

**UNIVERSIDAD DE LAS CIENCIAS
INFORMÁTICAS
Facultad 9**



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

Título: Propuesta de Guía Electrónica de Programación para un Sistema IPTV en la Universidad de las Ciencias Informáticas. Rol Analista.

Autor: Juan José Milanés Álvarez

Tutor: Ing. Víctor Frank Molina López

Co-tutor: Ing. Abel Díaz Berenguer

**Ciudad de la Habana, junio 2010
"Año del 52 Aniversario del Triunfo de la Revolución"**

“El porvenir de un hijo es siempre obra de su madre”.

Napoleón Bonaparte

Dedicatoria

*El presente trabajo está dedicado a cinco de las personas más importantes de mi vida:
Mi madre por darme todo su amor, confianza, dedicación y comprensión, por su apoyo en los
momentos más difíciles, a ti te debo mi vida, gracias por existir.*

*Mis abuelos Isis y René por ser tan buenos y comprensivos, por ayudarme y apoyarme
siempre, los quiero mucho, ustedes son mis segundos padres.*

*Mi abuela Nancy por su cariño, preocupación y apoyo. Por toda la fe de sus rezos que me
ayudaron mucho.*

*Mi abuelo Iven que no está presente pero sé que me apoyaría y se sentiría orgulloso de su
nieto mayor.*

Agradecimientos

Primeramente agradecerle a mi madre todo lo que ha hecho para que yo pudiera salir adelante en mi carrera.

A mis abuelos por su apoyo incondicional.

Al Villa por ser tan atento conmigo, ayudarme en todo lo que ha podido, mereces mi respeto y mi admiración. Te agradezco especialmente el cariño y amor que le has dado a mi mamá en todo el tiempo que he estado lejos de ella.

A Yunelis por ser tan buena amiga, compañera y novia, por tener tanta paciencia, quererme y apoyarme. Te quiero mucho. Y su familia gracias por quererme y darme tanto ánimo.

A mi papá por el apoyo que me ha dado, con este trabajo culmina la promesa que hicimos los dos un día.

A mi tío Alexis, Anita y Amandita a ustedes le debo mucho, por ayudarme, guiarme y apoyarme en todo momento. En especial a mi tío que se que estas orgulloso de mi como yo lo estoy de ti aunque estés pasando por un momento de tu vida que no mereces y te apoyaré hasta el final como mismo lo has hecho conmigo.

Al resto de mi familia gracias por apoyarme y darme tanto aliento, eso me ayudó mucho.

A Víctor como tutor, Abelito y Ernesto por ayudarme tanto con este trabajo, por sus sugerencias, y por ser tan buenos amigos. Gracias.

A mis amigos de Bayamo que siempre estuvieron preocupados y al tanto de mi tesis. Jorge y su familia, Albert e Isa y mi hermanita Katy.

A los samuráis que son mis amigos desde el principio y han sido incondicional todo el tiempo. Erlys, el Yoa, Albert, el Levi, el Duni ustedes saben que los aprecio mucho.

A mis hermanitas la China, Esther, Greisy, no me han fallado nunca. Las quiero.

A mis amigos del aula Fale, Yosiel, Yumar, Yake, el Mello, Pacheco, Yaillet, Olivier, Yuyi y todos demás que hemos vividos momentos excepcionales de nuestras vidas juntos.

A todos muchas gracias, los quiero mucho.

Declaración de Autoría

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

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

Juan José Milanés Álvarez

Víctor Frank Molina López

Datos de contacto

Tutor

Nombre y apellidos: Víctor Frank Molina López

Sexo: X ___ M ___ F Institución: UCI

Dirección de la institución: Carretera a San Antonio de los Baños, km 2 ½, Boyeros, Ciudad de La Habana.

Correo electrónico: vfmolina@uci.cu

Teléfono del trabajo: Teléfono particular: 052252358

Categoría docente: Adiestrado

Grado científico:

Cargo del trabajador:

Título de la especialidad de graduado: Ingeniero en Ciencias Informáticas

Año de graduación: 2008

Institución donde se graduó: Universidad de las Ciencias Informáticas

Co-tutor

Nombre y apellidos: Abel Díaz Berenguer

Sexo: X ___ M ___ F Institución: UCI

Dirección de la institución: Carretera a San Antonio de los Baños, km 2 ½, Boyeros, Ciudad de La Habana.

Correo electrónico: aberenguer@uci.cu

Teléfono del trabajo: Teléfono particular:

Categoría docente: Adiestrado

Grado científico:

Cargo del trabajador:

Título de la especialidad de graduado: Ingeniero en Ciencias Informáticas

Año de graduación: 2009

Institución donde se graduó: Universidad de las Ciencias Informáticas

Resumen

En la Universidad de las Ciencias Informáticas (UCI), específicamente en la facultad 9, se encuentra el departamento de Señales Digitales, donde se ubican todos los proyectos de televisión de la universidad. Uno de estos es el de IPTV el cual surge dado al desarrollo tecnológico constante en que está inmersa la UCI y los beneficios que trae para los usuarios la creación de una plataforma de televisión digital por IP. Se decidió crear un sistema IPTV porque sus características se adecuan más que los demás tipos de televisión digital a la estructura y condiciones que se tienen en la institución. Un sistema IPTV brinda sencillez y comodidad en sus servicios, tiene una gran variedad de contenidos, presenta mejoras sustanciales en la calidad de la imagen y el sonido, y lo más importante posee aplicaciones de un alto nivel de interactividad entre el usuario y la televisión. La Guía Electrónica de Programación es un subsistema muy importante de este sistema porque es el encargado de visualizar los contenidos y la interacción con el usuario mediante su interfaz.

Esta investigación se centra en la realización de un análisis del entorno de dominio de la propuesta de una Guía Electrónica de Programación. Para ello se hizo un estudio de soluciones existentes en el mundo, así como del proceso de funcionamiento de varios subsistemas de este tipo, para lograr un mejor entendimiento y comenzar a realizar el análisis. En la realización de éste se utilizó como metodología de desarrollo de software a RUP, como lenguaje de modelado a UML y para modelar Visual Paradigm como herramienta CASE.

Debido a que este tema es nuevo y un poco engorroso, aparte de que no se contaba con información suficiente para realizar un proceso del negocio, se decidió realizar un modelo de clases conceptuales del dominio. Esto trajo consigo que se pudieran definir los requerimientos funcionales y no funcionales, se realizaron los diagramas de casos de usos del sistema, de clases del análisis y de colaboración. Además, se aplicaron las métricas de calidad a los artefactos generados en el análisis.

Abstract

The Digital Signs Department is located in the University of Informatics Sciences, precisely in nine faculty, where all university TV projects are registered. It can be mentioned IPTV as one of them, this project came alive due to the technological development undergone by the University, and also due to the benefits, the creation of a digital TV platform by IP, offer the customers.

It was taken the decision to create an IPTV system because its characteristics fit even more to the structure and condition of the institution than any other digital TV. An IPTV system offers simplicity and comfort in the service. It has got a great variety of contents, better quality of the image and sound, and the most important aspect is that it has got a high level of interactivity among the customers and the TV. The electronic guide of programming is a very important subsystem of this system because it is the one in charge of visualizing the contents and the interaction with the customer by means of the interface.

This research work pretends to carry out an analysis of the domain environment of a programming electronic guide. An study of the possible solutions already got along the world as well as the functioning process of some subsystems of this type was carried out to achieve a better understanding and starting the analysis.

RUP was the software developing methodology used, UML was the modeling language, visual Paradigm was the one used to model and CASE was the tool. Just because this item is new, and also because there was not enough information to do a business process, it was decided to conceive a model of conceptual classes of the domain that lead to the functional and non functional requirement definitions, the diagrams of use cases of the system, analysis and collaboration classes were done, and the quality metrical was applied to the artifacts generated from the analysis.

Índice de Figuras

Figura 1: Interacción de la EPG con los subsistemas	22
Figura 2: Visualizar contenidos.....	22
Figura 3: Listar contenidos	23
Figura 4: Autenticación del usuario.....	23
Figura 5: Fases y flujos de trabajos de RUP	38
Figura 7: Diagrama de clases conceptuales del dominio.....	49
Figura 8: Diagrama de Casos de Uso del Sistema	56
Figura 9: Diagrama de clase del análisis del CU Gestionar EPG	64
Figura 10: Diagrama de clase del análisis del CU Listar contenidos	64
Figura 11: Diagrama de clase del análisis del CU Visualizar contenidos	64
Figura 12: Diagrama de colaboración Personalizar EPG- CU Gestionar EPG.....	65
Figura 13: Diagrama de colaboración Eliminar EPG-CU Gestionar EPG	65
Figura 14: Diagrama de colaboración Actualizar EPG-CU Gestionar EPG	66
Figura 15: Diagrama de colaboración Búsqueda EPG-CU Gestionar EPG.....	66
Figura 16: Diagrama de colaboración Listar EPG general-CU Listar contenidos	67
Figura 17: Diagrama de colaboración Listar EPG personalizada-CU Listar contenidos.....	67
Figura 18: Diagrama de colaboración Pre-visualizar EPG-CU Visualizar contenidos.....	67

Índice de Tablas

Tabla 1: Diferencias entre metodologías ágiles y tradicionales.....	34
Tabla 2: Descripción de los actores del sistema	55
Tabla 3: Descripción del CU Gestionar EPG	61
Tabla 4: Descripción del CU Listar Contenidos	62
Tabla 5: Descripción del CU Visualizar Contenidos.....	63
Tabla 6: Métricas aplicadas al Diagrama de Casos de Uso del Sistema.....	71

Índice

Introducción	11
Capítulo 1: Fundamentación Teórica.....	16
Introducción.....	16
1.1 Conceptos asociados al dominio	16
1.2 Televisión Digital (TVD, DTV)	16
1.3.1 Televisión Digital vía Satélite.....	16
1.3.2 Televisión Digital por Cable	17
1.3.4 Televisión Digital por tecnologías IP (ADSL)	18
1.4 Guía Electrónica de Programación (EPG)	19
1.4.1 Metadatos	20
1.5 Proceso de funcionamiento de un sistema IPTV.....	20
1.6 Interacción de la EPG con los subsistemas.....	21
1.6.1 Visualización de los contenidos.....	22
1.6.2 Listar contenidos	23
1.6.3 Autenticación del usuario	23
1.7 Situación Problemática	24
1.8 Soluciones existentes	24
1.9 Conclusiones parciales.....	26
Capítulo 2: Tendencias y tecnologías actuales	27
2.1 Introducción.....	27
2.2 Política de migración al software libre.....	27
2.2.1 Software libre.....	27
2.2.2 Ventajas del software libre	28
2.2.3 Desventajas del Software Libre.....	29
2.3 Tendencias actuales y tecnologías	29
2.4 Metodología de desarrollo	32
2.4.1 Extreme Programming, XP.....	33
2.4.2 Rational Unified Process (RUP)	37
2.5 Selección de la metodología a utilizar.....	40
2.6 El analista de Sistema	41
2.6.1 El analista de Sistema en RUP	41
2.7 ¿Qué es UML?	42
2.8 Herramientas CASE	44
2.8.1 Rational Rose Enterprise	44
2.8.2 Visual Paradigm.....	45
2.9 Selección de la herramienta CASE a utilizar.....	46
2.10 Conclusiones parciales.....	47
Capítulo 3: Análisis de la solución propuesta	48
3.1 Introducción.....	48
3.2 Modelo del dominio	48
3.3 Definición de las clases conceptuales del modelo del dominio	49

3.3.1 Administración	49
3.3.2 Guía	49
3.3.3 Planificación.....	49
3.3.4 Canales.....	49
3.3.5 Paquetes.....	49
3.3.6 Contenido	49
3.3.7 Clasificación.....	50
3.3.8 Audiovisual	50
3.3.9 Audio	50
3.3.10 Imagen.....	50
3.3.11 Otros	50
3.3.12 Mediador.....	50
3.3.13 Usuario	50
3.4 Breve descripción del diagrama de dominio	50
3.5 Requerimientos	51
3.5.1 Requerimientos funcionales.....	51
3.5.2 Requerimientos no funcionales	52
3.6 Descripción del Sistema. Modelo de Casos de Usos del Sistema (MCUS).....	54
3.6.1 Modelo de Casos de Uso del Sistema	55
3.7 Descripción textual de los Casos de Uso del Sistema	56
3.7.1 Gestionar EPG.....	56
3.7.2 Listar Contenidos	61
3.7.3 Visualizar Contenidos	62
3.8 Diagramas de clases del análisis.....	63
3.8.1 Diagrama de clase del análisis del CU Gestionar EPG	64
3.8.2 Diagrama de clase del análisis del CU Listar contenidos	64
3.8.3 Diagrama de clase del análisis del CU Visualizar contenidos.....	64
3.9 Diagramas de colaboración del análisis.....	65
3.10 Métricas.....	68
3.11 Métricas de calidad aplicadas a los artefactos generados	68
3.12 Conclusiones parciales.....	71
Conclusiones generales.....	72
Recomendaciones	73
Bibliografía citada	74
Bibliografía consultada.....	75
Anexos.....	77
Glosario de términos.....	79

Introducción

La Universidad de las Ciencias Informáticas (UCI), tiene como principal misión formar y preparar profesionales de excelencia, ingenieros capaces de llevar en todo momento los ideales de la Revolución, de engrandecer y desarrollar cada vez más nuestra sociedad. Los medios de comunicación juegan un papel importante en esta tarea, por su capacidad para difundir información lo que favorece la consolidación de valores y principios.

Entre los principales medios de comunicación en la universidad se encuentra la televisión. La misma es una parte imprescindible en la vida diaria de la comunidad universitaria ya que es de gran importancia para la información, recreación y preparación profesional de los estudiantes y profesores, así como el desarrollo de la institución.

En la UCI se utiliza la televisión analógica, por la magnitud de esta y lo apartado que se encuentra de los centros de emisión televisiva existe un sistema que utiliza una antena comunitaria que está ubicada a gran altura y provee la señal que se distribuye por toda la comunidad mediante un cable coaxial. La televisión analógica por cables tiene características tradicionales de forma tal que el usuario solo tiene la posibilidad de bajar y subir el volumen y realizar cambios de canales, acciones que el usuario realiza que son propias del dispositivo de televisión y no modifican el contenido que se transmite.

