>



- Plan de evaluaciones del proyecto
- Registro de evaluaciones

### Prototipo de interfaz

+ **Formulario Insertar Planillas de Seguimiento.**

Insertar registro de evaluaciones del proyecto.	Modificar registro de ev
Nombre del Proyecto	<input style="width: 90%;" type="text"/>
	<input type="button" value="Aceptar"/>
Insertar registro de no conformidades.	Modificar registro de nc
Nombre del Proyecto	<input style="width: 90%;" type="text"/>
	<input type="button" value="Aceptar"/>
Insertar seguimiento a las no conformidades.	Modificar seguimiento :
Nombre del Proyecto	<input style="width: 90%;" type="text"/>
	<input type="button" value="Aceptar"/>
Insertar plan de evaluaciones del proyecto (modificar y eliminar).	Gestionar plan de evalu
Nombre del Proyecto	<input style="width: 90%;" type="text"/>

### Flujos Alternos

**Acción del Actor**

**Respuesta del Sistema**

5.1 Si los datos son incorrectos el sistema muestra un mensaje de error y vuelve a la acción 4 del flujo normal de los eventos.

### Sección “Modificar paquete de seguimiento”

**Acción del Actor**

**Respuesta del Sistema**

## Capítulo 2: Negocio, Requisitos, Análisis, Diseño

---

<p>1. El caso de uso se inicia cuando el usuario accede al paquete de seguimiento y presiona la opción modificar.</p>	<p>2. Se muestra una ventana con todas las planillas de seguimiento con la opción de modificar y visualizar.</p>
<p>3. Pulsa en el botón modificar de la planilla deseada.</p>	<p>4. Permite modificar las planillas referentes a seguimiento.</p> <ul style="list-style-type: none"><li>• Registro de evaluaciones del proyecto. Ver <b>CU Modificar Registro de evaluaciones del proyecto</b></li><li>• Registro de no conformidades. Ver <b>CU Gestionar Registro de no conformidades</b></li><li>• Seguimiento a las no conformidades. Ver <b>CU Gestionar Seguimiento a las no conformidades</b></li><li>• Gestionar plan de evaluaciones del proyecto (modificar y eliminar). Ver <b>CU Gestionar plan de evaluaciones del proyecto</b></li><li>• Registro de evaluaciones. Ver <b>CU Gestionar Registro de evaluaciones</b></li></ul>

Tabla 19 CUS Gestionar paquete de seguimiento (Elaboración propia)

