



Facultad 9
Universidad de las Ciencias Informáticas.

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

Título: Propuesta de un modelo de Ingeniería de Requisitos para los
proyectos del Polo Video y Sonido Digital.

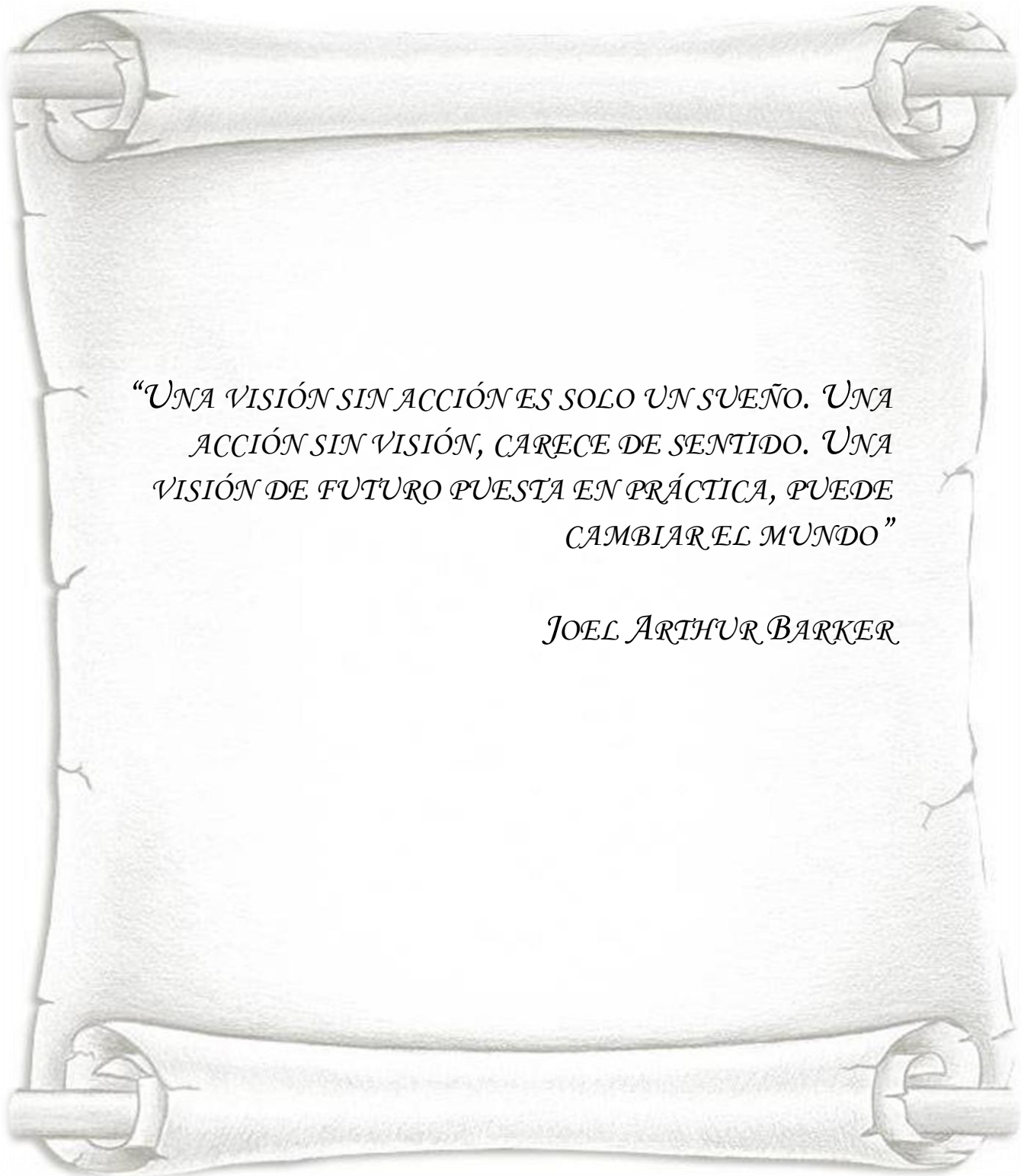
Autor: Maggie Arminda Silva de la Hera.

Tutor: Ing. Yunier Albrecht Delgado.

Co-tutor: Ms. Yeleny Zulueta Veliz.



Ciudad de la Habana, mayo de 2009.
Año del 50 aniversario del triunfo de la Revolución

A scroll with a quote and author name. The scroll is unrolled, showing a central rectangular area with text. The edges of the scroll are slightly frayed and have small holes at the corners, suggesting it was once tied with cords. The text is centered and written in a serif font.

*“UNA VISIÓN SIN ACCIONES SOLO UN SUEÑO. UNA
ACCIÓN SIN VISIÓN, CARECE DE SENTIDO. UNA
VISIÓN DE FUTURO PUESTA EN PRÁCTICA, PUEDE
CAMBIAR EL MUNDO”*

JOEL ARTHUR BARKER

DEDICATORIA

A mami y a papi por todo su sacrificio, a mi hermanita uno de mis grandes tesoros, al tío Pedro por ser ingeniero, licenciado, Dr., no te hacen falta los títulos. A mis abuelitos que no pudieron ver realizado este sueño: Marcela, Antonio, Aleida, Mario. A mi tía Cuqui, tu cariño no se olvida. A mi familia por ser la mejor del mundo. A todos los amigos. A Mailín, que hubiese sido mi compañera de tesis. A Fidel por la brillante idea de crear una universidad como la UCI.

AGRADECIMIENTOS

Agradezco a los que de una forma u otra pusieron su granito de arena para la realización de este trabajo.

A mi tutor Yunier Albrecht, a Solangel, a Yeleny Zulueta, y a los amigos de Video Web Frank, Angel, Yordany, Yake.

A Leonardo gracias por tanta paciencia y por el apoyo todos estos años.

A Roxi, Lily, Ivan, Yoly, Suyen.

A los que me han acompañado en estos 5 años de carrera dándome todo su apoyo para poder alcanzar mis sueños.

A mis padres, por todo el sacrificio que han hecho, el amor que me han dado y la persona que han hecho de mí. A mi hermanita, por todo el amor y apoyo que me ha brindado siempre.

A mi familia por apoyarme en todo momento.

A Peña y a Yaya por también ser mis padres y a toda su familia que también es la mía.

A mi familia y amigos de la UC9 (Roxi, Dayi, Paula, Lianne, Isachi, Yanet, Temis, Jeem, Yuya, Pimienta, Anibal, Yari, Ele, Glendy, Mairo, Ivis)

A mis amigos de Holguín Ailin, Frank, Yamill, Rebeca, Rosa, Abel, Tavo, Rey.

A UC9TeVe por los conocimientos aprendidos, los amigos, las malas noches y las risas.

A todos mis profesores y compañeros de grupo.

DECLARACIÓN DE AUTORÍA

Declaro que soy la única autora de este trabajo y autorizo a la facultad 9 y a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los 27 días del mes de mayo del año 2009.

Maggie Arminda Silva de la Hera

DATOS DE CONTACTO

Tutor: Ing. Yunier Albrecht Delgado, Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2007. Jefe del Polo de Video y Sonido Digital, Facultad 9.

Correo electrónico: yalbrecht@uci.cu

Co-Tutor: MsC. Yeleny Zulueta Veliz. Ingeniero en Informática, Universidad de Camagüey, 2004. Profesora Asistente del Departamento de Práctica Profesional e Ingeniería y Gestión de Software, Facultad 9.

Correo electrónico: yeleny@uci.cu. Teléfono: 837 2557

Co-Tutor: Ing. Aliosmi Lopez Velazquez. , Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2007.

Correo electrónico: alopezv@uci.cu

Asesor: Anisleidis Infante Rodríguez, Lic. en Lengua Inglesa con segunda Lengua Extranjera, Universidad de Oriente, 2007. Profesora, Adiestrada, Facultad 9.

Correo electrónico: anisleidisi@uci.cu

OPINIONES Y AVALES

OPINIÓN DEL TUTOR

RESUMEN

La mayoría de los proyectos de software que fracasan tienen su origen en las primeras fases de desarrollo. Una mala gestión de requisitos trae como consecuencia atrasos considerables en los proyectos y clientes insatisfechos. Los requisitos son la base para otras fases del desarrollo. En ocasiones resulta difícil procesar los requisitos de forma tal que al final del proyecto se satisfagan todas las necesidades del cliente. Para esto existe la Ingeniería de Requisitos (IR) que es la disciplina de la Ingeniería del Software dedicada a mejorar la captura, especificación, validación y la gestión de requisitos del software. La IR estudia y define métodos, notaciones y herramientas para garantizar una adecuada definición de los requisitos. La correcta especificación de requisitos determina en gran medida el éxito de un proyecto de desarrollo de software.

Esta investigación tiene como objetivo general proponer un procedimiento para la Ingeniería de Requisitos en los proyectos de desarrollo de software en el Polo de Video y Sonido Digital. Para realizar la propuesta se realiza un estudio de los modelos, así como las técnicas y herramientas que se utilizan en la IR, se exponen comparaciones entre éstas. Después de una entrevista realizada a los proyectos del Polo Video y Sonido Digital (PVSD), se evidenció que en ellos no se utilizaba algún modelo de IR ni herramientas de gestión de requisitos. La investigación muestra la propuesta de un modelo de acuerdo a las características de los proyectos del PVSD, se presentan además adaptaciones del modelo, y la presentación de un curso de capacitación enfocado a los analistas del polo para una mejor comprensión sobre los temas de la IR y en específico que aprendan a utilizar el modelo propuesto. Se aplica dicho modelo al proyecto Plataforma Video Web del PVSD, y se realizan comparaciones con el proyecto Sistema de Gestión de Televisión Universitaria (SG_DTU), demostrando la factibilidad del modelo propuesto.

PALABRAS CLAVES

Ingeniería de software, ingeniería de requisitos, requisitos.

DATOS EN INGLÉS

Summary

Most software projects that fail have their origin in the early phases of development. Poor management of requirements bring considerable delays in projects and unsatisfied customers. The requirements are the basis for further phases of development. Sometimes it's so hard to process requirements so that the final project will meet all customer needs. For this there is Requirements Engineering (IR) is the discipline of Software Engineering dedicated to improving the capture, specification, validation and requirements management software. The IR studies and define methods, notations and tools to ensure an adequate definition of the requirements. The correct specification of requirements largely determines the success of a software development project.

This research aims to propose a general procedure for the engineering requirements in software development projects in the Video and Digital Audio Pole. To make the proposed conducting a study of the models and the techniques and tools used in the IR, comparisons between these sets. After an interview with projects Video and Digital Audio Pole (VSDP), evidence that they did not use any model of IR or requirements management tools. Research shows a proposed model deal to the characteristics of projects VSDP, there are also adaptations of the model, and the presentation of a training course focused at analysts pole for a better understanding of the issues of IR and in particular to learn how to use the proposed model. It applies this model to the project's Web Video Platform VSDP, and comparisons are made with the project management system University Television (SG_DUT), demonstrating the feasibility of the proposed model.

Key words:

Software engineering, requirements engineering, requirements.

ÍNDICE DE TABLAS Y FIGURAS

<i>Figura 1 Modelo abstracto.</i>	16
<i>Figura 2 Modelo en cascada.</i>	17
<i>Figura 3 Modelo en espiral.</i>	18
<i>Figura 4 Modelo de procesos de ingeniería de requisitos de Pohl.</i>	19
<i>Figura 5 Modelo de Proceso de Duran.</i>	21
<i>Figura 6: Actividades del Modelo de Durán.</i>	37
<i>Figura 7 Aparición de las tareas generales en cada una de las actividades del modelo (Elaboración propia).</i>	39
<i>Figura 8 Representación del modelo modificado (Elaboración propia).</i>	40
<i>Figura 9 Procedimiento de gestión de riesgo (Elaboración propia).</i>	43
<i>Tabla 1 Factores del coste en proyectos software reales.</i>	10
<i>Tabla 2 Matriz de seguimiento genérica.</i>	23
<i>Tabla 3 Resumen de la Entrevista2.</i>	26
<i>Tabla 4 Resumen comparación de Modelos.</i>	27
<i>Tabla 5 Lista de Riesgos y clasificación.</i>	45
<i>Tabla 6 Riesgos por requisitos.</i>	45
<i>Tabla 7 Riesgos por tipos.</i>	46
<i>Tabla 8 Riesgos ordenados por efectos.</i>	47
<i>Tabla 9 Estrategias por riesgos.</i>	48
<i>Tabla 10 Indicadores potenciales por riesgos.</i>	49
<i>Tabla 11 Identificación de riesgos de los requisitos del proyecto VideoWeb.</i>	69
<i>Tabla 12 Probabilidad y efecto de los riesgos identificados en los requisitos del proyecto Plataforma VideoWeb.</i>	70
<i>Tabla 13 Estrategias para los riesgos identificados en los requisitos del proyecto VideoWeb.</i>	70
<i>Tabla 14 Comparación de los proyectos VideoWeb y Plataforma Tv, de acuerdo a la gestión de requisitos.</i>	73
<i>Tabla 15 Comparación de la gestión de requisitos entre la norma ISO y el modelo de Durán adaptado (Elaboración propia).</i>	74
<i>Tabla 16 Comparación de la gestión de requisitos entre la metodología de desarrollo RUP y el modelo de Durán adaptado (Elaboración propia).</i>	75

ÍNDICE DE CONTENIDO

INTRODUCCIÓN	1
1. CAPÍTULO 1: Ingeniería de Requisitos. Modelos, herramientas, actividades	6
1.1 Introducción	6
1.2 Conceptos asociados al dominio del problema.....	6
1.3 Modelos y procedimientos para la Ingeniería de Requisitos.....	8
1.3.1 Descripción General	8
1.3.2 Actividades fundamentales de la Ingeniería de Requisitos.	9
1.3.3 Técnicas para la identificación de requisitos (Escalona, y otros, 2005).	12
1.3.4 Técnicas para realizar la especificación de requisitos.	14
1.3.5 Técnicas para la realización de validación de los requisitos.	15
1.3.6 Modelos prácticos de IR.....	15
1.3.7 Modelo tradicional en cascada.....	16
1.3.8 Modelo en espiral.....	17
1.3.9 Modelo de Pohl.	18
1.3.10 Modelo de Durán.	20
1.3.11 Ingeniería de requisitos en RUP.	21
1.3.12 Ingeniería de requisitos en ISO 9001:2000.....	23
1.4 Situación actual del Polo Video y Sonido Digital.....	25
1.4.1 Situación Problemática	25
1.4.2 Análisis de los modelos.....	26
1.5 Herramientas automatizadas para la gestión de requisitos.	28
1.5.1 Herramientas automatizadas libres para la Gestión de Requisitos	33
1.6 Conclusiones del capítulo 1.....	35
CAPÍTULO 2: Solución Propuesta.	36
2.1 Breve descripción del modelo propuesto.....	36
2.2 Tareas recomendadas en cada una de las actividades del modelo propuesto.....	37
2.3 Adaptaciones del modelo de Durán para los proyectos del PVSD.	38
2.3.1 Elicitación de requisitos.....	40
2.3.2 Análisis de requisitos.	41
2.3.3 Validación de requisitos.	41

Índice de Contenido

2.3.4	Gestión de riesgos en la Ingeniería de Requisitos.....	42
2.3.5	Métricas en la Gestión de Requisitos.....	49
2.4	Herramienta a utilizar.....	52
2.4.1	Herramienta OSRMT Open Source Requirements Management Tool.....	53
2.5	Descripción del curso de capacitación para analistas del PVSD.....	56
2.5.1	Objetivos y metodología del curso de capacitación sobre Ingeniería de Requisitos orientado a los analistas del PVSD.....	56
2.5.2	Contenido del curso de capacitación.....	57
2.5.3	Descripción de temas.....	58
2.6	Aplicación del modelo de Duran al proyecto VideoWeb del PVSD.....	59
2.6.1	Elicitación de requisitos del proyecto VideoWeb del PVSD.....	59
2.6.2	Análisis de los requisitos del proyecto Plataforma VideoWeb del PVSD.....	67
2.6.3	Validación de requisitos del proyecto Plataforma VideoWeb.....	70
2.7	Conclusiones del capítulo 2.....	71
CAPÍTULO 3: Valoración del modelo Propuesto.....		72
3.1	Resumen de resultados en el desarrollo de la gestión de requisitos en el proyecto Plataforma VideoWeb.....	72
3.2	Comparación de los proyectos de acuerdo a la gestión de requisitos.....	72
3.3	Resultados obtenido con la gestión de requisitos según el modelo propuesto. (Contrastación de la hipótesis).....	73
3.4	Análisis con modelos de calidad.....	74
3.5	Análisis con metodologías de desarrollo.....	74
3.6	Mejoras del Modelo de Durán.....	75
3.7	Evaluación de la propuesta según expertos. Método Delphi.....	76
3.7.1	Pasos para la realización del Método Delphi.....	77
3.7.2	Selección de los expertos.....	78
3.7.3	Elaboración de la encuesta.....	78
3.7.4	Resultados de la evaluación.....	79
3.8	Conclusiones del capítulo 3.....	82
CONCLUSIONES.....		83
RECOMENDACIONES.....		84
BIBLIOGRAFÍA.....		85
ANEXOS.....		88
GLOSARIO.....		123

INTRODUCCIÓN

“La parte más difícil en la construcción de sistemas software es decidir precisamente qué construir. Ninguna otra parte del trabajo conceptual es tan dificultosa 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 es realizada en forma errónea. Ninguna otra parte es tan dificultosa de rectificar posteriormente” (FREDERICK P. BROOKS 1995).

Durante los primeros años de la informática, el software se consideraba como un añadido. La programación era un "arte", para el que no existían metodologías, era un proceso que se realizaba sin planificación alguna. En esta época toda la programación se desarrollaba a medida para cada necesidad concreta, y en consecuencia tenía muy poca difusión, habitualmente quien lo escribía era porque lo necesitaba, y era quien lo mantenía. Y esto trajo consigo retrasos considerables en la planificación, poca productividad, elevadas cargas de mantenimiento, demandas cada vez más desfasadas con las ofertas, baja calidad y fiabilidad del producto, dependencia de los realizadores. Esto se denominó crisis de software (Menéndez, 2008).

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no se lleva una metodología de por medio, lo que se obtiene es clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos. Crear un software sin una guía, es como estar en un lugar oscuro, no se sabe dónde se está ni hacia dónde se dirige.

La Ingeniería del Software, es el término utilizado por Fritz Bauer en la primera conferencia sobre desarrollo de software patrocinada por el Comité de Ciencia de la OTAN celebrada en Garmisch (Alemania), en octubre de 1968, previamente había sido utilizado por el holandés Edsger Dijkstra en su obra *The Humble Programmer*. Puede definirse según Alan Davis como "la aplicación inteligente de principios probados, técnicas, lenguajes y herramientas para la creación y mantenimiento, dentro de un coste razonable, de software que satisfaga las necesidades de los usuarios" (Menéndez, 2008).

A través de los años se ha podido constatar que los requerimientos o requisitos son la pieza fundamental en un proyecto de desarrollo de software, pues marcan el punto de partida para actividades como la planeación, básicamente en lo que se refiere a las estimaciones de tiempos y costos, así como la definición de recursos necesarios y la elaboración de cronogramas que será uno de los principales mecanismos de control con los que se contará durante la etapa de desarrollo. Además la especificación de requerimientos es la base que permite verificar si se alcanzaron o no los objetivos establecidos en el proyecto, estos son un reflejo detallado de las necesidades de los clientes o usuarios del sistema y es contra lo que se va a estar verificando si se están cumpliendo las metas trazadas (Chaves, 2006).

Es muy frecuente escuchar entre los concedores del desarrollo de software, que un gran número de los proyectos de software fracasan por no realizar una adecuada definición, especificación, y administración de los requerimientos. Dentro de esa mala administración se pueden encontrar factores como la falta de participación del usuario, requerimientos incompletos y el mal manejo del cambio a los requerimientos.

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software, esta se enfoca en un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas.

La IR no es más que la gestión de requisitos utilizando, herramientas, modelos y procedimientos. Los protagonistas fundamentales de este proceso son los requisitos que según la Real Academia Española (RAE) son circunstancia o condición necesaria para algo. Rational define requerimiento o requisito como: “Una condición o capacidad que el sistema (a construir) debe cumplir”

Merlin Dorfman y Richard H. Thayer lo precisan como:

- ✓ “Una capacidad de software necesaria para el usuario con el fin de resolver un problema o alcanzar un objetivo.”
- ✓ “Una capacidad de software que debe ser alcanzada o poseer para satisfacer un contrato, especificación, estándar u otra formalidad impuesta por la documentación.”

De igual forma, Ian Sommerville presenta una definición acerca de lo que es un Requerimiento”:

“Un requerimiento es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste” (Sommerville, 2005).

El desarrollo de software va tomando cada día más auge, cuando la mayoría de los países aboga por la informatización, Cuba no está exenta de esto, fue así cuando en el 2002 se materializó la idea genial y brillante del comandante Fidel Castro de crear la Universidad de Ciencias Informáticas (UCI) una universidad vinculada a las áreas de la investigación y la producción de software.

“La idea de convertir la informática en una de las ramas más productiva y aportadora de recursos para la nación. Es el empleo a fondo de la inteligencia y del capital humano que tenemos y principalmente del que podemos crear casi como espina dorsal de la economía, la Infraestructura Productiva de la UCI coordina la actividad productiva y brinda servicios para asegurar el correcto desarrollo y terminación de proyectos, productos o servicios. La producción se concentra en el desarrollo de proyectos en más de 30 Polos Productivos y se destacan resultados en las esferas de salud, educación, software libre, teleformación,

sistemas legales, realidad virtual, automatización, bioinformática, procesamiento de imágenes y señales, entre otras". (UCI, 2007)

Para realizar un desarrollo de software hay que saber lo que realmente necesita el usuario, quizás se entregue un producto pero si este no cumple con las necesidades del cliente, de qué sirve.

En los últimos años son varios los grupos que han trabajado y han propuesto metodologías que ofrecen procesos, modelos y técnicas adecuadas para trabajar con este tipo de sistemas (Koch, 1999 & 2001, Retschitzegger & Schwinger, 2000 y Escalona, Mejías & Torres, 2002). Sin embargo, si se analizan las diferentes propuestas, la gran mayoría de ellas enfocan su trabajo a la etapa de diseño del ciclo de vida, dando menor importancia a la ingeniería de requisitos, al testeo y a la gestión de calidad.

El tratamiento de requisitos 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 (Ferreira & Loucopoulos ,2001). La importancia de esta fase es esencial 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 requisitos.

En Cuba también se ha ido avanzando en estos aspectos, la Licenciada en Información Científico Técnica y Bibliotecología. Empresa DESOFT S. A. Cuba, Yadira Pérez Agusti realizó un breve estudio y lo publicó en una revista de la biblioteca virtual de salud, http://bvs.sld.cu/revistas/aci/vol17_3_08/aci09308.htm. En realidad fue algo muy sencillo, un documento pequeño, con información breve.

La UCI tiene un grupo de tesis que desarrollan la investigación de esta rama. Ejemplo de estas son:

- MIR-SWG: Modelo de Ingeniería de Requisitos para software de gestión en la Facultad 3.
- Proceso de Gestión de Requisito en el proyecto "Sistema Integrado para bibliotecas"
- Ingeniería de Requisitos para el desarrollo del Sistema de Gestión de Inventario Almacén (SIGIA).
- Procedimiento para desarrollar la Ingeniería de Requisitos en el proyecto Sistema de Gestión Penitenciaria

Cada una de estas tesis ha tenido en cuenta las características de los softwares a desarrollar. La IR requiere de los ajustes para cada proyecto en específico.

Este trabajo pretende darle solución a la siguiente **situación problemática**:

Después de realizar una entrevista al Jefe del Polo de Video y Sonido Digital de la Facultad 9, de la Universidad de las Ciencias Informáticas, se detecta que existen dificultades a la hora de realizar el levantamiento de requisito, estos incurren en lo siguiente:

- No utilización de herramientas para la Gestión de los Requisitos.
- No dedicar el tiempo que realmente se debe emplear en esta fase.
- No involucrar los roles que se requieren en el equipo de desarrollo.

Esto da a lugar al siguiente **problema a resolver**: No aplicación de modelos de Ingeniería de Requisitos en el desarrollo de software en el Polo de Video y Sonido Digital.

Se tiene como **objetivo general**: Proponer un procedimiento para la Ingeniería de Requisitos en los proyectos de desarrollo de software en el Polo de Video y Sonido Digital.

Donde se establece como **objeto de estudio**: Modelos y procedimientos para la Ingeniería de Requisitos.

Se señala como **campo de acción**: La Ingeniería de Requisitos en los proyectos de desarrollo de software en el polo de Video y Sonido Digital.

A partir de lo antes esbozado se plantea la siguiente **hipótesis**: Si se aplica un modelo para la IR en los desarrollos de software del Polo de Video y Sonido Digital, mejorará el flujo de trabajo de requerimientos en estos desarrollos.

Para dar solución al problema expuesto se estipulan las siguientes **tareas de la investigación**:

1. Evaluar el flujo de requerimiento de los proyectos de desarrollo de software en el Polo de Video y Sonido Digital.
 - 1.1 Seleccionar proyectos para evaluar el flujo de requerimiento.
 - 1.2 Seleccionar personal para aplicarle la entrevista.
 - 1.3 Aplicar entrevista al personal seleccionado.
2. Identificar técnicas, herramientas, actividades y modelos de la IR.
3. Analizar al menos 2 metodologías existentes para llevar a cabo el proceso de IR.
4. Proponer un procedimiento de Ingeniería de Requisitos para aplicar en los proyectos de desarrollo de software en el polo de Video y Sonido Digital.
5. Evaluar el modelo de IR propuesto para los proyectos de desarrollo de software del Polo de Video y Sonido Digital.
 - 5.1 Selección de los proyectos e indicadores para comparar.
 - 5.2 Evaluar los resultados de la comparación.

Al realizar las tareas debe tener los siguientes **posibles resultados**:

- Modelo de IR para los proyectos de desarrollo de software en el Polo de Video y Sonido Digital.
- Resumen de la aplicación del modelo en el proyecto Plataforma de VideoWeb del Polo de Video y Sonido Digital.

Métodos científicos de la investigación.

Métodos Teóricos:

- Análisis - Histórico – Lógico: Utilizado para determinar cuánto ha ido progresando la ingeniería de requisitos desde el 2000 hasta la actualidad (2009), especificar las tendencias actuales en este campo, las actividades, herramientas y modelos que se utilizan.
- Analítico – Sintético: Para entender con una mayor precisión la ingeniería de requisitos, de manera que el modelo propuesto sea el más conveniente y eficiente, Apoyándose en las tesis realizadas en la Universidad de las Ciencias Informáticas sobre el tema y la información encontrada en internet.

Métodos Empíricos:

- Entrevista: Este método se utilizará para percibir como se desarrolla el flujo de requerimiento en los proyectos de PVSD.
- Se utilizó además para definir la situación problemática y el problema de la investigación

Unidad de estudio de la Entrevista:

- ✓ Individuo con conocimientos de todo el trabajo que se realiza en cada uno de los proyectos del PVSD y experiencia en el levantamiento y gestión de requisitos.

Técnica de muestreo de la Entrevista:

- ✓ Muestreo Intencional: Esta técnica de muestreo no probabilística se aplica con el objetivo de obtener una muestra donde los integrantes sean los más representativos o con posibilidades de brindar mayor información. Se empleó esta técnica pues el objetivo de esta entrevista es comprender definir la situación problemática en el PVSD y esta persona es las que más información puede aportar al respecto.

Estructura de la tesis.

Para llevar a cabo el tema de la investigación se proponen 3 capítulos los cuales están estructurados de la siguiente forma:

En el capítulo 1 se analiza la situación actual del Polo Video y Sonido Digital (PVSD), se mencionan algunas herramientas, actividades y modelos, centrandolo en estos últimos, se llega a conclusiones sobre qué modelo aplicar en los proyectos del PVSD.

En el capítulo 2 se explica brevemente en qué consiste el modelo propuesto, se le realizan adaptaciones al mismo, además se especifican las herramientas a utilizar, se propone un *curso de capacitación* orientado a los Analistas del PVSD y se aplica el modelo al proyecto VideoWeb de dicho polo.

En el capítulo 3 se valida el modelo propuesto, mediante la comparación del proyecto VideoWeb y otro proyecto, el cual no haya utilizado el modelo para verificar que la propuesta es eficiente además se evalúa la misma mediante el método Delphi.

1. CAPÍTULO 1: Ingeniería de Requisitos. Modelos, herramientas, actividades.

1.1 Introducción

La Ingeniería de Requisitos es una disciplina joven aún, a pesar de esto se han desarrollado diversos modelos con actividades implícitas y herramientas, y con el transcurso de los años estos se han perfeccionado.

En este capítulo se analiza el funcionamiento del flujo de requerimientos de los distintos proyectos del Polo Video y Sonido Digital, además se aborda algunos temas de la IR como son: modelos, herramientas, actividades.

1.2 Conceptos asociados al dominio del problema

Para el dominio del problema es necesario entender con claridad algunos conceptos importantes como son los requisitos y la ingeniería de requerimientos.

La “Ingeniería de Requerimientos ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactuarán los usuarios finales con el software” (Pressman, 2005).

El proceso de recopilar, analizar y verificar las necesidades del cliente o usuario para un sistema es llamado ingeniería de requerimientos. La meta de la ingeniería de requerimientos (IR) es entregar una especificación de requisitos de software correcta y completa (Herrera J, 2003).

La Ingeniería de Requisitos cumple un papel primordial en el proceso de producción de software, pues enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente, el comportamiento del sistema.

Los requisitos o requerimientos son el fundamento de la IR.

De las muchas definiciones que existen para requerimiento, a continuación se presenta una que aparece en el glosario de la IEEE (Institute of Electrical and Electronics Engineers).

(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 (1) o (2).

Los requerimientos pueden dividirse en requerimientos funcionales y requerimientos no funcionales.

1.2.1 Clasificación de los requisitos, características y visión.

Los requerimientos funcionales definen las funciones que el sistema será capaz de realizar. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas.

Los requerimientos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares, etc.

Las características de un requerimiento están definidas por sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo. A continuación se presentan las más importantes.

- **Necesario:** Un requerimiento es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- **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.
- **Verificable:** Un requerimiento es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o pruebas (Herrera J, 2003).

Durante la etapa de especificación de requerimientos se pueden presentar muchos inconvenientes los cuales son importantes de identificar y prevenir, a continuación se presenta un listado con las principales dificultades para definir los requerimientos:

- ✓ Los requerimientos no son obvios y vienen de muchas fuentes.
- ✓ Son difíciles de expresar en palabras (el lenguaje es ambiguo).
- ✓ Existen muchos tipos de requerimientos y diferentes niveles de detalle.
- ✓ La cantidad de requerimientos en un proyecto puede ser difícil de manejar.
- ✓ Nunca son iguales. Algunos son más difíciles, más riesgosos, más importantes o más estables que otros.

- ✓ Los requerimientos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- ✓ Cada requerimiento tiene propiedades únicas y abarcan áreas funcionales específicas.
- ✓ Un requerimiento puede cambiar a lo largo del ciclo de desarrollo.
- ✓ Son difíciles de cuantificar, cada conjunto de requerimientos es particular para cada proyecto (Herrera J, 2003).

Los requisitos tradicionalmente han tenido la siguiente visión:

- ✓ Ancestralmente se ha supuesto que la elaboración de los requisitos era responsabilidad única del cliente.
- ✓ Todos los modelos de desarrollo comenzaban con el análisis de unos requisitos supuestamente proporcionados por el cliente.
- ✓ Actualmente, se asume que la elaboración de los requisitos es una responsabilidad compartida entre clientes, usuarios y desarrolladores.
- ✓ Esta nueva visión es la que ha llevado a definir una ingeniería de requisitos (Durán, 2006).

Dejando claro estos aspectos, se tiene un mayor dominio sobre el tema a tratar. Los requisitos son el ente de la IR.

1.3 Modelos y procedimientos para la Ingeniería de Requisitos

1.3.1 Descripción General

El proceso de IR puede ser descrito en 5 pasos distintos (Pressman, 2005): Identificación de Requisitos, Análisis y Negociación de Requisitos, Especificación de Requisitos, Modelado del Sistema, Validación de Requisitos y Gestión de Requisitos. Pohl en "Requirements Engineering: An Overview". Encyclopedia of Computer Science and Technology, 1997, define cuatro actividades fundamentales: Elicitación de Requisitos, Negociación de Requisitos, Validación y Verificación de Requisitos, Especificación y Documentación de Requisitos. La propuesta de (Durán, 2000) consta de tres actividades principales, elicitación, análisis y validación.

La adopción de uno u otro criterio está en dependencia de las características del proyecto que se lleva a cabo. Las actividades de la IR no son criterios esquemáticos sino que evolucionan según las experiencias de cada desarrollador en este campo de la IS.

Un modelo es una simplificación de la realidad que incluye aquellos elementos que tienen una gran influencia y omite aquellos elementos que no son relevantes para el nivel de abstracción dado.

En definitiva, los modelos son abstracciones simplificadas y estandarizadas de actividades repetitivas, generalmente producidos desde un punto de vista determinado, por lo que pueden existir diferentes modelos para un mismo proceso.