Actualmente la televisión ha tenido un gran desarrollo, principalmente en los países desarrollados ya se utiliza la Televisión Digital por el valor agregado que brinda a los usuarios, mejora la calidad de la imagen y el sonido, aumenta las opciones en cuanto a la cantidad de canales, posee una alta definición y las posibilidades de crear aplicaciones interactivas. Existen cuatro formas de Televisión Digital que se relacionan a continuación:

- Televisión Digital vía Satélite
- Televisión Digital por Cable
- Televisión Digital Terrestre (TDT)
- Televisión Digital por IP

La comunidad universitaria de la UCI cuenta con un sitio web llamado Inter-nos que brinda la

posibilidad de escuchar y visualizar las medias en línea mediante la tecnología de streaming¹ (viene del inglés) utilizada para aligerar la descarga y ejecución de audio y vídeo en la web, ya que permite reproducir los archivos mientras se están descargando.

En este sentido, se utiliza el streaming en directo para la reproducción de materiales en vivo, en este caso el servidor controla la transmisión de datos y el cliente solamente puede escuchar y visualizar las medias. Además, se utiliza el streaming bajo demanda, pues el archivo multimedia una vez grabado se aloja en el servidor y es posible reproducirlo en cualquier momento, permitiendo al cliente controlar la transmisión y recepción del fichero.

El sitio inter-nos, aunque facilita de alguna forma la interactividad entre el usuario y el contenido que el mismo reproduce, no favorece la total interoperabilidad necesaria entre el usuario y el contenido multimedia.

En la universidad se cuenta con computadoras en las aulas, laboratorios, en los comedores, apartamentos y todas están conectadas de la red, debido a ello que la forma de televisión digital que más se ajusta a estas condiciones es la Televisión Digital por IP. Para las otras formas no están creadas las condiciones dada la situación económica existente en el país, ya que los software que se deben utilizar son propietarios, también la dependencia de las tecnologías y licencias a utilizar son de gran costo.

En el nuestro centro no se puede utilizar la televisión digital vía satélite porque se necesita un servicio para conectar al satélite que resultaría muy costoso para la universidad y para el país, además de una antena parabólica que tiene que estar correctamente orientada al satélite de comunicación, un dispositivo de selección de bandas y amplificación denominado LNB² (Low Noise Block por sus siglas en inglés) y de un sintonizador de canales digitales o un decodificador.

La televisión digital por cable se hace un poco complicada dado el tamaño de la universidad y la estructura del cableado coaxial existente, el cual no está acorde a los estándares de transmisión de televisión digital planteado por la unión internacional de telecomunicaciones(UIT) por lo que para implementar esta variante habría que invertir en una reestructuración del cableado, también se necesita un descodificador externo, instalado en cada dispositivo, para poder acceder a los canales por lo que no cumple con la norma exigida por nuestra institución.

¹ Streaming: Se utiliza para aligerar la descarga y ejecución de audio y video en la web, ya que permite escuchar y visualizar los archivos mientras se están descargando.

²El Bloque de Bajo Ruido o LNB, por sus siglas inglesas, es un dispositivo utilizado en la recepción de señales procedentes de satélites.

La televisión digital terrestre (TDT) necesita un área de cobertura que el país no dispone, las antenas UHF (Ultra High Frequency por sus siglas en inglés) individuales o colectivas que se necesitan para captar la señal con adaptaciones en su instalación, para eso se necesita un personal técnicamente autorizado. También es necesaria la existencia de un dispositivo externo de descodificación o de lo contrario un dispositivo de televisión con un receptor integrado, por todas estas razones con las cuales el país no puede lidiar, no se puede utilizar TDT en la universidad.

Un sistema IPTV brinda sencillez y comodidad en sus servicios, una gran variedad de contenidos, soporta un número mayor de canales, a diferencia de otras formas de televisión que no se basan en Internet. Reduce los costos en infraestructura y recursos al mismo tiempo que emite sus contenidos de forma rápida y legible, ofrece una programación individual para cada usuario, así como nuevas formas de publicidad más interactiva.

También trae aparejado desventajas por utilizar tecnología con altos costos que nuestra institución no puede comprar. Dado que la transmisión de los datos ocurre mediante la red por protocolos IP puede ocurrir congestión, lo que trae consigo que se pierda información de la que se transmite. Otra desventaja es que es necesario tener una red IP que pueda soportar una plataforma de este tipo. Dadas todas las ventajas, que son mayores en número que las desventajas en la valoración y evaluación para la creación del sistema, surge la idea de realizar un sistema IPTV en la UCI.

El sistema IPTV tiene varios subsistemas que lo componen, uno de los principales, es el subsistema de la Guía Electrónica de Programación (EPG por sus siglas en inglés Electronic Programming Guide), este es un servicio algo similar a un teletexto³ pero mejorado, centrado en la programación de los canales de televisión y permite conocer la programación diaria de los canales mediante un navegador.

La Guía Electrónica de Programación tiene varias funciones que contribuyen con la vitalidad de esta. Entre ellas se encuentran: la creación, búsquedas, actualización, mantenimiento, importación de archivos y actualización de las listas de programas.

En la creación de la EPG interactúan para la información básica los parámetros: canales, lenguajes, nombre completo de la guía, la sinopsis del programa y la clasificación. Para la planificación de la información que se transmitirá se utilizan los parámetros: el período de planificación de los programas, el tiempo de inicio de programa y el tiempo de duración de este (2).

³El teletexto es un servicio de información en forma de texto que se emite junto con la señal de televisión.

Para las búsquedas en la EPG se utilizan los parámetros: canales, lenguaje de la guía, lenguaje del usuario, y la fecha. La búsqueda se hace solamente por canales, se selecciona el canal y se hace una consulta con las condiciones que necesita. Después de un tiempo el sistema muestra todos los programas con las condiciones demandas en ese canal. (2)

Para el proceso de actualización se seleccionan los programas que se van actualizar, se puede hacer una modificación de la configuración del programa y actualizarlo, se puede eliminar el programa y actualizar o copiar un nuevo programa y su posterior actualización. (2)

En la EPG se hace el mantenimiento semanal donde se selecciona un canal y se le establece la fecha y hora de inicio de la EPG semanal para ser consultado. Se pueden hacer modificaciones de la configuración a la EPG de acuerdo con las condiciones reales, eliminar la EPG y actualizar. (2)

Para el trabajo con los archivos se realiza la función de importación de archivos de la EPG, se añade la creación de archivos de una EPG a la EPG semanal. En la EPG de gestión se muestran todos los archivos que se pueden importar y se pueden hacer pruebas a la EPG, si un archivo está disponible, mediante el parámetro Validar. Se pueden ver detalles de un archivo mediante el parámetro Ver. Y finalmente se puede importar un archivo mediante el parámetro Cargar. (2)

Después de la publicación de un canal es necesario actualizar la EPG para que los suscriptores puedan ver mediante el STB el nuevo canal. Esta se actualiza con el parámetro Confirmar y seguido un tiempo el sistema actualiza toda la EPG. (2)

Dado que la UCI está inmersa en un constante proceso de desarrollo tecnológico, el grupo de IPTV se ha dado la tarea de implementar un sistema de este tipo. La EPG es un subsistema importante junto con el middleware⁴ de un sistema IPTV, que va estar basado en una plataforma totalmente libre. Lo que impone la necesidad de hacer un análisis detallado de este subsistema.

Por lo que se deriva el siguiente problema a resolver: ¿Cómo analizar el entorno del dominio del proceso de visualización de los contenidos en los terminales de un sistema IPTV? Como objeto de estudio se tienen los métodos, procesos y herramientas para el análisis de un entorno del dominio. Y nos enmarcamos en los métodos, procesos y herramientas para el análisis del entorno del dominio en el proceso de visualización de los contenidos en un Sistema IPTV como el campo de acción de este trabajo.

⁴El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.

Se traza como objetivo general elaborar la documentación técnica del análisis del entorno del dominio de una Guía Electrónica de Programación en un sistema IPTV y la idea a defender consiste en: la elaboración de la documentación técnica correspondiente al análisis del entorno de dominio, garantizará un mejor entendimiento del proceso de visualización de los contenidos en un sistema IPTV.

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

- Caracterizar el estado actual de los procesos de visualización de los contenidos en sistemas IPTV.
- Argumentar el uso de la metodología de desarrollo a utilizar, lenguaje de modelado y herramienta CASE.
- Caracterizar el rol Analista según la metodología seleccionada.
- Elaborar la documentación técnica correspondiente al Modelo del Dominio.
- Capturar los requisitos funcionales y no funcionales.
- Elaborar la documentación técnica correspondiente al Modelo de Casos de Uso del Sistema.
- Elaborar la documentación técnica correspondiente al Modelo de Análisis.
- Aplicar métricas para la evaluación de la calidad de la documentación técnica generada.

Métodos Teóricos:

- Histórico – lógico: Con el objetivo de caracterizar las etapas y tendencias actuales del desarrollo de sistemas IPTV en Cuba y en el mundo.
- Inducción – deducción: permitirá generalizar el conocimiento adquirido y realizar el análisis de la EPG del sistema IPTV.
- Modelación: Con el objetivo de describir los distintos procesos, mediante la creación de los modelos elaborados en el proceso de desarrollo del software.
- Análisis y síntesis: Permitirá definir las herramientas, plataformas y tecnologías necesarias para realizar el análisis de la EP.G del sistema IPTV.

1

Capítulo 1: Fundamentación Teórica

Introducción

En este capítulo se brinda información sobre los conceptos asociados al dominio del tema, así como una caracterización general de lo que es una Guía Electrónica de Programación. Funcionalidades de a Guía Electrónica de Programación como: visualización de los contenidos, listar los contenidos y autenticar usuario. También nos da la posibilidad de ver el proceso de funcionamiento de un sistema IPTV y soluciones existentes de estos sistemas que presentan Guías Electrónicas de Programación aunque no se obtiene el funcionamiento de esta en cada una de las soluciones porque no son reveladas por sus proveedores.

1.1 Conceptos asociados al dominio

En este epígrafe se abordarán conceptos asociados al dominio del tema como televisión digital, televisión digital vía satélite, televisión digital por cable, televisión digital terrestre, televisión digital por IP, EPG, metadatos, entre otros elementos. Que ayudarán a un mayor entendimiento del trabajo.

1.2 Televisión Digital (TVD, DTV)

La televisión digital es el conjunto de tecnologías de transmisión y recepción de imagen y sonido que se transmite en formato digital, en contraste con la televisión estándar que codifica la señal de manera analógica. La televisión digital codifica la señal de forma binaria en ceros y unos. Proporciona una mayor calidad en la señal. Evita interferencia o fallos en la transmisión. Habilita la posibilidad de crear vías de retorno entre el usuario y el productor de los contenidos. Facilita aplicaciones interactivas para los consumidores. (3)

1.3.1 Televisión Digital vía Satélite

Televisión digital vía satélite es el resultado de la aplicación de la tecnología digital a la señal de televisión. Para luego transmitirla a una amplia zona geográfica por medio de satélites de comunicaciones, en contraste con la televisión terrestre, cuyas ondas no salen de la atmósfera, o la televisión por cable, basada en la transmisión a través de redes de fibra óptica y cable coaxial.

La transmisión de televisión digital vía satélite se divide en dos tramos claramente diferenciados:

- El enlace ascendente o uplink, mediante el cual el centro emisor envía las señales de televisión al satélite utilizando grandes antenas parabólicas (de 9 a 12 metros de diámetro).
- Y el enlace descendente, o downlink, por medio del cual el satélite retransmite la señal de televisión recibida hacia su zona de cobertura sobre la superficie de la tierra, utilizando una banda de frecuencias diferente de la del enlace ascendente, para evitar interferencias. (1)

1.3.2 Televisión Digital por Cable

Televisión digital por cable es el resultado de la aplicación de la tecnología digital a la señal de televisión, para luego distribuirla por medio de redes híbridas de fibra óptica y cable coaxial. Junto con la señal de televisión digital, a través de estas redes se proporcionan otros servicios como radio, telefonía fija y acceso a internet.

Las redes utilizadas en la distribución de este tipo de servicios se dividen en cuatro secciones:

- Cabecera: es el centro de la red encargado de agrupar y tratar los diversos contenidos que se van a transmitir por la red. Cuentan con antenas parabólicas para recibir señales satelitales, antenas de alta ganancia para televisión abierta.
- Red troncal: transporta la señal a la zona que requiere del servicio. Busca garantizar la calidad de la señal, utilizando equipos amplificadores. Cuenta con un mapa de distribución para la red antes de comenzar a construir y se deben considerar los lugares por donde pasará la red.
- Red de distribución: se conecta a la red troncal mediante amplificadores puentes, su propósito es llevar las señales hasta las instalaciones del suscriptor.
- Red de acometida hacia los abonados: es un cable coaxial flexible utilizado para llevar la señal desde el cable de distribución hasta el lugar donde se encuentra el usuario. Las acometidas requieren de una conexión al sistema de tierra de la construcción y de un cable flexible entre la entrada y el receptor. (1)

1.3.3 Televisión Digital Terrestre

La televisión digital terrestre (TDT) es el resultado de la aplicación de la tecnología digital a la señal de televisión. Para luego transmitirla por medio de ondas hercianas terrestres. Es decir, aquellas que se transmiten por la atmósfera sin necesidad de cable o satélite y se reciben por medio de antenas de frecuencia ultra alta (UHF) convencionales. (1) Se escoge este concepto porque es el que define más claramente la TDT agregándole que la señal que se transmite por las ondas hercianas se

descodifica de forma binaria lo que trae consigo que se comprima la señal y el ancho de banda sea menor, entonces la transmisión va a ser más rápida y con mayor calidad de imagen y sonido.

1.3.4 Televisión Digital por tecnologías IP (ADSL)

La televisión digital por ADSL (Bucle de Abonado Digital Asimétrico, siglas en español) es el resultado de la aplicación de la tecnología digital a la señal de televisión. Para luego transmitirla por medio de protocolos asimétricos (XDSL por sus siglas en inglés). Se suministra junto con el servicio de conexión a internet. Es proporcionado por un operador de banda ancha sobre la misma infraestructura pero con un ancho de banda reservado.

El ADSL es una técnica de transmisión de banda ancha. Es aplicada sobre los bucles de abonado de la red telefónica. Lo que permite la transmisión sobre ellos de datos a alta velocidad. Para ello, se utilizan frecuencias más altas que las empleadas en el servicio telefónico, para de este modo no interferir con ellas. Permitiendo así el uso simultáneo del bucle para el servicio telefónico y para la transmisión de datos. (1)

Los operadores de televisión digital por ADSL ofrecen este servicio de forma separada o en paquetes combinados. Pudiendo además accederse a diferentes configuraciones de canales y a servicios de pago por visión. (1)

Existen varios subsistemas interrelacionados para poder ofrecer IPTV. Estos son:

- Middleware: es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.
- Head End (HE): son los elementos de la cabecera que se encargan de recibir las señales de televisión en vivo y de convertirlas al formato que sea necesario para su transmisión por la red y su posterior recepción por el STB.
- Servidor de video a bajo demanda: encargado de distribuir los archivos de audio y video, siguiendo la demanda de peticiones de los usuarios.
- Guía Electrónica de Programación (EPG): es una aplicación interactiva que se asemeja a un teletexto pero más avanzado. Con una alta calidad visual de sus gráficos y textos.
- Set Top Box (STB): es un dispositivo externo que se conecta al televisor y se encarga de recibir y descodificar la señal analógica a señal digital para transmitirla al dispositivo de televisión.

