UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA

CARRERA DE INGENIERIA DE SISTEMAS

TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS.

"METODOLOGÍA PARA LA ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE BASADO EN EL ESTÁNDAR IEEE 830-1998"

AUTORES:

Carlos Daniel Borja Buestán.

Valeria Alexandra Cuji Torres.

DIRECTOR:

Ing. Mauricio Ortiz.

CUENCA, AGOSTO 2013.

DECLARATORIA DE RESPONSABILIDAD

Yo, Carlos Daniel Borja Buestán, portador de la Cedula de Ciudadanía Nº 0105319685-5, declaro bajo juramento que la presente investigación es de total responsabilidad de los autores, y que ha respetado las diferentes fuentes de información realizando la citas correspondientes.

Carlos Daniel Borja Buestán.

Yo, Valeria Alexandra Cuji Torres, portador de la Cedula de Ciudadanía Nº 0104976964, declaro bajo juramento que la presente investigación es de total responsabilidad de los autores, y que ha respetado las diferentes fuentes de información realizando la citas correspondientes.

Valeria Alexandra Cuji Torres

CERTIFICACION DEL DIRECTOR DE TESIS

Certifico que el trabajo de tesis titulado ***"METODOLOGÍA PARA LA ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE BASADO EN EL ESTÁNDAR IEEE 830-1998",*** ha sido dirigido, asesorado, supervisado y realizado bajo mi dirección en todo su desarrollo, la misma que cumple con la reglamentación pertinente, así como lo programado en el plan de tesis y reúne la suficiente validez técnica y práctica, por consiguiente autorizo su certificación

Ing. Mauricio Ortiz

DIRECTOR DE TESIS

AUTORIZACIÓN

Los autores del trabajo de tesis: ***"METODOLOGÍA PARA LA ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE BASADO EN EL ESTÁNDAR IEEE 830-1998",*** Carlos Daniel Borja Buestan, Valeria Alexandra Cuji Torres; autorizan a la Universidad Politécnica Salesiana el uso de la misma con fines académicos.

Carlos Daniel Borja Buestán.

Valeria Alexandra Cuji Torres

AGRADECIMIENTO

Expresamos nuestro agradecimiento a nuestros padres, familiares y amigos por apoyarnos en el transcurso de nuestra vida y en la terminación de este proyecto, guiándonos con sus enseñanzas para el cumplimiento de cada meta propuesta.

A la Universidad Politécnica Salesiana, a la Facultad de Ingeniería y a sus docentes por sus saberes compartidos día a día, saberes que permitieron formarnos como ciudadanos al servicio de la sociedad con valores éticos, morales y profesionales, además agradecer de forma muy especial a nuestro director Ing. Mauricio Ortiz por ser guía en nuestro proyecto, por la dedicación y enseñanza.

Al colegio Técnico Industrial Gualaceo del Cantón Gualaceo, a sus autoridades y en especial a la colaboración del Arq. Marcelo Cando Rector de la Institución.

Al Ing. Cristian Tímbi, por haber ayudado a realizar las encuestas en las Diferentes empresas desarrolladoras de Software.

Índice de Contenido

DECLARATORIA DE RESPONSABILIDAD2

AGRADECIMIENTO

ÍNDICE DE CONTENIDO 1

CAPITULO I 3

GENERALIDADES 3

1.1 INTRODUCCION 3
1.2 PLANTEAMIENTO DEL PROBLEMA 5
1.3 OBJETIVOS 6
General 6
Específicos 6
1.4 JUSTIFICACION 7

CAPITULO II 8

ESTÁNDAR IEEE 830-1998 8

2.1 HISTORIA DE IEEE 8
2.2 Concepto 9
2.3 Estándares de Software IEEE 10
2.3.2 ESTÁNDARES DEL IEEE 11
2.4 IMPORTANCIA DEL USO DE ESTÁNDARES 11
2.5 ESTÁNDAR IEEE 830 13

CAPITULO III 21

ROL DE LA INGENIERÍA DE REQUERIMIENTOS 21

3.1 CONCEPTOS DE INGENIERÍA DE REQUERIMIENTOS 21
3.1.2 Características de los requerimientos 24
3.1.3 Clasificación de los requerimientos 25
3.1.4 Definición de Ingeniería de Requerimientos 28
3.2 IMPORTANCIA DE LA INGENIERÍA DE REQUERIMIENTOS 29
3.3 ESTUDIO DEL PROCESO DE INGENIERÍA DE REQUERIMIENTOS EN EMPRESAS DESARROLLADORAS DE SOFTWARE 30
3.4 PROCESOS DE INGENIERÍA DE REQUERIMIENTOS 38
3.4.1 Elicitación de Requerimientos 38
3.4.2 Análisis de Requerimientos 39
3.4.3 Especificación de Requerimientos 39
3.4.4 Validación de los Requerimientos 40

CAPITULO IV 41

PROPUESTA DE LA METODOLOGÍA PARA LA ESPECIFICACIÓN DE REQUERIMIENTOS 41

4.1 METODOLOGÍA 41
4.1.1 Concepto 41
4.1.2 Tipos de metodologías Para especificación de requerimientos 42
4.2 ESTADO DEL ARTE DE LAS METODOLOGÍAS DE ESPECIFICACIÓN DE REQUERIMIENTOS 47

4.2.1 Metodologías Especificación de Requerimientos *49*
4.3 DEFINICIÓN DE LA METODOLOGÍA A USAR BASADA EN EL ESTÁNDAR IEEE 830-1998 64
4.3.1 ¿Por qué esta metodología? *64*
4.3.2 Técnicas de Ingeniería de Requerimientos. *65*
4.3.3 Metodología Propuesta *68*

CAPITULO V **88**

CASO DE ESTUDIO: COLEGIO TÉCNICO INDUSTRIAL GUALACEO 88
5.1 Descripción de la empresa *88*
5.2 Fase I: Elicitación de Requerimientos *89*
5.3 Fase II: Análisis de Requerimientos *100*
5.4 Fase III: Especificación de Requerimientos *127*
5.5 Fase IV: Validación de Requerimientos *157*
5.6 Fase V: Verificación de Requerimientos *159*

CAPITULO VI **160**

IMPLEMENTACIÓN DE LA APLICACIÓN PARA REALIZAR LA ESPECIFICACIÓN DE REQUERIMIENTOS. **160**

6.1. ANÁLISIS 160
6.1.1 Fase de Elicitación *160*
6.1.2 Fase de Análisis. *167*
6.1.3 Fase de Especificación *174*
6.1.4 Fase de Validación y Verificación *198*
6.2 DISEÑO 202
1.3 IMPLEMENTACIÓN 211
6.4 PRUEBAS 214
6.5 MANUALES Y DOCUMENTACIÓN 222

CAPITULO VII **223**

CONCLUSIONES Y RECOMENDACIONES **223**

ANEXOS **225**

ANEXO A. ENCUESTA 225
ANEXO B. MANUAL DE USUARIO 226

BIBLIOGRAFÍA **236**

CAPITULO I

GENERALIDADES

1.1 INTRODUCCION

En el ámbito de desarrollo de software siempre ha existido una constante preocupación acerca del posible éxito del mismo y una de las inquietudes de la Ingeniería de Software es el garantizar ese éxito. Así a través de la experiencia en este campo se han ido identificando ramas y tópicos de especial importancia dentro del desarrollo de software, cuyo tratamiento es de suma relevancia si se desea que el proyecto a desarrollar cumpla con las necesidades del cliente.

Uno de estos tópicos es respecto a los requerimientos. Estos sometidos a diferentes análisis y debates, indican que repercuten fuertemente en el éxito o fracaso de los proyectos de software. En la publicación emitida por la revista electrónica *The Rational Edge*, acerca del estado y las prácticas recomendadas para el desarrollo de software y sistemas, se le otorga una sección a los requerimientos en la cual se enmarca reiterativamente que el precisarlos es una parte esencial de la fórmula para los proyectos de software exitosos (MCEWEN, 2004).

En el trabajo de tesis se plantea una metodología para la especificación de requerimientos de software la misma que se basa en el estándar IEEE 830 – 1998. Realizada la metodología presentamos una aplicación prototipo para la especificación de requerimientos de software de nombre (*SysToErs*). Esta metodología será aplicada para cubrir las necesidades del Colegio Técnico Industrial Gualaceo, que requiere un sistema educativo que permita el registro de notas y realizar las matriculas para los estudiantes.

Esta tesis abarca conceptos importantes de la ingeniería en requerimientos, asi como también una propuesta metodológica con la cual se pretende contribuir con la aplicación adecuada de las técnicas a usarse en las cuatro fases de la ingeniería de requerimientos.

El documento está organizado en 6 capítulos, que se detallan a continuación:

- El capítulo I: consta de la introducción, el planteamiento del problema, los objetivos y la justificación del proyecto.
- En el segundo capítulo: se estudia el estándar IEEE 830-1998, su concepto, historia, la importancia que tiene y se presenta la plantilla del estándar.
- En el tercer capítulo se desglosa el rol de la Ingeniería de Requerimientos, sus conceptos, importancia y demás definiciones que son necesarios para tener un amplio conocimiento acerca del tema; además se da a conocer los resultados del estudio realizado el cual está enfocado a los procesos de la ingeniería de requerimientos en empresas desarrolladoras de software.
- En el cuarto capítulo, se da a conocer la propuesta metodológica para la especificación de requerimientos, se desglosan las técnicas y herramientas que se utilizara en cada una de las fases de la ingeniera de requerimientos.
- En el capítulo 5, se aplica la propuesta metodológica detallando cada fase de la ingeniería de requerimientos.
- En el capítulo 6 consta de la fase de análisis, diseño, implementación y pruebas del prototipo realizado para la especificación de requerimientos.
- Al final de este documento se puede observar las conclusiones recomendaciones, anexos y referencias bibliográficas.

1.2 PLANTEAMIENTO DEL PROBLEMA

En la actualidad la ingeniera de software usa diferentes tipos de estándares enfocados en medir la calidad del sistema a desarrollar. Existe mucha información en la web, libros y artículos que abarcan este tema, además existen herramientas que buscan ayudar en la aplicación de los procesos de desarrollo de software.

Hoy en día pequeñas, medianas y grandes empresas usan sistemas automatizados dependiendo cada vez más de estos, los desarrolladores, analistas y demás equipo de desarrollo de software debe buscar estándares de calidad y la mejora de procesos para el desarrollo, a pesar de esto muchas aplicaciones fracasan, existen retrasos y siguen existiendo problemas de calidad y proyectos de software que no llegan a cumplir sus objetivos.

En nuestro país, existen empresas que para evitar los problemas ya mencionados anteriormente, deciden adquirir sistemas extranjeros para luego ser personalizados a medida de sus necesidades, lo que termina siendo más costoso e incluso tardando más tiempo de lo planificado.

Esto se da por la falta un correcto estudio previo, por no conocer bien las necesidades de los clientes e incluso porque los interesados en el desarrollo de un sistema no explican bien sus necesidades. A pesar de que existan herramientas que ayuden a mejorar los procesos y existan estándares de calidad; no se podrá obtener un buen sistema si no se cumple con las necesidades reales de los usuarios finales.

Según Walract, estudios realizados muestran que más del 56% de los proyectos de software fracasan por no realizar un estudio previo de requisitos, o por realizar esta fase de estudio y análisis con imprecisiones y vaguedad. (Zavala Ruiz, 2004)

Con respecto al costo de corregir los errores de desarrollo de software, según Walract, en la fase de Estudio Análisis es de un 82% siendo más costoso ya que se tiene que regresar a la fase inicial del proyecto (Zavala Ruiz, 2004).

Es por esto que el estudio de la Ingeniería de Requerimientos cumple un papel importante en el proceso de desarrollo de software, ya que enfoca un área fundamental: la definición de lo que se desea producir; formando una sociedad entre

el cliente y el desarrollador. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema; con el fin de minimizar los problemas relacionados al desarrollo de sistemas, y evitar que los proyectos fracasen debido a una mala gestión de los requerimientos e incluso puede tomarse como base para futuras mejoras.

1.3 OBJETIVOS

General

- Establecer una metodología para la especificación de requerimientos para el proceso de desarrollo de software, que contribuya a mejorar la calidad del sistema y realizar una aplicación de la misma.

Específicos

- Investigar qué rol desempeña la Ingeniería de requerimientos en el desarrollo de Software.
- Conocer las características de los requerimientos de software.
- Conocer los diferentes niveles de abstracción de los requerimientos.
- Aplicar el estándar IEEE 830-1998 para la especificación de requerimientos.
- Comprender la importancia de una correcta gestión de requerimientos.
- Conocer los posibles problemas que se darán a lo largo de la elicitación de los requerimientos de Software.
- Establecer las técnicas, métodos y herramientas que se van a usar en la metodología propuesta.

1.4 JUSTIFICACION

Uno de los aspectos más importantes para la implementación de un software es tener claro los requerimientos del usuario final. Es por esto que la ingeniería de requerimientos es parte clave del proceso de software.

A pesar de esta afirmación, la mayoría de programadores no le da la suficiente importancia a la especificación de requerimientos de software, pues dicen que es una pérdida de tiempo y que no presenta beneficio alguno, arriesgando así innecesariamente el desarrollo del software. Es muy habitual escuchar entre los entendidos del desarrollo de programas de software que gran número de los proyectos fracasan por no cumplir una adecuada definición y especificación de requerimientos.

La especificación de requerimientos de software es la pieza fundamental en un proyecto de desarrollo de software pues gracias a esto se marca el punto de partida para la creación de una aplicación como la planificación en lo que respecta a tiempos, costos, recursos y la elaboración de cronogramas con lo que se contará durante la fase de desarrollo.

Por tal motivo, se ha visto la necesidad de plantear una metodología que sirva de guía que permita especificar los requerimientos para el desarrollo de software basados en el estándar IEEE 830-1998, y así recoger información adecuada para establecer de forma clara las condiciones que el cliente y usuario final requiera.

CAPITULO II

Estándar IEEE 830-1998

2.1 Historia de IEEE

La IEEE una organización sin fines de lucro dedicado a promover la innovación y la excelencia tecnológica en beneficio de la humanidad.

Fue formada en 1963, por la fusión del IRE (Institute of Radio Engineers) en español (Instituto de Ingenieros de Radio), fundado en 1912 y el AIEE (The American Institute of Electrical Engineers), fundado en 1884 (IEEE, 2010). Este último surgió durante un período de optimismo y entusiasmo; debido a que las aplicaciones en electricidad se estaban incrementando rápidamente.

Desde el comienzo, las comunicaciones por cable y los sistemas de luz y potencia fueron los intereses principales del AIEE. Como antiguo y activo participante en el desarrollo de normas para la industria, el Instituto fundó las bases para todos los trabajos en normas eléctricas hechos en los Estados Unidos.

El IRE se trató sobre todo del radio en ingeniería, y se estableció a partir de dos organizaciones más pequeñas, la Sociedad de Ingenieros Wireless y Telégrafos y el Instituto Wireless. Con el auge de la electrónica en la década de 1930, ingenieros electrónicos por lo general se convirtieron en miembros del IRE.

Después de la segunda guerra mundial estas dos organizaciones llegaron a ser más competitivas y finalmente, en 1961 las dirigencias tanto del IRE como del AIEE resolvieron buscarle término a esas dificultades a través de la consolidación. El año siguiente se formuló y aprobó un plan de fusión, el cual entró en vigor el 1 de enero de 1963, se hicieron planes para unificar las actividades técnicas y las unidades geográficas de las dos sociedades y para establecer un programa de publicaciones

unificado para la nueva organización, el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) (EcuRed, 2010).

En la actualidad, el IEEE es la organización técnica profesional más grande y prestigiada del mundo, sus actividades se extienden mucho más allá de lo que sus predecesores podrían haber previsto. Sigue siendo, sin embargo y justo como hace más de un siglo, el vocero principal de los más importantes y apasionantes campos tecnológicos de su tiempo.

Ilustración 1: Evolucón de la IEEE: Fuente: (IEEE, n.d.)

2.2 Concepto

Según el mismo IEEE, es la Asociación de profesionales más grande del mundo dedicado a la promoción de la innovación tecnológica y excelencia en beneficio de la humanidad. IEEE y sus miembros inspiran a una comunidad global a través de sus publicaciones muy citadas, conferencias, estándares de tecnología y actividades profesionales y educativas (IEEE, 2010).

Mediante sus actividades de publicación técnica, conferencias y estándares, el IEEE produce más del 30% de la literatura publicada en el mundo sobre ingeniería eléctrica, en computación, telecomunicaciones y tecnología de control, organiza más de 350 grandes conferencias al año en todo el mundo, y posee cerca de 900 estándares activos, con otros 700 más bajo desarrollo (IEEE, 2010).

Su trabajo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales. Algunos de sus estándares son:

VHDL, POSIX, IEEE 1394, IEEE 488, IEEE 802, IEEE 802.11, IEEE 754, IEEE 830.

2.3 Estándares de Software IEEE

Para abordar este tema se va a definir primeramente algunos conceptos

2.3.1 ¿Qué es un estándar?

Estándar puede ser conceptualizado como la definición clara de un modelo, criterio, regla de medida o de los requerimientos mínimos aceptables para la operación de procesos específicos, con el fin asegurar la calidad.

Los estándares señalan claramente el comportamiento esperado y deseado y son utilizados como guías para evaluar su funcionamiento y lograr el mejoramiento continuo de los servicios. Requieren ser establecidos con el fin de contar con una referencia que permita identificar oportunamente las variaciones presentadas en el desarrollo de los procesos y aplicar las medidas correctivas necesarias.

Los estándares pueden ser desarrollados por la propia compañía, por sociedades profesionales, o por organismos internacionales.

Los estándares en ingeniería del software se ocupan de la práctica responsable de la misma, tratan con el proceso en vez del producto.

Tomando en cuenta principalmente la disciplina de ingeniería de software; el IEEE desarrolla sus estándares a través de una de sus entidades, el *IEEE Standards Association* (IEEE-SA). Asimismo, este desarrollo se potencia mediante otras entidades técnicas abarcadas por el Instituto. En el caso puntual de este conjunto de estándares, estas entidades técnicas son la *IEEE Computer Society* (IEEE-CS) y el IEEE *Technical Council on Software Engineering* (TCSE), las cuales participan de esta actividad mediante un comité: el *Software & Systems Engineering Standards Committee* (S2ESC). También hay que tener en cuenta que para el desarrollo de estos estándares participan organizaciones de todo tipo como empresas del sector privado, universidades, etc.

2.3.2 Estándares del IEEE

El IEEE tiene un conjunto de 22 estándares creados para software, estos se dividen de la siguiente manera (Jenkins, 2000):

- 15 estándares del proceso
- 5 estándares del producto
- 1 glosario de términos
- 1 taxonomía (clasificación)

2.4 Importancia del uso de estándares

La ingeniería de software se define como una disciplina para producir software de calidad desarrollado sobre las agendas y costes previstos y satisfaciendo los requerimientos, para cumplir con este concepto y llegar a un producto de calidad el uso de estándares aunque no es obligatorio es importante por las siguientes razones:

- Agrupan lo mejor y más apropiado de las buenas prácticas y usos del desarrollo de software.

- Engloban los "conocimientos".

- Proporcionan un marco para implementar procedimientos de aseguramiento de la calidad.

- Proporcionan continuidad y entendimiento entre el trabajo de personas y organizaciones distintas.

- Son indispensables cuando muchos productos, personas y herramientas deben coexistir. Promoviendo el uso correcto de métodos y herramientas y también permitiendo que los desarrolladores se comuniquen entre sí.

- **Facilitan el mantenimiento del software:** El uso correcto de un estándar permite a los desarrolladores realizar de manera ordenada y correcta el proceso de creación del producto, conociendo exactamente las variables,

atributos, métodos utilizados, el lugar en donde se encuentran, etc. Esto permite realizar correcciones o mejoras del sistema de manera más rápida.

- **Mejoramiento del producto:** El uso de estándares IEEE son voluntarios, las organizaciones que los usen lo hacen principalmente para mejorar los productos o a su vez para mejorar la percepción de sus productos en el mercado. Los estándares sirven para mejorar los procesos de negocios, permitiendo desarrollar los productos con costos más apropiados.

- **Protección al comprador:** Debido a la creciente complejidad de los productos tecnológicos es imposible examinar los mismos, y muchos aspectos de estos se mantienen ocultos hasta después de ser adquiridos. Por esta razón los estándares pueden jugar un rol importante cuando proveen información precisa acerca de la adecuación de los productos para usos específicos, permitiendo al comprador elegir productos necesarios para él.

- **Protección al negocio:** El uso de un estándar puede ayudar a empresas u organizaciones en los siguientes aspectos:

    - **Litigios:** el uso de un estándar puede servir de defensa en caso que se pretenda demostrar algún tipo de negligencia.
    - **Respaldo:** El adherirse voluntariamente a estándares respalda la seriedad y confiabilidad de la empresa que así lo hace.
    - **Contratos:** En situaciones contractuales la aplicación adecuada de estándares protegen a ambas partes divide responsabilidades, clarifica terminología y define procedimientos esperados

- **Incrementa la disciplina profesional:** La existencia de estándares y uso de los mismos es un paso importante en la formalización de la Ingeniería de Software. Define los métodos esperados en la práctica responsable de la ingeniería de software, siguiendo un proceso correcto para conseguir cumplir las metas de la mejor manera.
- **Introducción de tecnología:** Según el SEI (Software Engineering Institute), el uso de estándares juegan un rol importante en la evolución tecnológica, esto se debe a que permite la creación de aplicaciones de mejor calidad.

2.5 Estándar IEEE 830

1. Introducción

En esta sección se proporcionara una introducción a todo el documento de Especificación de Requerimientos Software (ERS). Consta de varias subsecciones: propósito, ámbito del sistema, definiciones, referencias y visión general del documento (IEEE, 2010).

1.1. Propósito

En esta subsección se definirá el propósito del documento ERS y se especificara a quien va dirigido el documento.

1.2. Ámbito del Sistema

En esta subsección:

Se podrá dar un nombre al futuro sistema (p.ej. Mi Sistema).
Se explicara lo que el sistema hará y lo que no hará.
Se describirán los beneficios, objetivos y metas que se espera alcanzar con el futuro sistema.

Se referenciaran todos aquellos documentos de nivel superior (p.e. en Ingeniera de Sistemas, que incluyen Hardware y Software, deberá mantenerse la consistencia con el documento de especificación de requerimientos globales del sistema, si existe).

1.3. Definiciones, Acrónimos y Abreviaturas

En esta subsección se definirán todos los términos, acrónimos y abreviaturas utilizadas en la ERS.

1.4. Referencias

En esta subsección se mostrara una lista completa de todos los documentos referenciados en la ERS.

1.5. Visión General del Documento

Esta subsección describe brevemente los contenidos y la organización del resto de la ERS.

2. Descripción General

En esta sección se describen todos aquellos factores que afectan al producto y a sus requerimientos. No se describen los requerimientos, sino su contexto. Esto permitirá definir con detalle los requerimientos en la sección 3, haciendo que sean más fáciles de entender.

Normalmente, esta sección consta de las siguientes subsecciones: Perspectiva del producto, funciones del producto, características de los usuarios, restricciones, factores que se asumen y futuros requerimientos.

2.1. Perspectiva del Producto

Esta subsección debe relacionar el futuro sistema (producto software) con otros productos. Si el producto es totalmente independiente de otros productos, también debe especificarse aquí. Si la ERS define un producto que es parte de un sistema mayor, esta subsección relacionara los requerimientos del sistema mayor con la funcionalidad del producto descrito en la ERS, y se identificaran las interfaces entre el producto mayor y el producto aquí descrito. Se recomienda utilizar diagramas de bloques.

2.2. Funciones del Producto

En esta subsección de la ERS se mostrará un resumen, a grandes rasgos, de las funciones del futuro sistema. Por ejemplo, en una ERS para un programa de contabilidad, esta subsección mostrará que el sistema soportará el mantenimiento de cuentas, mostrará el estado de las cuentas y facilitará la facturación, sin mencionar el enorme detalle que cada una de estas funciones requiere. Las funciones deberán mostrarse de forma organizada, y pueden utilizarse gráficos, siempre y cuando dichos gráficos reflejen las relaciones entre funciones y no el diseño del sistema.

2.3. Características de los Usuarios

Esta subsección describirá las características generales de los usuarios del producto, incluyendo nivel educacional, experiencia y experiencia técnica.

2.4. Restricciones

Esta subsección describirá aquellas limitaciones que se imponen sobre los desarrolladores del producto:

- Políticas de la empresa
- Limitaciones del hardware
- Interfaces con otras aplicaciones
- Operaciones paralelas
- Funciones de auditoría
- Funciones de control
- Lenguaje(s) de programación
- Protocolos de comunicación
- Requerimientos de habilidad
- Criticidad de la aplicación
- Consideraciones acerca de la seguridad

2.5. Suposiciones y Dependencias

Esta subsección de la ERS describirá aquellos factores que, si cambian, pueden afectar a los requerimientos. Por ejemplo, los requerimientos pueden presuponer una cierta organización de ciertas unidades de la empresa, o pueden presuponer que el sistema correrá sobre cierto sistema operativo. Si cambian dichos detalles en la organización de la empresa, o si cambian ciertos detalles técnicos, como el sistema operativo, puede ser necesario revisar y cambiar los requerimientos.

2.6. Requerimientos Futuros

Esta subsección esbozará futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.

3. Requerimientos Específicos

Esta sección contiene los requerimientos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requerimientos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requerimientos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los

usuarios, operadores y otros sistemas. Esta es la sección más larga e importante de la ERS. Deberán aplicarse los siguientes principios:

- El documento deberá ser perfectamente legible por personas de muy distintas formaciones e intereses.
- Deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requerimientos.
- Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.

Lo ideal, aunque en la práctica no siempre realizable, es que los requerimientos posean las siguientes características:

- Corrección: La ERS es correcta si y sólo si todo requisito que figura aquí (y que será implementado en el sistema) refleja alguna necesidad real. La corrección de la ERS implica que el sistema implementado será el sistema deseado.
- No ambiguos: Cada requisito tiene una sola interpretación. Para eliminar la ambigüedad inherente a los requerimientos expresados en lenguaje natural, se deberán utilizar gráficos o notaciones formales. En el caso de utilizar términos que, habitualmente, poseen más de una interpretación, se definirán con precisión en el glosario.
- Completos: Todos los requerimientos relevantes han sido incluidos en la ERS. Conviene incluir todas las posibles respuestas del sistema a los datos de entrada, tanto válido como no válido.
- Consistentes: Los requerimientos no pueden ser contradictorios. Un conjunto de requerimientos contradictorio no es implementable.
- Clasificados: Normalmente, no todos los requerimientos son igual de importantes. Los requerimientos pueden clasificarse por importancia (esencial, condicional u opcional) o por estabilidad (cambios que se espera que afecten al requisito). Esto sirve, ante todo, para no emplear excesivos recursos en implementar requerimientos no esenciales.
- Verificables: La ERS es verificable si y sólo si todos sus requerimientos son verificables. Un requisito es verificable (testeable) si existe un proceso finito y no costoso para demostrar que el sistema cumple con el requisito. Un

requisito ambiguo no es, en general, verificable. Expresiones como a veces, bien, adecuado, etc. Introducen Ambigüedad en los requerimientos. Requerimientos como \en caso de accidente la nube tóxica no se extenderá más allá de 25Km" no es verificable por el alto costo que conlleva.

- Modificables: La ERS es modificable si y sólo si se encuentra estructurada de forma que los cambios a los requerimientos pueden realizarse de forma fácil, completa y consistente. La utilización de herramientas automáticas de gestión de requerimientos (por ejemplo *RequisitePro o Doors*) facilitan enormemente esta tarea.
- Trazables: La ERS es trazable si se conoce el origen de cada requisito y se facilita la referencia de cada requisito a los componentes del diseño y de la implementación. La trazabilidad hacia atrás indica el origen (documento, persona, etc.) de cada requisito. La trazabilidad hacia delante de un requisito R indica qué componentes del sistema son los que realizan el requisito R.

3.1. Interfaces Externas

Se describirán los requerimientos que afecten a la interfaz de usuario, interfaz con otros sistemas (hardware y software) e interfaces de comunicaciones.

3.2. Funciones

Esta subsección (quizá la más larga del documento) deberá especificar todas aquellas acciones (funciones) que deberá llevar a cabo el software. Normalmente (aunque no siempre), son aquellas acciones expresables como el sistema deberá. . .". Si se considera necesario, podrán utilizarse notaciones gráficas y tablas, pero siempre supeditadas al lenguaje natural, y no al revés.

Es importante tener en cuenta que, en 1983, el Estándar de IEEE 830 establece que las funciones deberán expresarse como una jerarquía funcional (en paralelo con los DFDs propuestos por el análisis estructurado). Pero el Estándar de IEEE 830, en sus últimas versiones, ya permite organizar esta subsección de múltiples formas, y sugiere, entre otras, las siguientes:

- Por tipos de usuario: Distintos usuarios poseen distintos requerimientos. Para cada clase de usuario que exista en la organización, se especificaran los

requerimientos funcionales que le afecten o tengan mayor relación con sus tareas.

- Por objetos: Los objetos son entidades del mundo real que serán reflejadas en el sistema. Para cada objeto, se detallarán sus atributos y sus funciones. Los objetos pueden agruparse en clases. Esta organización de la ERS no quiere decir que el diseño del sistema siga el paradigma de Orientación a Objetos.
- Por objetivos: Un objetivo es un servicio que se desea que ofrezca el sistema y que requiere una determinada entrada para obtener su resultado. Para cada objetivo o sub objetivo que se persiga con el sistema, se detallarán las funciones que permitan llevarlo a cabo.
- Por estímulos: Se especificaran los posibles estímulos que recibe el sistema y las funciones relacionadas con dicho estímulo. Por jerarquía funcional: Si ninguna de las anteriores alternativas resulta de ayuda, la funcionalidad del sistema se especificara como una jerarquía de funciones que comparten entradas, salidas o datos internos. Se detallarán las funciones (entrada, proceso, salida) y las subfunciones del sistema. Esto no implica que el diseño del sistema deba realizarse según el paradigma de Diseño Estructurado.

- Para organizar esta subsección de la ERS se elegirá alguna de las anteriores alternativas, o incluso alguna otra que se considere más conveniente.

3.3. Requerimientos de Rendimiento

Se detallarán los requerimientos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc. También, si es necesario, se especificaran lo requerimientos de datos, es decir, aquellos requerimientos que afecten a la información que se guardará en la base de datos. Por ejemplo, la frecuencia de uso, las capacidades de acceso y la cantidad de registros que se espera almacenar (decenas, cientos, miles o millones).

3.4. Restricciones de Diseño

Todo aquello que restrinja las decisiones relativas al diseño de la aplicación: Restricciones de otros estándares, limitaciones del hardware, etc.

3.5. Atributos del Sistema

Se detallarán los atributos de calidad del sistema: Fiabilidad, mantenibilidad, portabilidad, y, muy importante, la seguridad. Deberá especificarse qué tipos de usuario estarán autorizados, o no, a realizar ciertas tareas, y cómo se implementarán los mecanismos de seguridad (por ejemplo, por medio de un login y un password).

3.5. Otros Requerimientos

Cualquier otro requisito que no encaje en otra sección.

4. Apéndices

Pueden contener todo tipo de información relevante para la ERS pero que, propiamente, no forme parte de la ERS. Por ejemplo:

1. Formatos de entrada/salida de datos, por pantalla o en listados.
2. Resultados de análisis de costes.
3. Restricciones acerca del lenguaje de programación

A continuación se muestra la estructura de la ERS propuesta por el IEEE en su estándar 830.

1. INTRODUCCIÓN

1.1. Propósito
1.2. Ámbito del Sistema
1.3. Definiciones, acrónimos y abreviaturas.
1.4. Referencias
1.5. Visión General

2. DESCRIPCION GENERAL

2.1. Perspectiva del producto.
2.2. Funcionalidad del producto
2.3. Características de los usuarios
2.4. Restricciones.
2.5. Suposiciones y dependencias
2.6. Evolución Previsible del sistema.

3. REQUERIMIENTOS ESPECIFICOS.

3.1. Requerimientos comunes de los interfaces

3.1.1. Interfaces de Usuario.

3.1.2. Interfaces de Hardware.

3.1.3. Interfaces de Software.

3.1.4. Interfaces de Comunicación.

3.2. Requerimientos Funcionales.

3.3. Requerimientos No Funcionales.

4. APENDICES.

CAPITULO III

Rol de la Ingeniería de Requerimientos

3.1 Conceptos de Ingeniería de Requerimientos

Tanto la ingeniería del software como la ingeniería de requerimientos son temas recientes, y es normal encontrar que hay autores que presentan diversos conceptos.

3.1.1 Algunos conceptos de requerimiento

Según el Estándar de IEEE, Glosario de terminología de ingeniería de software (1990) define a requerimiento como:

1. Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo.
2. Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal.
3. Una representación documentada de una condición o capacidad como en los puntos 1 o 2.

Sommerville y Sawyer (1997) expresan que requerimientos son una especificación de lo que debe ser implementado. Son descripciones de una propiedad o un atributo o de como el sistema debe comportarse. También pueden ser una restricción en el proceso de desarrollo del sistema (Richard H & Sidney C, 1997).

Según (Young, 2004), "un requerimiento es un atributo necesario para el sistema a desarrollar, en el cual se puede describir una funcionalidad o característica que tenga valor para los stakeholders dentro del mismo".

3.1.1.1 Fuentes de información de requerimientos

Del concepto de Young, se deduce que los requerimientos se pueden encontrar por medio de diferentes personas o departamentos de la organización. A continuación mencionaremos algunas de estas (Courage & Baxter, 2005):

Tabla 1: Fuentes de información de requerimientos

Fuentes	**Descripción**
Administración	Proporciona requerimientos del negocio y parámetros importantes dentro de la lógica del negocio que deben ser tenidos en cuenta para el desarrollo de la solución.
Aspectos legales	Se encargan de revisar los requerimientos legales, que son las implicaciones legales que afectarían el desarrollo de la solución, así como las licencias de las herramientas de desarrollo y componentes adicionales**.**
Analistas del negocio	Dan a conocer las necesidades del negocio, las necesidades funcionales y de rendimiento. También evalúan si es necesario hacer cambios a los requerimientos que actualmente se plantearon en la empresa.
Software	Hacen énfasis en los requerimientos de Software. También evalúan si es necesario hacer cambios a los requerimientos actualmente proyectados en la empresa.
Hardware	Especifican requerimientos a nivel de infraestructura y recursos físicos en los que se basará el desarrollo del sistema.
Usuarios	Describen requerimientos de usuarios y atributos a evaluar para los mismos.
Soporte Técnico	Dan asistencia a los usuarios en la descripción de los requerimientos del usuario. Buscan orientar lo que dice el usuario de una forma más técnica, facilitando así el entendimiento del equipo de desarrollo de la solución.

Fuente: (Courage & Baxter, 2005): **Autor:** Valeria Cuji.

3.1.1.2 Importancia de los requerimientos

Con los siguientes conceptos se podrán dar a conocer la importancia que tienen los requerimientos para el desarrollo de un software de calidad:

- Según Roger S. Pressman (1995), para que un esfuerzo de desarrollo de software tenga éxito, es esencial comprender perfectamente los requerimientos del software. Independientemente de lo bien diseñado o codificado que esté un programa, si se ha analizado y especificado pobremente, decepcionará al usuario y desprestigiará al que lo ha desarrollado (Palacios, 2006).
- Federick P. Brooks (1995) dice que la parte más difícil en la construcción de sistemas software es decidir precisamente qué construir. Ninguna otra parte del trabajo conceptual es tan ardua como establecer los requerimientos técnicos detallados, incluyendo todas las interfaces con humanos, máquinas y otros sistemas software. Ninguna otra parte del trabajo puede perjudicar tanto el resultado final si se realiza de forma errónea. Ninguna otra parte es tan difícil de rectificar posteriormente (Palacios, 2006).

Los dos autores citados afirman que para construir un sistema lo más importante es comprender lo que el usuario necesita y a su vez esta parte es la más difícil de realizar. También afirman que si se ha realizado de manera incorrecta este procedimiento el resultado final fallara siendo este el más difícil de corregir al final.

El recolectar buenos requerimientos trae consigo ciertos beneficios como son:

- **Acuerdo entre desarrolladores, clientes y usuarios sobre el trabajo que debe realizarse:** Unos requerimientos bien elaborados y validados con el cliente evitan descubrir al terminar el proyecto que el sistema no era lo que se pedía.

- **Acuerdo entre desarrolladores, clientes y usuarios sobre los criterios que se emplearán para su validación:** Resulta muy difícil demostrar al cliente que el producto desarrollado hace lo que el pidió si su petición no está documentada y validada por él.

- **Base objetiva para la estimación de recursos (coste, personal en número y competencias, equipos y tiempo):** Si los requerimientos no comprenden necesidades reales, las estimaciones no dejan de ser meras apuestas. Las estimaciones en el fondo son cálculos de probabilidad que siempre implican un margen de error; por esta razón disponer de la mayor información posible reduce el error.

- **Concreción de los atributos de calidad:** Más allá de funcionalidades precisas, los requerimientos recogen atributos de calidad necesarios que en ocasiones son desconocidos por los desarrolladores, produciendo finalmente sistemas sobredimensionados o con serias deficiencias de rendimiento.

- **Eficiencia en el consumo de recursos: reducción de la re-codificación, reducción de omisiones y malentendidos:** Tener un conocimiento preciso de lo que hay que hacer evita la prueba y error, repetición de partes ya desarrolladas, etc.

3.1.2 Características de los requerimientos

Senn James (1992) afirma que las características de los requerimientos son sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características de atributos tanto individualmente como en grupo tomado (Perez Huebe, 2005).

Un requerimiento debe ser (GONZALES, 2011):

- **Especificado por escrito:** Como todo contrato o acuerdo entre dos partes.
- **Posible de probar o verificar**: Si un requerimiento no se puede comprobar, entonces ¿cómo se sabe si se cumplió con él o no?
- **Conciso:** Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- **Completo:** Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.

- **Consistente:** Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- **No ambiguo:** Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.

3.1.3 Clasificación de los requerimientos

De acuerdo a Ma. Teresa Ventura (2002), se identifican principalmente tres niveles de requerimientos que se expresan en el siguiente cuadro:

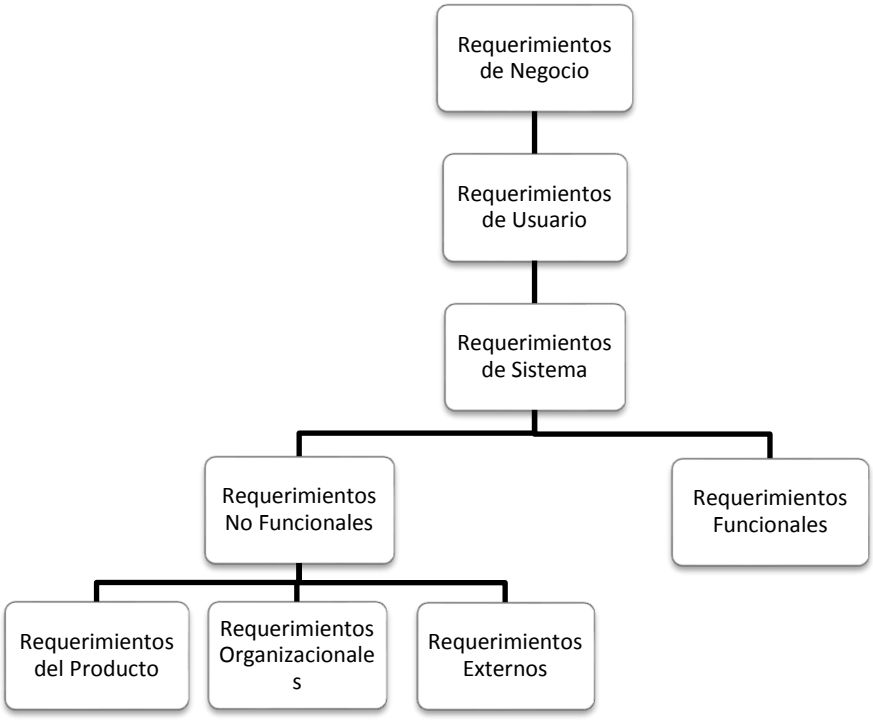


Ilustración 2 Clasificación de requerimientos: Autor: Valeria Cuji. **Fuente: (Somerville, 2005),**

3.1.3.1 Requerimientos de negocio

Estos requerimientos **"**describen como sería mejorar el mundo para ciertas comunidades si el producto estuviera disponible" (Wiegers, 2003), representan los objetivos establecidos por la organización, representan las necesidades de los usuarios y otros involucrados que están siendo afectados por el problema y/o serán afectados por la solución sin descuidar las reglas del negocio de la organización

(Martínez Guerrero & Silva Delgado, 2010). Un requerimiento de negocio es una reflexión del negocio, personal, un problema operacional u oportunidad que debe ser destinada en orden para justificar el desarrollo, compra o uso de un nuevo sistema.

En el proceso de transformar las reglas de negocio en requerimientos, intervienen desde los expertos del negocio, que son los que tienen la información, hasta los analistas de Sistemas que son los que la interpretan, pasando por un comité ejecutivo de la organización quien es la que avala la información que se va a proporcionar al analista (Martínez Guerrero & Silva Delgado, 2010).

3.1.3.2 Requerimientos de Usuario

Son descripciones simples, en el lenguaje del usuario que se utilizan para comunicar la solución de alto nivel a los involucrados. Se denominan características, es decir servicios que el sistema proporciona para cumplir una o más necesidades de los involucrados.

3.1.3.3 Requerimientos de Sistema

Son los que describen de manera más específica la solución. Canalizan una o más características requeridas por los usuarios en soluciones de software específicas.

3.1.3.4 Requerimientos funcionales

Describen lo que el sistema debe hacer, es decir especifican acciones que el sistema debe ser capaz de realizar, sin considerar restricciones físicas. Los requerimientos funcionales especifican el comportamiento del sistema.

3.1.3.5 Requerimientos no funcionales

Describen únicamente atributos del sistema o atributos del ambiente del sistema y pueden ser por ejemplo: requerimientos de interfaz, de diseño, de implementación, legales, físicos, de costo, de tiempo, de calidad, de seguridad, de construcción, de operación, entre otros (Young, 2004).

Los requerimientos no funcionales tienen las siguientes características (Aurum & Wohlin, 2005):

- Requerimientos no funcionales están relacionadas con función en conjunto del sistema.
- Son propiedades de las funciones que el sistema deben tener. Por ende no pueden existir requerimientos no funcionales sin que hayan requerimientos funcionales.
- El analista debe asegurarse que haya un equilibrio entre estos requerimientos, ya que es posible que ocurran contradicciones entre varios de estos.

Estos requerimientos también consideran algunas restricciones, es decir, se encargan de definir aspectos como por ejemplo los estándares de calidad se deben seguir en el desarrollo del sistema (Somerville, 2005).

3.1.4.5.1 Tipos de requerimientos no funcionales

En la siguiente tabla se describe la clasificación de los requerimientos no funcionales:

Ilustración 3: Tipos de Requerimientos no Funcionales **Fuente:** (Somerville, 2005),

Requerimientos del Producto

Estos requerimientos especifican el comportamiento del producto. Ejemplos: rapidez de la ejecución, capacidad de memoria, fiabilidad, etc.

Requerimientos Organizacionales:

Derivan de políticas y procedimientos existentes en la organización del cliente y del desarrollador. Ejemplos: Estándares de procesos, métodos de diseño, lenguajes de programación, métodos de entrega, etc.

3.1.4 Definición de Ingeniería de Requerimientos

Existen varias las definiciones de Ingeniería de Requerimientos según diferentes autores, de las cuales hemos escogido las siguientes:

Para (Nuseibeh, 2000) (Zave, 1997) podemos decir que:

"La Ingeniería de Requerimientos es la rama de la ingeniería de software que descubre los objetivos del mundo real de un sistema, por medio de la identificación de los interesados y sus necesidades, y documentándolos de forma adecuada para el análisis, la comunicación y la posterior implementación. También se determina por las relaciones entre la función del sistema y las restricciones que este tiene, con el fin de precisar especificaciones del comportamiento del software a medida que pasa el tiempo y que llegan nuevas versiones".

Según Goguen (1994) la ingeniería de requerimientos se define como:

"Uno de los aspectos más importantes de la ingeniería de requerimientos es la comunicación, característica esta que vuelve el proceso complejo por la alta presencia del factor humano que contiene y es la responsable de que la disciplina contenga aspectos sociales y culturales y no solo de índole técnica" citado de (Tuesta, Cataldi, & Neil , 2011) .

Para el Instituto de Ingenieros en Electricidad y Electrónica (IEEE) la ingeniería de requerimientos es:

"La ciencia y disciplina relacionada con el análisis y documentación de requerimientos, incluyendo el análisis de necesidades, el análisis de requerimientos y la especificación de requerimientos. También proporciona los mecanismos adecuados para facilitar las actividades de análisis, documentación y verificación de estos."

Según los conceptos citados anteriormente se define a la ingeniería de requerimientos como la ciencia que descubre las necesidades que el usuario, cliente o interesado busca al desarrollar un sistema, así como también esta rama incluye el análisis de las necesidades adquiridas así como también la documentación adecuada de los mismos a la que también se le conoce como especificación de requerimientos.

3.2 Importancia de la ingeniería de requerimientos

Según Macaulay (1996), los principales beneficios que se obtienen de la Ingeniería de Requerimientos son (Perez Huebe, 2005):

- Permite gestionar las necesidades del proyecto en forma estructurada: Cada actividad de la IR consiste de una serie de pasos organizados y bien definidos.

- Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados: La IR proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.

- Disminuye los costos y retrasos del proyecto: Muchos estudios han demostrado que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro; especialmente aquellas decisiones tomadas durante la especificación de requerimientos (RE).

- Mejora la calidad del software: La calidad en el software tiene que ver con cumplir un conjunto de requerimientos (funcionalidad, facilidad de uso, confiabilidad, desempeño, etc.).

- Mejora la comunicación entre equipos: La especificación de requerimientos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será exitoso.

- Evita rechazos de usuarios finales: La ingeniería de requerimientos obliga al cliente a considerar sus requerimientos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto.

3.3 Estudio del Proceso de Ingeniería de requerimientos en empresas desarrolladoras de software

Según un estudio estadístico exploratorio realizado en las empresas desarrolladoras de software en Ecuador en las ciudades Guayaquil, Quito y Cuenca (R. Salazar, K. Villavicencio, & Monique), los resultados indican que la mayoría de están empresas están ubicadas en la ciudad de Quito (61%), aunque las ciudades de Guayaquil y Cuenca son las que le siguen en esta práctica. A pesar de que la cuidad de Cuenca cuenta con pocas empresas desarrolladoras de software hemos realizado una encuesta con el objetivo de conocer si los desarrolladores de las mismas conocen metodologías y técnicas que les ayuden a tener claros los requerimientos de los clientes, a analizarlos para obtener resultados finales exitosos.

Se ha tomado una población de 20 personas tomando en cuenta que en la ciudad de cuenca no existen empresas desarrolladoras de Software de gran Magnitud.

Las preguntas que se han propuesto, tienen el objetivo de conocer si los desarrolladores conocen a cerca de la ingeniería de requerimientos, las metodologías que existen y que se pueden aplicar así como de las técnicas que se pueden usar.

Para realizar este estudio se realizara una encuesta a ingenieros de empresas que desarrollan software (**Ver Anexo A**).