1.3.2 Actividades fundamentales de la Ingeniería de Requisitos.

Independientemente de cuál sea el modelo de Ingeniería de Requisitos, debe poseer algunas actividades fundamentales, estas actividades se dividen en 4, y se mencionan a continuación:

- Extracción
- Análisis
- Especificación
- Validación

Como toda división de tareas, no es una estricta representación de la realidad, sino que se hace con el fin de sistematizar la realización de la IR. En general la delimitación entre una actividad y la otra no es tan clara, están sumamente interrelacionadas, existiendo un alto grado de iteración y retroalimentación entre una y otra (Dávila 2001).

Extracción

El proceso mediante el cual los clientes o futuros usuarios del software descubren, revelen, articulan y comprenden los requisitos que desean.

Esta fase representa el comienzo de cada ciclo. Extracción es el nombre comúnmente dado a las actividades involucradas en el descubrimiento de los requerimientos del sistema. Aquí, los Analistas deben trabajar junto al cliente para descubrir el problema que el sistema debe resolver, los diferentes servicios que el sistema debe prestar, las restricciones que se pueden presentar, etc.

Esto no suele ser tarea fácil: muchas veces los clientes/usuarios no tienen una idea clara de sus necesidades reales, diversas personas dentro de la organización tienen necesidades encontradas, pueden existir limitaciones técnicas o tecnológicas para cumplir con algunos requerimientos, etc. Pero, en definitiva, descubrir los requerimientos del sistema no sólo implica preguntar a las personas qué quieren: es un proceso delicado que involucra comprender el dominio de aplicación, es decir, obtener un conocimiento del área general de aplicación del sistema; comprender el problema en sí, lo que implica que se debe extender y especializar el conocimiento sobre el dominio general para que se aplique al cliente en particular; comprender el negocio, por tanto, se debe entender en profundidad cómo es que este sistema interactuará con las partes del negocio que estarán involucradas y cómo puede contribuir a lograr las metas de la empresa; finalmente, comprender las necesidades y restricciones de los usuarios del sistema,

en particular, se deben entender los procesos de trabajo que se supone que el sistema apoyará y el rol de cualquier otro sistema que actualmente se involucre en dichos procesos.

Es importante, entonces, que la extracción sea efectiva, pues la aceptación del sistema dependerá de cuán bien éste satisfaga las necesidades del cliente y de cuán bien asista a la automatización del trabajo.

Análisis

Sobre la base de la extracción realizada previamente, comienza esta fase que se presenta sumamente compleja en un proyecto donde el dominio es desconocido en la cual se apunta a descubrir problemas con los requerimientos del sistema identificados hasta el momento (Dávila, 2001).

El análisis es el proceso de razonamiento sobre los requisitos obtenidos, detectando y resolviendo posibles inconsistencias o conflictos. Es el estudio de las necesidades de los usuarios para llegar a una definición de los requisitos del sistema.

Importancia del análisis de requisitos.

Los problemas con los requisitos constituyen la principal fuente de problemas (37%)

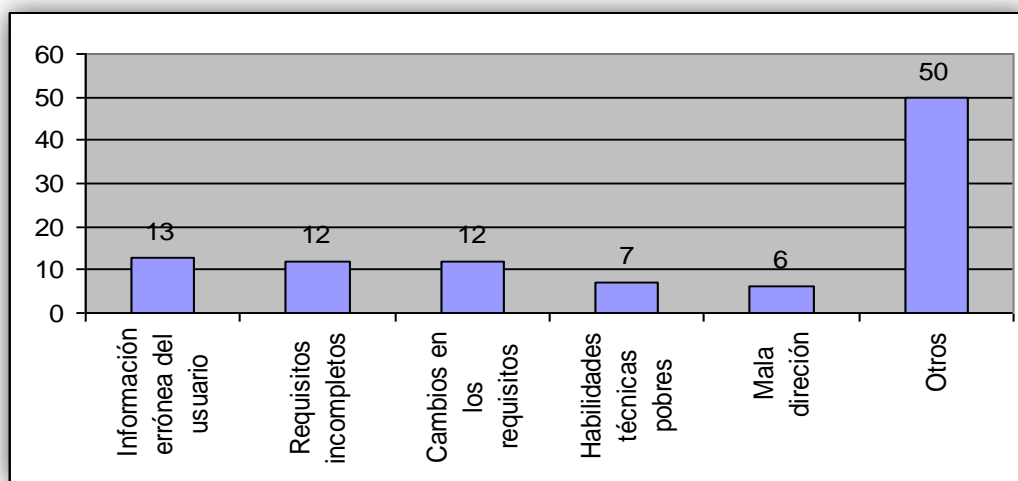


Tabla 1 Factores del coste en proyectos software reales (Standish94, <http://www.standishgroup.com/chaos/toc.php>)

Especificación de requisitos

Es el proceso de redacción o registro de los requisitos. Para este proceso puede recurrirse al lenguaje natural, lenguajes formales (Dávila, 2001).

En la práctica, esta etapa se va realizando conjuntamente con el análisis, se puede decir que la especificación es el "pasar en 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 (La Ingeniería de Requerimiento y su importancia en el desarrollo de proyectos de software, 2006)

Validación de los requisitos.

Es el proceso de confirmación, por parte de los usuarios o clientes, de que los requisitos especificados son válidos, consistentes, completos (Dávila, 2001).

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 para asegurarse que representan una descripción, por lo menos, aceptable del sistema que se debe implementar. Esto implica verificar que los requerimientos sean consistentes y que estén completos (La Ingeniería de Requerimiento y su importancia en el desarrollo de proyectos de software, 2006).

Esta etapa puede confundirse con la de análisis, pero la diferencia es clara: mientras que en el análisis se trabaja sobre el boceto del documento de requerimientos, en la validación se utiliza el documento final, lo que equivale a decir, los requerimientos "depurados" (Dávila, 2001).

Los requisitos una vez definidos necesitan ser validados. La validación de requisitos tiene como misión demostrar que la definición de los requisitos define realmente el sistema que el usuario necesita o el cliente desea. Es necesario asegurar que el análisis realizado y los resultados obtenidos de la etapa de definición de requisitos son correctos.

Se puede apreciar que el proceso de ingeniería de requerimientos es un conjunto estructurado de actividades, mediante las cuales se obtiene, se valida y se logra dar un mantenimiento adecuado al documento de especificación de requerimientos, que es el documento final, de carácter formal, que se obtiene de este proceso. Es necesario recalcar que no existe un proceso único que sea válido de aplicar en todas las organizaciones. Cada organización debe desarrollar su propio proceso de acuerdo al tipo de producto que se esté desarrollando, a la cultura organizacional, y al nivel de experiencia y habilidad de las personas involucradas en la ingeniería de requerimientos. Hay muchas maneras de organizar el proceso de ingeniería de requerimientos y en otras ocasiones se tiene la oportunidad de recurrir a consultores, ellos tienen una perspectiva más objetiva que las personas involucradas en el proceso.

1.3.3 Técnicas para la identificación de requisitos (Escalona, y otros, 2005).

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

Entrevistas: resultan una técnica muy aceptada dentro de la ingeniería de requisitos y su uso está ampliamente extendido. Las entrevistas le permiten al analista tomar conocimiento del problema y comprender los objetivos de la solución buscada. A través de esta técnica el equipo de trabajo se acerca al problema de una forma natural. Existen muchos tipos de entrevistas y son muchos los autores que han trabajado en definir su estructura y dar guías para su correcta realización. Básicamente, la estructura de la entrevista abarca tres pasos: identificación de los entrevistados, preparación de la entrevista, realización de la entrevista y documentación de los resultados (protocolo de la entrevista). Las entrevistas, sin embargo, no es una técnica sencilla de aplicar. Requiere que el entrevistador sea experimentado y tenga capacidad para elegir bien a los entrevistados y obtener de ellos toda la información posible en un período de tiempo siempre limitado. Bajo este aspecto la preparación de la entrevista representa un papel esencial.

JAD (Joint Application Development/Desarrollo conjunto de aplicaciones): esta técnica resulta una alternativa a las entrevistas. Es una práctica de grupo que se desarrolla durante varios días y en la que participan analistas, usuarios, administradores del sistema y clientes. Está basada en cuatro principios fundamentales dinámica de grupo, el uso de ayudas visuales para mejorar la comunicación, mantener un proceso organizado y racional y una filosofía de documentación WYSIWYG (What You See Is What You Get, lo que ve es lo que obtiene), es decir, durante la entrevista se trabajará sobre lo que se generará. Tras una fase de preparación del JAD al caso concreto, el equipo de trabajo se reúne en varias sesiones. En cada una de ellas se establecen los requisitos de alto nivel a trabajar, el ámbito del problema y la documentación. Durante la sesión se discute en grupo sobre estos temas llegando a una serie de conclusiones que se documentan. En cada sesión se van concretando más las necesidades del sistema. Esta técnica presenta una serie de ventajas frente a las entrevistas tradicionales, la misma ahorra tiempo al evitar que las opiniones de los clientes se tengan que contrastar por separado. Pero requiere un grupo de participantes bien integrados y organizados.

Brainstorming (Tormenta de ideas): es también una técnica de reuniones en grupo cuyo objetivo es que los participantes muestren sus ideas de forma libre. Consiste en la mera acumulación de ideas y/o información sin evaluar las mismas. El grupo de personas que participa en estas reuniones no debe ser muy numeroso (máximo 10 personas), una de ellas debe asumir el rol de moderador de la sesión, pero sin carácter de controlador. Como técnica de captura de requisitos es sencilla de usar y de aplicar, contrariamente al JAD puesto que no requiere tanto trabajo en grupo como éste. Además suele ofrecer

una visión general de las necesidades del sistema, pero normalmente no sirve para obtener detalles concretos del sistema, por lo que suele aplicarse en los primeros encuentros.

Concept Mapping: Los concept maps son grafos en los que los vértices representan conceptos y las aristas representan posibles relaciones entre dichos conceptos. Estos grafos de relaciones se desarrollan con el usuario y sirven para aclarar los conceptos relacionados con el sistema a desarrollar. Son muy usados dentro de la Ingeniería de Requisitos pues son fáciles de entender por el usuario, más aún si el equipo de desarrollo hace el esfuerzo de elaborarlo en el lenguaje de éste. Sin embargo, deben ser usados con cautela porque en algunos casos pueden ser muy sugestivos y pueden llegar a ser ambiguos en casos complejos si no se acompaña de una descripción textual.

Sketches y Storyboards: Esta técnica es frecuentemente usada por los diseñadores gráficos de aplicaciones en el entorno web. La misma consiste en representar sobre papel en forma muy esquemática las diferentes interfaces al usuario (sketches). Estos sketches pueden ser agrupados y unidos por enlaces dando idea de la estructura de navegación (storyboard).

Casos de Uso: Aunque inicialmente se desarrollaron como técnica para la definición de requisitos, algunos autores proponen casos de uso como técnica para la captura de requisitos. Los casos de uso permiten mostrar el contorno (actores) y el alcance (requisitos funcionales expresados como casos de uso) 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 una determinada función. Los actores son elementos externos (personas, otros sistemas, etc.) que interactúan con el sistema como si de una caja negra se tratase. Un actor puede participar en varios casos de uso y un caso de uso puede interactuar con varios actores. 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 si no se acompañan con una información textual o detallada con otra técnica como pueden ser diagramas de actividades.

Cuestionarios y Checklists: Esta técnica requiere que el analista conozca el ámbito del problema en el que está trabajando. Consiste en redactar un documento con preguntas cuyas respuestas sean cortas y concretas, o incluso cerradas por unas cuantas opciones en el propio cuestionario (Checklist). Este cuestionario será cumplimentado por el grupo de personas entrevistadas o simplemente para recoger información en forma independiente de una entrevista.

Comparación de terminología: Uno de los problemas que surge durante la elicitación de requisitos es que usuarios y expertos no llegan a entenderse debido a problemas de terminología. Esta técnica es utilizada en forma complementaria a otras para obtener consenso respecto de la terminología a ser usada en el proyecto de desarrollo.

Para ello es necesario identificar el uso de términos diferentes para los mismos conceptos (correspondencia), misma terminología para diferentes conceptos (conflictos) o cuando no hay concordancia exacta ni en el vocabulario ni en los conceptos (contraste).

1.3.4 Técnicas para realizar la especificación de requisitos.

También para la actividad especificación de requisitos en el proceso de ingeniería de requisitos hay un gran número de técnicas propuestas. A continuación se presentan algunas de ellas.

Lenguaje natural: Resulta una técnica muy ambigua para la definición de los requisitos. Consiste en definir los requisitos en lenguaje natural sin usar reglas para ello. Pero, a pesar de que son muchos los trabajos que critican su uso, es cierto que a nivel práctico se sigue utilizando.

Glosario y ontologías: La diversidad de personas que forman parte de un proyecto software hace que sea necesario establecer un marco de terminología común. Esta necesidad se vuelve más patente en los sistemas de información web puesto que el equipo de desarrollo en ellas suele ser más interdisciplinario. Por esta razón son muchas las propuestas que abogan por desarrollar un glosario de términos en el que se recogen y definen los conceptos más relevantes y críticos para el sistema. En esta línea se encuentra también el uso de ontologías, en las que no sólo aparecen los términos, sino también las relaciones entre ellos. Ninguna de las metodologías para el entorno web incluidas en este estudio comparativo por su relevancia en la ingeniería de requisitos propone la definición de una ontología.

Plantillas o patrones: Esta técnica, recomendada por varios autores, tiene por objetivo el describir los requisitos mediante el lenguaje natural pero de una forma estructurada. Una plantilla es una tabla con una serie de campos y una estructura predefinida que el equipo de desarrollo va cumplimentando usando para ello el lenguaje del usuario. Las plantillas eliminan parte de la ambigüedad del lenguaje natural al estructurar la información; cuanto más estructurada sea ésta menos ambigüedad ofrece. Sin embargo, si el nivel de detalle elegido es demasiado detallado, el trabajo de rellenar las plantillas y mantenerlas puede ser demasiado tedioso.

Escenarios: La técnica de los escenarios consiste en describir las características del sistema a desarrollar mediante una secuencia de pasos. La representación del escenario puede variar dependiendo del autor. Esta representación puede ser casi textuales o ir encaminada hacia representaciones gráficas en forma de diagramas de flujo. El análisis de los escenarios, hechos de una forma u otra, pueden ofrecer información importante sobre las necesidades funcionales de sistema.

Casos de uso: Es como técnica de definición de requisitos como más ampliamente han sido aceptados los casos de uso. Actualmente se ha propuesto como técnica básica del proceso RUP. Sin embargo, son varios los autores que defienden que pueden resultar ambiguos a la hora de definir los

requisitos por lo que hay propuestas que los acompañan de descripciones basadas en plantillas o de diccionarios de datos que eliminen su ambigüedad.

Lenguajes Formales: Otro grupo de técnicas que merece la pena resaltar como extremo opuesto al lenguaje natural, es la utilización de lenguajes formales para describir los requisitos de un sistema. Las especificaciones algebraicas como ejemplo de técnicas de descripción formal, han sido aplicadas en el mundo de la ingeniería de requisitos desde hace años. Sin embargo, resultan muy complejas en su utilización y para ser entendidas por el cliente. El mayor inconveniente es que no favorecen la comunicación entre cliente y analista. Por el contrario, es la representación menos ambigua de los requisitos y la que más se presta a técnicas de verificación automatizadas.

1.3.5 Técnicas para la realización de validación de los requisitos.

Reviews o Walk-throughs: Está técnica consiste en la lectura y corrección de la completa documentación o modelado de la definición de requisitos. Con ello solamente se puede validar la correcta interpretación de la información transmitida. Más difícil es verificar consistencia de la documentación o información faltante.

Auditorías: La revisión de la documentación con esta técnica consiste en un chequeo de los resultados contra una checklist predefinida o definida a comienzos del proceso, es decir que sólo una muestra es revisada.

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

Prototipos: Algunas propuestas se basan en obtener de la definición de requisitos prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea de la estructura de la interfaz del sistema con el usuario. Esta técnica tiene el problema de que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final.

1.3.6 Modelos prácticos de IR

Todos los modelos existentes sobre la IR tienen su punto de partida en uno base el cual es denominado "modelo abstracto". Muestra una visión de las actividades que se realizan en la IR. Esto se percibe en la figura siguiente:

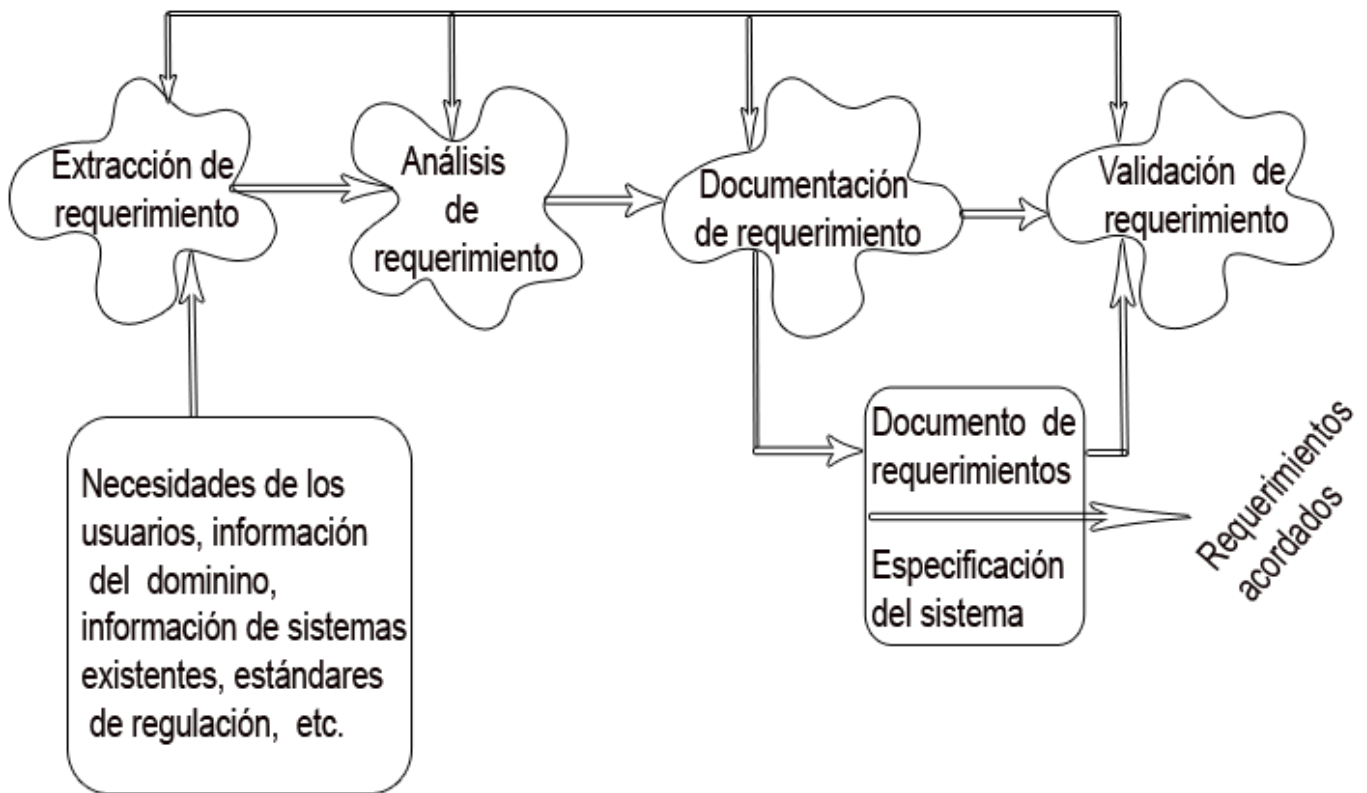


Figura 1 Modelo abstracto.

Las diversas necesidades de las diferentes organizaciones comienzan a surgir a partir de la aplicación de modelos más detallados. Así, se tienen dos modelos básicos que permiten estudiar el proceso de IR y del cual se derivan numerosas variantes que dependerán del caso de estudio en cuestión.

1.3.7 Modelo tradicional en cascada

Este modelo sugiere que los resultados de una tarea del proceso llevan a la siguiente, y así sucesivamente. En el ejemplo presentado, la extracción lleva al análisis, el análisis desencadena la documentación, y la documentación inicia la validación (Dávila, 2001).

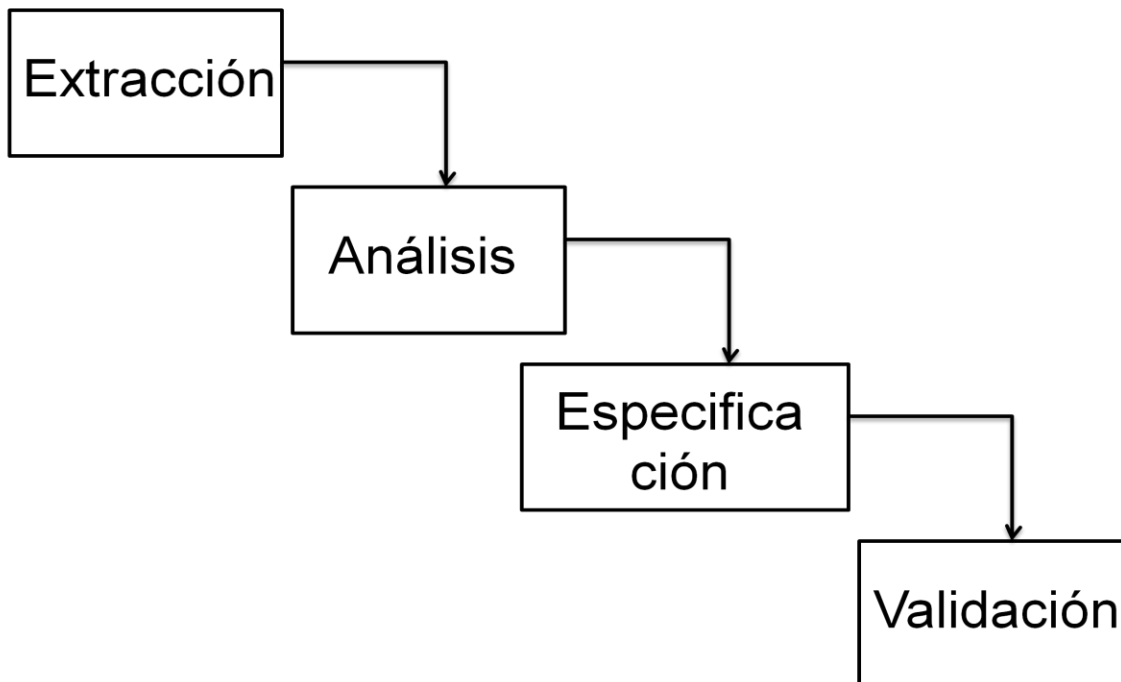


Figura 2 Modelo en cascada.

El modelo es útil si se ve de modo general para comprender el proceso. En la IR no van a existir fases delimitadas claramente, porque de una forma u otra los requisitos suelen cambiar durante el desarrollo del mismo, y hay que volver hacia atrás para darle solución a los cambios. O sea que las etapas se retroalimentan.

1.3.8 Modelo en espiral

Un modo alternativo de presentar modelos de actividad que toma en cuenta la retroalimentación entre etapas y la repetición de tareas, es el llamado Modelo en Espiral (Dávila, 2001).



Figura 3 Modelo en espiral

En este diagrama, el uso de la espiral implica que las diferentes actividades de la ingeniería de requerimientos son repetidas hasta que se toma la decisión final, que es la aceptación del documento de especificación de requerimientos.

Es decir, si en el diseño preliminar se encuentran problemas, entonces se recorre el ciclo nuevamente (extracción-análisis-especificación-validación) hasta que todos sean resueltos, que es lo mismo que decir que este ciclo continuará hasta que se pueda elaborar un documento aceptable.

Pero también existen factores externos que pueden determinar la finalización del ciclo, como por ejemplo la presión por cumplir con un determinado cronograma.

1.3.9 Modelo de Pohl.

El modelo de Pohl (Pohl, 1997) es un modelo iterativo en el que se definen las cuatro actividades que pueden verse en la figura 4, el orden de realización de las actividades puede ser cualquiera, se asume

una secuencia en la que los requisitos son elicitados, a continuación son negociados entre los participantes, se integran con el resto de la documentación y finalmente se validan y verifican para asegurar que corresponden con las necesidades reales de los clientes y usuarios y que no presentan conflictos con los demás requisitos.



Figura 4 Modelo de procesos de ingeniería de requisitos de Pohl.

Este modelo presenta cuatro características principales que se explican a continuación:

Elicitación de requisitos: El objetivo de la elicitación es mostrar el conocimiento oculto sobre las necesidades de clientes y usuarios y el sistema a desarrollar para que sean de fáciles para la comprensión de todos los participantes en el problema. En este modelo durante el desarrollo de las actividades es necesario identificar a las fuentes de información, conocer lo mejor posible el dominio del problema, reutilizar especificaciones de requisitos similares en la medida de lo posible y utilizar las técnicas habituales de elicitación como son las entrevistas, casos de uso, cuestionarios, prototipos, etc.

Negociación de requisitos: Tiene como objetivo lograr acuerdos entre todos los participantes sobre los requisitos elicitados, avanzando en la dimensión de acuerdo del proceso. Para ello hay que tener en cuenta cuatro factores:

- ✓ Hacer explícitos los conflictos y evitar los conflictos emocionales entre los participantes, de forma que quede claro qué es lo que se negocia y que dicha negociación no se vea afectada por motivos personales.
- ✓ Hacer explícitos para cada conflicto las alternativas, las argumentaciones y las razones subyacentes que los provocan, de forma que la negociación pueda basarse en las raíces del conflicto.
- ✓ Asegurarse de que se toman las decisiones correctas, de forma que la mayoría de los participantes estén de acuerdo en los resultados de la negociación y no se sientan desplazados del proceso.
- ✓ Asegurarse de involucrar a las personas adecuadas en el momento adecuado, para evitar tener que volver a replantear las negociaciones porque alguno de los participantes afectados no participó en las negociaciones oportunas.

Especificación/Documentación de requisitos. El objetivo es claro: documentar los requisitos elicitados y negociados. Pohl propone utilizar las notaciones que sean necesarias para que todos los participantes la entiendan. De este modo, según su propuesta, se avanza en la dimensión de formalidad del proceso.

Validación/Verificación de requisitos. El objetivo es comprobar que los requisitos documentados corresponden a las necesidades de los clientes y usuarios (validación) y comprobar que la especificación cumple los criterios de calidad oportunos (verificación).

1.3.10 Modelo de Durán.

Este modelo consta de consta de tres actividades fundamentales: elicitación, análisis y validación, y su principal característica es la interactividad. Ver figura 5.

El proceso de elicitar y negociar requisitos, analizarlos y validarlos es iterativo por naturaleza, pues prácticamente es imposible obtener todos los requisitos y que éstos tengan las propiedades antes discutidas sin tener que volver atrás en algún momento del proceso. Sin embargo, no por ello se debe dejar de intentar elicitar, y posteriormente analizar y validar la mayor cantidad posible de requisitos en cada iteración, intentando alcanzar la situación ideal en el que el proceso sería lineal. Al igual que en cualquier otra fase de la ingeniería de software, en la ingeniería de requisitos se incluyen también aspectos de gestión de recursos, verificación, control de calidad, control de configuración, etc.

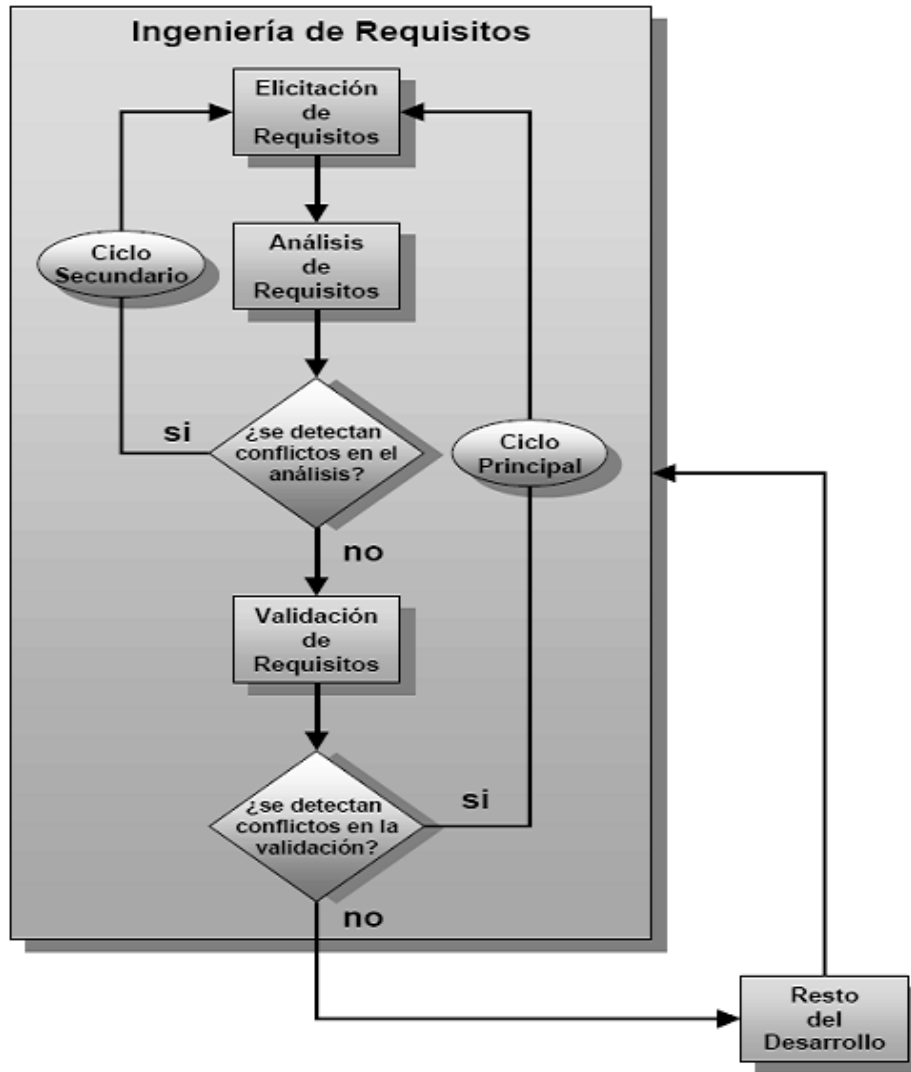


Figura 5 Modelo de Proceso de Duran.

1.3.11 Ingeniería de requisitos en RUP.

El Proceso Unificado de Rational (RUP, el original inglés Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Está basado a seis principios claves: adaptar el proceso, balancear prioridades, colaboración entre equipos, demostrar valor iterativamente, elevar el nivel de abstracción, enfocarse en la calidad (E.Fenton, y otros, 1998).

RUP brinda una guía para encontrar, organizar, documentar, y seguir los cambios de los requisitos funcionales y restricciones. Utiliza una notación de Caso de Uso y escenarios para representar los requisitos.

Dentro de RUP está presente el proceso de ingeniería de requisitos, el cual garantiza que se apliquen mecanismos para comprender lo que verdaderamente quiere el cliente; analizando sus necesidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedades, validando la especificación y gestionando los requisitos. Estos mecanismos se definen en cinco pasos:

Identificación de requisitos: los requisitos se identifican teniendo en cuenta cuestiones de alcance, de comprensión y de volatilidad, RUP proporciona vías para solucionar estas cuestiones de forma organizada, sugiere un conjunto de actuaciones que van a ser útiles a la hora de obtener los requisitos, incluyendo tareas adicionales para grandes sistemas establece que cada uno de los productos obtenidos debe ser revisado por las personas que hayan participado en la obtención de sus requisitos.

Análisis y negociación de requisitos: los requisitos se agrupan por categorías y se organizan en subconjuntos, se estudia cada requisito en relación con el resto, se examinan los requisitos en su consistencia, complejidad y ambigüedad, y se clasifican en base a las necesidades de los clientes. Para esto se plantean una serie de cuestiones. Si los diferentes clientes no pueden facilitar los requisitos, el riesgo de error es muy alto. Los involucrados deben clasificar sus requisitos y discutir los posibles conflictos según su prioridad. Se identifican y analizan los riesgos asociados a cada requisito, se estima esfuerzo para valorar el impacto de cada requisito en el coste del proyecto y en el plazo de entrega. Se utiliza un procedimiento iterativo donde se irán eliminando, combinando y/o modificando cada requisito para cumplir los objetivos planteados.

Especificación de requisitos: es la actividad en la cual se genera el documento, que contiene las necesidades y funcionalidades del sistema, alcance del sistema, definiendo los requerimientos funcionales y los no funcionales del sistema.

Modelado del sistema: En este paso se trata de obtener la mayor cantidad de información. Es importante evaluar los componentes del sistema y sus relaciones entre sí, determinar cómo están reflejados los requisitos y valorar como se ha concebido la estética en el sistema.

Validación de requisitos: permite demostrar que los requerimientos definidos en el sistema son los que realmente quiere el cliente. Garantiza que todos los requerimientos presentes en el documento de especificación sigan los estándares de calidad. Revisa el cumplimiento de las características de la especificación de requisitos.

Gestión de requisitos: Son un conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y los cambios en cualquier momento. Para identificar cada requisito se le asigna un identificador que toma la forma <tipo de requisito><número de requisito>.