- Sistema de control de derechos (DRM): se encarga de encriptar los contenidos de manera tal que no exista vulnerabilidad alguna de los derechos de la propiedad intelectual de los contenidos que se transmiten.

1.4 Guía Electrónica de Programación (EPG)

La EPG es el acrónimo inglés de Electronic Program Guide, en español guía electrónica de programación, es una aplicación interactiva que se asemeja a un teletexto pero más avanzado. Con una alta calidad visual de sus gráficos y textos. Presenta como objetivo centrarse en la programación de canales. Esta es de las múltiples opciones que nos brinda la televisión digital. En ella encontramos una gran cantidad de canales de forma rápida y sencilla. Permite conocer la programación diaria de las cadenas de televisión mediante un navegador intuitivo a través de un mando a distancia. Brinda la posibilidad de crear programaciones individuales para cada usuario. Así este puede seleccionar cualquier canal sin tener que utilizar el llamado zapping⁵, por lo incómodo que le resultaría por el número de canales existentes en la televisión digital. Esta proporciona la facilidad de búsqueda, los canales favoritos, recordatorios y la ayuda.

Esta contiene funciones como cambiar idiomas, cambiar los colores de la pantalla, reordenar los canales, cambiar el número de los canales, todo para una mayor interactividad para el cliente. La EPG es una de las bases de la competencia en condiciones equitativas, no discriminadas y razonables por su capacidad de dirigir al usuario hacia ciertos programas y servicios.

Las tres grandes tipologías de servicios de información y navegación de la programación son:

- Guía de recepción: es un identificador de canal con una presentación mínima de número y nombre del canal. Se muestra automáticamente al sintonizar un canal.
- Guía de canal o identificativo de servicio (Service Id): es lo que determina el identificador de servicio (Service Id) ilustrando los contenidos de la programación y opciones del usuario dentro de los diferentes niveles de prestaciones. Se accede al presionar una tecla del mando a distancia del Set Top Box y tras una detallada información sobre el programa en emisión, duración del evento, características fundamentales de su tipo de emisión (versiones, subtítulos, audios, clasificación por edades). Su navegación está limitada a dos servicios fundamentales: primero, detalle de descripción del evento de emisión y segundo, siguiente programa en emisión. Según modelos y operadores, este servicio también puede informar de otros programas de canales en un entorno

⁵Zapping: Cambio manual de canales.

multicanal. La información que facilita este servicio está imbricada en su trama de emisión y por ello simultánea al programa

- Guía de distribución: es un servicio de información de toda la oferta de programación accesible a un usuario a través de una plataforma de recepción multicanal, incluyendo la oferta en un periodo más extenso que el mismo día de emisión (lo frecuente es de 3 a 7 días). Este servicio es técnicamente independiente de la trama de emisión de los canales que describe. La información está dispuesta en multiniveles de prestaciones y diseño según el editor del servicio y el entorno tecnológico en el que opere (por ejemplo, con función Personal Video Recorder (PVR) permite la grabación automática de canal y el programa). En un análisis del mercado nacional y en el internacional se ven las diferentes modalidades al respecto: desde países que por consenso han lanzado una guía de distribución (Reino Unido) a aquellos que no la tienen (Italia y Francia). (3)

1.4.1 Metadatos

Metadato es la información descriptiva y su estructura para un tipo de contenido dado que será adquirida, catalogada y servida alrededor de la solución IPTV. Este describe la información propia de cada contenido. Contiene los elementos básicos de este como el proveedor del mismo, título, ranura de tiempo para eventos en vivo, vistas de pago, clasificación, sinopsis, duración, precio de ventas para el suscriptor, licenciamiento, director, entre otros. Para contenidos en vivo o para difusión (Broadcast TV) el contenido del Metadato será el listado de programación.

La gestión del metadato debería comprender:

- Actualización, modificación y baja periódica de los listados de programación en vivo. Se deberá disponer de capacidad para almacenar un mes de programación.
- Posibilidad de introducir, modificar y eliminar información descriptiva de contenidos de VoD (video bajo demanda).
- Posibilidad de introducir, modificar y eliminar información descriptiva de eventos en la modalidad vistas de pago.

1.5 Proceso de funcionamiento de un sistema IPTV

El proceso de funcionamiento de un sistema IPTV es una parte imprescindible en la investigación que se está realizando. Este sistema está integrado por varios subsistemas como son el middleware, la cabecera o Head End, el subsistema de video a bajo demanda, el Set Top Box, el subsistema de control de derechos y la EPG.

El middleware (MW) hace función de sistema operativo porque es el encargado de gestionar todo el flujo de datos. Es el que indica cómo interactúa cada subsistema. Define la forma en que el usuario interactúa con el servicio IPTV y soporta la interacción con los distintos servidores de aplicaciones. El Head End son los elementos de la cabecera que se encargan de recibir las señales de televisión en vivo y de convertirlas al formato que sea necesario para su transmisión por la red y su posterior recepción por el STB. El subsistema de video bajo demanda (VoD) tiene como función principal almacenar y transmitir mediante la red videos y pistas de audio que sean solicitadas por los usuarios para ser recibidos a demanda.

El Set Top Box (STB) es un equipo que se conecta con el dispositivo de televisión y tiene como función principal decodificar las señales que se transmiten como flujos de información multimedia por los protocolos IP para que sean compatibles con la televisión digital. Cada STB dispone de un control remoto que será utilizado por el usuario para dar las órdenes al sistema. El subsistema de control de derechos (DRM) se encarga de encriptar los contenidos de manera tal que no exista vulnerabilidad alguna de los derechos de la propiedad intelectual de los contenidos que se transmiten. La guía electrónica de programación proporciona la visualización de los contenidos de la programación de los canales. La selección y búsqueda de programas en los canales en vivo.

Todos estos subsistemas interactúan entre sí, mediante el mediador del middleware que administra todo el flujo de demandas y respuestas que se generan cuando el usuario necesita visualizar o listar los contenidos. Este gestiona toda la información que aparece en el sistema, la EPG por medio de este interactúa con el STB cuando le envía la información y este la almacena en su memoria y cuando se hacen cambios en la EPG este los envía hacia todos los demás STBs para que se actualicen.

La cabecera o Head End capta la señal de la televisión en vivo por canales y la EPG por medio del middleware le hace una petición al head end dada la exigencia del usuario. El servidor bajo demanda es el que almacena toda la información de las medias y brinda la respuesta de las peticiones que el usuario hace a la EPG mediante el middleware. La EPG gestiona la parte de la autenticación del usuario por lo que interactúa mediante el middleware con el subsistema de control de derechos (DRM) para obtener la información encriptada por este.

1.6 Interacción de la EPG con los subsistemas

Este trabajo se centra en la EPG, la cual es una vía de interactividad del usuario con la televisión digital, esta constituye una parte importante en el proceso de funcionamiento de la plataforma IPTV.

En este proceso interactúan todos los subsistemas mediante el mediador del middleware que es el que maneja el flujo de datos e indica como interactúa y se relaciona cada subsistema. En la siguiente figura se muestra la relación que existe entre varios subsistemas, para visualizar y listar los contenidos donde se observa las relaciones de la EPG mediante el middleware con los servidores de video bajo demanda y servidores de aplicaciones y para la autenticación del usuario, cómo interactúa la EPG con el sistema DRM mediante el middleware.

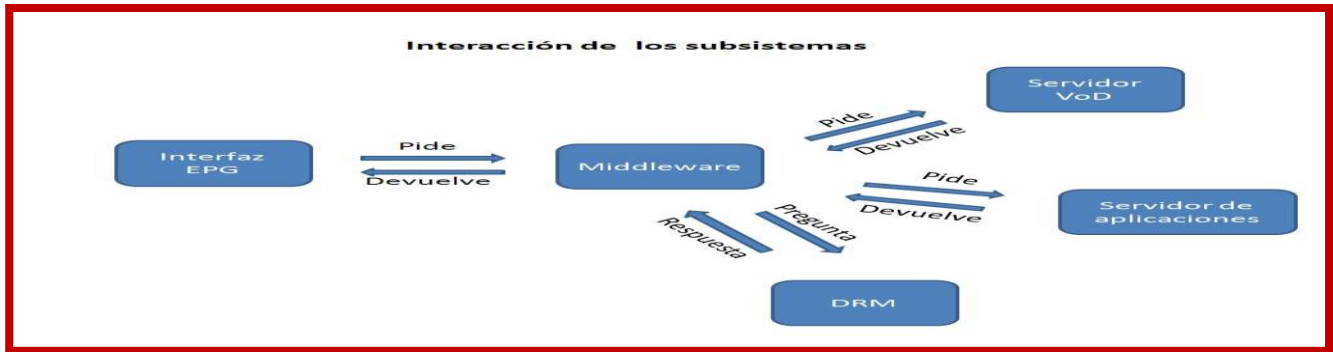


Figura 1: Interacción de la EPG con los subsistemas

En los próximos subepígrafes se desglosa esta figura en varias partes como son: Visualización de los contenidos, Listar los contenidos y Autenticar usuario, para un mejor entendimiento del contenido de estos procesos.

1.6.1 Visualización de los contenidos

El cliente cuando necesita visualizar los contenidos la interfaz hace un pedido al middleware y este a su vez le manda el pedido al servidor de video bajo demanda, este busca el contenido y devuelve la respuesta hacia el middleware con el contenido deseado por el cliente, esto se muestra en la figura 2.

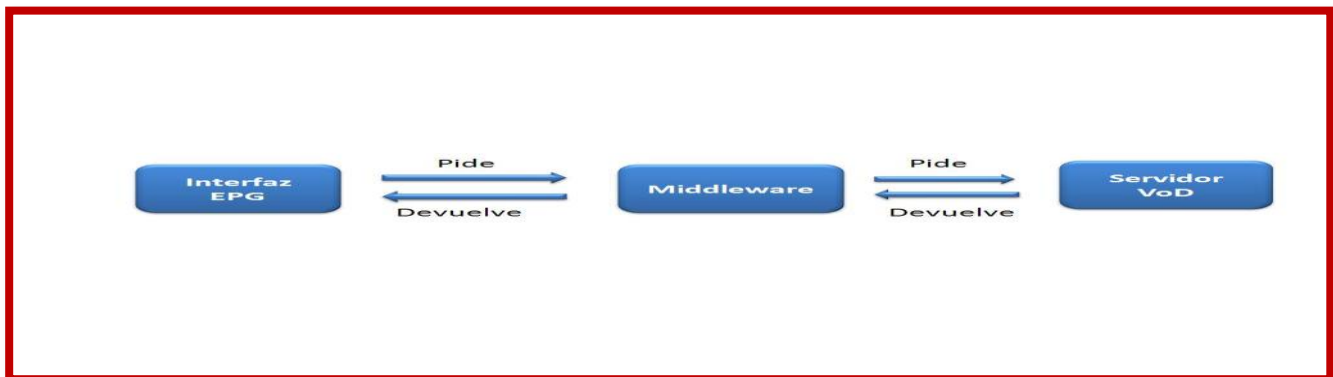


Figura 2: Visualizar contenidos

1.6.2 Listar contenidos

Cuando el usuario necesita listar los contenidos se hace una petición al middleware y este la envía hacia el servidor de aplicaciones y al servidor video bajo demanda, estos buscan la información pedida y la devuelven con una respuesta al middleware, con la información que va a ser la lista de los contenidos exigidos como se muestra en la figura 3.

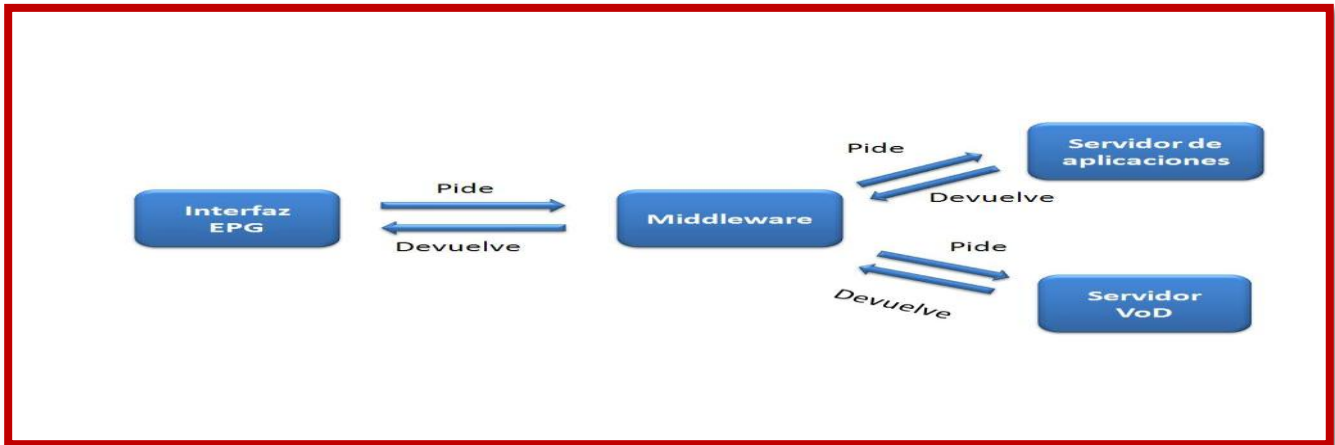


Figura 3: Listar contenidos

1.6.3 Autenticación del usuario

Cuando el usuario desea autenticarse envía una pregunta al middleware y este se la manda al sistema de control de derechos (DRM) y este devuelve la respuesta hacia el middleware que le responde mediante la interfaz de la guía electrónica de programación, si tiene permiso para autenticarse en el sistema. Esto se muestra en la figura 4. De esta forma es el proceso de funcionamiento más detallado de la EPG.