## 2.6. Análisis

### Diagrama de Clases del Análisis (DCA) para cada caso de uso del sistema

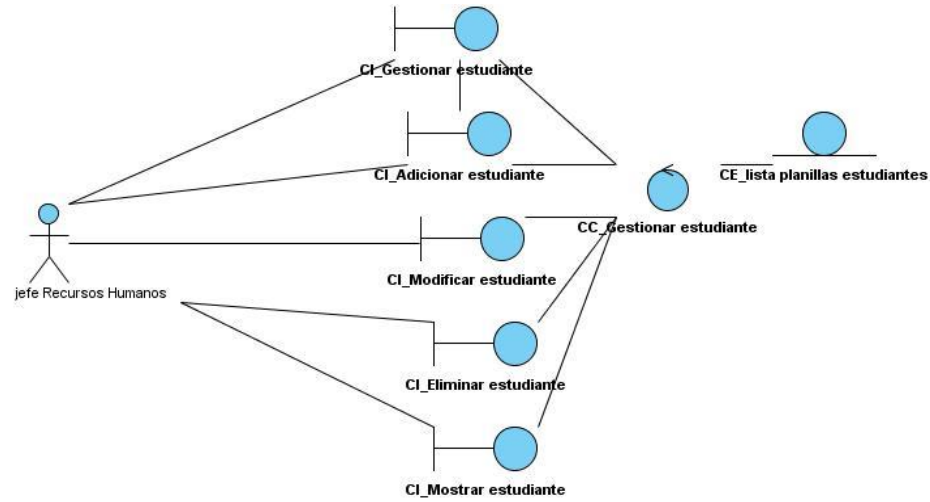


Figura 8 DCA Gestionar estudiante

### Diagrama de colaboración para cada caso de uso del sistema

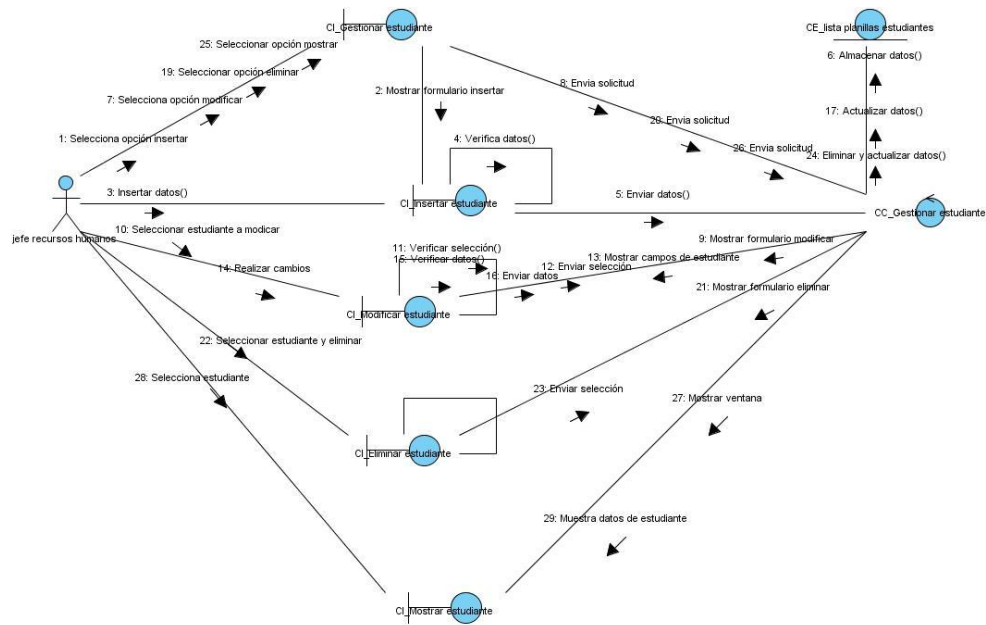


Figura 9 DC Gestionar estudiante

## 2.7. Diseño

Diagrama de clases del diseño para cada caso de uso del sistema

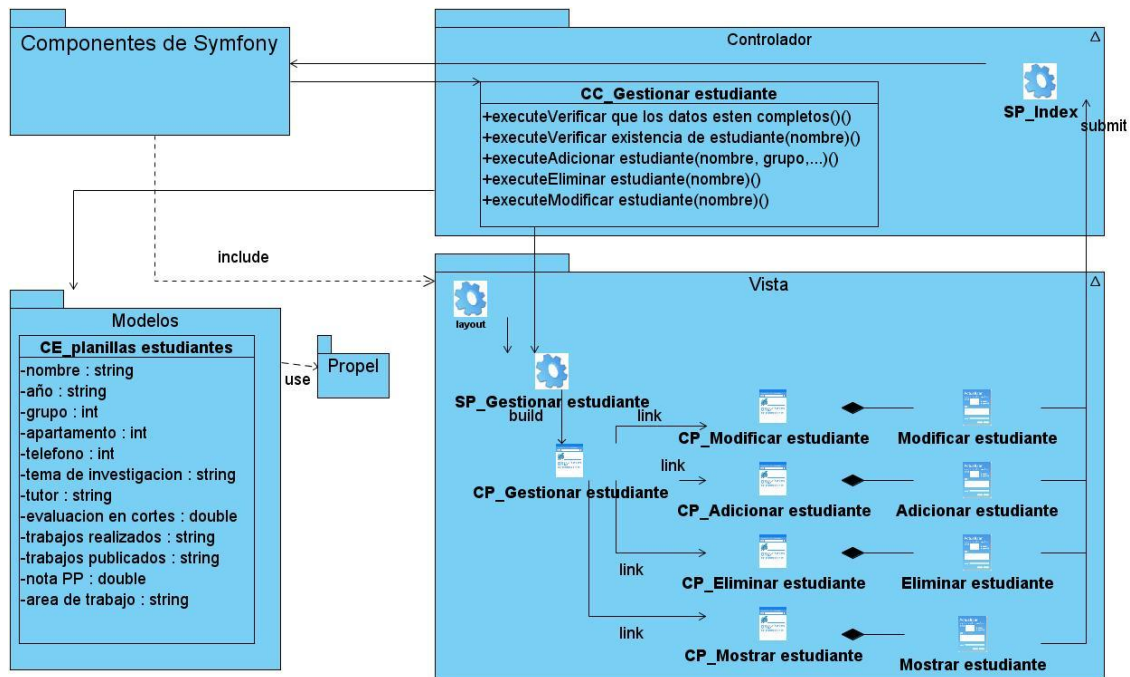


Figura 10 DCD Gestionar estudiante

## 2.8. Conclusiones

En el presente capítulo se ha desarrollado la propuesta de solución, obteniéndose a partir del análisis de los procesos de negocio, un grupo de funcionalidades que debe tener el sistema, representados mediante diagramas de casos de uso del análisis y diseño. De esta forma termina la propuesta de solución del Portal Calidad Facultad 9.

## CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

### 3.1. Introducción

La clave fundamental de éxito en la producción de un software es lograr la calidad requerida por el usuario. En este capítulo se emplean distintas métricas a los artefactos generados en el capítulo anterior que permitirán erradicar problemas en los requisitos o casos de uso del sistema.

### 3.2. Método de validación de los resultados

En toda investigación es imprescindible realizar el análisis de los resultados de la propuesta de solución al problema en cuestión, para de esta forma saber hasta que punto satisface realmente al favorecido. En la creación de un software también pasa lo mismo, se debe realizar la validación, la misma ayuda a saber al desarrollador si el sistema que ellos están construyendo es realmente lo que espera el cliente.

“Las métricas constituyen un método de evaluación de los productos y procesos de software, las cuales suelen ser aplicadas a muchas organizaciones, procesos y productos”. (13).

Se utilizaron varios métodos para la validación de manera que fuesen validados los artefactos generados durante la investigación.

### 3.3. Aplicación de la métrica de la calidad de especificación de los requisitos

“La métrica de la Calidad de Especificación de los Requisitos mide la especificidad de los requisitos, haciendo que la parte interesada pueda entenderlos de manera fácil y se puedan probar”. (14)

Primeramente se tiene **Nr** que representa el total de requisitos:

$$\mathbf{Nr = Nf + Nnf}$$

$$\mathbf{Nr = 23 + 12}$$

$$\mathbf{Nr = 35}$$

Donde **Nf** es el número de requisitos funcionales y **Nnf** es el número de requisitos no funcionales.

## Validación de la solución propuesta

---

Luego de tener este dato se puede determinar la especificación (ausencia de ambigüedad) y para ello se plantea:

$$Q = Nui / Nr$$

Donde **Nui** representa el número de requisitos para los que los revisores tuvieron interpretaciones idénticas, es importante resaltar que cuando más cerca de 1 este el valor **Q**, menos ambigüedad presentará la especificación.

En el presente trabajo se realizaron dos revisiones para llegar a obtener un análisis sin ambigüedad en los requisitos, y para que se reflejara con más claridad las necesidades del cliente, en las cuales se obtuvieron los siguientes resultados:

### Revisión 1:

Durante la revisión de un total de **35** requisitos, entre funcionales y no funcionales, se detectaron de los **23** requisitos funcionales dos que no estaban claramente especificados, es el caso de gestionar estudiante y mostrar reporte ya que los requisitos no expresaban con claridad su objetivo. Por lo que en todos los requisitos menos en 2, los revisores tuvieron interpretaciones bastante parecidas.

$$Q = Nui / Nr$$

$$Q = 33 / 35$$

$$Q = 0.94$$

### Revisión 2:

Durante esta revisión no se detectaron ambigüedades en los requisitos revisados, debido a la realización de un profundo análisis. Por lo que de los 35 requisitos con los que se cuentan los revisores tuvieron interpretaciones bastante parecidas.

$$Q = Nui / Nr$$

$$Q = 35 / 35$$

$$Q = 1$$

Después de analizados los resultados de las anteriores revisiones se llega a la conclusión que los requisitos presentan un bajo nivel de ambigüedad, pues el valor de Q en la primera revisión se acercó bastante al valor 1 y ya en la segunda no presenta ningún nivel de

## Validación de la solución propuesta

---

ambigüedad. Lo cual es bastante satisfactorio para el presente trabajo ya que ayuda a que el desarrollo de software no tenga que atrasarse por algún desperfecto en los requisitos obtenidos.

### 3.4. Métricas de casos de uso del sistema

Para lograr la calidad del artefacto del Caso de Uso del Sistema obtenido anteriormente se le aplicó un modelo de métricas. El modelo usa 4 atributos, los cuales son (completitud, comprensibilidad, concisión, no trivialidad).

**Completitud:** Un caso de uso es completo si especifica todo lo que deben hacer el actor y el sistema para alcanzar el objetivo del caso de uso.