El tipo de requisito toma algunos de los siguientes valores: F=requisito funcional, D=requisito de datos, C=requisito de comportamiento, I=requisito de interfaz, y S=requisito de salida. De esta forma, un requisito identificado como F09 indica que se trata de un requisito funcional y que tiene asignado el número nueve dentro de los citados requisitos.

Una vez identificados los requisitos, se desarrollan un conjunto de matrices para dar seguimiento a los mismos. La tabla 2 muestra de forma esquemática este planteamiento. Cada matriz de seguimiento identifica los requisitos relacionados con uno o más aspectos del sistema o su entorno. Entre las posibles matrices de seguimiento se citan las siguientes:

Matriz de seguimiento de características, matriz de seguimiento de orígenes, matriz de seguimiento de dependencias, matriz de seguimiento de subsistemas, matriz de seguimiento de interfaces.

Requisitos	Aspecto específico del sistema o de su entorno							
	A01	A02	A03	A04	A05			Aii
R01			✓		✓			
R02	✓		✓					
R03	✓			✓				✓
R04		✓			✓			
R05	✓	✓		✓				✓
Rnn	✓		✓					

Tabla 2 Matriz de seguimiento genérica.

1.3.12 Ingeniería de requisitos en ISO 9001:2000

La Organización Internacional para la Estandarización es una organización internacional no gubernamental que produce normas internacionales industriales y comerciales. Tienen como objetivo facilitar el comercio, el intercambio de información, y contribuir con unos estándares comunes para el

desarrollo y transferencia de tecnologías. Dentro de la familia ISO se encuentra la norma ISO 9001:2000 que es la encargada de especificar los requisitos de un sistema de gestión de la calidad aplicada a empresas u organizaciones. Esta hace referencia a la Gestión de Requisitos en el proceso relacionado con el cliente, que se proponen varias actividades como:

- Determinar requisitos relacionados con el producto, donde la organización debe determinar: los requisitos que especifica el cliente, incluyendo otros para actividades de entrega y posterior a esta; los que no establece el cliente pero son necesarios para el uso especificado o previsto, cuando sea conocido; los legales y reglamentarios relacionados con el producto; y cualquier requisito adicional que decida incluir la organización.
- Revisión por parte de la organización de los requisitos relacionados con el producto, lo que debe efectuarse antes de obtener un compromiso a facilitar un producto al cliente. Se definen los requisitos del producto; así como deben resolver las diferencias entre los de contrato o pedido y los expresados previamente; y la organización tiene la capacidad para cumplir estos requisitos previamente definidos.
- Mantener registros de los resultados de la revisión y de las acciones originadas por la misma. Cuando el cliente no proporcione una declaración documentada de los requisitos, la organización debe confirmarlos antes de la aceptación. Hay que tener presente que cuando se cambien los requisitos del producto, la organización debe asegurarse de que la documentación pertinente sea modificada y de que el personal correspondiente sea consciente de los requisitos modificados. Hay que saber que en algunas situaciones, tales como las ventas por Internet, no resulta práctico efectuar una revisión formal de cada pedido. En su lugar, la revisión puede cubrir la información pertinente del producto, como son los catálogos o el material publicitario.
- La comunicación con el cliente, donde la organización debe determinar e implementar disposiciones eficaces para la comunicación con los clientes, relativas a la información sobre el producto; las consultas, contratos o atención a pedidos, incluyendo las modificaciones; y la retroalimentación del cliente, con sus quejas incluidas.
- Identificación y trazabilidad: la organización debe identificar el producto por medios adecuados, a través de toda la realización del producto; debe identificar el estado del producto con respecto a los requisitos de seguimiento y medición así como controlar y registrar la identificación única del producto.

En el proceso de seguimiento y medición también propone hacer énfasis en el producto, se plantea que la organización debe medir y hacer un seguimiento de las características del producto para verificar que se cumplen los requisitos del mismo. Esto debe realizarse en las etapas apropiadas del proceso de

realización del producto de acuerdo con las disposiciones planificadas. Debe mantenerse evidencia de la conformidad con los criterios de aceptación, los registros deben indicar la(s) persona(s) que autoriza(n) la liberación del producto. La liberación y la prestación del servicio no deben llevarse a cabo hasta que se hayan completado satisfactoriamente las disposiciones planificadas, a menos que sean aprobados de otra manera por una autoridad pertinente y, cuando corresponda, por el cliente.

1.4 Situación actual del Polo Video y Sonido Digital.

La Universidad de las Ciencias Informáticas (UCI) para lograr una mayor organización en la producción (desarrollo de proyectos productivos) se trazó la estrategia de dividir la producción por facultades y en estas crear los denominados polos productivos que en cada uno de ellos se reúnen un conjunto de proyectos con temas afines. La Facultad 9 desde su creación contó con el proyecto UCITeVe un proyecto de prestación de servicios, en los que se desarrollaba como trabajo fundamental la creación de las teleclases de la universidad, desde la filmación, animación, y edición de las mismas. Aún siendo un proyecto de servicios se desarrollaron algunos proyectos de desarrollo de software tales como Canal ACN versión 1 y el portal de Internos. En el curso 2007 – 2008 se conforman los polos productivos donde UCITeVe pasa a formar parte del Polo Video y Sonido Digital, al polo también se le sumó el proyecto Primicia. Así paulatinamente se fueron creando proyectos tales como Canal Informativo MENPET (TVEnergía), Plataforma de VideoWeb, ACN v 2.0, Plataforma de Transmisión de TV y Radio, Canal Asamblea Nacional VNZ, CEVALE, UCICuest, SGP-DTU, Captura y Catalogación de Medias. Son proyectos pequeños y medianos de corta duración. Dichos proyectos no aplican los modelos de Ingeniería de Requisitos en el desarrollo de software en el Polo de Video y Sonido Digital. No existe una estrategia para la gestión de requisitos.

1.4.1 Situación Problemática

Después de realizar una entrevista en la que se tomó como población los proyectos del Polo Video y Sonido Digital y la muestra fueron los proyectos siguientes: Canal Informativo MENPET (TVEnergía), Plataforma de Transmisión de TV y Radio, CEVALE, UCICuest, SG_DTU, se demuestra que los proyectos que han pasado ya la etapa de requerimientos, además de los que ya son productos, no han realizado de la forma más eficiente el levantamiento de requisitos o la gestión de estos, aunque hayan cumplido con las necesidades de los clientes. Los miembros de los proyectos tienen poco conocimiento sobre el tema de Ingeniería de Requisitos, por lo tanto tampoco conocen de los modelos, herramientas o actividades de la misma.

Proyectos	Tipo de proyecto	Metodología que utiliza	Utilización de modelo para la IR	Utilización de herramientas para la IR
Canal Informativo MENPET (TV Energía)	Desarrollo	RUP	No	No
Plataforma de Transmisión de TV y Radio	Desarrollo	RUP	No	No
CEVALE	Innovación Pedagógica	RUP	No	No
UCICuest	Desarrollo	RUP	No	No
G_DTU	Desarrollo	RUP	No	No

Tabla 3 Resumen de la Entrevista2.

Como se evidencia en la tabla la mayoría de los proyectos son de desarrollo, ninguno utiliza herramientas para la gestión de requisitos, ni optan por modelos para la IR. En el polo no se ha definido un procedimiento para llevar a cabo la ingeniería de requisitos, siendo esta una tarea tan importante.

1.4.2 Análisis de los modelos.

Después de mencionar algunos modelos de IR y analizar la situación problemática del Polo Video y Sonido Digital en cuanto a la gestión de requisitos, se procede a hacer un análisis de dichos modelos para optar por uno de estos de acuerdo a las necesidades de los proyectos de desarrollo de software del PVSD.

Modelos	Cantidad de Actividades	Retroalimentación entre las etapas	Iterativo	Repetición de actividades	Basamento teórico	Realización de forma lineal
Cascada	4	no	no	no	Ninguno	si
Espiral	4	si	si	si	Cascada	no
Phol	4	si	si	si	Espiral	no
Durán	3	si	si	si	Phol	no

Tabla 4 Resumen comparación de Modelos.

Los modelos de IR se han ido perfeccionando o adaptando a las características de cada tipo proyecto en específico. Todos se basan en el modelo madre o modelo abstracto, el modelo en cascada es el más parecido a éste pero tiene como inconveniencia que se muestra muy sencillo y que en la realidad el proceso de IR es más complejo de lo que representa dicho modelo, delimita las fases o actividades y en la disciplina IR no existen etapas delimitadas. Para darle solución a esto surge el modelo en espiral, él cual presenta interactividad entre las etapas y repetición de actividades, hasta que se toma una decisión final (aceptación del documento de especificación de requerimiento), este presenta una dificultad y es la imposibilidad en muchas ocasiones de determinar un punto de terminación porque los requisitos nunca llegarán a ser perfectos. Basado en este patrón surge el modelo de Pohl en 1997, que es iterativo, el orden de las actividades puede ser cualquiera, aunque se sugiere una secuencia en la que los requisitos son elicitados, negociados, especificados y documentados y finalmente validados y verificados. Lo anteriormente planteado tiene sus ventajas, pues no necesariamente se tiene que completar el ciclo, por ejemplo si se encuentra en la etapa de negociación y se da cuenta que debe retornar a la fase de elicitación, puede hacerlo. Bajo esos principios aparece el modelo de Durán en el año 2000 mucho más pequeño y en el que se fusionan actividades, es más sencillo que el modelo de Pohl, solo cuenta con 3 actividades, y sugiere gestionar la mayor cantidad de requisitos en cada iteración intentando alcanzar la situación ideal donde el proceso sea lineal, para realizar la IR de los proyectos del Polo Video y Sonido Digital se necesitaría un modelo como este, porque como anteriormente se menciona son proyectos generalmente pequeños, de corta duración, y se necesita optimizar tiempo, lo que no significa que se haga superficialmente, sino de la forma más sencilla y eficiente posible.

1.5 Herramientas automatizadas para la gestión de requisitos.

En el desarrollo de software se cuenta con una ventaja proporcionada por las herramientas CASE. Las herramientas CASE (Ingeniería del Software Asistida por Computadora) se le conoce a todo aquel software que es usado para ayudar a las actividades del proceso de desarrollo del software, en donde se ubica la ingeniería de requerimientos. Estas herramientas se concentran en capturar requerimientos, administrarlos y producir una especificación de requisitos. Existen variadas herramientas CASE que pueden ser utilizadas por los desarrolladores de software en sus proyectos, y de la forma más conveniente para ellos. Si es importante hacer ver que estas herramientas funcionan como un medio facilitador para agilizar y mejorar los procesos involucrados en todo el ciclo de vida presentado por la IR, y que en conjunto ayudan a la construcción final de un producto de software terminado.

Estas herramientas permiten entre otras cosas tener un mayor control de los proyectos, reducir costos y retrasos en los proyectos, ayudan a determinar la complejidad y los esfuerzos necesarios.

Entre las herramientas más utilizadas en el mundo se encuentran las siguientes:

➤ IBM Rational RequisitePro

Es una herramienta centrada en documentos, que almacena los requisitos asociándolos a documentos (aunque también permite guardarlos directamente en la base de datos).

- Auxilia especialmente en el control de cambio de requisitos, con trazabilidad para especificaciones de software y pruebas.
- Está muy unido a MS Word, es parte de Microsoft Development.
- La herramienta permite el uso de Oracle sobre Unix o Windows como “back-end database” y también soporta SQL Server sobre Windows.

RequisitePro es la herramienta que ofrece Rational Software para tener un mayor control sobre los requerimientos planteados por el usuario y todos aquellos requerimientos técnicos o nuevos requerimientos de usuario que surjan durante el ciclo de vida del proyecto. En RequisitePro los requerimientos se encuentran documentados bajo un esquema organizado de documentos; estos esquemas cumplen completamente con los estándares requeridos por algunas de las instituciones a nivel mundial más reconocidas en el desarrollo de software, tales como: IEEE (Instituto de Ingenieros Eléctricos y Electrónicos), ISO, CMM (Modelo de Capacidad de Madurez) y por el RUP (Proceso Unificado Racional).

Esta herramienta se integra con aplicaciones para la administración de cambios, herramientas de modelado de sistemas y con herramientas de pruebas. Esta integración asegura que los diseñadores conocen los requerimientos del usuario, del sistema y del software en el momento de su desarrollo.

El desarrollo de software es una tarea de equipo, de tal forma, es crítico que todos los miembros del equipo posean un entendimiento compartido de la visión de sus proyectos, metas, especificaciones y

requerimientos; pero, ¿cómo puede conseguirse cuando los equipos se encuentran geográficamente distribuidos y funcionalmente aislados, no pudiendo comunicarse entre sí en tiempo y forma? La solución a esta necesidad es IBM Rational RequisitePro. IBM Rational RequisitePro es una solución fácil de usar, es una herramienta de administración de requerimientos que le permite al equipo crear y compartir sus requerimientos utilizando métodos familiares basados en documentos potenciados por la aplicación de las capacidades de una base de datos, tales como la trazabilidad y análisis de impacto. El resultado es una mejor comunicación y administración de requerimientos con una mayor probabilidad de completar los proyectos en tiempo, dentro del presupuesto y superando las expectativas. Los proyectos exitosos comienzan con una buena administración de requerimientos, cuanto más efectiva sea su ejecución, mayor será el resultado en calidad y satisfacción del cliente.

Según la promoción hecha en Internet mediante la página Web para esta herramienta, algunas de sus *ventajas* son:

- ✓ Un producto potente y fácil de utilizar para la gestión de requisitos y casos de uso que propicia una mejor comunicación, mejoras en el trabajo en equipo y reduce el riesgo de los proyectos.
- ✓ Combina la interfaz conocida y fácil de utilizar de los documentos de Microsoft Word con potentes funciones de base de datos para conseguir la máxima eficacia en análisis y consulta de requisitos.
- ✓ Proporciona a los equipos la posibilidad de comprender el impacto de los cambios.
- ✓ Garantiza que todos los componentes del equipo estarán informados de los requisitos más actuales para asegurar la coherencia.
- ✓ Proporciona acceso basado en Web para los equipos distribuidos.

La ventaja de utilizar herramientas como la de RequisitePro, es que el desarrollo de software se ve beneficiado de muchas maneras, y en el caso de la ingeniería de requerimientos, le ayuda notablemente, como se ha venido hablando la IR constituye una de las etapas más importantes a tomar en cuenta en el ciclo de desarrollo de software, en ella se definen los requerimientos con los que debe de contar el software; e incluso, podría llegar a determinar la viabilidad de implementar ese software no es del todo posible, y poder cancelar a tiempo un desarrollo no productivo (La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software, 2007).

La mayoría de los proyectos de desarrollo de software de la UCI que realizan la gestión de requisitos basándose en la IR utilizan la herramienta RequisitePro.

➤ DOORS

DOORS: es la herramienta de administración de requisitos creada por Quality Systems and Software. Esta herramienta permite capturar, relacionar, analizar y administrar un rango de información para asegurar el cumplimiento del proyecto en materia de requerimientos. Esta herramienta tiene varias *funcionalidades* como son:

- Permite el acceso de un gran número de usuarios concurrentes en la red, manteniendo en línea un gran número de requerimientos así como su información asociada.
- Ayuda al usuario a procesar las solicitudes de cambios de requerimientos en línea.
- Permite realizar cualquier modificación vía remota cuando la base de datos está off-line, incorporando sus actualizaciones a la base de datos maestra. Esto hace más fácil la comunicación del equipo con otras organizaciones, subcontratistas y proveedores.
- A diferencia del resto de las herramientas, considera los requisitos como objetos y los documentos como módulos.
- Tiene una orientación basada en objetos, frente a RequisitePro y Caliber-RM, que manejan solamente requisitos y sus atributos.
- Es una herramienta para organizaciones grandes que necesitan controlar complejos conjuntos de usuarios y requisitos de sistemas con una completa trazabilidad (Bacallao, y otros, 2007).

Beneficios de DOORS

Entre los beneficios de door se pueden mencionar los siguientes:

- Permite analizar y comprar los requerimientos.
- Permite clasificar los requerimientos.
- Identificación de Inconsistencias.
- Permite compartir requerimientos entre proyectos.
- Permite crear relaciones entre requerimientos.
- Envía una notificación vía email cuando los cambios son revisados.
- Permite visualizar los cambios pendientes de otros usuarios para anticipar el impacto que ocasionará.
- Despliega estadísticas y métricas a través de gráficas.
- Los documentos están escritos en lenguaje claro, lo que proporciona una comprensión inmediata de cada requerimiento.
- Permite importar sus documentos a formatos de herramientas de Microsoft Office, RTF, HTML, texto, entre otros.

➤ IRQA

IRQA: Integral Requisite Analyzer (IRqA) es una herramienta de soporte a la fase de captura y análisis de requisitos, dentro del ciclo de vida de un desarrollo de software. Irqa es una herramienta de ingeniería de requisitos especialmente diseñada para soportar el proceso completo. En esta herramienta el ciclo de especificación completo incluye la captura de requisitos, análisis, especificación de sistema, validación y la organización de requisitos es soportada por modelos estándares. La aplicación práctica de Irqa en el desarrollo de proyectos software se basa en una metodología que sistematiza los procesos y actividades

de la IR. El objetivo básico que se pretende conseguir con la aplicación de Irqa en un proyecto es construir una especificación de requisitos correcta, consistente y completa. Esta especificación debe incluir tanto el conjunto de requisitos que expresa las necesidades de los usuarios, como la especificación de la solución propuesta, describiendo su comportamiento por medio de sus interacciones con el exterior. (Bacallao, y otros, 2007)

Irqa proporciona las siguientes *funcionalidades* para ayudar a las empresas a conseguir la máxima eficiencia en la gestión de las especificaciones de requisitos:

Captura de requisitos

- Captura automática de requisitos desde MS Word, MS Excel y XML.
- Integración con herramientas de gestión de configuración (MS Visual SourceSafe, Borland StarTeam).
- Jerarquía de requisitos.

Gestión de requisitos

- Búsqueda y filtros avanzados.
- Atributos definidos por el usuario.
- Vistas definibles por el usuario para una gestión rápida y eficaz de los elementos.

Análisis de requisitos

- Construcción del modelo de dominio del problema, conceptos de negocio y facetas del dominio.

Representación gráfica del modelo de conceptos:

- Diagrama de clases (UML).
- Diagrama Entidad-Relación.
- Analizador automático.
- Relación entre requisitos configurables por el usuario (IRQA).

Beneficios de Irqa

- Permite a los equipos construir especificaciones de requisitos de gran calidad: los usuarios pueden capturar y gestionar los requisitos, analizarlos con respecto al dominio del negocio en el que se generaron y representar gráficamente el sistema.
- Permite una gestión real y potente de los requisitos.
- Mantiene la trazabilidad a lo largo de las distintas etapas del ciclo de vida del desarrollo de los sistemas.
- Analiza el impacto de un cambio en cualquiera de los elementos de la especificación.
- Proporciona funcionalidades gráficas para la organización de la especificación garantizando la calidad de la misma.

- Ayuda a sistematizar el proceso de Ingeniería de Requisitos, convirtiendo el proceso de desarrollo de sistemas en un proceso optimizado, más eficiente y productivo.

➤ Caliber-RM

Caliber-RM: Herramienta para administración de requisitos, permite que sean definidas matrices y rastreabilidad. Ofrece como recurso la priorización y acompañamiento de los requisitos durante todo el ciclo de vida de la aplicación. Caliber-RM es para sistemas grandes y complejos y proporciona una base de datos de requisitos con trazabilidad. La compañía ve a los requisitos como parte del proceso de gestión de la calidad del software, el cual es considerado también, las pruebas y el trazado de defectos. Caliber maneja referencia de documentos, responsabilidad de usuario, trazabilidad, prioridad y estado entre otras características. Caliber-RM™ 2005 es una solución para la gestión de requisitos dentro del proceso de entrega del software. Diseñado para capturar y gestionar los requisitos de negocio, técnicos, funcional, y requisitos operacionales. Permite la colaboración eficaz a través de la organización destinada a entregar los proyectos cumpliendo el tiempo, el presupuesto, y las especificaciones (Bacallao, y otros, 2007).

Beneficios de CaliberRM

La herramienta CaliberRM posee una variedad de beneficios de los que se hace mención a continuación:

- Ofrece un apoyo eficaz a las organizaciones distribuidas y asiste a los equipos en la gestión de las expectativas y del ámbito de los proyectos, en iniciativas de cualquier tamaño.
- Automatiza la gestión de requisitos para facilitar la asignación de prioridades a tareas y actualizaciones del proyecto, la simplificación de la comunicación y la gestión del ámbito del proyecto.
- Permite gestionar requisitos para una amplia gama de proyectos, ofreciendo una verdadera ventaja competitiva a las organizaciones en numerosos mercados verticales.
- Ayuda a las organizaciones a asegurar que las soluciones de software resuelven las expectativas proporcionando un repositorio central para mantener todos los requisitos del proyecto en una sola localización.
- Mantiene a todos los miembros del equipo sincronizados - incluso a través de ambientes distribuidos - proporcionando el acceso fácil a una vista centralizada, exacta y actualizada de todos los requisitos.
- Hace más fácil manejar, hacer seguimiento y comunicar los cambios de los requisitos a través de todo el Equipo de desarrollo del software.
- Esta herramienta tiene un conjunto de funciones claves como son: Estimación de gestión de requisitos completo, procesos de gestión de requisitos personalizable, estimación basada en los requerimientos, gestión integrada del ciclo de vida de la aplicación y análisis de impacto.

Funciones clave de Caliber RM

Sistema de gestión de requisitos completo: CaliberRM proporciona un repositorio central seguro para gestionar los requerimientos de los proyectos a través del ciclo de vida de la aplicación, mejorando la comunicación a través de todos los participantes para establecer una visión consistente desde el principio del proyecto. Las capacidades sin par de edición aseguran que los usuarios puedan escribir requisitos usando su formato preferido incluyendo casos de uso, escenarios, definiciones funcionales, y documentos del diseño.

A continuación se analiza la herramienta CaliberRM de acuerdo a determinadas características.

Procesos de gestión de requisitos personalizable: CaliberRM se puede modificar fácilmente para soportar procesos de gestión de requisitos particulares, asegurando que las organizaciones y los equipos conserven control y trabajen en la manera que desean trabajar.

Estimación basada en los requerimientos: Las potentes capacidades de estimación basadas en los requerimientos ayudan a los gestores a planificar el alcance del proyecto, agenda, y recursos a través del ciclo de vida del desarrollo del software con gran exactitud. Enlazando el alcance del proyecto, agenda, y coste con la asignación de recursos y la gestión de riesgo, cuando una variable cambia, el impacto en otras variables se puede determinar inmediatamente.

Gestión integrada del ciclo de vida de la aplicación: La trazabilidad de los requisitos a través del proceso de desarrollo resulta ser el mejor control del proyecto. La arquitectura abierta de CaliberRM permite enlazar directamente los requerimientos con una variedad de aplicaciones - tales como gestiones de configuración de software. Esto mantiene a todos los miembros del equipo enfocados y actualizados para acelerar la producción.

Análisis de impacto: La visualización de la trazabilidad ayuda a los usuarios a evaluar el alcance de los cambios de requisitos. El rastro revela cómo los cambios afectan a los requisitos, tareas, pruebas y/o código de fuente, permitiendo el análisis en tiempo real.

En resumen existe una variada cantidad de herramientas de apoyo para la gestión de requisitos cada una con características específicas y todas con numerosas ventajas. Pero como desventaja principal son herramientas realizadas para que se utilicen en sistemas operativos propietarios, y además las propias herramientas son propietarias.

1.5.1 Herramientas automatizadas libres para la Gestión de Requisitos

Además de las herramientas comerciales habituales, en el campo de la Gestión de Requisitos poco a poco han ido surgiendo una serie de herramientas de libre distribución y Open Source, bastante elaboradas. La más conocida es OSRMT (Open Source Requirements Management Tool), cuyo nombre lo dice todo. Es independiente de la plataforma, pues corre sobre Java. Es una herramienta diseñada para

dar cobertura a todo el ciclo de vida del desarrollo de software (SDLC). En este ciclo de vida se incluye el análisis de requisitos, diseño, implementación y pruebas. Entre las características principales se encuentran que: dispone de control de versiones, permite definir requerimientos derivados, es posible definir atributos para los requisitos como son el riesgo, esfuerzo, etc, se puede representar tanto casos de uso como casos de prueba, su Web Oficial es sourceforge.net/projects/osrmt, la última versión encontrada es la versión 1.50, el tipo de licencia es Open Source. GNU General Public License (GPL). (Sánchez, 2008)

Otras herramientas libres son las siguientes:

➤ REM

REM (Requisite Management) diseñada para soportar la fase de definición e ingeniería de requisitos de un proyecto de desarrollo software. Permite reflejar la correcta captura de requisitos según lo indicado por el cliente obteniendo un documento ordenado y sin perder flexibilidad a la hora de definir el formato estándar de dicho documento. La forma de trabajo de REM consiste en la elaboración de un documento con los requisitos que se quieran introducir. El documento a generar se puede indicar que sea de varios tipos: documento de requisitos del sistema, documento de análisis del sistema, registro de conflictos y defectos, registro de peticiones de cambio.

➤ ReqMan

ReqMan: Este proyecto proporciona una herramienta para la captura del requisito, y el análisis de requisito en una disposición multiusos, posee licencia GPL. Corre sobre Linux.

➤ Jeremia

Jeremia: Se trata de una aplicación cliente exclusivamente, con lo cual no cubre la posibilidad de trabajar en equipo.

➤ Rambutan

Rambutan: Una herramienta creada en una universidad en Indonesia. Tampoco permite trabajar en equipo. La portabilidad entre las plataformas de escritorio es apoyado por la edición estándar de Java 2: Windows, Macintosh, y Unix (Solaris incluyendo y linux). Diseño orientado a objeto, patrón-basado extensible. Entre los requisitos de Sistema de escritorio se puede mencionar que: por lo menos se necesita una computadora Pentium II 366Mhz con una memoria de 128MB.

Lo anteriormente expuesto demuestra que existen pocas herramientas libres y que se puedan utilizar en sistemas operativos libres, es solo el comienzo, pero aún así las que existen a juzgar por las características hay unas que son muy buenas, y otras muy experimentales.

1.6 Conclusiones del capítulo 1.

En el capítulo se mencionó un conjunto de conceptos para el entendimiento de la investigación, además de exponer una visión de la situación actual del Polo Video y Sonido Digital (PVSD), se mostró la diversidad de modelos, herramientas de gestión de requisitos y técnicas para llevar a cabo las actividades que proponen los modelos para establecer un procedimiento de IR. Después de realizar un análisis de los modelos se decide proponer el modelo de Durán para utilizarlo en la propuesta a establecer en los proyectos del Polo Video y Sonido Digital por ser el más propicio para los mismos.

CAPÍTULO 2: Solución Propuesta.

De acuerdo a las necesidades de los proyectos del PVSD se propone utilizar el modelo de Durán, este es un modelo realizado por el Dr. Amador Durán Toro en su tesis doctoral “Un entorno metodológico de ingeniería de requisitos para sistemas de información”, bajo la tutoría del Dr. Catedrático Miguel Toro Bonilla de la Universidad de Sevilla.

En este capítulo se explica brevemente el procedimiento a llevar a cabo en los proyectos del Polo Video y Sonido Digital (PVSD), especificando el modelo propuesto y las herramientas y técnicas a utilizar en cada una de las actividades del modelo. Y se aplica a uno de los proyectos, el cual será el proyecto VideoWeb de dicho polo.

2.1 Breve descripción del modelo propuesto

El modelo de Durán como se menciona en el capítulo anterior se basa en el modelo de Pohl y consta de tres actividades principales, elicitación, análisis y validación, y su principal característica es la iteratividad. El modelo describe un entorno metodológico para la ingeniería de requisitos, enuncia una metodología para:

La elicitación de requisitos de sistemas de información, incluyendo las tareas a realizar, los productos a obtener y las técnicas a emplear, principalmente plantillas y patrones de requisitos, así como la posibilidad de introducir la reutilización en el proceso.

El análisis de requisitos de sistemas de información, incluyendo las tareas a realizar, los productos a obtener y las técnicas a emplear, basadas en el estándar UML y con relaciones de rastreabilidad hacia los productos de la actividad anterior que facilita la reutilización de elementos complejos.

La validación de requisitos de sistemas de información, incluyendo las tareas a realizar, los productos a obtener y las técnicas a emplear.

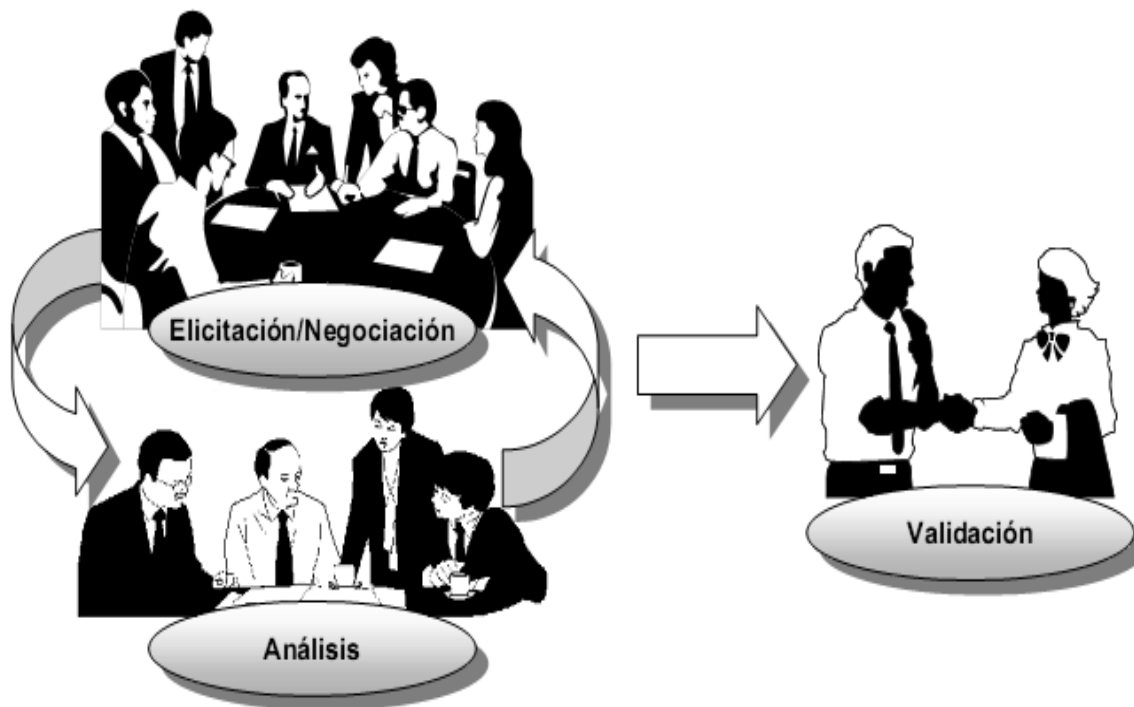


Figura 6: Actividades del Modelo de Durán.

2.2 Tareas recomendadas en cada una de las actividades del modelo propuesto.

Las tareas recomendadas en la actividad de Elicitación se mencionan a continuación.

Tarea 1: Obtener información sobre el dominio del problema y el sistema actual

Tarea 2: Preparar y realizar las reuniones de elicitación/negociación

Tarea 3: Identificar/revisar los objetivos del sistema

Tarea 4: Identificar/revisar los requisitos de almacenamiento de información

Tarea 5: Identificar/revisar los requisitos funcionales

Tarea 6: Identificar/revisar los requisitos no funcionales

Tarea 7: Priorizar objetivos y requisitos

El orden recomendado de realización para estas tareas es: 1... 7, aunque las tareas 4, 5, y 6 pueden realizarse simultáneamente o en cualquier orden que se considere oportuno. La tarea 1 es opcional y depende del conocimiento previo que tenga el equipo de desarrollo sobre el dominio del problema y el sistema actual.

Las tareas recomendadas en la actividad de Análisis son las siguientes.

Tarea 1: Analizar los requisitos de almacenamiento de información

Tarea 2: Analizar los requisitos funcionales

Tarea 3: Analizar los requisitos no funcionales

Tarea 4: Desarrollar prototipos

El orden recomendado de realización para estas tareas es: 1... 4, aunque las tareas 1, 2 y 3 pueden realizarse simultáneamente y el tipo de sistema puede determinar un orden u otro. La tarea 4 es opcional, dependiendo del ciclo de vida adoptado en el desarrollo.

Las tareas que se recomiendan realizar en el modelo propuesto en la actividad de Validación son las siguientes.

Tarea 1: Validar los requisitos de almacenamiento de información y funcionales

Tarea 2: Validar los requisitos no funcionales

Tarea 3: Cerrar la versión de los requisitos

El orden recomendado de realización para estas tareas es: 1... 3, aunque las tareas 1 y 2 pueden realizarse simultáneamente y el tipo de sistema puede determinar un orden u otro (Duran, 2001).

Para una mejor comprensión del modelo transmitirse al documento de la tesis doctoral de Amador Durán disponible en: <http://dialnet.unirioja.es/servlet/tesis?codigo=12692>.