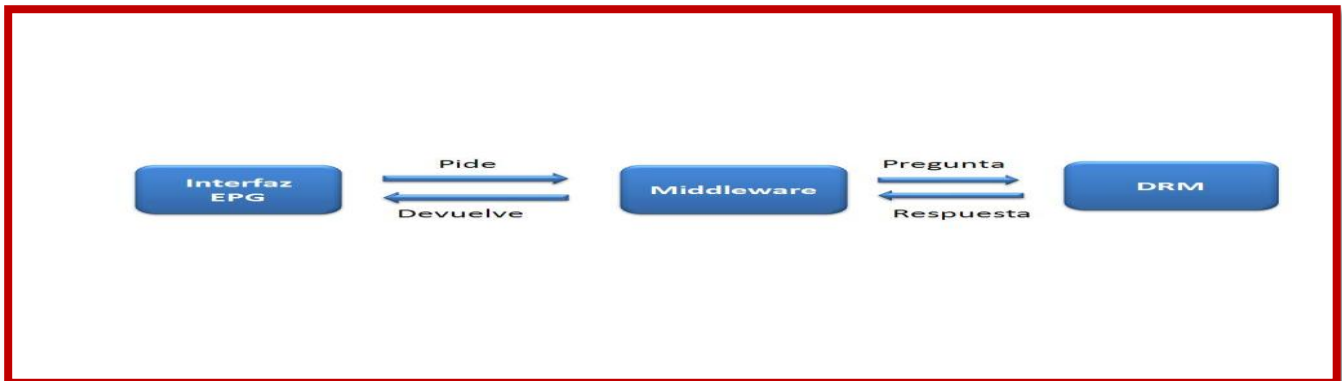


Figura 4: Autenticación del usuario

1.7 Situación Problemática

El grupo IPTV de la Universidad de las Ciencias Informáticas se le ha asignado la tarea de desarrollar una plataforma IPTV en el centro. Dado que la televisión digital ha tenido un gran auge en el mundo y por supuesto la UCI no está ajena a esto. Por las condiciones que existen y por las restricciones económicas se escogió a IPTV como forma de mejorar las transmisiones de la TV. Se selecciona principalmente por la interactividad que existe entre los usuarios y ella. Brinda con un menor ancho de banda, una gran calidad de imagen y sonido. Posibilita a cada usuario hacer una programación de canales personalizada. Se pueden reducir los costos de infraestructura y recursos, al mismo tiempo que muestra los contenidos de forma rápida y legible.

Tiene disímiles funcionalidades como el sistema de menús para la selección, búsqueda y compra de videos bajo demanda. Permite la grabación por parte del usuario de la televisión en vivo (PVR) para visualizarla en el momento que desee. El sistema es flexible a la repetición inmediata y el congelado de escenas de televisión en vivo. Dentro de este gran sistema interactúan varios subsistemas entre los que se encuentra la guía electrónica de programación (EPG). Esta facilita el cambio de canales y así mejora el zapping (cambio de canal).

Navegando por la EPG se puede acceder a detalles de los metadatos de los programas actuales, pasados o futuros como: duración, género, origen, director, actores, entre otros. Algunas características que benefician al usuario son: cambiar el idioma, cambiar los colores de la pantalla, reordenar los canales y cambiar los números de los canales. La EPG presenta varios valores agregados que le facilitan al cliente un mejor entendimiento como son las sesiones de canales favoritos, búsquedas, recordatorios y la ayuda. Por todas estas características que trae consigo la gran interactividad de los usuarios con la televisión digital y como la universidad está en un constante desarrollo tecnológico se impone la necesidad de crear una guía electrónica de programación para un sistema IPTV en la UCI. Por lo que en este trabajo se va hacer un análisis detallado del proceso para visualizar los contenidos de la EPG.

1.8 Soluciones existentes

El mundo está inmerso en un desarrollo tecnológico que se incrementa más cada día que pasa y con los países desarrollados como principales exponentes de la gran era de la televisión digital y el fin de la era analógica. Existen varias empresas en el mundo que desarrollan soluciones de IPTV, una de las principales empresas es Huawei. Esta empresa ofrece una solución completa que consta de cinco capas: red doméstica, red portadora, distribución del contenido, soporte de servicios y operación de contenido. Teniendo en cuenta la arquitectura de servicios abiertos, la solución proporciona posibilidades de integración de servicios E2E (End-to-End). Esta solución permite crear un hogar

digital mediante una serie de pasarelas domésticas y terminales multimedia. La tecnología IPTV es una parte crucial de un hogar digital.

Huawei ha diseñado uniformemente sus soluciones de IPTV, red futura y hogar digital para ofrecer a los usuarios la experiencia de una amplia gama de servicios y ayudar a los proveedores de contenido, los proveedores de servicios y los operadores de red a proporcionar una gama de aplicaciones cada vez más amplia. Huawei ha estado operando en 13 países en América Latina: Argentina, Chile, Uruguay, Colombia, Venezuela, México, Ecuador y Brasil. La compañía ha escogido Brasil como la oficina matriz para América Latina y ha estado en el país desde 1999. En un corto período de tiempo, Huawei ha sido señalado como uno de los principales jugadores del mercado. (2)

La empresa ZTE (Zhong Xing Telecommunication Equipment) es otra de las punteras en el mercado mundial. La solución ha tenido una gran penetración en los hogares en este último año, con tendencia a seguir aumentando el número de usuarios en los próximos años.

Gracias a su excelente calidad, constante innovación y habilidad para detectar las necesidades de sus usuarios, la solución de ZTE lidera el mercado de IPTV en China, proporcionando un soporte clave a los operadores de la región.

Cada vez son más los usuarios que demandan nuevos servicios de valor agregado, mayor interacción con la guía de programación (EPG) y paquetes de programación atractivos. Adicionalmente, la televisión por IP proporciona un alto nivel de personalización y control ya que el usuario puede guardar sus películas favoritas; grabarlas, retroceder un canal en vivo o pausar la grabación, votación interactiva desde la pantalla de TV entre otros beneficios.

En otras partes del mundo, los países están ofreciendo los servicios de la televisión digital por IP. En Europa el principal exponente es España que cuenta con varias empresas de telecomunicaciones que están ofreciendo estos servicios. Telefónica ofrece desde el año 2000 un servicio de televisión IP bajo el nombre de Imagenio. La compañía de telecomunicaciones Jazztel también se ha unido al grupo de esta tecnología y ofrece el servicio con el nombre Jazztelia TV.

La compañía Ya.com cuando pertenecía a T-Online, filial de Deutsche Telekom también prestó servicios de IPTV. Se basaba en la compañía Microsoft TV e incluían funcionalidades que no existían en las otras compañías como el cambio instantáneo de canal o la posibilidad de programación remota de la grabación. El decodificador poseía un disco duro interno para realizar las grabaciones.

El país pionero fue el Reino Unido y su empresa “Kingston interactive TV”. En Francia, France Telecom lanzó su primer producto de IPTV a finales de 2003. Deutsche Telecom en Alemania lanzó

su apuesta por IPTV en 2004. La empresa italiana Fastweb está ofreciendo IPTV sobre redes con conexiones 20 veces mayor que la actual y es uno de los mayores referentes en Europa de estos servicios.

Los países del tercer mundo no están ajenos a estas tecnologías, en América Latina países como Chile la empresa Telefónica Chile lanzó en el 2007 su servicio de IPTV, complementando su actual servicio de Televisión Satelital “Telefónica TV Digital”, para así ofrecer diversos servicios interactivos como video bajo demanda entre otros. Por su parte, la empresa Telefónica del Sur lanzó este servicio aproximadamente en las mismas fechas para así complementar su actual oferta de Televisión proporcionada por DirecTV.

Es válido aclarar que en este epígrafe no se habla nada concreto de todos estos sistemas y plataformas IPTV que utilizan EPG, ya que realmente están embebidas en ellos y no se tratan como temas aparte.

1.9 Conclusiones parciales

En este capítulo se abordaron temas relacionados con la televisión digital, profundizando en la Guía Electrónica de Programación donde se llegó a la conclusión que es un subsistema importante para el sistema IPTV por la interactividad que presenta para los usuarios. Se realizó un estudio detallado de definiciones de los conceptos básicos asociados al dominio del problema, contribuyendo a enriquecimiento del desarrollo de la investigación y logrando un mayor entendimiento de esta. Se efectuó un análisis de cómo es el proceso de funcionamiento de un sistema IPTV llegando al conocimiento de cómo interactúan cada uno de los subsistemas que lo integran. Se hizo un estudio de varias soluciones existentes de sistemas IPTV a nivel mundial que utilizan Guías Electrónicas de Programación pero se llegó a la conclusión de que no se pueden utilizar estas soluciones para el desarrollo del trabajo ya que son soluciones privatizadas y muy costosas, son utilizadas para empresas o comunidades de gran magnitud por lo que requieren grandes tecnologías y no se adecua al tamaño de nuestra universidad y además no brindan datos de cómo hacen las EPGs.

2

Capítulo 2: Tendencias y tecnologías actuales

2.1 Introducción

En el presente capítulo se hace un análisis de las tendencias y tecnologías que existen en la actualidad a nivel mundial que pudieran ser útiles en el desarrollo de la propuesta de solución. Se analizan las distintas metodologías de desarrollo de software y los lenguajes de modelación. Así como las herramientas CASE.

2.2 Política de migración al software libre

En los últimos años en el mundo ha ido en ascenso la tendencia de migrar del software propietario al software libre y la UCI no está ajena a esto. Debido a que los sistemas con licencia comercial están bajo el poder de empresas que monopolizan la informática a un alto nivel y se reservan los derechos de vender sus productos a clientes según su conveniencia. Se promociona cada vez más la migración hacia los software que se denominan "libres", aquellos cuyo uso por parte de cualquier cliente que esté interesado, está exento de pago.

Gracias al gran desarrollo que viene teniendo la UCI como institución de avance tecnológico se hace necesario llevar a cabo lo más rápido posible la migración al software libre. En el presente trabajo se decidió hacer uso de las herramientas y tecnologías libres partiendo de esta premisa.

2.2.1 Software libre

Debido al sello del término original en inglés (free software), se suele confundir este término con software gratuito, ya que la palabra "free" significa al mismo tiempo "gratis" y "libre". Sin embargo, el concepto de Software Libre es un asunto de libertad, y no de precio. Esta libertad es la que debería tener un usuario para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

Para los usuarios de programas existen cuatro libertades esenciales.

- La libertad de ejecutar el programa, para cualquier propósito.
- La libertad de estudiar cómo trabaja el programa, y cambiarlo para que haga lo que usted quiera. El acceso al código fuente es una condición necesaria para ello.

- La libertad de redistribuir copias para que pueda ayudar al prójimo.
- La libertad de distribuir copias de sus versiones modificadas a terceros. Si lo hace, puede dar a toda la comunidad una oportunidad de beneficiarse de sus cambios. El acceso al código fuente es una condición necesaria para ello.

Un programa es software libre si los usuarios tienen todas esas libertades. Entonces, debería ser libre de redistribuir copias, tanto con o sin modificaciones, ya sea gratis o cobrando una tarifa por distribución, a cualquiera en cualquier parte. El ser libre de hacer estas actividades significa, entre otras, que no tiene que pedir o pagar el permiso.

También debería tener la libertad de hacer modificaciones y usarlas en privado, en su propio trabajo u obra, sin siquiera mencionar que existen. Si publica sus cambios, no debería estar obligado a notificarlo a alguien en particular, o de alguna forma en particular.

La libertad de ejecutar el programa significa la libertad para cualquier tipo de persona u organización de usarlo en cualquier tipo de sistema de computación, para cualquier tipo de trabajo y propósito, sin estar obligado a comunicarlo a su programador, o alguna otra entidad específica. En esta libertad, el propósito de los usuarios es el que importa, no el propósito de los programadores. Como usuario es libre de ejecutar un programa para sus propósitos; y si lo distribuye a otra persona, también es libre para ejecutarlo para sus propósitos, pero usted no tiene derecho a imponerle sus propios propósitos.

(4)

2.2.2 Ventajas del software libre

- Económico.
- Libertad de uso y redistribución.
- Independencia tecnológica.
- Fomento de la libre competencia al basarse en servicios y no en licencias.
- Soporte y compatibilidad a largo plazo.
- Formatos estándar.
- Sistemas sin puertas traseras y más seguros.
- Corrección más rápida y eficiente de fallos.
- Métodos simples y unificados de gestión de software.
- Sistema en expansión.

2.2.3 Desventajas del Software Libre

- El soporte no es directo, salvo adquisición de licencia
- Disponibilidad de herramientas específicas reducida.
- No tiene garantía proveniente del autor.
- Relativa dificultad de aprendizaje. (4)

2.3 Tendencias actuales y tecnologías

En 1985, la Trakker Inc., unidad de la empresa Vídeo Participaciones Unidos (United Video Holdings) lanzó el primer servicio de EPG de América del Norte, conocido simplemente como el canal de guía electrónica de programas. Le permitía a los Estados Unidos y a los sistemas por cable de Canadá, proporcionar anuncios de pantalla a sus suscriptores las 24 horas del día en un canal por cable. Los listados de datos para el servicio se suministran a través de un satélite donde participan los sistemas de cable, cada uno de ellos está instalado en un ordenador donde sus instalaciones de cabecera presentan los datos a los suscriptores en un formato personalizado a la línea de canal único del sistema en marcha. El canal de EPG más tarde pasaría a denominarse Guía Previa y va a servir como el servicio de EPG de facto para sistemas de cable de América del Norte durante el resto de la década de 1980, en la década de 1990, y como red de guías de televisión para la primera década del siglo XXI. (5)

Guías de programación interactiva (GPI) son casi omnipresentes en la mayoría de los medios de difusión en la actualidad. Para la televisión, GPI se construyen en casi todos los receptores modernos para el cable digital, satélite digital, y más de la radiodifusión digital por aire. También comúnmente aparecen en los grabadores de vídeo digital como TiVo y MythTV. Los receptores finales superiores de la emisora de radio digital y radio digital por satélite, frecuencia característica incorporada en bienes públicos internacionales.

La demanda de las guías electrónicas de programas (EPG) en los canales de televisión interactiva muestra los listados de la actualidad al aire y la programación en prensa ha sido casi obviada por la amplia disponibilidad de guías interactivas de programas para la televisión. Televisión GPI basada en proporcionar la misma información que las EPG, pero de una manera más rápida y, a menudo con mucho más detalle. Cuando GPI televisión está conectada al PVR o grabadores de vídeo personales, que además, permiten a un espectador planificar su visión mediante la programación de emisiones que se registran en el disco duro para verlos más tarde.

El elemento más predominante de un GPI es su interfaz gráfica de usuario. Normalmente, adopta la forma de una rejilla o listado de nombres en las tablas de canales, los títulos de programas, y las horas de inicio. La televisión-GPI son interfaces que permiten al usuario marcar cualquier lista dada

y solicitar información adicional sobre el mismo, tales como sinopsis del programa, año de producción, actores, directores, géneros, y los metadatos descriptivos. La GPI para la radiodifusión alternativa ofrece información adicional sobre el artista y nombres de álbumes, títulos de las pistas, y los títulos de la serie. Programas de la oferta de subcanales también pueden ser listados.