Esta encuesta se repartió entre desarrolladores de la Cooperativa JEP (*Juventud Ecuatoriana Progresista*), la Cooperativa Jardín Azuayo, y la Universidad Politécnica Salesiana. Con esta encuesta obtuvimos los siguientes resultados:

Pegunta 1:

¿Cree que el software que desarrolla cumple con las necesidades de los usuarios?

Objetivo: Identificar si las empresas desarrolladoras de software cumplen con las necesidades de los usuarios al entregar el producto

RESPUESTA	Porcentaje	Cantidad
Si	82,4%	14
No	17,6%	3
Total	100,0%	17

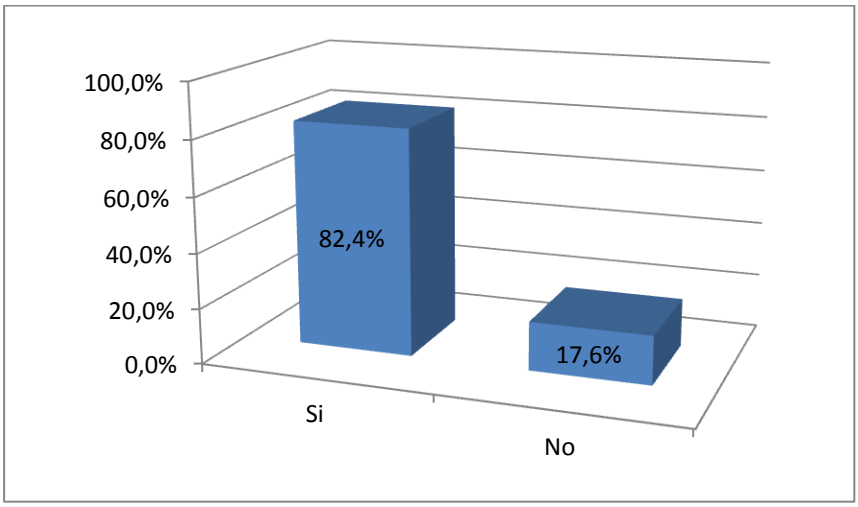


Ilustración 4 : Porcentajes de las empresas que creen que su producto cumple con las necesidades de los clientes.

Interpretación: La gráfica refleja que el 82,4 % de las personas encuestadas respondieron a que el software que desarrollan cumple con las necesidades del cliente y el 17,6% expreso que no cumplen con las necesidades de los usuarios.

Análisis: Se demuestra que la mayoría de personas encuestadas considera que desarrollan software que cumple con las necesidades de los usuarios finales.

Pregunta 2:

¿Qué fases de la ingeniería de requerimientos cubre para el desarrollo de software?

Objetivo: Identificar si se emplea las fases de la ingeniería de requerimientos en el desarrollo de software.

RESPUESTA	Porcentaje	Cantidad
Elicitación	23,6%	13
Análisis	30,9%	17
Especificación	21,8%	12
Validación	21,8%	12
Ninguna	1,8%	1
Total	100,0%	55

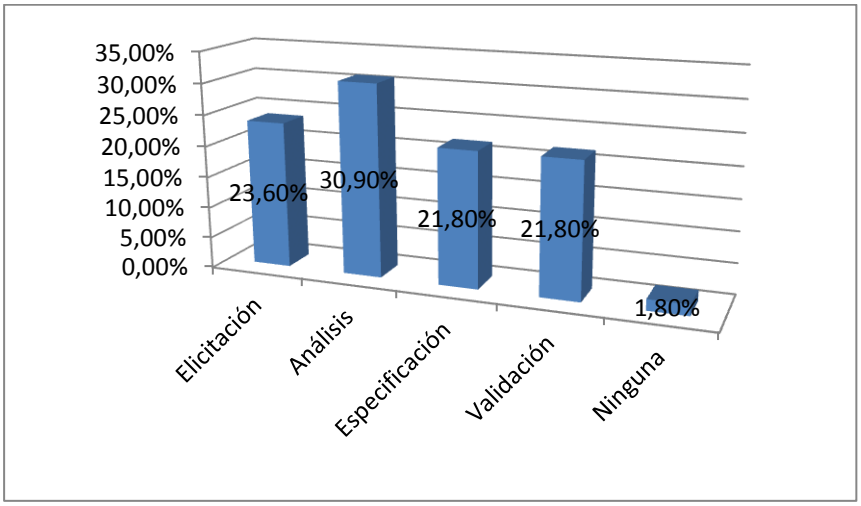


Ilustración 5: Porcentajes de uso de fases de la Ingeniería de Requerimientos

Interpretación: La grafica muestra que un 23.6 % de las personas encuestadas respondieron que aplican la fase de Elicitación de la Ingeniería de Requerimientos, un 30.9 % en la fase de análisis, un 21,8% en la fase de Especificación, un 21.8% en la Validación y un 1.8 % no aplican ninguna Fase.

Análisis: Se demuestra que la mayoría de personas encuestadas considera las cuatro fases de Ingeniería de requerimientos prestando más relevancia a la fase de análisis en el momento de desarrolla software.

Pregunta 3

¿Usted cree que la ingeniería de Requerimientos es importante?

Objetivo: Identificar la importancia de la Ingeniería de Requerimientos en el desarrollo de software.

RESPUESTA	Porcentaje	Cantidad
Si	100,0%	17
No	0,0%	0
Total	100,0%	17

Interpretación: de las 17 encuestas realizadas en su totalidad contestan que la Ingeniería de Requerimientos es importante en el desarrollo de software.

Análisis: Los motivos más relevantes expresados en las encuestas por lo cual consideran que la Ingeniería de Requerimientos son los siguientes:

- Permite obtener un desarrollo óptimo y lo más apegado a lo que el usuario necesita
- Para poder desarrollar paso a paso un proyecto
- Un requerimiento claro es la base de las fases de cualquier metodología de software.
- Mediante la Ingeniería de Requerimientos podemos descubrir la mayor cantidad de problemas.

Pregunta 4.

¿Usa algún tipo de metodología para la Ingeniería de Requerimientos?

Objetivo: Determinar el uso de metodologías para la Ingeniería de Requerimientos

RESPUESTA	Porcentaje	Cantidad
Si	35,3%	6
No	64,7%	11
Total	100,0%	17

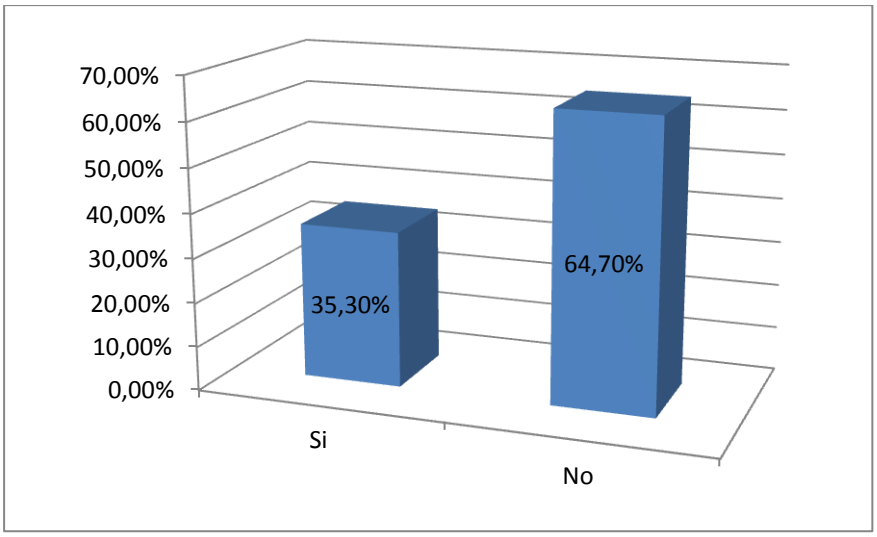


Ilustración 6: Uso de algún tipo de metodología para la Ingeniería de Requerimientos

Interpretación: La grafica muestra que un 35.3% utilizan una metodología para la Ingeniería de requerimientos, un 64.7% no utilizan ninguna técnica.

Análisis: el porcentaje de las empresas que utilizan técnicas para la Ingeniería de Requerimientos es bajo, las técnicas que estas empresas utilizan son:

- RUP[1](PROCESO UNIFICADO RATIONAL)
- Documento técnico donde se señala los requerimientos.
- Metodologías establecidas por la propia empresa.

Pregunta 5

¿Utiliza alguna técnica y/o herramienta para la elicitación de requerimientos?

Objetivo: determinar si las empresas que desarrollan software utilizan una técnica de elicitación de requerimientos.

RESPUESTA	**Porcentaje**	**Cantidad**
Si	5,9%	1
No	94,1%	16
Total	100,0%	17

[1] RUP: Forma disciplinada de asignar tareas y responsabilidades en una empresa de desarrollo (quién hace qué, cuándo y cómo).

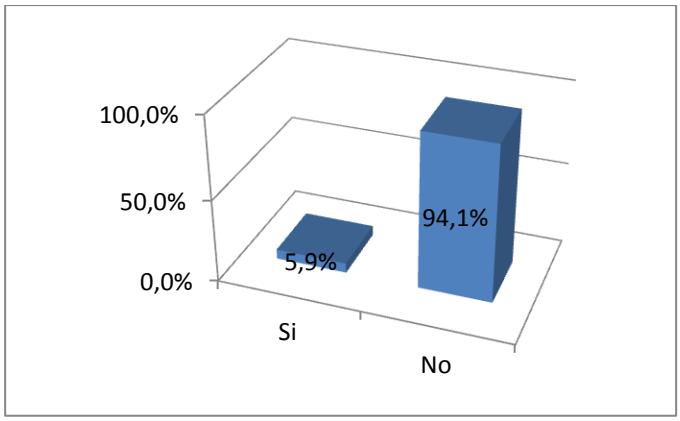


Ilustración 7: Uso de técnicas en la captura de requerimientos

Interpretación: Se puede observar en la gráfica que un 94.1 % no utilizan técnicas para la elicitación frente a un 5.9% de las empresas que si utilizan técnicas de elicitación.

Análisis: es muy alto el porcentaje de las empresas que no hacen uso de ninguna técnica para la elicitación de requerimientos. Con esto se deduce que en esta fase los desarrolladores utilizan su experiencia para la captura de requerimientos.

Pregunta 6.

En la fase de análisis de requerimientos ¿usa alguna técnica?

Objetivo Averiguar si las empresas desarrolladoras de software utilizan alguna técnica para la fase de análisis de requerimientos.

RESPUESTA	Porcentaje	Cantidad	Técnicas
Si	41,2%	7	RUP, Técnicas establecidas por la propia empresa
No	58,8%	10	
Total	100,0%	17	

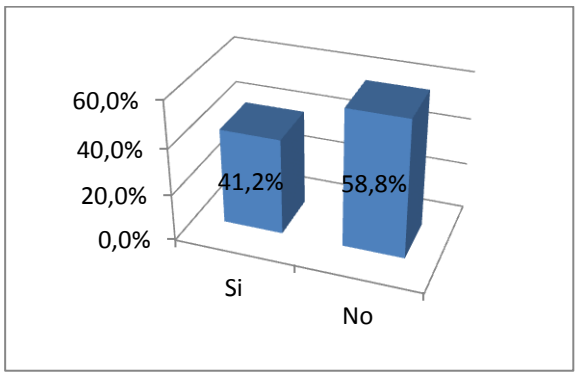


Ilustración 8: Uso de técnicas en la fase de análisis

Interpretación: En la gráfica 8 se muestra que un 58.8 % no utilizan ninguna técnica en el análisis de requerimientos y 41.2% de los encuestados utilizan técnicas como RUP, Casos de uso, Técnicas establecidas por la propia empresa.

Análisis: A diferencia de la fase de elicitación en esta fase el porcentaje de las empresas que utilizan técnicas para el análisis de requerimientos aumenta un 35%.

Pregunta 7.

Para la elaboración del documento de la Especificación de Requerimientos ¿usa algún tipo de estándar?

Objetivo: identificar si las empresas que desarrollan software utilizan alguna técnica en la fase de Especificación de Requerimientos.

RESPUESTA	Porcentaje	Cantidad
Si	41,2%	7
No	58,8%	10
Total	100,0%	17

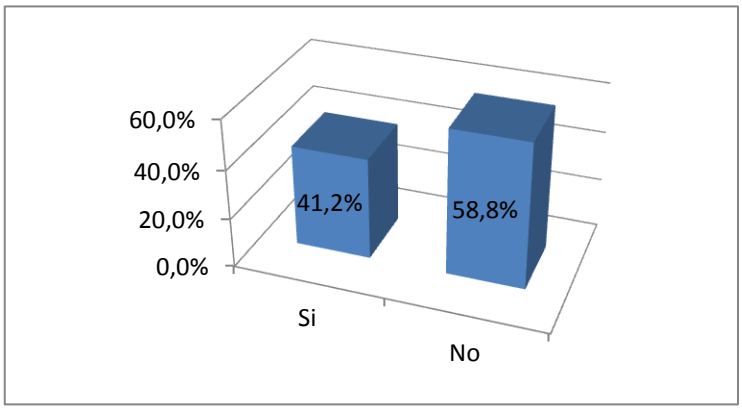


Ilustración 9: Uso de estándares en la especificación

Interpretación: se observa en la gráfica que las empresas en la mayoría con un 58.8% no utilizan técnicas para la especificación de requerimientos de software, frente a un 41.2 % que si lo hacen.

Análisis: Las técnicas para la Especificación se Requerimientos que los encuestados dicen usar son la siguientes

- RUP, UML
- Técnicas establecidas por la empresa.

Se puede recalcar que según la encuesta la metodología RUP junto con el lenguaje unificado de modelado UML, para realizar los casos de uso, son las que más se aplican.

Pregunta 8.

En la fase de validación/ verificación de requerimientos ¿Usa alguna técnica?

Objetivo: identificar las técnicas utilizadas en la verificación/validación de requerimientos

RESPUESTA	Porcentaje	Cantidad
Si	29,4%	5
No	70,6%	12
Total	100,0%	17

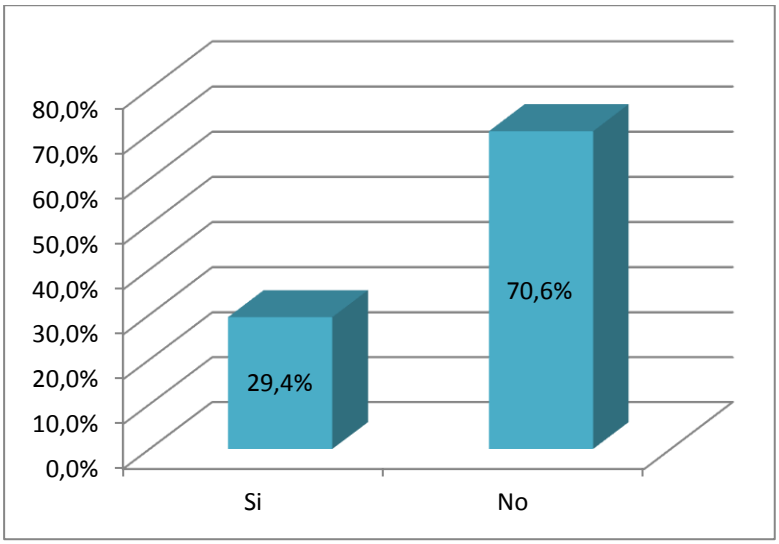


Ilustración 10. Uso de técnicas para la Validación/Verificación

Interpretación: se observa que en un 70.6 % las empresas de desarrollo de software no utilizan técnicas para la verificación de requerimientos frente a un 29.4 % que utilizan técnicas para la verificación de requerimientos.

Análisis: se puede constatar en las encuesta que la mayoría de empresas desarrolladoras de software no utilizan técnicas específicas para la fase de validación y verificación.

3.4 Procesos de Ingeniería de Requerimientos

En ingeniería de requerimiento existen procesos distintos que se complementan entre sí, estas actividades varían de un autor a otro. En este punto para nuestro estudio nos enfocaremos en los siguientes: la elicitación de requerimientos, análisis de requerimientos, especificación de requerimientos y su validación/ verificación de requerimientos, las cuales se realizan a lo largo del desarrollo y se pueden observar en la ilustración 11.

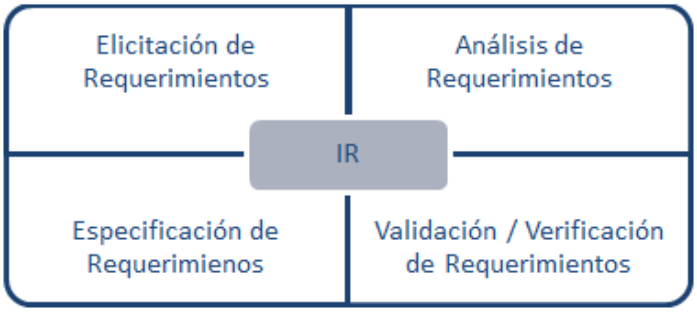


Ilustración 11: Proceso de Ingenieria de Requerimientos: Fuente: (Sommerville, 2002)

3.4.1 Elicitación de Requerimientos

La obtención de requerimientos es la consecución de los requerimientos y restricciones de los involucrados en el sistema, del dominio de la aplicación, del ambiente operacional del sistema y de la organización. En el proceso de obtención se identifican y comprenden las necesidades, problemas y restricciones de los distintos tipos de usuarios, mediante la comunicación con los clientes, usuarios del sistema y otros involucrados en su desarrollo. Para ello es importante tener conocimiento del problema, del dominio de la aplicación y del negocio.

Los requerimientos del sistema son obtenidos de diversos orígenes, por ejemplo de la consulta con los involucrados, de documentos relacionados, del conocimiento del dominio y otras como son:

- Estudios de mercado sobre las necesidades de informatización de las organizaciones similares.
- Comentarios de los usuarios sobre los sistemas actuales, los procesos de negocio o la problemática de la organización.

- Especificaciones generales preparadas por los usuarios con las necesidades básicas del área.
- Documentos de organización y procedimientos.
- Observaciones, salidas y medidas de los sistemas actuales.
- Entrevistas con los clientes y usuarios.
- Documentación de los sistemas actuales.
- Modelos y prototipos.
- Información sobre otros productos de software en el mercado que cubran necesidades similares.

3.4.2 Análisis de Requerimientos

Sobre la base de la extracción realizada previamente, comienza esta fase en la cual se enfoca en descubrir problemas con los requerimientos del sistema identificados hasta el momento. Usualmente se hace un análisis luego de haber producido un bosquejo inicial del documento de requerimientos; en esta etapa se leen los requerimientos, se conceptúan, se investigan, se intercambian ideas con el resto del equipo, se resaltan los problemas, se buscan alternativas y soluciones, y luego se van fijando reuniones con el cliente para discutir los requerimientos.

El término análisis de requerimientos se ha utilizado durante bastante tiempo para referirse al conjunto global de las actividades relacionadas con los requerimientos en el desarrollo de software, considerando que:

- Se asumía que eran proporcionados directamente por el cliente.
- No era labor del equipo de desarrollo la adquisición de dichos requerimientos
- Ni había necesidad de una validación por parte del cliente, ya que era él mismo el que los había producido.

3.4.3 Especificación de Requerimientos

En esta fase se documentan los requerimientos acordados con el cliente, en un nivel apropiado de detalle.

En la práctica, esta etapa se la realiza conjuntamente con el análisis, se puede decir que la especificación es el "pasar a limpio" el análisis realizado previamente, aplicando técnicas y/o estándares de documentación, como la notación UML (Lenguaje de Modelado Unificado), que es un estándar para el modelado orientado a objetos, por lo que los casos de uso y la obtención de requerimientos basada en casos de uso se utiliza cada vez más para la obtención de requerimientos.

3.4.4 Validación de los Requerimientos

La validación es la etapa final de la IR. Su objetivo es, ratificar los requerimientos, es decir, verificar todos los requerimientos que aparecen en el documento especificado representa una descripción, por lo menos, aceptable del sistema que se debe implementar. Esto implica verificar que los requerimientos sean consistentes y que estén completos (Somerville, 2005)

Según el IEEE se entiende por validación lo siguiente:

Verificación y validación: el proceso de determinar si los requerimientos para un sistema o componente son completos y correctos, los productos de cada fase de desarrollo satisfacen los requerimientos o condiciones impuestas por la fase previa y el sistema o componente final es acorde con los requerimientos especificados.

En él, se abordan tres aspectos fundamentales: la calidad de los requerimientos, la calidad de los productos intermedios y las pruebas de aceptación del sistema.

Cuando se está trabajando desde el punto de vista de la Ingeniería de Requerimientos (IR), entonces se puede entender por:

- **Validación de requerimientos:** conjunto de actividades encaminadas a llegar a un acuerdo entre todos los participantes en el que se ratifique que los requerimientos adquiridos y analizados representan realmente las necesidades de clientes y usuarios y que, por lo tanto, deberían llevar a la construcción de software útil.
- **Por verificación** se entiende el conjunto de actividades relacionadas con la calidad de las especificaciones de RC y RD respecto a sus propiedades deseables.

CAPITULO IV

Propuesta de la metodología para la Especificación de Requerimientos

4.1 Metodología

4.1.1 Concepto

Una metodología es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de vida de un proyecto de desarrollo.

Las metodologías se basan en una combinación de los modelos de proceso genéricos. Definen artefactos, roles y actividades, junto con prácticas y técnicas recomendadas.

La metodología para el desarrollo de software en un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito, comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que cumplimos el objetivo por el cual fue creado.

Para ingeniería del software, se puede destacar que una metodología:

- Optimiza el proceso y el producto software.
- Métodos que guían en la planificación y en el desarrollo del software.
- Define qué hacer, cómo y cuándo durante todo el desarrollo y mantenimiento de un proyecto.

Una metodología define una estrategia global para enfrentarse con el proyecto. Entre los elementos que forman parte de una metodología se pueden destacar:

- **Fases:** tareas a realizar en cada fase.
- **Productos:** E/S de cada fase, documentos.
- **Procedimientos y herramientas:** apoyo a la realización de cada tarea.
- **Criterios de evaluación**: del proceso y del producto. Saber si se han logrado los objetivos.

Una metodología de desarrollo de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Una gran variedad de estos marcos de trabajo han evolucionado durante los años, cada uno con sus propias fortalezas y debilidades. Una metodología de desarrollo de sistemas no tiene que ser necesariamente adecuada para usarla en todos los proyectos.

Una metodología de desarrollo de software o metodología de desarrollo de sistemas en ingeniería de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información.

El marco de trabajo de una metodología de desarrollo de software consiste en:

Una filosofía de desarrollo de software, con el enfoque o enfoques del proceso de desarrollo de software.

Múltiples herramientas, modelos y métodos para ayudar en el proceso de desarrollo de software.

4.1.2 Tipos de metodologías Para especificación de requerimientos

Modelo* d*e Pohl

Se trata de un modelo iterativo en el que se definen las cuatro actividades que pueden verse en la Ilustración 13. Aunque el orden de realización de las actividades puede

ser cualquiera, se asume una secuencia en la que los requerimientos son adquiridos o identificados; negociados entre los participantes; integrados con el resto de la documentación; y, finalmente, validados y verificados (Durán Toro, Amador, 2000).

En la Ilustración 13 se pueden ver indicados en el círculo exterior los roles de los participantes involucrados en el desarrollo del proyecto. Aparecen rodeando todas las etapas ya que su participación puede tener lugar en cualquiera de ellas, aunque a un rol se le puede reconocer más vinculado a la etapa donde se encuentra situado.

Ilustración 12 Modelo de Pohl: Fuente: (Durán Toro, Amador, 2000)

Metodología DoRCU

La metodología DorCU (Documentación de Requerimientos centrada en el Usuario) (M.Grisselda Báez), consta de las siguientes etapas:

- Elictación de Requerimientos
- Análisis de Requerimientos
- Especificación de Requerimientos
- Validación y Certificación de Requerimientos.

Y los objetivos que se proponen para cada una de ellas son:

Elicitación de Requerimientos.

Esta es la etapa en donde se adquiere el conocimiento del trabajo del cliente/usuario, se busca comprender sus necesidades y se detallan las restricciones medioambientales. Como resultado de las acciones realizadas se tiene el conjunto de los requerimientos en todas las partes involucradas.

Análisis de Requerimientos.

En esta etapa se estudian los requerimientos extraídos en la etapa previa a los efectos de poder detectar, entre otros, la presencia de áreas no específicas, requerimientos contradictorios y peticiones que aparecen como vagas e irrelevantes. El resultado de haber llevado a cabo las tareas que involucran estos términos puede, en más de una oportunidad, hacer que se deba regresar a la primera etapa, a los efectos de eliminar todas las inconsistencias y falencias que se han detectado. En esta etapa ya se realizan aproximaciones a un lenguaje técnico.

Especificación de Requerimientos

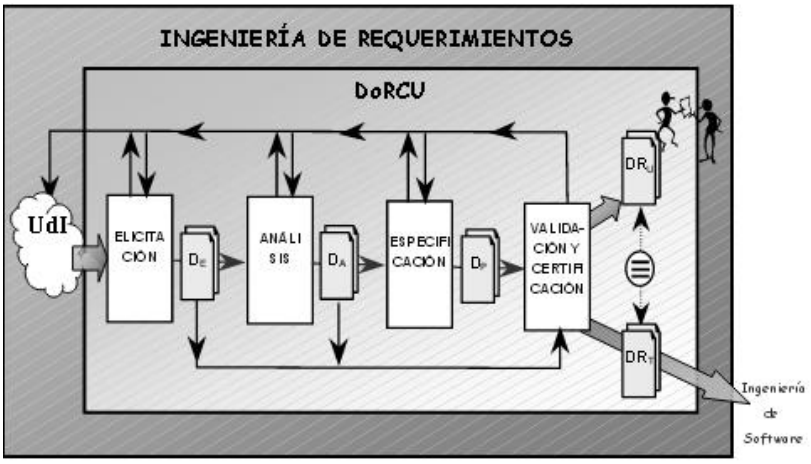
Partiendo de lo elaborado en la parte anterior tales como funciones, datos requerimientos no funcionales, objetivos, restricciones de diseño-implementación o costos e independientemente de la forma en que se realice, esta etapa es un proceso de descripción del requerimiento. Si se presentan dificultades para especificar un requerimiento se debe volver a la etapa anterior que se crea conveniente.

Validación y Certificación de los Requerimientos.

Esta etapa final se nutre de las anteriores y realiza la integración y validación final de lo obtenido en cada una de las etapas anteriores dando como resultado final el Documento de Requerimientos. Este documento no es uno solo sino que, como mínimo, existen dos que son isomórficos entre sí: uno destinado al cliente/usuario a los efectos de la certificación de los requerimientos y el otro técnico, orientado a nutrir las restantes etapas de la Ingeniería de Software. Y, al igual que en el caso

anterior, su resultado puede ser la necesidad de retomar la especificación e incluso de la elicitación, iterando entre etapas y sin perder contacto con el cliente/usuario.

Representación gráfica de la propuesta metodológica que se hace para la ingeniería de requerimientos.



Ilustracion 1 Esquema de la Metodología DoRCU, Fuente **(M.Grisselda Báez)**

Modelo espiral

En este modelo se asume una naturaleza iterativa del proceso y la dificultad de establecer un punto de terminación del mismo, dado que los requerimientos nunca llegarán a ser perfectos. El modelo asume que existe la actividad de gestión de requerimientos, que se realiza durante todo el proceso y que se encarga de gestionar la obtención incremental de los requerimientos y los inevitables cambios a los que están sujetos (Somerville, 2005)

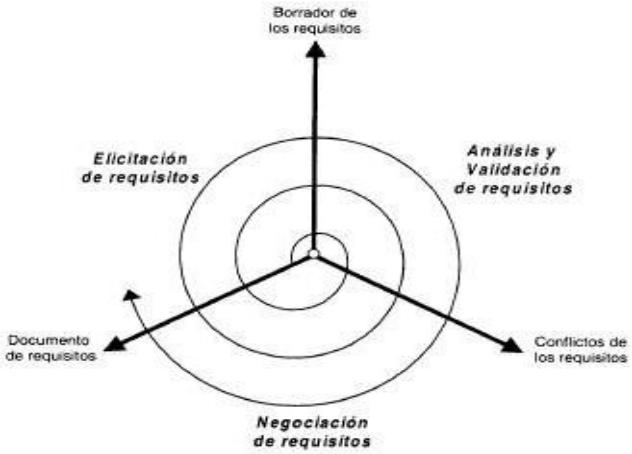


Ilustración 13 Modelo Espiral: Fuente: **(Somerville, 2005)**

Como se puede observar en la Ilustración 14, este ciclo en espiral continúa repitiendo las fases hasta que llegue un momento en que la negociación de requerimientos se considera finalizada, pues no se detectan más requerimientos que incluir y los que ya se han incluido no presentan conflictos ni contradicción. En ese momento, se daría por finalizado el documento de requerimientos y se continuaría con el resto del proceso de desarrollo. Si todavía quedaran requerimientos por identificar o negociar para resolver conflictos, se daría otra vuelta a la espiral (Somerville, 2005).

Modelo SWEBOK

El proyecto SWEBOK (*Software Engineering Body of Knowledge*) es un proyecto conjunto del IEEE y de la ACM (*Association for Computing Machinery*) para producir un cuerpo de conocimiento sobre ingeniería de software que siente las bases de dicha ingeniería como una profesión (SWEBOK, 2004).

Dentro de las diez áreas de conocimiento que se establecen en dicho proyecto, la novena corresponde a la ingeniería de requerimientos.

La Ilustración 15 recoge el proceso propuesto por SWEBOK para la ingeniería de requerimientos. En la ilustración, no se recoge una actividad horizontal encaminada a la gestión de requerimientos, tal y como se define en el modelo espiral.

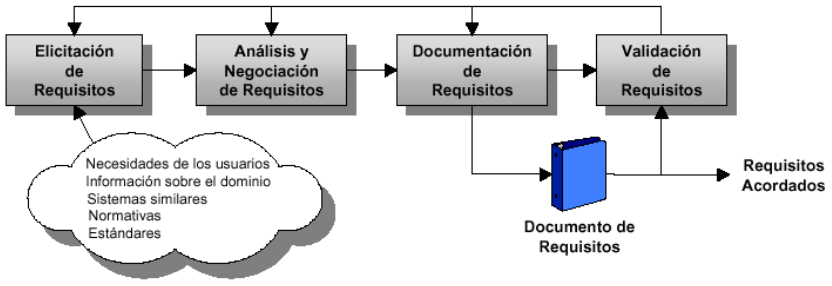


Ilustración 14 Modelo SWEBOK. Fuente: (SWEBOK, 2004)

Para concluir esta sección, comentamos que los modelos de gestión de requerimientos analizados comparten muchas fases o etapas, variando el orden de su

ejecución o la integración de unas fases en otras. En definitiva, todas conducen a la definición sistemática de un proceso a seguir. Pero en ninguno de estos modelos se especifica ninguna técnica ni método concreto para aplicar en cada una de las fases que definen. Es decir, se deja a juicio y experiencia del ingeniero de requerimientos la elección de una metodología concreta o la técnica a aplicar las fases propuestas.

4.2 Estado del Arte de las metodologías de especificación de requerimientos

El avance de las comunicaciones en los últimos años viene provocando gran interés por el desarrollo de propuestas metodológicas que presenten un marco de referencia adecuado cuando se desarrolla aplicaciones o sistemas de información.

El tratamiento de requerimientos es el proceso mediante el cual se especifican y validan los servicios que debe proporcionar el sistema así como las restricciones sobre las que se deberá operar. *"Consiste en un proceso iterativo y cooperativo de análisis del problema, documentando los resultados en una variedad de formatos y probando la exactitud del conocimiento adquirido"* (Jacobsob, 1993). Es esencial esta fase puesto que los errores más comunes y más costosos de reparar, así como los que más tiempo consumen se deben a una inadecuada ingeniería de requerimientos.

Este trabajo presenta el estado del arte y un estudio comparativo de las actuales propuestas que existen en el entorno de desarrollo de software y que utilizan diferentes técnicas y modelos para la elicitación, análisis y verificación de requerimientos.

En el proceso de desarrollo de un sistema, el equipo de desarrollo se enfrenta al problema de la identificación de requerimientos. La definición de las necesidades del sistema es un proceso complejo, pues en él hay que identificar los requerimientos que el sistema debe cumplir para satisfacer las necesidades de los usuarios finales y de los clientes. Para realizar este proceso, no existe una única técnica estandarizada y estructurada que ofrezca un marco de desarrollo que garantice la calidad del resultado. Existe en cambio un conjunto de técnicas, cuyo uso proponen las diferentes metodologías para el desarrollo de aplicaciones. Se debe tener en cuenta

que la selección de las técnicas y el éxito de los resultados que se obtengan, depende en gran medida tanto del equipo de análisis y desarrollo, como de los propios clientes o usuarios que en ella participen.

Antes de la aparición de la ingeniería de requerimientos, estos eran competencia exclusiva del Análisis de sistemas. En esta área se elaboraron algunos métodos de desarrollo estructurado como SA/SD (análisis y diseño estructurados) (De Marco, 1978) , SADT(Análisis de Sistemas y Técnica de Diseño) (Ross, 1977) o SSADM(análisis estructurado de sistema y método de diseño) (Downs et al, 1992). La idea general de todos estos métodos consiste en comenzar analizando los requerimientos mediante un enfoque divide y vencerás que permita ir fraccionando el sistema en piezas más pequeñas y después definir las funciones u objetivos que cada parte del sistema debería realizar.

El análisis de sistemas está profundamente enraizado con los sistemas de información, una comunidad centrada en las metodologías y el modelado conceptual, de modo que probablemente el concepto de análisis de los requerimientos surgió ligado a los esquemas tradicionales de descomposición funcional[2] top-down. Los requerimientos eran capturados y listados como objetivos del usuario, y después elaborados como un conjunto de funciones representadas en diagramas de flujo de datos, procesos SADT o cualquier otra técnica de modelado.

El enfoque de los métodos estructurados para el análisis de requerimientos fue desafiado en los 80 por las técnicas JAD (Joint Applications Development, Desarrollo de conjunto de aplicaciones en español) (DSMD, 1995), que trataron de superar el incómodo y lento proceso de modelado involucrando a los usuarios en el diseño a través de reuniones, tormentas de ideas y escenarios. El enfoque del análisis funcional *top-down* seria también desafiado por la comunidad partidaria de la orientación a objetos, más inclinada por modelar el dominio del problema antes que establecer objetivos y funciones. Esto allano el camino a la generación actual de métodos orientados a objetos: ingeniería de sistemas orientados a objetos (Jacobson I. , 1992), la técnica de modelado de objetos (Rumbaugh, 1991) y el análisis y diseño

[2] El diseño top-down es una herramienta que presenta en primer lugar una solución a un problema general utilizando tres o cuatro pasos solamente. Cada uno de esos pasos en la primera solución se dividen en otros subpasos. Este proceso se repite varias veces, en cada iteración se produce una solución más detallada al problema original. Cuando los pasos ya no se pueden subdividir, el algoritmo ha terminado. El diseño top-down también se conoce como descomposición funcional o refinamiento de pasos.

orientado a objetos (Coad, 1991). Más recientemente, la comunidad de la orientación a objetos definió un estándar de desarrollo con la creación de UML y el proceso unificado, aunque sin aportar nada en cuanto al análisis de requerimientos más allá de los casos de uso y escenarios. La otra influencia de la ingeniería de requerimientos la encontramos en la Ingeniería de Software, una comunidad tradicionalmente interesada en la especificación formal y la entrega de productos software fiable, a menudo para sistemas de telecomunicaciones en tiempo real y aplicaciones críticas. Los ingenieros de software se encontraron con que, a pesar de contar con detalladas especificaciones formales, algunos sistemas no eran aceptados o fallaban en circunstancias que no habían podido predecirse.

La ingeniería de requerimientos ha crecido a partir de estas áreas, a las que además ha contribuido de manera muy significativa. Asimismo, se ha centrado en investigar técnicas y herramientas que complementen los métodos de ingeniería de software.

La fundación de la ingeniería de requerimientos puede encontrarse en una colección de artículos (Thayer, 1990) y ediciones especiales de *Transactions on Software Engineering*, seguidos por las conferencias del IEEE (Filkenstein, 1993). Estos eventos revelaron una importante diversidad de temas de investigación y prácticas industriales que estaban más relacionadas con el qué construir que con el cómo construirlo. En 1993 Lubars,

Potts y Ritcher (Lubars, 1993), en el marco de una investigación sobre las prácticas de ingeniería de requerimientos en las empresas, expusieron que los requerimientos ambiguos y cambiantes constituían un problema constante y que los desarrolladores preferían soluciones organizacionales antes que tecnológicas para hacer frente a los problemas relacionados con la ingeniería de requerimientos.

4.2.1 Metodologías Especificación de Requerimientos

El desarrollo de sistemas web agrupa una serie de características que lo hacen diferente del desarrollo de otros sistemas (Koch N. , 2001). Por un lado, hay que

tener en cuenta la variada cantidad de involucrados en el proceso: analistas, clientes, usuarios, diseñadores gráficos, expertos en multimedia y seguridad, etc. Por otro lado, la existencia en estos sistemas de una importante estructura de navegación obliga a un desarrollo preciso de este aspecto que garantice que el usuario no se pierda en el momento de navegar en la aplicación. Estas ideas unidas al hecho que los sistemas web suelen tratar con múltiples medios y es esencial que ofrezcan una interfaz adecuada en cada momento, obligan a que estos aspectos propios de la web deban ser tratados de una forma especial en el proceso de desarrollo.

Estas características especiales también hay que tenerlas en cuenta en la fase de especificación de requerimientos (Escalona, 2002). Por ello, la mayoría de las propuestas estudiadas ofrecen diferentes clasificaciones de los requerimientos. Sin embargo, la terminología usada no es siempre la misma. Para facilitar la comprensión de las propuestas, antes de presentarlas, repasamos una clasificación de requerimientos relevantes en sistemas web.

Requerimientos de datos, también denominados requerimientos de contenido, requerimientos conceptuales o requerimientos de almacenamiento de información. Éstos requerimientos responden a la pregunta de qué información debe almacenar y administrar el sistema.

Requerimientos de interfaz (al usuario), también llamados en algunas propuestas requerimientos de interacción o de usuario. Responden a la pregunta de cómo va a interactuar el usuario con el sistema.

Requerimientos navegacional, recogen las necesidades de navegación del usuario.

Re**q**uerimientos de personalización, describen cómo debe adaptarse el sistema en función de qué usuario interactúe con él y de la descripción actual de dicho usuario.

Requerimientos transaccionales o funcionales internos, recogen qué debe hacer el sistema de forma interna, sin incluir aspectos de interfaz o interacción. También son conocidos en el ambiente web como requerimientos de servicios.

Requerimientos no funcionales, son por ejemplo los requerimientos de portabilidad, de reutilización, de entorno de desarrollo, de usabilidad, de disponibilidad, etc.

Son muchos los autores que han propuesto técnicas y modelos para el desarrollo de este tipo de sistemas, pero, estas propuestas se centran principalmente en las últimas fases del proceso de desarrollo. En este apartado se van a presentar aquellas técnicas que incluyen el proceso de definición de requerimientos dentro del ciclo de vida. Alguna de las propuestas que describimos cubrirá la captura y definición de requerimientos desde sus primeras propuestas.

El criterio que se ha seguido para presentar las propuestas es el cronológico, desde la primera publicación que incluye tratamiento de requerimientos. Esto permitirá en la comparativa evaluar la evolución de las propuestas para requerimientos.

WSDM: Web Site Design Method

La principal característica de esta metodología es el acercamiento a los usuarios (la audiencia o público). Esto significa que se orienta a la creación de sitios Web basados en los requerimientos de los usuarios. De esta manera WSDM aporta pautas para el hecho de que los sitios Web usualmente tienen diferentes tipos de visitantes o usuarios que tendrán diferentes necesidades. Su proceso de desarrollo se divide en cuatro fases: modelo de usuario, diseño conceptual, diseño de la implementación (De Troyer, 1997). La fase que más repercusión tiene para este trabajo es la primera en la que intenta detectar los perfiles de usuarios para los cuales se construye la aplicación. Para ello, se deben realizar dos tareas:

Clasificación de usuarios: en este paso se deben identificar y clasificar a los usuarios que van a hacer uso del sistema. Para ello, WSDM propone el estudio del entorno de la organización donde se vaya a implantar el sistema y los procesos que se vayan a generar, describiendo las relaciones entre usuarios y actividades que realizan estos usuarios.

Descripción de los grupos de usuarios: en esta segunda etapa se describen con más detalles los grupos de usuarios detectados en la etapa anterior. Para ello, se debe elaborar un diccionario de datos, en principio con formato libre, en el que indican los requerimientos de almacenamiento de información, requerimientos funcionales y de seguridad para cada grupo de usuarios.

El resto del proceso de WSDM se hacen en base a la clasificación de usuarios que se realiza en esta primera etapa ver: Figura 2.

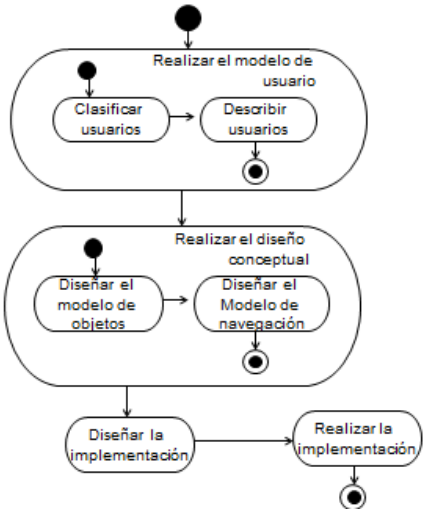


Figura 4 : Fuente: (De Troyer, 1997)

SOHDM

Es un Método de Desarrollo de Diseño en panoramas (escenarios) Orientada a Objetos en Hipermedia (*Scenario - based Object-oriented Hypermedia Design Methodology*).

Esta propuesta presenta la necesidad de disponer de un proceso que permita capturar las necesidades del sistema. Para ello, propone el uso de escenarios (Lee, 1998)

Es una de las primeras propuestas para la web y brinda más importancia a la tarea de tratamiento de requerimientos. Se caracteriza principalmente porque su ciclo de vida comienza con la aplicación de los escenarios como técnica de elicitación y definición de requerimientos.

El proceso de definición de requerimientos parte de la realización de un diagrama de contexto tal y como se propone en los diagramas de flujos de datos (DFD) de (Yourdon, 1989). En este diagrama de contexto se identifican las entidades externas que se comunican con el sistema, así como los eventos que provocan esa comunicación. La lista de eventos es una tabla que indica en qué eventos puede participar cada entidad.

SOHDM propone las siguientes 6 fases:

Análisis del dominio: se indican los límites de la aplicación que se va a desarrollar y se representan mediante un diagrama de contexto, que no es más que un diagrama de flujo de datos (DFD). En esta fase se identifican los eventos de las entidades externas que interactuarán con la aplicación, por ejemplo: una acción la cual inicia la ejecución del sistema. SOHDM propone la utilización de escenarios para identificar los requerimientos de la aplicación, estos escenarios son representados mediante los Scenarios Activity Charts(SACs[3]).

Modelo de Objetos: Los SACs generados en la fase anterior son usados para modelar objetos. Estos escenarios son transformados en objetos en forma de las tarjetas CRC (Clase-Responsabilidad-Colaboración). Los usuarios son los principales candidatos a ser objetos, donde también se toman en cuenta las actividades. Luego que los objetos son identificados estos son expresados en un documento en el cual se incluyen los atributos y asociaciones de cada uno así como la cardinalidad de la asociación.

Diseño de la vista: En esta fase los objetos son organizados en unidades de navegación, en donde cada unidad representa una vista OO. La utilización de vistas en el diseño de una aplicación Web tiene varias ventajas: la primera es que las vistas pueden soportar varios usuarios los cuales tienen diferentes requerimientos, la segunda es que las vistas pueden reducir las características cognitivas de una aplicación Web, ya que las diferentes responsabilidades y atributos de los objetos se pueden agrupar dentro de las vistas.

Diseño Navegacional: en esta fase se diseña la forma cómo los usuarios utilizan y aglutinan la información sobre la base de los escenarios. En una aplicación Web, la manera como los usuarios exploran la hipermedia es el factor más relevante dentro del proceso de diseño, ya que un buen diseño de navegación evitaría la información

[3] Los SAC describen los procesos del negocio según los tipos de usuarios que interactuarán con la aplicación.

redundante y ayuda a prevenir que el usuario se pierda en el hiperespacio. En esta fase las vistas OO y los nodos de la estructura de acceso (ASN) son adoptados por las unidades de navegación. Los ASN se utilizan para agrupar las características y funciones de los diferentes actores de la aplicación, lo cual permite a los usuarios acceder a otras partes de la aplicación. También proporcionan a los usuarios el acceso a las estructuras que pueden utilizar.

Diseño de Implementación: en esta fase se genera la estructura de la página, el flujo de la página, la interfaz de usuario, la lógica y esquema de base de datos para la construcción de un entorno de desarrollo.

Construcción: En este paso los desarrolladores generan una aplicación hipermedia, la cual cumple con todas las características especificadas en las fases anteriores.

RNA: Relationship-Navegational Analysis

RNA plantea una secuencia de pasos para el desarrollo de aplicaciones web, centrándose fundamentalmente en el flujo de trabajo de análisis (Lange, 1995). El proceso de trabajo que presenta RNA se basa en la realización de las siguientes fases:

Fase 1- Stakeholder analysis (Análisis de los interesados): el propósito de esta fase es el de estudiar las características de la audiencia. Consiste en determinar y clasificar a los usuarios finales de la aplicación en grupos según sus perfiles.

Fase 2- Elementos de interés: en esta fase se listan todos los elementos de interés de la aplicación. Por elementos de interés se entienden los documentos, las pantallas que se van a requerir, la información, etc.

Fase 3- Análisis del conocimiento: esta fase consiste en desarrollar un esquema que represente a la aplicación. Para ello RNA propone identificar los objetos, los procesos y las operaciones que se van a poder realizar en la aplicación, así como las relaciones que se producen entre estos elementos.

Fase 4- Análisis de la navegación: en esta fase el esquema obtenido en la fase anterior es enriquecido con las posibilidades de navegación dentro de la aplicación.

Fase 5- Implementación del análisis: una vez obtenido el esquema final en el que ya se encuentran incluidos los aspectos de navegación, se pasa el esquema a un lenguaje entendible por la máquina.

La propuesta de RNA es quizás una de las que más ha resaltado la necesidad de trabajar con la especificación de requerimientos, incluyendo tareas como el análisis del entorno y de los elementos de interés. Además, resulta interesante pues plantea la necesidad de analizar los requerimientos conceptuales de manera independiente a los navegacionales.

HFPM: Hypermedia Flexible Process Modeling

La propuesta de HFPM (Olsila, 1998) describe un proceso detallado que cubre todo el ciclo de vida de un proyecto software. HFPM propone un total de trece fases para las cuales se especifican a su vez una serie de tareas. Para este estudio es principalmente relevante la primera fase, denominada modelado de requerimientos, cuyas tareas son las siguientes:

- Descripción breve del problema. No indica ninguna técnica concreta pudiendo realizarse esta descripción mediante el lenguaje natural.
- Descripción de los requerimientos funcionales mediante casos de uso.
- Realizar un modelo de datos para esos casos de uso, proponiendo el uso de un modelo de clases.
- Modelar la interfaz de usuario. Para ello, propone el uso de sketches y prototipos que permitan presentar los datos al usuario.
- Modelar los requerimientos no funcionales. En éstos incluyen la navegación, la seguridad, etc.