2.3 Adaptaciones del modelo de Durán para los proyectos del PVSD.

Los proyectos de desarrollo de software necesitan minimizar tiempo, algunos desarrolladores plantean que los modelos llevan consigo pérdida de tiempo a la hora de llenar una determinada cantidad de plantillas, que forman parte de la documentación de la gestión de requisitos.

En aras de minimizar la documentación y hacer el proceso de gestión de requisitos más rápido, lo que no quiere decir que se haga de manera superficial, se decide resumir un poco el modelo de Durán. Se seleccionan las tareas fundamentales y de las técnicas propuestas se toman las que son necesarias utilizar.

Tareas generales

Identificar Requisitos

Analizar Requisitos

Gestionar Riesgos

Validar Requisitos

Aplicar métricas

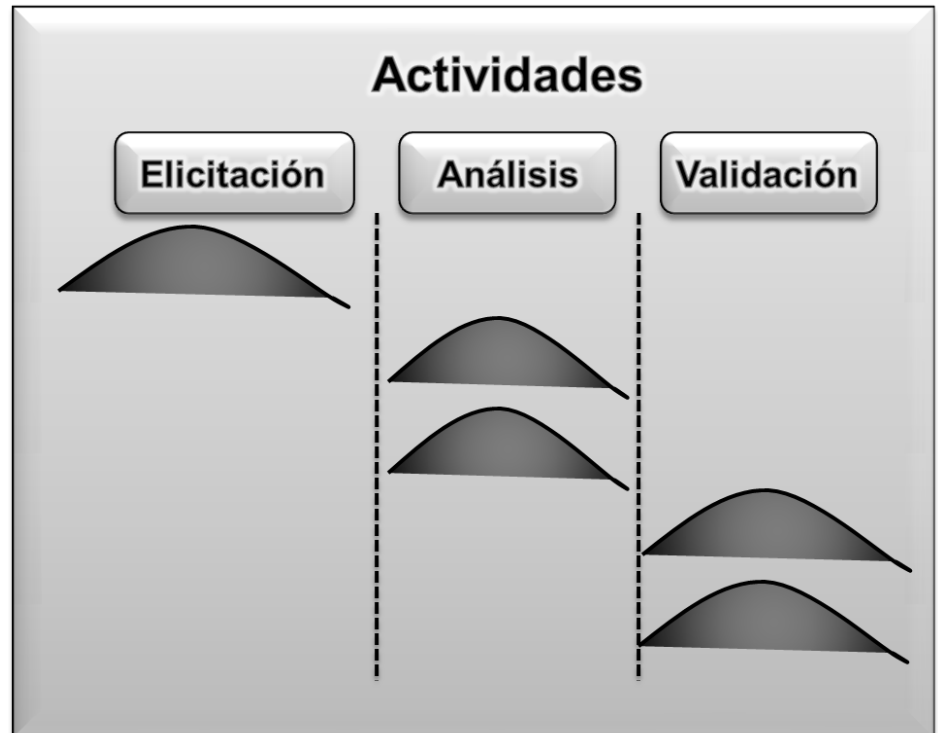


Figura 7 Aparición de las tareas generales en cada una de las actividades del modelo (Elaboración propia).

En la figura 7 se muestra donde se realizan cada una de las tareas en las actividades del modelo, entre las modificaciones se encuentran, la gestión de riesgos y la aplicación de métricas.

Cada una de las actividades que se realizan en el modelo adaptado se explica a continuación.

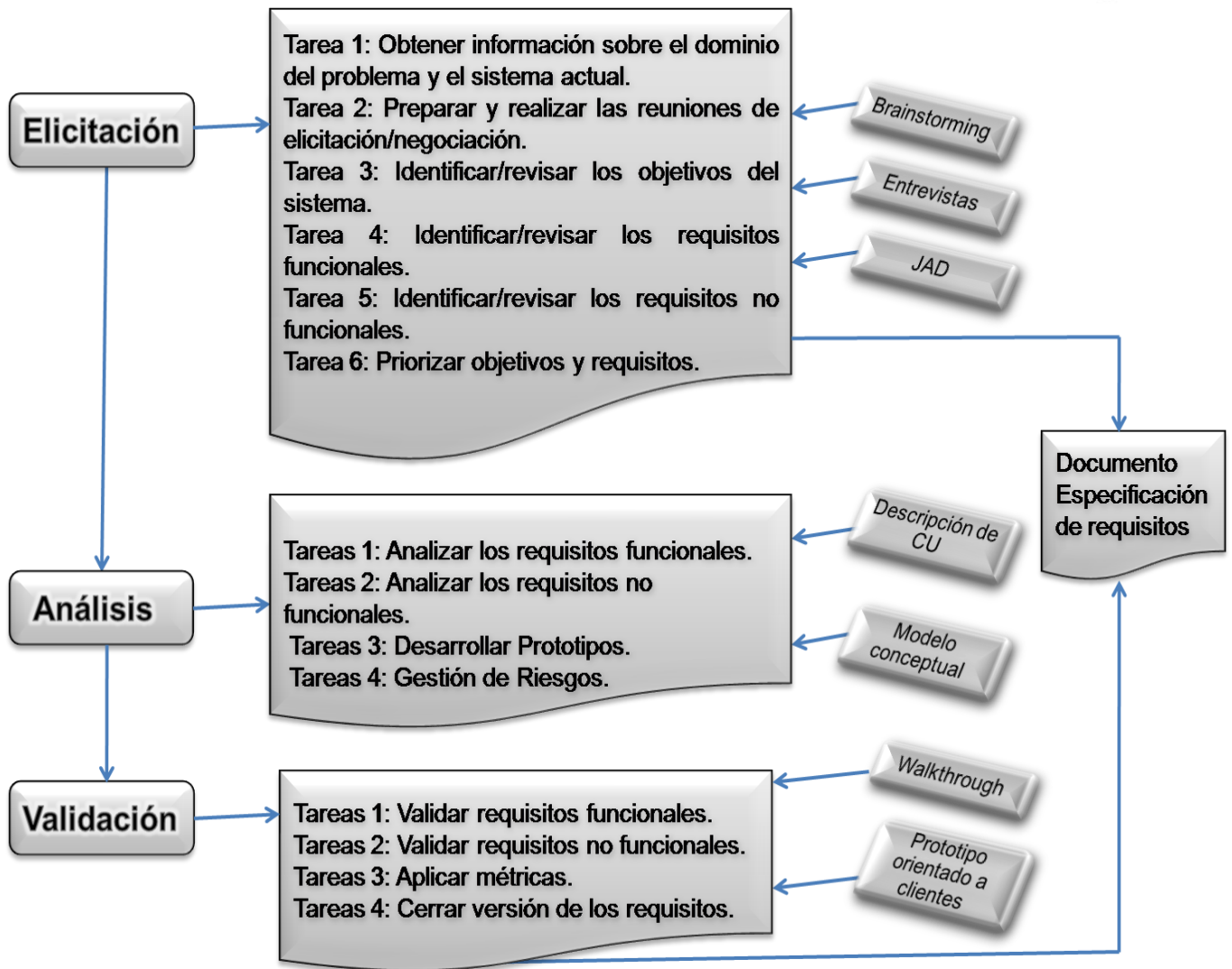


Figura 8 Representación del modelo modificado (Elaboración propia).

2.3.1 Elicitación de requisitos.

La elicitación de requisitos no se ha considerado parte del ciclo de vida de desarrollo de software hasta hace relativamente pocos años. Se solía asumir que el cliente proporcionaba los requisitos, de forma que el ciclo de vida comenzaba siempre por el análisis de unos requisitos ya dados, pues las actividades de elicitación y validación no se consideraban necesarias. Cuando se ha detectado que los problemas en los requisitos son uno de los principales factores de los fracasos de los proyectos software, es cuando se le ha comenzado a dar importancia al proceso de obtención de esos requisitos.

Tareas recomendadas en la actividad de Elicitación

Tarea 1: Obtener información sobre el dominio del problema y el sistema actual.

Tarea 2: Preparar y realizar las reuniones de elicitación/negociación.

Tarea 3: Identificar/revisar los objetivos del sistema.

Tarea 4: Identificar/revisar los requisitos funcionales.

Tarea 5: Identificar/revisar los requisitos no funcionales.

Tarea 6: Priorizar objetivos y requisitos

Técnicas recomendadas en la actividad de Elicitación

Entrevistas

Joint Application Development

Brainstorming

Plantillas recomendadas a utilizar en la actividad de Elicitación

Plantilla para objetivos

Plantilla de Especificación de requisitos (Producto entregable).

2.3.2 Análisis de requisitos.

El análisis de requisitos se considera como una actividad dentro de la fase de ingeniería de requisitos cuyo objetivo principal es descubrir conflictos en los requisitos—C elicitados previamente.

Tareas recomendadas en la actividad de Análisis.

Tareas 1: Analizar los requisitos funcionales.

Tareas 2: Analizar los requisitos no funcionales.

Tareas 3: Desarrollar Prototipos.

Tareas 4: Gestión de Riesgos.

Técnicas recomendadas en la actividad de Análisis.

Descripción de CU.

Construir un modelo conceptual a partir de los objetivos y requisitos documentados, teniendo en cuenta también el modelo del negocio, el glosario de términos, etc.

2.3.3 Validación de requisitos.

La validación de requisitos se considera como la actividad de la ingeniería de requisitos en la que clientes y usuarios, junto con la ayuda de los ingenieros de requisitos, revisan los productos obtenidos durante las actividades anteriores para confirmar que realmente reflejan sus necesidades y que definen el producto deseado. La validación de requisitos es otra de las actividades de la ingeniería de requisitos, que junto con la elicitación, han recibido tradicionalmente poca atención. La razón, al igual que en el caso de la elicitación, ha sido la suposición de que los requisitos eran proporcionados directamente por el cliente, con

lo que, implícitamente, se asumían validados. La validación de requerimientos es importante pues de ella depende que no existan elevados costos de mantenimiento para el software desarrollado.

Tareas recomendadas en la actividad de Validación.

Tareas 1: Validar requisitos funcionales.

Tareas 2: Validar requisitos no funcionales.

Tareas 3: Aplicar métricas.

Tareas 4: Cerrar versión de los requisitos.

Técnicas recomendadas en la actividad de Validación.

Walkthroughs

Prototipos orientados a clientes/usuarios.

Plantillas recomendadas a utilizar en la actividad de Validación.

Plantilla de Especificación de requisitos (Producto entregable, versión mejorada).

2.3.4 Gestión de riesgos en la Ingeniería de Requisitos.

El modelo de Durán no tiene en cuenta la gestión de riesgo, por lo que entre las modificaciones del modelo se encuentra incluir esta etapa porque la identificación y gestión de los riesgos asociados a los requisitos del software, individuales y a grupos de ellos, desde la fase de ingeniería de requisitos puede permitir minimizarlos, evadirlos y controlarlos.

La gestión de riesgos se llevará a cabo en la actividad de análisis, se realizará como una tarea más de esta actividad.

Para lograr producir aquello que el cliente requiere, en el plazo solicitado y ajustados al presupuesto asignado, se necesita desarrollar un proceso que incluya desde la etapa más temprana la gestión de los riesgos asociados a los requisitos, de forma que se contribuya al mejoramiento gradual del proceso de desarrollo y la gestión de un proyecto de software que logre la satisfacción del cliente en estas organizaciones. La gestión de riesgos en el ámbito del software procura formalizar conocimiento orientado a la minimización o evitación de riesgos en proyectos de desarrollo de software, mediante la generación de principios y buenas prácticas de aplicación realista.

Existen dos tipos de estrategias frente al riesgo, por un lado están las reactivas, cuyo método es evaluar las consecuencias del riesgo cuando este ya se ha producido (ya no es un riesgo) y actuar en consecuencia. Este tipo de estrategias acarrea consecuencias negativas, al poner el proyecto en peligro. Y por el otro las preactivas, que aplican el método de evaluación previa y sistemática de los riesgos y sus

posibles consecuencias, a la par que conforman planes de contingencias para de evitar y minimizar las consecuencias. (Fernández, 2005)

De acuerdo con (Pressman, 2005) la administración o gestión de riesgos es un proceso iterativo que se aplica durante todo el proyecto y se desarrolla en cuatro etapas. Los resultados de la administración de riesgos deben ser documentados en un plan de administración de riesgos. La figura 7 refleja el procedimiento.

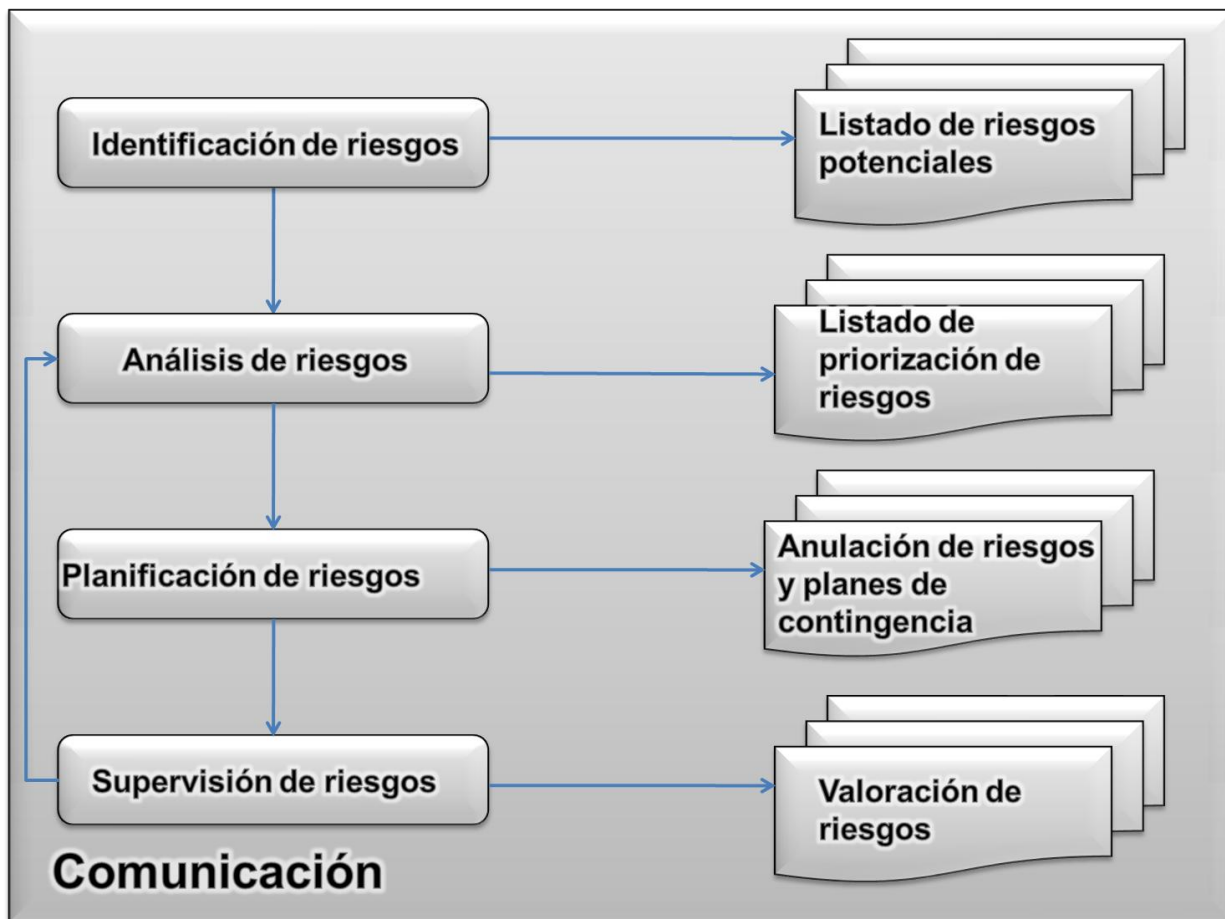


Figura 9 Procedimiento de gestión de riesgo (Elaboración propia).

Propuesta de la Actividad de Gestión de Riesgo en la etapa de IR

Un riesgo es aquel factor que influye negativamente en el éxito del proyecto. El riesgo en un proyecto de desarrollo de software incluye componentes técnicos y de conocimiento del mismo.

Si a la vez que se inicia la actividad de elicitación de los requisitos del software a construir, se inicia la identificación de los riesgos asociados a los requisitos individuales y a grupos de ellos, será posible gestionarlos tempranamente para minimizarlos, evadirlos y controlarlos. El jefe o administrador de

proyectos anticipa riesgos que pueden afectar al desarrollo o a la calidad de los requisitos y emprende acciones para evitarlos.

Esta actividad garantiza que, desde el inicio del proceso de desarrollo del software, se realicen las tareas encaminadas a garantizar la calidad del producto.

Basado en el modelo dado por (Pressman, 2005), descrito anteriormente y representado en la figura 7, se define el procedimiento siguiente:

Paso 1: Identificación de riesgos. Problemas potenciales que pueden ocurrir en el proceso de IR o en los requisitos, o en la Especificación de los Requisitos del Software (ERS), como de presupuesto, de personal, del usuario, de organización, técnicos, de comunicación u otros. Debe comenzar con el análisis de los riesgos genéricos, que constituyen una amenaza potencial para todos los proyectos de software, que puedan estar presentes en el proyecto en curso. Después se deben identificar los riesgos específicos, que implican un conocimiento profundo del proyecto, y están relacionados con el entorno de desarrollo, la tecnología, la experiencia y el tamaño del equipo.

De no gestionarse el riesgo en esta etapa inicial, puede hacerse difícil su control efectivo una vez iniciado el proceso de desarrollo. Desequilibrios en el comportamiento de las características de calidad de los requisitos se traducen en que la aparición de riesgos haga vulnerable el proyecto y no práctica la aplicación de cualquier plan de calidad, razón por la cual conviene su detección y tratamiento temprano. En esta situación, se recomienda la redefinición de los requisitos afectados (o la ERS), que es posible por iteraciones en este procedimiento.

En este procedimiento se establece definir y listar los riesgos que pueden afectar al propio proceso de ingeniería de requisitos como aparece en las tablas 5, 6 y 7.

Riesgo	Tipo de riesgo	Descripción
Rotación de personal	Proyecto, producto y negocio	Personal con experiencia abandona el proyecto antes de que finalice
Cambios de requisitos	Proyecto y producto	Existencia de más cambios de requerimientos de los previstos inicialmente
Retrasos en la especificación	Proyecto y producto	Retrasos en las especificaciones de interfaces esenciales
Subestimación del tamaño	Proyecto y producto	El tamaño del requisito (, del proceso de IR) se ha subestimado
Bajo rendimiento de la herramienta CASE	Producto	Las herramientas CASE que ayudan al proyecto no tienen el rendimiento y las funcionalidades esperadas

Tabla 5 Lista de Riesgos y clasificación.

ID Requisito	Tipo de Riesgo	Riesgos
R1	De personal	El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad del requisito
		Miembros del equipo no disponibles en momentos críticos
	De requisitos	Cambios de requisitos que precisan modificaciones en el diseño
		Los clientes no comprenden el impacto de los cambios en los requerimientos
	De estimación	El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado
	De comunicación	El cliente no pueda participar en revisiones y en reuniones

Tabla 6 Riesgos por requisitos

Tipo de riesgo	Posibles riesgos
Personal	Imposible contratar personal con los conocimientos requeridos.
Organizativos	La organización se reestructura y una nueva administración se responsabiliza del proyecto.
Herramientas	Las distintas herramientas CASE no están disponibles
Requerimientos	Cambios de requerimientos que precisan modificaciones en el diseño.
Estimación	El tamaño del sistema a desarrollar está subestimado.

Tabla 7 Riesgos por tipos.

Algunos riesgos que pueden afectar el desarrollo del proceso de IR son:

- ✓ Sobrepasar los límites de los recursos asignados.
- ✓ Finalización fuera de plazos originales (a veces ni se finaliza).
- ✓ Pobre o descontrolada gestión de los requisitos.
- ✓ Incompatibilidad con el entorno.
- ✓ Riesgo más grave: que no se comprendan y no se satisfagan las necesidades de los usuarios.
- ✓ Problemas en la comunicación entre clientes y proveedores, entre usuarios, u otros grupos.

Paso 2: Evaluación de los riesgos. Determinar en qué indicador se verá reflejado que un problema se presente, se deben establecer puntos de referencia para cada riesgo, que permita decidir si el riesgo, según su prioridad de atención, se sale del manejo aceptable.

Proyección de los riesgos. Consiste en determinar la probabilidad de que un riesgo ocurra y las consecuencias que puede tener, por ejemplo: incremento de costos, cancelación del proyecto, insatisfacción del cliente. Implica ordenar la lista de riesgos teniendo en cuenta la probabilidad de que ocurra y el impacto de cada riesgo. Se asigna el nivel de probabilidad, que puede ser alta, media o baja. Se valora el impacto (consecuencias) en cuanto al alcance (cuánto se afecta) y la duración (por cuánto tiempo se manifiesta).

En el análisis de riesgos se considera cada riesgo por separado y se valora en intervalos su probabilidad e impacto:

- probabilidad del riesgo valorada como *muy bajo* (<10%), *bajo* (10-25%), *moderado* (25-50%), *alto* (50-75%) o *muy alto* (>75%)
- efectos del riesgo valorados como *catastrófico*, *serio*, *tolerable* o *insignificante*

El resultado se registra en una tabla ordenada por la probabilidad o por el efecto del riesgo. Se decide del total, cuáles son los más importantes, considerados entonces como riesgos claves durante el proyecto debe ser un número manejable.

Por ejemplo, todos los serios o catastróficos con cualquier probabilidad. Obsérvese la tabla 8.

Riesgo	Probabilidad	Efectos
Problemas financieros de la organización reducen el presupuesto del proyecto	baja	catastrófico
Imposible contratar personal con los conocimientos requeridos	alta	catastrófico
Personal clave enfermo o no disponible en momentos críticos	moderada	serio
Cambios de requerimientos que precisan modificaciones en la codificación	moderada	serio
El tiempo requerido para desarrollar el proceso de IR está subestimado	alta	serio
Los clientes no comprenden el impacto de los cambios en los requerimientos	moderada	tolerable

Tabla 8 Riesgos ordenados por efectos.

Paso 3: Planificación de riesgos. Este paso tiene como objetivo desarrollar una estrategia para tratar los riesgos. Si el equipo de trabajo adopta un enfoque proactivo frente al riesgo, evitarlo será siempre la mejor estrategia. Esto se consigue desarrollando los planes de reducción del riesgo y de contingencia.

En la planificación de riesgos se considera cada uno de los riesgos claves identificados y las estrategias para administrarlos, que vendrán dadas por el juicio y la experiencia del administrador del proyecto.

Las estrategias de anulación intentan reducir la probabilidad de que surja el riesgo, las estrategias de disminución: intentan reducir el impacto del riesgo. Los planes de contingencia se elaboran para estar preparados por si el riesgo ocurre poder actuar con una estrategia determinada. Obsérvese la tabla 9.

Riesgo	Estrategia
Problemas financieros de la organización	Preparar un documento breve para la dirección de la empresa que muestra que el proyecto hace contribuciones muy importantes a las metas del negocio
Problemas de reclutamiento	Organizar cursos de capacitación para el personal ya existente, investigar la posibilidad de contratar en otras regiones del país
Enfermedad del personal	reorganizar el equipo de tal forma que se solapen el trabajo y los miembros comprendan el trabajo de los demás
Cambios en los requisitos	Rastrear la información para valorar el impacto de los requerimientos, maximizar la información oculta en ellos
Tiempo de IR subestimado	Alertar al cliente de las dificultades potenciales y las posibilidades de retraso

Tabla 9 Estrategias por riesgos.

Paso 4: Supervisión de los riesgos. Consiste en hacer el plan de supervisión, dado el caso de que se acepte continuar con el proyecto. Indicar que acciones y decisiones se tomarán ante un problema que ya ha sido identificado, proyectado y evaluado.

La supervisión de riesgos valora cada uno de los riesgos identificados para decidir si es más o menos probable y cuándo han cambiado sus posibles efectos. Hay que controlar factores que pueden indicar cambios en la probabilidad y el impacto. Obsérvese la tabla 10.

Tipo de riesgo	Indicadores potenciales
Tecnología	Entrega retrasada del hardware. Existencia de informes sobre problemas tecnológicos.
Personal	Baja moral del personal, malas relaciones entre miembros del equipo, plazas vacantes,
Organizacional	Rumores. Falta de iniciativa de la dirección.
Herramientas	Rechazo de los miembros del equipo a utilizar herramientas. Quejas sobre las CASE
Requisitos	Peticiones de muchos cambios en los requisitos. Quejas del cliente
Estimación	Fracaso en el cumplimiento de los tiempos planificados.

Tabla 10 Indicadores potenciales por riesgos.

2.3.5 Métricas en la Gestión de Requisitos.

Existen un conjunto de métricas que se pueden utilizar para la validación de los requisitos, a continuación se explican los conceptos fundamentales sobre métricas y algunas de ellas las cuales se utilizarán en la validación de los requisitos. La aplicación de métricas se realizará como una tarea más de la actividad de validación.

¿Qué es una métrica?

Una métrica es la variable o medida de ciertos aspectos cuantitativos de un sistema los cuales serían el alcance, el tamaño, el coste, los riesgos, el esfuerzo, etc.

¿Qué es una métrica del software?

“Una métrica software es un atributo del entorno de desarrollo del software, derivada de la medida de los atributos de ciertos componentes del software”. Un atributo puede ser una cualidad, una propiedad o una característica de un objeto. En el entorno de desarrollo del software, el tamaño, el coste y el esfuerzo son los atributos del proyecto software.

El estándar IEEE 610.12 expresa que: “Una métrica del software es una medida cuantitativa del grado en el que un sistema, componente o proceso dispone de un atributo dado”

Desde otro punto de vista una métrica de software es una correspondencia entre uno o más atributos del entorno de desarrollo del software, y cualquier otro atributo.

¿Qué es medir?

Proceso a través del cual se asigna o hace corresponder un número a los atributos de los objetos. Estos números son asignados por reglas claras y bien definidas. Hay que disponer de una descripción precisa de los atributos de los objetos que se pretenden captar numéricamente.

¿Qué es medir el software?

Proceso a través del cual se puede cuantificar el software, medir el conjunto de recursos involucrados en el ciclo de desarrollo y medir también el propio proceso de desarrollo. Incluye elementos que son directamente medibles (como las líneas de código), y otros que son calculados a través de fórmulas o ecuaciones (como el esfuerzo de desarrollo o el coste del proyecto).

Las métricas del software se pueden clasificar de disímiles formas, estas pueden ser: directas o indirectas, primarias o secundarias, internas o externas, públicas o privadas, simples o complejas, primitivas o calculadas.

Métricas que se pueden utilizar en el proceso de gestión de requisitos:

Porcentaje de requisitos que cumplen con los criterios de evaluación y aceptación establecidos.

Esta métrica permite conocer el porcentaje de requisitos que cumplen con los criterios de evaluación y aceptación establecidos. El por ciento de RCC se puede calcular de la siguiente forma:

$$RCC (\%) = (RCC \times 100) \div TR$$

Donde:

TR: es el total de Requisitos del Sistema.

RCC: es la cantidad de requisitos que cumplen con los criterios de evaluación y aceptación.

Porcentaje de requisitos que satisfacen las necesidades del cliente.

Calcula el por ciento de requisitos que cumplen con las necesidades del cliente. La métrica se calcula de la siguiente forma:

$$RCNC (\%) = (RCNC \times 100) \div TR$$

Donde:

TR: Total de Requisitos.

RCNC: Requisitos que cumplen con las necesidades del cliente.

Cantidad de proveedores de requisitos seleccionados

Permite calcular el total de cambios que se realizan en los requisitos.

$$TCRR = CRR_1 + CRR_2 + CRR_3 + \dots + CRR_n$$

Donde:

CRR1: Cambios realizados al primer requisito.

CRR2: Cambios realizados al segundo requisito.

CRR3: Cambios realizados al tercer requisito.

CRRn: Cambios realizados al enésimo requisito.

Cantidad de cambios realizados.

Muestra la cantidad de cambios que se realizan en los planes del proyecto, actividades y productos de trabajo.

$$CTCR = CR_1 + CR_2 + CR_3 + \dots + CR_n$$

Donde:

CTCR, es cantidad total de cambios realizados.

CR1, es el primer cambio realizado.

CR2, es el segundo cambio realizado.

CR3, es el tercer cambio realizado.

CRn, es el enésimo cambio realizado (Díaz, y otros, 2007).

Alan Davis en 1993 propone una lista de características que pueden emplearse para valorar la calidad del modelo de análisis y la correspondiente especificación de requisitos: Especificidad (ausencia de ambigüedad, corrección, completión, comprensión, capacidad de verificación, consistencia externa e interna, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización. Además apuntan que las especificaciones de alta calidad deben estar almacenadas electrónicamente, ser ejecutables o al menos interpretables, anotadas por importancia y estabilidad relativas, con su versión correspondiente, organizadas, con referencias cruzadas y especificadas al nivel correcto de detalle.

Aunque muchas de las características anteriores pueden ser de naturaleza cuantitativa, Davis sugiere que todas puedan representarse usando una o más métricas. Por ejemplo se asume que hay n_r requisitos en una especificación, tal como

$$n_r = n_f + n_{nf}$$

Donde n_f es el numero de requisitos funcionales y n_{nf} es el número de requisitos no funcionales (por ejemplo, rendimiento).

Para *determinar la especificidad de los requisitos*, Davis (Pressman, 2003) sugiere una métrica basada en la consistencia de la interpretación de los revisores para cada requisito:

$$Q_1 = n_{ui} / n_r$$

Donde n_{ui} es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas. Cuanto más cerca de uno este el valor de Q_1 menor será la ambigüedad de la especificación.

La compleción de los requisitos funcionales puede terminarse calculando la relación:

$$Q_2 = n_u / (n_i * n_s)$$

Donde n_u es el número de requisitos de función únicos, n_i es el número de entradas (estímulos) definidos o implicados por la especificación y n_s es el número de estados especificados. La relación Q_2 mide porcentaje de funciones necesarias que se han especificado para un sistema, sin embargo, no trata los requisitos no funcionales.

Para incorporarlos a una métrica global completa, se debe considerar *el grado de validación de los requisitos:*

$$Q_3 = n_c / (n_c * n_{nv})$$

Donde n_c es el número de requisitos que se han validados como correctos y n_{nv} el número de requisitos que no se han validado todavía.

2.4 Herramienta a utilizar

Durante el desarrollo de la investigación se encontraron varias herramientas de Gestión de Requisitos. Después de que se analizara cada una, se puede afirmar que existen varias similitudes entre ellas. Todas tienen un alto nivel de personalización desde cualquiera de sus ambientes de trabajo, permitiendo un buen desarrollo y manejo de la misma a cada empresa y cliente que requiere de su uso. Aunque existan tales similitudes, también existen las diferencias que marcan la preferencia de unas por encima de otras. Su principal diferencia radica en su disponibilidad para las diferentes plataformas virtuales que existen en el mundo, y su accesibilidad.

En el caso de las herramientas IRqA y Requisite Pro son totalmente propietarias y están sujetas a pago y otras como CaliberRM, DOORS, REM tienen determinadas restricciones y funcionan solo bajo plataformas totalmente privadas.

En cambio una herramienta como OSRMT que es gratis y está publicada bajo la licencia libre GPL, es multiplataforma y totalmente independiente del sistema operativo en que funcione, dado que su lenguaje de programación es Java, y que además posee un código completamente abierto, sería la herramienta más adecuada a aplicar en los proyectos productivos para la gestión de requisitos.

Esta herramienta desde su creación posee 5 versiones, cada una mejorando su funcionamiento y tratando de satisfacer al máximo las necesidades de sus usuarios y clientes. La versión OSRMT1.5 está muy bien elaborada y diseñada, cumple con las características básicas de una herramienta de gestión de requisitos además de poseer algunas características auxiliares. Sus funcionalidades son similares a las de muchas herramientas propietarias y su utilización le permite a cada proyecto controlar la especificación, minimizar el impacto de errores, reducir las no conformidades, mejorar la calidad mediante un adecuado análisis y gestión de los requisitos y además que proporciona una trazabilidad completa de la especificación.

2.4.1 Herramienta OSRMT Open Source Requirements Management Tool.

➤ Características generales de OSRMT.

Contiene un interfaz amigable e intuitiva para el usuario, es independiente de la plataforma y cumple con una de las características determinantes en la comparación que es un producto de software libre publicado bajo licencia GPL.

➤ Características específicas de OSRMT.

Base de datos

La base datos utilizada en la instalación fue PostgreSQL 8.1, aunque permite la migración completa de la herramienta sin dependencias de otras bases de datos a las siguientes: MS Access, MySQL, SQL Server y Oracle, que cualquiera de ellas actualmente son las más utilizadas en el mercado internacional según las necesidades específicas.

Licencia

Está publicado bajo Open Source General Public License (GPL) por Aron Smith, licencia de software libre. Que permite realizar cualquier tipo de cambio en el software mientras se mantenga los términos de la licencia.

Sistema Operativo

La herramienta es considerada multiplataforma debido a que es posible su instalación bajo cualquier sistema operativo, pues su código fuente es java para su instalación solo requiere la máquina virtual de java, la cual existe en casi todos los sistemas conocidos.

Lenguaje de programación

JAVA.

Interfaz en la que se utiliza

Java Swing browser.