Las GPI típicas también permiten a los usuarios la opción de buscar por género, así como un inmediato acceso de un toque, o la grabación de un programa seleccionado. Los recordatorios y las funciones de control parental también se incluyen a menudo. La GPI en algunos IRD DirecTV puede controlar reproductores de video usando un emisor de infrarrojos conectado que emula su control remoto.

La última revolución en la GPI es la personalización, donde la semántica se utiliza para permitir sugerencias basadas en los intereses de uno o varios visores en lo que quiere ver. Una GPI de este tipo es totalmente personalizable y permite a los usuarios utilizar o crear aplicaciones personalizadas similares a la imagen de escritorio de un ordenador personal, aunque es consciente de los tipos de programas que desea ver. También registra estos programas para que los usuarios puedan verlos más adelante en los momentos de su elección.

Las normas para la entrega de información de programación de la televisión-GPI base, varían de una aplicación a otra, y por país. La GPI de televisión más antigua como Guide Plus + se basó en la tecnología analógica (como el intervalo de supresión vertical de señales de vídeo analógico de televisión) para distribuir anuncios de datos a IPG-permitió recibir los equipos de consumo. En Europa, el European Telecommunications Standards Institute (ETSI) publicó la norma ETS 300 707 para facilitar la entrega de las GPI de datos a través de las señales de radiodifusión terrestre analógica de televisión. Listados de datos para las GPI integrado en la televisión terrestre digital de hoy y receptores de radio normalmente se envía dentro de cada estación de flujo de transporte MPEG, o junto a ella en una secuencia de datos especial. La norma ATSC para la televisión digital terrestre, por ejemplo, utiliza cuadros enviados PSIP de cada estación. Estas tablas son para contener los tiempos de inicio del programa y los títulos junto con el programa adicional de metadatos descriptivos. Señales de la hora actual también se incluyen para fines de visualización en pantalla, y también son utilizados por los dispositivos de grabación para que puedan empezar así mismos y además en el tiempo que se necesita.

Los dispositivos integrados en el cable digital moderno y receptores de televisión vía satélite, por otro lado, depende de terceros agregadores de anuncios. Parte de metadatos que les proporcione sus datos en unas listas en la pantalla. Estas empresas incluyen Tribuna TV de datos, Gemstar-TV Guide, y FYI Television, Inc. en los Estados Unidos y Europa, TV Media en los Estados Unidos y Canadá, DataServices Radiodifusión Dayscript en Europa y en América Latina.

Una tendencia creciente es para los fabricantes como Elgato y Topfield y desarrolladores de software, como Windows Media Center para utilizar una conexión a Internet para adquirir datos de su construcción en bienes públicos internacionales. Esto permite una mayor interactividad con la GPI, como descargas de los medios de comunicación, grabación de series y la programación de las grabaciones para la GPI de forma remota. IceTV en Australia, es un buen ejemplo de esto, lo que permite Tivo como los servicios de DVR en competencia y manufacturas PVR y compañías de software.

En el desarrollo de software de la GPI, los fabricantes deben incluir funciones para hacer frente a los crecientes volúmenes de datos cada vez más complejos asociados con la programación. Estos datos incluyen descripciones de los programas, horarios, clasificaciones, información de configuración de usuarios tales como listas de canales favoritos, y los contenidos multimedia. Para satisfacer esta necesidad, algunos diseños de set-top-box software de incorporar una "capa de base de datos" que utiliza cualquiera de las funciones de propiedad o una actividad comercial, fuera de la plataforma de base de datos integrados para clasificar, almacenar y recuperar datos de programación. (6)

Londres, 7 de agosto de 2009: WRN anunció hoy que se ha asociado con especialistas en tecnología en todos los medios (AIM) para proporcionar el estado de la técnica de EPG soluciones de software para los actuales y nuevos clientes.

WRN es una compañía internacional de televisión y radio, para facilitar la distribución mundial de contenido y proporcionar soluciones para los clientes de radiodifusión. WRN se ha asociado con AIM para autorizar el uso de su software en la gestión y la presentación en la guía electrónica de programación (EPG) de los horarios. El software de la Guía se ha integrado en ofrecer WRN EPG para permitir que un cliente utilice el método más fácil y eficiente tanto en la manipulación y presentación de los datos en la EPG.

AIM es una herramienta para recibir actualizaciones de EPG de los organismos de radiodifusión, validación, almacenamiento y gestión de datos de la EPG. La asociación con AIM permite a los clientes WRN para tomar ventaja de esta tecnología que puede ser configurado para ingerir listados horario de forma automática desde el sistema de gestión de un organismo de radiodifusión y publicar en una variedad de formatos diferentes, por ejemplo, Sky, Freesat, DAB, Virgin Media, así como sitio web de la emisora.

WRN está ofreciendo el software guía a los clientes nuevos y existentes.

Sophie Wilson, Jefe de Ventas y Desarrollo de Negocios de WRN, dice: "WRN está muy satisfecho de haber trabajado con AIM en la integración de la Guía de software. WRN se enorgullece de ser en

la punta de la tecnología y esta alianza significa que tendremos no sólo ser capaces de racionalizar el proceso de gestión EPG para los clientes existentes, sino que también será capaz de ofrecer un estado del arte de la solución para los clientes de radiodifusión que deseen tomar ventaja de la tecnología de guía dentro de WRN existentes, la presentación de EPG completa y que ofrece la gestión. "

Chris Gould, Director de All In Media, dijo: Estamos encantados de asociarnos con WRN para tener nuestro software Guía para una base más amplia de clientes. Sabemos de los clientes existentes que nuestra tecnología es la solución perfecta para una eficiente y al mismo tiempo la publicación de datos EPG a través de múltiples plataformas. Como EPG cada vez más importante para los organismos de radiodifusión, la solución EPG va a ser una herramienta indispensable.

Acerca de WRN

WRN es un líder en la televisión internacional y la compañía de transmisión de radio ofrece una amplia cartera de servicios de radiodifusión, incluidos los enlaces ascendentes y la capacidad de servicios en todo el mundo prácticamente cualquier satélite, el suministro internacional de satélites de acontecimientos específicos y sindicación de contenidos, AM internacionales , FM y onda corta de corretaje, la EPG de gestión y servicios de presentación, audio y video streaming, ingerir, reproducción, codificación y transcodificación y de consultoría.

Con sede en Londres, los clientes WRN incluyen los organismos de radiodifusión pública nacional e internacional, las estaciones comerciales y comunitarias, operadores de satélite y ramo de las compañías de cable. Sobre todo en medios de comunicación.

En todos los medios de comunicación (AIM) es una empresa tecnológica que proporciona software y conocimiento para aplicaciones de datos de potencia para las emisoras de radio tradicionales, así como la nueva generación de servicios en línea de audio. Los productos de AIM se utilizan en las plataformas digitales, ya sea en línea o móvil y sus aplicaciones, mejora el número de la audiencia, ofreciendo oportunidades comerciales, a los organismos de radiodifusión y medios de comunicación. Más del 50% de todas las emisoras de radio digitales en el mercado de la radio australiana hacen uso de la tecnología de AIM y está creciendo rápidamente en nuevos mercados. (7)

2.4 Metodología de desarrollo

Se conoce por metodología de desarrollo a una lista de documentos oficiales referente a las políticas, procedimientos y procesos que intervienen en el desarrollo de un software. Una metodología no es más que un conjunto de métodos y reglas que se emplean en el desarrollo de sistemas automatizados. Ofrece la posibilidad de rectificar errores detectados en cualquier fase del ciclo de desarrollo. Posibilita la relación entre los distintos trabajadores, clientes y usuarios para tomar las decisiones correctas para tener más seguridad y calidad en el trabajo. Las metodologías,

además, detallan cuales son las tareas a realizar y quiénes son los responsables de las mismas. Por lo que permite controlar el trabajo de los integrantes del grupo de desarrollo.

Las metodologías se dividen en dos categorías:

- Metodologías tradicionales o pesadas: se centran principalmente en el reconocimiento del proceso, detallando rigurosamente las actividades que están implicadas, los artefactos que se deben producir, los trabajadores que intervienen, las herramientas y notaciones que se serán utilizadas.
- Metodologías ágiles o livianas: se centran generalmente en que el software funcione, con una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. (8)

2.4.1 Extreme Programming, XP

Es la metodología ágil más radical, se enfoca en el proceso de desarrollo de software y dirige las fases de análisis, desarrollo y pruebas aspirando en lograr la calidad en el producto final. XP, nacida en el verano de 1996 de la mano de Kent Beck, tiene como principal objetivo fortalecer las relaciones interpersonales como clave fundamental para el éxito del desarrollo de software, promoviendo el trabajo en equipo, procurándose por el aprendizaje de los desarrolladores, y proporcionado un buen clima de trabajo. La ventaja básica de XP es que todo el proceso es visible y responsable. Los desarrolladores harán compromisos concretos acerca de lo que se logrará, mostrar avances concretos en la forma de despliegue de software, y cuando se alcanza un hito que se describen exactamente lo que hicieron y cómo y por qué difería de que el plan. Esto permite a las empresas orientadas a las personas a hacer su propio negocio con la confianza de los compromisos, para aprovechar las oportunidades que vayan surgiendo, y eliminar rápidamente callejones sin salida. Sin embargo, XP se ha observado durante varios posibles inconvenientes, en comparación con más metodologías basadas en el documento, incluidos los problemas con requisitos inestables, no documentada compromisos de los conflictos con los usuarios, y la falta de un diseño global o un documento de especificaciones. (8)

Características fundamentales

- Desarrollo iterativo e incremental.
- Pruebas unitarias continuas, frecuentemente repetidas y automatizadas.
- Programación en parejas.
- Frecuente interacción del equipo de programación con el cliente o usuario.

- Corrección de todos los errores antes de añadir una nueva funcionalidad.
- Propiedad del código compartido.
- Simplicidad en el código.

A continuación se muestra una tabla de diferencias entre las metodologías ágiles y las metodologías tradicionales.

<u>Metodologías Tradicionales o Robustas</u>	<u>Metodologías ágiles</u>
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurística provenientes de prácticas de producción de código
Cierta resistencia a los cambios	Especialmente preparados para cambios durante el proyecto
Impuestas externamente	Impuestas internamente (por el equipo)
Proceso mucho más controlado, con numerosas políticas/normas	Proceso menos controlado, con pocos principios
Existe un contrato prefijado	No existe contrato tradicional o al menos es bastante flexible
El cliente Interactúa con el equipo de desarrollo	El cliente es parte del equipo de desarrollo
Grupos grandes y posiblemente distribuidos	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio
Más artefactos	Pocos artefactos
Más roles	Pocos roles
La arquitectura del software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura de software

Tabla 1: Diferencias entre metodologías ágiles y tradicionales

XP se divide en varias partes como son:

Historias de usuario

Es la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas. (9)

Roles

Los roles de acuerdo con la propuesta original de Beck son:

- **Programador:** el programador escribe las pruebas unitarias y produce el código del sistema.
- **Cliente:** escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
- **Encargado de pruebas (Tester):** Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- **Encargado de seguimiento (Tracker):** Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- **Entrenador (Coach):** Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- **Consultor:** Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- **Gestor (Big boss):** Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación. (9)

Proceso

El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración. (9)

Prácticas

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas:

- El juego de la planificación. Hay una comunicación frecuente el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.
- Entregas pequeñas: producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.
- Metáfora: el sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema (conjunto de nombres que actúen como vocabulario para hablar sobre el dominio del problema, ayudando a la nomenclatura de clases y métodos del sistema).
- Diseño simple: se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- Pruebas: la producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.
- Refactorización (Refactoring): es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.
- Programación en parejas: toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores).
- Propiedad colectiva del código: cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- Integración continua: cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.

- 40 horas por semana: se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.
- Cliente in-situ: el cliente tiene que estar presente y disponible todo el tiempo para el equipo. Éste es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.
- Estándares de programación. XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible. (9)

2.4.2 Rational Unified Process (RUP)

El Proceso Unificado de Rational (habitualmente resumido como RUP) 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. RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Captura varias de las mejores prácticas en el desarrollo moderno de software en una forma que es aplicable para un amplio rango de proyectos y organizaciones. Es una guía de cómo utilizar de manera efectiva UML. Provee a cada miembro de un equipo un fácil acceso a una base de conocimiento con guías, plantillas y herramientas para todas las actividades críticas de desarrollo. Crea y mantiene modelos, en lugar de enfocarse en la producción de una gran cantidad de papeles de documentación. (3)

La metodología RUP es más apropiada principalmente para proyectos grandes lo que no quiere decir que no se pueda utilizar en proyectos pequeños, dado que requiere un equipo de trabajo capaz de administrar y desarrollar un proceso complejo en varias fases. En proyectos pequeños, es posible que no se puedan cubrir con las ganancias, los costos de dedicación de los integrantes del grupo de trabajo.

RUP presenta varios flujos de trabajos en su ciclo de vida como son:

- Modelo del Negocio: comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema y los problemas actuales de la organización e identificar las mejoras potenciales.
- Requerimientos: se recogen los requisitos funcionales y no funcionales de acuerdo con las necesidades y peticiones del cliente.
- Análisis y Diseño: se analiza la arquitectura y se diseña un modelo de acuerdo con esta.