Se puede observar que la propuesta de HFPM ofrece mayor detalle a la hora de realizar el tratamiento de los requerimientos. Sin embargo, no ofrece técnicas concretas, especialmente a la hora de trabajar con los requerimientos no funcionales.

OOHDM: Object Oriented Hypermedia Design Model / Método de Diseño Hipermedia Orientado a Objetos

Es una propuesta metodológica ampliamente aceptada para el desarrollo de aplicaciones de la web (Schwabe D., 1998). En sus comienzos no contemplaba la fase de captura y definición de requerimientos, pero actualmente propone el uso de User Interaction Diagrams (UIDs) definidos por (Vilain P. S., 2002)Esta propuesta parte de los casos de uso, que considera una técnica muy difundida, ampliamente aceptada y fácilmente entendible por los usuarios y clientes no expertos, pero que resulta ambigua para el equipo de desarrollo en fases posteriores del ciclo de vida.

Igualmente, resalta la necesidad de empezar el diseño del sistema, especialmente en los entornos web, teniendo un claro y amplio conocimiento de las necesidades de interacción, o lo que es lo mismo de la forma en la que el usuario va a comunicarse con el sistema.

Partiendo de estas dos premisas, OOHDM propone que la comunicación con el usuario se realice utilizando los casos de uso y a partir de ellos los analistas elaboran los UIDs (User Integration Diagram).

Estos UIDs son modelos gráficos que representan la interacción entre el usuario y el sistema, sin considerar aspectos específicos de la interfaz. El proceso de transformación de un caso de uso a un UIDs es descrito detalladamente en la propuesta.

OOHDM centra el desarrollo de un sistema de información web entorno al modelo conceptual de clases. Este diagrama debe surgir de los requerimientos que se definan del sistema, pero los casos de uso resultan demasiado ambiguos para ello. Así, propone refinar el proceso de desarrollo descrito en UML, de forma que de los casos de uso se generen los UIDs que concreticen más la definición de los requerimientos para, a partir de ellos, obtener el diagrama conceptual.

UWE: UML-Based Web Engineering

Propuesta metodológica basada en el Proceso Unificado (Jacobsob, 1993) y UML para el desarrollo de aplicaciones web (Koch N. , 2001). UWE cubre todo el ciclo de vida de este tipo de aplicaciones, centrando además su atención en aplicaciones personalizadas. Para este trabajo, nos interesa principalmente analizar la propuesta de captura de requerimientos de UWE. Esta metodología distingue entre la tarea de elicitar requerimientos, definir y validar los requerimientos. El resultado final de la captura de requerimientos en UWE es un modelo de casos de uso acompañado de documentación que describe los usuarios del sistema, las reglas de adaptación, los casos de uso y la interfaz.

UWE clasifica los requerimientos en dos grandes grupos: funcionales y no funcionales. Los requerimientos funcionales tratados por UWE son:

- requerimientos relacionados con el contenido.
- requerimientos relacionados con la estructura.
- requerimientos relacionados con la presentación.
- requerimientos relacionados con la adaptación.
- requerimientos relacionados con los usuarios.

Además, UWE propone como técnicas apropiadas para la captura de los requerimientos de un sistema web las entrevistas, los cuestionarios, los checklist, los casos de uso, los escenarios y el glosario para la definición de requerimientos. Para la validación propone walk-throughs, auditorías y prototipos.

W2000

W2000 (Baresi L., 2001) supone una propuesta que amplía la notación de UML con conceptos para modelar elementos de multimedia heredados de la propuesta HDM (Hypermedia Design Model). El proceso de desarrollo de W2000 se divide en tres

etapas: análisis de requerimientos, diseño de hipermedia y diseño funcional. El primero de ellos es el que resulta interesante para este trabajo.

La especificación de requerimientos en W2000 se divide en dos subactividades: análisis de requerimientos funcionales y análisis de requerimientos navegacionales. La especificación de requerimientos comienza haciendo un estudio de los diferentes roles de usuario que van a interactuar con el sistema. Cada actor potencialmente distinto tendrá su modelo de requerimientos de navegación y de requerimientos funcionales. El modelo de requerimientos funcionales es representado como un modelo de casos de uso tal y como se propone en UML. En él se representa la funcionalidad principal asociada a cada rol y las interacciones que se producen entre el sistema y cada rol. El segundo modelo consiste en otro diagrama de casos de uso pero que no representa funcionalidad sino posibilidades de navegación de cada actor. La representación gráfica es realizada con una extensión de UML

UWA: Ubiquituos Web Applications

UWA ha nacido de la colaboración entre diferentes grupos de trabajo, por lo que resulta realmente una agrupación de propuestas y técnicas. En concreto, la propuesta de W2000 se encuentra incluida en UWA. Sin embargo, W2000 ha sido incluida en UWA sólo en la fase de diseño hipermedia, siendo ambas propuestas diferentes en la fase de definición de requerimientos. Por esta razón han sido incluidos en este trabajo de forma separada.

El proceso de captura de requerimientos en UWA (Uwa Requirements Elicitation) (UWA, 2001) comienza definiendo los diferentes roles de usuario que pueden interactuar con el sistema, los objetivos globales de éste y la relación entre ellos. El proceso continúa haciendo un refinamiento de esos objetivos globales, concretándolos en sub-objetivos.

Estos sub-objetivos son estudiados y refinados para detectar conflictos entre ellos. De esta forma, se concretizan aún más dividiéndolos en requerimientos. Los requerimientos son clasificados en varios tipos: de contenido, de estructura de contenido, de acceso, de navegación, de presentación, de operaciones de usuario y de operaciones del sistema.

De esta forma, los requerimientos se van refinando hasta que solo pertenezcan a uno de estos grupos. Y finalmente los requerimientos son asignados a artefactos de diseño o a reglas de customización.

Para definir los objetivos, UWA propone una notación propia, basada en una plantilla. La definición de los actores y la relación con los objetivos se hace usando un diagrama basado en casos de uso. Por último, para definir y refinar los sub objetivos y los requerimientos, utilizan una notación gráfica propia que denominan grafo de refinamiento de objetivos, el refinamiento de este grafo permite ir representando la relación entre los requerimientos y hacer un seguimiento para validar la consecución de los objetivos del sistema. Una vez que los requerimientos son detectados, hacen uso de XML para definirlos de una manera formal.

NDT - Navigational Development Techniques

NDT (Navigational Development Techniques) (Escalona, 2004) es una técnica para especificar, analizar y diseñar el aspecto de la navegación en aplicaciones web. Para este trabajo, solo es relevante la propuesta que ofrece para la definición y captura de requerimientos. El flujo de especificación de requerimientos de NDT comienza con la fase de captura de requerimientos y estudio del entorno. Para ello, plantea el uso de técnicas como las entrevistas, brainstorming y JAD. Tras esta fase, se propone la definición de los objetivos del sistema. En base a estos objetivos, el proceso continúa definiendo los requerimientos que el sistema debe cumplir para cubrir los objetivos marcados. NDT clasifica los requerimientos en:

- Requerimientos de almacenamiento de información
- Requerimientos de actores
- Requerimientos funcionales
- Requerimientos de interacción, representados mediante:
    - Frases, que recogen cómo se va a recuperar la información del sistema utilizando un lenguaje especial denominado BNL (Bounded Natural Language) (Brisaboa, 2001).
    - Prototipos de visualización, que representan la navegación del sistema, la visualización de los datos y la interacción con el usuario.
- Requerimientos no funcionales.

Todo el proceso de definición y captura de requerimientos y objetivos que propone NDT se basa principalmente en plantillas o patrones, pero también hace uso de otras técnicas de definición de requerimientos como son los casos de uso y los glosarios. La propuesta ofrece una plantilla para cada tipo de requerimiento, lo que permite describir los requerimientos y objetivos de una forma estructurada y detallada. Algunos de los campos de los patrones son cerrados, es decir, solo pueden tomar valores predeterminados. Estos campos permiten que en el resto del proceso del ciclo de vida de NDT se puedan conseguir resultados de forma sistemática. El flujo de trabajo de especificación de requerimientos termina proponiendo la revisión del catálogo de requerimientos y el desarrollo de una matriz de trazabilidad que permite evaluar si todos los objetivos han sido cubiertos en la especificación.

La propuesta viene acompañada de una herramienta case, NDT-Tool, que facilita la cumplimentación de los patrones y que permite automatizar el proceso de consecución de resultados.

Design-driven Requirements Elicitation

Design-driven Requirements Elicitation es parte del proceso design-driven que proponen (Lowe, 2002)para el desarrollo de aplicaciones en el entorno Web La propuesta consiste en realizar la captura, definición y validación de requerimientos durante el proceso de diseño. Ello hace necesario que las actividades de diseño sean realizadas de modo que los requerimientos pueden ser tratados y administrados. El

proceso se basa en el uso de prototipos para ayudar al cliente en la exploración de las posibles soluciones y de los problemas que tienen que ser resueltos. Los usuarios o clientes definen sus requerimientos basándose en la observación o trabajo con estos prototipos. Es un proceso iterativo que consiste en reducir la incertidumbre del cliente. El ciclo tiene tres fases: evaluación, especificación y construcción.

Este proceso fue definido sobre la base de un exhaustivo análisis de "best practices" en el desarrollo de aplicaciones comerciales para el entorno Web. Esta metodología trata a todos los requerimientos de la misma manera; estos requerimientos son: de contenido, de protocolo de interfaces, de estructura navegacional, de "look and feel", de representación interna de datos, de versionamiento, de control de cambios, de seguridad, de gestión de contenido, de acceso de control, de eficiencia, de monitoreo del usuario, de soporte de funcionalidad, de adaptación del sistema, de identificación del usuario y sus derechos de acceso, etc.

Comparativa de Metodologías

Una vez enunciadas las propuestas, se presenta en este apartado una serie de estudios comparativos de las mismas. Los mismos se basan en dos aspectos. El primero analiza qué requerimientos son cubiertos en cada metodología. El segundo estudio presenta las fases dentro del proceso de tratamiento de requerimientos que cada una afronta y las técnicas que para ello proponen.

Requerimientos Utilizados en las diferentes metodologías

En base a la clasificación de los requerimientos que se realiza al principio, la primera comparativa que se realiza de las propuestas estudiadas consiste en determinar qué tipos de requerimientos contemplan cada propuesta. En la tabla 2 se presenta los diferentes requerimientos y se indica cuáles de ellos son tratados en cada metodología.

Tabla 2: Tipos de requerimientos utilizados en cada metodología:

	REQ. DE DATOS	REQ. DE USUARIO	REQ. NAVECIONALES	REQ. PERSONALIZACIÓN	REQ. TRANSACCIONALES
WSDM	X			X	
SOHDM	X	X			X
RNA	X	X	X		X
HFPM	X	X	X		
OOHDM	X	X	X		
UWE	X	X	X	X	
W2000			X	X	X
UWA	X	X	X	X	X
NDT	X	X	X	X	X
DDDP	X	X	X	X	X

Fuente: (ESCALONA, 2004)

En esta tabla las propuestas analizadas están en orden cronológico, teniendo en cuenta esto se puede observar cual ha sido el avance en cuanto al tratamiento de los requerimientos. Analizando estos resultados se puede ver que las primeras propuestas están centradas más en los requerimientos de datos e interfaz de usuario. Observando la tabla se puede notar que las propuestas resaltan la necesidad de tratar los requerimientos de personalización y navegación, así como los transaccionales de forma independiente (ESCALONA, 2004) .

TÉCNICAS CONTEMPLADAS EN CADA UNA DE LAS METODOLOGÍAS PROPUESTAS

Tabla 3: Técnicas contempladas en cada una de las metodologías propuestas

		WSDM	SOHDM	RNA	HFPM	OOHDM	UWE	W2000	UWA	NDT	DDDP
Elicitación	**Entrevistas**	X		X			X			X	X
	JAD									X	
	Brainstorming									X	
	Concept Mapping(Conceptos y relaciones (Vértices y aristas)**)**	X									
	Casos de Uso					X					
	Cuestironarios/Checklist						X				
	Prototipos										X
	Otras Técnicas		**DFD**								
Análisis	**Lenguaje Natural**	X		X	X						
	Glosarios				X		X			X	
	Patrones/Plantillas								X	X	
	Escenarios		**SAC(interacción usuario - sistema)**				X				
	Casos de Uso				X	X	X	X	X	X	
	Lenguaje Formal								X		
	Sketches de Interfaz				X						
	Prototipos										
	Otras Técnicas		**Lista Evento**			**UIDs**			**Grafo Refinamientos Requerimientos**		
Validación	**Manuales**						X			X	
	Auditorias						X				
	Matriz de Trazabilidad									X	
	Prototipos				X		X				X
	Otras Técnicas								**Grafo Refinamientos Req.**		

Fuente: (ESCALONA, 2004)

De esta tabla comparativa se pueden sacar varias conclusiones. Por un lado, es necesario indicar que dentro de la fase de captura de requerimientos, la técnica más enunciada es la de las entrevistas. Con respecto a la fase Análisis de requerimientos, se puede observar que es el aspecto central del tratamiento de requerimientos para todas las propuestas. Se puede concluir que existe una tendencia a usar la técnica de casos de uso como base. Sin embargo, ante esto conviene decir que hay dos opiniones encontradas. Algunas propuestas consideran los casos de uso una técnica óptima para representar los requerimientos, como es el caso de UWE. Sin embargo, hay otras como OOHDM o NDT que aunque indican que es una buena técnica, resaltan que es ambigua y que es necesario obtener modelos más concretos para sistematizar más el resto del proceso de ciclo de vida.

Otra conclusión muy relevante que se obtiene de este estudio es el hecho de la poca importancia que las propuestas han prestado a la validación de requerimientos. Las técnicas de validación que proponen se basan principalmente en la revisión de los modelos y resultados de la definición de requerimientos. La gran mayoría de las propuestas ni siquiera contemplan esta fase en su proceso de ingeniería de requerimientos.

4.3 Definición de la metodología a usar basada en el estándar IEEE 830-1998

4.3.1 ¿Por qué esta metodología?

La ingeniería de requerimiento es un área relativamente nueva, que antes se la realizaba dentro del ciclo de vida del desarrollo del software sin darle la importancia que esta tiene.

Existe en el mercado y en la red una gran cantidad de modelos y metodologías dirigidas a la ingeniería de requerimientos basadas en diferentes estándares, esto se debe principalmente, a que esta fase del desarrollo de software es la más importante. En

estudios realizados se afirma que la principal causa del fracaso del desarrollo de software está en los requerimientos mal obtenidos o de mala calidad.

Con la realización de esto se busca que las empresas desarrolladoras de software realicen el proceso de la ingeniería de requerimientos de una manera sencilla, entendible tanto para el usuario como para el desarrollador, evitando así futuros inconvenientes entre ambas partes ya que a través de diferentes fuentes de información se obtendrá un documento validado tanto por el usuario como por el desarrollador, detallando claramente sus necesidades.

Esta metodología busca el éxito del proceso de ingeniería de requerimientos, y para obtener esta meta se debe tener en cuenta la calidad con que se desarrolla el mismo.

La metodología que se va a describir más adelante está basada en el estándar IEEE 830, el uso de este estándar se debe a que es sencillo de usar, muy conocido, y sobretodo detalla rápidamente lo más importante y necesario de la ingeniería de requerimientos.

4.3.2 Técnicas de Ingeniería de Requerimientos.

Existen varias técnicas para la ingeniería de requerimientos, en esta parte se mencionaran algunas de las más importantes. Cada una de estas se puede aplicar en una o más actividades de la ingeniería de requerimientos.

Entrevistas

Es la técnica más la simple y más utilizada para la educción de requerimientos. Las entrevistas se utilizan para obtener información verbal, principalmente cara a cara, a través de preguntas que propone el ingeniero a uno o más interesados. Existen dos tipos de entrevistas, las estructuradas y las no estructuradas. Siendo el grado de detalle buscado más específico o más general respectivamente, por lo que normalmente se

utilizan primero las no estructuradas y luego las estructuradas para profundizar sobre los conocimientos adquiridos (Pytel, y otros, 2009). En esta técnica se pueden identificar tres fases: la preparación, la realización y el análisis de la información obtenida (Durán & Bernádez, 2002).

Cuestionarios

Son esencialmente una entrevista escrita en papel u otro medio que la persona debe responder. La diferencia es que como el interesado puede tomarse más tiempo para responder, en un momento y lugar adecuado. Además, el cuestionario tiene la ventaja que puede ser distribuido a un gran número de personas que pueden estar ubicadas en lugares remotos. Se los suelen utilizar en las etapas tempranas de la educción para recolectar rápidamente de grupos numerosos.

Joint Application Development (JAD)

En español Desarrollo Conjunto de Aplicaciones, es elaborada por IBM en 1997, esta técnica consiste en reuniones en grupo las cuales duraría de 2 a 4 días, en las cuales se ayuda a los clientes y a los usuarios a formular problemas y explorar posibles soluciones.

Grabaciones de video y de audio

Básicamente existen dos formas de utilizar las grabaciones: como registro y apoyo de las entrevistas, y para analizar algún proceso en particular. En cuanto a su función de apoyo, es importante porque permite centrar la atención en la entrevista en sí, en vez de distraerse tomando notas de todo lo que se dice. Cuando se trata de analizar algún proceso en particular, su ayuda es inestimable (sobre todo las filmaciones de video) porque permite ver y analizar en detalle ese proceso la cantidad de veces que sea necesario.

Brainstorming

Conocida como tormenta de ideas, el objetivo de esta técnica es la generación de ideas en un ambiente libre de críticas o juicios, en esta técnica participan de 4 a 10 personas.

Prototipado

Se usa para los procesos de elicitación donde existe una gran incertidumbre sobre los requerimientos, o donde es necesaria una realimentación rápida. Por ejemplo, se puede usar un prototipo para producir una discusión en una técnica de elicitación en grupo, o como base para un cuestionario.

Casos de Uso

Aunque inicialmente se desarrollaron como técnica para la definición de requerimientos (Jacobson I. , 1995), algunos autores proponen casos de uso como técnica para la captura de requerimientos (Pan, 2001). Los casos de uso permiten mostrar el contorno (Actores) y el alcance (Requerimientos Funcionales expresados como casos de uso).

Como técnica de definición de requerimientos es como más ampliamente han sido aceptados los casos de uso. Actualmente se ha propuesto como técnica básica del proceso RUP (Kruchten, P, 1998). Sin embargo, son varios los autores que defienden que pueden resultar ambiguos a la hora de definir los requerimientos de un sistema. Un caso de uso describe la secuencia de interacciones que se producen entre el sistema y los actores del mismo para realizar determinada función. Los actores son elementos externos (personas, otros sistemas, etc.)

La ventaja esencial de los casos de uso es que resultan muy fáciles de entender para el usuario o cliente sin embargo carecen de la precisión necesaria (Vilain P. S., 2000) si no se acompañan con una información textual.

Plantillas usadas para el proceso de Ingeniería de requerimientos

Una técnica o estrategia usada para realizar el proceso de ingeniería de requerimientos son las plantillas que permiten organizar requerimientos, usuarios, objetivos de manera uniforme, en el siguiente punto en la fase de análisis, se darán a conocer estas plantillas.

A continuación se muestra en una tabla las herramientas y las fases en las que son usadas cada una de estas.

Tabla 44: Herramientas usadas en cada fase de la IR

Herramientas	Elicitación	Análisis	Especificación	Validación
Entrevistas y Cuestionario	X			
Sistemas Existentes	X	X		
Grabaciones de video y de audio.	X	X		
Brainstorming (Tormenta de Ideas)	X	X		
Sistemas Existentes	X	X		
Arqueología de Documentos	X	X		
Observación	X			
Prototipo Bosquejado	X	X	X	
Prototipo Tangible/usable	X		X	X
FODA		X		
Modelo de clase conceptual		X	X	
Diagrama de actividad		X	X	
ESRE	X	X	X	X
Casos de uso	X	X	X	X
Checklist	X			X

4.3.3 Metodología Propuesta

Luego de haber analizado y estudiado varios materiales e información acerca de este tema, se propone una metodología iterativa e incremental para la ingeniería de

requerimientos esta se apoya en definiciones de autores como son (Somerville, 2005), (Durán & Bernádez, 2002).

Esta metodología tiene cuatro etapas: Elictación (Recolección/obtención) de requerimientos, análisis, especificación y validación de requerimientos.

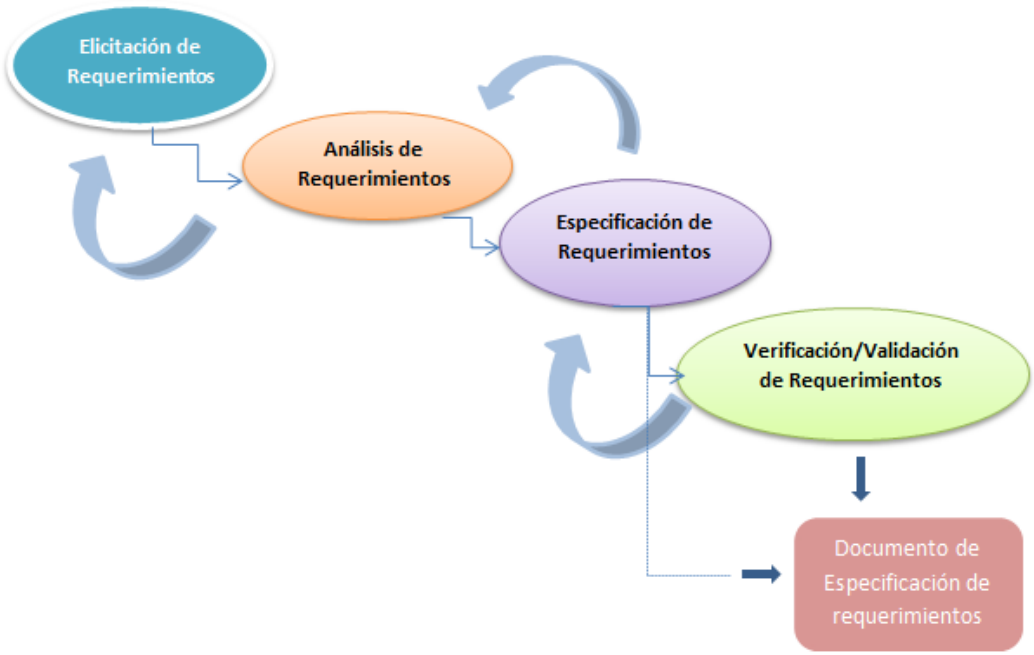


Ilustración 15: Etapas de la metodologia propuesta **Autores:** Valeria Cuji **Fuente: (Somerville, 2005)**

En el grafico se puede observar las etapas que tendrá la metodología, si se detecta algún error o fallo en una etapa se regresara a la etapa anterior en busca del posible error, hasta llegar a la etapa de elicitación en donde se revisará los datos recolectados.

Cada etapa mencionada anteriormente posee una sub-etapa las mismas se muestran en la siguiente tabla.

Ilustración 16: Metodología propesta Autor: Valeria Cuji, Daniel Borja

Fase de Elictación:

La elicitación es la parte más importante del proceso de ingeniería de requerimientos En esta fase se tiene como objetivo principal, manifestar de cualquier fuente de información proporcionada por el cliente, los procesos de la organización que permitirá identificar las necesidades de los futuros usuarios del sistema a desarrollar.

Para esta fase se tienen las siguientes actividades:

Tarea I: Realizar el estudio inicial

En esta actividad se pretende descubrir cuál es el problema que se quiere resolver, y de esta manera poder identificar los límites del sistema que se va a construir.

Antes de realizar la recolección de requerimientos es necesario conocer las características principales del negocio tal como el vocabulario que se utiliza en el mismo. Esto es importante ya que en esta tarea se planificaran las reuniones con el cliente y la falta de conocimiento de las características de su actividad hará que los desarrolladores no entiendan las necesidades, lo que provocaría la recolección errónea de requerimientos.

Además, se identificará a las diferentes clases de usuarios, sus características y los roles que cada uno de estos cumple en la organización.

Para realizar esta tarea se recomienda empezar las entrevistas o investigación preferiblemente con los líderes funcionales, esto se debe a que estas personas son las que comprenden el dominio del problema y además tienen una visión de los procesos y se continuará con los usuarios finales ya que son los que aportaran información detallada acerca del entorno organizacional lo que permitirá conocer la situación actual.

En esta tarea se tendrá:

Tabla 4 Técnicas/Herramientas para el desarrollo del estudio inicial

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
Investigación bibliográfica para el conocimiento del negocio. Técnicas de Elicitación: como entrevistas, cuestionarios, etc. Plantillas como la de Datos de los participantes.	Información recolectada.	Introducción: Ámbito del Sistema Participantes: Cliente, Desarrolladores, Usuarios del sistema. Sistema Actual.

Tarea II.- Identificar los objetivos del sistema

Luego del primer análisis a la organización se ha adquirido conocimiento acerca del funcionamiento de la misma. Esta tarea busca conocer los objetivos que tiene la empresa, esto abarca los objetivos del negocio como las necesidades que tiene el cliente, estas necesidades deberán ser expuestas de manera que expresen lo que se espera que haga el sistema, así como los resultados que se esperan lograr a corto, mediano y largo plazo con este. Esto permitirá conocer requerimientos relacionados con la escalabilidad.

En esta tarea se tendrá:

Tabla 5: Técnicas/Herramientas para identificar los Objetivos del sistema

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
• Técnicas de Elicitación: como entrevistas, cuestionarios. • Uso de plantillas-L para objetivos.	Información recolectada.	• Objetivos del sistema • Requerimientos • Restricciones • Alcance del proyecto • Participantes del proyecto.

Tarea III.- Identificar requerimientos.

Con la información obtenida previamente en los pasos 1 y 2, se procederá a identificar los requerimientos del sistema. En este paso se debe tener en cuenta que para los usuarios es difícil expresar las necesidades que tienen, es por esta razón que el analista por medio de preguntas a los usuarios ira construyendo los requerimientos del sistema.

También se pueden identificar los requerimientos del usuario al descomponer los objetivos del sistema que se recogieron en la tarea anterior.

Tabla 6: Técnicas/herramientas para identificar requerimientos

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
• Técnicas de Elicitación: como entrevistas, cuestionarios, etc.	Información recolectada por el analista	Requerimientos de Interfaces, Requerimientos funcionales y no funcionales.

- ***Tarea IV.- Priorizar los requerimientos.***

Este paso es necesario para mantener organizados los requerimientos en función de las necesidades del cliente y a la importancia que el mismo le de cada requerimiento. Esto es necesario para mantener orden tanto en los procesos de la ingeniería de requerimientos como en el de desarrollo de software.

A los requerimientos se le asignaran las siguientes calificaciones, según lo que el cliente decida: ALTA, MEDIA, BAJA, o POR DEFINIR si es que el usuario no supiera todavía.

Tabla 7:Tècnicas para Priorizar requerimientos.

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
Técnicas de elicitación como entrevistas, cheklist.	No hay entradas	Requerimientos tanto funcionales como los no funcionales.

Fase de Análisis:

Tarea V. Clasificar los requerimientos

Para clasificar los requerimientos, primero se tomara en cuenta los requerimientos generales de la interfaz, es decir aquellos requerimientos relacionados con la interacción de los usuarios, hardware, software y *comunicación*. También se clasificara a los requerimientos funcionales, la clasificación se fundamenta en el concepto que se menciona en el *cap. 3, sección 3.1.4* en donde aclara que "*un requerimiento funcional es aquel que concentra las operaciones del tratamientos de la información que realiza el sistema, tales como: el almacenamiento de la información, generación de informes, cálculos, estadísticas, operaciones, etc.",* sin considerar las restricciones físicas (Somerville, 2005) .

Los requerimientos serán clasificados también como no funcionales, como su nombre lo dice no están relacionados con las necesidades en cuanto a que va a realizar el sistema, este tipo de requerimientos se los clasificará de acuerdo a los atributos del sistema como son: rendimiento, disponibilidad, seguridad, fiabilidad, adaptabilidad, funcionabilidad, entrega, implementación, etc. Cada una de estas características están más detalladas en el cap. 3, sección 3.1.4.

En esta tarea se tendrá las siguientes plantillas.

Tabla 8: Requerimientos de Interfaz,

Requerimientos comunes de los Interfaces			
Usuario	**Hardware**	**Software**	**Comunicación**

Tabla 9: Requerimientos Funcionales, Autor: Daniel Borja, Valeria Cuji

Requerimientos Funcionales	
RF1	
RF2	
RF..n	

Tabla 10: Requerimientos No Funcionales: Autor: Daniel Borja, Valeria Cuji.

Requerimientos No Funcionales					
Rendimiento	**Seguridad**	**Disponibilidad**	**Comunicación**	**Mantenibilidad**	**Portabilidad**

Realizada la clasificación de los requerimientos se modela los diagramas de casos de uso de los requerimientos funcionales, luego se realiza la descripción de caca requerimiento funcional y no funcional.

MODELO DE CASOS DE USO A PARTIR DE LOS REQUERIMIENTOS FUNCIONALES

El objetivo principal del Modelo de Casos de Uso (MDC) es la especificación de actores y casos de uso y el establecimiento de las relaciones que entre ellos se producen. El modelado de requerimientos utiliza los elementos del MDC propuesto por Jacobson, bajo el esquema conceptual y notacional definido en UML (Jacobsob, 1993).

Estableciendo los casos de uso mediante los requerimientos ordenados anteriormente se realiza la especificación de casos de uso para esto utilizamos la siguiente plantilla:

Tabla 11: Plantilla de requerimientos funcionales

<table>
<tr><td>RF-1</td><td colspan="3"><Nombre del caso de Uso></td></tr>
<tr><td>Versión</td><td><Núm. Versión></td><td>Fecha</td><td><Fecha del caso de uso ></td></tr>
<tr><td>Autores</td><td colspan="3"><Autor del caso de uso></td></tr>
<tr><td>Fuentes</td><td colspan="3"><Fuente del Caso de uso></td></tr>
<tr><td>Descripción</td><td colspan="3"><Descripción del Caso de Uso></td></tr>
<tr><td>Actor</td><td colspan="3"><Actor></td></tr>
<tr><td>Precondición</td><td colspan="3"><lo que pasa antes de que este caso de uso suceda></td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td><Descripción de la acción></td></tr><tr><td>..</td><td></td></tr><tr><td></td><td></td></tr></table></td></tr>
<tr><td>Post Condición</td><td colspan="3"><Lo que sucede luego que termine el caso de uso></td></tr>
<tr><td>Excepciones</td><td colspan="3"><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td><Descripción de la acción></td></tr></table></td></tr>
<tr><td>Prioridad</td><td colspan="3"><baja><media><alta></td></tr>
<tr><td>Estado</td><td colspan="3"><Estado del requisito según el ciclo de vida adoptado por el proyecto></td></tr>
<tr><td>Estabilidad</td><td colspan="3"><baja><<media><alta></td></tr>
<tr><td>Comentarios</td><td colspan="3"><Comentario para el caso de uso></td></tr>
</table>

Autor: Duran Toro: Fuente: (Durán & Bernádez, 2002)

El significado de los campos específicos de esta plantilla es el siguiente:

- ***Identificador y nombre descriptivo:*** cada requerimiento debe identificarse por un código único y un nombre descriptivo. Con objeto de conseguir una rápida identificación, los identificadores de los requerimientos funcionales empiezan con RF y el nombre descriptivo suele coincidir con el objetivo que los actores esperan alcanzar al realizar el caso de uso. Por ejemplo: *"Registrar un nuevo alumno o consultar las calificaciones de alumnos"*
- ***Autores, Fuentes:*** estos campos contienen el nombre y la organización de los autores (normalmente desarrolladores) y de las fuentes (clientes o usuarios), de la versión actual del objetivo, de forma que la rastreabilidad pueda llegar hasta las personas que propusieron la necesidad del requisito.

- ***Descripción*:** para los requerimientos funcionales, este campo contiene un Patrón lingüístico que debe completarse de forma distinta en función de que el caso de uso sea abstracto o concreto.
- ***Precondición*:** en este campo se expresan en lenguaje natural las condiciones necesarias para que se pueda realizar el caso de uso.
- ***Actor:*** Los actores representan un tipo de usuario del sistema. Se entiende como usuario cualquier cosa externa que interactúa con el sistema. No tiene por qué ser un humano, puede ser un sistema informático, unidades organizativas o empresas.
- ***Secuencia normal:*** este campo contiene la secuencia normal de interacciones del caso de uso. En cada paso, un actor o el sistema realiza una o más acciones, o se realiza (se incluye) otro caso de uso. Un paso puede tener una condición de realización, en cuyo caso si se realizara otro caso de uso se tendría una relación de extensión. Se asume que, después de realizar el último paso, el caso de uso termina.
- **Postcondición:** en este campo se expresan en lenguaje natural las condiciones que se deben cumplir después de la terminación normal del caso de uso.
- ***Excepciones*:** este campo especifica el comportamiento del sistema en el caso de que se produzca alguna situación excepcional durante la realización de un paso determinado. Después de realizar las acciones o el caso de uso asociados a la excepción (una extensión), el caso de uso puede continuar la secuencia normal o terminar, lo que suele ir acompañado por una cancelación de todas las acciones realizadas en el caso de uso dejando al sistema en el mismo estado que antes de comenzar el caso de uso, asumiendo una semántica transaccional.

- **Importancia, Urgencia:** estos campos indican respectivamente la importancia y la urgencia de la resolución del conflicto.
- **Estado:** este campo indica el estado del requerimiento desde el punto de vista de su desarrollo. El Requerimiento puede estar en construcción si se está

elaborando, pendiente de negociación si tiene algún conflicto asociado pendiente de solución, pendiente de validación si no tiene ningún conflicto pendiente y está a la espera de validación o, por último, puede estar validado si ha sido validado por clientes y usuarios.

- **Estabilidad:** este campo indica la estabilidad del Requerimiento, es decir una estimación de la probabilidad de que pueda sufrir cambios en el futuro. Esta estabilidad puede indicarse mediante un valor numérico o mediante una expresión enumerada como alta, media o baja o PD(Por Determinar) en el caso de que aún no se haya determinado. La información sobre la estabilidad, bien a nivel de objetivos o bien a nivel de requerimientos, ayuda a los diseñadores a diseñar software que prevea de antemano la necesidad de posibles cambios futuros en aquellos aspectos relacionados con los elementos identificados como inestables durante la fase de ingeniería de requerimientos, favoreciendo así el mantenimiento y la evolución del software (Brackett , 1990).
- ***Comentarios*:** cualquier otra información sobre el objetivo que no encaje en los campos anteriores puede recogerse en este apartado.

La combinación de Casos de Uso se modela estableciendo relaciones entre ellos. Con esta finalidad, en la fase de Ingeniería de Requerimientos se utilizan relaciones de extensión e inclusión.

Extensión

Se caracteriza por estar establecida entre dos casos de uso y se define como una relación dirigida que indica que una instancia de un caso de uso(Caso de uso base) puede ser incrementada con la estructura y comportamiento definido en otro caso de uso (el caso de uso que extiende). Un caso de uso puede extender muchos casos de uso y, un caso de uso puede ser extendido por más de un caso de uso.

La relación extensión no implica que cada uno de los casos de uso que participan de la relación pierda toda o parte de su funcionalidad ni que ninguno de los casos de uso dependa del otro. Cada caso de uso tiene vida propia y podrían ser ejecutados por separado, sin la necesidad de que la relación se concrete.

Include.

En términos muy simples, cuando relacionamos dos casos de uso con un “include”, estamos diciendo que el primero (el caso de uso base) incluye al segundo (el caso de uso incluído). Es decir, el segundo es parte esencial del primero. Sin el segundo, el primero no podría funcionar bien; pues no podría cumplir su objetivo. Por ejemplo, el caso de uso “Cobrar Renta” está incluido en el caso de uso “Rentar Video”, o lo que es lo mismo “Rentar Video” incluye (<<include>>) “Cobrar Renta”.

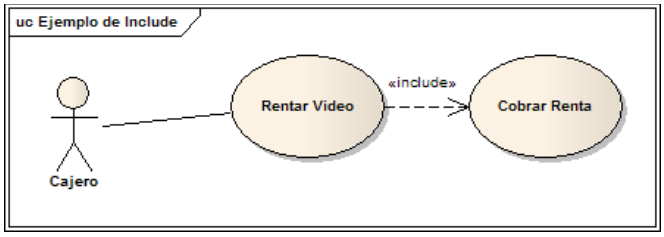


Ilustración 17: Ejemplo relación Include

La polémica al querer seleccionar una de las dos relaciones es que en el “extend” también podemos ver, desde la perspectiva del usuario, a los dos flujos como si fueran uno sólo. Y en ciertos escenarios el caso de uso base no podría cumplir su objetivo si no se ejecutara la extensión. Pero, una de las diferencias básicas es que en el caso del “extend” hay situaciones en que el caso de uso de extensión no es indispensable que ocurra, y cuando lo hace ofrece un valor extra (extiende) al objetivo original del caso de uso base. En cambio en el “include” es necesario que ocurra el caso incluido, tan sólo para satisfacer el objetivo del caso de uso base. Ejemplo: Puedes “Realizar Venta” sin “Acumular Puntos de Cliente VIP”, cuando no eres un cliente VIP. Pero, si eres un cliente VIP sí acumularás puntos. Por lo tanto, “Acumular Puntos” es una extensión de “Realizar Venta” y sólo se ejecuta para cierto tipo de ventas, no para todas.

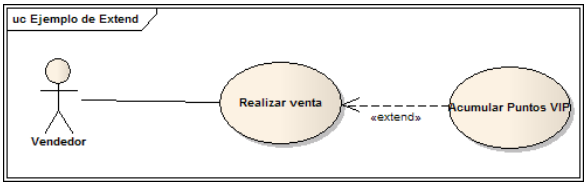


Ilustración 18: Ejemplo Relación Extends

Descripción de requerimietos no funcionales

Se utilizara la siguiente plantilla

Tabla 12: Plantilla y Patrones lingüísticos para los requerimientos no funcionales.

RNF-<id>	*<Nombre descriptivo>*	
Versión	*<Nro delaversión>*	*<Fecha de la versión>*
Autores	*<Autor de la versión>*	
Fuentes	*<Fuente de la Versión>*	
Descripción	*El sistema debera <capacidad del sistema>*	
Prioridad	*<Prioridad del requerimiento>*	
Estado	*<Estado del requerimiento>*	
Estabilidad	*<Estabilidad del requerimiento>*	
Comentarios	*<Comentarios adicionales sobre el requerimiento>*	

Fuente (Durán & Bernádez, 2002)

Descripción de los campos de la plantilla y los patrones lingüísticos.

- ***Identificador y nombre descriptivo*:** este campo tiene el mismo significado que en la plantilla para los requerimientos funcionales, pero en este caso el identificador comienza como RNF.
- ***Versión, fecha, autores, fuentes:*** estos campos tienen el mismo significado que en el patrón de requerimientos Funcionales.
- ***Descripción:*** describe el significado del requerimiento.
- *Importancia, urgencia, estado, estabilidad y comentarios:* estos campos tienen el mismo significado que en el patrón de la plantilla anterior (Plantilla y Patrones Lingüísticos para Requerimientos Funcionales).

En esta tarea se tendrá:

Tabla 13. Técnicas/Herramientas para la clasificación de requerimientos.

TECNICAS /HERRAMIENTAS	**Entrada**	**SALIDAS**
Tablas Clasificación de Requerimientos Diagramas de caso de uso Plantilla de descripción de Casos de Uso.	Documento Lluvia de ideas Objetivos del sistema	Casos de Uso (Diagramas y plantillas)

Tarea VI. Identificación de conflictos.

En esta tarea identificaremos los conflictos existentes entre los requerimientos del sistema, para esto se utilizará una tabla que permitirá comparar todos los requerimientos con las características de un buen requerimiento mencionado en la sección 3.1.3 de este documento.

En esta en la tabla se tomara en cuenta los siguientes aspectos:

- No ambiguo, completo y correcto.
- Medible y verificable
- Alcanzable y realista.
- Consistente (no contradictorio)
- Que no solape a otro.

La forma de llenar la siguiente tabla la(**tabla 13**) será: se va a comparar cada uno de los requerimientos con las características mencionadas anteriormente en caso de que el requerimiento no cumpla con la característica establecida en la tabla consideramos que existe un conflicto de manera que precederemos a registrar este conflicto en la tabla 15. Registrado el conflicto buscaremos una solución a los conflictos, esto se tratara más adelante.

Revisado cada uno de los requerimientos se registrara en la siguiente tabla:

Tabla 14: Matriz de Comprobación de características,

Requerimiento	No ambiguo, completo y correcto	Medible y Verificable	Alcanzable y realista	Consistente(No contradictorio)	No solape a otro requerimientos
R1					
R2					
…					
R..n					

Autor: Daniel Borja, Valeria Cuji.

En esta tarea se tendrá:

Tabla 15: Técnicas/Herramientas para Identificar Conflictos.

TECNICAS /HERRAMIENTAS	Entrada	SALIDAS
Tabla de Comparación de características de requerimientos. Experiencia del Analista	Documento con Requerimientos Clasificados.	Documento conflictos

Tarea VII. Negociar y solucionar conflictos.

Una vez identificado los conflictos en los requerimientos, procedemos a dar solución a los mismos, necesitaremos identificar a cada usuario que enuncio el requerimiento para esto utilizaremos lo que se conoce como trazabilidad hacia atrás esta trazabilidad consiste en relacionar cada requerimiento con su origen es decir con el responsable de crear el mismo. En el momento en que el analista detecte conflictos debe iniciar el proceso de negociación con los usuarios involucrados para esto el analista debe intentar que los usuarios se pongan de acuerdo a cerca de una solución que elimine dicho conflicto y que satisfaga a ambos. Durante la negociación el analista actuara como individuo neutral, es decir no se inclinara a favor de ninguna de las soluciones que los usuarios propongan, en caso de no llegar a un acuerdo mutuo, el analista acudirá al cliente del sistema e informara sobre el conflicto, de modo que el cliente utilice sus propios recursos para poner de acuerdo a los usuarios o alternativamente el mismo decidirá que requerimientos se implementara y cuáles no. Esto es lo que se denomina resolución por decreto, a pesar de que esta técnica es muy efectiva causa malestar en los usuarios por lo que debe ser evitada siempre que sea posible. Hay que tener en cuenta que el resultado de una solución de conflictos entre los requerimientos son nuevos requerimientos, estas soluciones se deben ir registrando en las columna soluciones de la **tabla 15**, de manera que sea fuente de información que favorezca a la trazabilidad de los requerimientos anteriores por ejemplo: “por el conflicto entre el requerimiento 1a y requerimiento 2c se ha obtenido la solución “x” o un nuevo requerimiento x”.

Tabla 16 Conflictos y soluciones.

Requerimiento	Motivo del conflicto	Solución

Autor: Daniel Borja, Valeria Cuji

En esta tarea se tendrá:

Tabla 17.Técnicas/Herramientas para solucionar y negociar conflictos.

Técnicas/Herramientas	Entrada	Salida
Trazabilidad hacia atrás Resolución por decreto	Tabla de conflictos	Tabla de solución de conflictos o nuevos requerimientos

Fase de Especificación

Tarea IX.- Realizar el Documento de Especificación de requerimientos de software (ERS)

En esta tarea se realizara la documentación de Especificación de Requerimientos de Software en el cual se describe de manera clara y precisa cada uno de los requerimientos esenciales (funcionalidad, rendimiento, restricciones de diseño y atributos de calidad (relacionados con los requerimientos no funcionales), de un software y sus interfaces externas.

Fase de Validación/Verificación

El objetivo principal de esta actividad es detectar conflictos, omisiones de requerimientos, ambigüedades y demás problemas que podrían haberse producido en las fases anteriores. Es necesaria esta etapa también ya que permitirá comprobar que cada requerimiento recolectado siga las normas de calidad establecidas.

Tarea X.-Validar Requerimientos (Funcionales y no Funcionales)

En esta tarea es necesaria ya que se va a validar los requerimientos elicitados y analizados para tener la certeza de que estos cumplan con las verdaderas necesidades de los usuarios y que se va a llegar a cumplir con el producto deseado. Si estos

requerimientos no cumplen con lo que el cliente/usuario necesita se identificara los errores que existan para su posterior negociación y corrección, esto se realizara con la información registrada en el documento de especificación de requerimientos de software.

Tabla 18.Técnicas/Herramientas Validar Requerimientos:

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
Revisión de Requerimientos Entrevistas Revisiones en grupo	Documento de Especificación de requerimientos	Lista de problemas. Lista de acciones.

Para la validación de los requerimientos se utilizará el siguiente cheklist con sus respectivas preguntas.

Tabla 19: Plantilla para validación de requerimientos

Nombre del Proyecto	
Nombre del Documento	
Fecha	

Criterio	Sí	No	NA
1. Correctitud — La Especificación de un Requerimiento es correcta si, y solo si, el sistema/software alcanza todos y cada uno de los requerimientos en él especificados.			
Desde el punto de vista del usuario, ¿se ha especificado el tiempo de respuesta esperado de todas las operaciones necesarias?			
¿Se han especificado otras consideraciones temporales tales como el tiempo de procesamiento, el de transferencia de datos o la tasa de transferencia?			
¿Se han especificado todas las tareas que debe realizar el sistema/software?			
Para cada tarea especificada, ¿se ha detallado el contenido de datos/información utilizado por la tarea y el contenido de datos/información que se obtendrá como resultado de la misma?			
¿Se han establecido los requerimientos sobre la seguridad física?			
¿Se han establecido los requerimientos sobre la seguridad operacional?			
¿Se ha especificado la fiabilidad del sistema/software, incluyendo las consecuencias en el caso de que falle, la información vital a proteger en caso de caída, la detección de los errores o el proceso de recuperación?			
¿Las compensaciones establecidas entre los atributos que compiten son aceptables, por ejemplo, entre robusteza y correctitud?			
¿Se han definido las interfaces internas, como por ejemplo el software o el hardware?			
¿Se han definido las interfaces externas, como por ejemplo usuarios o hardware?			