**Comprensibilidad:** Un caso de uso es comprensible si todos los tipos de lectores (cliente, usuario, jefe de proyecto) pueden entenderlo fácilmente con una mínima explicación del autor.

**Concisión:** Un caso de uso es conciso si no incluye información innecesaria.

**No trivialidad:** Un caso de uso es no trivial si su secuencia de pasos conduce al actor a conseguir el objetivo que persigue la realización del caso de uso. (14)

Métricas	Preguntas
<b>Completitud</b>	<ol style="list-style-type: none"><li>1. ¿Hay respuestas a todas las peticiones que el actor del caso de uso hace al sistema y viceversa?</li><li>2. ¿Se contemplan todos los posibles escenarios para poder alcanzar el objetivo del caso de uso?</li><li>3. ¿Se especifican todas las secuencias alternativas a la secuencia normal?</li><li>4. ¿Se contemplan todas las posibles excepciones a la secuencia normal?</li></ol>
<b>Comprensibilidad</b>	<ol style="list-style-type: none"><li>1. ¿Es posible leer el caso de uso sin</li></ol>

## Validación de la solución propuesta

---

	<p>volver atrás en repetidas ocasiones?</p> <ol style="list-style-type: none"><li>2. ¿Es difícil seguir la secuencia normal del caso de uso por la presencia de las relaciones de inclusión o extensión?</li><li>3. ¿Es difícil seguir la secuencia de pasos por la existencia de demasiados pasos alternativos?</li><li>4. ¿Se han desglosado demasiado los pasos de algún actor o del sistema provocando que el caso de uso avance a un ritmo lento?</li><li>5. ¿Aparecen pasos condicionales para expresar que el sistema comprueba una situación que permite al caso de uso continuar su realización?</li></ol>
<b>Concisión</b>	<ol style="list-style-type: none"><li>1. ¿Podría el caso de uso ser expresado con menos palabras?</li><li>2. ¿Existen elementos que se pueden obviar o aparecen anotaciones innecesarias y que dificultan la lectura del caso de uso?</li><li>3. ¿Aparecen demasiadas iteraciones entre el actor principal del caso de uso y otros elementos del entorno?</li></ol>
<b>No trivialidad</b>	<ol style="list-style-type: none"><li>1. ¿Expresa el nombre del caso de uso un objetivo de un usuario que el sistema debe implementar?</li></ol>



## Validación de la solución propuesta

---

	2. ¿Conduce el caso de uso al actor a conseguir algunos de sus objetivos sin representar un conjunto de iteraciones triviales?
--	--

**Tabla 20 Preguntas asociadas a las métricas (15)**

Una vez aplicadas en dos revisiones las métricas pertinentes a los Casos de Uso del Sistema, se obtuvo la siguiente información:

### **Primera revisión:**

Para el atributo concisión : Se encontró que los casos de uso Gestionar profesor, Gestionar estudiante, Gestionar computadora, Mostrar reporte, Gestionar casos de prueba, Gestionar paquete de revisiones contenían información innecesaria por lo que surgió un error del 27%, lo que implica que el 73% de los casos de uso cumplen adecuadamente con el atributo.

Para el atributo comprensibilidad: Se encontró que los casos de uso Gestionar estudiante y Gestionar profesor contenían información que no permitían un fácil entendimiento, por lo que surgió un error del 9%, lo que implica que el 91% de los casos de uso cumplen adecuadamente con el atributo.

### **Segunda revisión:**

Luego de rectificadas los errores de la primera revisión, basándose en estos se efectuó una segunda revisión en la cual los Casos de Uso del Sistema se encontraban en correcto estado en cuando a completitud, Comprensibilidad, concisión y no trivialidad, por lo que se puede decir que los casos de uso cumplen en un 100% con el modelo de métricas.

## **3.5. Conclusiones**

Durante el capítulo se comprobó la calidad de los artefactos que se generaron una vez culminada la solución propuesta. Para lograr la calidad de dichos artefactos se realizaron dos revisiones en cuanto a la especificación de los requisitos y calidad de la confección del diagrama de casos de uso del sistema. Luego de erradicar los defectos detectados en el transcurso del trabajo, se logró obtener la satisfacción de poder contar con un producto que indemniza las necesidades del cliente.