- Implementación: es la fase de implementación del software ajustándose a la arquitectura diseñada.
- Pruebas: se realizan las pruebas para que el comportamiento del software requerido sea correcto y que todo lo solicitado esté en orden.
- Configuración y administración del cambio: guardando todas las versiones del proyecto.
- Administrando el proyecto: administrando horarios y recursos.
- Ambiente: administrando el ambiente de desarrollo.
- Distribución: hacer todo lo necesario para la salida del proyecto

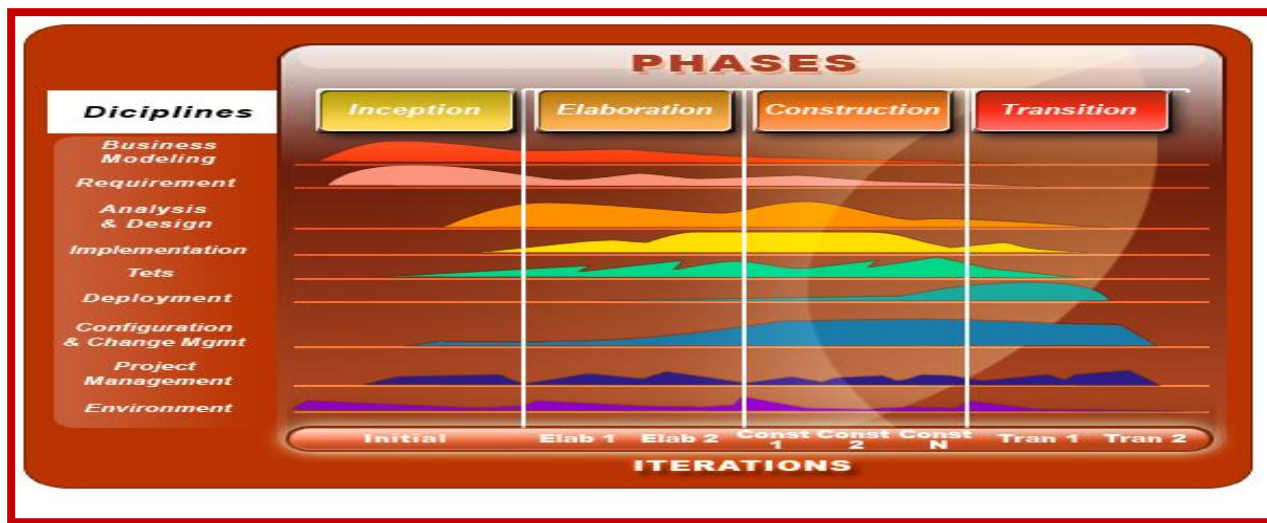


Figura 5: Fases y flujos de trabajos de RUP

RUP presenta tres características fundamentales en su desarrollo como son:

- Dirigido por los Casos de Uso: Con esto se refiere a la utilización de los Casos de Uso para el desenvolvimiento y desarrollo de las disciplinas con los artefactos, roles y actividades necesarias. Los Casos de Uso son la base para la implementación de las fases y disciplinas del RUP. Un Caso de Uso es una secuencia de pasos a seguir para la realización de un fin o propósito, y se relaciona directamente con los requerimientos, ya que un Caso de Uso es la secuencia de pasos que conlleva la realización e implementación de un Requerimiento planteado por el Cliente.
- Iterativo e Incremental: Es el modelo utilizado por RUP para el desarrollo de un proyecto de software. Este modelo plantea la implementación del proyecto a realizar en iteraciones, con lo cual se pueden definir objetivos por cumplir en cada iteración y así poder ir completando todo el proyecto iteración por iteración, con lo cual se tienen varias ventajas, entre ellas se puede mencionar la de

tener pequeños avances del proyecto que son entregables al cliente el cual puede probar mientras se está desarrollando otra iteración del proyecto, con lo cual el proyecto va creciendo hasta completarlo en su totalidad. Este proceso se explica más adelante a detalle.

- **Centrado en la Arquitectura:** Define la Arquitectura de un sistema, y una arquitectura ejecutable construida como un prototipo evolutivo. Arquitectura de un sistema es la organización o estructura de sus partes más relevantes. Una arquitectura ejecutable es una implementación parcial del sistema, construida para demostrar algunas funciones y propiedades. RUP establece refinamientos sucesivos de una arquitectura ejecutable, construida como un prototipo evolutivo. (3)

Métricas

Métrica es una metodología de planificación, desarrollo y mantenimiento de sistemas de información, promovida por el Ministerio de Administraciones Públicas del Gobierno de España para la sistematización de actividades del ciclo de vida de los proyectos software en el ámbito de las administraciones públicas. Esta metodología está basada en el modelo de procesos del ciclo de vida de desarrollo ISO/IEC 612207 así como en la norma ISO/IEC 15504 SPICE (Mejora de Procesos de Software y Determinación de Capacidad de estándares).

Elementos fundamentales:

- Procesos
- Interfaces
- Técnicas y Prácticas
- Roles o Perfiles

Procesos principales de métricas:

Al igual que ISO/IEC 12207, Métrica está orientada al proceso y, en su versión 3, estos procesos son:

- Planificación de Sistemas de Información (PSI).
- Desarrollo de Sistemas de Información (DSI). Debido a su complejidad, está a su vez dividido en cinco procesos:
 - ✓ Estudio de Viabilidad del Sistema (EVS).
 - ✓ Análisis del Sistema de Información (ASI).
 - ✓ Diseño del Sistema de Información (DSI).
 - ✓ Construcción del Sistema de Información (CSI).
 - ✓ Implantación y Aceptación del Sistema (IAS).

- ✓ Mantenimiento de Sistemas de Información (MSI).

Interfaces de métricas:

Métrica, en su versión 3, proporciona también cuatro interfaces que definen actividades orientadas a la mejora y perfeccionamiento de los procesos principales para garantizar la consecución del objetivo del desarrollo.

- Gestión de proyectos (GP).
- Seguridad (SEG).
- Aseguramiento de la Calidad (CAL).
- Gestión de la Configuración (GC).

Técnicas de métricas:

Métrica, en su versión 3, distingue entre:

- Técnicas de desarrollo (Casos de Uso, Diagramas de Clases, Diagrama de flujo de datos).
- Técnicas de gestión de proyectos (Técnicas de estimación, Planificación)
- Prácticas (Análisis de impacto, Presentaciones, Prototipado)

Perfiles de métricas:

Métrica establece los siguientes perfiles para los participantes en el proceso de desarrollo de un sistema de información:

- Directivo (Comité de Dirección, Directores de Usuarios).
- Jefe de Proyecto (Responsable de Implantación, Responsable de Seguridad).
- Consultor (Consultor Informático, Técnico de Sistemas).
- Analista (Analista, Administrador de bases de datos).
- Programador.

2.5 Selección de la metodología a utilizar

Todo desarrollo de software es riesgoso y difícil de controlar, pero si no llevamos una metodología de por medio, lo que obtenemos es clientes insatisfechos con el resultado y desarrolladores aún más insatisfechos. Sin embargo, muchas veces no se toma en cuenta el utilizar una metodología adecuada, sobre todo cuando se trata de proyectos pequeños de dos o tres meses. Cuando los proyectos que se van a desarrollar son de mayor envergadura, ahí sí toma sentido el basarse en una metodología de desarrollo, y se empezó a buscar cual sería la más apropiada. Luego de analizar las

metodologías antes mencionadas se puede apreciar que todas tienen características que las hacen específicas, pero siempre orientadas al control del proceso de desarrollo de software y se puede llegar a la conclusión que para el desarrollo del proyecto la metodología RUP es la más adaptable por ser un proceso de desarrollo de software, constituir la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos, dirigida por casos de uso, técnica muy desarrollada en la actualidad que permite capturar y analizar la funcionalidad de un sistema desde el punto de vista de un usuario y que ampliado a través de diferentes modelos, permite su implementación y posterior actualización.

2.6 El analista de Sistema

El análisis de sistemas es un paso fundamental antes de empezar un proyecto de software; se hace con el propósito de obtener una visión más clara sobre lo que el sistema debe hacer, determinando tanto las necesidades del cliente y los límites del sistema, como su estructura y funcionamiento. Se entiende por análisis al proceso de ingeniería que busca comprender el problema que tendrá que resolver el sistema, definir el alcance del sistema, asegurar que el sistema satisfaga las necesidades del usuario, definir los criterios de aceptación y proporcionar una base para el desarrollo de un sistema.

El analista de sistemas surge de la necesidad de analizar, identificar y separar en procesos toda la información referente al software que se desee construir o mejorar, es el encargado de proponer soluciones y seleccionar la idea más idónea para el problema en cuestión. (8)

Los analistas poseen un amplio rango de habilidades. La primera y principal es que el analista soluciona problemas, le motiva el reto de analizar un problema y encontrar una respuesta funcional que satisfaga al cliente. Los analistas de sistemas requieren habilidades de comunicación que les permitan relacionarse en forma significativa con muchos tipos de personas diariamente, así como habilidades de computación. Para el éxito del analista es necesario que se involucre el usuario final.

En cualquier proceso de desarrollo de software es indispensable el analista de sistemas. Este se encarga no sólo de investigar lo referente al sistema que se quiere desarrollar; es más que eso; su labor es lograr que tanto el cliente como los desarrolladores hablen el mismo idioma en cuanto a lo que se quiere desarrollar. Selecciona la metodología y define la estrategia de captura de requisitos con el propósito de lograr los objetivos que se proponen.

2.6.1 El analista de Sistema en RUP

RUP considera al analista de sistema imprescindible para la calidad del software. Las responsabilidades del analista de sistemas de acuerdo con esta metodología son:

- Realizar la modelación del negocio o del dominio, en este caso se realizará un modelo de dominio debido a la falta de un proceso de negocio bien definido.
- Confeccionar el documento de requisitos funcionales y no funcionales del sistema, para el desarrollo del presente trabajo se realizarán varias reuniones con el cliente para saber sus necesidades y detectar los requisitos funcionales y no funcionales del sistema.
- Delimitar el sistema mediante los actores y casos de uso, teniendo en cuenta que el modelo de caso de uso sea completo y consistente, ya que RUP está guiado por casos de uso, en el presente trabajo se definirán los casos de uso de acuerdo con las necesidades planteadas por el cliente.
- Describir cada caso de uso teniendo en cuenta las acciones del actor y del sistema, en este caso se realizarán las descripciones de todos los casos de uso teniendo en cuenta los flujos alternos correspondientes a cada uno.
- Modelar las Clases del Análisis y las Clases del Diseño.

La metodología RUP plantea que para que exista una representación efectiva del software debe existir un buen análisis de requisitos, por lo que el analista de sistemas es un factor fundamental en el desarrollo del software.

2.7 ¿Qué es UML?

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es un lenguaje de modelado de sistemas de software. Es uno de los más utilizados y conocidos en la actualidad. Es un lenguaje gráfico para visualizar, documentar, construir y especificar un sistema o una aplicación de software. Ofrece un estándar para el modelado de los sistemas por lo que logra una descripción a un alto grado nivel formal y hace que se haga más entendible para los clientes o usuarios finales. Incluye aspectos conceptuales tales como procesos de negocio y funciones del sistema y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. UML está diseñado para ser utilizado en software orientados a objetos y es limitado para otros tipos de programaciones.

Sus funciones se pueden resumir en:

- Visualizar: UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.

- Especificar: UML permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión.

Características de UML

- Permite modelar sistemas utilizando técnicas orientadas a objetos (OO).
- Permite especificar todas las decisiones de análisis, diseño e implementación, construyéndose así modelos precisos, no ambiguos y completos.
- Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).
- Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, etc.).
- Cubre las cuestiones relacionadas con el tamaño, propias de los sistemas complejos y críticos.
- Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.
- UML es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental. (10)

Diagramas de UML

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

En los que se encuentran: los diagramas de estructura que enfatizan en los elementos que deben existir en el sistema modelado.

- Diagrama de Clases.
- Diagrama de Objetos.
- Diagrama de Componentes.
- Diagrama de Despliegue.

Los diagramas de comportamiento enfatizan en lo que debe suceder en el sistema modelado:

- Diagrama de Estados.
- Diagrama de Actividades.

- Diagrama de Casos de Uso.

Los Diagramas de Interacción son un subtipo de diagramas de comportamiento, que enfatizan sobre el flujo de control y de datos entre los elementos del sistema modelado:

- Diagrama de Secuencia.
- Diagrama de Colaboración. (3)

2.8 Herramientas CASE

Las herramientas case son distintas aplicaciones informáticas con el objetivo de aumentar la productividad y de reducir los tiempos de trabajo y costos. Estas herramientas ayudan en todos los aspectos del ciclo de vida de desarrollo del software. En tareas como el proceso de realizar un diseño del proyecto, para calcular costos, en la implementación de parte del código con el diseño dado, compilación automática, documentación o detección de errores.

El concepto de CASE es muy amplio; y una buena definición genérica, que pueda abarcar esa amplitud de conceptos, sería la de considerar a la Ingeniería de Software Asistida por Computación (CASE), como la aplicación de métodos y técnicas a través de las cuales se hacen útiles a las personas, comprender las capacidades de las computadoras, por medio de programas, de procedimientos y su respectiva documentación. (11)

2.8.1 Rational Rose Enterprise

Rational Rose es la herramienta CASE que comercializan y utilizan los desarrolladores de UML y que soporta de forma completa la especificación del UML. Esta herramienta propone la utilización de cuatro tipos de modelo para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software.

Es el producto más completo de la familia Rational Rose. Siendo la mejor elección para el ambiente de modelado que soporte la generación de código a partir de modelos en Ada, ANSI C++, C++, CORBA, Java/J2EE, Visual C++ y Visual Basic. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente.

Rational Rose Enterprise presenta varias características fundamentales:

- Capacidad de Análisis de calidad de código.

- Característica de control por separado de componentes modelo que permite una administración más granular y el uso de modelos.
- Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
- Integración con otras herramientas de desarrollo de Rational.
- Capacidad para integrarse con cualquier sistema de control de versiones SCC-compliant, incluyendo a Rational ClearCase. (3)

Rational Rose proporciona mecanismos para realizar la denominada Ingeniería Inversa, es decir, a partir del código de un programa, se puede obtener información sobre su diseño. Los costos de las licencias son muy altos. Cubre todo el ciclo de vida del desarrollo de un proyecto con:

- Concepción y formalización de un proyecto.
- Construcción de los componentes.
- Transición a los usuarios y certificación de las distintas fases.

2.8.2 Visual Paradigm

Visual Paradigm es una herramienta CASE que utiliza como lenguaje modelado a UML. Está creada para una gran cantidad de usuarios que deseen construir sistemas con un alto grado de fiabilidad, usando siempre el paradigma orientado a objetos. Incluye actividades como análisis de negocio, análisis de sistemas e ingeniería de software. Ayuda al equipo de desarrollo a mejorar todo el modelo de construcción y despliegue del software. La licencia es comercial y gratuita.

Emplea las últimas notaciones de UML, utiliza la generación del código, importa del rational rose, exportación/importación XML, ingeniería inversa, generador de impresos, integración del ms Visio, IDE Integration de UML con Eclipse, NetBeans.

Soporta un conjunto de lenguajes, tanto en la generación de código y como de ingeniería inversa sobre Java, C++, CORBA IDL, PHP, XML Schema, Ada y Python. Además, apoya la generación de código C#, VB. NET, código script de Flash, Delphi, Perl, Ruby, entre otros. Ingeniería Inversa también apoya clase Java, .NET, .dll y .exe. (10)

Visual Paradigm tiene diversas características como son:

- Producto de calidad.
- Soporta aplicaciones web.