Requerimientos de hardware y software

- Hardware

Para una PC cliente:

Computadora con cualquier tipo CPU a 1 MHz

Memoria RAM de 128 MB

Capacidad de Disco Duro en dependencia de documentos que se quieran guardar.

Para una PC servidor:

Computadora con cualquier tipo CPU a 1 MHz.

Memoria RAM de 256 MB

Capacidad de Disco Duro en dependencia de documentos que se quieran guardar.

- Software

Cualquier sistema operativo libre o privativo.

Se puede utilizar o no un servidor Web en dependencia de lo que se necesite.

 Servidor Base de Datos.

El servidor Web y de base datos pueden o no estar en la misma máquina.

 Se recomienda:

Para el servidor Web utilizar Apache 1.3.x ó Apache 2.x. En caso de no utilizar Apache, puede instalar cualquier servidor que sea capaz de ejecutar scripts hechos en SQL.

Para la base de datos utilizar PostgreSQL. La utilización de otras bases de datos será en dependencia de las necesidades de cada cliente.

 Interfaz gráfica

En la interfaz gráfica que muestra en el escritorio, pueden ser abiertos múltiples artefactos, actualizados o creados. Presenta propiedades de configuración que permite la personalización de la aplicación de manera visual sin necesidad de modificar el código fuente.

 Equipo de trabajo

Cuando se analizaban anteriormente las características generales, se plasmaba la importancia del equipo de trabajo y es una de las cosas por la que se aboga en el software libre (el trabajo en comunidad). Se puede determinar la agrupación de los especialistas en grupos de trabajos y crear un sistema más robusto durante el ciclo de vida del proyecto.

 Usuarios

A medida que se necesite incorporar nuevos usuarios a la aplicación, se pueden agregar cuentas de usuarios, cambiarlas o invalidarlas. Una vez creadas se le asigna un rol y al menos un grupo al que va a pertenecer y la contraseña asociada al usuario, que por directivas de seguridad, esta debe ser cambiado cuando se entra en la aplicación por primera vez.

 Grupos

Soporta grupos de trabajo con usuarios diferentes y con conexión al mismo equipo. Los grupos están compuestos por usuarios que tienen el mismo rol dentro del proyecto de trabajo.

Permisos

La concesión de permisos siempre es importante para lograr una mayor integridad y seguridad de los datos. En dependencia de la función que realice y al grupo de roles a que pertenezca le son asignados los permisos.

Acceso restringido a usuarios registrados.

Gestión de privilegios sobre determinadas tareas.

Estos se manejan a través de la base de datos, donde según el tipo de usuario y el rol que esta tiene dentro del equipo de trabajo así será el grupo donde será ubicado. Luego de acuerdo a los permisos así será el nivel de acceso que tendrá sobre todas las partes de la aplicación. Se puede establecer un sistema de autenticación a través de LDAP, viene implementado solo restaría ponerlo a funcionar.

Roles

La definición de roles está dirigida para lograr un mejor manejo de tareas, y contribuir con el control de permisos de acceso. La herramienta maneja el sistema de roles específico, donde los diferentes usuarios del sistema serán ubicados en estos grupos de roles según su función:

- Administradores
- Analista del negocio
- Desarrolladores
- Documentadores
- Jefes de proyectos
- Reguladores de la calidad
- Probadores de casos de prueba.

Aplicaciones adicionales

- ✓ La versión OSRMT 1.5 puede integrarse con otras aplicaciones que le permitan su buen funcionamiento.
- ✓ Se utiliza pgAdmin 3 para administrar la base datos: toda la administración posibles creaciones de nuevos campos pueden ser gestionados.
- ✓ El JasperReports con el iReport: para el manejo y creación de nuevos reportes.
- ✓ La herramienta iReport es un constructor/diseñador de informes visual, poderoso, intuitivo y fácil de usar para JasperReports escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos con cartas, imágenes, subinformes.

Seguridad de la información

OSRMT cuenta con un sistema de exportación e importación de artefactos que permite hacer copias de seguridad (backup), donde se les puede realizar a toda la información que se maneja como: requisitos, casos de usos, casos de pruebas, características para en un determinado momento recuperar la información.

Además la base datos donde se guarda la información está sujeta a todas las directivas de seguridad que establece PostgreSQL, y para la administración de la misma solo el administrador posee la contraseña y a través de pgAdmin3 puede administrar la misma. Toda la información de los artefactos que se generan son guardados en la base datos y bien identificados en el sistemas de tablas predefinidos (Ramos, y otros, 2008).

2.5 Descripción del curso de capacitación para analistas del PVSD.

Después de realizar una entrevista a los analistas de los proyectos del PVSD, se llegó a la conclusión que estos tenían poco conocimiento sobre la Ingeniería de Requisitos, es por esto que se decide proponer un curso de capacitación para dichos analistas y demás interesados sobre el tema en el PVSD, en dicho curso se abordarán conceptos fundamentales de la IR, técnicas a utilizar, herramientas de gestión de requisitos, todo enfocado a que comprendan la importancia que tiene la Ingeniería de Requisitos y aprendan a aplicar el modelo que se propone utilizar.

2.5.1 Objetivos y metodología del curso de capacitación sobre Ingeniería de Requisitos orientado a los analistas del PVSD.

Objetivo General:

Aplicar el modelo de Durán de ingeniería de requisitos.

Objetivos específicos:

- Explicar los conceptos básicos y estado actual de la Ingeniería de Requisitos.
- Describir ejemplos de casos reales que ilustren la importancia de aplicar adecuadamente Ingeniería de requisitos en el proceso de desarrollo de software.
- Caracterizar las principales metodologías y modelos en la aplicación de la Ingeniería de requisitos.
- Utilizar el modelo de Durán en el caso de estudio indicado.

Objetivos Educativos:

- Consolidar un estilo de trabajo independiente colectivo y creativo en la solución de los problemas, con un nivel de profesionalidad que garantice la sencillez y elegancia, así como la eficiencia y el ahorro de recursos en las soluciones adoptadas.
- Que los analistas utilicen y desarrollen una forma dialéctica de pensamiento y que lo apliquen consecuentemente en su enfoque sistémico de análisis.
- Que los analistas desarrollen la previsión, que sean capaces de anticiparse a problemas futuros y asumir posiciones responsables una vez que los conocen.

Metodología:

Teórico-práctico, con lectura previa de parte de los participantes sobre documentos sugeridos, exposiciones por parte del profesor o alumno ayudante. Trabajo práctico en forma de taller. Aplicación de seminarios y preguntas escritas. Interacción con el profesor a cargo del curso.

2.5.2 Contenido del curso de capacitación.

1. Introducción.

1.1 Necesidad de la Ingeniería de Requisitos

1.2 Definición de ingeniería de requisitos.

1.3 El concepto de requisito.

1.3.1 Características deseables de los requisitos.

1.3.2 Visión de los requisitos.

1.3.3 Dificultades para identificar requisitos.

2. Modelos de Ingeniería de Requisitos.

2.1 Otros modelos

2.1.1 El modelo de Pohl.

2.1.2 El modelo de Durán.

2.2 Comparación de modelos de procesos de ingeniería de requisitos.

2.3 Conclusiones

3. Elicitación y negociación de requisitos según modelo de Durán.

3.1 Introducción.

3.2 La elicitación de requisitos.

3.3 Técnicas de elicitación de requisitos.

3.3.1 Entrevistas.

3.3.2 Joint Application Development.

3.3.3 Brainstorming.

- 3.3.4 Casos de uso.
- 3.4 Tareas a realizar en la elicitación de requisitos.
- 4. Análisis de requisitos.
 - 4.1 Introducción.
 - 4.2 Tareas a realizar en el análisis de requisitos.
 - 4.3 Técnicas a utilizar en el análisis de requisitos
 - 4.4 Conclusiones
- 5. Validación de requisitos.
 - 5.1 Introducción.
 - 5.2 Técnicas a utilizar en la validación de requisitos.
 - 5.3 Tareas a realizar en la validación de requisitos.
- 6. Herramientas para la gestión de requisitos.
 - 6.1 Herramientas privativas para la gestión de requisitos.
 - 6.2 Herramientas libres para la gestión de requisitos.

2.5.3 Descripción de temas

La asignatura recorre 1 tema básico, el cuales presenta un sistema de evaluación basado en talleres y otras actividades de carácter obligatorio.

Tiene 4 actividades presenciales: estas consisten en el desarrollo de tres conferencias y un taller integrador al final de curso donde se presentarán por parte de los estudiantes trabajos referativos, sobre temas específicos.

El estudiante deberá ir estudiando cada contenido orientado con la finalidad de realizar cada una de las tareas que se presentan en los temas que se irán orientando.

Las áreas temáticas, conferencias que se presentan siguen la siguiente estructura:

TEMA I: La Ingeniería de Requisitos protagonista principal del desarrollo de software.

Introducción al curso. Presentación del curso IR.

Ejercicio: recoger expectativas de los estudiantes sobre la asignatura.

Conf. 1 Ingeniería de requisitos, necesidad de gestionar requisitos.

Ingeniería de requisitos como disciplina: definiciones y principios.

Ejercicio: Hacer mapa conceptual sobre el tema.

Conf.2: Modelos de Ingeniería de Requisitos.

El modelo de Pohl, modelo de Durán, comparación de modelos de procesos de ingeniería de requisitos.

Ejercicio práctico: Realizar una propuesta de qué modelo utilizar en el PVSD.

Conf. 3: Elicitación y análisis de IR.

Técnicas, tareas y plantillas a utilizar en cada una de las etapas.

Ejercicio práctico: Realizar la elicitación y el análisis del caso de estudio.

Conf. 4: Validación y herramientas de gestión de requisitos.

Técnicas, tareas y plantillas a utilizar en la actividad de Validación, y herramientas libres y propietarias de gestión de requisitos.

Ejercicio práctico: Realizar la validación del caso de estudio.

Seminario 1: La Gestión de Riesgos en la IR.

Taller Integrador: Aplicación del modelo de Durán en los distintos proyectos en desarrollo, utilizando algunas herramienta de gestión de requisitos.

Cada equipo presenta un glosario de términos de la asignatura. Aplicar encuesta sobre la asignatura.

Con el curso propuesto quedarán capacitados los analistas del Polo Video y Sonido Digital para poder aplicar el modelo propuesto de acuerdo a las necesidades del polo.

2.6 Aplicación del modelo de Duran al proyecto VideoWeb del PVSD.

El proyecto seleccionado para aplicar el modelo propuesto es el proyecto VideoWeb, fue elegido puesto que era el único que se encontraba en la fase de Inicio, el proyecto desarrolla un sistema web encaminado a la publicación y administración de archivos multimedia, basándose en la tecnología de streaming para brindar estos contenidos audiovisuales a través de la red de datos.

2.6.1 Elicitación de requisitos del proyecto VideoWeb del PVSD.

Tarea 1: Obtener información sobre el dominio del problema y el sistema actual.

En la actualidad muchas empresas, universidades, entidades en general con un grado de informatización alto están interesadas en que sus trabajadores tengan acceso a información mediática pero con el menos costo de espacio posible, además de que sufren la necesidad de reunir muchas personas para transmitirles información con el elemental problema de planificación que esto significa, por lo que una vía fácil de resolver esta situación sería a través de la solución de Streaming que brinda la Plataforma de VideoWeb, a través de la cual se puede disponer de contenido mediático en la web y puede ser visto sin ser descargado en las PC o pueden transmitirse conferencias o eventos y los usuarios sin moverse de su puesto de trabajo pueden tener acceso al evento con solo unos clics.

Muchas de estas organizaciones cuentan con una red interna pero no disponen de una solución que de las posibilidades de acceso a información sin que sea necesaria la copia física de la misma en cada PC personal.

El Ministerio del Poder Popular para la Energía y Petróleo de Venezuela tiene la necesidad de transmitir contenidos multimedia a través de la red datos sin la necesidad que los usuarios se vean obligados a descargar los mismos y puedan acceder a ellos de una manera fácil.

Tarea 2: Preparar y realizar las reuniones de elicitación/negociación.

Las reuniones fueron preparadas, fijando fecha de realización.

Tarea 3: Identificar/revisar los objetivos del sistema.

Entidad Cliente: Ministerio del Poder Popular para la Energía y Petróleo de Venezuela (MENPET)

Clasificación del proyecto: Desarrollo de producto

Objetivos del sistema

El sistema tiene como objetivo proveer la gestión y transmisión de contenidos multimedia a través de la red de datos por medio de un flujo constante de manera que los usuarios no tengan que descargar los materiales.

Transmitir de Contenidos Audiovisuales a través de la red de datos sin necesidad de copiar los ficheros en las PCs terminales.

Gestionar usuarios y roles según las políticas del cliente

Gestionar todo el contenido audiovisual que se quiera poner al servicio.

Publicar los materiales audiovisuales y las noticias que se requieran.

Para el total cumplimiento de lo anterior se incluyen los siguientes objetivos específicos del proyecto:

- Personalizar una distribución de LINUX para ser utilizada como sistema operativo que soporte la solución.
- Identificar y adaptar las funcionalidades del producto de acuerdo a las necesidades y entorno del Ministerio.
- Definir los formatos y características de los contenidos audiovisuales a utilizar por la plataforma.
- Desarrollar una interfaz de usuario acorde a los intereses del ministerio.
- Suministrar la tecnología capaz de soportar la solución informática.
- Capacitar al personal que administrará el sistema.

Tarea 4: Identificar/revisar los requisitos funcionales.

Requerimientos funcionales

El sistema debe ser capaz de:

RF1 Gestionar usuarios: El sistema debe permitir gestionar los datos referentes a los usuarios.

RF1.1 Adicionar usuario: El sistema debe permitir crear nuevos usuarios.

RF1.2 Modificar información de usuario: El sistema debe permitir modificar la información de los usuarios.

RF1.3 Modificar rol de usuario: El sistema debe permitir modificar los roles de cada uno de los usuarios.

RF1.4 Eliminar usuario: El sistema debe permitir eliminar los usuarios.

RF2 Autenticar usuario: El sistema debe permitir que los usuarios se autenticuen, una vez chequeado el rol que poseen, se les dará su acceso al sistema.

RF3 Gestionar sesiones: El sistema debe permitir a los usuarios con permisos gestionar las sesiones.

RF3.1 Adicionar sesión: El sistema debe permitir a los usuarios con permisos crear una nueva sesión de contenido.

RF3.2 Modificar sesión: El sistema debe permitir a los usuarios con permisos modificar los datos de una sesión de contenido.

RF3.3 Eliminar sesión: El sistema debe permitir a los usuarios con permisos eliminar una sesión de contenido.

RF4 Gestionar categorías: El sistema debe permitir a los usuarios con permisos gestionar las categorías.

RF4.1 Adicionar categoría: El sistema debe permitir a los usuarios con permisos crear una nueva categoría de contenido dentro de una sesión existente.

RF4.2 Modificar categoría: El sistema debe permitir a los usuarios con permisos modificar los datos de una categoría de contenido.

RF4.3 Eliminar categoría: El sistema debe permitir a los usuarios con permisos eliminar una categoría de contenido.

RF5 Gestionar tipologías de medias: El sistema debe permitir a los usuarios con permisos gestionar las sesiones.

RF5.1 Adicionar tipología de media: El sistema debe permitir a los usuarios con permisos crear una nueva tipología de media y definir los datos que va a incluir dicha tipología.

RF5.2 Modificar tipología de media: El sistema debe permitir a los usuarios con permisos modificar los datos de una tipología de media.

RF5.3 Eliminar tipología de media: El sistema debe permitir a los usuarios con permisos eliminar una tipología de media.

RF6 Gestionar archivos multimedia: El sistema debe permitir a los usuarios con permisos gestionar archivos multimedia y los datos asociados a estos.

RF6.1 Adicionar archivos multimedia: El sistema debe permitir a los usuarios con permisos almacenar los archivos multimedia en el servidor de medias y crear las referencias pertinentes en la base de datos con todos los datos asociados a estos.

RF6.2 Modificar archivos multimedia: El sistema debe permitir a los usuarios con permisos modificar los datos asociados a los archivos multimedia.

RF6.3 Eliminar archivos multimedia: El sistema debe permitir a los usuarios con permisos eliminar archivos multimedia con los datos asociados a estos.

RF7 Gestionar publicaciones multimedia: El sistema debe permitir a los usuarios con permisos gestionar publicaciones multimedia.

RF7.1 Crear publicación multimedia: El sistema debe permitir a los usuarios con permisos crear publicaciones multimedia a partir de los archivos multimedia previamente almacenados en el servidor de medias, permitiendo predefinir el tiempo de publicación acotado por un inicio y fin.

RF7.2 Modificar publicación multimedia: El sistema debe permitir a los usuarios con permisos modificar los datos de la publicación multimedia.

RF7.3 Eliminar publicación multimedia: El sistema debe permitir a los usuarios con permisos eliminar publicaciones multimedia.

RF8 Publicar archivo multimedia: El sistema debe permitir a los usuarios con permisos activar las publicaciones multimedia previamente elaboradas, copiando el archivo multimedia asociado a la publicación al servidor de streaming.

RF9 Dejar de publicar archivo multimedia: El sistema debe permitir a los usuarios con permisos desactivar las publicaciones multimedia previamente publicadas, eliminando el archivo multimedia asociado a la publicación del servidor de streaming.

RF10 Reproducir archivos multimedia: El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario.

RF10.1 Reproducir archivos multimedia desde un reproductor externo: El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario desde un reproductor externo instalado en el sistema.

RF10.2 Reproducir archivos multimedia desde un reproductor embebido: El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario mediante un reproductor embebido en el navegador.

RF11 Gestionar artículo de contenido: El sistema debe permitir gestionar noticias.

RF11.1 Adicionar artículo de contenido: El sistema debe permitir a los usuarios con permisos adicionar noticias permitiendo predefinir el tiempo de publicación de la noticia, acotado por un inicio y fin, aunque la noticia no estará publicada hasta que un usuario con permisos de publicación la publique.

RF11.2 Modificar artículo de contenido: El sistema debe permitir a los usuarios con permisos modificar noticias.

RF11.3 Eliminar artículo de contenido: El sistema debe permitir a los usuarios con permisos eliminar noticias.

RF12 Publicar artículo de contenido: El sistema debe permitir a los usuarios con permisos publicar las noticias previamente adicionadas.

RF13 Dejar de publicar artículo de contenido: El sistema debe permitir a los usuarios con permisos dejar de publicar las noticias previamente publicadas.

RF14 Visualizar artículo de contenido: El sistema debe permitir a los usuarios visualizar noticias previamente publicadas.

RF15 Gestionar lista de reproducción: El sistema debe permitir gestionar listas de reproducción.

RF15.1 Crear lista de reproducción: El sistema debe permitir a los usuarios crear listas de reproducción.

RF15.2 Reproducir lista de reproducción: El sistema debe permitir a los usuarios reproducir sus listas de reproducción.

RF15.3 Modificar lista de reproducción: El sistema debe permitir a los usuarios modificar sus listas de reproducción.

RF15.3 Eliminar lista de reproducción: El sistema debe permitir a los usuarios eliminar sus listas de reproducción.

RF16. Realizar búsquedas: El sistema debe permitir realizar búsquedas de los contenidos que posee la plataforma.

RF16.1 Realizar búsqueda de artículo de contenido: El sistema debe permitir realizar búsquedas de las noticias publicadas en la plataforma.

RF16.2 Realizar búsqueda de archivos multimedia: El sistema debe permitir realizar búsquedas de los archivos multimedia almacenados en el servidor de medias.

RF16.3 Realizar búsqueda de los usuarios registrados: El sistema debe permitir a los usuarios con permisos realizar búsquedas de los usuarios registrados.

RF16.4 Realizar búsqueda de publicaciones de archivos multimedia: El sistema debe permitir realizar búsquedas de publicaciones de archivos multimedia.

RF17 Crear cuenta de usuario: El sistema debe permitir a los invitados crearse una cuenta de usuario.

Tarea 5: Identificar/revisar los requisitos no funcionales.

Requerimientos no funcionales del sistema

Requerimientos de Software

- Se debe utilizar como servidor web el “Apache”.
- Se debe utilizar como servidor Streaming el “Darwin Streaming Server”.
- Se debe utilizar para captura y transmisión en vivo el “Mp4live”.
- Se debe utilizar como sistema operativo en los servidores el “Nova-Linux”.
- Para la reproducción se los archivos multimedia se debe utilizar el reproductor VLC, como reproductor externo y como plugin embebido en el navegador, por ser multiplataforma y soportar tanto el protocolo de transmisión como el formato de fichero empleado.

Requerimientos de Hardware

- Servidor de archivos multimedia con un soporte de almacenamiento de 15 Terabytes.
- Servidor de aplicaciones con disco duro SCSI de al menos 500 Gb, 4 Gb de RAM, procesador Intel Quad-Core Xeon a 3.00 GHz y tarjeta de red Gigabit Ethernet.
- Al menos dos estaciones de captura con disco duro SCSI de 250 Gb, 2 Gb de RAM, Dual-Core Xeon a 3.00 GHz y tarjeta de red Gigabit Ethernet.
- Al menos dos tarjetas de captura HAUPPAGE WINTV-PVR 350 PCI WATCH RECORD TV.
- Al menos una Cámara de video Sony HDR-FX1000E con sus accesorios.

Restricciones en el diseño y la implementación.

- Diseño e implementación de una arquitectura flexible, que permita la fácil integración o desintegración de componentes.
- La arquitectura debe soportar migrar la interfaz de usuario sin impactos considerables en re-implementación.
- El patrón arquitectónico que se debe emplear en el desarrollo es el modelo-vista-controlador.
- El lenguaje de programación que se debe utilizar es PHP.
- Los protocolos para la comunicación que se deben usar son HTTP entre el cliente y el servidor web y RTSP entre el cliente y el servidor streaming.

Requerimientos de apariencia o interfaz externa

- Interfaz amigable, interactiva, intuitiva y de fácil comprensión para el usuario.

Requerimientos de confidencialidad

- Permitir autenticación segura.
- La seguridad se establecerá por roles que se le asignarán a los usuarios que interactúen con el sistema.
- Acceder a la información según el rol del usuario.

Requerimientos de disponibilidad

- El sistema debe de estar disponible las 24 horas del día.

Requerimientos de Usabilidad

- El módulo de administración debe caracterizarse por un alto grado de flexibilidad.
- Disponibilidad de visualizar los archivos multimedia en un reproductor.
- Mostar la información de forma lógica y correctamente estructurada.
- El servidor streaming debe mantener buenas prestaciones (demanda de archivos)

Requerimientos de Soporte

- El soporte y/o mantenimiento del sitio no debe detener el servicio.
- La capa de acceso a datos, debe soportar una migración del Gestor de Bases de Datos en proyecciones futuras.

Tarea 6: Priorizar objetivos y requisitos

RF1 Gestionar usuarios. (Crítico)

RF1.1 Adicionar usuario. (Crítico)

RF1.2 Modificar información de usuario. (Crítico)

RF1.3 Modificar rol de usuario. (Crítico)

RF1.4 Eliminar usuario. (Crítico)

RF2 Autenticar usuario. (Crítico)

RF3 Gestionar sesiones. (Secundario)

RF3.1 Adicionar sesión. (Secundario)

RF3.2 Modificar sesión. (Secundario)

RF3.3 Eliminar sesión. (Secundario)

RF4 Gestionar categorías. (Secundario)

RF4.1 Adicionar categoría. (Secundario)

RF4.2 Modificar categoría. (Secundario)

RF4.3 Eliminar categoría. (Secundario)

RF5 Gestionar tipologías de medias. (Crítico)

RF5.1 Adicionar tipología de media. (Crítico)

RF5.2 Modificar tipología de media. (Crítico)

RF5.3 Eliminar tipología de media. (Crítico)

RF6 Gestionar archivos multimedia. (Crítico)

RF6.1 Adicionar archivos multimedia. (Crítico)

RF6.2 Modificar archivos multimedia. (Crítico)

- RF6.3 Eliminar archivos multimedia. (Crítico)
- RF7 Gestionar publicaciones multimedia. (Crítico)
 - RF7.1 Crear publicación multimedia. (Crítico)
 - RF7.2 Modificar publicación multimedia. (Crítico)
 - RF7.3 Eliminar publicación multimedia. (Crítico)
- RF8 Publicar archivo multimedia. (Crítico)
- RF9 Dejar de publicar archivo multimedia. (Crítico)
- RF10 Reproducir archivos multimedia. (Crítico)
 - RF10.1 Reproducir archivos multimedia desde un reproductor externo. (Crítico)
 - RF10.2 Reproducir archivos multimedia desde un reproductor embebido. (Crítico)
- RF11 Gestionar artículo de contenido. (Secundario)
 - RF11.1 Adicionar artículo de contenido. (Secundario)
 - RF11.2 Modificar artículo de contenido. (Secundario)
 - RF11.3 Eliminar artículo de contenido. (Secundario)
- RF12 Publicar artículo de contenido. (Secundario)
- RF13 Dejar de publicar artículo de contenido. (Secundario)
- RF14 Visualizar artículo de contenido. (Secundario)
- RF15 Gestionar lista de reproducción. (Opcional)
 - RF15.1 Crear lista de reproducción. (Opcional)
 - RF15.2 Reproducir lista de reproducción. (Opcional)
 - RF15.3 Modificar lista de reproducción. (Opcional)
 - RF15.3 Eliminar lista de reproducción. (Opcional)
- RF16. Realizar búsquedas. (Secundario)
 - RF16.1 Realizar búsqueda de artículo de contenido. (Secundario)
 - RF16.2 Realizar búsqueda de archivos multimedia. (Crítico)
 - RF16.3 Realizar búsqueda de los usuarios registrados. (Secundario)
 - RF16.4 Realizar búsqueda de publicaciones de archivos multimedia. (Secundario)
- RF17 Crear cuenta de usuario. (Crítico)
- R18 Gestionar señal en vivo. (Opcional)
 - R18.1 Publicar señal de audio o video en vivo. (Opcional)
 - R18.2 Modificar señal de audio o video en vivo. (Opcional)
 - R18.3 Eliminar señal de audio o video en vivo. (Opcional)
- R19. Cambiar interfaz visual. (Secundario)

Luego de identificar los requisitos no funcionales y funcionales y priorizar estos últimos se procede a realiza la elaboración del documento de Especificación de requisitos (Ver Anexo 5).

2.6.2 Análisis de los requisitos del proyecto Plataforma VideoWeb del PVSD.

Tareas recomendadas en la actividad de Análisis.

Analizar los requisitos funcionales.

Los requisitos funcionales se analizan teniendo como resultado la descripción de los casos de uso.

Analizar los requisitos no funcionales.

El análisis de los requisitos no funcionales se realiza mediante una lectura detenida de su contenido, y combinando esta lectura con la experiencia, detectar posibles conflictos como la imposibilidad técnica de la implementación de ciertos requisitos, la necesidad de optar por unas características u otras.

Después de analizar los requisitos funcionales y no funcionales se desarrollan los prototipos que es una tarea opcional en el caso del análisis. Ver Anexo # 3

Gestión de Riesgos.

ID Requisito	Tipo de Riesgo	Riesgos
RF1 Gestionar usuarios.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF2 Autenticar usuario.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF3 Gestionar sesiones.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF4 Gestionar categorías.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF5 Gestionar	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.

tipologías de medias.		
RF6 Gestionar archivos multimedia.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF7 Gestionar publicaciones multimedia.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF8 Publicar archivo multimedia.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF9 Dejar de publicar archivo multimedia.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF10 Reproducir archivos multimedia.	De requisitos. De personal. De estimación.	Los clientes no comprenden el impacto de los cambios en los requerimientos. El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad del requisito. El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado.
RF11 Gestionar artículo de contenido.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF12 Publicar artículo de contenido.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.

RF13 Dejar de publicar artículo de contenido.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF14 Visualizar artículo de contenido.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF15 Gestionar lista de reproducción.	De requisitos. De personal. De estimación.	Los clientes no comprenden el impacto de los cambios en los requerimientos. El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad del requisito. El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado.
RF16. Realizar búsquedas.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
RF17 Crear cuenta de usuario.	De requisitos.	Los clientes no comprenden el impacto de los cambios en los requerimientos.
R18 Gestionar señal en vivo.	De requisitos. De personal. De estimación.	Los clientes no comprenden el impacto de los cambios en los requerimientos. El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad del requisito. El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado.
R19. Cambiar interfaz visual.	De requisitos. De personal. De estimación.	Los clientes no comprenden el impacto de los cambios en los requerimientos. El personal no cuenta con los conocimientos requeridos para enfrentar la complejidad del requisito. El tiempo requerido para desarrollar el proceso de ingeniería de requisitos está subestimado,

Tabla 11 Identificación de riesgos de los requisitos del proyecto VideoWeb.

Se comunica al equipo de desarrollo los riesgos potenciales identificados.

Riesgo	Probabilidad	Efectos
Imposible contratar personal con los conocimientos requeridos.	alta	catastrófico
El tiempo requerido para desarrollar el proceso de IR está subestimado.	alta	serio
Los clientes no comprenden el impacto de los cambios en los requerimientos.	moderada	tolerable

Tabla 12 Probabilidad y efecto de los riesgos identificados en los requisitos del proyecto Plataforma VideoWeb.

Se comunica al equipo de desarrollo los las probabilidades de que ocurran los riesgos detectados y los efectos que estos causarían.

Riesgo	Estrategia
Problemas de reclutamiento.	Organizar cursos de capacitación para el personal ya existente.
Tiempo de IR subestimado.	Alertar al cliente de las dificultades potenciales y las posibilidades de retraso.
Cambios en los requisitos.	Rastrear la información para valorar el impacto de los requerimientos, maximizar la información oculta en ellos.

Tabla 13 Estrategias para los riesgos identificados en los requisitos del proyecto VideoWeb.

Se comunica al equipo de desarrollo el plan de mitigación de los riesgos identificados.

2.6.3 Validación de requisitos del proyecto Plataforma VideoWeb.

Tareas recomendadas en la actividad de Validación.

Validar requisitos funcionales.

Validar requisitos no funcionales.

Para el cumplimiento de estas tareas se utilizó la técnica de validación Walkthroughs en la que se especificaron los participantes siguientes.

Autor: Analista Maggie A. Silva de la Hera

Presentador: Maggie A. Silva de la Hera

Revisores: Abel.

Moderador: Jefe de Proyecto Yunier Albrecht.

Otra de las técnicas que se aplicó fue la entrevista en la que se desarrollaron las siguientes preguntas.

- ¿Están incluidas todas las funciones requeridas por el cliente? (completa)
- ¿Existen conflictos en los requerimientos? (consistencia)
- ¿Tiene alguno de los requerimientos más de una interpretación? (no ambigua)
- ¿Está cada requerimiento claramente representado? (entendible)
- ¿Pueden los requerimientos ser implementados con la tecnología y el presupuesto disponible? (factible)
- ¿Existe facilidad para hacer cambios en los requerimientos? (modificable)
- ¿Está claramente definido el origen de cada requisito? (rastreadable)

Además de la aplicación de la técnica especificada anteriormente se le presentó al cliente prototipos del sistema y se modificó y actualizó la Plantilla de Especificación de requisitos.

2.7 Conclusiones del capítulo 2.

En el capítulo se realizó una breve descripción del modelo propuesto en el capítulo 1, además se presentó los cambios, o sea las adaptaciones que se le aplican al modelo, incluyendo la gestión de riesgos y definiendo además la herramienta de gestión de requisitos que se va a utilizar que será la Open Source Requirements Management Tool, por las ventajas que se mencionaron. Además se aplicó el modelo con las modificaciones al proyecto VideoWeb del PVSD. Se considera que el modelo con las modificaciones realizadas se puede aplicar a los proyectos del PVSD, para esto se describió también un curso de capacitación destinado a los analistas del polo, para que estos aprendan a utilizar el modelo.

CAPÍTULO 3: Valoración del modelo Propuesto.

En este capítulo se verifica que el modelo propuesto es válido, aplicable, esto se comprobará de 2 formas, a través del método experto Delphi y mediante la comparación de los resultados de la aplicación entre el proyecto VideoWeb y otro proyecto del PVSD el cual no haya utilizado el modelo.

3.1 Resumen de resultados en el desarrollo de la gestión de requisitos en el proyecto Plataforma VideoWeb.

Al ser aplicado el modelo se obtuvieron requisitos con las características deseadas, requisitos: consistentes, no ambiguos, necesarios, completos. Además un buen entendimiento en el equipo de trabajo, de manera que a la hora de pasar al flujo de trabajo de análisis fue mucho más fácil su desarrollo. Se tuvo en cuenta la gestión de riesgo con lo cual se detectaron los riesgos, y se mitigaron, evitando en todo momento caer en los problemas. Con la utilización de la herramienta OSMRT se tiene mejor organizada y registrada la información.

Con una estrategia para la gestión de requisitos, el flujo de trabajo de requerimiento se realiza con eficiencia.

El documento más importante obtenido fue la Especificación de Requisitos. Otros de los documentos generados fue la Descripción de los Casos de uso, los cuales se obtuvieron de la forma más refinada posible.