Criterio	Sí	No	NA
¿Se ha incluido la definición de *éxito*? ¿Se ha incluido la definición de *fracaso*?			
¿Cada requerimiento es relevante para el problema y su solución?			
2. No Ambiguo — Una Especificación de los Requerimientos es no ambigua si, y solo si, cada requerimiento especificado en ella posee exclusivamente una única interpretación.			
¿Los requerimientos se han especificado de forma suficientemente clara para que si se entregan a un grupo independiente para la implementación, dicho grupo sea capaz de entenderlos?			
¿Los requerimientos funcionales se encuentran separados de los no-funcionales?			
¿Los requerimientos están especificados de forma concisa, de modo que evitan la posibilidad de hacer múltiples interpretaciones de ellos?			
¿Todos los requerimientos evitan conflictos con otros requerimientos?			
3. Completitud — Una Especificación de los Requerimientos es completa si, y solo si, incluye los siguientes elementos: • Todos los requerimientos significativos, ya sea relacionados con la funcionalidad, con el rendimiento, las limitaciones de diseño, los atributos o las interfaces externas. • Las definiciones de las respuestas del sistema/software a todas las clases posibles de datos de entrada en todos los tipos posibles de situaciones. • Etiquetas descriptivas y referencias a todas las figuras, tablas y diagramas de la Especificación de los Requerimientos, así como la definición de todos los términos y unidades de medición.			
¿Se han especificado todas las interfaces de comunicación, incluyendo su aceptación de la negociación, su control de errores y los protocolos de comunicación?			
¿Se ha realizado el análisis para identificar los requerimientos que no se han tenido en cuenta?			
¿Se han especificado las áreas de incompletitud para cuando la información no esté disponible?			
¿Los requerimientos son completos, tales que si el producto satisface todos estos requerimientos, será aceptable?			
¿Es posible implementar todos y cada uno de los requerimientos?			
¿Se ha especificado la Mantenibilidad del sistema/software, incluyendo la habilidad de respuesta a los cambios en el entorno operativo, las interfaces, la precisión, el rendimiento, y otras capacidades adicionales predecibles?			
¿Se han especificado los requerimientos para la comunicación entre los componentes del sistema/software?			
¿Se ha definido la funcionalidad y el comportamiento global de todo el sistema/software?			
¿Se han establecido de forma explícita y sin ambigüedades las restricciones, suposiciones y dependencias apropiadas?			
¿Se ha especificado adecuadamente la infraestructura tecnológica para el sistema/software?			
¿Se han etiquetado de forma descriptiva todas las figuras, tablas y diagramas?			
¿Se han referenciado dentro del documento todas las figuras, tablas y diagramas?			
4. Consistencia — La consistencia se refiere a la consistencia interna. Si la Especificación de los Requerimientos no concuerda con el resto de documentación de la organización y del proyecto, significa que no es correcta.			
¿Se han especificado los requerimientos con un nivel de detalle consistente?			
¿Algunos de los requerimientos tienen que especificarse con mayor detalle?			

Criterio	Sí	No	NA
¿Algunos de los requerimientos deben ser especificados con menor detalle?			
¿Los requerimientos están en concordancia con el contenido del resto de documentación de la organización o del proyecto?			
5. Categorizado por importancia y/o estabilidad – Una Especificación de los Requerimientos se categoriza por importancia y/o estabilidad si cada requerimiento particular especificado en ella posee un identificador que establece su importancia o estabilidad. Ejemplos de rangos de categorización incluyen esencial, condicional u opcional. La estabilidad puede ser especificada en términos del número de cambios esperados para un requerimiento.			
a. ¿Los requerimientos poseen asociado un identificador para indicar la importancia o la estabilidad de un requerimiento en particular?			
b. ¿Existen conflictos en relación a la categorización de la importancia y/o estabilidad de los requerimientos?			
6. Verificable — Una Especificación de los Requerimientos es verificable si, y solo si, cada requerimiento especificado en ella es verificable. Un requerimiento es verificable si, y solo si, existe un proceso finito y rentable con el cual una persona o máquina puede comprobar que el sistema/software cumple con dicho requerimiento.			
¿El lenguaje y vocabulario con el que están escritos los requerimientos es entendible para los stakeholders? ¿Los stakeholders coinciden?			
¿Cada requerimiento puede ser probado? A partir de pruebas independientes, ¿puede ser posible determinar cuándo se satisface cada requerimiento?			
7. Modificable — Una Especificación de los Requerimientos es modificable si, y solo si, su estructura y estilo son tales que cualquier cambio en los requerimientos puede realizarse de forma fácil, completa y consistente, conservando la estructura y el estilo.			
¿Los requerimientos se identifican de forma única?			
¿Se han consolidado los requerimientos redundantes?			
¿Cada requerimiento se ha especificado de forma separada, evitando requerimientos compuestos?			
8. Trazable — Una Especificación de los Requerimientos es trazable si el origen de cada uno de sus requerimientos es claro y si facilita la referenciación de cada requerimiento en el desarrollo futuro o mejora la documentación.			
¿Se ha identificado cada requerimiento con el fin de facilitar su referenciación en el futuro desarrollo o en los esfuerzos de mejora?			
¿Cada requerimiento posee una referencia a los requerimientos previos del proyecto que están relacionados con él?			

Tarea XI.- Verificar el Documento

Se va a verificar que el documento ERS cumpla con las condiciones necesarias como no ambigüedad, fácil lectura para el desarrollador como para el usuario/cliente, etc.

Esta tarea será específicamente realizada por el equipo (desarrollador, analista, ingeniero en sistemas, cliente)

TECNICAS/HERRAMIENTAS	ENTRADAS	SALIDAS
Conocimiento del analista.	Documento ERS	Errores

Tarea XII Actualizar la versión del requerimiento y del documento.

En esta actividad se colocara la última versión del requerimiento, si este ha sido modificado por varias ocasiones, así como también del documento en general, esto es importante ya que permitirá conocer como se ha mejorado el requerimiento en conflicto de acuerdo a las necesidades del usuario.

Para esta tarea no se ha considerado ninguna herramienta ni técnica.

CAPITULO V

Caso de Estudio: Colegio Técnico Industrial Gualaceo

5.1 Descripción de la empresa

Misión

El colegio Técnico Industrial Gualaceo, fundamentara su accionar en los principios de unidad, equidad, cooperación y solidaridad, dando cumplimiento a cada uno de nuestros compromisos, para el logro de los objetivos del plan de transformación institucional de modo que contribuyamos al buen vivir

Visión

La comunidad Educativa del Colegio Técnico Industrial Gualaceo, en el lapso de seis años propiciara un ambiente de calidez, de armonía cordialidad y respeto; fortaleciendo los principios y valores para la consecución de una formación humana, científica y técnica, preparando bachilleres idóneos en el campo laboral

Objetivos Institucionales

- Formar al estudiante con mentalidad crítica y creadora capaz de que se constituya en un factor positivo en la sociedad.
- Fomentar el acervo cultural y deportivo.
- Presentar rendición de cuentas del desempeño de la Institución en la obtención de resultados durante el año escolar.
- Optimizar la prestación de servicios en el campo educativo y productivo.
- Monitorear y evaluar periódicamente los alcances de los programas operativos que conforman el plan de transformación institucional

Descripción de la problemática actual de la institución

El colegio Técnico Industrial Gualaceo cuenta con un sistema informático para el registro de notas y para la recepción de matrículas de los diferentes estudiantes del establecimiento. El sistema actual es una aplicación creada en visual Basic 5, con una base de datos en Microsoft Access 1997.

Este sistema permite el registro de notas de los alumnos de los diferentes cursos. El sistema solamente visualiza las notas del alumno y el estado en el que estos están al final del periodo lectivo (Aprobado, Reprobado, Supletorio.)

Los docentes únicamente pueden registrar las notas de los alumnos en los laboratorios de computación de la institución; y los alumnos no tienen acceso a esta información, solamente mediante los reportes que realizan al final del periodo los docentes.

Para la creación de nuevos años lectivos y materias, el profesor de computación, el cual da mantenimiento al sistema y a los equipos copia una tabla de la base de datos y cambia el nombre de la materia, del periodo lectivo.

5.2 Fase I: Elicitación de Requerimientos

Tarea I. Estudio Inicial.

Esta encuesta tiene el objetivo de capturar información referente al registro de notas y matriculas en la institución.

Para la realización de esta fase se realizó la siguiente entrevista al Rector del Colegio Técnico Industrial Gualaceo:

1. ¿Tiene planteados la misión, visión y objetivos de la institución?

   Sí.

2. ¿La organización cuenta con algún tipo de sistema informático?

   Por el momento si, la institución cuenta con un sistema de registro de notas y de matrículas obsoleto.

3. ¿Si tuviera la oportunidad de crear un sistema o mejorar el sistema existente lo haría? ¿Cuánto recurso usted estaría dispuesto a aportar?

Se está buscando mejorar el sistema existente o a su vez implementar uno nuevo de acuerdo a las nuevas especificaciones del Ministerio de Educación para el registro de notas. En cuanto a los recursos al ser una institución pública, estos dependen si el proyecto es aprobado y de los recursos que el Ministerio de Educación asigne al establecimiento.

4. ¿Cómo se registran las notas de los alumnos?

Mediante el sistema con el que cuenta la institución se registran las notas de cada estudiante en las diferentes materias, al estar este sistema obsoleto se registran las notas sobre un total de 20, pero según las nuevas disposiciones del Ministerio las notas se registraran sobre diez la misma que se divide en dos el ochenta por ciento son de actividades como deberes, lecciones, pruebas, etc. y el veinte por ciento el examen. Estas notas se deben registrar cada seis semanas al terminar los bloques programáticos. El promedio del quimestre es el ochenta por ciento del promedio de los tres bloques programáticos correspondientes al quimestre más el veinte por ciento del examen lo que daría una nota sobre diez. Para que un estudiante pierda el año debe tener una nota menor a cinco y haber perdido el examen remedial y luego el examen de gracia y para que se quede en supletorios la nota debe ser mayor o igual a cinco y menor que siete si no pasa este examen de supletorio que es sobre 10 el estudiante pasa a dar el examen remedial, algo muy importante es que el estudiante que no pase el examen remedial y tenga opción a rendir el examen de gracia es que tiene que haberse quedado en una sola materia, caso contrario automáticamente pierde el año.

5. ¿Cuántos docentes trabajan en la institución?

Actualmente trabajan 35 profesores en la institución.

6. ¿Cuántos estudiantes están registrados en la institución?

Existen 700 estudiantes registrados en la institución.

7. ¿Quiénes intervienen en el proceso de registro de notas y matriculas?

Bueno, intervienen los profesores de las diferentes asignaturas los cuales ingresan al sistema las notas obtenidas por los estudiantes, la secretaria quien verifica que las notas estén registradas y genera los reportes que se entregaran a los padres de familia. En caso de rectificación interviene el rector quien autoriza la rectificación de notas, el docente y el encargado del laboratorio de cómputo quien habilita la modificación de las notas a los profesores.

En una matrícula interviene la secretaria que ingresa al sistema para realizar una matrícula de un estudiante ya sea este antiguo o nuevo.

8. ¿Qué objetivo persigue el desarrollo del sistema?

Realizar el registro de notas de cada alumno y de cada materia según los nuevos parámetros establecidos por el ministerio de educación, obtener los reportes de cada parcial, de los dos quimestres, el reporte final de las notas y el estado académico del estudiante.

Realizar el registro de matrícula a un estudiante, generar reportes del comprobante de matrícula del estudiante.

9. ¿En qué tipo de tareas de la organización debe ayudar el sistema?

Primero en mantener un registro de notas rápido y sin inconvenientes tanto para los docentes como para la secretaria y los alumnos. También ayudara a tener un rápido conocimiento las calificaciones de los estudiantes permitiendo saber el estado del

estudiante en el periodo lectivo actual. Ayudará agilizar el proceso de matrícula de un estudiante.

10. ¿Qué beneficios espera obtener?

Mejorar el funcionamiento de la institución al registrar las notas de manera más eficiente, agilizando la entrega de calificaciones a los padres de familia y permitiendo llevar registros estadísticos de la evolución de los estudiantes individual y colectivamente.

Permitir un rápido proceso de Matricula evitando la pérdida de tiempo de los responsables de cada estudiante, presentar información sobre la cantidad de matrículas realizadas en el periodo lectivo.

11. ¿Qué tareas debe realizar el nuevo sistema?

El registro de notas: debe obtener el listado de los alumnos en los diferentes cursos y paralelos, las materias que se imparten en cada curso y que profesor la dicta y registrar cuatro notas, obtener el promedio de los parciales en el quimestre, Obtener el estado del estudiante, generar los certificados de las notas.

Matrícula del estudiante: el sistema permitirá matricular a estudiantes nuevos y estudiantes que pasen a un nivel superior. Para alumnos nuevos: el sistema permitirá ingresar la información correspondiente del estudiante. Para alumnos antiguos el sistema debe verificar que el estudiante haya pasado al siguiente nivel de educación.

12. ¿Quiénes son las personas que usaran el nuevo sistema?

Este sistema será usado básicamente por los docentes de la institución, que son los que realizan el registro de notas de los estudiantes, por la secretaria quien realizara las matriculas, generará los reportes de calificaciones de cada estudiante para ser

entregados posteriormente a sus representantes y por los estudiantes que podrán realizar las consultas de sus calificaciones.

13. ¿Qué terceras personas se verán afectadas con la implementación del sistema?

Ninguna persona se verá afectada.

14. ¿El sistema que se desarrollara debe interactuar con otros sistemas preexistentes o que existan en el futuro?

No, se pretende cambiar de sistema ya que el que se usa es un sistema obsoleto que no cumple con las necesidades de la institución. Pero en un futuro podría interactuar con otros módulos que fueran implementados en la institución.

15. ¿Existe alguna restricción para el desarrollo de la aplicación?

La restricción principal es la asignación de recursos por parte del Ministerio.

16. ¿Conoce los planes de la dirección para desarrollar un sistema que permita registrar las notas del estudiante?

No.

17. ¿Cómo le ayudaría el nuevo sistema en sus tareas?

Permitirá agilizar los procesos de matrícula y calificaciones de los estudiantes.

Permitirá además efectuar un seguimiento completo y detallado al proceso de matrícula y registro de notas mediante el análisis de los informes que proveerá. Permitirá generar reportes sobre las calificaciones de los estudiantes, las materias y los cursos asignados en el año lectivo.

A partir de esta entrevista se obtuvo los siguientes participantes para el proyecto:

Nombres:	Carlos Daniel
Apellidos:	Borja Buestan
Dirección:	Bullcay
Teléfono:	0984718801
E-mail	danny_guala@hotmail.com
Instrucción	Superior
Rol:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Analizarlos para su posterior implementación.

Nombres:	Valeria Alexandra
Apellidos:	Cuji Torres
Dirección:	Benigno Vásquez y Cuenca
Teléfono:	3050876
E-mail	valeria_alex06@hotmail.com
Instrucción:	Superior
Tipo:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Mauricio
Apellidos:	Ortiz
Dirección:	Cuenca
Teléfono:	-
E-mail	mortizo@ups.edu.ec
Instrucción:	Superior
Rol:	Analista/Desarrollador
Cargo:	Supervisor del proyecto
Responsabilidades:	Realizar las revisiones del proyecto.

Nombres:	Marcelo
Apellidos:	Cando
Dirección:	Dávila Chica
Teléfono:	-
E-mail	-
Instrucción:	Superior
Tipo:	Cliente
Cargo:	Rector
Responsabilidades:	Dar información acerca de las necesidades de la empresa para la implementación del proyecto.

Nombres:	José Luis
Apellidos:	Rodas
Dirección:	Bullcay
Teléfono:	-
E-mail	-
Instrucción:	Superior
Tipo:	Cliente
Cargo:	Técnico Informático
Responsabilidades:	Dar información acerca de las necesidades de la empresa para la implementación del proyecto.

Nombres:	Carmen
Apellidos:	Brito
Dirección:	Jaime Roldos
Teléfono:	2143229
E-mail	-
Instrucción:	Superior
Tipo:	Usuario
Cargo:	Secretaria
Responsabilidades:	Dar información acerca de las necesidades de la empresa para la implementación del proyecto.

Características de los Usuarios

Tipo de usuario	Docente.
Formación	Universitario, con título de tercer nivel.
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá listar los alumnos de los distintos años de básica, gestionar notas, faltas y generar reporte de notas

Tipo de usuario	Secretaria.
Formación	Universitario, con título de tercer nivel.
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá listar los alumnos de los distintos años de básica, gestionar notas, faltas y generar reporte de notas, gestionar, materias, estudiantes, docentes, matriculas, generar reportes varios.

Tipo de usuario	Estudiante
Formación	Estudiante
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá visualizar la información de la institución así como las calificaciones registradas en las materias que este cursando y que haya cursado.

Tipo de usuario	Administrador
Formación	Universitario, con título de tercer nivel. Conocimientos altos en informática
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario se encargará de la gestión de la base de datos del sistema. Es decir, efectuará el alta y baja de los usuarios, asignaturas, año de básica, etc. Así como las modificaciones. En general tiene acceso a todo el sistema y podrá realizar todo tipo de procesos

Tarea II. Objetivos del Sistema

Se obtuvieron varias ideas para el proyecto las mismas son:

- Desarrollar un sistema de información automatizado para el control de matrícula y calificaciones de los estudiantes del Colegio Técnico Industrial Gualaceo.
- Facilitar del proceso de matriculación y registro de notas
- Identificar los procedimientos de control de matrícula y calificaciones para generar reportes que agilicen los procesos para la toma de decisiones.
- Permitir que el acceso al sistema sea seguro para evitar posible violación a la información de la institución.
- Tener una base de datos completa y actualizada de los alumnos, profesores, materias, notas de las materias.
- Obtener reportes de la información almacenada respecto a los objetos involucrados en este sistema (Docentes, Estudiantes, Materias, etc.)
- Gestionar información de Estudiantes (Insertar, Actualizar, Listar)
- Gestionar información de Docentes (Insertar, Actualizar, Listar).
- Gestionar de información Matricula (Insertar, Actualizar, Listar, Eliminar).

- Gestionar Notas de cada Estudiante (Insertar, Actualizar, Listar).
- Gestionar horarios de clase de profesores (por día, por clase y hora).

OBJETIVOS DEL SISTEMA

OBJ-1. Crear un sistema de información automatizado para el control de matrícula y calificaciones de los estudiantes del Colegio Técnico Industrial Gualaceo.

OBJ-1	**Crear sistema para automatizar el registro de notas y matriculas**
Versión	1.0
Autores	Daniel Borja, Valeria Cuji
Fuentes	Rector del colegio Arq. Marcelo Cando
Descripción	El sistema permitirá agilizar los procesos de registro de notas y realización de matrículas del estudiante.
Importancia	Alta
Urgencia	Indispensable
Estado	En construcción
Estabilidad	5

OBJ-2. Almacenar información del proceso de matrícula y registro de notas tal como (Estudiantes, Profesores, Materias, Notas, Responsable del estudiante)

OBJ-2	**Almacenar información concerniente al registro de notas y matriculas**
Versión	1.0
Autores	Valeria Cuji
Fuentes	Rector del colegio Arq. Marcelo Cando,
Descripción	El sistema almacenar información referente al registro de notas y realización de matrículas del estudiante.
Importancia	Alta
Urgencia	Indispensable
Estado	En construcción
Estabilidad	5

OBJ-3. Contar con un sistema que permita administrar información almacenada (Insertar, Modificar, Listar, Eliminar)

OBJ-3	**Gestionar la información almacenada**
Versión	1.0
Autores	Valeria Cuji, Daniel Borja
Fuentes	Rector del colegio Arq. Marcelo Cando,
Descripción	El sistema permitirá al usuario insertar , modificar, listar, eliminar información
Importancia	Alta
Urgencia	Indispensable
Estado	
Estabilidad	5

OBJ-4. Crear control de acceso al sistema

OBJ-4	**Controlar el acceso al sistema**
Versión	1.0
Autores	Valeria Cuji
Fuentes	José Luis Rodas
Descripción	El sistema pedirá Usuario y contraseña para ingresar a la página que muestra la información para el registro de notas y matrículas.
Importancia	Alta
Urgencia	Indispensable
Estado	En construcción
Estabilidad	5

Tarea III y Tarea IV.- identificar Requerimientos del Sistema y Priorizar Requerimientos

ID	REQUERIMIENTO	OBJ. ASOCIADO	FUENTE	AUTOR	PRIORIDAD
RU-1	El sistema gestionara (insertar, modificar, listar) la información del estudiante	OBJ 2 OBJ 3	**Secretaria Rector, Técnico Informático**	Valeria Cuji	ALTA
RU-2	El sistema gestionara (insertar, modificar, eliminar, visualizar) información de las diferentes materias.	OBJ 2 OBJ 3	**Secretaria Rector.**	Daniel Borja	ALTA
RU-3	El sistema gestionara (insertar, modificar) información de la matrícula.	OBJ 2 OBJ 3	**Secretaria Rector,**	Daniel Borja	ALTA
RU-4	El sistema gestionara (insertar, modificar, eliminar, listar) los de los Docentes de la institución.	OBJ 2 OBJ 3	**Secretaria**	Valeria Cuji	ALTA
RU-5	El sistema deberá generar reportes.	OBJ 1	**Secretaria, Rector, Docentes.**	Valeria Cuji	ALTA
RU-6	El sistema contara con claves de acceso	OBJ 4	**Técnico Informático**	Daniel Borja	ALTA
RD-1	El sistema deberá poder ser ampliado en un futuro con módulos necesarios en el ámbito de la educación primaria y secundaria.		**Técnico Informático, Rector**	Daniel Borja	MEDIA
RD-2	El sistema será creado en lenguaje *Java Enterprice Edition*		**Técnico Informático**	Ing. Mauricio Ortiz	MEDIA
RD-3	El sistema se diseñara como una interfaz web		**Técnico Informático, Rector**	Ing. Mauricio Ortiz	ALTA
RD-4	Se usara una base de datos relacional.		**Técnico Informático**	Ing. Mauricio Ortiz	MEDIA
RD-5	Se debe evitar el ingreso de información fallida por medio de mensajes que genere el sistema.	OBJ 4	**Técnico Informático**	Daniel Borja	ALTA
RD-6	La información que se muestre en el reporte deberá ser clara y precisa.	OBJ 1	**Técnico Informático**	Valeria Cuji	ALTA
RI-1	Interfaz amigable y fácil de comprender	OBJ 1	**Técnico Informático**	Valeria Cuji	ALTA
RI-2	Se usaran colores estándares de Windows		**Técnico Informático**	Daniel Borja	ALTA
RI-3	En la pantalla principal del sistema habrá una imagen de bienvenida y un botón que lleve a la página de inicio de sesión	OBJ 4	**Técnico Informático, Rector**	Daniel Borja	MEDIA

5.3 Fase II: Análisis de Requerimientos

Tarea V Clasificación de requerimientos

Requerimientos comunes de los Interfaces			
Usuario(RIU)	**Hardware**(RIHw)	**Software**(RISw)	**Comunicación**(RIC)
1. La interfaz gráfica se creara de una manera de fácil comprensión para el usuario de manera que este no requiera mayor esfuerzo para utilizar el sistema. **2.** El sistema contara con manual de usuario para ir guiando en el manejo del sistema. **3.** Los usuarios ya deberían conocer, por experiencia en otros sistema como email y el navegador de internet las funcionalidades generales de un sistema web	CPU con las siguientes características: Memoria mínima: 1 GB Memoria recomendada: 2 GB, Procesador mínimo: DualCore 1.6 GHz o similar, Procesador recomendado: Core i3 2.1 GHz o similar. Espacio en disco mínimo: 500 MB de espacio libre, Espacio en disco recomendado: 1 GB de espacio libre.	1. La base de datos que se utilizara es MySql versión 5.0 2. Se requiere que el Equipo cuente con sistema operativo WINDOWS 7 o superior 3. El sistema será compatible con el explorador Web MozilaFirefox. **4.** Lenguaje de programación para el sistema va a ser JAVA 5. Como servidor web se utilizara Apache 2.0.	1. Para la interacción con la base de datos se utilizara JDBC

Requerimientos Funcionales	
RF1	El sistema permitirá ingresar al sistema mediante usuario y contraseña
RF2	El sistema debe permitir gestionar periodos lectivos.
RF3	El sistema debe permitir registrar a los docentes(Nombres, Apellido, Domicilio, Teléfono)
RF4	El sistema permitirá ingresar estudiantes(Nombres, Apellido, Domicilio, Teléfono, Nombre de representante, Fecha de nacimiento)
RF5	El sistema permitirá modificar estudiantes((Nombres, Apellido, Domicilio, Teléfono, Nombre de representante, Fecha de nacimiento))
RF6	El sistema permitirá listar estudiantes(Nombres, Apellido, Domicilio, Teléfono, Nombre de representante, Fecha de nacimiento)
RF7	El sistema permitirá ingresar materias(Nombre, Descripción)
RF8	El sistema permitirá modificar materias(Nombre, Descripción)
RF9	El sistema permitirá listas las materias(Nombre, Descripción)
RF10	El sistema debe permitir crear una matrícula (Información del estudiante, Fecha de matrícula, Grado en que se matricula, especialidad)
RF11	El sistema debe permitir modificar una matrícula(Información del estudiante, Fecha de matrícula, Grado en que se matricula, especialidad)
RF12	El sistema debe permitir listar una matrícula(Información del estudiante, Fecha de matrícula, Grado en que se matricula, especialidad)
RF13	El sistema debe permitir eliminar una matricula
RF14	El sistema permitirá generar reportes del total de estudiantes matriculados en un periodo lectivo.
RF15	El sistema permitirá generar reportes de las notas de los estudiantes por curso.
RF16	El sistema permitirá ingresar calificaciones
RF17	El sistema permitirá modificar calificaciones
RF18	El sistema permitirá listar calificaciones
RF19	El sistema permitirá generar certificados de inscripción de matricula
RF20	El sistema debe mostrar a cada docente a que curso tiene que impartir sus clases.
RF21	El sistema debe mostrar a cada docente la cantidad de alumnos por aulas

Requerimientos No Funcionales					
Rendimiento	**Seguridad**	**Disponibilidad**	**Comunicación**	**Mantenibilidad**	**Portabilidad**
1. El servidor de base de datos, deberá tener un respaldo apropiado, así como personal técnico listo para cualquier eventualidad 2. Razonablemente el sistema debería tardar 75 milisegundos en cada transacción, lo que se traduce en 5 transacciones cada 16.6 segundos. Soportando a 25 usuarios concurrentemente.	1. Uso de contraseñas para cada usuario (administrador, Docente, Secretaria). Esto permitirá que tengan acceso al sistema solo las personas que tienen autorización.	El sistema se debe encontrar disponible en todo momento.	1. El sistema manejara mensaje de errores y confirmaciones. 2. Se requiere que la aplicación soporte la arquitectura cliente-servidor. Se tendrá un solo servidor ubicado en la dirección al que accederán los diferentes terminales.	El sistema incluirá documentación de ayuda, incluyendo manual de usuario.	1. Utilizar el lenguaje y plataforma JAVA. 2. Utilizar como servidor de Base de datos MySQL

Modelo de casos de uso a partir de los requerimientos funcionales.

Identificar Casos de uso y actores

Casos de uso del sistema.

1. Ingresar al sistema
2. Gestionar Periodos Lectivos.
3. Gestionar Cursos.
4. Gestionar Estudiantes.
5. Gestionar Materias.
6. Gestionar Docentes.
7. Gestionar Notas.
8. Gestionar Matriculas
9. Generar Reporte

Tabla 20 Actores del Sistema

Actor		Descripción
Usuario del Sistema	**Profesores/Docentes**	Deben identificarse en el sistema para poder tener acceso al mismo y poder luego, observar información relacionadas con las notas , reportes de horarios, reportes de estudiantes, reportes de materias, pudiendo también realizar la impresión de estos reportes
	Alumnos/Estudiantes	Deben identificarse en el sistema para poder visualizar: horario de clases y las notas de cada materia cursada.
	Secretaria	Deben identificarse en el sistema para poder luego tener privilegios de gestión de matrículas, gestión de profesores, gestión de Alumnos, gestión de notas/calificaciones, gestión de periodos lectivos, gestión de cursos.
	Administrador	Debe identificarse para luego poder dar y quitar privilegios a (Profesores, Estudiante y Secretaria) tiene además de estos permisos todos los privilegios de los actores mencionados anteriormente

CASOS DE USO:

Tomaremos en cuenta los casos de uso del sistema de cada uno de estos se desglosara casos de uso más detallados.

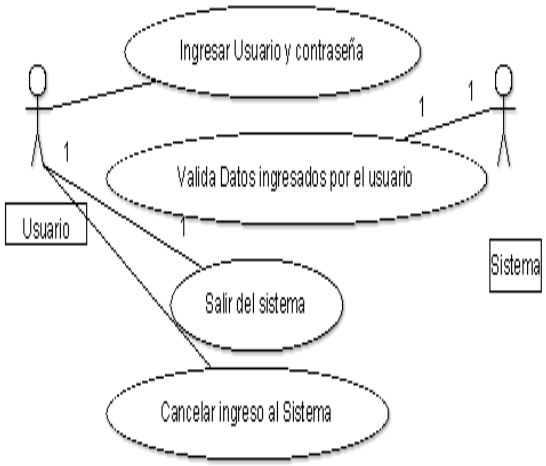


Ilustración 19 Caso de uso Ingreso al sistema

RF-1	Ingresar al sistema		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Valeria Cuji		
Fuentes	Administrador Informático de la Institución		
Descripción	Permite ingresar al sistema		
Actor	Usuario del sistema		
Precondición	Ninguna		
Secuencia Normal	Paso: Acción 1: El usuario ingresa nombre y contraseña 2: El sistema verifica los datos y dependiendo del usuario y clave le proporciona permisos de acceso 3: El usuario termina la sesión		
Post Condición	Periodo electivo guardado en la Base de Datos		
Excepciones	Paso: Acción 1: El usuario puede salir del sistema o cancelar el ingreso al sistema		
Prioridad	Alta		
Estado	En Análisis		
Estabilidad	Alta		
Comentarios	No		

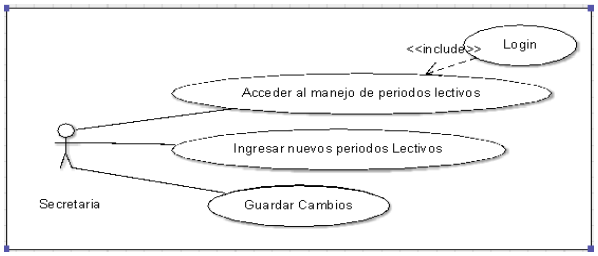


Ilustración 20: Caso de uso Ingresar Periodo Lectivo.

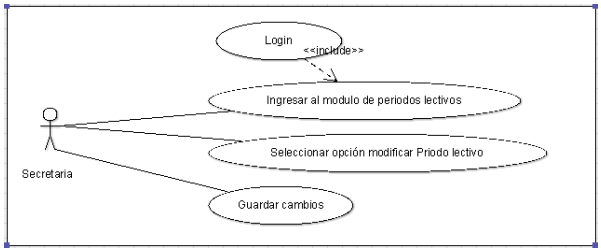


Ilustración 21: Caso de uso Modificar Periodo Lectivo

Descripción de caso de uso Gestión Año lectivo

RF-2	Crear Periodos Lectivos		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Valeria Cuji		
Fuentes	Secretaria		
Descripción	Permite ingresar y crear en la base de datos periodos lectivos		
Actor	Usuario del sistema		
Precondición	El usuario debe haber ingresado al sistema con los respectivos privilegios para gestionar periodos lectivos		
Secuencia Normal	Paso	Acción	
	1	El usuario accede al módulo de periodos lectivos y seleccionar la opción crear nuevo periodo lectivo.	
	2	El usuario ingresa la fecha de inicio y la fecha final del periodo lectivo	
	3	El sistema comprueba que se hayan ingresado correctamente los datos y los almacena	
Post Condición	Periodo electivo guardado en la Base de Datos		
Excepciones	Paso	Acción	
	1	Si existen errores el sistema informa al usuario sobre el error y le perm modificaciones	
Prioridad	Alta		
Estado	En construcción		
Estabilidad	Alta		
Comentarios	No		

<table>
<tr><td>RF-3</td><td colspan="3">Modificar Periodos Lectivos</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Valeria Cuji</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar en la base de datos periodos lectivos</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema con los respectivos privilegios para
gestionar periodos lectivos</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de periodos lectivos y seleccionar la opción modificar nuevo periodo lectivo.</td></tr>
<tr><td>2</td><td>Es el sistema permite buscar el periodo lectivo que desee modificar</td></tr>
<tr><td>3</td><td>El usuario modifica la fecha de inicio o la fecha final del periodo lectivo.</td></tr>
<tr><td>4</td><td>Usuario guarda información modificada</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Periodo electivo modificado y guardado en la Base de Datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

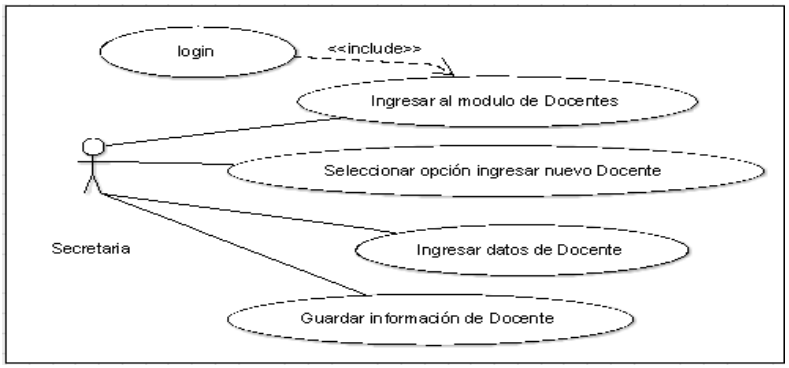


Ilustración 22: Ingresar Docentes

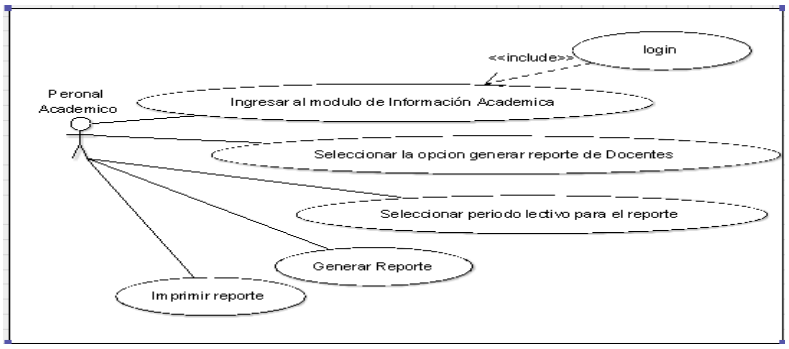


Ilustración 23: Caso de Uso Modificar Docente

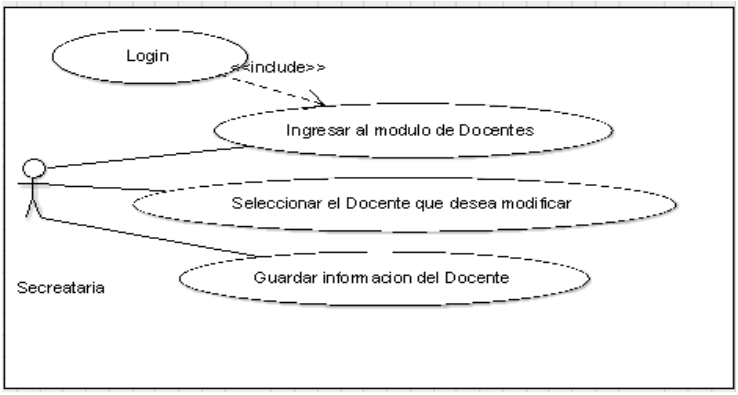


Ilustración 24: Caso de uso Generar reporte de Docentes

<table>
<tr><td>RF-4</td><td colspan="3">Ingresar Docente</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear Docentes</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber Ingresado al sistema. Con privilegios para gestionar docentes</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de Docentes y selecciona la opción ingresar nueva Docente.</td></tr>
<tr><td>2</td><td>El usuario ingresa datos de Docente.</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4</td><td>El usuario accede al módulo de Docentes y selecciona la opción ingresar nueva Docente.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Datos de Docente almacenado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-5</td><td colspan="3">Modificar Docente</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar Docentes</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber Ingresado al sistema. Con privilegios para gestionar docentes</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>El usuario accede al módulo de Docentes y selecciona la opción modificar Docente.</td></tr>
<tr><td>1.</td><td>El sistema presenta los docentes.</td></tr>
<tr><td>2.</td><td>El usuario realiza la búsqueda del docente por apellido</td></tr>
<tr><td>3.</td><td>El usuario selecciona el docente que desea modificar</td></tr>
<tr><td>4.</td><td>El sistema presenta al usuario la información del docente</td></tr>
<tr><td>5.</td><td>El usuario modifica la información del Docente</td></tr>
<tr><td>6.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>7.</td><td>El usuario modifica la información del Docente</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Datos de Docente modificado y almacenado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-6</td><td colspan="3">Generar reporte de docentes</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Valeria Cuji</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite generar un reporte o listado de los docentes que son profesores de un curso o que poseen más de un grado asignado</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema y accedido al módulo Información académica</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Docentes y selecciona la opción modificar Docente.</td></tr>
<tr><td>2.</td><td>El sistema presenta los docentes.</td></tr>
<tr><td>3.</td><td>El usuario realiza la búsqueda del docente por apellido</td></tr>
<tr><td>4.</td><td>El usuario selecciona el docente que desea modificar</td></tr>
<tr><td>5.</td><td>El sistema presenta al usuario la información del docente</td></tr>
<tr><td>6.</td><td>El usuario modifica la información del Docente</td></tr>
<tr><td>7.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>8.</td><td>El usuario modifica la información del Docente</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos consultados no se alteran en la base de datos y el reporte debe poder imprimirse.</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El sistema genera el reporte pero no devuelve ningún resultado debido a que los registros que existen no cumplen con los parámetros especificados, por lo que se notifica al usuario y se regresa al formulario de ingreso de parámetros para generar otro reporte</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Baja</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

GESTIÓN DE ESTUDIANTES:

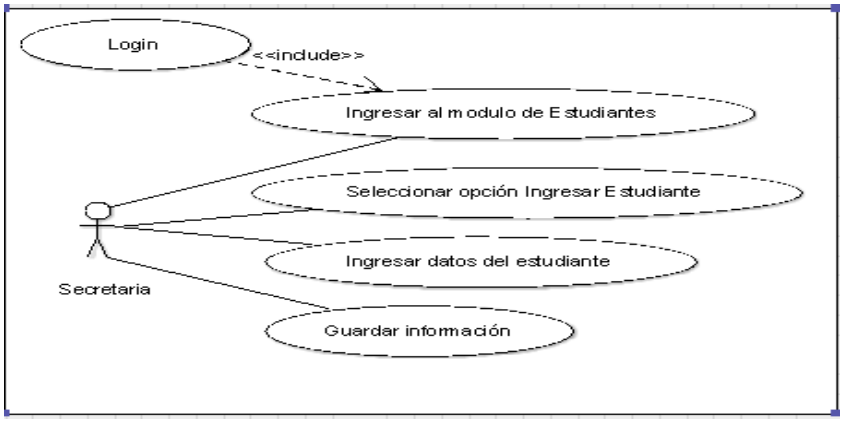


Ilustración 25: Caso de uso Ingresar Estudiante

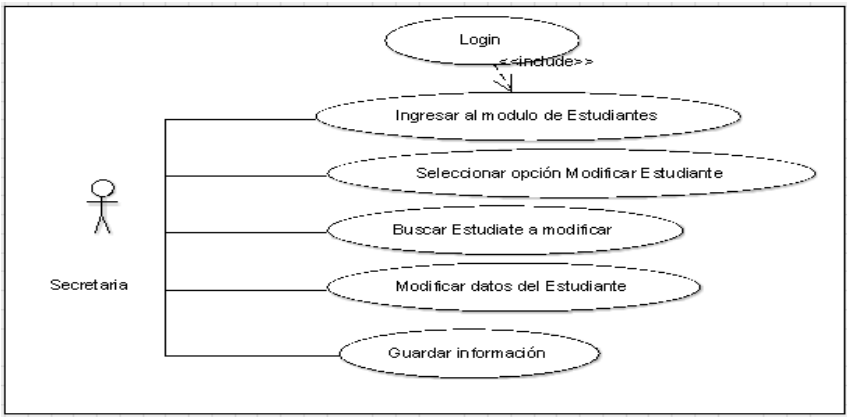


Ilustración 26: Caso de uso Modificar Estudiante

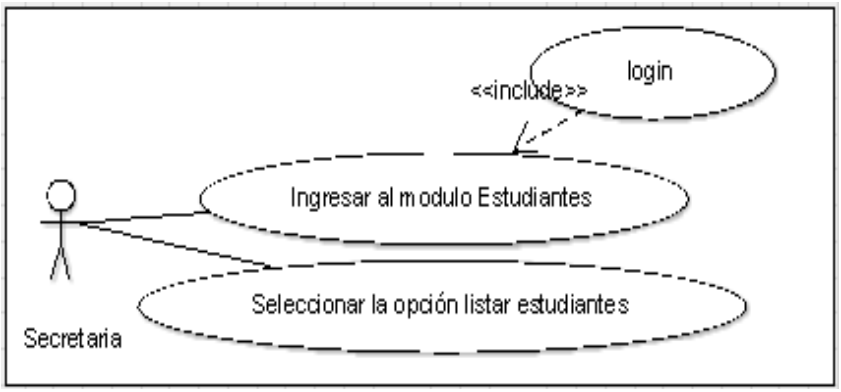


Ilustración 27: Caso de Uso Listar Estudiante

<table>
<tr><td>RF-7</td><td colspan="3">Ingresar Estudiante</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear estudiantes</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de estudiantes y selecciona la opción ingresar nuevo estudiante.</td></tr>
<tr><td>2</td><td>El usuario ingresa datos de estudiante.</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4</td><td>El usuario accede al módulo de estudiantes y selecciona la opción ingresar nuevo estudiante.</td></tr>
<tr><td>5</td><td>El usuario ingresa datos de estudiante.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Estudiante ingresado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
</table>

RF-8	Modificar Estudiante		
Versión	*1.0*	Fecha	07-08-2013
Autores	Valeria Cuji		
Fuentes	Secretaria		
Descripción	Permite ingresar y modificar estudiantes		
Actor	Usuario del sistema		
Precondición	El usuario debe haber ingresado al sistema.		
Secuencia Normal	Paso	Secuencia	
		El usuario accede al módulo de estudiantes y selecciona la opción modificar nuevo estudiante.	
		El usuario modifica datos de estudiante.	
		El sistema comprueba que se hayan ingresado correctamente los datos y los almacena	
		El usuario accede al módulo de estudiantes y selecciona la opción modificar nuevo estudiante.	
		El usuario modifica datos de estudiante.	
Post Condición	Datos de Estudiante modificados		
Excepciones	Paso	Acción	
	1	Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones	
Prioridad	Media		
Estado	En Construcción		
Estabilidad	Media		

RF-9	Listar Estudiantes		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Valeria Cuji		
Fuentes	Secretaria, Docentes		
Requerimientos asociados	Ninguno		
Descripción	Permite listar información del estudiante		
Actor	Usuario del sistema		
Precondición	El usuario debe haber ingresado al sistema		
Secuencia Normal	**Paso**	**Secuencia**	
	1.	El usuario accede al módulo de estudiantes y selecciona la opción mostrar estudiantes.	
	2.	El usuario lista datos de estudiante.	
Post Condición	Sistema presenta datos del estudiante		
Excepciones	Paso	Acción	
	1	Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones	
Prioridad	Baja		
Estado	En Construcción		
Estabilidad	Baja		
Comentarios	No		

GESTIÓN CURSOS/GRADOS

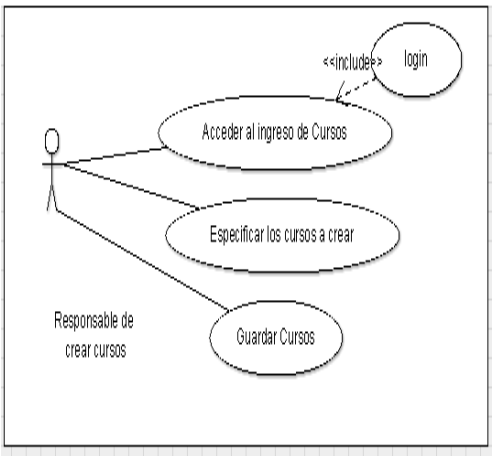


Ilustración 28 Caso de uso Ingresar de Cursos

RF-10	Crear cursos/grados		
Versión	1.0	Fecha	07-08-2013
Autores	Daniel Borja		
Fuentes	Secretaria		
Descripción	Permite ingresar y crear en la base de datos Cursos		
Actor	Usuario del sistema		
Precondición	El usuario debe haber : Ingresado al sistema. Registrado docentes. Registrado Materias. Registrado periodo lectivo.		
Secuencia Normal	Paso: — Secuencia: El usuario accede al módulo de Grados y selecciona la opción crear nuevo grado. — El usuario ingresa el periodo lectivo, materia, el docente, para el curso — El sistema comprueba que se hayan ingresado correctamente los datos y los almacena — El usuario accede al módulo de Grados y selecciona la opción crear nuevo grado.		
Post Condición	Los grados o curso se asignan		
Excepciones	Paso: 1 — Acción: Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

GESTIONAR MATERIAS.