- Las imágenes y reportes generados, no son de muy buena calidad.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones. (10)

Visual Paradigm ofrece servicios tales como:

- Entorno de creación de diagramas para UML 2.0
- Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa (versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDEs.
- Disponibilidad en múltiples plataformas. (10)

2.9 Selección de la herramienta CASE a utilizar

Después de hacer un análisis en el estudio de estas dos herramientas CASE, se determinó que la herramienta que se utilizará en el trabajo para la parte de la ingeniería de software es el Visual Paradigm. Tanto el Rational Rose como el Visual Paradigm son herramientas con mucha potencia, esto se pudo demostrar en los epígrafes anteriores, pero el Visual Paradigm tiene ventaja en cuanto a que es multiplataforma, se puede utilizar lo mismo en sistemas operativos propietarios como Windows o en sistemas operativos libres como Linux. Además, facilita documentar con exactitud todo el trabajo a realizar. Permite dibujar todos los diagramas de clases, el código que se genera desde los diagramas. Realiza un diseño que se centra en los casos de usos enfocados en el negocio. Todo esto favorece a la realización de un magnífico desarrollo del software lo que trae consigo un producto con una alta calidad y resultados exitosos.

2.10 Conclusiones parciales

En el presente capítulo se realizó una investigación relacionada con las tendencias y tecnologías de la actualidad para el desarrollo del trabajo. Donde se abordaron temas sobre las políticas de migración al software libre, definiciones, ventajas y desventajas para llegar hasta el punto de que el desarrollo del trabajo se hará sobre una plataforma libre. Se describieron características de las metodologías de desarrollo, llegando a la conclusión que se utilizará la metodología RUP por las facilidades de detalles que brinda, lo que contribuye con una excelente organización del trabajo. Se hace una caracterización del rol de analista obteniendo el resultado de las responsabilidades del analista para la metodología RUP. Se escogió a UML como lenguaje de modelación. La herramienta CASE que se seleccionó fue Visual Paradigm por su característica principal, de ser multiplataforma. Teniendo en cuenta el estudio realizado se podría afirmar que la selección realizada es la óptima para efectuar un trabajo con la calidad que requiere.

3

Capítulo 3: Análisis de la solución propuesta

3.1 Introducción

En este capítulo se realizará el análisis de la solución propuesta con el objetivo de entender a fondo lo que se quiere desarrollar. Para lograr este objetivo se realizará el modelo del dominio para comprender los conceptos más significativos del tema, así como el levantamiento de requisitos funcionales y no funcionales. También se representa el Diagrama de Casos de Uso del Sistema (DCUS) así como la descripción de los mismos.

3.2 Modelo del dominio

Cuando el flujo de información es difuso, dudoso, inexacto, con múltiples orígenes o solo contiene eventos o sucesos, cuando existe la imposibilidad de determinar subsistemas por el exceso de interconexiones, cuando hay un solapamiento de responsabilidades y además es difícil establecer reglas de funcionamiento, por citar algunos casos, estamos en presencia de un entorno organizacional el cual funciona como un dominio. (12)

El modelado de dominio permite hacer un correcto levantamiento de requisitos, su objetivo es comprender y describir las clases conceptuales más significativas dentro del dominio del problema, lo cual permite definir los procesos y roles más significativos. Esto ayuda a los usuarios, desarrolladores y clientes a tener un vocabulario común, contribuyendo con la aceleración del desarrollo de la aplicación. Debido a la falta de un proceso de negocio bien definido se decidió desarrollar un modelo de este tipo.

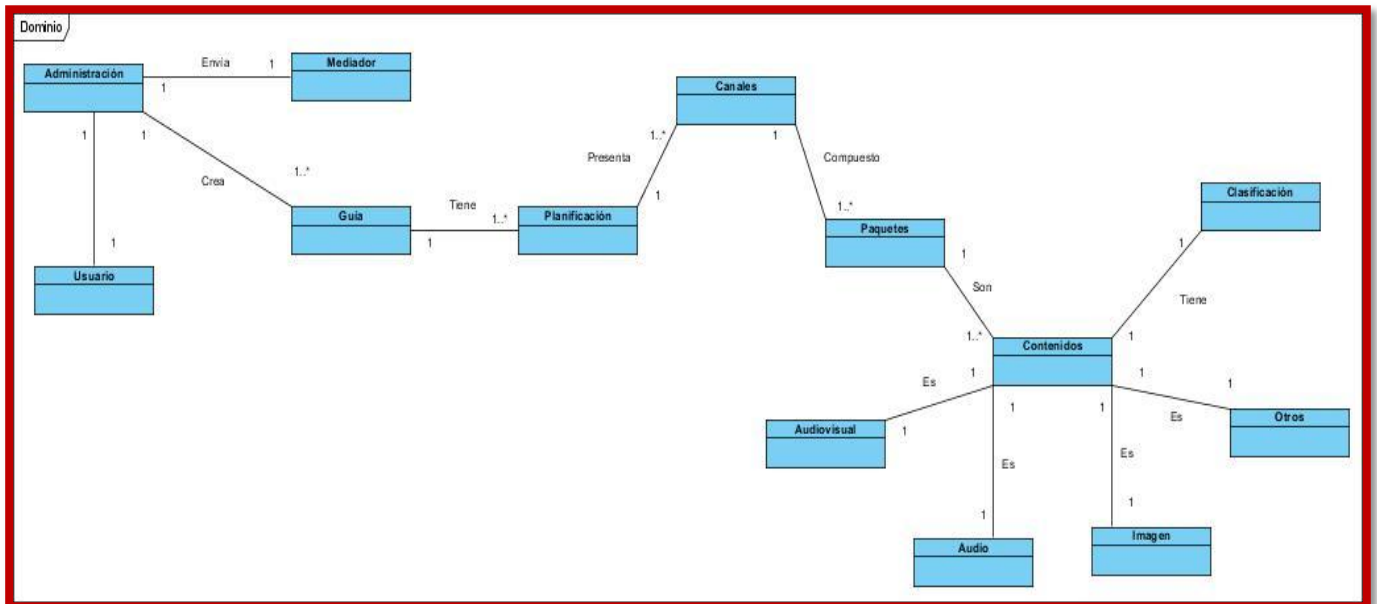


Figura 6: Diagrama de clases conceptuales del dominio

3.3 Definición de las clases conceptuales del modelo del dominio

En los siguientes subepígrafos se muestra una definición de las clases conceptuales del modelo del dominio lo que proporciona un mayor entendimiento de los procesos identificados.

3.3.1 Administración

La clase administración es la que gestiona todos los procesos.

3.3.2 Guía

Es la que contiene la información de las guías personalizadas.

3.3.3 Planificación

Es donde se hace la planificación de cada guía con los elementos siguientes: fecha de inicio, fecha de fin y días de la semana en que se van a usar.

3.3.4 Canales

Los canales son el medio de transmisión del contenido.

3.3.5 Paquetes

Los paquetes son un conjunto de información a transmitir.

3.3.6 Contenido

Los contenidos pueden ser audiovisuales, solo audio, imagen, entre otros

3.3.7 Clasificación

Es donde se clasifican los contenidos de acuerdo a su información.

3.3.8 Audiovisual

Es un contenido donde se integran e interrelacionan audio y video.

3.3.9 Audio

Es el contenido que emite ondas sonoras que son audibles para los usuarios.

3.3.10 Imagen

Es la representación solo visual de algo.

3.3.11 Otros

Los otros son contenidos que no entran en las categorías anteriores pero que son importantes.

3.3.12 Mediador

Es el encargado de gestionar, manipular el flujo de información y servir de intermediario en la comunicación entre todas las partes de la plataforma.

3.3.13 Usuario

Es una persona que inicia la interacción, la cual está previamente autenticada.

3.4 Breve descripción del diagrama de dominio

En este diagrama se evidencian conceptos importantes para el dominio del problema, se divide en los siguientes subprocesos: el subproceso de gestión de la EPG, listar los contenidos de la EPG y visualizar los contenidos de la EPG.

En el subproceso de gestión intervienen clases como la de administración, que funciona como controladora, que es donde se crean las guías, cada una de estas guías tiene una planificación. La planificación se realiza con varios canales y estos a su vez poseen varios programas. Los programas están compuestos por varias medias que tienen una clasificación y son de tipo audiovisual, audio, imagen u otros. En la categoría otros se encuentran los teletextos, pdf y archivos XML.

En el subproceso de listar los contenidos existe la clase usuario que interactúa con la controladora. Una vez autenticado el usuario, es la primera vez que se autentifica, la controladora le va a mostrar la guía general, si no le muestra la guía personalizada.

Para el subproceso de visualizar los contenidos, el usuario hace una demanda a la controladora

donde ésta a su vez le hace el pedido al middleware. El middleware le devuelve la respuesta con el contenido demandado, y este se visualiza al usuario. De esta forma, se resume el funcionamiento de este diagrama de dominio.

3.5 Requerimientos

El proceso de construcción de un software está dado por varias actividades que han de ser llevadas a cabo por una guía metodológica, garantizando la calidad del producto y la satisfacción de los clientes y desarrolladores, al mismo tiempo, está dado también por técnicas y procedimientos que se llevan a cabo con el objetivo de realizar una documentación relacionada con cada uno de los flujos de actividades que tienen lugar durante la construcción del sistema.

Una etapa inicial y muy importante dentro del proceso de la Ingeniería de Software, es la Ingeniería de Requerimientos, donde se lleva a cabo el proceso de descubrir, analizar, describir y verificar las actividades, servicios y restricciones del sistema de software que se desea hacer; este proceso se realiza mediante la obtención, el análisis, la especificación, la validación y la administración de los requerimientos del software. Los requerimientos de software son las necesidades de los clientes, los servicios que los usuarios desean que proporcione el sistema de desarrollo y las restricciones en las que debe operar.

Para la obtención de los requerimientos del subsistema de la guía electrónica de programación se hizo necesario acudir a varias técnicas relacionadas con dicho proceso como es la entrevista a varios integrantes del proyecto así como a los líderes de este. Se hizo además el previo estudio e investigación de software en los que de una forma u otra se desarrolla este subsistema de visualización de contenidos personalizados. También se utilizó la técnica llamada lluvia de ideas así como la del enfoque grupal, donde cada uno de los integrantes del equipo de desarrollo expresa su punto de vista e ideas relacionadas con las funcionalidades que requiera el producto.

3.5.1 Requerimientos funcionales

Los requerimientos funcionales son las acciones o funcionalidades que el sistema de ser capaz de realizar, aquí se incluyen las acciones que son ejecutadas por el terminal, acciones que están ocultas y que el sistema debe cumplir y funciones extremas a determinar por el mismo sistema.

RF1 Personalizar la guía electrónica de programación

El subsistema debe permitir personalizar una guía electrónica de programación por un usuario que es primera vez que entra al sistema.

RF2 Eliminar la guía electrónica de programación

El subsistema debe permitir eliminar una guía electrónica de programación creada por el usuario en el momento que este desee.

RF3 Actualizar la guía electrónica de programación

El subsistema debe permitir actualizar la guía electrónica de programación creada por el usuario en un momento dado.

RF4 Buscar en la guía electrónica de programación

El subsistema debe permitir realizar búsquedas de canales y programas según las preferencias del usuario en la guía electrónica de programación.

RF5 Listar contenidos de la guía electrónica de programación

El subsistema debe permitir que la guía electrónica de programación liste los contenidos una vez que el usuario sea autenticado.

RF6 Pre-visualizar los contenidos

El subsistema debe permitir la pre-visualización de los contenidos en una pequeña pantalla a medida que van siendo seleccionados.

3.5.2 Requerimientos no funcionales

Los requerimientos no funcionales, son propiedades o cualidades que el producto debe tener. Estas propiedades se ven como las características que hacen al producto agradable, usable, rápido o confiable. A continuación se definen los siguientes:

RNF1 Usabilidad

El sistema podrá ser usado por personas con al menos conocimientos básicos en el manejo de computadoras. El software brindará la posibilidad de tener siempre visible la opción de ayuda, lo que trae consigo un mejor aprovechamiento por parte de los usuarios de las funcionalidades que el sistema ofrece.

RNF2 Requerimientos del Software

Requerimientos mínimos para las PCs clientes:

- Un Navegador como Mozilla Firefox, Safari u otro.
- Sistema operativo: GNU/Linux, Windows.

Requerimientos mínimos para los servidores:

- Servidor Web Apache 2.0 o superior.
- PostgreSQL como Sistema Gestor de Base de Datos.

RNF3 Requerimientos de Hardware

Para las PCs clientes:

- Se requiere tengan tarjeta de red.
- Al menos 256 MB de memoria RAM porque es el valor mínimo más estable para sistemas IPTV planteado en las recomendaciones de la Unión Internacional de Telecomunicaciones (UIT).
- Procesador 1024 MHz como mínimo porque es el valor mínimo más estable para sistemas IPTV planteado en las recomendaciones de la Unión Internacional de Telecomunicaciones (UIT).

Para los servidores:

- Se requiere tarjeta de red.
- Se requiere tarjeta de video y de captura.
- Procesador 3 GHz como mínimo porque es el valor mínimo más estable para sistemas IPTV planteado en las recomendaciones de la Unión Internacional de Telecomunicaciones (UIT).

RNF4 Confiabilidad

La herramienta de implementación a utilizar debe tener soporte para recuperación de información ante fallos y errores que se puedan cometer. La información manejada por el sistema deberá estar protegida del acceso no autorizado y de divulgación a terceras personas.

RNF5 Rendimiento

La aplicación deberá disminuir el tiempo de respuesta al máximo, el cual está dado por la cantidad de información a procesar, entre mayor cantidad de información mayor será el tiempo de procesamiento, es por eso que se tratará que el tiempo de demora sea lo suficientemente corto, como plantean las recomendaciones de la UIT para sistemas de este tipo.

RNF6 Apariencia o interfaz externa

La aplicación deberá ser sencilla, intuitiva, amigable, debido a que los usuarios no son expertos en el uso de la misma, que le permita realizar acciones con fácil manejo y que posibilite también el buen desempeño en la ejecución de sus funciones.

RNF7 Portabilidad

El sistema será multiplataforma y su puesta en marcha debe ser compatible con los sistemas operativos Windows y Linux.

RNF8 Legales

El sistema debe ajustarse y regirse por la ley, decretos leyes, decretos, resoluciones y manuales establecidos, que norman los procesos que serán automatizados. La mayoría de las herramientas de desarrollo son libres y del resto, las licencias están avaladas.

Como producto, los componentes de la EPG para un sistema IPTV se distribuyen amparados bajo las normativas legales establecidas en el registro comercial emitido por las entidades jurídicas de la Universidad de las Ciencias Informáticas.

RNF9 Seguridad

Los usuarios antes de acceder a la EPG tienen que ser autenticados a través del middleware.