# Conclusiones Generales

---

## Conclusiones Generales

1. En el diagnóstico de exploración inicial se constató que existen deficiencias en la gestión de la información referente a cada una de las áreas del Grupo Calidad Facultad 9, lo que dificulta el acceso a la documentación.
2. Los referentes teóricos estudiados y los métodos investigativos aplicados como la entrevista y la observación, permitieron obtener la información necesaria para conocer las particularidades y funcionamiento del negocio, lo que posibilitó realizar el proceso de modelación del mismo.
3. El levantamiento de requisitos permitió obtener las funcionalidades requeridas por el cliente, se realizó la documentación referente al análisis y diseño del sistema posibilitando el cumplimiento del objetivo trazado.
4. Se aplicaron métricas para evaluar la calidad de los artefactos generados, lo cual arrojó un resultado satisfactorio.
5. El desarrollo del presente trabajo permitió la elaboración de todos los artefactos correspondientes al análisis y diseño del sistema, logrando establecer un idioma común entre el cliente y el equipo de desarrollo, obteniendo como resultado una amplia y organizada documentación sobre los procesos que se desarrollan en el Grupo Calidad de la Facultad 9.

# Recomendaciones

---

## Recomendaciones

1. Utilizar los artefactos generados en el presente trabajo de diploma, como entrada en los flujos siguientes que propone RUP.
2. Redefinir con el cliente en próximas iteraciones o versiones del producto los requisitos funcionales, con el objetivo de aumentar las funcionalidades del sistema.

# Bibliografía Referenciada

---

## Bibliografía Referenciada

1. Manuel Henríquez Lagarde .2009."Cubasi". [Consultado: 2, 11,2009]. Disponible en <http://www.cubasi.cu/>.
2. Rolando Alfredo Hernández León, Sayda Coello González. 2002. "El paradigma cuantitativo de la investigación científica". [Consultado: 9, 11,2009].
- 3."Mi Tecnológico". [Consultado: 2, 12,2009]. Disponible en <http://www.mitecnologico.com/Main/DefinicionCalidadDeSoftware>
4. Dayli WP. 2008."definición.de". [Consultado: 10, 12, 2009]. Disponible en <http://definicion.de/proceso/>
5. Woodman, Lynda. 1985. "Information management from strategies to action". [Consultado: 5, 1, 2009].
- 6 ."Bibliografica". [Consultado: 10, 12,2009]. Disponible en [Biografica.info](http://Biografica.info).
7. Arles Amado Tamallo Rosales, Yosvany Núñez Figueiras. 2009. "Pruebas de Liberación a la Plataforma de Cálculo Distribuido T-arenal".[Consultado: 20, 2, 2010]. Disponible en <http://biblioteca.uci.cu/sbd/biuci/index.html>
8. Carrion, Juan y Medina, Sergio. 2000. "Los portales y la gestión del conocimiento. Fundación iberoamericana del conocimiento". [Consultado: 9, 11,2009].
9. Informático, Diccionario. 2009. "Portal". [Consultado: 3, 1, 2009]. Disponible en <http://diccionario.babylon.com/Portal>.
10. Rolando Alfredo Hernández León, Sayda Coello González. 2002. "El paradigma cuantitativo de la investigación científica". [Consultado: 9, 11,2009].
11. María A. Mendoza Sanchez.2004." Metodologías De Desarrollo De Software". [Consultado: 3, 2, 2010]. Disponible en [http://www.informatizate.net/articulos/metodologias\\_de\\_desarrollo\\_de\\_software\\_07062004.html](http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html)
12. Evelyn Menéndez Alonso. "Monografias.com" [Consultado: 3, 1, 2009]. Disponible en <http://www.monografias.com/trabajos73/herramientas-case-proceso-desarrollo-software/herramientas-case-proceso-desarrollo-software2.shtml>

## Bibliografía Referenciada

---

13. Edith Maylen, Ortiz, Leydis. 2008. "Análisis y diseño de un nodo virtual de procesos". [Consultado: 15, 4, 2010].
14. Pressman, Roger S. 2005. Ingeniería del Software. "Un enfoque práctico". [Consultado: 5, 4, 2010].
15. B. Bernárdez, A. Durán, M. Toro. 2004. "Una propuesta para la verificación de requisitos basados en métricas". [Consultado: 9, 4, 2010].
16. Oscar M. Fernández Carrasco, Delba García León, Alfa Beltrán Benavides. 1995. "Un enfoque actual sobre la calidad del software". [Consultado: 4, 1, 2009]. Disponible en [http://bvs.sld.cu/revistas/aci/vol3\\_3\\_95/aci05395.htm](http://bvs.sld.cu/revistas/aci/vol3_3_95/aci05395.htm)
17. WordReference. 2010. "Diccionario de la lengua española". [Consultado: 4, 1, 2009]. Disponible en <http://www.wordreference.com/definicion/proceso>
18. Curtis Street, Creswick. 2010. "Métricas de caso de uso". [Consultado: 10, 11, 2009]. Disponible en <http://www.sparxsystems.com.ar/resources/ucmetrics.html>