3.2 Comparación de los proyectos de acuerdo a la gestión de requisitos.

En la tabla 14 se muestra como la plataforma VideoWeb tuvo que redefinir los requisitos y para esto se planificaron 30 días, lo cual trae como consecuencia atrasos en el proyecto. Por otra parte la plataforma VideoWeb en solo 31 días se gestionaron los requisitos teniendo como resultado un documento de especificación de requisitos lo más refinado posible. En la tabla se especifica cada una de las ventajas que tiene la aplicación del modelo. En cuanto al control de cambios, si se aplica se puede conocer cuáles fueron los requisitos originales; también se aprecia la gestión de riesgo lo cual brinda la posibilidad de minimizar, evadir y controlar los riesgos. Otro de los aspectos es la utilización de métricas, estas fueron de mucha ayuda en la actividad de validar los requisitos con el cliente. La especificación de requisitos en el proyecto VideoWeb con la aplicación del modelo

Parámetros de Comparación	VideoWeb	Plataforma TV
Tiempo destinado a la gestión de requisitos.	31 día	54 días, 30 días de redefinición de requisitos.
Control de cambios de los requisitos	Se aplica.	No se aplica, desconocimiento de los requisitos originales.
Uso de la Gestión de riesgos	Se aplica, posibilidad de minimizar, evadir y controlar los riesgos.	No se aplica, propenso a riesgos.
Utilización de Métricas para la gestión de requisito	Se aplica.	No se aplica, imposibilidad de conocer por ciento de requisito que cumplen con las necesidades del cliente, entre otros.
Utilización de herramienta para la Gestión de Requisitos	Se aplica.	No se aplica, impedimento de realizar reutilización de requisitos en proyectos futuros por el equipo de desarrollo.

Tabla 14 Comparación de los proyectos VideoWeb y Plataforma Tv, de acuerdo a la gestión de requisitos.

3.3 Resultados obtenido con la gestión de requisitos según el modelo propuesto. (Contrastación de la hipótesis).

Con la utilización del modelo y la herramienta de ingeniería de requisitos, en el proyecto plataforma VideoWeb, se especificaron los requisitos de una forma más completa, en un menor tiempo, utilizando el control de cambios para que en proyectos futuros sea posible reutilizar los requisitos, se realizó la gestión de riesgos de manera que en etapas tempranas se puedan mitigar los riesgos o por lo menos guiar el proyecto para que no sucedan, la utilización de las métricas dio como resultado una mejor validación de los requisitos. El flujo de trabajo estuvo muy bien organizado, basándose en cada una de las etapas del modelo y realizando las tareas de cada una de ellas. Se puede concluir que con la aplicación del modelo

de IR mejoró el flujo de trabajo de requerimientos los desarrollos de software del Polo de Video y Sonido Digital.

3.4 Análisis con modelos de calidad.

Parámetros de Comparación	Modelo Durán adaptado	ISO 9001-2000
¿Plantea actividades para la gestión de requisitos?	Si	Si
¿Incorpora herramientas para la gestión de requisitos?	Si	No
¿Plantea técnicas para la gestión de requisitos?	Si	No
¿Menciona gestión de riesgos en requisitos?	Si	No
¿Indica la aplicación de métricas en la gestión de requisitos?	Si	No
¿Sugiere control de cambio de los requisitos?	Si	Si

Tabla 15 Comparación de la gestión de requisitos entre la norma ISO y el modelo de Durán adaptado (Elaboración propia).

Dentro de la familia ISO se encuentra la norma ISO 9001:2000 que es la encargada de especificar los requisitos de un sistema de gestión de la calidad aplicada a empresas u organizaciones. Esta hace referencia a la Gestión de Requisitos en el proceso relacionado con el cliente.

Esta norma está compuesta por varios capítulos que hacen referencia a los procesos de la Gestión de Requisitos. En el proceso relacionado con el cliente, se proponen 5 actividades pero a diferencia del modelo no explica que tareas se pueden llevar a cabo para el cumplimiento de las actividades, ni plantea técnicas para el desarrollo de las tareas, tampoco menciona alguna herramienta para apoyar la gestión de requisitos.

3.5 Análisis con metodologías de desarrollo.

Parámetros de Comparación	Modelo Durán adaptado	RUP
¿Plantea actividades para la gestión de requisitos?	Si	Si
Cantidad de actividades que plantea.	3	5
¿Incorpora herramientas para la gestión de requisitos?	Si	No
¿Propone técnicas para la gestión de requisitos?	Si	Si
¿Menciona la gestión de riesgos para la gestión de requisitos?	Si	Si
¿Indica la aplicación de métricas en la gestión de requisitos?	Si	No
¿Sugiere el control de cambio de los requisitos?	Si	Si

Tabla 16 Comparación de la gestión de requisitos entre la metodología de desarrollo RUP y el modelo de Durán adaptado (Elaboración propia).

RUP brinda una guía para encontrar, organizar, documentar, y seguir los cambios de los requisitos funcionales y restricciones. Utiliza una notación de Caso de Uso y escenarios para representar los requisitos.

Dentro de RUP está presente el proceso de ingeniería de requisitos, el cual garantiza que se apliquen mecanismos para comprender lo que verdaderamente quiere el cliente; analizando sus necesidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedades, validando la especificación y gestionando los requisitos. Estos mecanismos se definen en cinco pasos: Identificación de requisitos, Análisis y negociación de requisitos, Especificación de requisitos, Modelado del sistema, Validación de requisitos, Gestión de requisitos. Si se analiza el modelo propuesto se demuestra que RUP y el modelo cumplen con las actividades que se deben llevar a cabo en la gestión de requisitos, además los 2 hacen alusión a técnicas para la gestión de requisitos y plantean la gestión de riesgos, otro aspecto importante que tienen en cuenta es el control de cambios de los requisitos.

3.6 Mejoras del Modelo de Durán.

- Utilización de la gestión de riesgos, para permitir minimizar, evadir y controlar los riesgos. Para lograr producir aquello que el cliente requiere, en el plazo solicitado y ajustados al presupuesto

asignado, se necesita desarrollar un proceso que incluya desde la etapa más temprana la gestión de los riesgos asociados a los requisitos, de forma que se contribuya al mejoramiento gradual del proceso de desarrollo y la gestión de un proyecto de software que logre la satisfacción del cliente en estas organizaciones. La gestión de riesgos en el ámbito del software procura formalizar conocimiento orientado a la minimización o evitación de riesgos en proyectos de desarrollo de software, mediante la generación de principios y buenas prácticas de aplicación realista.

- Realización de una propuesta para proyectos de complejidad media y pequeña, minimizando el tiempo de demora en esta etapa, para dichos proyectos y evitando el poco entendimiento con los desarrolladores.
- Disminución de la cantidad de plantillas a llenar para la gestión de requisitos.
- Utilización de Métricas para la Gestión de requisitos en el modelo.

3.7 Evaluación de la propuesta según expertos. Método Delphi.

Los métodos de evaluación de expertos tienen como objetivos la búsqueda de experiencia y conocimientos de un grupo de personas, considerados expertos en la materia a evaluar, permitiendo la obtención de criterios subjetivos, mediante cuestionarios, sobre la calidad y efectividad de los resultados de las investigaciones.

Teniendo en cuenta que el objetivo de este trabajo, es dar una propuesta para la aplicación de la Ingeniería de requisitos en los proyectos del PVSD, la misma debe ser evaluada por un personal con capacitación sobre el tema.

Con este fin se utiliza el Método Delphi, el cual consiste en la selección de un grupo de expertos a los que se les pregunta su opinión sobre cuestiones referidas a acontecimientos del futuro. Este procedimiento extrae y maximiza las ventajas que presentan los métodos basados en grupos de expertos y minimiza sus inconvenientes. Para ello se aprovecha la sinergia del debate en el grupo y se eliminan las interacciones sociales indeseadas que existen dentro de todo grupo. De esta forma se espera obtener un consenso lo más fiable posible del grupo de expertos.

Este método presenta tres características fundamentales:

- Anonimato: Durante un Delphi, ningún experto conoce la identidad de los otros que componen el grupo de debate. Esto tiene una serie de aspectos positivos, como son:
 - Impide la posibilidad de que un miembro del grupo sea influenciado por la reputación de otro de los miembros o por el peso que supone oponerse a la mayoría. La única influencia posible es la de la congruencia de los argumentos.

- Permite que un miembro pueda cambiar sus opiniones sin que eso suponga una pérdida de imagen.
- El experto puede defender sus argumentos con la tranquilidad que da saber que en caso de que sean erróneos, su equivocación no va a ser conocida por los otros expertos.
- Iteración y realimentación controlada: La iteración se consigue al presentar varias veces el mismo cuestionario. Como, además, se van presentando los resultados obtenidos con los cuestionarios anteriores, se consigue que los expertos vayan conociendo los distintos puntos de vista y puedan ir modificando su opinión si los argumentos presentados les parecen más apropiados que los suyos.
- Respuesta del grupo en forma estadística: La información que se presenta a los expertos no es sólo el punto de vista de la mayoría, sino que se presentan todas las opiniones indicando el grado de acuerdo que se ha obtenido.

3.7.1 Pasos para la realización del Método Delphi.

- Paso 1: Formulación del problema

Se trata de una etapa fundamental en la realización de un Delphi. En un método de expertos, la importancia de definir con precisión el campo de investigación es muy grande por cuanto que es preciso estar muy seguros de que los expertos reclutados y consultados poseen todas las mismas nociones de este campo. La elaboración del cuestionario debe ser llevada a cabo según ciertas reglas: las preguntas deben ser precisas, cuantificables (versan por ejemplo sobre probabilidades de realización de hipótesis y/o acontecimientos, la mayoría de las veces sobre datos de realización de acontecimientos) e independientes (la supuesta realización de una de las cuestiones en una fecha determinada no influye sobre la realización de alguna otra cuestión).

- Fase 2: Elección de expertos.

La etapa es importante puesto que el término de "experto" es ambiguo. Con independencia de sus títulos, su función o su nivel jerárquico, el experto será elegido por su capacidad de encarar el futuro y los conocimientos que posea sobre el tema consultado. La falta de independencia de los expertos puede constituir un inconveniente; por esta razón los expertos son aislados y sus opiniones son recogidas por vía postal o electrónica y de forma anónima; así pues se obtiene la opinión real de cada experto y no la opinión más o menos falseada por un proceso de grupo (se trata de eliminar el efecto de los líderes).

- Fase 3: Elaboración y lanzamiento de los cuestionarios (en paralelo con la fase 2)

Los cuestionarios se elaboran de manera que faciliten, en la medida en que una investigación de estas características lo permite, la respuesta por parte de los consultados.

- Fase 4: Desarrollo práctico y explotación de resultados.

El cuestionario es enviado a cierto número de expertos (hay que tener en cuenta las no-respuestas y abandonos. Naturalmente el cuestionario va acompañado por una nota de presentación que precisa las finalidades, el espíritu del Delphi, así como las condiciones prácticas del desarrollo de la encuesta (plazo de respuesta, garantía de anonimato).

Luego se procede al estudio de resultados para llegar a conclusiones.

3.7.2 Selección de los expertos

Para la evaluación de la propuesta, se han escogido 7 expertos, se señala que es necesario esta cantidad como mínimo, debido a que el error disminuye notablemente por cada experto añadido hasta llegar a los siete y como máximo 30 pues la mejora en la previsión es muy pequeña y normalmente el incremento en costo y trabajo de investigación no compensa la mejora, los expertos seleccionados son profesores integrantes del polo VSD.

Existen una serie de características propias que poseen cada uno de los expertos seleccionados y que se tuvieron en cuenta para conformar el panel.

- Responsabilidad.
- Competencia.
- Creatividad.
- Seriedad.
- Honestidad.
- Disposición en participar en la encuesta.
- Capacidad de análisis.

El objetivo de este proceso es darle validez a la solución propuesta, mediante la aplicación de cuestionarios a la muestra seleccionada. Esta selección se lleva a cabo, teniendo en cuenta la experiencia y capacitación que presenta el profesional sobre el tema. Las respuestas de este grupo de expertos pueden contribuir a perfeccionar el proceso propuesto.

3.7.3 Elaboración de la encuesta.

Para la elaboración de la encuesta se tuvieron en cuenta los principios básicos que debería cumplir la propuesta presentada para su utilización en el proceso productivo del PVSD. La encuesta fue conformada con preguntas abiertas de enfoque investigativo, sobre la validez de la solución propuesta al problema planteado y la evaluación del proceso. Algunos de los criterios que se tuvieron en cuenta fueron los siguientes:

- Satisfacción a las necesidades de los proyectos productivos.

- Calidad de la investigación.
- Novedad científica.
- Aporte científico.
- Facilidades de comprensión.
- Adaptabilidad a diferentes entornos de producción de software.
- Repercusión en los proyectos productivos.
- Contribución al proceso de desarrollo de software.
- Contribución a la Gestión de Proyectos.

En todos los casos los expertos recibieron la documentación del proceso propuesto y se les requirió cumplir con un lapso de tiempo determinado para dar las respuestas o hacer las preguntas pertinentes que les hubiesen surgido al estudiar el documento presentado. (Ver Anexo 4).

3.7.4 Resultados de la evaluación.

Después de procesada la encuesta se pudo determinar que los expertos estuvieron de acuerdo en que era importante asegurar una correcta aplicación de la Ingeniería de requisitos en los proyectos productivos del PVSD. Las referencias del porqué de esa respuesta, influyeron en que de esa manera se podía obtener en el PVSD un software con mayor calidad, basada su elaboración en las funcionalidades descritas por requisitos con superior grado de eficacia. Además, alegaron la importancia de presentar una documentación completa de este proceso, debido a que con ella se evita mal entendidos entre las personas que participan en esta disciplina.

Los criterios brindados por los expertos sobre la necesidad de un mejoramiento en esta disciplina en el polo se basan en que ésta es la única manera de tener claridad en lo que se quiere con la aplicación. Además, con ella se define el qué desde un inicio y es la forma de prepararse en temas de organización y procesos que a veces no se imparten en la carrera.

Otros criterios se refieren a la repercusión que tiene una mala gestión de requisitos dentro del proceso de desarrollo de software, lo cual puede provocar incumplimientos en el plan organizacional del proyecto, teniendo en cuenta que esto podría provocar demoras e insuficiencias en la calidad del producto.

Todos los expertos estuvieron de acuerdo en que la propuesta realizada está a la altura de las necesidades y posibilidades de aplicación de los proyectos productivos, además podrá aumentar la efectividad del trabajo de producción de software, y tendrá óptimas posibilidades de uso en los proyectos productivos del PVSD, aunque algunos alegan que esto depende mucho del personal y de la capacitación que se le pueda dar a los integrantes del Polo y los proyectos respectivamente, también le atribuyen importancia a la utilización del modelo. (Ver tabla 17)

Preguntas	Exp	Exp	Exp	Exp	Exp	Exp	Exp
	1	2	3	4	5	6	7
¿La propuesta está a la altura de las necesidades y posibilidades de aplicación de los proyectos productivos del PVSD?	Si	Si	Si	Si	Si	Si	Si
¿Podrá aumentar la efectividad del trabajo?	Si	Si	Si	Si	Si	Si	Si
¿Tendrá óptimas posibilidades de uso en los proyectos productivos del PVSD?	Si	Si	Si	Si	Si	Si	Si
¿Tiene importancia la utilización del modelo propuesto en los proyectos de desarrollo del PVSD?	Si	Si	Si	Si	Si	Si	Si

Tabla 17 Resumen de respuestas a las preguntas formuladas a los expertos.

A continuación se representan algunos de los aspectos más importantes tenidos en cuenta en la encuesta:

- El 85.71% de los expertos evalúa de 5 la satisfacción a las necesidades de los proyectos productivos con el modelo propuesto, el resto lo evalúa de 4, lo que representa un 97,14% de aceptación.
- El 100 % de los expertos evalúa de 5 la calidad de la investigación.
- El 71.42% de los expertos evalúa de 5 la novedad científica de la investigación, el resto lo valora de 4, lo que representa un 94, 28% de aceptación.
- El 85.71% de los expertos evalúa de 5 el aporte científico de la investigación, el resto lo evalúa de 4, lo que representa un 97,14% de conformidad.
- El 57.14% de los expertos evalúa de 5 la facilidad de comprensión, el resto la valora de 4. Para un 91.42% admisión.
- El 71.42% de los expertos evalúa de 5 la adaptabilidad a diferentes entornos de producción de software, el resto lo evalúa de 4, teniendo un 94.28% de aceptación.
- El 71.42% de los expertos evalúa de 5 la repercusión en los proyectos productivos, el resto lo valora de 4, lo que representa un 94, 28 de conformidad.
- El 100 % de los expertos evalúa de 5 la contribución al proceso de desarrollo de software.
- El 100 % de los expertos evalúa de 5 la contribución a la Gestión de Proyectos.

Criterios	Exp	Exp	Exp	Exp	Exp	Exp	Exp
	1	2	3	4	5	6	7
Satisfacción las necesidades de los proyectos productivos.	4	5	5	5	5	5	5
Calidad de la investigación.	5	5	5	5	5	5	5
Novedad científica.	4	5	5	5	5	4	5
Aporte científico.	4	5	5	5	5	5	5
Facilidades de comprensión.	4	5	5	4	5	4	5
Adaptabilidad a diferentes entornos de producción de software.	4	5	5	5	5	5	4
Repercusión en los proyectos productivos.	4	5	5	5	4	5	5
Contribución al proceso de desarrollo de software.	5	5	5	5	5	5	5
Contribución a la Gestión de Proyectos.	5	5	5	5	5	5	5

Tabla 18 Resumen de las respuestas de los expertos según los criterios definidos.

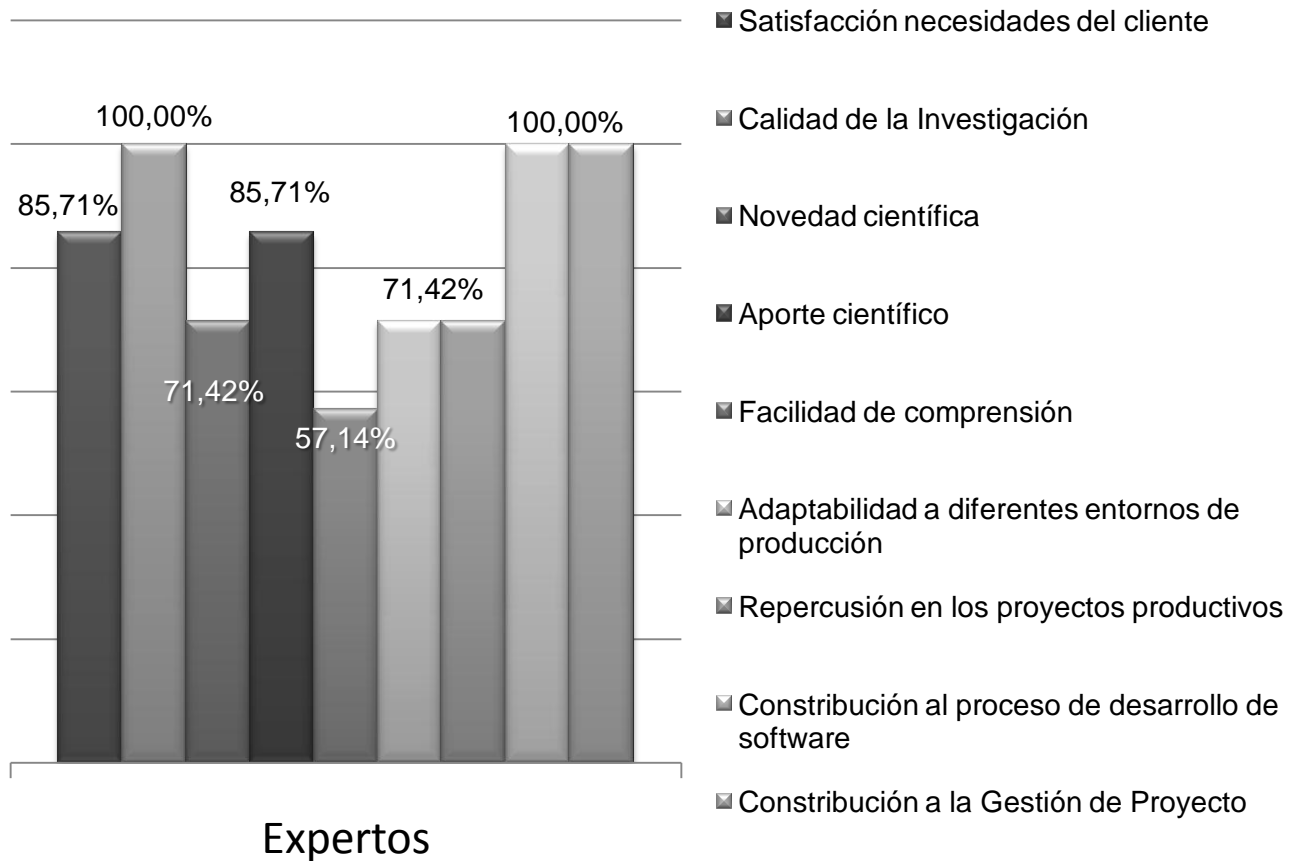


Figura 10 por ciento de respuestas de los expertos según los criterios definidos.

3.8 Conclusiones del capítulo 3.

En el capítulo se realizó un análisis del modelo propuesto, comparándolo con otro proyecto en cuanto a la ingeniería de requisitos lo que demostró ser superior en este aspecto, también se compara con el modelo de calidad ISO-690 y con la metodología de desarrollo RUP, evidenciando que el modelo está a la altura de los mismos y que además tiene otras ventajas. Se demuestra que el modelo es factible, está acorde con los proyectos del Polo Video y Sonido Digital, y sería muy beneficiosa su utilización para el desarrollo de software en el PVSD, según la opinión de los expertos.

CONCLUSIONES

Para darle solución al problema planteado en la investigación, se realizaron cada una de las tareas para alcanzar los resultados esperados. Se efectuó un análisis de los modelos y herramientas de ingeniería de requisitos para optar por el más adecuado a utilizar en los proyectos del PVSD. Se confeccionó un curso de capacitación sobre ingeniería de requisitos para los analistas del polo, y se construyó una multimedia para complementar el mismo. Se le realizaron adaptaciones y mejoras al modelo propuesto (Modelo de Durán). El modelo adaptado fue aplicado al proyecto VideoWeb de dicho polo. Se compararon los proyectos Video Web y Plataforma de TV en cuanto a la gestión de requisitos, demostrando el 1ro su superioridad en este aspecto. La propuesta se evaluó mediante el método científico, método Delphi, los resultados demuestran que los expertos están de acuerdo con la propuesta realizada y le atribuyen gran importancia a su utilización.

- El análisis de los modelos, herramientas y técnicas para llevar a cabo la gestión de requisitos dio como resultado que se escogiera el modelo acorde con las necesidades del Polo Video y Sonido Digital (PVSD).
- Con la realización de adaptaciones y mejoras al modelo propuesto se logró optimizar documentación e incluir temas tan importantes como la gestión de riesgos y la aplicación de métricas para la validación de los requisitos.
- La herramienta OSRMT es la más adecuada para utilizarla en la gestión de requisitos en los proyectos de desarrollo del PVSD.
- Con la propuesta del curso de capacitación para los analistas del PVSD, el cual se apoya en una multimedia con información sobre la ingeniería de requisito, se prepara a los analistas del PVSD de modo que sepan aplicar el modelo propuesto, e incentiva el estudio de la Ingeniería de Requisitos.
- Sin una buena gestión de requisitos, es imposible definir claramente las metas a lograr, no se puede inspeccionar y probar el trabajo de manera adecuada, ni satisfacer a los clientes, en resumen no hay ingeniería profesional sin requisitos bien gestionados.
- Si se almacenan los requisitos con la definición de sus riesgos, pruebas, código, se puede reutilizar en proyectos futuros con un coste mínimo.
- Con un modelo como el que se propone mejorará el desarrollo del flujo de requerimiento de los proyectos del PVSD.

RECOMENDACIONES

- Impartir el curso de capacitación propuesto para que los analistas tengan conocimientos sólidos de cómo aplicar el modelo.
- Publicar el curso en el portal del Polo Video y Sonido Digital.
- Aplicar el modelo a los proyectos del Polo Video y Sonido Digital.
- Verificar por parte de la dirección del Polo Video y Sonido Digital que se aplique el modelo a los proyectos del mismo.
- Utilizar la herramienta propuesta, OSRMT.
- Incluir el tema de los modelos y herramientas de gestión de requisitos en la asignatura de Ingeniería de Software 1, en la UCI.
- Continuar evaluando el modelo con los resultados obtenidos en el polo con su aplicación.

BIBLIOGRAFÍA

Bibliografía citada

Chaves, Michael Arias. 2006. InterSedes. *InterSedes*. [En línea] Universidad de Costa Rica, 7 de Julio de 2006. [Citado el: 1 de Diciembre de 2008.] <http://www.intersedes.ucr.ac.cr>.

Davyt Dávila, Nicolas. 2001. *Ingeniería de Requerimientos: Una guía para extraer, analizar, especificar y validar los requerimientos de un proyecto*. Monte Video : s.n., 2001.

Díaz, Andy y Osmel, Ruíz. 2007. *Proceso de Gestión de Requisitos en el proyecto "Sistema Integrado para Bibliotecas"*. Ciudad Habana : s.n., 2007. pág. 51.

Durán Toro, Amador. 2000. *Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información*. Sevilla : Universidad de Sevilla, 2000.

Duran, Amador. 2001. Dialnet. *Dialnet*. [En línea] 2001. [Citado el: 25 de 2 de 2009.] <http://dialnet.unirioja.es/servlet/tesis?codigo=12692>.

Escalona, M.J. y Koch, N. 2005. *Ingeniería de Requisitos en Aplicaciones para la Web: Un estudio comparativo*. Sevilla : s.n., 2005.

Fernández, Leidy. 2005. Monografías. *Monografías*. [En línea] 2005. [Citado el: 1 de 4 de abril.] <http://www.monografias.com/trabajos41/riesgo-etapa-requisitos/riesgo-etapa-requisitos.shtml>.

Herrera J., Lizka Johany (2003) "Ingeniería de Requerimientos, Ingeniería de Software", Recuperado el 25 de mayo de 2006 en: <http://www.monografias.com/trabajos6/resof/resof.shtml>

Pohl, K. 1997. Encyclopedia of Computer Science and Technology. *Encyclopedia of Computer Science and Technology*. [En línea] 1997. [Citado el: 15 de enero de 2009.] <http://sunsite.informatik.rwth-aachen.de/CREWS/reports96.htm..>

Pressman, Roger. 2003. *Ingeniería de Software, un enfoque práctico*. New York : MacGraw-Hill, 2003.

Pressman, Roger S. 2005. *Ingeniería del Software. Un enfoque práctico.* Ciudad de La Habana : Felix Varela, 2005.

Ramos, Leidy y Socarrás, Renier. 2008. *Propuesta de una herramienta libre para la Gestión de Requisitos en proyectos productivos de la facultad 10.* Ciudad Habana : s.n., 2008.

Real Academia Española. 2001. *Diccionario de la Lengua Española.* 22a Edición. s.l. : Espasa Calpe S.A., 2001. Edición Electrónica.

Menéndez, Rafael. 2008. Universidad de Mursia. *Universidad de Murcia.* [En línea] 25 de noviembre de 2008. [Citado el: 1 de diciembre de 2008.] <http://www.um.es/docencia/barzana/IAGP/IAGP2-Ingenieria-software-introduccion.html1>.

Sánchez, Sergio. 2008. Petra. *Petra.* [En línea] 28 de octubre de 2008. [Citado el: 1 de marzo de 2009.] http://petra.euitio.uniovi.es/~i1942861/HD/Trabajos/Informe_Herramientas_Analisis_y_Gestion_Requisitos_v1_0.pdf.

UCI. 2007. Portal UCI- Universidad de las Ciencias Informáticas. *Portal UCI- Universidad de las Ciencias Informáticas.* [En línea] 2007. [Citado el: 1 de 12 de 2008.] <http://www.uci.cu/?q=node/46> .

Universidad de Murcia. Universidad de Murcia. [En línea] [Citado el: 1 de diciembre de 2008.] <http://www.um.es/docencia/barzana/IAGP/IAGP2-Ingenieria-software-introduccion.html>.

Bibliografía consultada

Álvarez de Zayas, C. 1995. *Metodología de la Investigación Científica.* Stgo. de Cuba : Universidad de Oriente, 1995.

Aportela Rodríguez, I. M. 2007. *Intranets: las tecnologías de información y comunicación en función de la organización.* 2007.

Bayarre, H y Hersford, R. 2004. *Metodología de la Investigación.* Ciudad de La Habana : Ciencias Médicas, 2004.

Cera, Daily y Mompié, Liudmila. 2008. *Validación de un procedimiento para la captura de requisitos en los proyectos de portales web.* Ciudad Habana : s.n., 2008.

- E.Fenton, Norman y Lawrence Pfleeger, Shari. 1998.** *Software Metrics: A Rigorous and Practical Approach*. 1998.
- Fernández, Alicia y Gutiérrez, Mariela. 2008.** *Ingeniería de Requisitos para un Sistema de Control de Medios Técnicos en la ADUANA*. Ciudad Habana: s.n., 2008.
- IBM. 2006.** *IBM Rational Suite family of products*. United States of America : s.n., 2006.
- Larman, Craig. 1999.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos*. México : Prentice Hall, 1999. ISBN: 970-17-0261-1.
- Medero, Maily y Carralero, Katia. 2008.** *Estrategia para el desarrollo y gestión de requisitos en proyectos de Realidad Virtual*. Ciudad Habana : s.n., 2008.
- Mendoza Sanchez, María A. 2007.** *Metodologías De Desarrollo De Software*. Perú : S.A.C., 2007.
- Sigler, Lisset y Salas, Sailyn. 2008.** *Definición y Aplicación de las Técnicas Adecuadas para el Desarrollo de Requisitos en el Simulador Quirúrgico*. Ciudad Habana : s.n., 2008.
- Visual Paradigm.** *UML CASE Tools - Free for Learning UML, Cost-Effective for Business Solutions*. [En línea] [Citado el: 26 de Enero de 2009.] <http://www.visual-paradigm.com/product/vpuml/>.

ANEXOS

Anexo 1 P1 del Curso Ingeniería de Requisitos.

ASIGNATURA: Ingeniería de Requisitos

Secuencia de actividades

No	Semana	Actividad	Descripción
1	1	Conf. 1	TEMA I: La Ingeniería de Requisitos protagonista principal del desarrollo de software. Introducción al curso. Presentación del curso IR. Ingeniería de requisitos, necesidad de gestionar requisitos. Ingeniería de requisitos como disciplina: definiciones y principios. Ejercicio: Hacer mapa conceptual sobre el tema.
2	1	Conf.2	Modelos de Ingeniería de Requisitos. El modelo de Pohl, modelo de Durán, comparación de modelos de procesos de ingeniería de requisitos. Ejercicio práctico: Realizar una propuesta de qué modelo utilizar en el PVSD.
3	2	Seminario 1	La Gestión de Riesgos en la IR.
4	2	Conf. 3	Elicitación y análisis de IR. Técnicas, tareas y plantillas a utilizar en cada una de las etapas. Ejercicio práctico: Realizar la elicitación y el análisis del caso de estudio.
5	4	Conf. 4	Conf. 4: Validación y herramientas de gestión de requisitos. Técnicas, tareas y plantillas a utilizar en la actividad de Validación, y herramientas libres y propietarias de gestión de requisitos.

			Ejercicio práctico: Realizar la validación del caso de estudio.
16	11	Taller Integrador: 4 horas	<p>Presentación de trabajos referativos por equipos:</p> <p>Aplicación del modelo de Durán en los distintos proyectos en desarrollo, utilizando alguna herramienta de gestión de requisitos.</p> <p>Cada equipo presenta un glosario de términos de la asignatura y un mapa conceptual por temas. Aplicar encuesta sobre la asignatura.</p>

Anexo 2 Manual de instalación de la herramienta OSRMT

Índice de contenido

1. Instalación de la versión 5.0 estándar de Java.

Entorno de trabajo.

Es necesaria la instalación de Java puesto que la herramienta está programada sobre lenguaje Java y el instalador es un archivo .jar que necesita ser ejecutado con la máquina virtual de Java.

De las versiones anteriores a la 5.0 de Java solo se puede instalar hasta la 1.5, que permite el funcionamiento de la herramienta OSRMT.

2. Instalación del Cliente OSRMT para la versión 1.5

Instalación del software.

Se descarga del sitio

<http://www.osrmt.com>

La versión osrmt_v1_5.zip

Cuando se extraiga el archivo osrmt.jar se ejecuta mediante líneas de comandos:

```
java -jar osrmt15.jar
```

O simplemente clic sobre el archivo. Ver figura 1.

Para verificar cual versión de Java es la que usted tiene funcionando solo con los comandos en la consola cmd para Windows y un terminal para Linux, podrá determinarla y si es menor a la versión Java1.5 debe ser cambiada y se recomienda la versión Java 5.0.

Si se siente inseguro de cual instalar se recomienda el link JRE.

Para más información visitar el sitio oficial:

<http://java.sum.com>



Figura 1

Seleccione el lenguaje de instalación y presione OK.

Después de leer y estar de acuerdo con el Readme.txt presione siguiente y seleccione la trayectoria que tomara la instalación (Ver figura 2), si lo instala sobre un directorio existente SOBRESCRIBE la base datos y el connection. xml y se pierden los datos ya existentes.