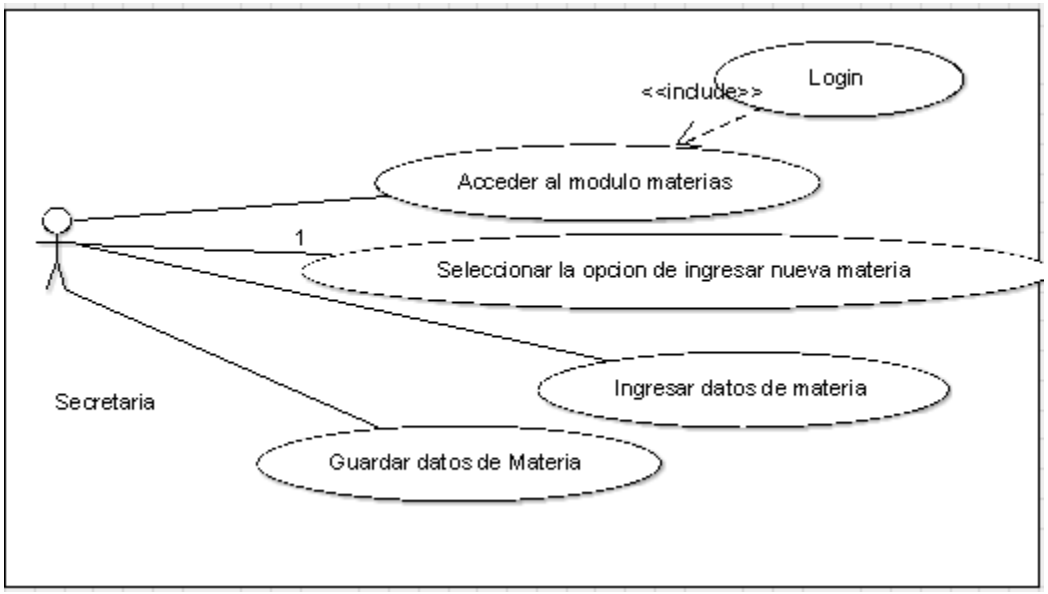


Ilustración 29: Caso de uso Ingresar Materias

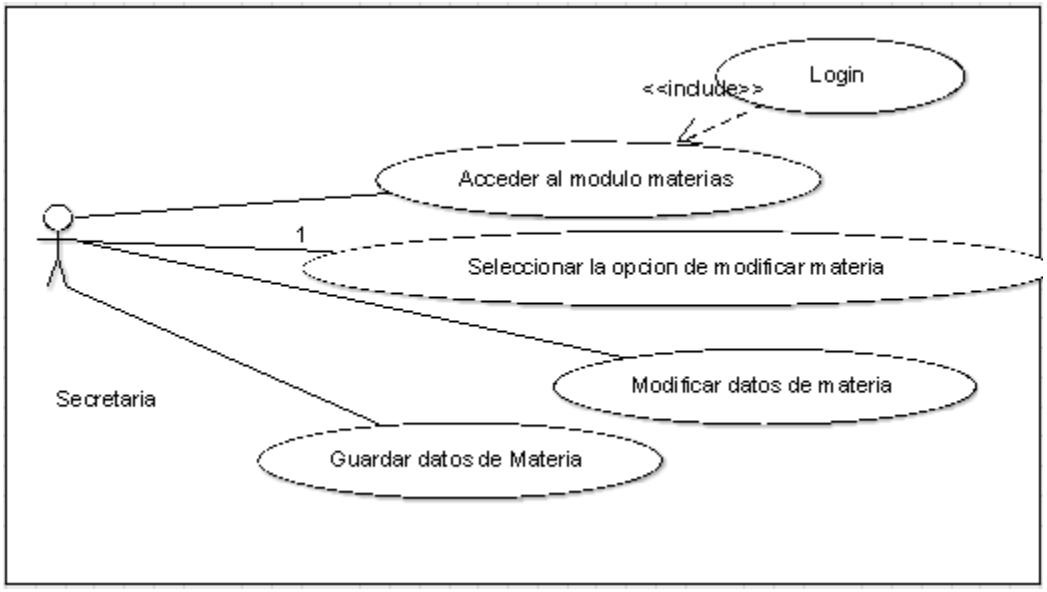


Ilustración 30: Caso de uso Modificar Materias

<table>
<tr><td>RF-11</td><td colspan="3">Ingresar Materias</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear materias</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber :
1. ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Materia y selecciona la opción ingr</td></tr>
<tr><td>2.</td><td>El usuario ingresa datos de Materia.</td></tr>
<tr><td>3.</td><td>El sistema comprueba que se hayan ingresado correctamente los da</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Materia ingresada en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le perm
modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-12</td><td colspan="3">Modificar Materia</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Valeria Cuji</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar materias</td></tr>
<tr><td>Actor</td><td colspan="3">Usuario del sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber :
1. ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Materia y selecciona la mater</td></tr>
<tr><td>2.</td><td>El usuario modifica la Materia.</td></tr>
<tr><td>3.</td><td>El sistema comprueba que se hayan ingresado correctamente</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Materia modificada y almacenada en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

Gestionar Notas/Calificaciones

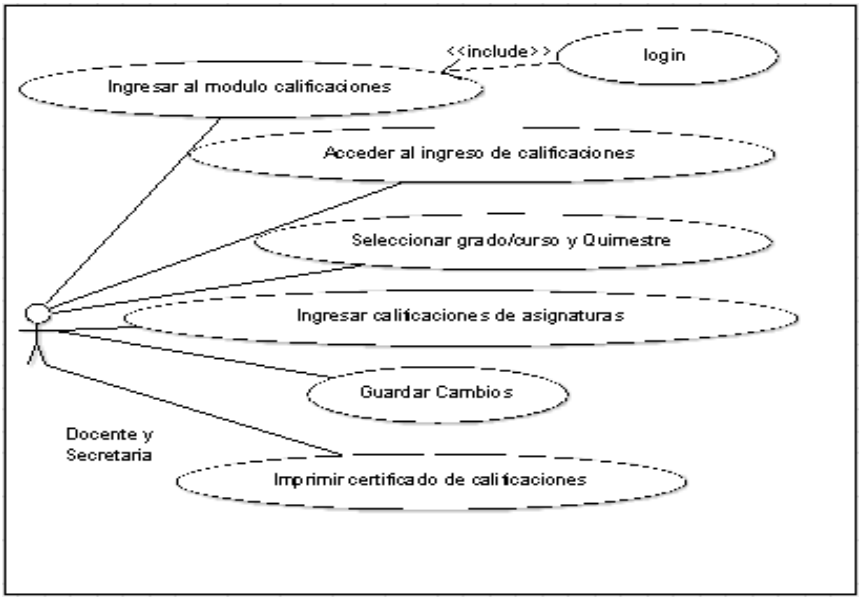


Ilustración 31: Caso de uso Ingresar Calificaciones.

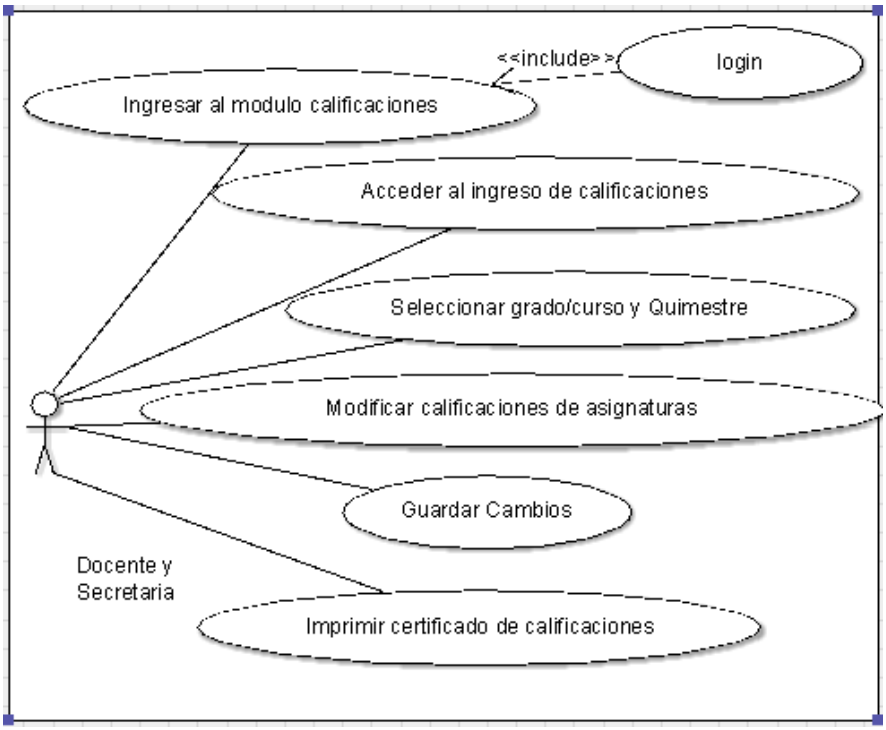


Ilustración 32: Caso de uso modificar Calificaciones

RF-13	Ingresar Notas/Calificaciones		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Daniel Borja		
Fuentes	Docente		
Descripción	Permite ingresar las calificaciones de los estudiantes de cada grado o curso		
Actor	Usuario del sistema		
Precondición	Ingresado al sistema. Acceder al módulo de calificaciones		
Secuencia Normal	**Paso**	**Secuencia**	
	1.	El usuario accede al menú de calificaciones y selecciona la opción de calificaciones.	
	2.	Es usuario selecciona el año lectivo, grado o curso y quimestre del cu las calificaciones.	
	3.	El usuario ingresa las calificaciones de cada estudiante dentro de cad presiona el botón guardar.	
	4.	El sistema comprueba la validez de los datos, realiza cálculos para ob almacena los datos.	
	5.	El sistema presenta el certificado o libreta de cada uno de los estudia calificaciones del quimestre del grado o curso seleccionado	
Post Condición	Los datos ingresados por el usuario se almacena en la base de datos		
Excepciones	Paso	Acción	
	1	Si existe algún problema con los datos ingresados, el sistema informa a permite realizar las respectivas correcciones	
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

RF-14	Modificar Notas/Calificaciones		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Valeria Cuji		
Fuentes	Docente		
Descripción	Permite modificar las calificaciones de los estudiantes de cada grado o curso		
Actor	Usuario del sistema		
Precondición	Ingresado al sistema. Acceder al módulo de calificaciones		
Secuencia Normal	**Paso**	**Secuencia**	
	1.	El usuario accede al menú de calificaciones y selecciona la opción d calificaciones.	
	2.	Es usuario selecciona el año lectivo, grado o curso y quimestre del c las calificaciones.	
	3.	El usuario modifica las calificaciones de cada estudiante dentro de c presiona el botón guardar.	
	4.	El sistema comprueba la validez de los datos, realiza cálculos para o almacena los datos.	
	5.	El sistema presenta el certificado o libreta de cada uno de los estudia calificaciones del quimestre del grado o curso seleccionado	
Post Condición	Los datos modificados por el usuario se almacena en la base de datos		
Excepciones	Paso	Acción	
	1	Si existe algún problema con los datos ingresados, el sistema informa a permite realizar las respectivas correcciones	
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

GENERAR REPORTES

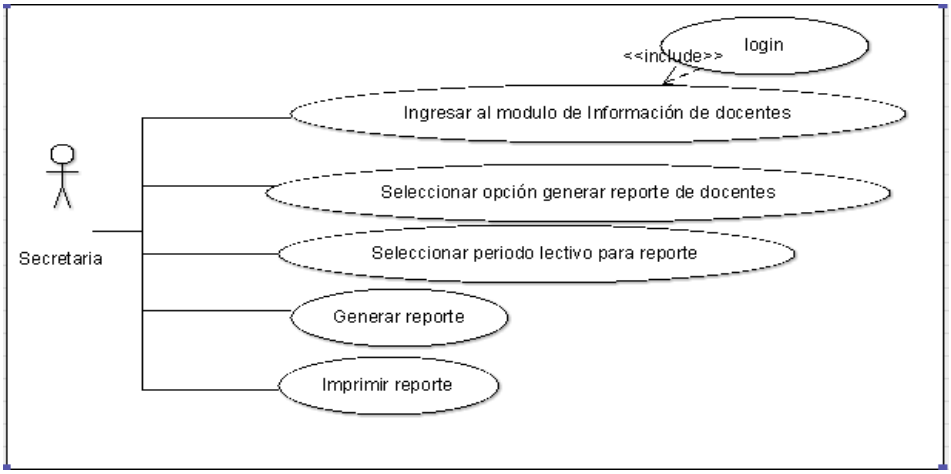


Tabla 21: Caso de uso Generar reporte de Docentes

RF-15	Generar reporte de Docentes		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Daniel Borja		
Fuentes	Secretaria		
Descripción	Permite generar un reporte o listado de los docentes que son guías de grado o curso		
Actor	Usuario del sistema		
Precondición	El actor secretaria debe estar conectado al sistema y haber accedido al módulo de información personal.		
Secuencia Normal	**Paso**	**Secuencia**	
	1.	El usuario accede al menú de Información del personal y selecciona reporte de Docentes.	
	2.	El sistema muestra en pantalla el formulario de ingreso de parámetro	
	3.	El usuario selecciona el periodo lectivo para el reporte	
	4.	El sistema genera el reporte según los parámetros ingresados y lo mu que pueda ser impreso por el usuario	
Post Condición	Los datos modificados por el usuario se almacena en la base de datos		
Excepciones	Paso	Acción	
	1.	El sistema genera el reporte pero no devuelve ningún resultado debido existen no cumplen con los parámetros de especificados, por lo que not regresa al formulario de ingreso de parámetros para generar otro reporte	
Prioridad	Baja		
Estado	En Construcción		
Estabilidad	Media		
Comentarios	No		

Tarea VI: Identificación de conflictos

	#	**No ambiguo, completo y correcto**	**Medible y Verificable**	**Alcanzable y realista**	**Consistente(No contradictorio)**	**No solape a otro requerimientos**
Interfaz De usuario (RIU)	1	Si	Si	Si	Si	Si
	2	Si	Si	Si	Si	Si
	3	Si	Si	Si	Si	Si
Interfaz de Hardware (RIU)	1	Si	Si	Si	Si	Si
Interfaz De Software (RISw)	1	Si	Si	Si	Si	Si
	2	Si	Si	Si	Si	Si
	3	Si	No	Si	Si	Si
	4	Si	Si	Si	Si	Si
	5	Si	Si	Si	Si	Si
Interfaz De Comunicación (RIC)	1	Si	Si	Si	Si	Si

	#	No ambiguo, completo y correcto	Medible y Verificable	Alcanzable y realista	Consistente(No contradictorio)	No solape a otro requerimientos
Requerimiento Funcional (RF)	1	Si	Si	Si	Si	Si
	2	Si	Si	Si	Si	Si
	3	Si	Si	Si	Si	Si
	4	Si	Si	Si	Si	Si
	5	Si	Si	Si	Si	Si
	6	Si	Si	Si	Si	Si
	7	Si	Si	Si	Si	Si
	8	Si	Si	Si	Si	Si
	9	Si	Si	Si	Si	Si
	10	No	No	Si	No	Si
	11	Si	Si	Si	Si	Si
	12	No	No	Si	Si	Si
	13	No	No	Si	No	Si
	14	Si	Si	Si	Si	Si
	15	Si	Si	Si	Si	Si
	16	Si	Si	Si	Si	Si
	17	Si	Si	Si	Si	Si
	18	Si	Si	Si	Si	Si
	19	Si	Si	Si	Si	Si
	20	Si	Si	Si	Si	Si
	21	Si	Si	Si	Si	Si

Tabla 22: Identificación de conflictos en requerimientos de Interfáz

Requerimiento no Functional (RNF)	#	No ambiguo, completo y correcto	Medible y Verificable	Alcanzable y realista	Consistente(No contradictorio)	No solape a otro requerimientos
Rendimiento	1	Si		Si	Si	Si
	2	Si		Si	Si	Si
Seguridad	1	Si	si	Si	Si	Si
Disponibilidad	1	Si		Si	Si	Si
Comunicación	1	Si		Si	Si	Si
	2	Si		Si	Si	Si
Mantenibilidad	1	Si	Si	Si	Si	Si
	1	Si	Si	Si	Si	Si
	2	Si	Si	Si	Si	Si

Tabla 23: Identificar conflictos en requerimientos no funcionales

Tarea VII: Solución y negociación de conflictos.

Requerimiento	Motivo del conflicto	Solución
El sistema debe permitir crear una matrícula (Información del estudiante, Fecha de matrícula, año lectivo) (RF10)	El requerimiento es ambiguo ya que no especifica o aclara a que estudiantes debe registrar en la matrícula, pues en esta actividad debería tomarse en cuenta si el estudiante ha aprobado el curso del periodo lectivo anterior y también debería tomarse en cuenta si el alumno que va a matricularse es nuevo y así realizar las siguiente matrícula.	Ubicamos la fuente de este requerimiento, Informamos sobre la ambigüedad de este requerimiento, pudimos llegar entender la necesidad real de este requerimiento. Que quedo planteado de esta manera: El sistema debe permitir realizar el registro de un estudiante en la institución validando que el estudiante haya aprobado el curso anterior, en caso de que el estudiante sea nuevo y desee matricularse el sistema permitirá ingresar los datos del nuevo estudiante y luego realizar su respectiva matrícula.
El sistema debe permitir listar una matrícula(Información del estudiante, Fecha de matrícula, Grado en que se matricula, especialidad)(RF12)	El requerimiento es ambiguo. Desde el punto de vista del analista el requerimiento debería estar expresado con más detalle.	Se ubica a la fuente de este requerimiento y se le informa de este conflicto con el fundamento de que el mismo no está expresado de forma adecuada, sugerimos que el requerimiento debería ser expresado así: El sistema debe permitir generar un reporte el cual mostrara información de la matrícula realizada, esta información podrá imprimirse y así entregar al

		matriculado un comprobante de la matrícula.
El sistema debe permitir eliminar una matrícula (RF13)	El requerimiento es ambiguo, y desde el punto de vista del analista este requerimiento no sería necesario.	Concertamos la reunión con la fuente de este este requerimiento, indagamos a cerca de la necesidad de este requerimiento la fuente explica que este requerimiento no tiene mayor relevancia si lo hacemos a un lado, entonces el analista dio explicación de una solución para este conflicto, entonces se estableció que en el requerimiento RF11 "El sistema permitirá modificar una matrícula" abarca la necesidad del usuario pues el mismo quiso interpretar como modificación de la matrícula no es necesario que una matrícula sea eliminada. Este conflicto permitió obtener otro requerimiento que estuvimos dejando de lado que es que el sistema permitirá cancelar la matricula

Estas soluciones planteadas se expresaran en el documento SRS.

5.4 Fase III: Especificación de Requerimientos

Especificación de requerimientos de Software

Proyecto: Sistema de Matrículas y registro de notas.

Versión: 1.0

Contenido

ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE 127

FICHA DEL DOCUMENTO 129

1. INTRODUCCION 130

1.1 PROPÓSITO 130
PERMITIR ESTABLECER LAS BASES DEL ACUERDO ENTRE USUARIOS EN LO QUE AL PROYECTO DE SOFTWARE SE REFIERE 130
1.2 ÁMBITO DEL SISTEMA 130
1.3 PERSONAL INVOLUCRADO 131
1.4 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS. 132
1.5 REFERENCIAS 133
1.6 VISIÓN GENERAL 133

2. DESCRIPCION GENERAL 133

2.1 PERSPECTIVA DEL PRODUCTO 133
2.2 FUNCIONALIDAD DEL PRODUCTO 134
2.2 CARACTERÍSTICAS DE LOS USUARIOS 135
2.3 RESTRICCIONES. 136
2.5 SUPOSICIONES Y DEPENDENCIAS. 137
2.6 EVOLUCIÓN PREVISIBLE DEL SISTEMA 137

3. REQUERIMIENTOS ESPECIFICOS 137

3.1 REQUERIMIENTOS COMUNES DE LOS INTERFACES. 137
3.1.1 Interfaces de usuario 137
3.1.2 Interfaces de Hardware 137
3.1.3 Interfaces de Software 138
3.1.4 Interfaces de Comunicación 138
3.2 REQUERIMIENTOS FUNCIONALES 138
3.3 REQUERIMIENTOS NO FUNCIONALES. 154

2.RAZONABLEMENTE EL SISTEMA DEBERÍA TARDAR 75 MILISEGUNDOS EN CADA TRANSACCIÓN, LO QUE SE TRADUCE EN 5 TRANSACCIONES CADA 16.6 SEGUNDOS. SOPORTANDO A 25 USUARIOS CONCURRENTEMENTE. ..... ¡ERROR! MARCADOR NO DEFINIDO.

3.3.1 Requerimientos de Rendimiento 154
3.3.2 Requerimientos de Seguridad 155
3.3.3 Requerimientos de Fiabilidad 155
3.3.4 Requerimientos de Disponibilidad 156
3.3.5 Requerimientos de Mantenibilidad 156
3.3.6 Requerimientos de Portabilidad 156

Ficha del Documento

Fecha	Versión	Autor	Verificado por
06/08/2013	1.0	Daniel Borja Valeria Cuji	

Documento validado por las parte en la fecha: 06/08/2013

Por el cliente	Por la empresa de desarrollo
Arq. Marcelo Cando	Daniel Borja Valeria Cuji.

1. INTRODUCCION

La presente especificación de requerimientos de software del sistema a construir surge para ser un conjunto de información necesaria que ayuda a los desarrolladores de software a analizar y entender todos los requerimientos que nuestro cliente desea, de la misma forma constituye un informe útil para que el cliente del producto final describa lo que el realmente desea obtener y de esta manera lograr tener un documento necesario cuya información en el futuro servirá para el desarrollo del software, es decir en la codificación correcta del mismo.

Se describirá en forma detallada las interfaces de usuario, de software, del hardware y comunicaciones, así como de los requerimientos funcionales y no funcionales.

1.1 Propósito

Permitir establecer las bases del acuerdo entre usuarios en lo que al proyecto de software se refiere.

1.2 Ámbito del sistema

Identificación del producto de software Sistema de Matrícula y calificaciones. Soportará las actividades relacionadas con la matrícula académica de los estudiantes, el registro de sus notas y la generación de reportes y certificados.

Objetivos del sistema.

- El sistema permitirá reemplazar el sistema actual.
- Crear sistema para automatizar el registro de notas y matriculas
- Almacenar información concerniente al registro de notas y matriculas
- Gestionar la información almacenada.
- Crear control de acceso al sistema de matrícula y el registro de notas
- Obtener reportes de la información almacenada respecto a los objetos involucrados en este sistema (Docentes, Estudiantes, Materias, Notas y Matriculas)

1.3 Personal involucrado

Nombres:	Carlos Daniel
Apellidos:	Borja Buestán
Dirección:	Bullcay
Teléfono:	0984718801
E-mail	danny_guala@hotmail.com
Instrucción	Superior
Rol:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Valeria Alexandra
Apellidos:	Cuji Torres
Dirección:	Benigno Vásquez y Cuenca
Teléfono:	3050876
E-mail	valeria_alex06@hotmail.com
Instrucción:	Superior
Tipo:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Mauricio
Apellidos:	Ortiz
Dirección:	Cuenca
Teléfono:	-
E-mail	mortizo@ups.edu.ec
Instrucción:	Superior
Rol:	Analista/Desarrollador
Cargo:	Supervisor del proyecto
Responsabilidades:	Realizar las revisiones del proyecto.

1.4 Definiciones, acrónimos y abreviaturas.

1.4.1 Definiciones

- Almacenamiento.- En relación con ordenadores o computadoras, cualquier dispositivo capaz de almacenar información procedente de un sistema informático.
- Base de Datos.- Cualquier conjunto de datos organizados para su almacenamiento en la memoria de un ordenador o computadora, diseñado para para facilitar su mantenimiento y acceso de una forma estándar.
- Interfaz.- medio que permite la comunicación entre el usuario y el sistema.
- Login.- Nombre o alias que se le da a una persona para permitirle el acceso al sistema.
- Password/Contraseña.- Clave para autentificar el ingreso a un lugar o sitio.
- Sistema Operativo.-Software básico que controla una computadora.
- MySQL: es un sistema de gestión de bases de datos relacional de licenciamiento libre.
- Gestión.- Administrar información de la base de datos (Insertar, Actualizar, Listar, Eliminar).

1.4.2 Acrónimos

- GUI o acrónimo de Graphical User Interface.- en informática, tipo de entorno que permite al usuario elegir comandos, iniciar programas, ver listas de archivos y otras opciones utilizando las representaciones visuales.
- ODBC.- Herramienta que conecta la base de datos con la interfaz.

1.4.3 Abreviaturas

- HW: Hardware
- SW: Software
- ARQ: Arquitecto.
- ING: Ingeniero.
- SRS: Especificación de Requerimientos de Software

1.5 Referencias

Referencia	Titulo	Ruta	Fecha	Autor
IEEE 830-1998	IEEE 830- 1998	http://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf	03-01-2013	IEEE
MEC	Ministerio de Educación	www.educacion.gov.ec	12/03/2013	Ministerio de Educación

1.6 Visión General

El sistema permitirá gestionar información referente a una matrícula y registro de notas en el colegio Técnico Industrial Gualaceo.

EL SRS está compuesto por:

Introducción: En esta sección se detalla los objetivos que tienen el SRS y nuestro sistema en forma general.

Descripción General: Describe una perspectiva general del producto a desarrollarse, como también las características del usuario y las limitaciones que podría tener.

Requerimientos específicos: Muestra paso a paso todos los requerimientos que el usuario desea en el producto final.

2. DESCRIPCION GENERAL

2.1 Perspectiva del producto

Se realizará un sistema para el Colegio Técnico Industrial Gualaceo, el cual necesita automatizar el proceso de matrícula y registro de notas. Este sistema pretende agilizar su proceso de matrícula de los cursos que ofrece la misma, así como almacenar distintas entidades que tienen relación con esta como lo son: los estudiantes, profesores, materias,

notas. A diferencia del sistema actual que según fuentes de la institución se encuentra obsoleto, este sistema nuevo pretende un producto más elaborado que permita un funcionamiento de calidad para que la información se mantenga integra y permita presentar reportes de información necesaria para toma de decisiones, además que la utilización de la aplicación sea de uso sencillo para los usuarios y así facilite su implementación.

1.2 Funcionalidad del producto

En términos generales el sistema deberá proporcionar soporte a las siguientes tareas de gestión de las matrículas y registro de calificaciones de los estudiantes de la Institución:

- Gestión de Docentes.
- Gestión de Notas de los estudiantes.
- Gestión de Materias.
- Gestión de Estudiantes.
- Realizar el proceso de matrícula de un estudiante.
- Generar Reportes.
    - Estudiantes registrados en un programa académico o periodo lectivo.
    - Docentes guías de las materias en el periodo lectivo.
    - Comprobante de registro de matrícula del estudiante.
    - Calificaciones del estudiante.

A continuación una descripción general de estas tareas:

Gestión de Docentes:

La información del docente será ingresada al sistema pudiendo modificar listar y eliminar al docente el usuario que realice estas operaciones deberá tener los permisos respectivos.

Gestión de Notas de los estudiantes.

Se ingresara las notas de los trabajos y tareas que el docente de a los estudiantes, el sistema evaluara todas estas notas permitiendo saber si el estudiante aprueba o no el curso que esté tomando.

Gestión de materias.

Se ingresara las diferentes materias que se vayan a dictar en un curso, pudiendo modificar, listar y eliminar información de esta.

Gestión de estudiantes

La información personal del estudiante se ingresara, modificara, se presentara una lista de los datos de este estudiante.

Realizar matrícula del estudiante.

Este proceso se realizara tomando en cuenta dos aspectos: Si el estudiante pertenece a la institución el sistema validara que haya aprobado el curso anterior para permitir matricular. Si el estudiante se inscribe por primera vez el sistema permitirá ingresar los datos del estudiante para realizar la respectiva matrícula.

Generar Reportes.

Permitirá presentar información referente a los estudiantes registrados en el periodo lectivo, los docentes guías de las materias en un curso, mostrar e imprimir comprobante de matrícula y generar reporte de las calificaciones del estudiante.

2.2 Características de los usuarios

Tipo de usuario	Docente.
Formación	Universitario, con título de tercer nivel.
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá listar los alumnos de los distintos años de básica, gestionar notas, faltas y generar reporte de notas

Tipo de usuario	Secretaria.
Formación	Universitario, con título de tercer nivel.
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá listar los alumnos de los distintos años de básica, gestionar notas, faltas y generar reporte de notas, gestionar, materias, estudiantes, docentes, matriculas, generar reportes varios.

Tipo de usuario	Estudiante
Formación	Estudiante
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario podrá visualizar la información de la institución así como las calificaciones registradas en las materias que este cursando y que haya cursado.

Tipo de usuario	Administrador
Formación	Universitario, con título de tercer nivel. Conocimientos altos en informática
Habilidades	Conocimiento mínimos en manejo de tecnología informática
Actividades	Este tipo de usuario se encargará de la gestión de la base de datos del sistema. Es decir, efectuará el alta y baja de los usuarios, asignaturas, año de básica, etc. Así como las modificaciones. En general tiene acceso a todo el sistema y podrá realizar todo tipo de procesos.

2.3 Restricciones.

El sistema de matrículas y calificaciones debe ser desarrollado con las siguientes restricciones:

Plataforma Windows, servidor de aplicación web Apache, Base de datos MySql, y lenguaje de programación JSF, para la interface Prime Faces.

Tiempo máximo de desarrollo 10 meses.

Costo máximo de desarrollo dependerá en este caso de lo que el cliente esté dispuesto a pagar ya que en la entrevista realizada una de las restricciones que expresa el entrevistado es que la asignación de recursos lo da el Ministerio de Educación. Como Ingenieros de requerimientos y, estimamos que el costo máximo del producto es $ 3500

2.5 Suposiciones y dependencias.

- El sistema se desarrollara con arquitectura de tres capas.
- El equipo en el cual funcionara la aplicación deberá contar con los paquetes de Java y la base de datos MySQL.

2.6 Evolución Previsible del sistema
Ninguna.

3. REQUERIMIENTOS ESPECIFICOS

3.1 Requerimientos comunes de los interfaces.

La interfaz del usuario deberá contener los campos necesarios para que el usuario o administrador del sistema agregue la información que requiera.

1.1.1 Interfaces de usuario

- La interfaz gráfica se creara de una manera de fácil comprensión para el usuario de manera que este no requiera mayor esfuerzo para utilizar el sistema.
- El sistema contara con manual de usuario para ir guiando en el manejo del sistema.
- Los usuarios ya deberían conocer, por experiencia en otros sistema como email y el navegador de internet las funcionalidades generales de un sistema web

1.1.2 Interfaces de Hardware

El sistema se ejecutara en una maquina con las siguientes características:

- Memoria mínima: 1 GB
- Memoria recomendada: 2 GB,
- Procesador mínimo: DualCore 1.6 GHz o similar,
- Procesador recomendado: Core i3 2.1 GHz o similar.
- Espacio en disco mínimo: 500 MB de espacio libre.
- Espacio en disco recomendado: 1 GB de espacio libre.

1.1.3 Interfaces de Software

- La base de datos que se utilizara es MySql versión 5.0
- Se requiere que el Equipo cuente con sistema operativo WINDOWS 7 o superior
- El sistema será compatible con el explorador Web MozilaFirefox.
- Lenguaje de programación para el sistema va a ser JAVA
- Como servidor web se utilizara Apache 2.0.

3.1.4 Interfaces de Comunicación

- Para la interacción con la base de datos se utilizara JDBC

1.2 Requerimientos Funcionales

<table>
<tr><td>RF-1</td><td colspan="3">Ingresar al sistema</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria, Estudiante, Docente, Administrador</td></tr>
<tr><td>Autor</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Administrador Informático de la Institución</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar al sistema</td></tr>
<tr><td>Precondición</td><td colspan="3">Usuario y contraseña</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td>El usuario ingresa nombre y contraseña</td></tr><tr><td>2</td><td>El sistema verifica los datos y dependiendo del usuario y clave le proporciona permisos de acceso</td></tr><tr><td>3</td><td>El usuario termina la sesión</td></tr></table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Mostrar la página de bienvenida</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td>El usuario puede salir del sistema o cancelar el ingreso al sistema</td></tr></table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Análisis</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-2</td><td colspan="3">Crear Periodos Lectivos</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear en la base de datos periodos lectivos</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema con los respectivos privilegios para gestionar periodos lectivos</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de periodos lectivos y seleccionar la opción crear nuevo periodo lectivo.</td></tr>
<tr><td>2</td><td>1. El usuario ingresa la fecha de inicio y la fecha final del periodo lectivo</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Periodo electivo guardado en la Base de Datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-3</td><td colspan="3">Modificar Periodos Lectivos</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar en la base de datos periodos lectivos</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema con los respectivos privilegios para gestionar periodos lectivos</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de periodos lectivos y seleccionar la opción modificar nuevo periodo lectivo.</td></tr>
<tr><td>2</td><td>Es el sistema permite buscar el periodo lectivo que desee modificar</td></tr>
<tr><td>3</td><td>El usuario modifica la fecha de inicio o la fecha final del periodo lectivo.</td></tr>
<tr><td>4</td><td>Usuario guarda información modificada</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Periodo electivo modificado y guardado en la Base de Datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-4</td><td colspan="3">Ingresar Docente</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear Docentes</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber Ingresado al sistema. Con privilegios para gestionar docentes</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de Docentes y selecciona la opción ingresar nueva Docente.</td></tr>
<tr><td>2</td><td>El usuario ingresa datos de Docente.</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4</td><td>El usuario accede al módulo de Docentes y selecciona la opción ingresar nueva Docente.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Datos de Docente almacenado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-5</td><td colspan="3">Modificar Docente</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar Docentes</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber Ingresado al sistema. Con privilegios para gestionar docentes</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>El usuario accede al módulo de Docentes y selecciona la opción modificar Docente.</td></tr>
<tr><td>1.</td><td>El sistema presenta los docentes.</td></tr>
<tr><td>2.</td><td>El usuario realiza la búsqueda del docente por apellido</td></tr>
<tr><td>3.</td><td>El usuario selecciona el docente que desea modificar</td></tr>
<tr><td>4.</td><td>El sistema presenta al usuario la información del docente</td></tr>
<tr><td>5.</td><td>El usuario modifica la información del Docente</td></tr>
<tr><td>6.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>7.</td><td>El usuario modifica la información del Docente</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Datos de Docente modificado y almacenado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-6</td><td colspan="3">Generar reporte de docentes</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite generar un reporte o listado de los docentes que son profesores de un curso o que poseen más de un grado asignado</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema y accedido al módulo Información académica</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Docentes y selecciona la opción modificar Docente.</td></tr>
<tr><td>2.</td><td>El sistema presenta los docentes.</td></tr>
<tr><td>3.</td><td>El usuario realiza la búsqueda del docente por apellido</td></tr>
<tr><td>4.</td><td>El usuario selecciona el docente que desea modificar</td></tr>
<tr><td>5.</td><td>El sistema presenta al usuario la información del docente</td></tr>
<tr><td>6.</td><td>El usuario modifica la información del Docente</td></tr>
<tr><td>7.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y almacena</td></tr>
<tr><td>8.</td><td>El usuario modifica la información del Docente</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos consultados no se alteran en la base de datos y el reporte debe poder imprimirse.</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El sistema genera el reporte pero no devuelve ningún resultado debido a que los registros que existen no cumplen con los parámetros especificados, por lo que se notifica al usuario y se regresa al formulario de ingreso de parámetros para generar otro reporte</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Baja</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-7</td><td colspan="3">Ingresar Estudiante</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear estudiantes</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de estudiantes y selecciona la opción ingresar nuevo estudiante.</td></tr>
<tr><td>2</td><td>El usuario ingresa datos de estudiante.</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4</td><td>El usuario accede al módulo de estudiantes y selecciona la opción ingresar nuevo estudiante.</td></tr>
<tr><td>5</td><td>El usuario ingresa datos de estudiante.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Estudiante ingresado en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
</table>

<table>
<tr><td>RF-8</td><td colspan="3">Modificar Estudiante</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y modificar estudiantes</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td>1</td><td>El usuario accede al módulo de estudiantes y selecciona la opción modificar nuevo estudiante.</td></tr>
<tr><td>2</td><td>El usuario modifica datos de estudiante.</td></tr>
<tr><td>3</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4</td><td>El usuario accede al módulo de estudiantes y selecciona la opción modificar nuevo estudiante.</td></tr>
<tr><td>5</td><td>El usuario modifica datos de estudiante.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Datos de Estudiante modificados</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-9</td><td colspan="3">Listar Estudiantes</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite listar información del estudiante</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber ingresado al sistema</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><td>Paso</td><td>Secuencia</td></tr>
<tr><td></td><td>El usuario accede al módulo de estudiantes y selecciona la opción mostrar estudiantes.</td></tr>
<tr><td></td><td>El usuario lista datos de estudiante.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Sistema presenta datos del estudiante</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Baja</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Baja</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-10</td><td colspan="3">Crear cursos/grados</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear en la base de datos Cursos</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber :
1. Ingresado al sistema.
2. Registrado docentes.
3. Registrado Materias.
4. Registrado periodo lectivo.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Grados y selecciona la opción crear nuevo grado.</td></tr>
<tr><td>2.</td><td>El usuario ingresa el periodo lectivo, materia, el docente, para el curso</td></tr>
<tr><td>3.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
<tr><td>4.</td><td>El usuario accede al módulo de Grados y selecciona la opción crear nuevo grado.</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los grados o curso se asignan</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-11</td><td colspan="3">Ingresar Materias</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar y crear materias</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber :
1. ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Materia y selecciona la opción ingresar nueva Materia.</td></tr>
<tr><td>2.</td><td>El usuario ingresa datos de Materia.</td></tr>
<tr><td>3.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Materia ingresada en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-12</td><td colspan="3">Modificar Materia</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar materias</td></tr>
<tr><td>Precondición</td><td colspan="3">El usuario debe haber :
2. ingresado al sistema.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al módulo de Materia y selecciona la materia que desea modificar</td></tr>
<tr><td>2.</td><td>El usuario modifica la Materia.</td></tr>
<tr><td>3.</td><td>El sistema comprueba que se hayan ingresado correctamente los datos y los almacena</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Materia modificada y almacenada en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existen errores el sistema informa al usuario sobre el error y le permite hacer modificaciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Media</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-13</td><td colspan="3">Ingresar Notas/Calificaciones</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Docente</td></tr>
<tr><td>Actores</td><td colspan="3">Docente, Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite ingresar las calificaciones de los estudiantes de cada grado o curso</td></tr>
<tr><td>Precondición</td><td colspan="3">Ingresado al sistema.
Acceder al módulo de calificaciones</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al menú de calificaciones y selecciona la opción de ingreso de calificaciones.</td></tr>
<tr><td>2.</td><td>Es usuario selecciona el año lectivo, grado o curso y quimestre del cual desea modificar las calificaciones.</td></tr>
<tr><td>3.</td><td>El usuario ingresa las calificaciones de cada estudiante dentro de cada asignatura y presiona el botón guardar.</td></tr>
<tr><td>4.</td><td>El sistema comprueba la validez de los datos, realiza cálculos para obtener promedios y almacena los datos.</td></tr>
<tr><td>5.</td><td>El sistema presenta el certificado o libreta de cada uno de los estudiantes y el cuadro de calificaciones del quimestre del grado o curso seleccionado</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos ingresados por el usuario se almacena en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existe algún problema con los datos ingresados, el sistema informa al usuario y se le permite realizar las respectivas correcciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-14</td><td colspan="3">Modificar Notas/Calificaciones</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Docente</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria, Docente</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite modificar las calificaciones de los estudiantes de cada grado o curso</td></tr>
<tr><td>Precondición</td><td colspan="3">Ingresado al sistema.
Acceder al módulo de calificaciones</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al menú de calificaciones y selecciona la opción de modificar calificaciones.</td></tr>
<tr><td>2.</td><td>Es usuario selecciona el año lectivo, grado o curso y quimestre del cual desea modificar las calificaciones.</td></tr>
<tr><td>3.</td><td>El usuario modifica las calificaciones de cada estudiante dentro de cada asignatura y presiona el botón guardar.</td></tr>
<tr><td>4.</td><td>El sistema comprueba la validez de los datos, realiza cálculos para obtener promedios y almacena los datos.</td></tr>
<tr><td>5.</td><td>El sistema presenta el certificado o libreta de cada uno de los estudiantes y el cuadro de calificaciones del quimestre del grado o curso seleccionado</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos modificados por el usuario se almacena en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>Si existe algún problema con los datos ingresados, el sistema informa al usuario y se le permite realizar las respectivas correcciones</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-15</td><td colspan="3">Matricular estudiante Nuevo</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Cuando el estudiante ingresa por primera vez a la Institución se le toman ciertos datos básicos. Su cédula, nombre, fecha de nacimiento, dirección, teléfono, escuela de donde viene.</td></tr>
<tr><td>Precondición</td><td colspan="3">El actor secretaria debe estar conectado al sistema y con permisos de registrar nuevos estudiantes.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3">
<table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El sistema inicia este caso de uso cuando el actor Secretaria ingresa a la opción de registrar nuevos estudiantes.</td></tr>
<tr><td>2.</td><td>El actor Secretaria ingresa la identificación del estudiante y el programa académico (Periodo Lectivo) que va a cursar.</td></tr>
<tr><td>3.</td><td>El sistema valida que el estudiante sea nuevo para ese programa en el sistema. Es decir, valida que no esté registrado en el sistema asociado al curso al cual se inscribe, y que no este activo en otro curso.</td></tr>
<tr><td>4.</td><td>El actor secretaria ingresa los datos básicos como nombre, fecha de nacimiento, dirección, teléfono, colegio de egreso, entre otros.</td></tr>
<tr><td>5.</td><td>El sistema valida que los datos ingresados estén correctos y completos.</td></tr>
<tr><td>6.</td><td>El sistema registra el estudiante y lo asocia de forma activa al programa académico al cual se inscribe.</td></tr>
</table>
</td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos modificados por el usuario se almacena en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3">
<table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1.</td><td>Si el estudiante está registrado y activo en el programa académico al cual se inscribe, el sistema presenta un mensaje informando que el estudiante ya está ingresado en el programa académico</td></tr>
<tr><td>2.</td><td>Si los datos ingresados no están correctos y/o completos el sistema solicita verificación o corrección de dato por dato y retorna al paso 4, a continuación este caso de uso continúa.</td></tr>
</table>
</td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

<table>
<tr><td>RF-16</td><td colspan="3">Generar reporte de Docentes</td></tr>
<tr><td>Versión</td><td>1.0</td><td>Fecha</td><td>07-08-2013</td></tr>
<tr><td>Autores</td><td colspan="3">Daniel Borja</td></tr>
<tr><td>Fuentes</td><td colspan="3">Secretaria</td></tr>
<tr><td>Actores</td><td colspan="3">Secretaria</td></tr>
<tr><td>Descripción</td><td colspan="3">Permite generar un reporte o listado de los docentes que son guías de grado o curso</td></tr>
<tr><td>Precondición</td><td colspan="3">El actor secretaria debe estar conectado al sistema y haber accedido al módulo de información personal.</td></tr>
<tr><td>Secuencia Normal</td><td colspan="3"><table>
<tr><th>Paso</th><th>Secuencia</th></tr>
<tr><td>1.</td><td>El usuario accede al menú de Información del personal y selecciona la opción Generar reporte de Docentes.</td></tr>
<tr><td>2.</td><td>El sistema muestra en pantalla el formulario de ingreso de parámetros para el reporte</td></tr>
<tr><td>3.</td><td>El usuario selecciona el periodo lectivo para el reporte</td></tr>
<tr><td>4.</td><td>El sistema genera el reporte según los parámetros ingresados y lo muestra en pantalla para que pueda ser impreso por el usuario</td></tr>
</table></td></tr>
<tr><td>Post Condición</td><td colspan="3">Los datos modificados por el usuario se almacena en la base de datos</td></tr>
<tr><td>Excepciones</td><td colspan="3"><table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1.</td><td>El sistema genera el reporte pero no devuelve ningún resultado debido a que los registros que existen no cumplen con los parámetros de especificados, por lo que notifica al usuario y se regresa al formulario de ingreso de parámetros para generar otro reporte</td></tr>
</table></td></tr>
<tr><td>Prioridad</td><td colspan="3">Baja</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Media</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
<tr><td>Prioridad</td><td colspan="3">Alta</td></tr>
<tr><td>Estado</td><td colspan="3">En Construcción</td></tr>
<tr><td>Estabilidad</td><td colspan="3">Alta</td></tr>
<tr><td>Comentarios</td><td colspan="3">No</td></tr>
</table>

RF-17	Matricular estudiante antiguo		
Versión	*1.0*	**Fecha**	07-08-2013
Autores	Daniel Borja		
Fuentes	Secretaria		
Actores	Sistema		
Requerimientos asociados	Ninguno		
Descripción	Este caso de uso describe el registro de un estudiante que ya pertenece a la institución		
Precondición	El estudiante debe estar registrado en la base de datos del sistema		
Secuencia Normal	**Paso**	**Secuencia**	
	1.	**El sistema** valida que el usuario hay aprobado el curso de un periodo anterior.	
	2.	**El sistema** registra al estudiante al nuevo programa académico que le corresponda	
Post Condición	Registro del estudiante a un programa académico.		
Excepciones	Paso	Acción	
	1.	**Si** el estudiante no ha aprobado el nivel anterior el sistema no permitirá registrar al estudiante en un programa académico. Presentando un mensaje informando que el estudiante no puede ser matriculado ya q ya no ha aprobado el nivel anterior.	
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3 Requerimientos no Funcionales.

3.3.1 Requerimientos de Rendimiento

RNF-1	**Respaldo y soporte**		
Descripción	El servidor de base de datos, deberá tener un respaldo apropiado, así como personal técnico listo para cualquier eventualidad		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

RNF-2	Respuesta a consultas		
Descripción	Razonablemente el sistema debería tardar 75 milisegundos en cada transacción, lo que se traduce en 5 transacciones cada 16.6 segundos. Soportando a 25 usuarios concurrentemente		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.2 Requerimientos de Seguridad

RNF-3	Respuesta a consultas		
Descripción	Uso de contraseñas para cada usuario (administrador, Docente, Secretaria). Esto permitirá que tengan acceso al sistema solo las personas que tienen autorización.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.3 Requerimientos de Fiabilidad

RNF-4	Información incorrecta		
Descripción	El sistema deberá evitar el ingreso de información incorrecta permitiendo al usuario corregir al usuario guiando por medio de mensajes de información que genere el sistema.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	Arq. Marcelo Cando		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.4 Requerimientos de Disponibilidad

RNF-5	**Información incorrecta**		
Descripción	El sistema deberá evitar el ingreso de información incorrecta por medio de mensajes que genere el sistema.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.5 Requerimientos de Mantenibilidad

RNF-6	**Manejo del sistema**		
Descripción	El sistema incluirá documentación de ayuda, incluyendo manual de usuario.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.6 Requerimientos de Portabilidad

RNF-7	**Lenguaje de programación**		
Descripción	Utilizar el lenguaje y plataforma JAVA.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes			
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

RNF-8	**Lenguaje de programación**		
Descripción	Utilizar como servidor de Base de datos MySQL		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Prioridad	Alta		
Estado	En Construcción		

Estabilidad	Alta
Comentarios	No

5.5 Fase IV: Validación de Requerimientos

Verificación de los Requerimientos

Para la verificación de:

Nombre del Proyecto	**Sistema de matrículas y calificaciones**
Nombre del Documento	**Especificación de requerimientos de software de Sistema de matrículas y calificaciones**
Fecha	**09/08/2013**

Criterio	Sí / No / NA
1. Correctitud — La Especificación de un Requerimiento es correcta si, y solo si, el sistema/software alcanza todos y cada uno de los requerimientos en él especificados.	
Desde el punto de vista del usuario, ¿se ha especificado el tiempo de respuesta esperado de todas las operaciones necesarias?	Si
¿Se han especificado otras consideraciones temporales tales como el tiempo de procesamiento, el de transferencia de datos o la tasa de transferencia?	Si
¿Se han especificado todas las tareas que debe realizar el sistema/software?	Si
Para cada tarea especificada, ¿se ha detallado el contenido de datos/información utilizado por la tarea y el contenido de datos/información que se obtendrá como resultado de la misma?	Si
¿Se han establecido los requerimientos sobre la seguridad física?	NA
¿Se han establecido los requerimientos sobre la seguridad operacional?	NA
¿Se ha especificado la fiabilidad del sistema/software, incluyendo las consecuencias en el caso de que falle, la información vital a proteger en caso de caída, la detección de los errores o el proceso de recuperación?	NA
¿Las compensaciones establecidas entre los atributos que compiten son aceptables, por ejemplo, entre robusteza y correctitud?	NA
¿Se han definido las interfaces internas, como por ejemplo el software o el hardware?	Si
¿Se han definido las interfaces externas, como por ejemplo usuarios o hardware?	NA
¿Se ha incluido la definición de *éxito*? ¿Se ha incluido la definición de *fracaso*?	NA
¿Cada requerimiento es relevante para el problema y su solución?	Si
2. No Ambiguo — Una Especificación de los Requerimientos es no ambigua si, y solo si, cada requerimiento especificado en ella posee exclusivamente una única interpretación.	
¿Los requerimientos se han especificado de forma suficientemente clara para que si se entregan a un grupo independiente para la implementación, dicho grupo sea capaz de entenderlos?	Si
¿Los requerimientos funcionales se encuentran separados de los no-funcionales?	Si
¿Los requerimientos están especificados de forma concisa, de modo que evitan la posibilidad de hacer múltiples interpretaciones de ellos?	Si
¿Todos los requerimientos evitan conflictos con otros requerimientos?	Si
3. Completitud — Una Especificación de los Requerimientos es completa si, y solo si, incluye los siguientes elementos: • Todos los requerimientos significativos, ya sea relacionados con la	

funcionalidad, con el rendimiento, las limitaciones de diseño, los atributos o las interfaces externas. • Las definiciones de las respuestas del sistema/software a todas las clases posibles de datos de entrada en todos los tipos posibles de situaciones. • Etiquetas descriptivas y referencias a todas las figuras, tablas y diagramas de la Especificación de los Requerimientos, así como la definición de todos los términos y unidades de medición.	
¿Se han especificado todas las interfaces de comunicación, incluyendo su aceptación de la negociación, su control de errores y los protocolos de comunicación?	Si
¿Se ha realizado el análisis para identificar los requerimientos que no se han tenido en cuenta?	Si
¿Se han especificado las áreas de incompletitud para cuando la información no esté disponible?	Si
¿Los requerimientos son completos, tales que si el producto satisface todos estos requerimientos, será aceptable?	Si
¿Es posible implementar todos y cada uno de los requerimientos?	Si
¿Se ha especificado la mantenibilidad del sistema/software, incluyendo la habilidad de respuesta a los cambios en el entorno operativo, las interfaces, la precisión, el rendimiento, y otras capacidades adicionales predecibles?	NA
¿Se han especificado los requerimientos para la comunicación entre los componentes del sistema/software?	Si
¿Se ha definido la funcionalidad y el comportamiento global de todo el sistema/software?	Si
¿Se han establecido de forma explícita y sin ambigüedades las restricciones, suposiciones y dependencias apropiadas?	Si
¿Se ha especificado adecuadamente la infraestructura tecnológica para el sistema/software?	Si
¿Se han etiquetado de forma descriptiva todas las figuras, tablas y diagramas?	Si
¿Se han referenciado dentro del documento todas las figuras, tablas y diagramas?	Si
4. Consistencia — La consistencia se refiere a la consistencia interna. Si la Especificación de los Requerimientos no concuerda con el resto de documentación de la organización y del proyecto, significa que no es correcta.	
¿Se han especificado los requerimientos con un nivel de detalle consistente?	Si
¿Algunos de los requerimientos tienen que especificarse con mayor detalle?	Si
¿Algunos de los requerimientos deben ser especificados con menor detalle?	Si
¿Los requerimientos están en concordancia con el contenido del resto de documentación de la organización o del proyecto?	
5. Categorizado por importancia y/o estabilidad – Una Especificación de los Requerimientos se categoriza por importancia y/o estabilidad si cada requerimiento particular especificado en ella posee un identificador que establece su importancia o estabilidad. Ejemplos de rangos de categorización incluyen esencial, condicional u opcional. La estabilidad puede ser especificada en términos del número de cambios esperados para un requerimiento.	
a. ¿Los requerimientos poseen asociado un identificador para indicar la importancia o la estabilidad de un requerimiento en particular?	Si
b. ¿Existen conflictos en relación a la categorización de la importancia y/o estabilidad de los requerimientos?	No
6. Verificable — Una Especificación de los Requerimientos es verificable si, y solo si, cada requerimiento especificado en ella es verificable. Un requerimiento es verificable si, y solo si, existe un proceso finito y rentable con el cual una persona o máquina puede comprobar que el sistema/software cumple con dicho requerimiento.	
¿El lenguaje y vocabulario con el que están escritos los requerimientos es entendible para los stakeholders? ¿Los stakeholders coinciden?	Si
¿Cada requerimiento puede ser probado? A partir de pruebas independientes,	Si

¿puede ser posible determinar cuándo se satisface cada requerimiento?	
7. Modificable — Una Especificación de los Requerimientos es modificable si, y solo si, su estructura y estilo son tales que cualquier cambio en los requerimientos puede realizarse de forma fácil, completa y consistente, conservando la estructura y el estilo.	
¿Los requerimientos se identifican de forma única?	
¿Se han consolidado los requerimientos redundantes?	
¿Cada requerimiento se ha especificado de forma separada, evitando requerimientos compuestos?	
8. Trazable — Una Especificación de los Requerimientos es trazable si el origen de cada uno de sus requerimientos es claro y si facilita la referenciación de cada requerimiento en el desarrollo futuro o mejora la documentación.	
¿Se ha identificado cada requerimiento con el fin de facilitar su referenciación en el futuro desarrollo o en los esfuerzos de mejora?	Si
¿Cada requerimiento posee una referencia a los requerimientos previos del proyecto que están relacionados con él?	Si

5.6 Fase V: Verificación de Requerimientos

Se entrega el documento de especificación de requerimientos de software al cliente en este caso al Arq. Marcelo Cando rector del Colegio Técnico Industrial Gualaceo, quien supo informar que el documento estas detallado claramente y que lo había entendido en su totalidad, entonces para verificar que el documento sea el adecuado se entregó también a otros usuarios como son la secretaria y el encargado del departamento de informática estos usuarios expresaron sentirse identificados con lo antes dicho por el Rector de la institución que el Documento cumple con las necesidades expresadas en el momento de la captura.