# Bibliografía Consultada

---

## Bibliografía Consultada

1. Juan Pablo Gomez Gallego, Ing Jorge Galvez. 2007."Fundamentos de la Metodología RUP". [Consultado: 10, 11,2009]. Disponible en <http://www.scribd.com/doc/297224/RUP>.
2. Roberto Hernández Sampier, Carlos Fernández Collado, Pilar Baptista Lucio, "Metodología de la Investigación". Segunda Edición. [Consultado: 15, 11,2009]. Disponible en <http://bibliodoc.uci.cu/pdf/8478290362.pdf> .
3. Curtis Street, Creswick.2010."Métricas de caso de uso". [Consultado: 10, 11,2009]. Disponible en <http://www.sparxsystems.com.ar/resources/ucmetrics.html> .
- 4.Ivar, Jacoboson, Grady, Booch, James, Rumbaugh. 2000. "El Lenguaje Unificado de Modelado". [Consultado: 12, 11,2009].
5. Canós, José H., Letelier, Patricio.2003."Metodologías ágiles en el desarrollo de software". [Consultado: 10, 12, 2009].
6. Instituto Nacional de Estadística e Informática. 1999. "Herramientas CASE". [Consultado: 13, 12, 2009]. Disponible en <http://www.inei.gob.pe/biblioineipub/bancopub/Inf/Lib5103/Libro.pdf>
7. Bayarre, H y Hersford, R. 2004. "Metodología de la Investigación". [Consultado: 23, 12, 2009].
8. Carlos Álvarez de Zayas. 1995. Metodología de la investigación científica. [Consultado: 5, 01, 2010].
9. Rolando Alfredo Hernández León, Sayda Coello González. 2002. El paradigma cuantitativo de la investigación científica. [Consultado: 6, 01, 2010].
10. "Visual Paradigm International". [Consultado: 13, 01, 2010]. Disponible en <http://www.visual-paradigm.com/product/vpum/communityedition.jsp>
11. García, Joaquín. "Ingeniero Software". [Consultado: 14, 01, 2010]. Disponible en <http://www.ingenierosoftware.com/analisisydiseno>

## Bibliografía Consultada

---

12. García, Joaquín. “Ingenieros de Software. *UML e Ingeniería*”. [Consultado: 14, 01, 2010]. Disponible en <http://www.ingenierossoftware.com/analisisydiseno/uml.php>.
13. “Conferencia 1: Introducción a la Ingeniería de Software”. [Consultado: 15, 01, 2010]. Disponible en [http://eva.uci.cu/file.php/42/Clases\\_Curso\\_2009-2010/conferencias/Conferencia\\_1/Profesores/Conferencia\\_1.pdf](http://eva.uci.cu/file.php/42/Clases_Curso_2009-2010/conferencias/Conferencia_1/Profesores/Conferencia_1.pdf) .
14. Hernández Orallo, Enrique. “*El Lenguaje Unificado de Modelado (UML)*”. [Consultado: 16, 01, 2010]. Disponible en <http://eva.uci.cu/mod/resource/view.php?id=11361> .
15. “Instituto Nacional de Estadística e Informática. *Herramientas CASE*”. [Consultado: 18, 01, 2010]. Disponible en <http://www.inei.gob.pe/biblioineipub/bancopub/Inf/Lib5103/Libro.pdf> .
16. **H. Canós, José, Letelier, Patricio y Penadés, Carmen.** “Metodologías Ágiles en el Desarrollo de Software”. [Consultado: 18, 01, 2010]. Disponible en [www.willydev.net/descargas/prev/TodoAgil.pdf](http://www.willydev.net/descargas/prev/TodoAgil.pdf) .
17. “Centro de Informática Ingeniería”. [Consultado: 18, 01, 2010]. Disponible en <http://www.ciinsrl.com/index.php?modules=productos> .
18. **Mendoza Sánchez, María A.** [Consultado: 19, 01, 2010]. Disponible en <http://www.Informatizate.net> .
19. Centro Nacional de Tecnologías de Información. [Consultado: 19, 01, 2010]. Disponible en <http://www.google.com.cu/search?hl=es&q=define%3A+plugin&btnG=Buscar&meta=>.