Figura 2

Luego se seleccionan los paquetes que se desean instalar. Ver figura 3

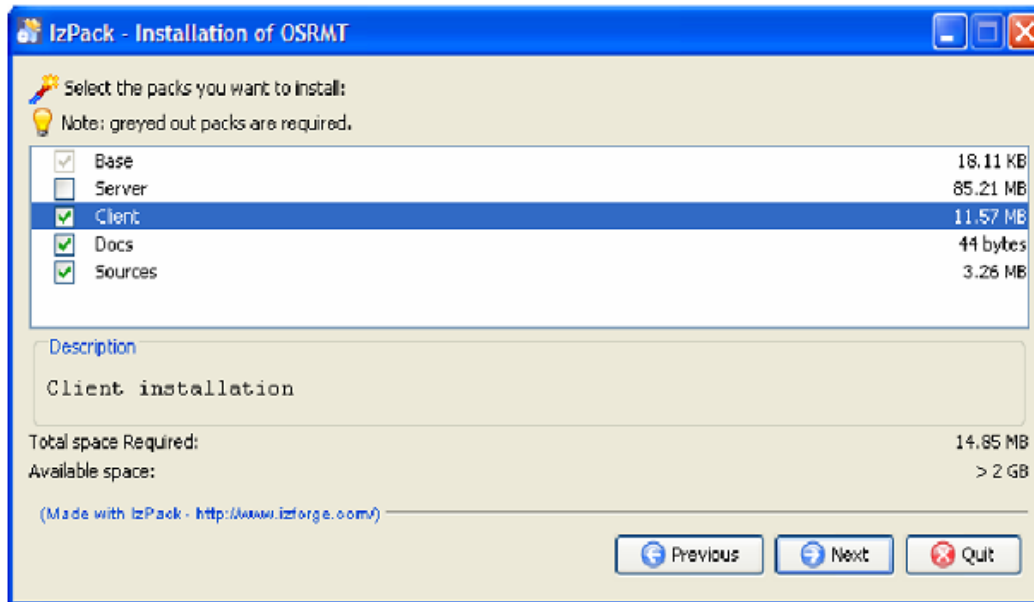


Figura 3

Ejecute el acceso directo creado o solamente vaya al directorio client/ y para Windows se ejecuta run.bat y para Linux se ejecuta run.sh dándole los permisos necesarios a run.sh para la ejecución.

Si se desea se pueden crear accesos directos en el escritorio. Ver figura 4

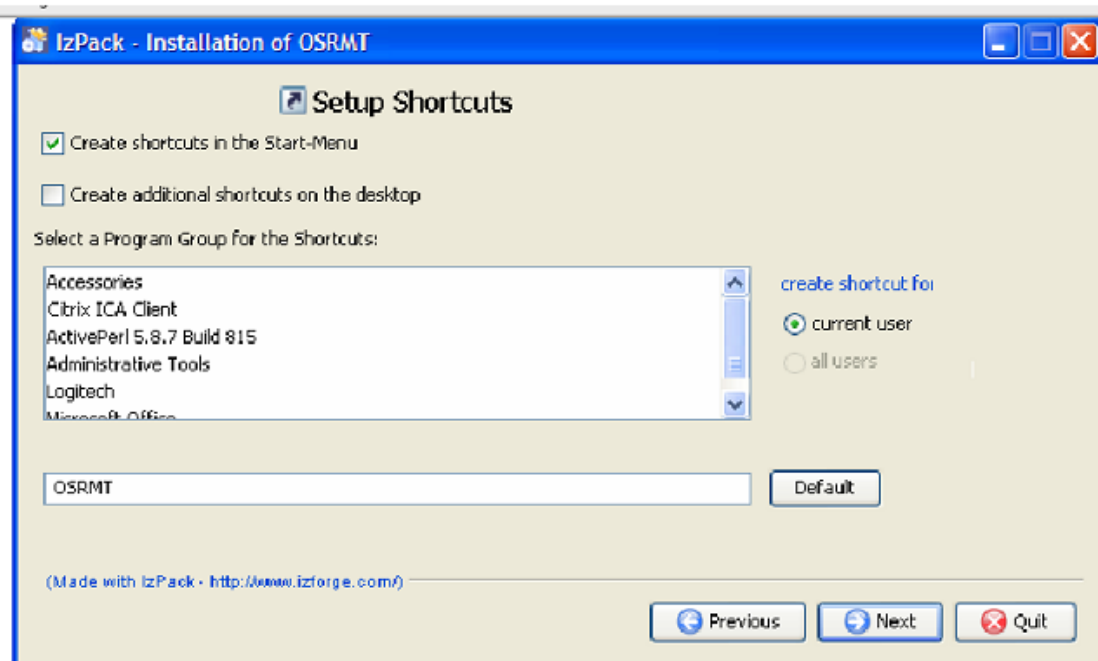


Figura 4

3. Creando nuevas bases datos para su uso.

Utilizando base datos libre para su uso.

Dentro del sistema de directorios, escoger en “client” el fichero “connection.xml” que le interese en función de la base de datos que se desee utilizar. Ver figura 5

Se pueden encontrar:

connection.access.xml

connection.mysql.xml

connection.oracle.xml

connection.postgresql.xml

connection.sqlserver.xml

Basado en que será un sistema para el uso dentro del software libre, se escogió el connection.postgresql.xml y se copia el código dentro de “connection.xml” sin modificar valores específicos a menos que sea necesario para el uso en específico dentro de un ambiente de trabajo. Se pueden mantener las opciones por defecto.

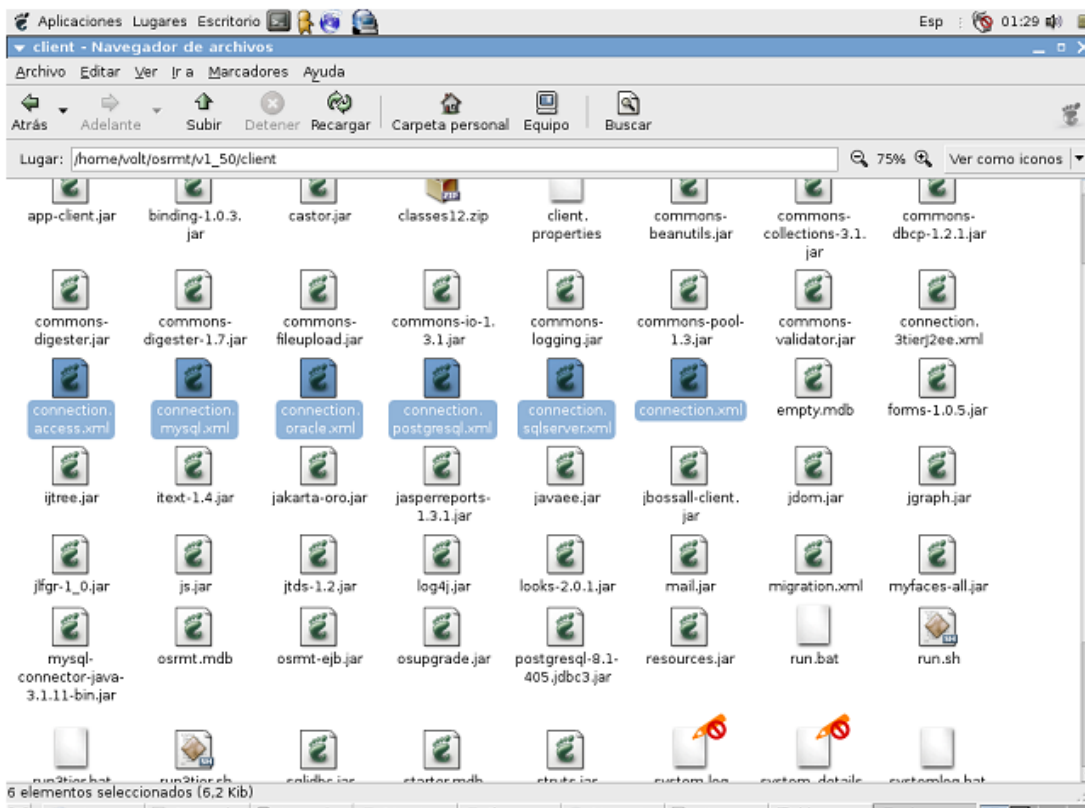


Figura 5

Servidor necesario para el funcionamiento de la base datos.

Se necesita la instalación de PostgreSQL 8.1.x o cualquier otra versión más reciente. Además de utilizar el gestor de bases de datos que sea de su agrado para manejar y mantener el control de la base de datos.

Se propone el uso de pgadmin3. Ver figura 6

También en líneas de comandos se puede hacer el trabajo aunque es más tedioso.

```
user:-$
```

```
user:-$ psql -U postgres template1
```

Ejecutar los script siguiendo el orden

Entrar en el directorio `osrmt/v1_50/client/schema` ahí se encuentran los script específicos según la base de datos a utilizar.

El primer archivo a ejecutar sería `postgresql_create_user.sql` que permitirá establecer cuál va a ser el usuario. De forma visual a través de la herramienta de gestión de base de datos puede ser establecido este usuario.

El segundo archivo a ejecutar sería `postgresql_create_schema.sql` que mediante este se establece el sistema de tablas que tendrá la base de datos necesaria para el buen funcionamiento.

Por último crear la vista asociada `postgresql_create_views.sql`

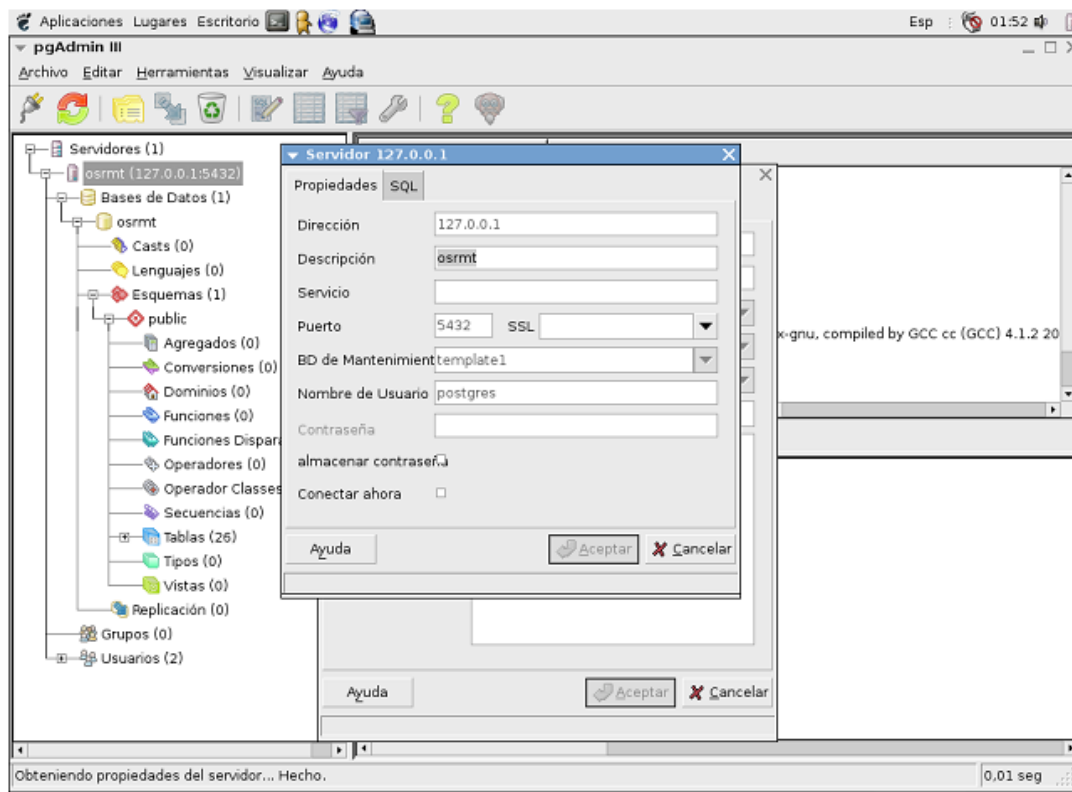


Figura 6

4. Importar contenido de la base de datos

Se corrige `connection.xml` según los pasos siguientes. Para esto se ejecuta el archivo `client\upgrade.sh`

Seleccione la opción 4

Select configuration option 1, 2, 3 or 4

- 1) Define a new connection
- 2) Test the connection
- 3) Save the new connection
- 4) Initialize a new database
- 5) Upgrade 1.3 to 1.4 database
- 6) Migrate database contents
- 7) Export language file
- 8) Import language file
- 9) Exit

Enter option number [Exit]

Se confirma si quiere inicializar la base datos. Ver figura 7.

Enter option number [Exit]: 4

initializing database defined in connection.xml:

```
jdbc:mysql://localhost/osrmt?useUnicode=true&characterEncoding=UTF-8
```

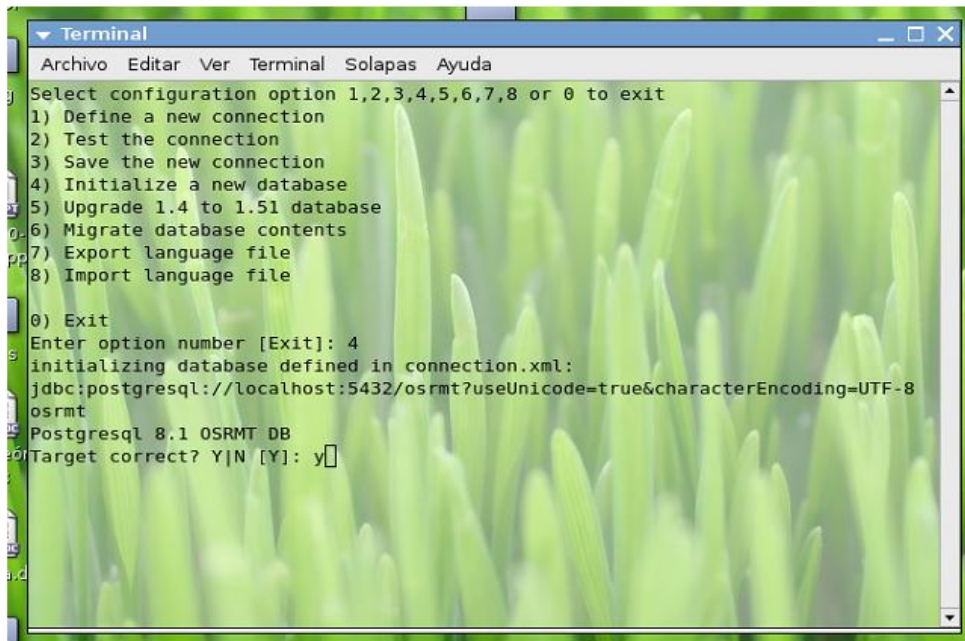
```
osrmt
```

```
pgsql
```

```
Target correct? Y|N [Y]: y
```

```
Empty schema located - initialize and populate schema? [Y]:
```

Solo presione y se actualizaran los datos, luego regresará al menú principal.



```
Terminal
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
Select configuration option 1,2,3,4,5,6,7,8 or 0 to exit
1) Define a new connection
2) Test the connection
3) Save the new connection
4) Initialize a new database
5) Upgrade 1.4 to 1.51 database
6) Migrate database contents
7) Export language file
8) Import language file


0) Exit
Enter option number [Exit]: 4
initializing database defined in connection.xml:
jdbc:postgresql://localhost:5432/osrmt?useUnicode=true&characterEncoding=UTF-8
osrmt
Postgresql 8.1 OSRMT DB
Target correct? Y|N [Y]: y
```

Figura 7


Solo resta iniciar el servidor "server/jboss-4.0.3/bin/run" y comprobar que no hay errores de permisos, ni puertos ocupados.

Ejecutar el cliente "client/run", usuario DEMO/demo, viene con 2 ejemplos vacios que puede utilizar para ver el funcionamiento.

Anexo 3 Prototipos de interfaz de usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF1.1 Adicionar usuario.	

Prototipos de interfaz de usuario del requisito Adicionar usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF1.2 Modificar información de usuario.	


Prototipos de interfaz de usuario del requisito Modificar información de usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF1.3 Modificar rol de usuario.	<div style="text-align: center;"> <p>Modificar Usuario</p> </div> <p>Usuario <input type="text"/></p> <p>Dirección de correo electrónico <input type="text"/></p> <p>Contraseña <input type="text"/></p> <p>Confirmar contraseña <input type="text"/></p> <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p>Estado <input type="radio"/> Bloqueado</p> <p><input type="radio"/> Activo</p> </div> <p>Rol <input type="text" value="--Seleccione--"/> ▼</p> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Modificar usuario"/> </div>

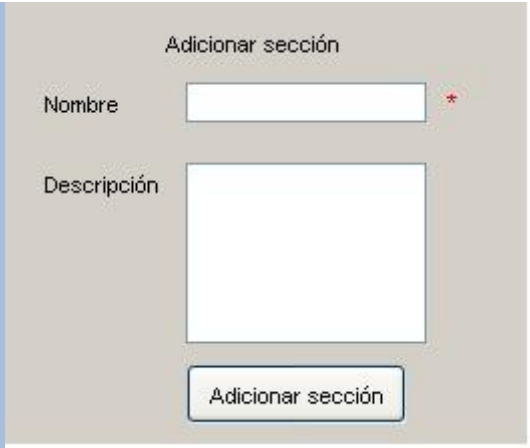
Prototipos de interfaz de usuario del requisito Modificar rol de usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario												
RF1.4 Eliminar usuario.	<div style="text-align: center;"> <p>Eliminar usuario</p> </div> <p>Filtro <input type="text"/> <input type="button" value="Ir"/> <input type="button" value="Restablecer"/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 65%;">Usuario</th> <th style="width: 15%;">Estado</th> <th style="width: 15%;">Rol</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>usuario1</td> <td>Activo</td> <td>Actor</td> </tr> <tr> <td><input type="checkbox"/></td> <td>usuario2</td> <td>Bloqueado</td> <td>Revisor</td> </tr> </tbody> </table> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Eliminar usuario"/> </div>		Usuario	Estado	Rol	<input type="checkbox"/>	usuario1	Activo	Actor	<input type="checkbox"/>	usuario2	Bloqueado	Revisor
	Usuario	Estado	Rol										
<input type="checkbox"/>	usuario1	Activo	Actor										
<input type="checkbox"/>	usuario2	Bloqueado	Revisor										

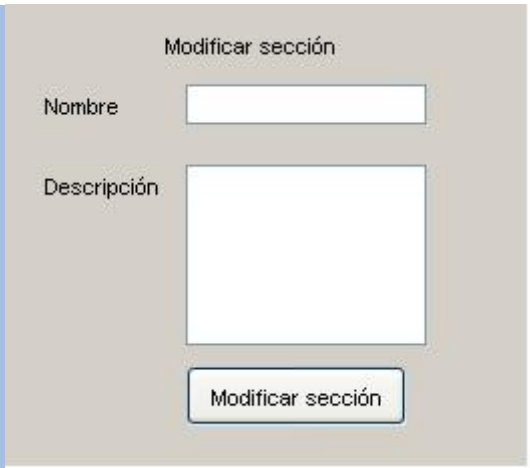
Prototipos de interfaz de usuario del requisito Eliminar usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF2 Autenticar usuario.	 <p>The prototype shows a login form titled 'Iniciar sesión'. It contains two input fields: 'Usuario' and 'Contraseña', both with red asterisks indicating required fields. Below the fields is a button labeled 'Iniciar sesión'.</p>

Prototipos de interfaz de usuario del requisito Autenticar usuario del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF3 Gestionar sesiones. RF3.1 Adicionar sesión.	 <p>The prototype shows a form titled 'Adicionar sección'. It has a 'Nombre' field with a red asterisk and a larger 'Descripción' text area. A button labeled 'Adicionar sección' is at the bottom.</p>


Prototipos de interfaz de usuario del requisito Adicionar sesión del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF3.2 Modificar sesión.	 <p>The prototype shows a form titled 'Modificar sección'. It has a 'Nombre' field and a larger 'Descripción' text area. A button labeled 'Modificar sección' is at the bottom.</p>


Prototipos de interfaz de usuario del requisito Modificar sesión del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF3.3 Eliminar sesión.	

Prototipos de interfaz de usuario del requisito Eliminar sesión del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF4.1 Adicionar Categoría.	

Prototipos de interfaz de usuario del requisito Adicionar Categoría del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF4.2 Modificar categoría.	

Prototipos de interfaz de usuario del requisito Modificar categoría del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF4.3 Eliminar categoría.	

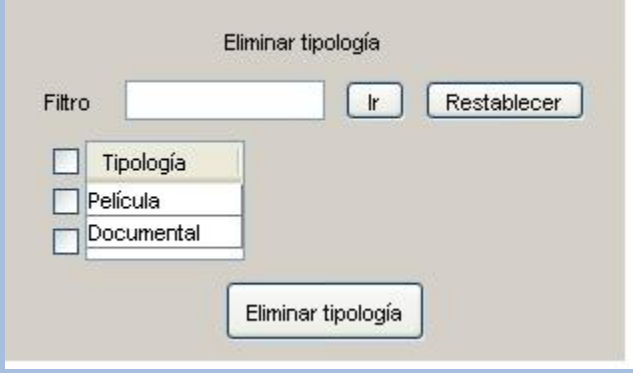
Prototipos de interfaz de usuario del requisito Eliminar categoría del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
F5 Gestionar tipologías de medias. RF5.1 Adicionar tipología de media.	

Prototipos de interfaz de usuario del requisito Adicionar tipología de media del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF5.2 Modificar tipología de media.	

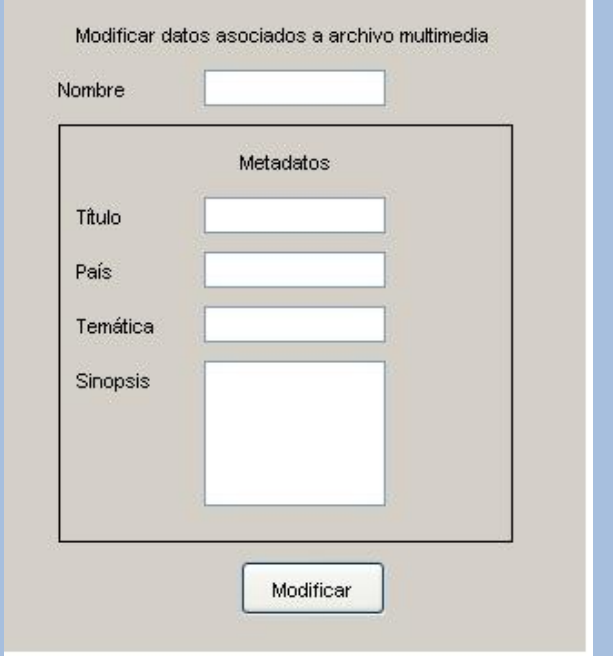
Prototipos de interfaz de usuario del requisito *Modificar tipología de media* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF5.3 Eliminar tipología de media.	


Prototipos de interfaz de usuario del requisito *Eliminar tipología de media* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF6 Gestionar archivos multimedia. RF6.1 Adicionar archivos multimedia.	


Prototipos de interfaz de usuario del requisito *Adicionar archivos multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF6.2 Modificar archivos multimedia.	 <p>Modificar datos asociados a archivo multimedia</p> <p>Nombre <input type="text"/></p> <div style="border: 1px solid black; padding: 5px;"> <p style="text-align: center;">Metadatos</p> <p>Título <input type="text"/></p> <p>País <input type="text"/></p> <p>Temática <input type="text"/></p> <p>Sinopsis <input style="width: 100%; height: 40px;" type="text"/></p> </div> <p style="text-align: center;"><input type="button" value="Modificar"/></p>


Prototipos de interfaz de usuario del requisito *Modificar archivos multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario									
RF6.3 Eliminar archivos multimedia.	 <p style="text-align: center;">Eliminar archivo multimedia</p> <p>Filtro <input type="text"/> <input type="button" value="Ir"/> <input type="button" value="Restablecer"/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 45%;">Archivo</th> <th style="width: 50%;">Tipología</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Código Linux</td> <td>Documental</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Dioses rotos</td> <td>Película</td> </tr> </tbody> </table> <p style="text-align: center;"><input type="button" value="Eliminar archivo"/></p>		Archivo	Tipología	<input type="checkbox"/>	Código Linux	Documental	<input type="checkbox"/>	Dioses rotos	Película
	Archivo	Tipología								
<input type="checkbox"/>	Código Linux	Documental								
<input type="checkbox"/>	Dioses rotos	Película								

Prototipos de interfaz de usuario del requisito *Eliminar archivos multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF7 Gestionar publicaciones multimedia. RF7.1 Crear publicación multimedia.</p>	

Prototipos de interfaz de usuario del requisito *Crear publicación multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF7.2 Modificar publicación multimedia.</p>	

Prototipos de interfaz de usuario del requisito *Modificar publicación multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario															
RF7.3 Eliminar publicación multimedia.	<p style="text-align: center;">Eliminar publicación de archivo multimedia</p> <p>Filtro <input type="text"/> <input type="button" value="Ir"/> <input type="button" value="Restablecer"/></p> <table border="1"> <thead> <tr> <th><input type="checkbox"/></th> <th>Título</th> <th>Sección</th> <th>Categoría</th> <th>Publicado</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Código Linux</td> <td>Videos</td> <td>Documentales</td> <td>Si</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Lista de espera</td> <td>Videos</td> <td>Películas</td> <td>No</td> </tr> </tbody> </table> <p style="text-align: center;"><input type="button" value="Eliminar publicación"/></p>	<input type="checkbox"/>	Título	Sección	Categoría	Publicado	<input type="checkbox"/>	Código Linux	Videos	Documentales	Si	<input type="checkbox"/>	Lista de espera	Videos	Películas	No
<input type="checkbox"/>	Título	Sección	Categoría	Publicado												
<input type="checkbox"/>	Código Linux	Videos	Documentales	Si												
<input type="checkbox"/>	Lista de espera	Videos	Películas	No												

Prototipos de interfaz de usuario del requisito *Eliminar publicación multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario												
RF8 Publicar archivo multimedia.	<p style="text-align: center;">Publicar archivo multimedia</p> <p>Filtro <input type="text"/> <input type="button" value="Ir"/> <input type="button" value="Restablecer"/></p> <table border="1"> <thead> <tr> <th><input type="checkbox"/></th> <th>Título</th> <th>Sección</th> <th>Categoría</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Código Linux</td> <td>Videos</td> <td>Documentales</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Lista de espera</td> <td>Videos</td> <td>Películas</td> </tr> </tbody> </table> <p style="text-align: center;"><input type="button" value="Publicar"/></p>	<input type="checkbox"/>	Título	Sección	Categoría	<input type="checkbox"/>	Código Linux	Videos	Documentales	<input type="checkbox"/>	Lista de espera	Videos	Películas
<input type="checkbox"/>	Título	Sección	Categoría										
<input type="checkbox"/>	Código Linux	Videos	Documentales										
<input type="checkbox"/>	Lista de espera	Videos	Películas										

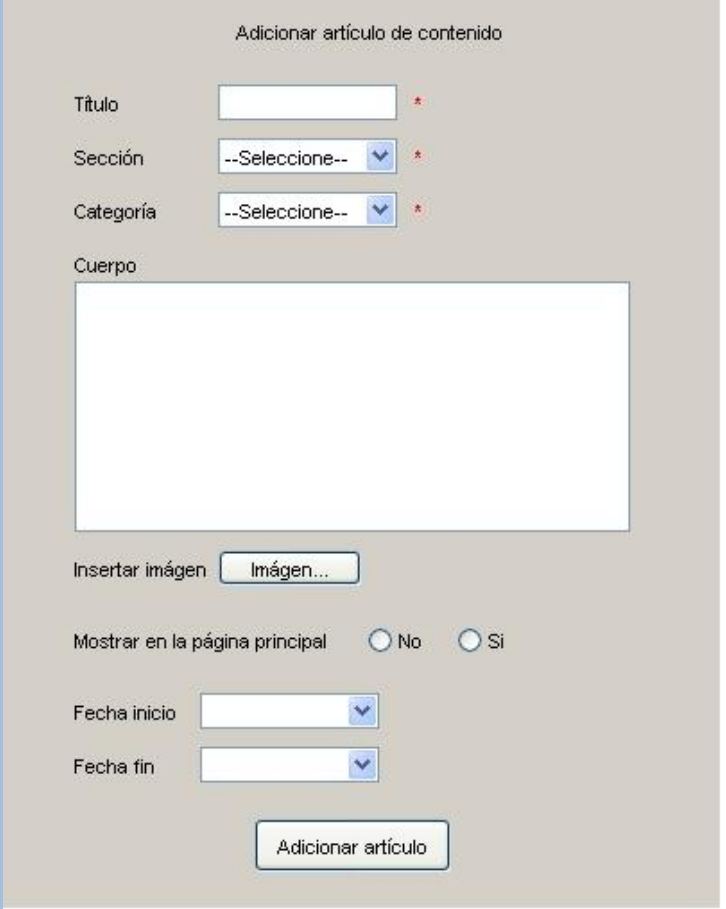
Prototipos de interfaz de usuario del requisito *Publicar archivo multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario												
RF9 Dejar de publicar archivo multimedia.	<p style="text-align: center;">Dejar de publicar archivo multimedia</p> <p>Filtro <input type="text"/> <input type="button" value="Ir"/> <input type="button" value="Restablecer"/></p> <table border="1"> <thead> <tr> <th><input type="checkbox"/></th> <th>Título</th> <th>Sección</th> <th>Categoría</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Código Linux</td> <td>Videos</td> <td>Documentales</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Lista de espera</td> <td>Videos</td> <td>Películas</td> </tr> </tbody> </table> <p style="text-align: center;"><input type="button" value="Dejar de publicar"/></p>	<input type="checkbox"/>	Título	Sección	Categoría	<input type="checkbox"/>	Código Linux	Videos	Documentales	<input type="checkbox"/>	Lista de espera	Videos	Películas
<input type="checkbox"/>	Título	Sección	Categoría										
<input type="checkbox"/>	Código Linux	Videos	Documentales										
<input type="checkbox"/>	Lista de espera	Videos	Películas										

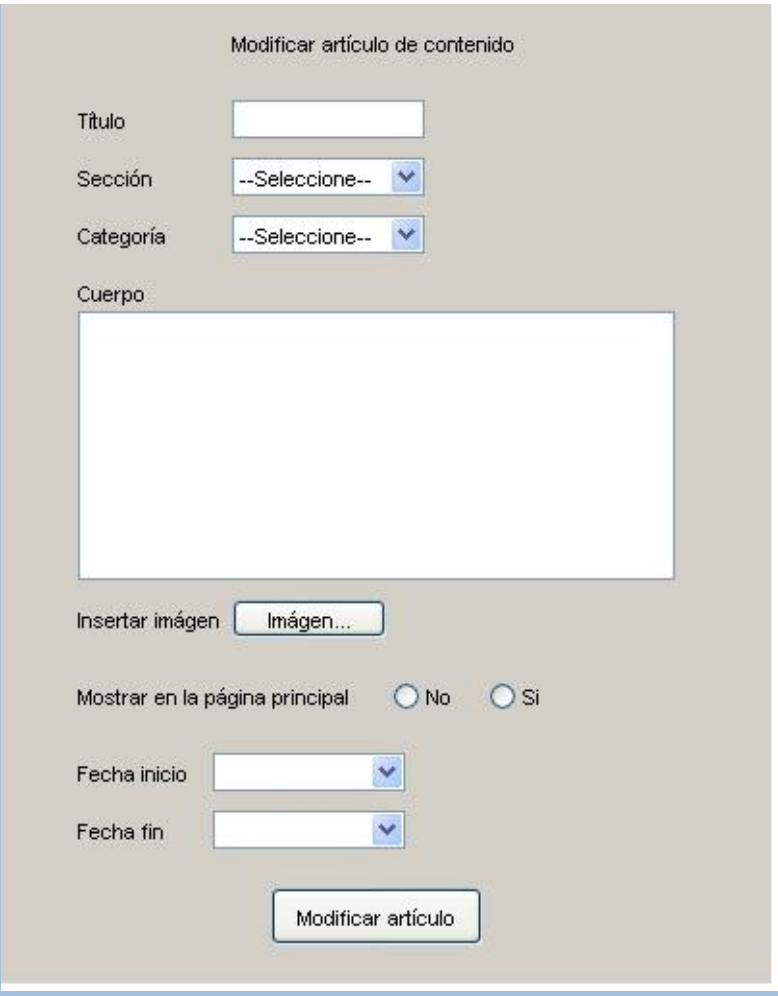
Prototipos de interfaz de usuario del requisito *Dejar de publicar archivo multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF10 Reproducir archivos multimedia.	


Prototipos de interfaz de usuario del requisito *Reproducir archivos multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
F11 Gestionar artículo de contenido. RF11.1 Adicionar artículo de contenido.	