CAPITULO VI

Implementación de la aplicación para realizar la Especificación de Requerimientos.

6.1. Análisis

Para realizar el análisis de software se va a aplicar la metodología que se ha propuesto en el capítulo 5, con esto se busca conseguir mejores resultados.

6.1.1 Fase de Elicitación

Tarea I. Estudio Inicial

En primer lugar se debe considerar que el proyecto a realizarse no es un proyecto solicitado por clientes determinados sino más bien un proyecto dirigido al mercado de desarrollo de software que recolecte los requerimientos y utilice el documento de especificación de requerimientos.

Con la implementación del sistema se busca principalmente obtener el informe de manera rápida ingresando todos los datos de manera manual.

Como se ha visto a lo largo de la investigación en los capítulos anteriores existen diferentes metodologías creadas para realizar el proceso de ingeniería de requerimientos

y así también existen herramientas automatizadas que ayudan en la ejecución de este proyecto.

Al no ser un proyecto a la medida no se utilizan técnicas como entrevistas, cuestionarios en esta etapa sino más bien se investiga diferentes herramientas existentes en el mercado, e información sobre el tema, se realiza brainstorming y si fuera necesario se utilizaran prototipos para conocer como funcionara el mismo.

A partir de investigaciones realizadas sobre el tema se considera factible la creación de este sistema para generar el Documento de Especificación de Requerimientos, ya que según autores una buena propuesta metodológica lleva consigo una herramienta que facilite el uso de esta.

De acuerdo a las técnicas usadas se consideran los siguientes participantes en este proyecto:

Personal Involucrado

Nombres:	Carlos Daniel
Apellidos:	Borja Buestan
Dirección:	Bullcay
Teléfono:	0984718801
E-mail	danny_guala@hotmail.com
Instrucción	Superior
Rol:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Valeria Alexandra
Apellidos:	Cuji Torres
Dirección:	Benigno Vásquez y Cuenca
Teléfono:	3050876
E-mail	valeria_alex06@hotmail.com
Instrucción:	Superior
Tipo:	Analista/Desarrollador

Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Mauricio
Apellidos:	Ortiz
Dirección:	Cuenca
Teléfono:	-
E-mail	mortizo@ups.edu.ec
Instrucción:	Superior
Rol:	Analista/Desarrollador
Cargo:	Supervisor del proyecto
Responsabilidades:	Realizar las revisiones del proyecto.

Características de los Usuarios

Tipo de usuario	Usuario del Sistema.
Formación	Universitario, con formación en Ingeniera de sistemas informáticos y afines
Habilidades	Conocimiento en la recolección de requerimientos para la creación de proyectos.
Actividades	Podrá agregar, modificar, eliminar información a la base de datos del sistema, así como también podrá generar los reportes necesarios.

Tarea II. Objetivos del Sistema

Se obtuvieron varias ideas para el proyecto las mismas son:

- Que cumpla con el estándar IEEE 830-1998.
- Registrar los objetivos del sistema
- Que permita ingresar el nombre de la empresa que requiere el sistema y
- El nombre de la empresa que suministra el sistema
- El sistema estará basado solo en el documento de especificación no en ninguna otra fase del proceso.
- El sistema debe tener seguridad

- El sistema debe permitir registrar una introducción la cual se dividirá en propósito, alcance, personal involucrado, definiciones, acrónimos y abreviaturas, también referencias y resumen.
  - Permitirá ingresar los Stakeholders que tendrán nombre, apellidos, dirección, teléfono, cargo, actividad que realiza, experiencia.
  - Cada requerimiento deberá tener un identificador único.
  - Se podrá administrar la información almacenada(eliminar, modificar, insertar)
  - El sistema pedirá usuario y contraseña.
  - Cada uno delos campos del sistema llevara consigo etiquetas de información
  - Fácil de entender y manejar
  - Los reportes deben salir con el logotipo de la empresa.
  - Permitirá visualizar reportes(paginas enumeradas),
  - Ingresar requerimientos funcionales y requerimientos no funcionales
  - Clasificar requerimientos no funcionales.
  - Permitirá ingresar actores.
  - Cada requerimiento se debe registrar su versión de manera manual.
  - El usuario final podrá crear nuevos proyectos para cada cliente.

OBJETIVOS DEL SISTEMA

OBJ-1. Crear una herramienta que genere el Documento de Especificación de Requerimientos de Software basada en el estándar IEEE 830-1998

OBJ-1	**Generar el ERS basado en el estándar IEEE 830-1998**
Versión	1.0
Autores	Daniel Borja
Fuentes	Plantilla ERS IEEE 830-1998
Descripción	El sistema deberá generar un reporte en donde el documento tenga los puntos establecidos en el estándar IEEE 830-1998
Importancia	Alta
Urgencia	Indispensable
Estado	En construcción
Estabilidad	5

OBJ-2. Almacenar la información del proyecto tal como: nombre de la empresa, participantes, introducción, descripción del proyecto y requerimientos.

OBJ-2	**Almacenar información del proyecto**
Versión	1.0
Autores	Valeria Cuji
Fuentes	Plantilla ERS IEEE 830-1998
Descripción	El sistema deberá almacenar los datos que el usuario agrega tales como introducción, descripción del proyecto y requerimientos.
Importancia	Alta
Urgencia	Indispensable
Estado	
Estabilidad	5

OBJ-3. Contar con un sistema que permita administrar los datos almacenados (modificar, eliminar)

OBJ-3	**Gestionar los datos almacenados**
Versión	1.0
Autores	Daniel Borja
Fuentes	Plantilla ERS IEEE 830-1998
Descripción	El sistema permitirá al usuario modificar y eliminar información almacenada.
Importancia	Alta
Urgencia	Indispensable
Estado	
Estabilidad	5

OBJ-4. Tener un sistema seguro que no permita la manipulación de datos a terceras personas sino solo al personal involucrado en el proyecto

OBJ-4	**Contar con usuario y contraseña.**
Versión	1.0
Autores	Valeria Cuji
Fuentes	

Descripción	El sistema deberá contar con la seguridad necesaria para cada usuario.
Importancia	Alta
Urgencia	Indispensable
Estado	
Estabilidad	5

Tarea III y Tarea IV.- identificar Requerimientos del Sistema y Priorizar Requerimientos

ID	REQUERIMIENTO	OBJ. ASOCIADO	FUENTE	AUTOR	PRIORIDAD
RU-1	El sistema gestionara (insertar, modificar, eliminar) la información de los participantes del proyecto así como de los usuarios del sistema.	OBJ 2 OBJ 3		Valeria Cuji	ALTA
RU-2	El sistema gestionara (insertar, modificar, eliminar, visualizar) información de la introducción del ERS.	OBJ 2 OBJ 3		Daniel Borja	ALTA
RU-3	El sistema gestionara (insertar, modificar) información de la Descripción General del Proyecto.	OBJ 2 OBJ 3		Daniel Borja	ALTA
RU-4	El sistema gestionara (insertar, modificar, eliminar, listar) los requerimientos descritos por el cliente.	OBJ 2 OBJ 3		Valeria Cuji	ALTA
RU-5	El sistema gestionara los requerimientos no funcionales	OBJ 2 OBJ 3		Valeria Cuji	ALTA
RU-6	El sistema deberá permitir la clasificación de los requerimientos no funcionales para la aplicación de la plantilla.	OBJ 1		Valeria Cuji	ALTA
RU-7	El sistema deberá generar reportes.	OBJ 1		Daniel Borja	ALTA
RU-8	El sistema permitirá cargar imágenes si lo deseara el usuario final, las imágenes que se suban será porque en la funcionalidad del producto se puede describir textualmente o subir una imagen con la estructura que indique la funcionalidad del producto	OBJ 1		Daniel Borja	MEDIA
RU-9	El sistema contara con claves de acceso	OBJ 4			
RD-1	El sistema deberá poder ser ampliado en un futuro con módulos necesarios en la ingeniería de requerimientos			Daniel Borja	MEDIA
RD-2	El sistema será creado en lenguaje Java Enterprice Edition			Ing. Mauricio Ortiz	MEDIA
RD-3	El sistema se diseñara como una interfaz web			Ing. Mauricio Ortiz	ALTA
RD-4	Se usara una base de datos relacional.			Ing. Mauricio Ortiz	MEDIA
RD-5	Se debe evitar el ingreso de información fallida por medio de mensajes que genere el sistema.	OBJ 4		Daniel Borja	ALTA
RD-6	La información que se muestre en el reporte deberá ser clara y precisa.	OBJ 1		Valeria Cuji	ALTA
RI-1	Interfaz amigable y fácil de comprender			Valeria Cuji	ALTA
RI-2	Se usaran colores estándares de Windows			Daniel Borja	ALTA
RI-3	En la pantalla principal del sistema habrá una imagen de bienvenida y un botón que lleve a la página de inicio de sesión	OBJ 4		Daniel Borja	MEDIA

6.1.2 Fase de Análisis.

Tarea V. Clasificación de Requerimientos.

Requerimientos comunes de los Interfaces			
Usuario	**Hardware**	**Software**	**Comunicación**
1. La interfaz del usuario deberá ser amigable y fácil de manejar, de esta manera el usuario podrá interactuar con el sistema sin que exista problemas. 2. En la pantalla principal existirá un mensaje de bienvenida con el logotipo de la herramienta, e incluirá un link que enlazara a la pantalla del inicio de sesión. 3. El sistema permitirá cargar imágenes al reporte que se generara en caso que el usuario lo requiera	1. **Sistema operativo:** Windows 7 de 32 bits, en todas sus versiones 2. **Procesador mínimo:** DualCore 1.6 GHz o similar 3. **Espacio en disco mínimo:** 500 MB de espacio libre 4. **Procesador recomendado:** Core i3 2.1 GHz o similar 5. **Espacio en disco recomendado:** 1 GB de espacio libre 6. **Memoria mínima:** 1 GB 7. **Memoria recomendada:** 2 GB	1. **Base de Datos:** Mysql 2. El sistema no será integrado a ningún otro. **3. Plataforma de Desarrollo:** *Java Enterprise Edition* 4. **Lenguajes de Programación:**Java, JSF, XML, JPQL, SQL **5. Framework MVC:** JSD 6. **Implementación JSF para Vista**: Primefaces **7. Servidor de Aplicaciones:** GlassFish Server	1. Se usara el protocolo de comunicación entre el programa y la base de datos el driver de java JDBC

Requerimientos Funcionales

REQUERIMIENTOS ESPECÍFICOS					
Requerimientos Funcionales					
RF-1	Autentificar usuarios			**RF-30**	Modificar resumen
RF-2	Ingresar Actor	**RF-15**	Ingresar Requerimientos Interfaz de software	**RF-31**	Modificar referencias
RF-3	Ingresar personal involucrado			**RF-32**	Modificar perspectiva del producto
RF-4	Ingresar definiciones acrónimos y abreviaturas	**RF-16**	Ingresar Requerimientos Interfaz de Comunicación	**RF-33**	Eliminar perspectiva del producto
RF-5	Ingresar Referencias			**RF-34**	Modificar Funcionalidad
RF-6	Ingresar Resumen	**RF-17**	Ingresar Requerimientos No Funcionales	**RF-35**	Listar Características de los usuarios
RF-7	Ingresar perspectiva del producto				
RF-8	Ingresar Funcionalidad del producto	**RF-18**	Ingresar Requerimientos Funcionales	**RF-36**	Modificar Características de los usuarios
RF-9	Ingresar Características de los Usuarios	**RF-19**	Ingresar Otros Requerimientos	**RF-37**	Listar Restricciones del producto
		RF-20	Ingresar Apéndices		
		RF-22	Generar Reportes		
RF-10	Ingresar Restricciones del Producto	**RF-23**	Listar Actores	**RF-38**	Modificar Restricciones del producto
		RF-24	Modificar actores		
RF-11	Ingresar Suposiciones y Dependencias	**RF-25**	Eliminar Actor	**RF-39**	Listar requerimientos de interfaz de usuario.
RF-12	Ingresar evolución previsible del	**RF-26**	Listar Personal Involucrado	**RF-40**	Modificar requerimientos de interfaz de usuario.
RF-13	Ingresar Requerimientos Interfaz de Usuario	**RF-27**	Modificar Personal Involucrado	**RF-41**	Eliminar requerimientos de interfaz de usuario.
RF-14	Ingresar Requerimientos	**RF-28**	Eliminar Personal Involucrado	**RF-42**	Listar requerimientos de interfaz de
		RF-28	Eliminar definiciones, acrónimos y abreviaturas		
			Interfaz de Hardware		

	hardware.		interfaz de hardware.		funcionales
RF-43	Modificar requerimientos de interfaz de hardware.	**RF-48**	Listar requerimientos no funcionales.	**RF-54**	Listar otros requerimientos
RF-44	Eliminar requerimientos de interfaz de hardware.	**RF-49**	Modificar requerimientos no funcionales.	**RF-55**	Modificar otros requerimientos
RF-45	Listar requerimientos de interfaz de comunicación.	**RF-50**	Eliminar requerimientos no funcional	**RF-56**	Eliminar otros requerimientos
RF-46	Modificar requerimientos de interfaz de comunicación.	**RF-51**	Listar requerimientos funcionales.	**RF-57**	Generar Reporte del proyecto
RF-47	Eliminar requerimientos de	**RF-52**	Modificar requerimientos funcionales.		
		RF-53	Eliminar requerimientos		

Requerimientos NO funcionales.

Requerimientos No Funcionales					
Rendimiento	**Seguridad**	**Disponibilidad**	**Fiabilidad**	**Mantenibilidad**	**Portabilidad**
El sistema permite el uso de dos usuarios al mismo tiempo permitiendo un manejo más eficiente y confortable del sistema. El tiempo de respuesta del sistema en cada función que el usuario solicite será de 8 segundos. El sistema permite el registro de varios usuarios, el registro de mucha información que el usuario necesite	La seguridad de sistema es por uso de contraseñas la cual será renovada cada mes. La información de los reportes deberá ser clara y precisa de acuerdo a lo que el usuario solicite.	En caso de que el usuario olvide su usuario/contraseña podrá contactar al administrador quien le proveerá una nueva. El sistema deberá poder ser usado en toda la jornada laboral del usuario. La información ingresada en el sistema podrá ser visualizada , cambiada y usada para obtener reportes por el usuario registrado	El sistema deberá evitar el ingreso de información incorrecta por medio de mensajes que genere el sistema.	El sistema deberá ser creado de manera tal que en un futuro se puedan agregar más módulos.	El sistema será desarrollado en su totalidad en Java Enterprise Edition y con MySQl como base de datos debido a su licencia GPL para evitar costos y problemas de licenciamiento.

Tarea VI. Identificar conflictos

Comparativa de requerimientos Funcionales

Tabla 24 Matriz comparativa entre requerimientos Funcionales.

Requerimiento	No ambiguo, completo y correcto	Medible y Verificable	Alcanzable y realista	Consistente(No contradictorio)
RF – 1	Si	Si	Si	Si
RF – 2	Si	Si	Si	Si
RF – 3	Si	Si	Si	Si
RF – 4	Si	Si	Si	Si
RF – 5	Si	Si	Si	Si
RF – 6	Si	Si	Si	Si
RF – 7	Si	Si	Si	Si
RF – 8	Si	Si	Si	Si
RF – 9	Si	Si	Si	Si
RF – 10	Si	Si	Si	Si
RF – 11	Si	Si	Si	Si
RF – 12	Si	Si	Si	Si
RF – 13	Si	Si	Si	Si
RF – 14	Si	Si	Si	Si
RF – 15	Si	Si	Si	Si
RF – 16	Si	Si	Si	Si
RF – 17	Si	Si	Si	Si
RF – 18	Si	Si	Si	Si
RF – 19	Si	Si	Si	Si
RF – 20	Si	Si	Si	Si
RF – 21	Si	Si	Si	Si
RF – 22	No	Si	Si	Si
RF – 23	Si	Si	Si	Si
RF – 24	Si	Si	Si	Si
RF – 25	Si	Si	Si	Si
RF – 26	Si	Si	Si	Si
RF – 27	Si	Si	Si	Si
RF – 28	Si	Si	Si	Si
RF – 29	Si	Si	Si	Si

RF – 30	Si	Si	Si	Si
RF – 31	Si	Si	Si	Si
RF – 32	Si	Si	Si	Si
RF – 33	Si	Si	Si	Si
RF – 34	Si	Si	Si	Si
RF – 35	Si	Si	Si	Si
RF – 36	Si	Si	Si	Si
RF – 37	Si	Si	Si	Si
RF – 38	Si	Si	Si	Si
RF – 39	Si	Si	Si	Si
RF – 40	Si	Si	Si	Si
RF – 41	Si	Si	Si	Si
RF – 42	Si	Si	Si	Si
RF – 43	Si	Si	Si	Si
RF – 44	Si	Si	Si	Si
RF – 45	Si	Si	Si	Si
RF – 46	Si	Si	Si	Si
RF – 47	Si	Si	Si	Si
RF – 48	Si	Si	Si	Si
RF – 49	Si	Si	Si	Si
RF – 50	Si	Si	Si	Si
RF – 51	Si	Si	Si	Si
RF – 52	Si	Si	Si	Si
RF – 53	Si	Si	Si	Si
RF – 54	Si	Si	Si	Si
RF – 55	Si	Si	Si	Si
RF – 56	Si	Si	Si	Si
RF – 57	Si	Si	Si	Si

Tarea VII. Solucionar conflictos

Requerimiento	Motivo del conflicto	Solución
El sistema debe permitir generar reportes.(RF-22)	Ambigüedad en la descripción del requrimiento.	El requerimiento quedará de la siguiente manera: El sistema permitirá generar un reporte del documento de especificación de requerimientos de software el informe deberá poder visualizarse en el navegador web, se podrá guardar el archivo generado en formato .pdf.

6.1.3 Fase de Especificación

1 Introducción

La presente especificación de requerimientos de software (SRS) del proyecto a desarrollar se realiza con el objetivo de facilitar un conjunto de información necesaria que ayuda a los desarrolladores del software a analizar y entender todos los requerimientos que el usuario final desea , así mismo esta información contiene descripciones útiles sobre las funciones reales del sistema y de esta manera lograr tener un documento necesario cuya información en el futuro servirá para el desarrollo del software, es decir en la codificación correcta del mismo. Se describirá en forma detallada las interfaces de usuario, de software, del hardware y comunicaciones, así como de los requerimientos del cliente, atributos del sistema entre otros.

1.1 Propósito

- Permitir establecer las bases de acuerdo entre usuarios en lo que al proyecto de software se refiere.
- Ayudar a los usuarios finales del software a entender exactamente qué es lo que el cliente de software desea.
- El documento va dirigido a los usuarios directos de este sistema es decir a las personas que labora en el departamento de Sistemas y desarrollan software.

1.2 Ámbito del sistema

El sistema a desarrollar se denominara SysToERS (Sistema para Especificación de Requerimientos de Software)

Este tipo de herramienta permite automatizar la fase de descripción de los requerimientos, ya que consta de métodos que permitirá el ingreso y el almacenamiento de la información que se obtiene en la fase de elicitación y análisis y tendrá como resultado final el documento que se presentara al cliente (ERS).

1.3 Personal Involucrado

Intervendrá el siguiente personal

Nombres:	Carlos Daniel
Apellidos:	Borja Buestan
Dirección:	Bullcay
Teléfono:	0984718801
E-mail	danny_guala@hotmail.com
Instrucción	Superior
Rol:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Valeria Alexandra
Apellidos:	Cuji Torres
Dirección:	Benigno Vásquez y Cuenca
Teléfono:	3050876
E-mail	valeria_alex06@hotmail.com
Instrucción:	Superior
Tipo:	Analista/Desarrollador
Cargo:	Analista, Desarrollador
Responsabilidades:	Recolectar requerimientos, analizarlos para su posterior implementación.

Nombres:	Mauricio
Apellidos:	Ortiz
Dirección:	Cuenca
Teléfono:	-
E-mail	mortizo@ups.edu.ec
Instrucción:	Superior
Rol:	Analista/Desarrollador
Cargo:	Supervisor del proyecto
Responsabilidades:	Realizar las revisiones del proyecto.

1.4 Definiciones, Acrónimos, Abreviaturas

1.4.1 DEFINICIONES

- **Gestión**: Insertar, Modificar, Eliminar y Listar la información almacenada en el sistema.

- **Base de datos:** es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores, y en donde todas las operaciones de la base de datos operan sobre estas tablas.

- **IEEE:** Instituto de ingenieros eléctricos y electrónicos.

- **IEEE 830-1998:** Estándar que sirve para realizar el documento de especificación de requerimientos de software, la última versión es la del año 1998**.**

- **MySQL:** es un sistema de gestión de bases de datos relacional de licenciamiento libre.

1.4.2 ACRÓNIMOS

- **ERS:** Especificación de Requerimientos de Software.

1.4.3 ABREVIATURAS

- **SW**: Software
- **Ing.:** Ingeniero(a)
- **RU:** Requerimientos de Usuario
- **RD:** Requerimientos de Desarrollador
- **RI:** Requerimientos de Interfaz

1.5 Referencias

Referencia	Titulo	Ruta	Fecha	Autor
IEEE 830- 1998	IEEE 830-1998	http://www.ctr.unican.es/asignaturas/is1/IEEE830_esp.pdf	03-01-2013	IEEE

1.6 Visión General

El sistema permitirá gestionar (ingresar, editar, eliminar) información referente al documento de Especificación de Requerimientos de Software basado en el estándar IEEE 830-1998, permitiendo el acceso únicamente a usuarios previamente registrados y así el manejo de la información sea el adecuado.

Consta de las siguientes secciones:

- Una introducción, en esta sección se describe los objetivos existentes para el sistema a desarrollar.
- Descripción General, donde se describe la perspectiva general del sistema futuro, también se describirán las características de los usuarios y las distintas restricciones o limitaciones que podrían tener.
- Requerimientos Específicos, Muestra paso a paso todos los requerimientos que el usuario desea en el producto final.

2. DESCRIPCION GENERAL

2.1 Perspectivas del Producto

El sistema es un producto independiente que permitirá la gestión de la información relativa al Documento de Especificación de Requerimientos de Software basado en el estándar IEEE 830-1998.

2.2 Funcionalidad del producto

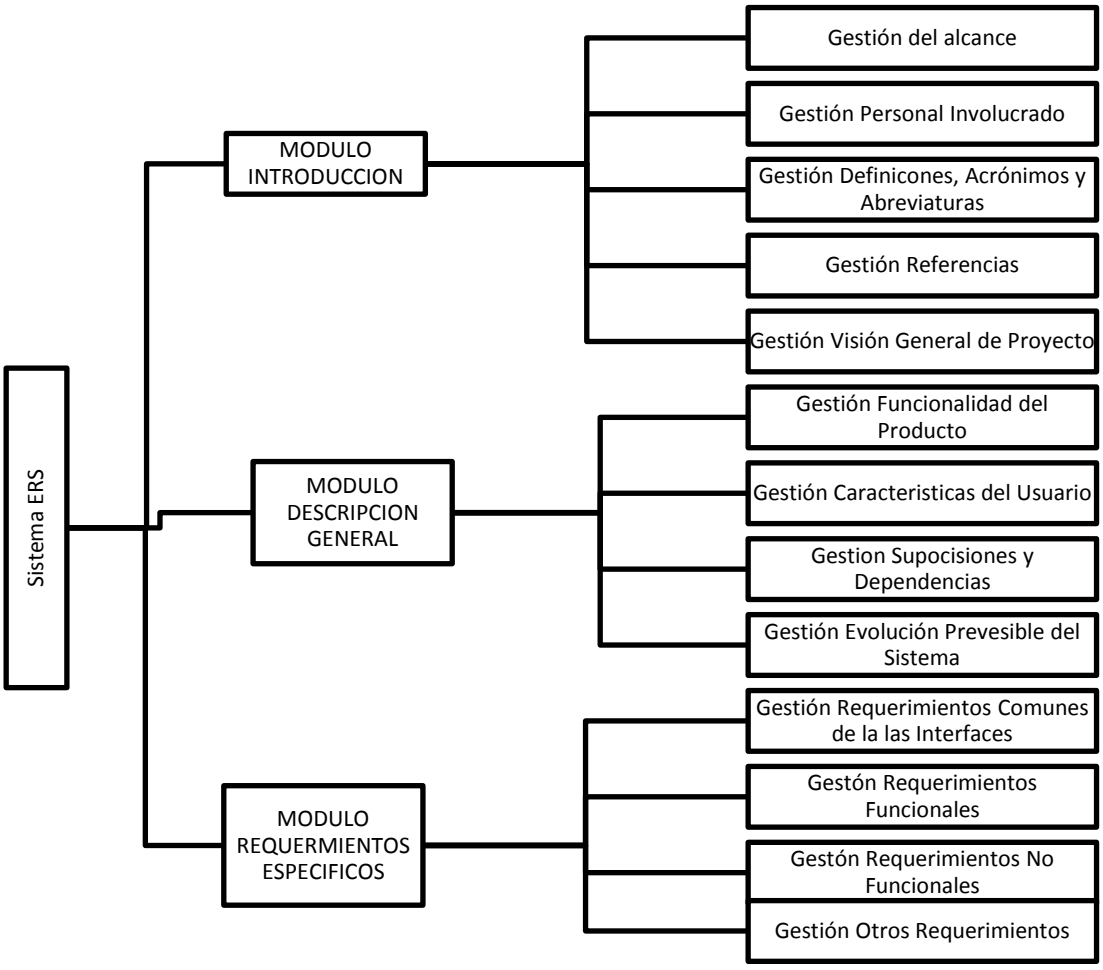


Ilustración 33: Funcionalidad del Producto.

2.3. Características de los Usuarios

Tipo de usuario	Usuario del Sistema.
Formación	Universitario, con formación en Ingeniera de sistemas informáticos y afines
Habilidades	Conocimiento en la recolección de requerimientos para la creación de proyectos.
Actividades	Podrá agregar, modificar, eliminar información a la base de datos del sistema, así como también podrá generar los reportes necesarios.

2.4. Restricciones

- Para desarrollar la propuesta se usara lenguaje java específicamente j2ee, que consta Jsf + Pimefaces para la interfaz, EclipseLink + JPA como motor de base de datos. Los datos serán almacenados en la base de datos relacional MySql.

- El sistema operativo que se usara es Windows 7
- Se usara como servidor apache
- La aplicación será diseñada para funcionar en Mozilla Firefox.

2.5 SUPOCISIONES Y DEPENDENCIAS

- El sistema se desarrollara con arquitectura de tres capas.
- El equipo en el cual funcionara la aplicación deberá contar con los paquetes de Java y la base de datos MySQL.

2.6 EVOLUCIÓN PREVISIBLE DEL SISTEMA

- Interfaz más amigable al usuario
- En un futuro se podrá agregar a la herramienta un módulo para las diferentes fases de la ingeniería de requerimientos.

3 REQUERIMIENTOS ESPECÍFICOS

3.1 Requerimientos comunes de los interfaces

La interfaz del usuario deberá contener los campos necesarios para que el usuario o administrador del sistema agregue la información que requiera.

3.1.1 Interfaces de usuario

- La interfaz del usuario deberá ser amigable y fácil de manejar, de esta manera el usuario podrá interactuar con el sistema sin que exista problemas

- En la pantalla principal existirá un mensaje de bienvenida con el logotipo de la herramienta, e incluirá un link que enlazara a la pantalla del inicio de sesión.

- El sistema permitirá cargar imágenes al reporte que se generara en caso que el usuario lo requiera.

3.1.2 Interfaces de Hardware

El sistema se ejecutara en un equipo con las siguientes características:

- **Sistema operativo:** Windows 7 de 32 bits, en todas sus versiones
- **Procesador mínimo:** DualCore 1.6 GHz o similar
- **Procesador recomendado:** Core i3 2.1 GHz o similar
- **Memoria mínima:** 1 GB
- **Memoria recomendada:** 2 GB
- **Espacio en disco mínimo:** 500 MB de espacio libre
- **Espacio en disco recomendado:** 1 GB de espacio libre.

3.1.3 Interfaces de Software

- **Plataforma de Desarrollo:** Java Enterprise Edition
- **Lenguajes de Programación:**Java, JSF, Javascript, XML
- **Framework MVC:** JSF
- **Implementación JSF para Vista**: Primefaces
- **Servidor de Aplicaciones:** GlassFish
- **Base de Datos:** Mysql
- El sistema no será integrado a ningún otro.

3.1.3 Interfaces de Comunicación

- Se usara el protocolo de comunicación entre el programa y la base de datos el driver de java JDBC.

3.2 REQUERIMIENTOS FUNCIONALES.

Actores del sistema.

Actor	**Usuario Final del Sistema**
Descripción	Este actor representa al personal que hará uso del sistema.

Actor	**Sistema**
Descripción	Este actor representa el sistema que se encargara de realizar todas las validaciones de los datos que el Usuario Final del Sistema ingrese.

RF-1	Autentificar usuarios.
Versión	Versión 1, 01-01-2013
Autores	Valeria Cuji
Fuentes	Plantilla ERS
Actor	Sistema
Descripción	Iniciar sesión en el sistema mediante el ingreso del usuario y contraseña
Precondición	Ninguna
Secuencia Normal	Paso: Acción 1: El usuario ingresa usuario y contraseña 2: El sistema valida datos ingresados 3: El sistema presenta la ventana con opciones para realizar en ingreso de los datos para la especificación de software
Post Condición	Sesión Iniciada
Excepciones	Paso: Acción 1: Si el usuario o contraseña son incorrectos, el sistema presenta un mensaje indicando el error y luego permite que el usuario ingrese de nuevo los datos.
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-2	Ingresar actores
Versión	Versión 1, 01-01-2013
Autores	Valeria Cuji
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar datos de los actores(nombre, apellido, descripción del actor)
Precondición	Inicio de Sesión
Secuencia Normal	**Paso** / **Acción** 1 / Ingresar información del actor en los campos de que presente la interfaz del sistema
Post Condición	Información del actor almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En Construcción
Estabilidad	Alta
Comentarios	Ninguna

RF-3	Ingresar personal involucrado
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema debe permitir registrar los datos del personal involucrado en el proyecto(nombre, rol, categoría profesional, responsabilidades, información de contacto aprobación)
Precondición	Inicio de sesión (usuario y contraseña)
Secuencia Normal	Paso / Acción 1 / El usuario ingresa información del personal involucrado en proyecto a desarrollar 2 / El sistema almacena la información
Post Condición	Información del personal involucrado almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

<table>
<tr><td>RF-4</td><td>Ingresar definiciones, acrónimos y abreviaturas</td></tr>
<tr><td>Versión</td><td>Versión 1, 01-01-2013</td></tr>
<tr><td>Autores</td><td>Daniel Borja</td></tr>
<tr><td>Fuentes</td><td>Plantilla SRS IEEE 830</td></tr>
<tr><td>Actor</td><td>Usuario Final del Sistema</td></tr>
<tr><td>Descripción</td><td>El sistema permitirá ingresar las definiciones los acrónimos y abreviaturas referente a el proyecto a desarrollar</td></tr>
<tr><td>Precondición</td><td>el usuario final debe iniciar sesión (usuario y contraseña)</td></tr>
<tr><td>Secuencia Normal</td><td>
<table>
<tr><td>Paso</td><td>Acción</td></tr>
<tr><td>1</td><td>El usuario ingresa las abreviaturas del proyecto a desarrollar</td></tr>
<tr><td>2</td><td>El usuario ingresa la definición de los términos a utilizar en el proyecto a desarrollar</td></tr>
<tr><td>3</td><td>El usuario ingresa los acrónimos del proyecto a desarrollar</td></tr>
<tr><td>4</td><td>El sistema almacena la información de las definiciones de los términos, acrónimos y abreviaturas.</td></tr>
</table>
</td></tr>
<tr><td>Post Condición</td><td>Definiciones, abreviaturas y acrónimos almacenados en la base de datos</td></tr>
<tr><td>Excepciones</td><td>Ninguna</td></tr>
<tr><td>Prioridad</td><td>Media</td></tr>
<tr><td>Estado</td><td>En construcción</td></tr>
<tr><td>Estabilidad</td><td>Alta</td></tr>
<tr><td>Comentarios</td><td>Ninguno</td></tr>
</table>

RF-5	Ingresar Referencias
Versión	Versión 1, 01-01-2013
Autores	Valeria Cuji
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar las referencias del utilizadas para la realización del documento de especificación requerimientos(Titulo, Ruta, Fecha, Autor)
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario ingresa el título, la ruta, la fecha y el autor del documento que ha utilizado para la especificación de requerimientos 3 | El sistema almacena la información de las referencias ingresadas.
Post Condición	Referencias almacenados en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-6	Ingresar Resumen
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar las un resumen del sobre el contenido de información del documento de especificación de requerimientos)
Precondición	El usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario ingresa el resumen del documento de especificación de requerimientos 2 | El sistema almacena la información registrada por el usuario.
Post Condición	Resumen almacenado en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-7	Ingresar Perspectiva del Producto
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar información referente a la perspectiva del producto
Precondición	El usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa la descripción de la perspectiva del producto a desarrollar Paso: 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Perspectiva del producto almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-8	Ingresar Funcionalidad del Producto
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar información referente a la funcionalidad del producto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: — El usuario ingresa la descripción de la funcionalidad del producto a desarrollar Paso: — La descripción de la funcionalidad deberá permitir ingresar la descripción de funcionalidad ya sea de forma textual o cargar una imagen que muestre una descripción de la misma. Paso: — El sistema almacena la información registrada por el usuario.
Post Condición	Funcionalidad del producto almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-9	Ingresar Características de los Usuarios
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar las características de los usuarios que utilizaran el proyecto a desarrollar (Tipo de usuario, formación, habilidades y actividades)
Precondición	El usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario ingresa las características de *usuario* 2 | El sistema almacena la información registrada por el usuario.
Post Condición	Características de los usuarios almacenado en la base de datos
Excepciones	Ninguna
Prioridad	Media
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-10	Ingresar Restricciones del Producto
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar las restricciones existentes en el desarrollo del proyecto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario ingresa la descripción de la funcionalidad del producto a desarrollar 2 | El sistema almacena la información registrada por el usuario.
Post Condición	Restricciones del producto almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-11	Ingresar Suposiciones y Dependencias
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar las suposiciones y dependencias existentes para el desarrollo del proyecto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa suposiciones y dependencias del producto a desarrollar 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Suposiciones y dependencias almacenadas en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-12	Ingresar evolución previsible del sistema
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar la información de acuerdo a la evolución previsible del sistema para el desarrollo del proyecto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa evolución previsible del producto a desarrollar 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Evolución previsible del sistema almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-14	Ingresar Requerimientos Interfaz de Usuario
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar la información referente a Requerimientos relacionados con la interfaz de usuario para el desarrollo del proyecto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — Acción: El usuario ingresa la información relacionada con la interfaz de usuario Paso: 2 — Acción: El sistema almacena la información registrada por el usuario.
Post Condición	Información relacionada con la interfaz de usuario almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-15	Ingresar Requerimientos Interfaz de Hardware
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar la información referente a Requerimientos relacionados con la interfaz de hardware para el desarrollo del proyecto
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — Acción: El usuario ingresa la información relacionada con la interfaz de Hardware Paso: 2 — Acción: El sistema almacena la información registrada por el usuario.
Post Condición	Información relacionada con la interfaz de hardware almacenada en la base de datos
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

<table>
<tr><td>RF-16</td><td>Ingresar Requerimientos Interfaz de software</td></tr>
<tr><td>Versión</td><td>Versión 1, 01-01-2013</td></tr>
<tr><td>Autores</td><td>Daniel Borja</td></tr>
<tr><td>Fuentes</td><td>Plantilla SRS IEEE 830</td></tr>
<tr><td>Actor</td><td>Usuario Final del Sistema</td></tr>
<tr><td>Descripción</td><td>El sistema permitirá ingresar la información referente a Requerimientos relacionados con la interfaz de software para el desarrollo del proyecto</td></tr>
<tr><td>Precondición</td><td>el usuario final debe iniciar sesión (usuario y contraseña)</td></tr>
<tr><td>Secuencia Normal</td><td><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td>El usuario ingresa la información relacionada con la interfaz de Software</td></tr><tr><td>2</td><td>El sistema almacena la información registrada por el usuario.</td></tr></table></td></tr>
<tr><td>Post Condición</td><td>Información relacionada con la interfaz de Software almacenada en la base de datos</td></tr>
<tr><td>Excepciones</td><td>Ninguna</td></tr>
<tr><td>Prioridad</td><td>Alta</td></tr>
<tr><td>Estado</td><td>En construcción</td></tr>
<tr><td>Estabilidad</td><td>Alta</td></tr>
<tr><td>Comentarios</td><td>Ninguno</td></tr>
</table>

<table>
<tr><td>RF-17</td><td>Ingresar Requerimientos Interfaz de Comunicación</td></tr>
<tr><td>Versión</td><td>Versión 1, 01-01-2013</td></tr>
<tr><td>Autores</td><td>Daniel Borja</td></tr>
<tr><td>Fuentes</td><td>Plantilla SRS IEEE 830</td></tr>
<tr><td>Actor</td><td>Usuario Final del Sistema</td></tr>
<tr><td>Descripción</td><td>El sistema permitirá ingresar la información referente a Requerimientos relacionados con la interfaz de comunicación</td></tr>
<tr><td>Precondición</td><td>el usuario final debe iniciar sesión (usuario y contraseña)</td></tr>
<tr><td>Secuencia Normal</td><td><table><tr><td>Paso</td><td>Acción</td></tr><tr><td>1</td><td>El usuario ingresa la información relacionada con la interfaz de comunicación</td></tr><tr><td>2</td><td>El sistema almacena la información registrada por el usuario.</td></tr></table></td></tr>
<tr><td>Post Condición</td><td>Información relacionada con la interfaz de Comunicación almacenada en la base de datos</td></tr>
<tr><td>Excepciones</td><td>Ninguna</td></tr>
<tr><td>Prioridad</td><td>Alta</td></tr>
<tr><td>Estado</td><td>En construcción</td></tr>
<tr><td>Estabilidad</td><td>Alta</td></tr>
<tr><td>Comentarios</td><td>Ninguno</td></tr>
</table>

RF-18	Ingresar Requerimientos No Funcionales
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar los Requerimientos funcionales del sistema a desarrollar (identificador, nombre del requerimiento, Versión, Autores, Fuentes, Objetivos asociados, Requerimientos asociados, Descripción, Excepciones, Prioridad, Estado, Estabilidad y comentarios)
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa información referente a los requerimientos no funcionales del proyecto en desarrollo Paso: 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Requerimientos no funcionales almacenados.
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-19	Ingresar Requerimientos Funcionales
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar los Requerimientos funcionales del sistema a desarrollar (identificador, nombre del requerimiento, Versión, Autores, Fuentes, Objetivos asociados, Requerimientos asociados, Descripción, Precondición, Secuencia normal, Post Condición, Excepciones, Prioridad, Estado, Estabilidad y comentarios)
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa información referente a los requerimientos funcionales del proyecto en desarrollo Paso: 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Requerimientos funcionales almacenados.
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-20	Ingresar Otros Requerimientos
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar otros Requerimientos
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa información referente Otros requerimientos Paso: 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Otros Requerimientos almacenados.
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-21	Ingresar Apéndices
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Plantilla SRS IEEE 830
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá ingresar Apéndices
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — El usuario ingresa información referente al Apéndice del Documento de especificación de requerimientos Paso: 2 — El sistema almacena la información registrada por el usuario.
Post Condición	Apéndice almacenado.
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-22	Generar Reportes
Versión	Versión 1, 01-01-2013
Autores	Daniel Borja
Fuentes	Ing. Mauricio Ortiz
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá generar el reporte del documento de ERS
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso: 1 — Acción: El sistema Genera el reporte ERS
Post Condición	Reporte visualizado
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-23	Listar Actores
Versión	Versión 1, 02-01-2013
Autores	Daniel Borja
Fuentes	Ing. Mauricio Ortiz
Actor	Usuario Final del Sistema
Descripción	Permitir listar a los Actores del sistema
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso 1 — El usuario selecciona la opción de listar Actores Paso 2 — El sistema presenta lista de los actores
Post Condición	Información del actor Modificada
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-24	Modificar actores
Versión	Versión 1, 02-01-2013
Autores	Daniel Borja
Fuentes	NO
Actor	Usuario Final del Sistema
Descripción	El sistema permitirá la modificación de los actores del proyecto en desarrollo
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario selecciona el actor a modificar 2 | El sistema presenta la información a modificar 3 | El usuario modifica la información del actor
Post Condición	Información del actor modificada
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

RF-25	Eliminar Actores
Versión	Versión 1, 02-01-2013
Autores	Daniel Borja
Fuentes	NO
Actor	Usuario Final del Sistema
Descripción	Permitir eliminar actores
Precondición	el usuario final debe iniciar sesión (usuario y contraseña)
Secuencia Normal	Paso | Acción 1 | El usuario selecciona al actor 2 | El usuario elimina al actor
Post Condición	Actor eliminado
Excepciones	Ninguna
Prioridad	Alta
Estado	En construcción
Estabilidad	Alta
Comentarios	Ninguno

3.3 REQUERIMIENTOS NO FUNCIONALES

3.3.1 REQUERIMIENTOS DE RENDIMIENTO

RNF-1	**Número de usuarios**		
Descripción	El sistema permite el uso de dos usuarios al mismo tiempo permitiendo un manejo más eficiente y confortable del sistema		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	Plantilla ERS		
Prioridad	Alta		
Estado	En Construcción		
Comentarios	no		

RNF-2	**Tiempo de Respuesta**		
Descripción	El tiempo máximo de respuesta del sistema en cada función que el usuario solicite será de 8 segundos.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	Plantilla ERS		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	no		

RNF-3	**Registro de varios usuarios**		
Descripción	El sistema permite el registro de varios usuarios, el registro de mucha información que el usuario necesite		
Versión	Versión 1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	Plantilla ERS		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	No		

3.3.2 REQUERIMIENTOS DE SEGURIDAD

RNF-4	**Claves de acceso**		
Descripción	La seguridad de sistema es por uso de contraseñas la cual será renovada cada mes.		
Versión	1.0	**Fecha**	30-12-212
Autores	Daniel Borja		
Fuentes	Plantilla ERS		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	no		

RNF-5	**Información clara**		
Descripción	La información de los reportes deberá ser clara y precisa de acuerdo a lo que el usuario solicite.		
Versión	1.0	**Fecha**	30-12-212
Autores	Daniel Borja		
Fuentes	Plantilla ERS		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	no		

3.3.3 REQUERIMIENTOS DE FIABILIDAD

RNF-6	**Información incorrecta**		
Descripción	El sistema deberá evitar el ingreso de información incorrecta por medio de mensajes que genere el sistema.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

3.3.4 REQUERIMIENTOS DE DISPONIBILIDAD

RNF-7	**Restaurar claves**		
Descripción	En caso de que el usuario olvide su usuario/contraseña podrá contactar al administrador quien le proveerá una nueva.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

RNF-8	**Uso del sistema**		
Descripción	El sistema deberá poder ser usado en toda la jornada laboral del usuario.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

RNF-9	**Información del sistema**		
Descripción	La información ingresada en el sistema podrá ser visualizada , cambiada y usada para obtener reportes por el usuario registrado		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