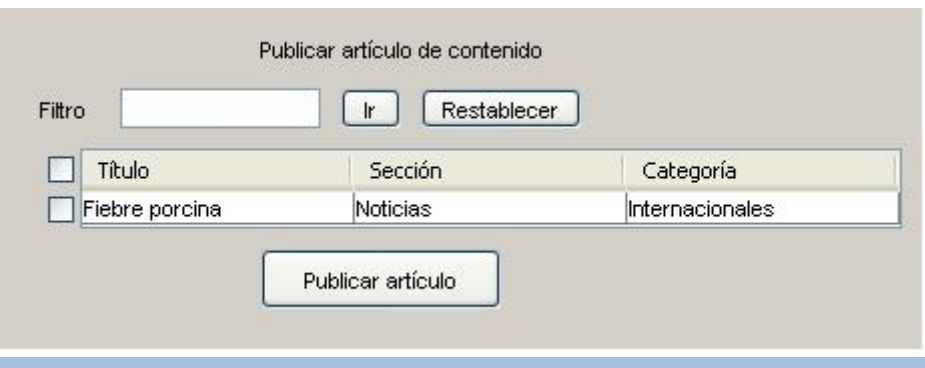
Prototipos de interfaz de usuario del requisito *Adicionar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF11.2 Modificar artículo de contenido.	

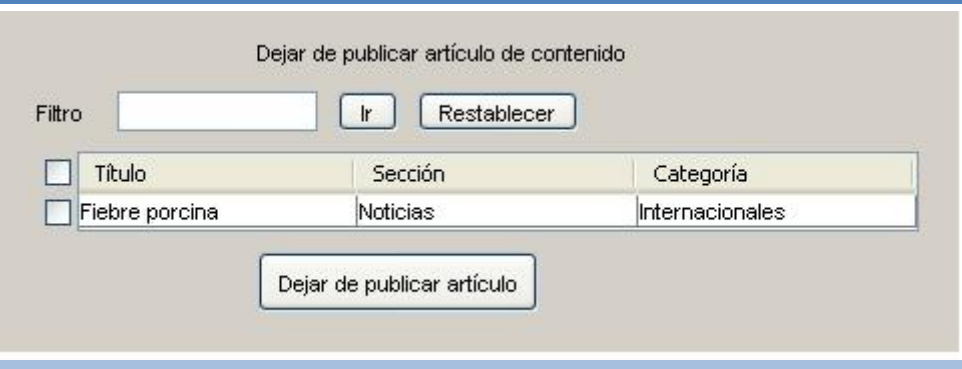
Prototipos de interfaz de usuario del requisito *Modificar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF11.3 Eliminar artículo de contenido.	

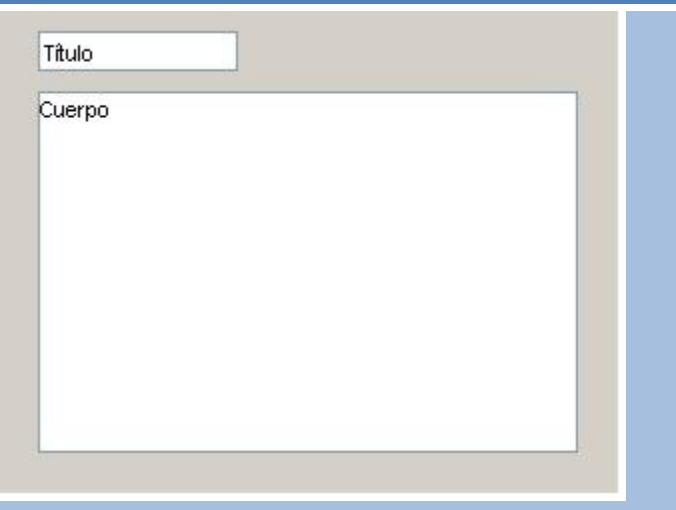
Prototipos de interfaz de usuario del requisito *Eliminar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF12 Publicar artículo de contenido.	

Prototipos de interfaz de usuario del requisito *Publicar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF13 Dejar de publicar artículo de contenido.	


Prototipos de interfaz de usuario del requisito *Dejar de publicar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF14 Visualizar artículo de contenido.	

Prototipos de interfaz de usuario del requisito *Visualizar artículo de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF15 Gestionar lista de reproducción.</p> <p>RF15.1 Crear lista de reproducción.</p>	

Prototipos de interfaz de usuario del requisito *Crear lista de reproducción* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF15.2 Reproducir lista de reproducción.</p>	

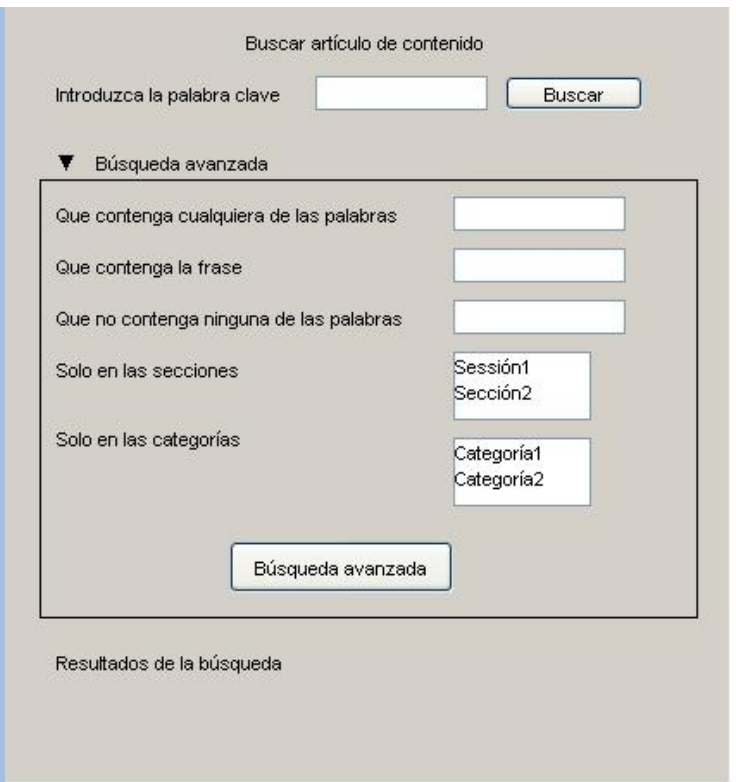
Prototipos de interfaz de usuario del requisito *Reproducir lista de reproducción* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF15.3 Modificar lista de reproducción.	

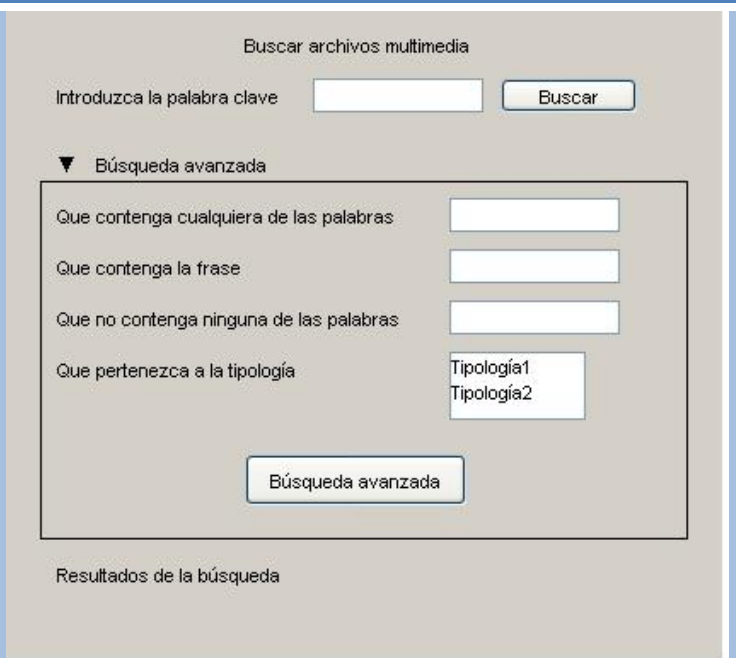
Prototipos de interfaz de usuario del requisito *Modificar lista de reproducción* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
F15.3 Eliminar lista de reproducción.	

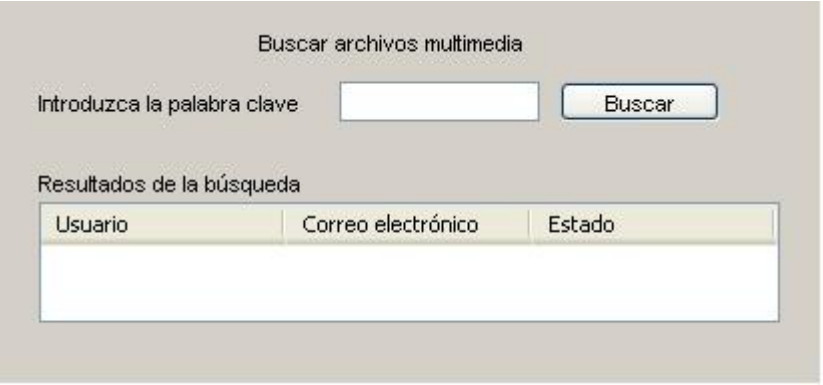
Prototipos de interfaz de usuario del requisito *Eliminar lista de reproducción* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF16. Realizar búsquedas.</p> <p>RF16.1 Realizar búsqueda de artículos de contenido.</p>	 <p>Buscar artículo de contenido</p> <p>Introduzca la palabra clave <input type="text"/> <input type="button" value="Buscar"/></p> <p>▼ Búsqueda avanzada</p> <p>Que contenga cualquiera de las palabras <input type="text"/></p> <p>Que contenga la frase <input type="text"/></p> <p>Que no contenga ninguna de las palabras <input type="text"/></p> <p>Solo en las secciones <input type="text" value="Sesión1"/> <input type="text" value="Sección2"/></p> <p>Solo en las categorías <input type="text" value="Categoría1"/> <input type="text" value="Categoría2"/></p> <p><input type="button" value="Búsqueda avanzada"/></p> <p>Resultados de la búsqueda</p>

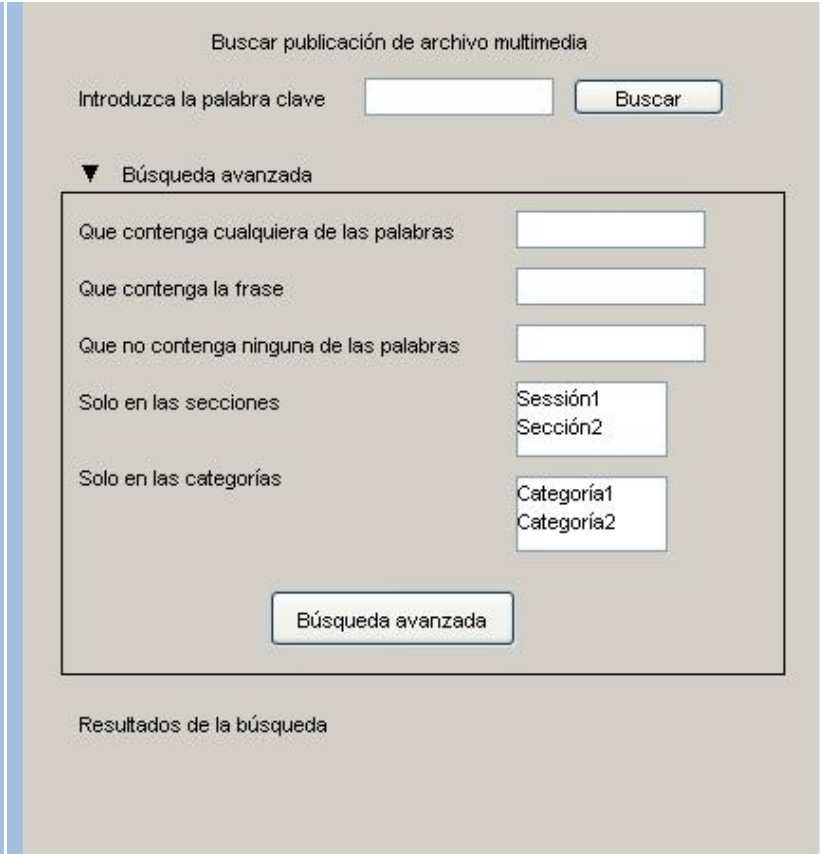
Prototipos de interfaz de usuario del requisito *Realizar búsqueda de artículos de contenido* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF16.2 Realizar búsqueda de archivos multimedia.</p>	 <p>Buscar archivos multimedia</p> <p>Introduzca la palabra clave <input type="text"/> <input type="button" value="Buscar"/></p> <p>▼ Búsqueda avanzada</p> <p>Que contenga cualquiera de las palabras <input type="text"/></p> <p>Que contenga la frase <input type="text"/></p> <p>Que no contenga ninguna de las palabras <input type="text"/></p> <p>Que pertenezca a la tipología <input type="text" value="Tipología1"/> <input type="text" value="Tipología2"/></p> <p><input type="button" value="Búsqueda avanzada"/></p> <p>Resultados de la búsqueda</p>

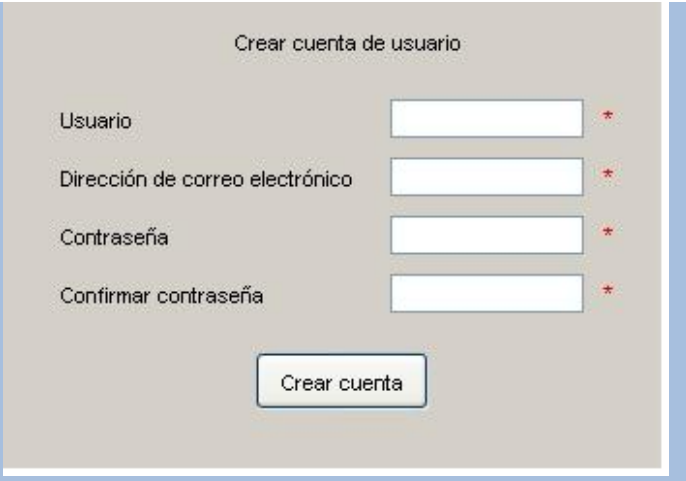
Prototipos de interfaz de usuario del requisito *Realizar búsqueda de archivos multimedia* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF16.3 Realizar búsqueda de los usuarios registrados.</p>	

Prototipos de interfaz de usuario del requisito *Realizar búsqueda de los usuarios* registrados del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
<p>RF16.4 Realizar búsqueda de publicaciones de archivos multimedia.</p>	


Prototipos de interfaz de usuario del requisito *Realizar búsqueda de publicaciones de archivos multimedia* registrados del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
RF17 Crear cuenta de usuario.	


Prototipos de interfaz de usuario del requisito *Crear cuenta de usuario* registrados del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
R18 Gestionar señal en vivo. R18.1 Publicar señal de audio o video en vivo.	


Prototipos de interfaz de usuario del *Publicar señal de audio o video en vivo* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
18.2 Modificar señal de audio o video en vivo.	

Prototipos de interfaz de usuario *Modificar señal de audio o video en vivo* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
R18.3 Eliminar señal de audio o video en vivo.	

Prototipos de interfaz de usuario *Eliminar señal de audio o video en vivo* del proyecto VideoWeb.

Requerimiento	Interfaz de usuario
R19. Cambiar interfaz visual.	

Prototipos de interfaz de usuario *Cambiar interfaz visual* del proyecto VideoWeb.

Anexo 4 Encuesta realizada a los expertos para validar el modelo propuesto.

Usted ha sido seleccionado por su conocimiento en el proceso de desarrollo de software, por sus años de experiencia en proyectos productivos y los resultados alcanzados en su labor profesional, como experto para evaluar los resultados teóricos de esta investigación.

1. ¿Considera usted que la propuesta está a la altura de las necesidades y posibilidades de aplicación de los proyectos productivos del PVSD?

Si___ No___ ¿Por qué?

2. ¿Con la propuesta establecida para erradicar los problemas que existen en el proceso de ingeniería de requisitos, cree usted que se podrá aumentar la efectividad del trabajo?

Si___ No___ ¿Por qué?

3. Considera usted que la propuesta tenga óptimas posibilidades de uso en los proyectos productivos del PVSD.

Si___ No___ ¿Por qué?

4. Le otorga usted importancia a la utilización del modelo propuesto en los proyectos de desarrollo del PVSD.

Si___ No___

5. En una escala del 1 al 5 confiera una evaluación a la propuesta según los siguientes criterios:

___ Satisfacción a las necesidades de los proyectos productivos.

___ Calidad de la investigación.

___ Novedad científica.

___ Aporte científico.

___ Facilidades de comprensión.

___ Adaptabilidad a diferentes entornos de producción de software.

___ Repercusión en los proyectos productivos.

___ Contribución al proceso de desarrollo de software.

___ Contribución a la Gestión de Proyectos.

Anexo 5

Especificación de Requisitos. Plataforma de VideoWeb.

Introducción

La Plataforma de VideoWeb es un sistema web encaminado a la publicación y administración de archivos multimedia, basándose en la tecnología de streaming para brindar estos contenidos audiovisuales a través de la red de datos.

Este documento contiene el listado de los requerimientos funcionales y no funcionales de la Plataforma de VideoWeb, en un lenguaje entendible por el cliente y por los desarrolladores del sistema.

Alcance

Este documento está dirigido a los Analistas y Clientes. Contiene los requerimientos funcionales, que son condiciones o capacidades que el sistema debe cumplir y los no funcionales que son propiedades o cualidades que el sistema debe tener. Por el lenguaje usado en la descripción de los requerimientos pueden considerarse entendibles tanto por los clientes como por los desarrolladores.

Requisitos Funcionales

R1. Gestionar usuarios.		
	Descripción	El sistema debe permitir gestionar los datos referentes a los usuarios.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R1.1 Adicionar usuario.		
	Descripción	El sistema debe permitir crear nuevos usuarios.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R1.2 Modificar información de usuario.		
	Descripción	El sistema debe permitir modificar la información de los usuarios.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R1.3 Modificar rol de usuario.		
	Descripción	El sistema debe permitir modificar los roles de cada uno de los usuarios.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R1.4 Eliminar usuario.		
	Descripción	El sistema debe permitir eliminar los usuarios.

Prioridad	Crítico.
Estimado	Final de la primera versión.

R2. Autenticar usuario.	
Descripción	El sistema debe permitir que los usuarios se autenticuen, una vez chequeado el rol que poseen, se les dará su acceso al sistema.
Prioridad	Crítico.
Estimado	Final de la primera versión.

R3. Gestionar secciones.	
Descripción	El sistema debe permitir a los usuarios con permisos gestionar las secciones.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R3.1 Adicionar sección.	
Descripción	El sistema debe permitir a los usuarios con permisos crear una nueva sección de contenido.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R3.2 Modificar sección.	
Descripción	El sistema debe permitir a los usuarios con permisos modificar los datos de una sección de contenido.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R3.3 Eliminar sección.	
Descripción	El sistema debe permitir a los usuarios con permisos eliminar una sección de contenido.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R4. Gestionar categorías.	
Descripción	El sistema debe permitir a los usuarios con permisos gestionar las categorías.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R4.1 Adicionar categoría.	
Descripción	El sistema debe permitir a los usuarios con permisos crear una nueva categoría de contenido dentro de una sesión existente.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R4.2 Modificar categoría.	
Descripción	El sistema debe permitir a los usuarios con permisos modificar los datos de una categoría de contenido.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R4.3 Eliminar categoría.	
Descripción	El sistema debe permitir a los usuarios con permisos eliminar una

n	categoría de contenido.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R5. Gestionar tipologías de archivo multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos gestionar las tipologías de archivo multimedia.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R5.1 Adicionar tipología de archivo multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos crear una nueva tipología de archivo multimedia y definir los datos que va a incluir dicha tipología.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R5.2 Modificar tipología de archivo multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos modificar los datos de una tipología de archivo multimedia.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R5.3 Eliminar tipología de archivo multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos eliminar una tipología de archivo multimedia.
Prioridad	Crítico.
Estimado	Final de la primera versión.

R6. Gestionar archivos multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos gestionar archivos multimedia y los datos asociados a estos.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R6.1 Adicionar archivos multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos almacenar los archivos multimedia en el servidor de medias y crear las referencias pertinentes en la base de datos con todos los datos asociados a estos.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R6.2 Modificar datos asociados a los archivos multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos modificar los datos asociados a los archivos multimedia.
Prioridad	Crítico.
Estimado	Final de la primera versión.
R6.3 Eliminar archivos multimedia.	
Descripción	El sistema debe permitir a los usuarios con permisos eliminar archivos multimedia con los datos asociados a estos.
Prioridad	Crítico.
Estimado	Final de la primera versión.

R7. Gestionar publicaciones multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos gestionar publicaciones multimedia.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	
R7.1 Crear publicación multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos crear publicaciones multimedia a partir de los archivos multimedia previamente almacenados en el servidor de medias, permitiendo predefinir el tiempo de publicación acotado por un inicio y fin.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	
R7.2 Modificar publicación multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos modificar los datos de la publicación multimedia.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	
R7.3 Eliminar publicación multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos eliminar publicaciones multimedia.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	

R8. Publicar archivo multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos activar las publicaciones multimedia previamente elaboradas, copiando el archivo multimedia asociado a la publicación al servidor de streaming.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	

R9. Dejar de publicar archivo multimedia.		
Descripción	El sistema debe permitir a los usuarios con permisos desactivar las publicaciones multimedia previamente publicadas, eliminando el archivo multimedia asociado a la publicación del servidor de streaming.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	

R10. Reproducir archivos multimedia.		
Descripción	El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario.	
Prioridad	Crítico.	
Estimado	Final de la primera versión.	
R10.1 Reproducir archivos multimedia desde un reproductor externo.		
Descripción	El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario desde un reproductor externo instalado en el sistema.	
Prioridad	Crítico	

	Estimado	Final de la primera versión.
R10.2 Reproducir archivos multimedia desde un reproductor embebido.		
	Descripción	El sistema debe ser capaz de reproducir los archivos multimedia, previamente publicados, especificados por el usuario mediante un reproductor embebido en el navegador.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.

R11. Gestionar artículos de contenido.		
	Descripción	El sistema debe permitir gestionar artículos de contenido.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.
R11.1 Adicionar artículo de contenido.		
	Descripción	El sistema debe permitir a los usuarios con permisos adicionar artículos de contenido permitiendo predefinir el tiempo de publicación acotado por inicio y fin, aunque el artículo no estará publicado hasta que un usuario con permisos de publicación lo publique.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.
R11.2 Modificar artículo de contenido.		
	Descripción	El sistema debe permitir a los usuarios con permisos modificar artículos de contenido.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.
R11.3 Eliminar artículo de contenido.		
	Descripción	El sistema debe permitir a los usuarios con permisos eliminar artículos de contenido.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.

R12. Publicar artículo de contenido.		
	Descripción	El sistema debe permitir a los usuarios con permisos publicar los artículos de contenido previamente adicionados.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.

R13. Dejar de publicar artículo de contenido.		
	Descripción	El sistema debe permitir a los usuarios con permisos dejar de publicar los artículos de contenido previamente publicadas.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.

R14. Visualizar artículo de contenido.		
--	--	--

Descripción	El sistema debe permitir a los usuarios visualizar artículos de contenido previamente publicados.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R15. Gestionar lista de reproducción.

Descripción	El sistema debe permitir gestionar listas de reproducción.
Prioridad	Opcional.
Estimado	Final de la segunda versión.

R15.1 Crear lista de reproducción.

Descripción	El sistema debe permitir a los usuarios crear listas de reproducción.
Prioridad	Opcional.
Estimado	Final de la segunda versión.

R15.2 Reproducir lista de reproducción.

Descripción	El sistema debe permitir a los usuarios reproducir sus listas de reproducción.
Prioridad	Opcional.
Estimado	Final de la segunda versión.

R15.3 Modificar lista de reproducción.

Descripción	El sistema debe permitir a los usuarios modificar sus listas de reproducción.
Prioridad	Opcional.
Estimado	Final de la segunda versión.

R15.3 Eliminar lista de reproducción.

Descripción	El sistema debe permitir a los usuarios eliminar sus listas de reproducción.
Prioridad	Opcional.
Estimado	Final de la segunda versión.

R16. Realizar búsquedas.

Descripción	El sistema debe permitir realizar búsquedas de los contenidos que posee la plataforma.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R16.1 Realizar búsqueda de artículos de contenido.

Descripción	El sistema debe permitir realizar búsquedas de los artículos de contenido publicados en la plataforma.
Prioridad	Secundario.
Estimado	Final de la primera versión.

R16.2 Realizar búsqueda de archivos multimedia.

Descripción	El sistema debe permitir realizar búsquedas de los archivos multimedia almacenados en el servidor de medias.
Prioridad	Crítico.
Estimado	Final de la primera versión.

R16.3 Realizar búsqueda de los usuarios registrados.

Descripción	El sistema debe permitir a los usuarios con permisos realizar
-------------	---

		búsquedas de los usuarios registrados.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.
R16.4 Realizar búsqueda de publicaciones de archivos multimedia.		
	Descripción	El sistema debe permitir realizar búsquedas de publicaciones de archivos multimedia.
	Prioridad	Secundario.
	Estimado	Final de la primera versión.
R17 Crear cuenta de usuario.		
	Descripción	El sistema debe permitir a los invitados crearse una cuenta de usuario.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.

R18 Gestionar señal en vivo.		
	Descripción	El sistema debe permitir gestionar una señal de audio o video que se esté transmitiendo en vivo.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R18.1 Publicar señal de audio o video en vivo.		
	Descripción	El sistema debe permitir a los usuarios con permisos publicar una señal de audio o video en vivo.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R18.2 Modificar señal de audio o video en vivo.		
	Descripción	El sistema debe permitir a los usuarios con permisos modificar una señal de audio o video en vivo previamente publicada.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.
R18.3 Eliminar señal de audio o video en vivo.		
	Descripción	El sistema debe permitir a los usuarios con permisos eliminar una señal de audio o video en vivo previamente publicada.
	Prioridad	Crítico.
	Estimado	Final de la primera versión.

R19. Cambiar interfaz visual.		
	Descripción	El sistema debe permitir a los usuarios cambiar la interfaz visual de la plataforma.
	Prioridad	Opcional.
	Estimado	Final de la primera versión.

Requisitos No Funcionales

Usabilidad

Interfaz amigable, interactiva e intuitiva con el cliente.

El módulo de administración debe caracterizarse por un alto grado de flexibilidad.

El sistema debe ser auto gestionable y dinámico.

El tiempo requerido para que usuarios normales y avanzados sean productivos operando el sistema, debe ser como mínimo 1 mes.

Fiabilidad

El sistema debe estar disponible las 24 horas.

Disponibilidad de visualizar los materiales de media en un reproductor.

El servidor streaming debe mantener buenas prestaciones (demanda de archivos)

Rendimiento

El sistema debe responder en un tiempo relativamente rápido a las peticiones del usuario (menos de 5 segundos)

Soporte

Diseño e implementación de una arquitectura flexible, que permita la fácil integración o desintegración de componentes.

La arquitectura debe soportar migrar la interfaz de usuario sin impactos considerables en re-implementación.

El soporte y/o mantenimiento del sitio no debe detener el servicio.

La capa de acceso a datos, debe soportar una migración del Gestor de Bases de Datos en proyecciones futuras.

Seleccionar un Gestor de Bases de Datos que posibilite migraciones futuras.

Restricciones de diseño

El lenguaje de programación que se debe utilizar es PHP.

La arquitectura que se debe desarrollar es la modelo-vista-controlador.

Se debe utilizar como servidor web: "Apache" y como servidor Streaming: "Darwin Streaming Server" y "Mp4live".

Como sistema operativo se debe recurrir a "Nova-Linux".

Requerimiento de ayuda y documentación

Se debe crear un conjunto de cursos de preparación orientado al usuario para que adquieran conocimientos de cómo utilizar el sistema.

Adquisición de Componentes

Servidor de archivos multimedia con un soporte de almacenamiento de 15 Terabytes.

Servidor de aplicaciones con disco duro SCSI de al menos 500 Gb, 4 Gb de RAM, procesador Intel Quad-Core Xeon a 3.00 GHz y tarjeta de red Gigabit Ethernet.

Dos estaciones de captura con disco duro SCSI de 250 Gb, 2 Gb de RAM, Dual-Core Xeon a 3.00 GHz y tarjeta de red Gigabit Ethernet.

Dos tarjetas de captura HAUPPAGE WINTV-PVR 350 PCI WATCH RECORD TV.

Cámara de video Sony HDR-FX1000E con sus accesorios.

Interfaz

Interfaces de Usuarios

Uso de un reproductor embebido en el navegador para la reproducción de archivos multimedia.

Uso de un reproductor externo instalado en la PC cliente para la reproducción de archivos multimedia.

Interfaz web para la visualización de artículos de contenido.

Interfaces con otros Software

Se reutilizan las funcionalidades brindadas por el Darwin Streaming Server para la transmisión de los archivos multimedia.

Se reutilizan las funcionalidades brindadas por el Mp4live para la captura y transmisión de señales en vivo.

Interfaces de Comunicación

Los protocolos para la comunicación que se deben usar son HTTP entre el cliente y el servidor web, RTSP entre el cliente y el servidor streaming y FTP entre el servidor de medias y el servidor streaming.

Requerimientos de licencias y patentes

Legalizaciones, Copyright, y otras especificaciones legales

La parte cubana tendrá todos los derechos de autor sobre el producto.

GLOSARIO

A

Arqueología de documentos: Es el estudio de todos los documentos que debe tener la organización desde su inicio, es la revisión detallada de los documentos de la organización que se desea informatizar.

Artefacto: Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.

C

Captura de Requisitos (CR): Proceso durante el cual se identifica un problema y se especifica los requisitos que debe cumplir un producto de software.

CMM: Capacity Maturity Model fue desarrollado por el Software Engineering Institute (SEI), su principal premisa es la calidad de un producto determinada por la calidad del proceso utilizado para desarrollarlo y mantenerlo. Es un modelo de referencias sobre buenas prácticas, consolidadas y probadas para el desarrollo y mantenimiento de productos y servicios, cubriendo todo el ciclo de vida, desde la concepción a la entrega y mantenimiento. Integra la Ingeniería de Software, la Ingeniería de Sistemas y la Adquisición de Productos y Servicios

Cuestionario: Lista de preguntas que se proponen con cualquier fin.

E

Entrevista: Acción y efecto de entrevistar o entrevistarse. Vista, concurrencia y conferencia de dos o más personas en lugar determinado, para tratar o resolver un negocio.

Elicitación: La Elicitación consiste en las primeras actividades a realizarse en la Ingeniería de requisitos, aunque esta etapa no se puede “divorciar” de las demás, ya que seguramente se iterará a través de las mismas durante el desarrollo de los requerimientos.

I

IEEE: Institute of Electrical and Electronics Engineers: Importante asociación de técnicos profesionales, con sede en los Estados Unidos. Fue fundada en 1884 y favorece la investigación en campos diversos, como la tecnología aeroespacial, la computación, las comunicaciones y la tecnología biomédica. Promueve la estandarización de normas

Ingeniería de Requerimiento (IR): Se define, como un conjunto de actividades en las cuales, utilizando técnicas y herramientas, se analiza un problema y se concluye con la especificación de una solución (a veces más de una).

Ingeniería de Software (IS): Disciplina o área de la Informática o Ciencia de la Computadora, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resultan problemas de todos tipos.

Ingeniería de Requisitos (IR): Disciplina que se enmarca dentro de la Ingeniería de software, la cual propone métodos, técnicas y herramientas que faciliten el trabajo de definición de lo que se quiere de un artefacto de software.

J

JAD: Joint Application Development desarrollada por IBM en 1977, es una alternativa a las entrevistas individuales que se desarrolla a lo largo de un conjunto de reuniones en grupo durante un período de 2 a 4 días. En estas reuniones se ayuda a los clientes y usuarios a formular problemas y explorar posibles soluciones, involucrándolos y haciéndolos sentirse partícipes del desarrollo.

M

Metodología: La metodología se entenderá aquí como la parte del proceso de investigación que sigue a la propedéutica y permite sistematizar los métodos y las técnicas necesarios para llevarla a cabo.

O

Organización: La organización es una unidad social coordinada, consciente, compuesta por dos personas o más, que funciona con relativa constancia a efecto de alcanzar una meta o una serie de metas comunes. Según esta definición, las empresas productoras, los organismos de los gobiernos locales, estatales y federal son organizaciones.

P

Procedimiento: es el modo de ejecutar determinadas acciones que suelen realizarse de la misma forma, con una serie común de pasos claramente definidos, que permiten realizar una ocupación o trabajo correctamente.

PVSD: Polo Video y Sonido Digital.

R

Requerimiento: Una condición o capacidad necesaria para un usuario para resolver un problema o alcanzar un objetivo.

Requerimiento funcional: Alguna actividad o servicio que debe realizar el software y que puede ser comprobada.

Requerimiento no funcional: Limitación de hardware o software bajo la cual el sistema debe operar.

Rol: Un conjunto de expectativas de conducta asociadas a una persona, un patrón de comportamiento que se espera de quién desempeñe cada puesto, con cierta independencia de la persona que sea.

T

Tormenta de ideas: Es una herramienta de planeamiento que se puede utilizar para obtener ideas a partir de la creatividad de un grupo y con ello resolver un problema. El fundamento del brainstorming es la generación de ideas, en modo individual o en grupo, evitando evaluaciones inmediatas: la investigación científica ha demostrado que este principio es altamente productivo tanto en el esfuerzo individual como en el trabajo de grupo.

U

UCI: Universidad de las Ciencias Informáticas

X

XP: Extreme Programming (Programación Extrema), es la más destacada de los procesos ágiles de desarrollo de software. Utilizada para proyectos de corto plazo. Consisten desarrollos rápidos e iterativos, cuya particularidad es tener como parte del equipo al usuario final.