3.3.5 REQUERIMIENTOS DE MANTENIBILIDAD

RNF-10	**Susceptible a cambios futuros**		
Descripción	El sistema deberá ser creado de manera tal que en un futuro se puedan agregar más módulos.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

3.3.6 REQUERIMIENTOS DE PORTABILIDAD

RNF-11	**Lenguaje programación**		
Descripción	El sistema será desarrollado en su totalidad en Java Enterprise Edition y con MySQl como base de datos debido a su licencia GPL para evitar costos y problemas de licenciamiento.		
Versión	1.0	**Fecha**	30-12-2012
Autores	Daniel Borja, Valeria Cuji		
Fuentes	-		
Prioridad	Alta		
Estado	En Construcción		
Estabilidad	Alta		
Comentarios	Ninguno		

4 APENDICES

6.1.4 Fase de Validación y Verificación

Verificación de los Requerimientos

Para la verificación de:	
Nombre del Proyecto	Sistema para la Especificación de Requerimientos de Software SysToErs
Nombre del Documento	Especificación de requerimientos de software para la Especificación de Requerimientos basado en el estándar IEEE 830-1998
Fecha	09/08/2013

Criterio	Sí / No / NA
1. Correctitud — La Especificación de un Requerimiento es correcta si, y solo si, el sistema/software alcanza todos y cada uno de los requerimientos en él especificados.	
Desde el punto de vista del usuario, ¿se ha especificado el tiempo de respuesta esperado de todas las operaciones necesarias?	Si
¿Se han especificado otras consideraciones temporales tales como el tiempo de procesamiento, el de transferencia de datos o la tasa de transferencia?	Si
¿Se han especificado todas las tareas que debe realizar el sistema/software?	Si
Para cada tarea especificada, ¿se ha detallado el contenido de datos/información utilizado por la tarea y el contenido de datos/información que se obtendrá como resultado de la misma?	Si
¿Se han establecido los requerimientos sobre la seguridad física?	NA
¿Se han establecido los requerimientos sobre la seguridad operacional?	NA
¿Se ha especificado la fiabilidad del sistema/software, incluyendo las consecuencias en el caso de que falle, la información vital a proteger en caso de caída, la detección de los errores o el proceso de recuperación?	
¿Las compensaciones establecidas entre los atributos que compiten son aceptables, por ejemplo, entre robusteza y correctitud?	NA
¿Se han definido las interfaces internas, como por ejemplo el software o el hardware?	Si
¿Se han definido las interfaces externas, como por ejemplo usuarios o hardware?	NA
¿Se ha incluido la definición de *éxito*? ¿Se ha incluido la definición de *fracaso*?	NA
¿Cada requerimiento es relevante para el problema y su solución?	Si
2. No Ambiguo — Una Especificación de los Requerimientos es no ambigua si, y solo si, cada requerimiento especificado en ella posee exclusivamente una única interpretación.	
¿Los requerimientos se han especificado de forma suficientemente clara para que si se entregan a un grupo independiente para la implementación, dicho grupo sea capaz de entenderlos?	Si

Criterio	Sí / No / NA
¿Los requerimientos funcionales se encuentran separados de los no-funcionales?	Si
¿Los requerimientos están especificados de forma concisa, de modo que evitan la posibilidad de hacer múltiples interpretaciones de ellos?	Si
¿Todos los requerimientos evitan conflictos con otros requerimientos?	Si
3. Completitud — Una Especificación de los Requerimientos es completa si, y solo si, incluye los siguientes elementos: • Todos los requerimientos significativos, ya sea relacionados con la funcionalidad, con el rendimiento, las limitaciones de diseño, los atributos o las interfaces externas. • Las definiciones de las respuestas del sistema/software a todas las clases posibles de datos de entrada en todos los tipos posibles de situaciones. • Etiquetas descriptivas y referencias a todas las figuras, tablas y diagramas de la Especificación de los Requerimientos, así como la definición de todos los términos y unidades de medición.	
¿Se han especificado todas las interfaces de comunicación, incluyendo su aceptación de la negociación, su control de errores y los protocolos de comunicación?	Si
¿Se ha realizado el análisis para identificar los requerimientos que no se han tenido en cuenta?	Si
¿Se han especificado las áreas de incompletitud para cuando la información no esté disponible?	Si
¿Los requerimientos son completos, tales que si el producto satisface todos estos requerimientos, será aceptable?	Si
¿Es posible implementar todos y cada uno de los requerimientos?	Si
¿Se ha especificado la mantenibilidad del sistema/software, incluyendo la habilidad de respuesta a los cambios en el entorno operativo, las interfaces, la precisión, el rendimiento, y otras capacidades adicionales predecibles?	NA
¿Se han especificado los requerimientos para la comunicación entre los componentes del sistema/software?	Si
¿Se ha definido la funcionalidad y el comportamiento global de todo el sistema/software?	Si
¿Se han establecido de forma explícita y sin ambigüedades las restricciones, suposiciones y dependencias apropiadas?	Si
¿Se ha especificado adecuadamente la infraestructura tecnológica para el sistema/software?	Si
¿Se han etiquetado de forma descriptiva todas las figuras, tablas y diagramas?	Si
¿Se han referenciado dentro del documento todas las figuras, tablas y diagramas?	Si
4. Consistencia — La consistencia se refiere a la consistencia interna. Si la Especificación de los Requerimientos no concuerda con el resto de documentación de la organización y del proyecto, significa que no es correcta.	
¿Se han especificado los requerimientos con un nivel de detalle consistente?	Si
¿Algunos de los requerimientos tienen que especificarse con mayor detalle?	Si

Criterio	Sí / No / NA
¿Algunos de los requerimientos deben ser especificados con menor detalle?	Si
¿Los requerimientos están en concordancia con el contenido del resto de documentación de la organización o del proyecto?	
5. Categorizado por importancia y/o estabilidad – Una Especificación de los Requerimientos se categoriza por importancia y/o estabilidad si cada requerimiento particular especificado en ella posee un identificador que establece su importancia o estabilidad. Ejemplos de rangos de categorización incluyen esencial, condicional u opcional. La estabilidad puede ser especificada en términos del número de cambios esperados para un requerimiento.	
a. ¿Los requerimientos poseen asociado un identificador para indicar la importancia o la estabilidad de un requerimiento en particular?	Si
b. ¿Existen conflictos en relación a la categorización de la importancia y/o estabilidad de los requerimientos?	No
6. Verificable — Una Especificación de los Requerimientos es verificable si, y solo si, cada requerimiento especificado en ella es verificable. Un requerimiento es verificable si, y solo si, existe un proceso finito y rentable con el cual una persona o máquina puede comprobar que el sistema/software cumple con dicho requerimiento.	
¿El lenguaje y vocabulario con el que están escritos los requerimientos es entendible para los stakeholders? ¿Los stakeholders coinciden?	Si
¿Cada requerimiento puede ser probado? A partir de pruebas independientes, ¿puede ser posible determinar cuándo se satisface cada requerimiento?	Si
7. Modificable — Una Especificación de los Requerimientos es modificable si, y solo si, su estructura y estilo son tales que cualquier cambio en los requerimientos puede realizarse de forma fácil, completa y consistente, conservando la estructura y el estilo.	
¿Los requerimientos se identifican de forma única?	Si
¿Se han consolidado los requerimientos redundantes?	Si
¿Cada requerimiento se ha especificado de forma separada, evitando requerimientos compuestos?	Si
8. Trazable — Una Especificación de los Requerimientos es trazable si el origen de cada uno de sus requerimientos es claro y si facilita la referenciación de cada requerimiento en el desarrollo futuro o mejora la documentación.	
¿Se ha identificado cada requerimiento con el fin de facilitar su referenciación en el futuro desarrollo o en los esfuerzos de mejora?	Si
¿Cada requerimiento posee una referencia a los requerimientos previos del proyecto que están relacionados con él?	Si

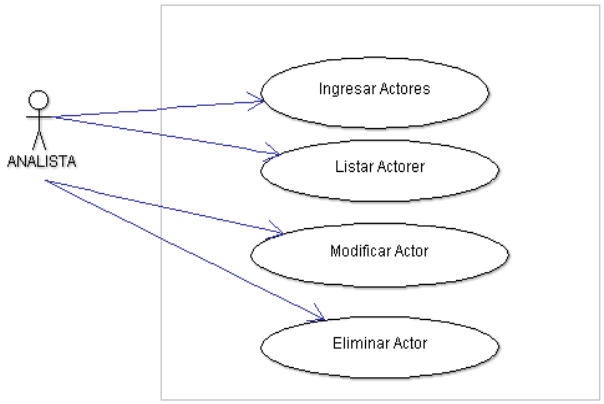
Verificación de Requerimientos.

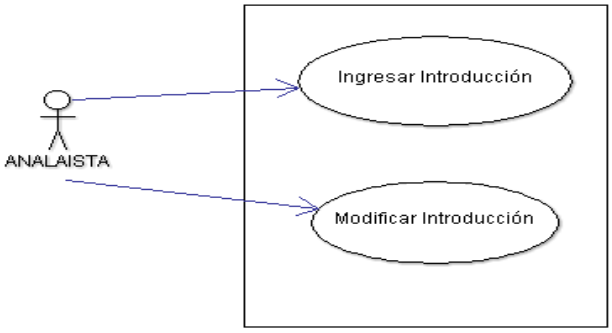
Se entregó el documento de especificación de requerimientos al Ing. Mauricio Ortiz quien en este caso cumple el rol de cliente en la propuesta que presentamos. Luego de la revisión realizada, informa que el documento entregado cumple con las necesidades de un usuario final, es así como proseguiremos con el avance de las demás etapas en el desarrollo de software.

6.2 Diseño

Presentamos solo algunos casos de uso de los muchos realizados en este proyecto ya que la mayoría cumple con una estructura parecida.

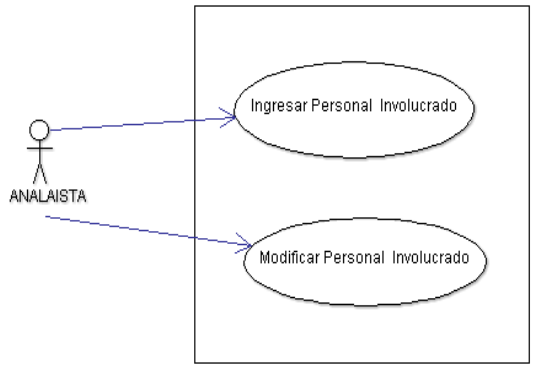
MODELO DE CASOS DE USO GESTION DE ACTORES



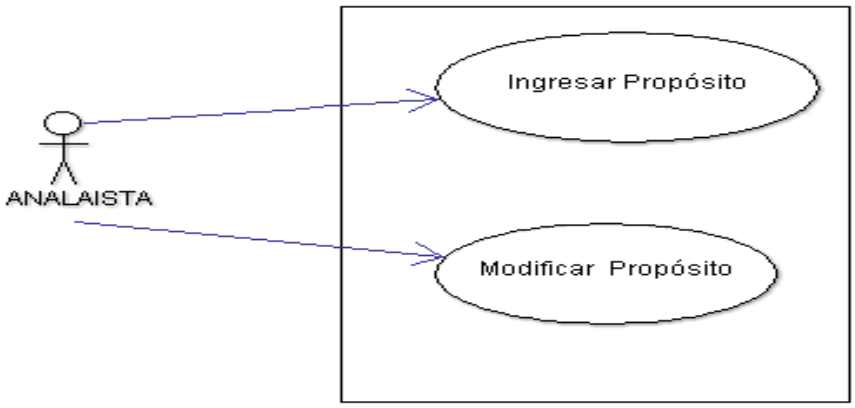


CASO DE USO GESTIÓN DE INTRODUCCIÓN

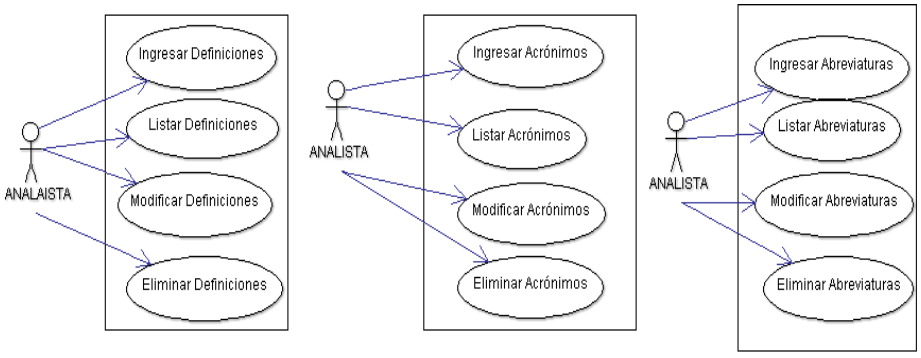
CASO DE USO PROPOSITO



CASO DE USO PERSONAL INVOLUCRADO



CASO DE USO DICCIONARIO



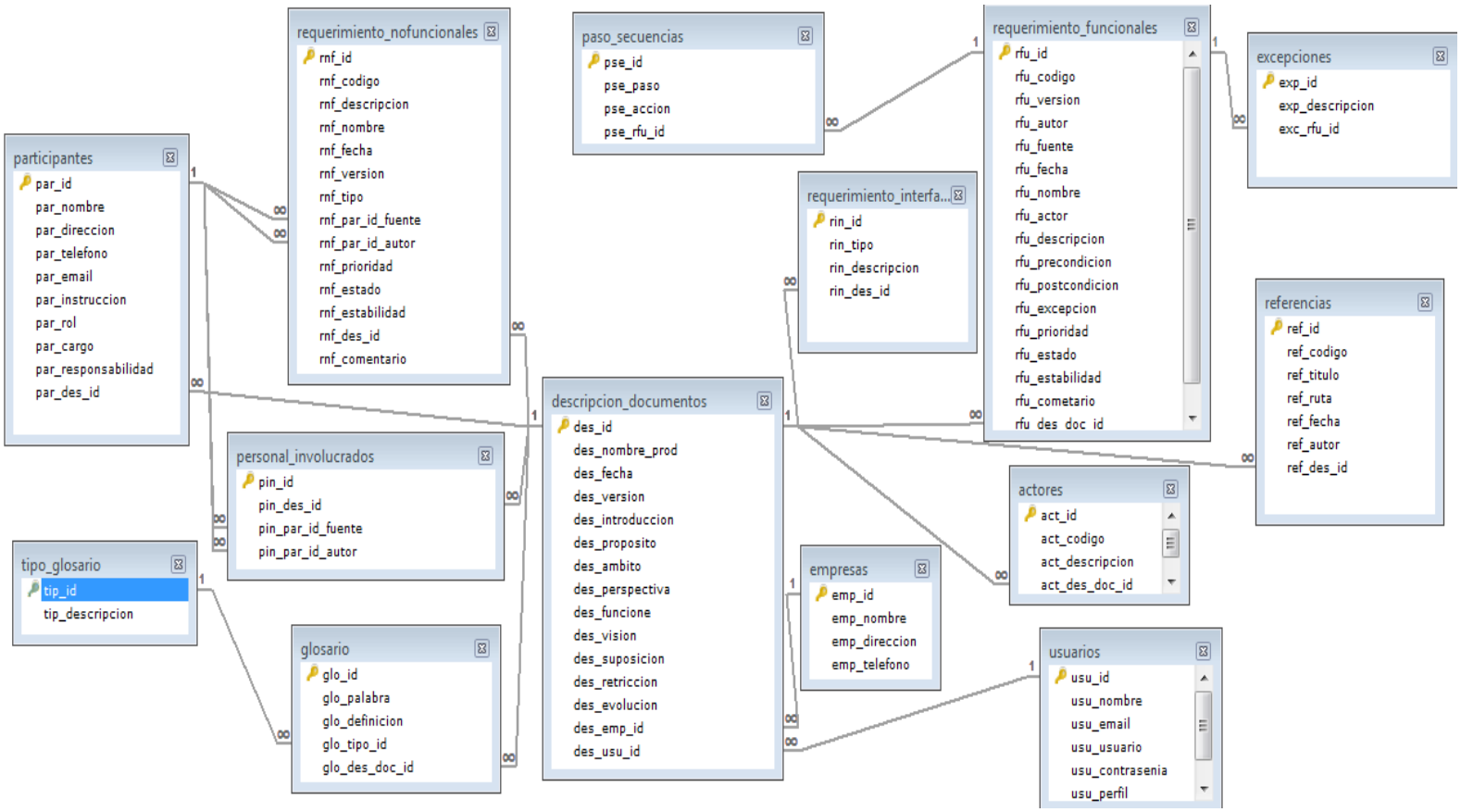


Ilustración 34: MODELO DE BASE DE DATOS

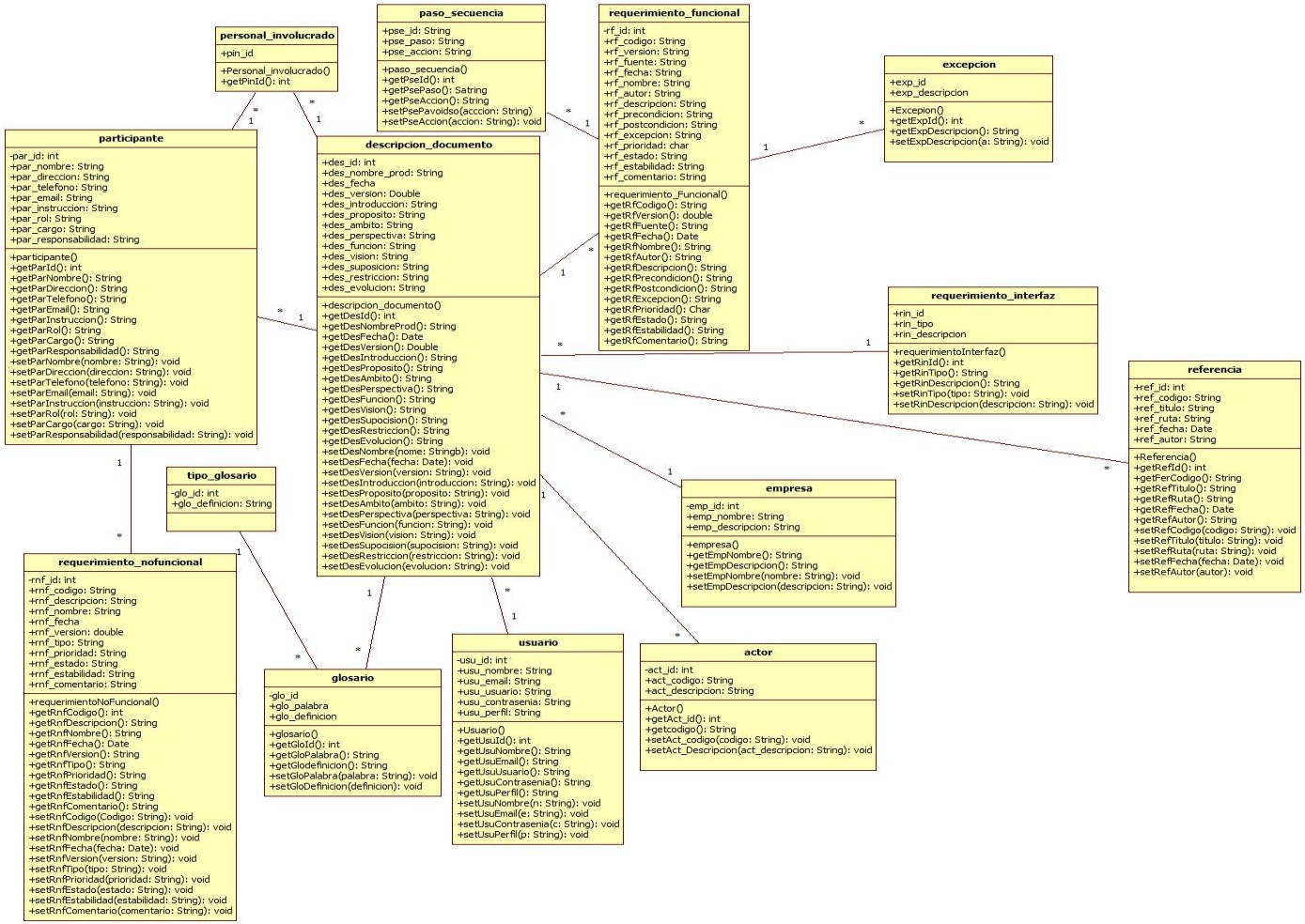
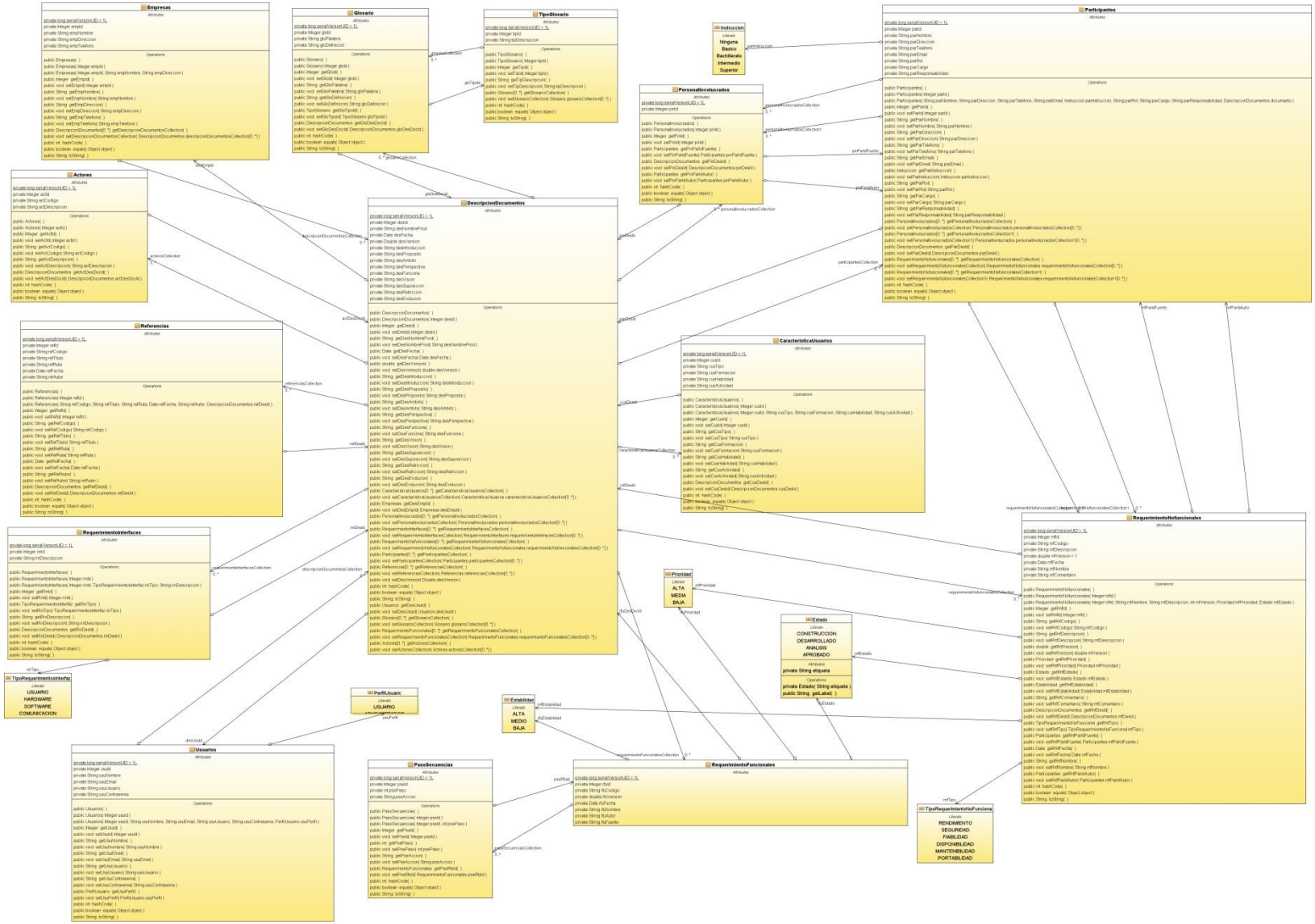
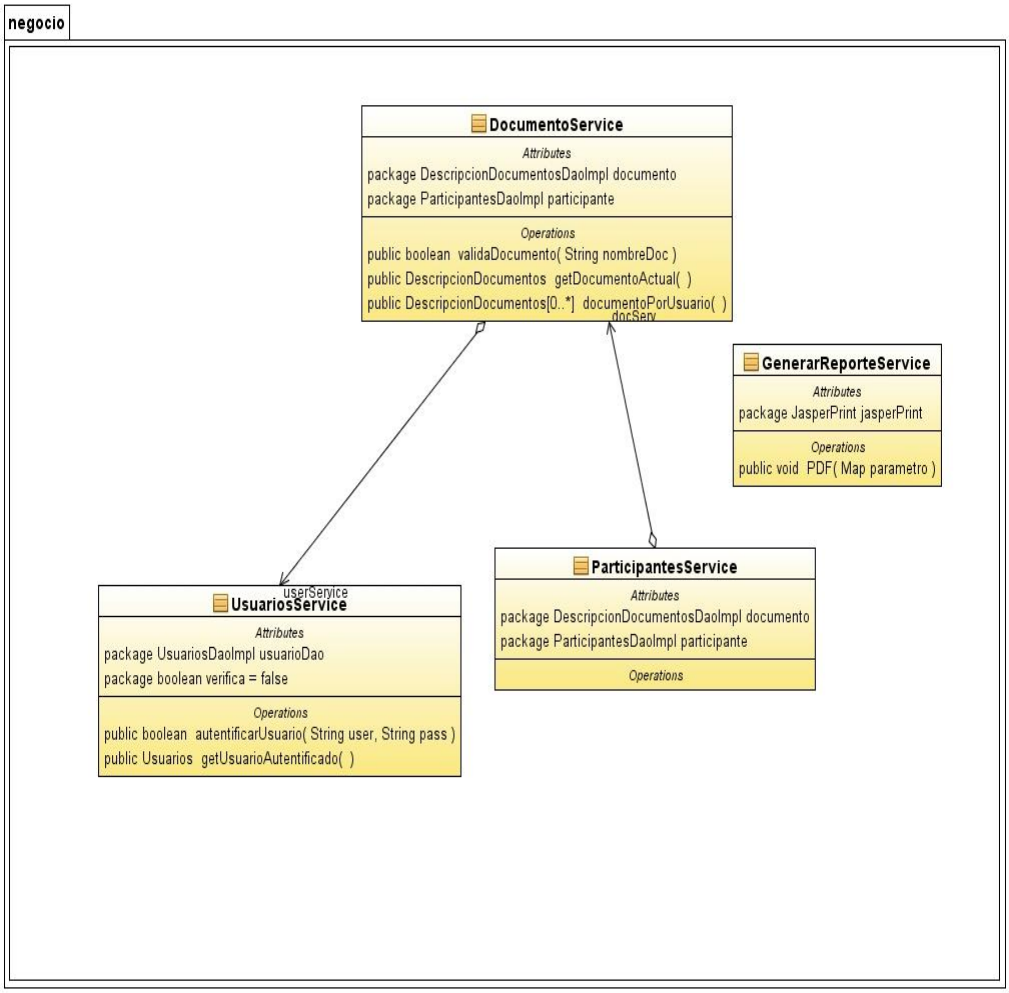


Ilustración 35: DIAGRAMA DE CLASES




negocio
DocumentoService
Attributes
package DescripcionDocumentosDaoImpl documento
package ParticipantesDaoImpl participante
Operations
public boolean validaDocumento(String nombreDoc)
public DescripcionDocumentos getDocumentoActual( )
public DescripcionDocumentos[0..*] documentoPorUsuario( )
docServ
GenerarReporteService
Attributes
package JasperPrint jasperPrint
Operations
public void PDF(Map parametro)
userService
UsuariosService
Attributes
package UsuariosDaoImpl usuarioDao
package boolean verifica = false
Operations
public boolean autentificarUsuario(String user, String pass)
public Usuarios getUsuarioAutentificado( )
ParticipantesService
Attributes
package DescripcionDocumentosDaoImpl documento
package ParticipantesDaoImpl participante
Operations

Al igual que los casos de uso presentados anteriormente en estos modelos de Diagrama de Actividad presentamos algunos como ejemplo, ya que las actividades realizadas en el sistema son similares.

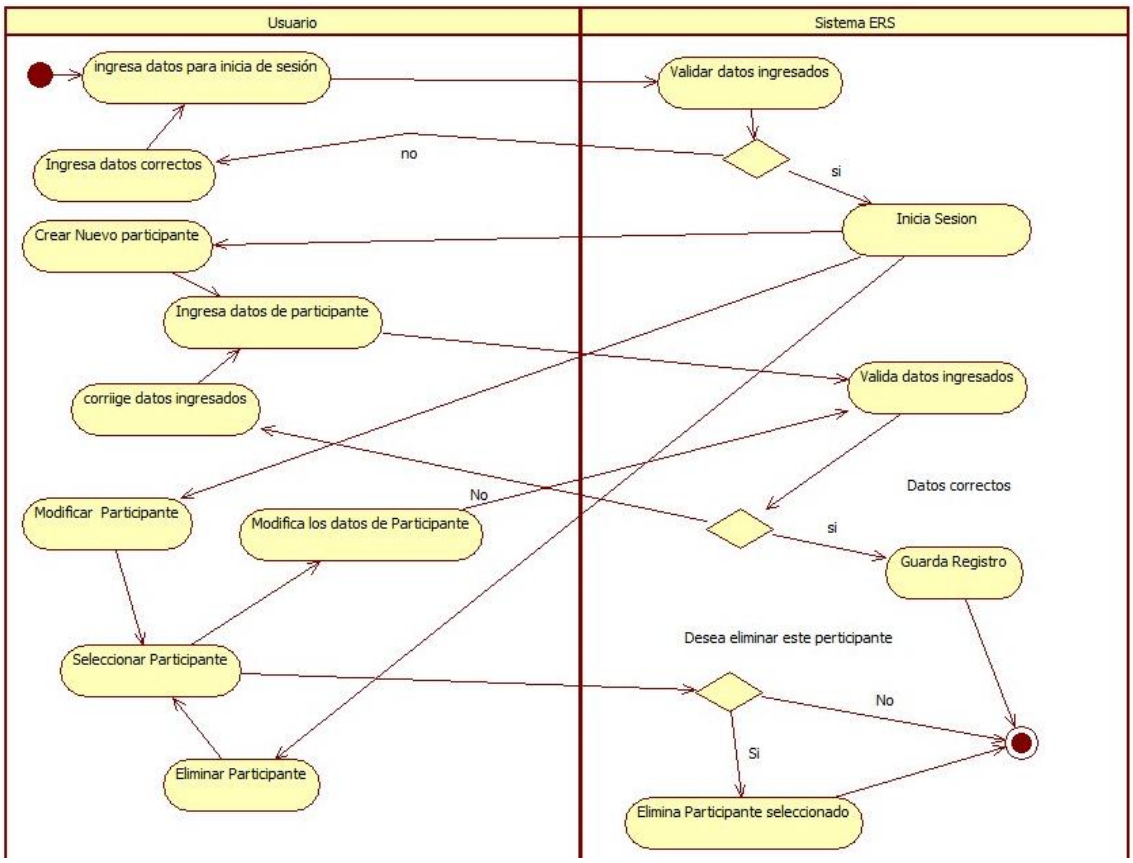


Ilustración 36: Diagrama de Actividad Gestión de Participantes

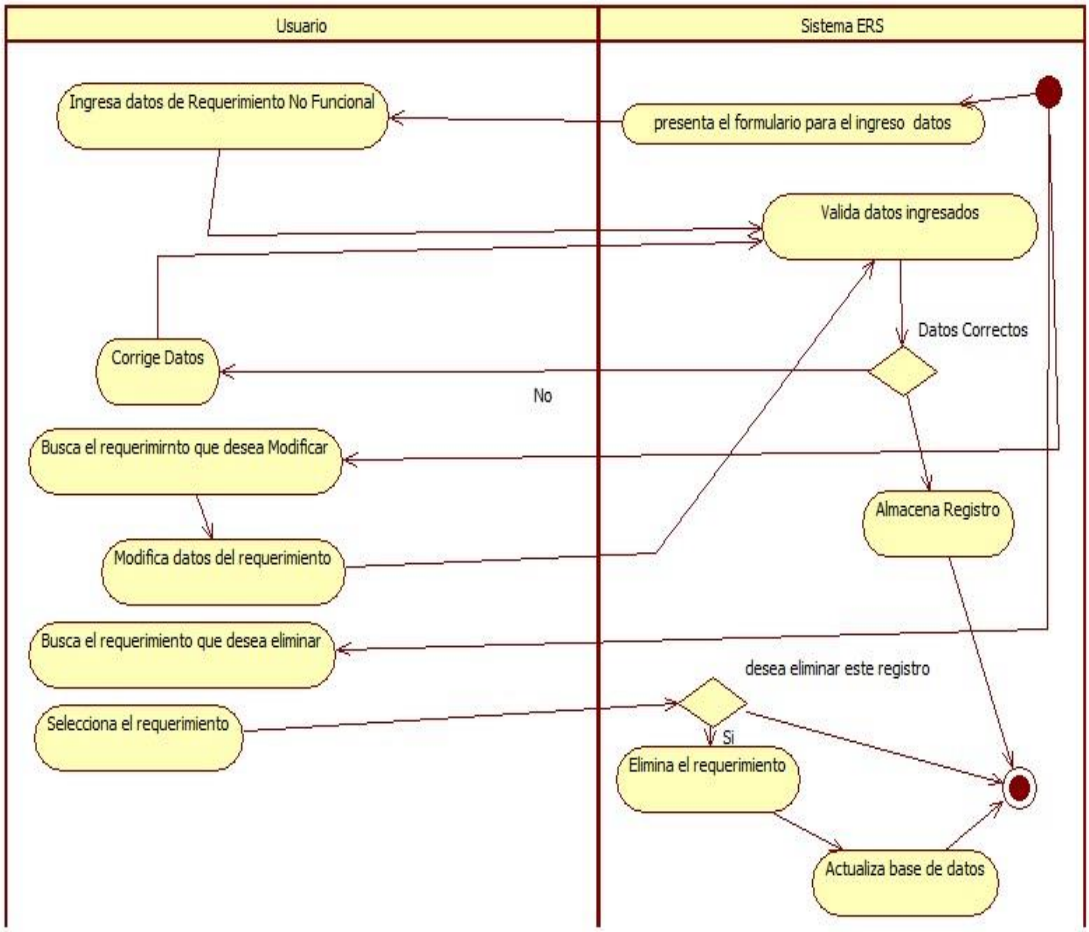


Ilustración 37: DIAGRAMMA DE ACTIVIDAD GESTION DE REQ. NO FUNCIONAL

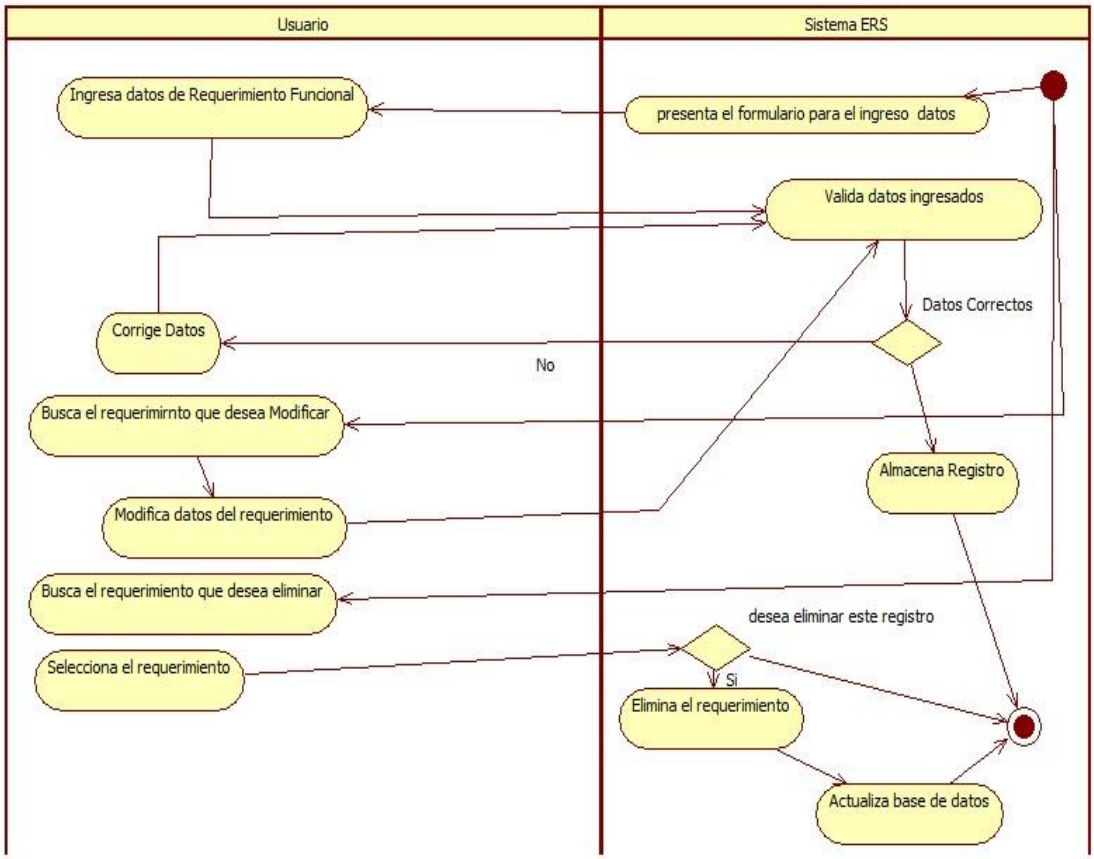


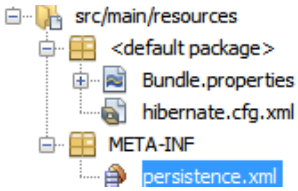
Ilustración 38: DIAGRAMA DE ACTIVIDAD GESTION DE REQ. FUNCIONALES

1.3 Implementación

En esta fase se mostraran los archivos de configuración, los paquetes y las clases implementadas para que la aplicación funcione.

Se podrá ver como se ha configurado el archivo web.xml, faces-config.xml, persistencia.xml, el pom.xml y otros.

1.3.1 ARCHIVOS DE CONFIGURACION



WEB.XML

```
<servlet-mapping>
   <servlet-name>Faces Servlet</servlet-name>
   <url-pattern>/faces/*</url-pattern>
</servlet-mapping>

<session-config>
   <session-timeout>
      30
   </session-timeout>
</session-config>


<welcome-file-list>
   <welcome-file>/faces/login.xhtml</welcome-file>
</welcome-file-list>

<filter>
<filter-name>LoginFiltro</filter-name>
<filter-class>com.systoers.filtro.LoginFiltro</filter-class>
</filter>
```

FACES-CONFIG.XHTML

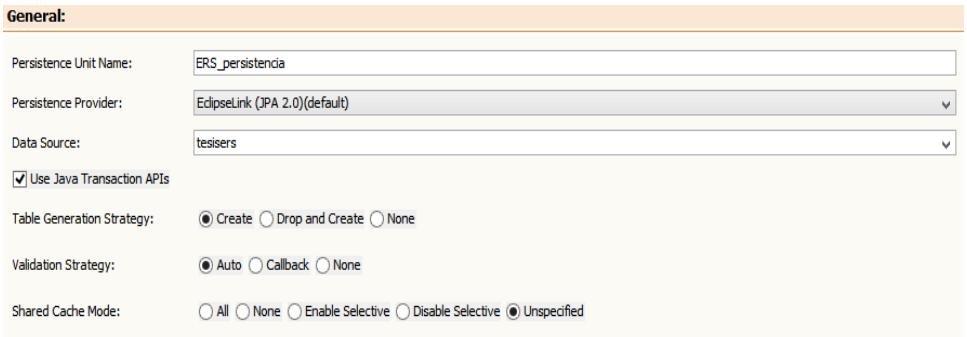
En este archivo se configuro la navegación en las páginas de la aplicación.

Regla de Navegación para la Pagina Login

```
<navigation-rule>
   <from-view-id>/login.xhtml</from-view-id>
   <navigation-case>

      <from-outcome>exito.login</from-outcome>
      <to-view-id>/pages/buscar.xhtml</to-view-id>
      <redirect>/pages/buscar.xhtml</redirect>
   </navigation-case>

   <navigation-case>
      <from-outcome>fallo.login</from-outcome>
      <to-view-id>/login.xhtml</to-view-id>
   </navigation-case>

</navigation-rule>
```

Persistence.xml

En donde se configuro el acceso a la base de datos.



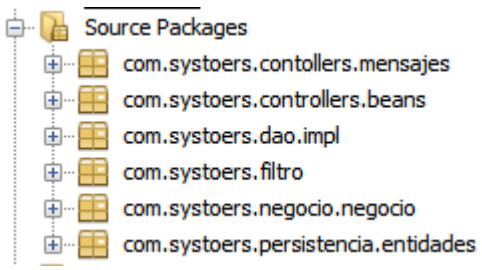
Pom.xml

Con el cual se accedió a los paquetes necesarios para la aplicación.

```
    <dependencies>

            <dependency>
             <groupId>org.primefaces</groupId>
             <artifactId>primefaces</artifactId>
             <version>3.5</version>
          </dependency>

 <dependency>
        <groupId>net.sf.jasperreports</groupId>
        <artifactId>jasperreports</artifactId>
        <version>5.1.0</version>
</dependency>
```

1.3.2 PAQUETES USADOS

Se está usando el Modelo Vista Controlador para esto se tiene los paquetes de persistencia que es donde esta las tablas mapeadas, las clases Dao que son los que interactúan con la base de datos y los Controles que son los que interactúan con la interfaz.



6.4 Pruebas

Pruebas del Sistema.

Las pruebas que realizadas están orientadas al rendimiento para esto se ha usado la herramienta JMeter[4], esta herramienta ayudara a ejecutar pruebas con un gran volumen de usuarios, simulando escenarios con gran cantidad de peticiones (F. Javier Diaz).

En el artículo realizado por F. Javier Díaz, Claudia M. Tzancoff Banchoff, Anahí S. Rodríguez y Valeria Soria, indican que según el autor Jakob Nielsen, en el libro "Usability Enginnering" (Morgan Kaufmann) existen tres límites en el tiempo de respuesta.

- 0,1 Segundo: es el límite en el cual el usuario siente que está manipulando los objetos desde la interfaz del usuario.
- 1 segundo: es el límite en el cual el usuario siente que está navegando libremente sin esperar demasiado una respuesta del servidor.
- 10 segundos: es el límite en el cual se pierde la atención del usuario, si la respuesta tarda más de 10 segundos deberá indicar algún mecanismo por el cual el usuario pueda interrumpir la operación.

A continuación se describe la herramienta utilizada y las pruebas que se realizaran con la misma.

Jmeter

[4] Jmeter es una de las herramientas de código abierto más extendidas para realizar pruebas de rendimiento en varios protocolos, como JDBC, FTP, LDAP, JMS y, el más utilizado, HTTP

Para analizar el tiempo de respuesta del servidor se utilizó la herramienta Jmeter en la versión 2.6. Jmeter es una herramienta open source muy completa, implementada en Java que permite realizar pruebas de comportamiento funcional y medir rendimiento. Para estas pruebas se configuro un servidor proxy que provee Jmeter, para construir un camino aleatorio de navegación aleatorio, y así simular la visita de un usuario.

Para evaluar los resultados se utilizaran 3 componentes provistos por la herramienta.

- Summary Report: permite visualizar los resultados de la prueba en una tabla con la información que describimos a continuación:

    - *Label:* etiqueta de la muestra.
    - *#Muestras*: cantidad de thread utilizados para la URL.
    - *Media:* tiempo promedio en milisegundos para un conjunto de resultados.
    - *Min:* tiempo mínimo que demora un thread en acceder a una página.
    - *Max:* tiempo máximo que demora un thread en acceder a una página.
    - Rendimiento: rendimiento medido en los requerimientos por segundo/minuto/hora.
    - *Kb/sec:* rendimiento medido en Kbytes por segundo.
    - *Media en* bytes: tamaño medido de respuesta del servidor en bytes.

- *Agreggate Graph:* componente similar a la anterior, pero permite obtener resultados más precisos. Utiliza más memoria, ya que calcula la mediana y la línea al 90%, la cual requieren que todos los datos estén almacenados. Los datos que se presentan es este componente son:

    - *URL:* etiqueta de la muestra.
    - *#Muestras:* cantidad de thread utilizados para la URL.
    - *Media:* tiempo promedio en milisegundos para un conjunto de resultados.
    - *Mediana:* tiempo promedio del percentil 50.
    - *Línea del 90%:* máximo tiempo utilizado por el 90% de la muestra.
    - *Min:* tiempo mínimo de la muestra de una determinada URL.
    - *Max:* tiempo máximo de la muestra de una determinada URL.

- *%Error:* porcentaje de requerimientos con errores.
- *Rendimiento:* rendimiento medio en los requerimientos por segundo/minuto/hora.
- *KB/sec:* rendimiento medido en Kbytes por segundo.

- *Gráfico de resultados:* este componente permite visualizar gráficamente los siguientes datos: media, mediana, dispersión y el rendimiento.

La prueba que se ha realizado consistió en definir 3 test 100,50 y 25 threads los cuales simulan 100,50 y 25 accesos de usuarios respectivamente.

Se analizaron los resultados a través de un intervalo de confianza[5] con un nivel de confianza al 95%[6].

Para un primer análisis de 25 usuarios, se supone que la población tiene una distribución Normal, para el análisis con 50 y 100 usuarios no se requiere hacer la suposición, ya que por el *Teorema Central del Límite (TCL), "para n grande implica que X tiene una distribución aproximadamente Normal sin importar la naturaleza de la distribución poblacional"* (Devore).

[5] Intervalo de confianza: intervalos aleatorios que se usan para acotar un valor con una determinada probabilidad alta. Por ejemplo, puedo calcular un intervalo de confianza para la media, o la distribución; pero nunca puedo llegar a tener un valor exacto con total seguridad.
Es decir, es un espacio en el que puedo afirmar que probablemente se halle ese valor.

[6] Un nivel de confianza del 95% lleva a un valor Z de 1,96.

Explicación detallada de los resultados.

Primera prueba.

La primera prueba se realizó el día viernes 06 de Septiembre a las 14:30 pm. Se configuraron 25 *threads* cada "*1 seg*". Los resultados totales obtenidos por el componente "Aggregate Graph" y "Summary Report" se muestran en las siguientes ilustraciones.

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
HTTP Request	375	2	1	3	0	38	0,00%	365,1/sec	181,1
Total	375	2	1	3	0	38	0,00%	365,1/sec	181,1

Ilustración 39 resultados de la prueba con 25 usuarios en el componente *"Aggegate Graph"*

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Bytes
HTTP Request	375	2	0	38	3,68	0,00%	365,1/sec	181,12	507,9
Total	375	2	0	38	3,68	0,00%	365,1/sec	181,12	507,9

Ilustración 40: resultado de la prueba con 25 usuario en el componente Summer Report

Como puede verse el tiempo promedio para acceder a una página es 2 segundos, realizándose un total de 375 requerimientos al servidor.

El tiempo total utilizado para los 25 usuarios se puede calcular con la siguiente formula:

Tiempo Total= (#Muestras)(Media)= (375) (2) = 750 milisegundos o 0,75 segundos.

El tiempo promedio total requerido por cada usuario, se puede calcular de la siguiente manera:

TP = Tiempo Total / Cantidad Usuarios=750/25=30 milisegundos o 0.03 segundos

Análisis realizado

Se han evaluado los resultados obtenidos a través de un intervalo de confianza para una distribución normal al 95% con la siguiente Formula:

$$[TP - Z_{0.95} * \frac{D}{\sqrt{n}}, TP + Z_{0.95} * \frac{D}{\sqrt{n}}]$$

Dónde:

- Tiempo promedio (TP) de respuesta es : 30
- Desviación (D) es: 3,68.
- Tamaño de la muestra (n) es de: 375
- $Z_{0.95}$ es: 1.96.

El intervalo resultante es el siguiente:

$$[(30 - 1.96)\frac{3,68}{\sqrt{375}}, (30 + 1.96)\frac{3,68}{\sqrt{375}}]$$

$$[29,6389; 30,3610]\ en\ milisegundos$$

Por lo tanto, se puede observar que el tiempo de respuesta promedio esté entre 29,6389 y 30,3610 milisegundos. Para la cantidad de 25 usuarios simultáneos realizando 375 solicitudes.

Segunda Prueba

La segunda prueba se realizó el día lunes 9 de septiembre a las 14: 30 pm. Se configuraron 50 *treads* cada "*1 seg*". Los resultados totales obtenidos por el componente "Aggegate Graph" y "Summary Report"se muestran en la siguientes ilustraciónes.

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
HTTP Request	750	2	1	5	0	156	0,00%	708,2/sec	351,3
Total	750	2	1	5	0	156	0,00%	708,2/sec	351,3

Ilustración 41: resultados de la prueba con 50 usuarios en el componente "Aggegate Graph"

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Bytes
HTTP Request	750	2	0	156	7,60	0,00%	708,2/sec	351,30	507,9
Total	750	2	0	156	7,60	0,00%	708,2/sec	351,30	507,9

Ilustración 42 : resultados de la prueba con 50 usuarios en el componente "Summer Report"

Como puede verse el tiempo promedio para acceder a una página es 2 segundos, realizándose un total de 750 requerimientos al servidor.

El tiempo total utilizado para los 50 usuarios se puede calcular con la siguiente formula:

Tiempo Total= (#Muestras)(Media)= (750) (2) = 1500 milisegundos o 1,5 segundos.

El tiempo promedio total requerido por cada usuario, se puede calcular de la siguiente manera:

Tiempo Total / Cantidad Usuarios=1500/50=30 milisegundos

Análisis realizado

Se evaluaron los resultados obtenidos a través de in intervalo de confianza para una distribución normal al 95% con la siguiente Formula:

$$[TP - Z_{0.95} * \frac{D}{\sqrt{n}}, TP + Z_{0.95} * \frac{D}{\sqrt{n}}]$$

Dónde:

- Tiempo promedio (TP) de respuesta es : 1500
- Desviación (D) es: 7.6.
- Tamaño de la muestra (n) es de: 750
- $Z_{0.95}$ es: 1.96.

El intervalo resultante es el siguiente:

$$[(1500 - 1.96)\frac{7,6}{\sqrt{750}}, (1500 + 1.96)\frac{7,6}{\sqrt{750}}]$$

$$[(29,4727); (30,5272)]\ en\ milisegundos$$

Por lo tanto, se puede observar que el tiempo de respuesta promedio esté entre 29,4727 y 30.5272 milisegundos. Para la cantidad de 50 usuarios simultáneos realizando 750 solicitudes.

Tercera prueba.

La primera prueba se realizó el día lunes 16 de Septiembre a las 14:30 pm. Se configuraron 100 *treads* cada "*1 seg*". Los resultados totales obtenidos por el componente "Aggregate Graph" y "Summary Report" se muestran en las siguientes ilustraciones.

Etiqueta	# Muestras	Media	Mediana	Linea de 90%	Mín	Máx	% Error	Rendimiento	Kb/sec
HTTP Request	1125	44	6	107	0	704	0,00%	641,4/sec	318,2
Total	1125	44	6	107	0	704	0,00%	641,4/sec	318,2

Ilustración 43 resultados de la prueba con 25 usuarios en el componente *"Aggegate Graph"*

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Media de Bytes
HTTP Request	1125	44	0	704	99,62	0,00%	641,4/sec	318,22	508,1
Total	1125	44	0	704	99,62	0,00%	641,4/sec	318,22	508,1

Ilustración 44: resultado de la prueba con 25 usuario en el componente Summer Report

Como puede verse el tiempo promedio para acceder a una página es 44 segundos, realizándose un total de 750 requerimientos al servidor.

El tiempo total utilizado para los 75 usuarios se puede calcular con la siguiente formula:

Tiempo Total= (#Muestras)(Media)= (750) (44) = 33000 *milisegundos*

El tiempo promedio total requerido por cada usuario, se puede calcular de la siguiente manera:

TP = Tiempo Total / Cantidad Usuarios=3300/75=400 milisegundos

Análisis realizado

Se evaluaron los resultados obtenidos a través de un intervalo de confianza para una distribución normal al 95% con la siguiente Formula:

$$[TP - Z_{0.95} * \frac{D}{\sqrt{n}}, TP + Z_{0.95} * \frac{D}{\sqrt{n}}]$$

Dónde:

- Tiempo promedio (TP) de respuesta es : 400
- Desviación (D) es: 78,57.

- Tamaño de la muestra (n) es de: 750
- $Z_{0.95}$ es: 1.96.

El intervalo resultante es el siguiente:

$$[(400 - 1.96)\frac{78.57}{\sqrt{750}}, (400 + 1.96)\frac{78,57}{\sqrt{375750}}]$$

$$[434,5489; 445,4510]\ en\ milisegundos$$

Por lo tanto, se puede observar que el tiempo de respuesta promedio esté entre 0,43 y 0,44 segundos. Para la cantidad de 25 usuarios simultáneos realizando 375 solicitudes.

Gráficos Obtenidos

Las figuras siguientes muestran los datos obtenidos para la primera, segunda y tercera prueba respectivamente.

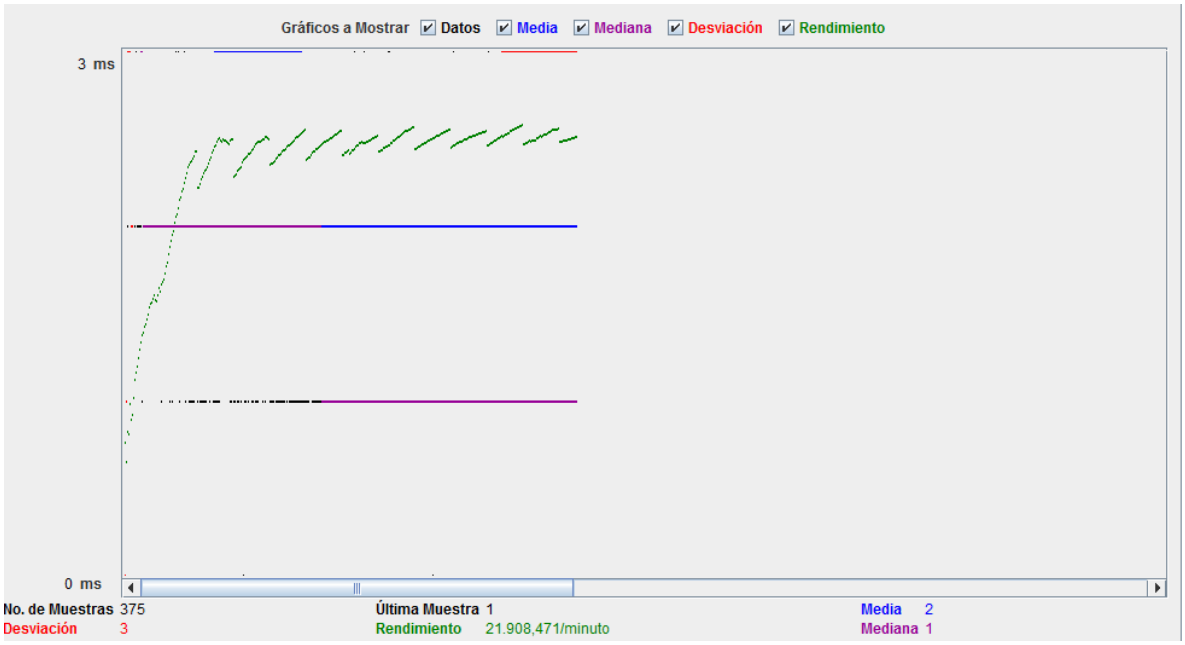


Figura 2: Resultado de Jmeter simulando 25 usuarios

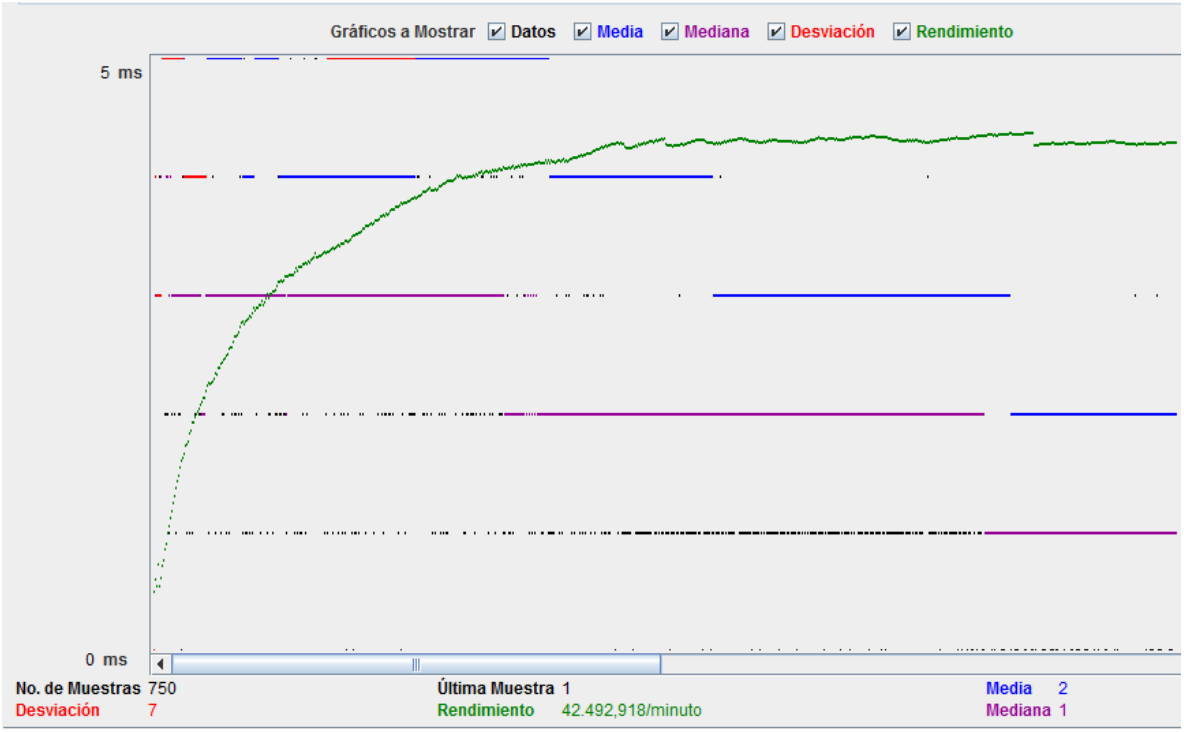


Figura 3: Simulando 50 usuarios

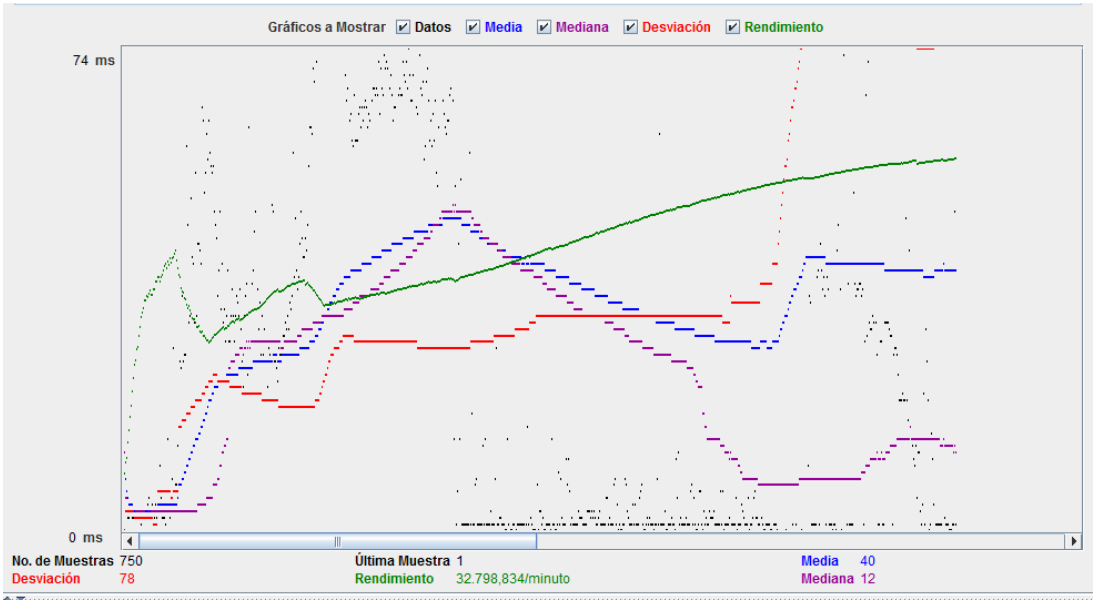


Figura 4 prueba 3 para solicitudes de 75 usuarios

6.5 Manuales y Documentación

Para ver el manual de usuario consultar en **Anexo B**

CAPITULO VII

Conclusiones y recomendaciones

Para concluir con este trabajo de tesis, este capítulo se dedicara a mostrar las conclusiones y las recomendaciones obtenidas a lo largo de este proyecto. Esto se realizara con el fin de que se pueda dar continuidad al proyecto así como mostrar los beneficios y resultados obtenidos.

6.1 Conclusiones

En esta tesis se ha creado una metodología para la especificación de requerimientos de software basándonos en el estándar IEEE 830-1998, metodología que ayudara al Ingeniero desarrollador de software a producir aplicaciones que cumplan adecuadamente con las necesidades de los diferentes clientes que requieran de un sistema para su negocio u empresa.

También conseguimos implementar un prototipo (SysToErs) para almacenar la información del documento de especificación, la herramienta creada nos permite realizar la gestión de los datos relacionados con la especificación realizada de acuerdo a la metodología propuesta.

La metodología propuesta en esta tesis se basa en las plantillas utilizadas por el Doctor Amador Duran Toro en su tesis Doctoral “Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información”, estas plantillas nos permitieron describir los

requerimientos funcionales y no funcionales para el sistema, estos recursos colaboran a que el documento sea comprensible tanto para el cliente como para el desarrollador del sistema.

Después del estudio realizado se ha podido constatar que la ingeniería de requerimientos es una rama no usada e incluso desconocida para varios desarrolladores, quienes buscan la implementación de software de calidad sin considerar lo que las necesidades de los clientes y usuarios finales, por lo que los desarrolladores invierten tiempo y costos innecesarios para el desarrollo del producto, es por esta razón que la ingeniería de requerimientos cumple un papel muy importante en el proceso de producción de software, ya que se enfoca en identificar las necesidades y en generar una especificación de requerimientos consistente, compacta y sin ambigüedades con el fin de minimiza problemas y evitar que los proyectos fracasen debido a una mala gestión de los requerimientos.

1.1 Recomendaciones

Las recomendaciones que surgen con base en el presente estudio son las siguientes:

En este documento, en caso de que a alguien le interesa nuestra tesis es que lo complemente incluyendo más funcionalidad al prototipo de manera que al presentar el reporte del documento; este presente información relacionada con los diagramas de casos de uso ya que el prototipo realizado solo presenta información en una plantilla que se toma de la tesis de doctorado de Amador Duran Toro.

En cuanto a la metodología creada en esta tesis en la etapa de Verificación del Documento se basa principalmente en la experiencia del analista, por lo que un estudio más profundo en esta fase de la ingeniería de requerimientos ayudaría a legalizar completamente este documento.

Anexos

Anexo A. Encuesta

UNIVERSIDAD POLITÉCNICA SALESIANA
FACULTAD DE INGENIERIA DE SISTEMAS
SEDE CUENCA

Esta encuesta tiene como objetivo, determinar el estado en el que se encuentra la Ingeniera de Requerimientos, en las empresas desarrolladoras de software en la ciudad de Cuenca. Los resultados de esta encuesta serán analizados con absoluta reserva y servirán para el desarrollo de una tesis.
Responda con toda sinceridad las preguntas que se plantean a continuación

Señale con una "X" en el lugar que corresponda.

1. **Cree que el software que desarrolla cumple con las necesidades de los usuarios finales**

Sí ☐ No ☐

2. **¿Qué fases de la ingeniería de requerimientos cubre para el desarrollo de software?**

Elicitación	☐
Análisis	☐
Especificación	☐
Validación	☐
Ninguna	☐

3. **¿Cree Ud. Que la Ingeniería de requerimientos es importante?**

Sí ☐ No ☐

¿Por qué?……..

4. **Usa algún tipo de metodología para la ingeniería de requerimientos**

Sí ☐ No ☐

Mencione cual utiliza ….……

5. **Utiliza alguna técnica y/o herramienta para la elicitación de requerimientos.**

Sí ☐ No ☐

¿Qué técnicas? ……

6. **En la fase de análisis de requerimientos usa alguna técnica**

Sí ☐ No ☐

¿Qué técnicas?.....……

7. **¿Para la elaboración del documento de Especificación de requerimientos usa algún tipo de estándar?**

Sí ☐ No ☐

¿Cuál?……

8. **En la fase de validación/verificación de requerimientos usa alguna técnica**

Sí ☐ No ☐

¿Cuál?……..

Anexo B. Manual de Usuario

Manual de Usuario

SISTEMA PARA ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

SysToERS

INGRESO AL SISTEMA

Para ingresar al sistema se debe ingresar un usuario y contraseña válidos, estos deben estar previamente agregados en la base de datos

Pantalla Principal

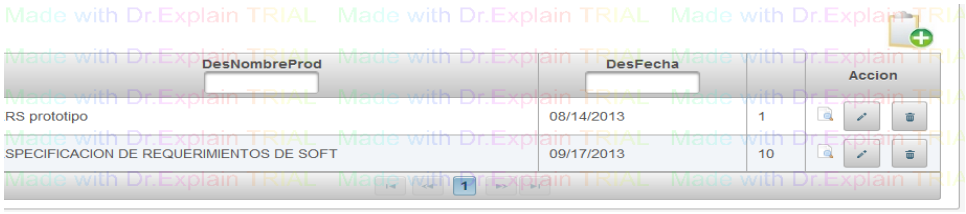
Al ingresar el usuario y contraseña correctos se podrá observar la pantalla principal del proyecto la cual contendrá todos los documentos que han sido creados. Se podrá Modificar y Eliminar los Documentos existentes así como también se podrá realizar la vista previa del documento en formato PDF. En esta ventana también se muestra un icono para agregar un nuevo documento

Menú de Opciones

- En este menú se podrá agregar un nuevo documento ERS,
- En caso de estar en una ventana diferente a la principal se podrá regresar a la misma para así administrar los documentos que han sido creados con anterioridad.

Panel en cual se Ve todos los Documentos existentes por nombre y Fecha de Creación

En este panel se encuentra una lista de los documentos creados por el usuario, también se tiene dos métodos de búsqueda para los documentos por Nombre del mismo y por su fecha de creación en la imagen se puede apreciar las diferentes opciones que se tiene tales como: Modificar, Eliminar y Vista Previa



Modificar Documentos

Para modificar los documentos existentes se usa el siguiente botón ubicado en la tabla anterior, al dar clic aparecerá el documento para modificar de la misma forma en la que se ve cuando se agrega.

Botón para eliminar documentos

Este botón permitirá eliminar los documentos que no se necesiten. Que se puede ver en la imagen

Vista Previa en PDF

Con este botón se podrá obtener una vista previa del documento con formato PDF que se puede ver en la imagen

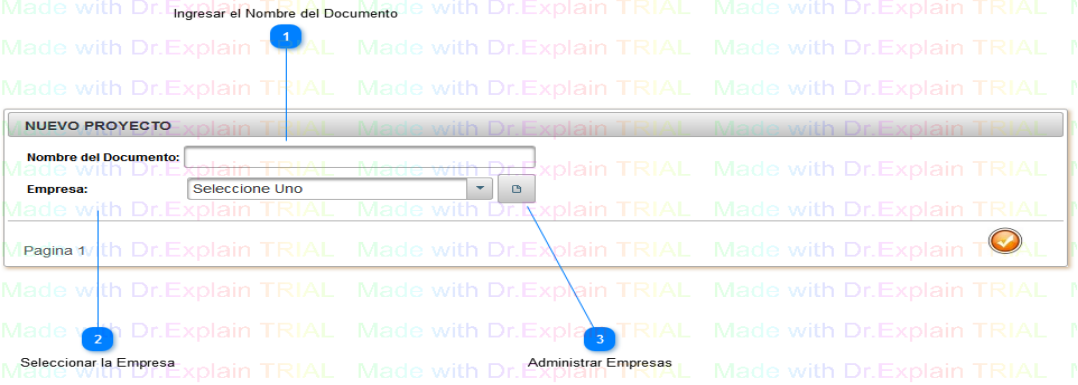
Para salir des sistema se usa el link

el cual está ubicado en la parte superior derecha de la pantalla.

AGREGAR UN DOCUMENTO

Para crear un nuevo documento se debe dar clic en el botón Agregar Documento , este llevara directamente a llenar los datos principales del documento.

3

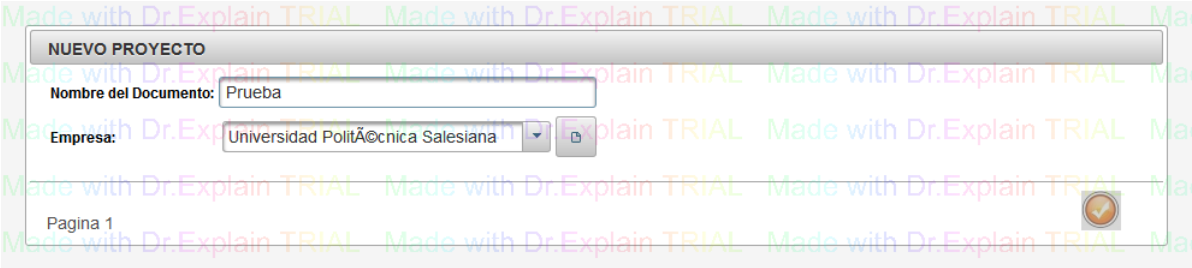


Administrar Empresas

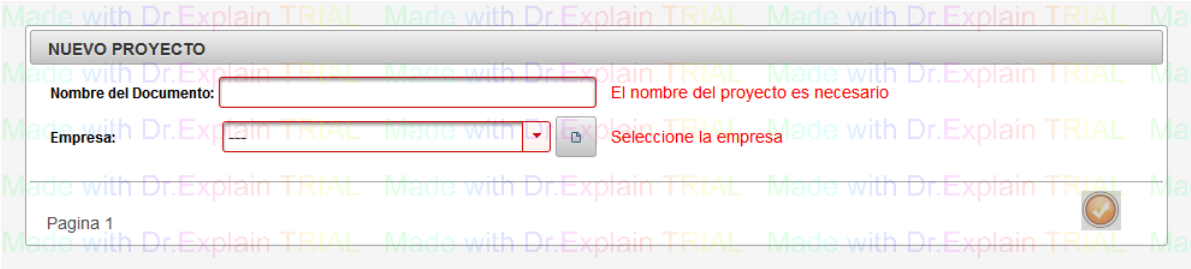
Este botón permitirá la administración de los Datos de las Empresas existentes, y si por lo contrario no existiera la empresa que se necesita, permitirá al usuario crear una empresa nueva. Se aparecerá la siguiente ventana

En la que se deberá ingresar los datos respectivos como el nombre de la empresa, la dirección y teléfono al ingresar esto dar clic en guardar.

NOMBRE DEL PROYECTO



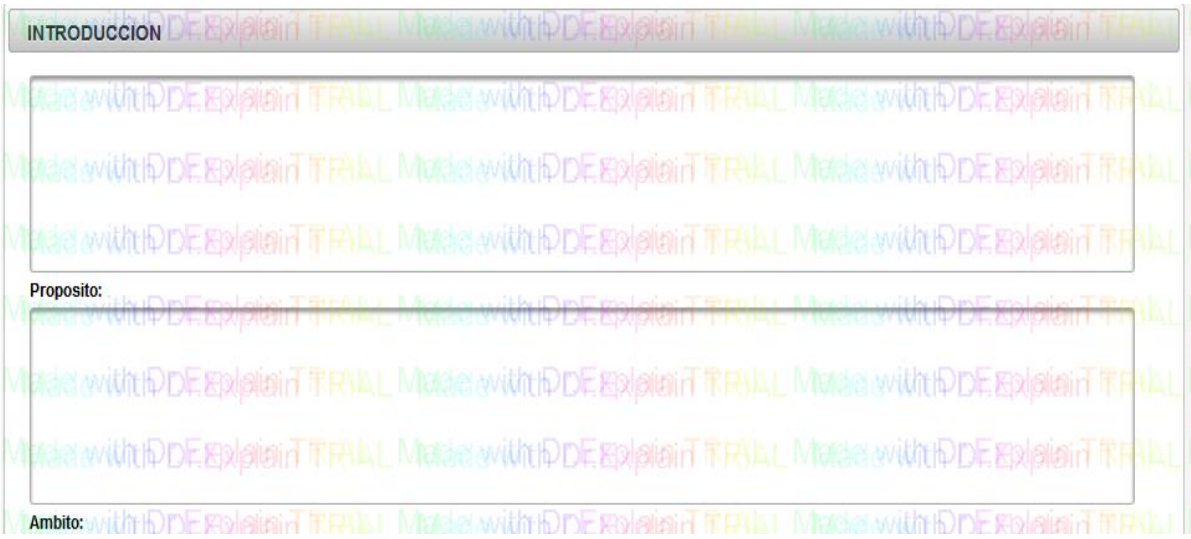
Si los campos estuvieran vacíos al tratar de pasar al siguiente campo mostrara unos mensajes de error y no se podrá continuar con el siguiente paso:



AGREGAR INTRODUCCION Y DESCRIPCION GENERAL

AGREGAR INTRODUCCION

Al estar basada en el estándar IEEE 830 de 1998 se tiene que hacer constar todos los datos para que cumpla con esto. Es por esto que se debe ingresar una Introducción como mínimo 500 palabras, Propósito, Ámbito etc. esto se aparecerá en la siguiente ventana:



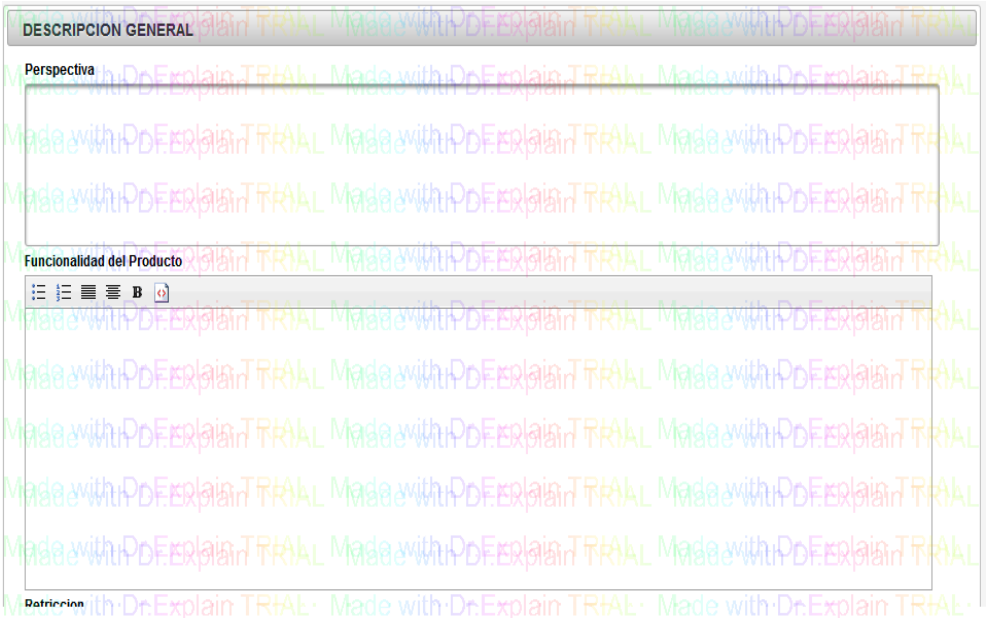
En esta ventana se tiene explícitamente en cual campo se debe ingresar la información requerida para presentar el documento de Especificación de Requerimientos.

Para pasar al siguiente punto se presiona y se pasa a la descripción General del Documento

AGREGAR DESCRIPCION GENERAL

Para añadir la Descripción General basta con llenar los campos de Perspectiva, Funcionalidad del Producto, Restricciones, Suposiciones y Evolución, todos estos campos son de tipo texto y abarcan gran cantidad de palabras, en la siguiente imagen se

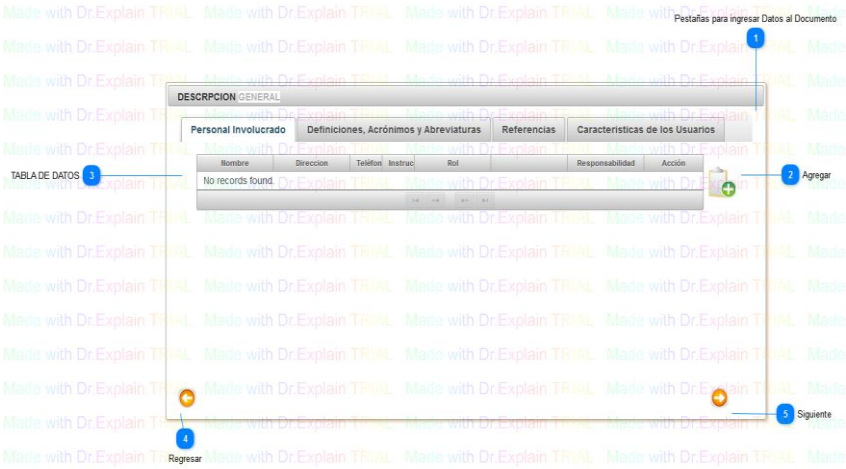
puede ver la parte superior de la ventana descripción general.



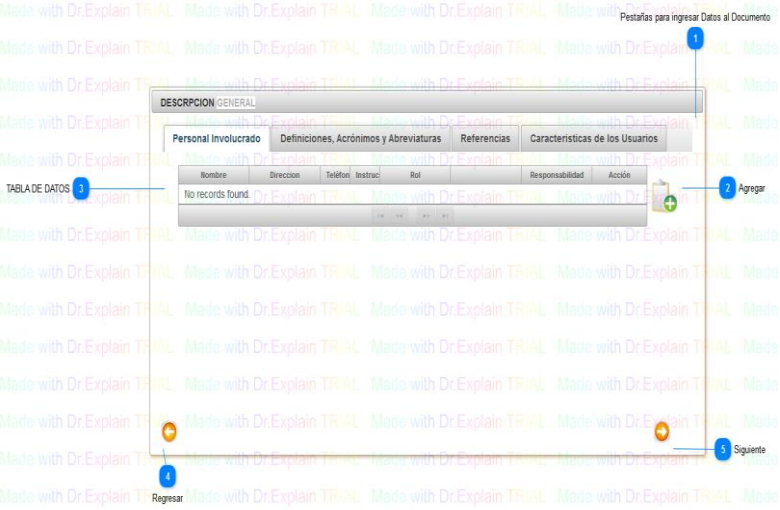
Al igual que en el punto anterior para continuar agregando información al documento actual se usa el botón siguiente , pasando de esta forma a la ventana Datos Generales del Documento.

AGREGAR DATOS GENERALES DEL PROYECTO

En esta ventana contiene 4 pestañas que al dar clic a cualquiera de estas se podrá agregar al Personal Involucrado del Proyecto, las Características de los Usuarios, Referencias del Proyecto y Definiciones, Acrónimos y Abreviaturas respectivamente.

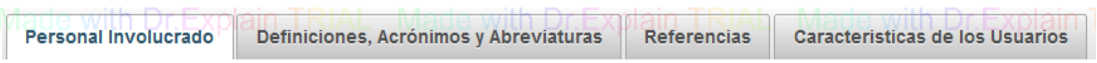


AGREGAR PERSONAL INVOLUCRADO, REFERENCIAS, CARACTERISTICAS USUARIOS Y DEFINICIONES



Pestañas para ingresar Datos al Documento

Cuatro pestañas en las que se podrá ingresar datos para el documento que se crea. Datos como personal involucrado, referencias, características de usuarios

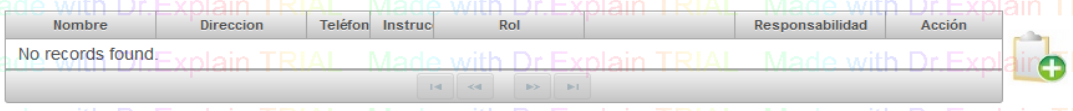


Agregar

Este botón está presente en todas las pestañas, sirve para abrir las ventanas en donde se podrá agregar un nuevo registro, dependiendo en cual pestaña se encuentre.

TABLA DE DATOS

Esta tabla visualiza la información que se va agregando, también depende de la pestaña en la que esté situado.

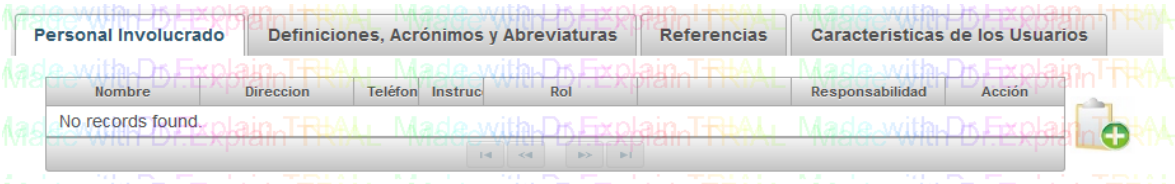


AGREGAR DATOS AL DOCUMENTO

Para agregar los datos generales se navega por las cuatro pestañas de la ventana.

- **Personal Involucrado:**

Clic en la pestaña Personal Involucrado se aparecerá la tabla con los participantes o personal involucrado ya agregados al documento.

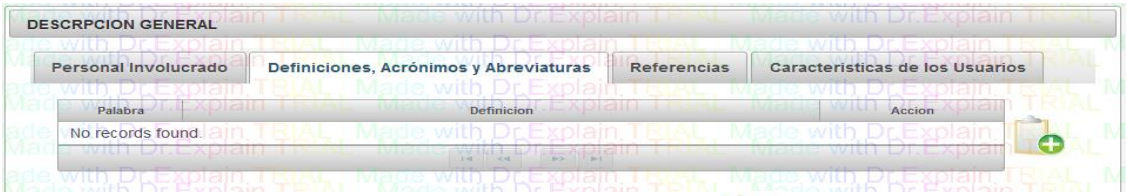


Para agregar un nuevo participante al documento luego de dar clic en se aparecerá la siguiente ventana:

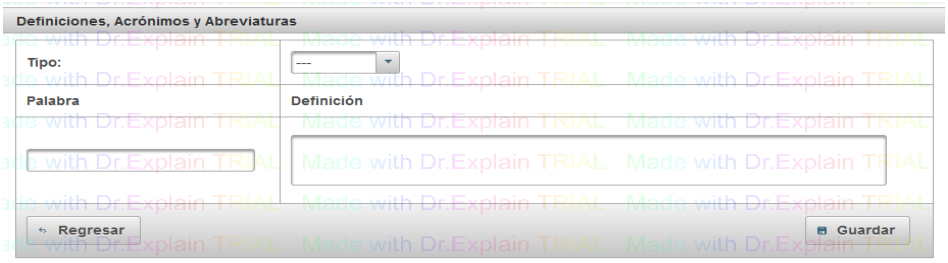
En esta tabla se ingresa todos los datos necesarios y se presiona el botón Guardar el cual almacena y muestra una nueva ventana en blanco para ingresar más datos si lo desea, caso contrario dar clic en regresar y aparecerá la ventana Descripción General para agregar otros datos.

- **Definiciones Acrónimos y Abreviaturas**

Clic en la pestaña Definiciones, Acrónimos y Abreviaturas, aparece de igual manera los datos agregados si los hubiera



Para agregar un dato nuevo al documento dar clic en y se aparecerá la siguiente ventana:



Se selecciona el tipo y agregamos la Palabra y su definición y guardamos.

- **AGREGAR REFERENCIAS**

Para agregar las referencias se aparecerá la siguiente ventana en donde se ingresaran los campos de texto.

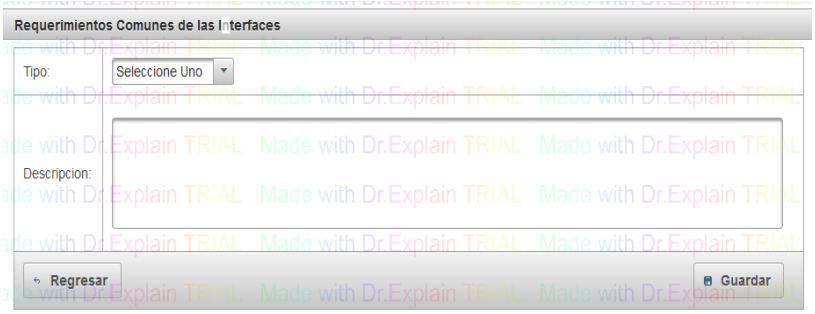
- **AGREGAR CARACTERISTICAS DE USUARIO**

En la siguiente ventana se podrá agregar características de usuario.

AGREGAR REQUERIMIENTOS ESPECIFICOS

Para agregar requerimientos a un Documento, después de agregar los datos generales se presiona el botón siguiente y se aparecerá la una ventana con 3 pestañas para cada grupo de requerimientos.

Para los requerimientos de interfaz se usa el botón agregar y en la ventana se agrega los datos y se guarda.



- Para los requerimientos funcionales y no funcionales se sigue el mismo proceso.

Bibliografía

Aurum, A., & Wohlin, C. (2005). *"Engineering and Managin Software Requirements"*. Springer.

Baresi L., G. F. (2001). *Extending UML form modeling Web Applications. In Proccedings of 34th annual Haway International Conference on System Sience.* .IEEE computer society.

Brisaboa, N. P. (2001). *A documental Database Query Language. String lPoccessing and Information Retrieval- SPIRE 2001*.

Coad, P. y. (1991). *Object-oriented analysis.* Englewood Cliffs: NJ: Yourdon Press.

Courage, C., & Baxter, K. (2005). *Understanding Your Users- A pracical Guide to User Requeriments Methods, Tools an Techniques.* Morgan Kauffman Publishers.

De Marco, T. (1978). *Structured analysis and systems sécification.* . Englewood: NJ: Prentice Hall.

De Troyer, O. (1997). *WSDM: A User Centred Design Method for Web SItes.* Belgium.

Devore, L. J. (s.f.). *Porbabilidades y Estadísticas para Ingeniria y Ciencias* (Vol. Sexta Edición).

Downs et al, E. y. (1992). *Structured systems analysis and design method: Application and context* (Vol. (2ª edición ed.)). New York: Prentice Hall.

DSMD, C. (1995). *Dynamic Systems Development Method.* Farnham Surrey. Inglaterra: Tesseract Publishers.

Durán Toro, Amador. (2000). *Un entorno metodógico de Ingeniería de Requisitos para Sistemas de Infirmación.* Sevilla.

Durán, A., & Bernádez, B. (2002). *"Metodología para la Elicitación de Requisitos de Sistemas Software (versión 2.3)".* Sevilla: Universidad de Sevilla.

EcuRed. (09 de 2010). *EcuRed:IEEE*. Recuperado el 28 de 07 de 2012, de EcuRed: http://www.ecured.cu/index.php/Institute_of_Electrical_and_Electronics_Engineers

ESCALONA, J. K. (2004). *REQUIREMENTS ENGINEERING FOR WEB APPLICATIONS. A COMPARATIVE STUDY.* Sevilla España.

Escalona, M. (2004). *Modelos y Técnicas para la especificación y el análisis de navegación en Systemas Software* . Sevilla- España: Ph. European Thesis. Deparment of Computer language an d systems.

F. Javier Diaz, M. C. (s.f.). *Usuando Jmeter para pruebas de rendimiento.* Recuperado el Domingo de Septiempre de 2013, de http://www.linti.unlp.edu.ar/uploads/docs/usando_jmeter_para_pruebas_de_rendimiento.pdf

Filkenstein, A. y. (1993). *Proceedings on 1993 IEEE International Symposium on Requirements Engineering* .

GONZALES, I. I. (19 de 11 de 2011). *Scribd*. Recuperado el 15 de 06 de 2012, de http://es.scribd.com/doc/73231315/34/Caracteristicas-de-un-Requerimiento

IEEE. (2010). *Un poco de historia*. Recuperado el 3 de Agosto de 2012, de IEEE-UCA: http://ewh.ieee.org/sb/el_salvador/uca/historia.html

IEEE. (s.f.). *IEEE-UCA*. Obtenido de RAMA ESTUDIANTIL DEL SALVADOR: http://ewh.ieee.org/sb/el_salvador/uca/historia.html

Jacobsob. (1993). *[Jacobson et al. 1993] I. Jacobson, M. Christerson, P. Jonsson, y G. Övergaard. Object–Oriented Software Engineering: A Use Case Driven Approach.*

Jacobson, I. (1992). *Object Oriented Software Engineering. A Use Case Driven.*

Jacobson, I. (1995). *Modeling whit use-case modelin. Journal of Object-Oriented.*

Jenkins, M. (2000). *Estandares de Calidad para el Desarrollo y Mantenimiento de Software.* Recuperado el 03 de 09 de 2013, de ftp://ftp.heanet.ie/disk1/sourceforge/i/in/ingdelsoftware/Tutorial.calidad.M.Jenkins.pdf

Koch, N. (2001). *Software Engineering for Adaptative Hypermedia Applications. Ph. Tesis Fast Reihe Softwaretechik Vol 12.* Uni-Drick Publishing Company.

Koch, N. (2001). *Software Engineering for Adaptative Hypermedia Applications. Ph. Tesis.* Desarrollo de sistemas web, Munich, Germany.

Lange, D. (1995). *An Object-Orientes Design Aproach for Developing Hypermedia Information System.* Germany: Research report RT00112, IBM Research, Tokyo Research Laboratory.

Lee, H. Y. (1998). *A Scenario-based object-orinted methodology for developer hypermedia information systems.* 31 st Annual Conference on Systems Science. Sprague R.

Lowe, E. J. (2002). *Client Needs and the Design Process in a web Proyects.* www2002 web Engineering Track.

Lubars, M. P. (1993). *A review of the state of the proactive in requirement modeling. Proceedings 1st International Symposium on Requirements Engineering.*

M.Grisselda Báez, S. I. (s.f.). *Metodologia DoRcu para la Ingenieria de Requerimientos.* La Habana, CUBA.

Martínez Guerrero, J. M., & Silva Delgado, C. A. (2010). *"Guía Metodológica para el levantamiento y análisis de requerimientos de Software en base a Procesos de Negocio".* Bogotá: Pontificia Universidad Javeriana.

MCEWEN. (2 de Abril de 2004). *Requirements: An Introdiction.* Obtenido de http://www.ibm.com/developerworks/rational/library/4166.html.

Morgan Kaufmann, S. (s.f.). *"Usability Engineering" .* Obtenido de http://www.linti.unlp.edu.ar/uploads/docs/usando_jmeter_para_pruebas_de_rendimiento.pdf

Olsila, L. (1998). *Building a Web -based information system applying the hypermedia flexible proces modeling strategy* (Vol. 1st International Workshop on Hypermedia Development. Hypertext 1998).

Palacios, J. (4 de Junio de 2006). *Compedio de Ingenieria de Software I.* Recuperado el 15 de Agosto de 2012, de safeCreative: http://www.safecreative.org/work/0710040047711

Pan, D. Z. (2001). *Requiremients Engineering Techniques. .* Canada: University of Calgary.

Perez Huebe, M. (2005). *Universidad Autonoma del Estado de Hidalgo.* Recuperado el 21 de Agosto de 2012, de Universidad Autonoma de Estado de Hidalgo: http://dgsa.uaeh.edu.mx:8080/bibliotecadigital/bitstream/231104/415/1/Ingenieria%20de%20requerimientos.pdf

Pytel, P., Uhalde, C., Ramón, H., Castello, H., Tomasello, M., Pollo-Cattaneo, M., . . . García Martínes, R. (2009). *Ingeniería de Requisitos basada en técnicasde ingeniería del conocimiento.* Buenos Aires.

R. Salazar, D., K. Villavicencio, M., & Monique, S. (s.f.). *Estudio estadístico exploratorio de las empresas desarrolladoras de.* Obtenido de http://www.fiec.espol.edu.ec/resources/investigacion/articulo90.pdf

Richard H, T., & Sidney C, B. (1997). *Software Requiremnts Engineering.* California: IEEE Computer Society.

Ross, D. y. (1977). Structrures analysis for requeriments definition. IEEE Transactions on Software Engineering(3),. In D. y. Ross, *Structrures analysis for requeriments definition. IEEE Transactions on Software Engineering(3),* (pp. 23-47).

Rumbaugh, J. (1991). *Object oriented modelling and design.* Englewood Cliffs: NJ:Prentice Hall.

Schwabe D., R. G. (1998). *Developing Hypermedia Applications usuing OOHDM. Workshop on Hypermedia development Process, Methods and Models .* Pittsburg: Hypertext ' 98.

Somerville, I. (2005). *Ingeniería de Software.* Madrid: Pearson Education.

Sommerville, I. (2002). *Procesos de la Ingeniería de Requerimientos, Cap.6.*

SWEBOK. (2004). *Software Engineering Body of Knowledge.*

Thayer, y. D. (1990). *Systems and Software Requirements Engineering. IEEE.*

Tuesta, A., Cataldi, Z., & Neil , C. (2011). *Metodologia para un portal educativo basada en spcificacion de requerimientos.* Recuperado el Julio de 2012, de Universidad Abierta Interamericana: caeti.uai.edu.ar/.../328_QUADERNS_DIGITALS_64.PDF

Vilain, P. S. (2000). *A diagrammatic Tool for Representing User Interaction in UML .* England: York.

Vilain, P. S. (2002). *A diagrammatic Tool for Representing User Interaction in UML. Lecture Notes in Computer Science UML' 2000.* England .

Wiegers, K. (2003). *Software Requirements.* Microsoft Press.

Young, R. (2004). *"The Requirements Engineering Handbook".* Artech House.

Yourdon, E. (1989). *Model Structures Analysis.* Prentice-Hall.

Zavala Ruiz, J. (2004). *Universidad Autónoma Metropolitana - Iztapalapa.* Recuperado el 7 de Marzo de 2013, de Universidad Autónoma Metropolitana - Iztapalapa: http://claroline.ucaribe.edu.mx/claroline/claroline/backends/download.php?url=L3Bvci1xdWUtZmFsbGFuLWxvcy1wcm95LWRlLXNvZnQucGRm&cidReset=true&cidReq=NI0215