3.6 Descripción del Sistema. Modelo de Casos de Usos del Sistema (MCUS)

Luego de haber realizado anteriormente la descripción de los requisitos, quienes constituyen las funcionalidades esenciales del sistema, se le da paso a la siguiente etapa de construcción de un software que es la descripción del Modelo de Casos de Uso del Sistema (MCUS). El MCUS describe un sistema en cuanto a su utilización, es un modelo que contiene actores, casos de uso y las relaciones que se establecen entre ellos.

Los requerimientos del software se agrupan según sus características, utilidad y usabilidad conformando así los llamados casos de uso, los cuales no son más que fragmentos de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. Estos casos de uso describen cómo se comporta y funciona un sistema específico, mediante un documento que narra de manera secuencial las acciones relevantes que realiza un actor en interacción con el sistema hasta completar un proceso, sin darle importancia a los detalles de la implementación.

Los actores se definen como los roles que puede tener un usuario, pueden ser humanos, otros sistemas, máquinas, hardware, etc. que interactúan con un sistema para de esta forma intercambiar datos, aunque en algunos casos pueden constituir un recipiente pasivo de información. Un actor no es parte del sistema en desarrollo, es un agente externo que interactúa con el mismo en pos de obtener un resultado esperado. El subsistema de la EPG para un sistema IPTV cuenta con un actor que se especifica en la siguiente tabla:

Actor	Descripción
Usuario	Es un usuario, que puede interactuar mediante un STB o una computadora.

Tabla 2: Descripción de los actores del sistema

3.6.1 Modelo de Casos de Uso del Sistema

A partir de la identificación del actor que interactúa con la aplicación, así como la recopilación del conjunto de funcionalidades escritas en forma de requerimientos que a su vez han sido agrupados en casos de uso según sus peculiaridades, se conforma el Modelo de Casos de Uso que se representa a continuación:

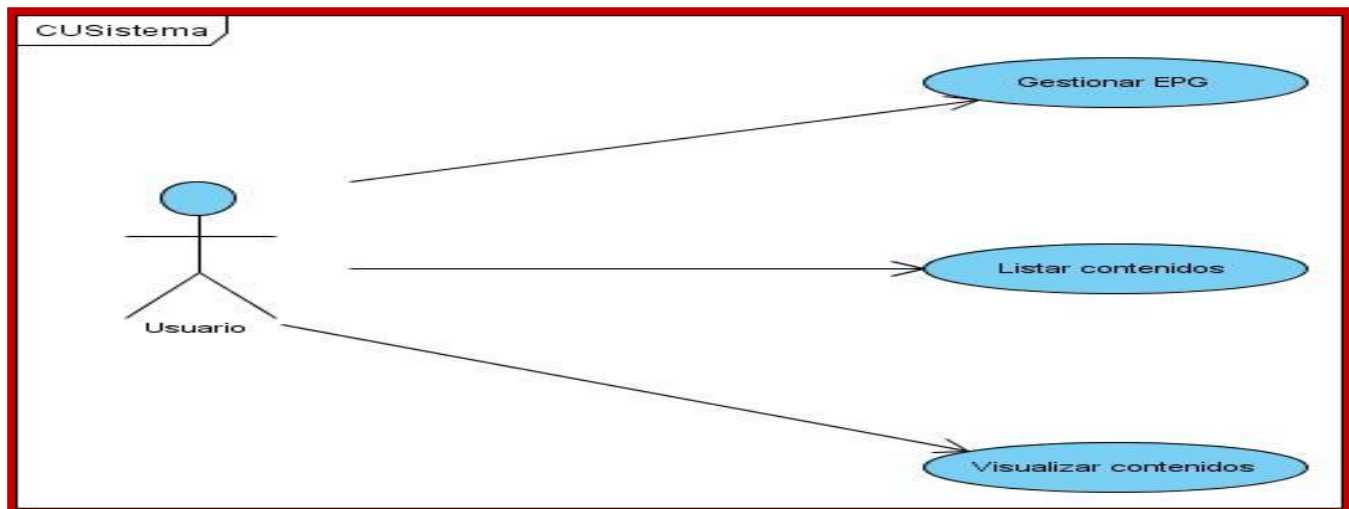


Figura 7: Diagrama de Casos de Uso del Sistema

3.7 Descripción textual de los Casos de Uso del Sistema

Cada uno de los casos de uso identificados presenta características particulares que para su mejor entendimiento se hace necesario describirlas textualmente. Además que para lograr un mejor rendimiento del tiempo en la construcción del software es recomendable identificar los casos de uso arquitectónicamente significativos, que en la descripción textual se diferencian porque tienen la prioridad de crítico.

3.7.1 Gestionar EPG

Caso de Uso:	Gestionar EPG
Actores:	(Usuario)
Resumen:	El caso de uso permite crear, eliminar, actualizar guías electrónicas de programación y hacer búsquedas en ellas.
Precondiciones:	Que el usuario se haya autenticado con anterioridad.
Referencias	RF1, RF2, RF3, RF4.
Prioridad	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. El usuario solicita la visualizar interfaz para crear, eliminar y actualizar guía o hacer búsquedas.	1.1.El sistema muestra la interfaz para crear, eliminar, actualizar o buscar guía. a) Si se desea crear guía, ir a la sección “Personalizar guía”. b) Si se desea eliminar guía, ir a la sección “Eliminar guía”. c) Si se desea actualizar guía, ir a la sección “Actualizar guía”. d) Si se desea hacer búsquedas por canales o programas, ir a la sección “Búsqueda”.

Sección “Personalizar EPG”

Acción del Actor	Respuesta del Sistema
1. El usuario selecciona personalizar una EPG.	1.1. Muestra formulario. 1.2. Brinda la posibilidad de personalizar una guía.
2. El usuario inserta los parámetros: canal, lenguaje, nombre del programa, nombre completo de la guía, sinopsis del programa, clasificación, período de duración, días de la semana, tiempo de inicio del programa y tiempo de duración del programa.	2.1. El sistema comprueba que los parámetros insertados por el usuario son válidos. 2.2. El sistema envía los parámetros a la interfaz del middleware. 2.3. La interfaz del middleware devuelve la respuesta. 2.4. El sistema guarda la guía personalizada en un XML TV. 2.5. El sistema muestra un mensaje "Guía personalizada satisfactoriamente".

Flujos Alternos

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

Prototipo de Interfaz

Sección "Eliminar EPG"

Acción del Actor	Respuesta del Sistema
2. El usuario selecciona eliminar EPG.	2.1. Verifica que el usuario tenga una guía personalizada creada. 2.2. Muestra la guía personalizada creada. 2.3. Elimina la guía personalizada. 2.4. Muestra mensaje de rectificación "Desea eliminar la guía". 2.5. Muestra un mensaje "guía eliminada".

Flujos alternos

Acción del Actor	Respuesta del Sistema
	2.1 Verifica que el usuario no tenga una guía personalizada creada. 2.2 Muestra un mensaje "no tiene guía personalizada creada".

Prototipo de Interfaz



Sección "Actualizar EPG"

Acción del Actor	Respuesta del Sistema
3. El usuario selecciona actualizar EPG.	3.1 El sistema verifica que el usuario tenga una guía personalizada

	<p>creada.</p> <p>3.2 Muestra la lista de guías personalizadas creadas.</p>
4. El usuario selecciona la guía personalizada que quiere modificar.	<p>4.1 El sistema muestra el formulario con los datos de la guía.</p> <p>4.2 Brinda la posibilidad de modificar la guía personalizada.</p>
5. El usuario modifica los datos.	<p>5.1 Verifica los parámetros modificados.</p> <p>5.2 Guarda los cambios en el XML TV.</p> <p>5.3 Envía mensaje de notificación.</p>
Flujos alternos	
Acción del Actor	Respuesta del Sistema
	<p>3.1 Verifica que el usuario no tenga una guía personalizada creada.</p> <p>3.2 Muestra un mensaje “no tiene guía personalizada creada”.</p> <p>3.3 Vuelve a ejecutar el flujo normal.</p>
	<p>4.1 El sistema detecta que los parámetros modificados no son válidos.</p> <p>4.2 El sistema muestra un mensaje “parámetros inválidos”.</p>
Prototipo de Interfaz	

Sección “Búsqueda”

Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la sección de búsqueda en la EPG.	1.1 El sistema solicita los parámetros de búsqueda.
2. El usuario le proporciona los parámetros de la búsqueda.	2.1 El sistema hace la búsqueda de acuerdo con los parámetros requeridos por el usuario. 2.2 Muestra una lista con la información requerida por el usuario.

Flujos alternos

Acción del Actor	Respuesta del Sistema
	2.1 Si no se encuentra información de los parámetros que el usuario requiere, se muestra un mensaje “no existe información”.

Prototipo de Interfaz

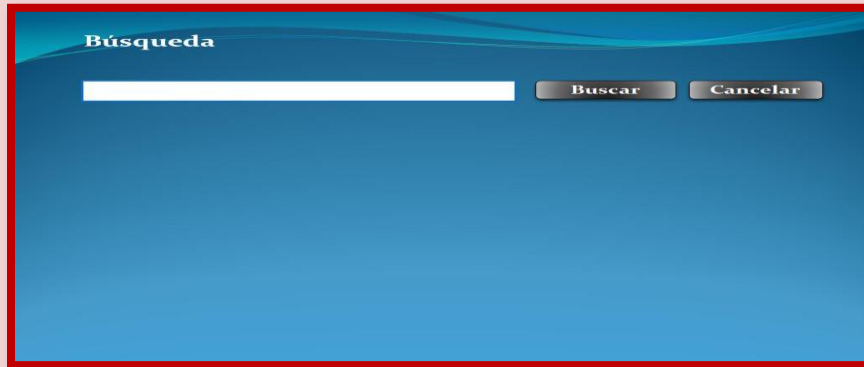


Tabla 3: Descripción del CU Gestionar EPG

3.7.2 Listar Contenidos

Caso de Uso:	Listar Contenidos
Actores:	(Usuario)
Resumen:	El caso de uso permite listar los contenidos de la guía general y de las guías personalizadas
Precondiciones:	Que el usuario se haya autenticado con anterioridad.
Referencias	RF5.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2. El usuario solicita listar los contenidos de la EPG.	2.1.El sistema muestra la interfaz para listar los contenidos. 2.2.El sistema muestra la guía general. a) Si se desea listar la guía personalizada, ir a la sección “Guía personalizada”.
Sección “Listar contenidos de la EPG general”	
Acción del Actor	Respuesta del Sistema
3. El usuario solicita listar los contenidos de la EPG general.	3.1.El sistema le pide al middleware el flujo de información a listar. 3.2. Una vez devuelto el flujo de información, crea la lista con esta

	información. 3.3.El sistema muestra la guía general solicitada.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Sección “Listar contenidos de la EPG personalizada”	
Acción del Actor	Respuesta del Sistema
3. El usuario selecciona la sección “Guía personalizada”.	5.4 El sistema muestra una lista de todas las guías personalizadas existentes.
6. El usuario selecciona su guía personalizada.	4.1 El sistema lista los contenidos de la guía seleccionada por el usuario.
Flujos alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz Interfaz 2	

Tabla 4: Descripción del CU Listar Contenidos

3.7.3 Visualizar Contenidos

Caso de Uso:	Visualizar Contenidos
Actores:	(Usuario)
Resumen:	El caso de uso permite visualizar los contenidos de la guía electrónica de programación.
Precondiciones:	Que el usuario se haya autenticado con anterioridad.
Referencias	RF6.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2. El usuario selecciona la guía que desea visualizar.	2.1 El sistema muestra la interfaz para visualizar los contenidos.

Sección “Pre-visualizar contenidos”	
Acción del Actor	Respuesta del Sistema
4. El usuario selecciona el contenido que desea pre-visualizar.	4.1. El sistema pre-visualiza el contenido de la guía personalizada escogida por el usuario.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Prototipo de Interfaz	
Interfaz 1	

Tabla 5: Descripción del CU Visualizar Contenidos

3.8 Diagramas de clases del análisis

El diagrama de clases del análisis es uno de los artefactos que se generan en el modelo del análisis. Se utilizan para el modelado de las realizaciones de los casos de uso, mostrando las clases y sus relaciones. Las clases que se encuentran en estos diagramas se clasifican en Interfaz, Controladora y Entidad.

- Las clases interfaz se encargan de modelar la interacción del actor con el sistema.
- Las clases controladoras modelan los aspectos dinámicos del sistema, de forma tal, que puedan coordinar las acciones y los flujos de control.
- Las clases entidades modelan información que posee una larga vida y que a menudo es persistente. Suelen mostrar una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema.

3.8.1 Diagrama de clase del análisis del CU Gestionar EPG

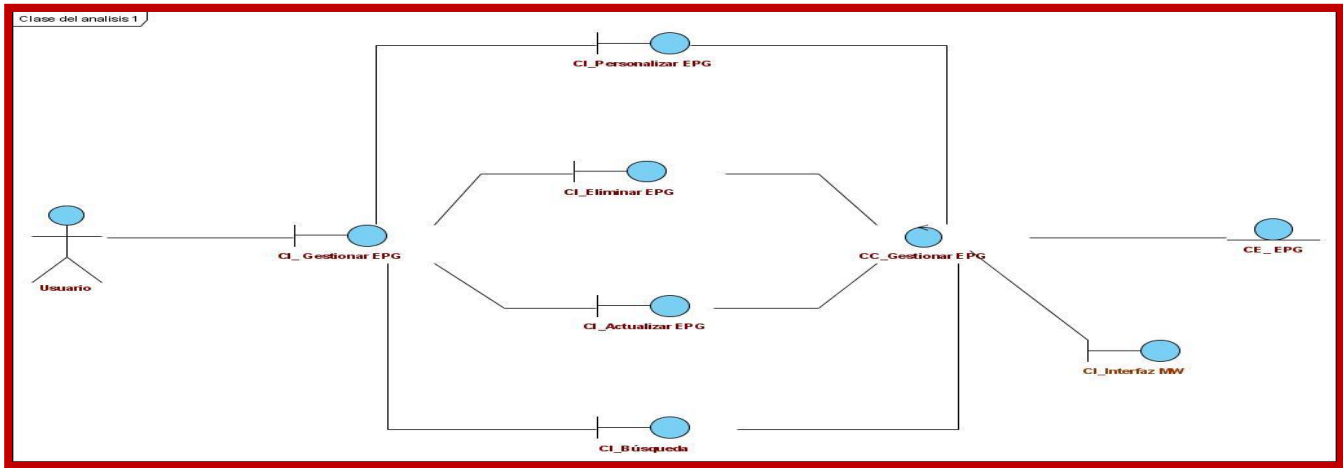


Figura 8: Diagrama de clase del análisis del CU Gestionar EPG

3.8.2 Diagrama de clase del análisis del CU Listar contenidos

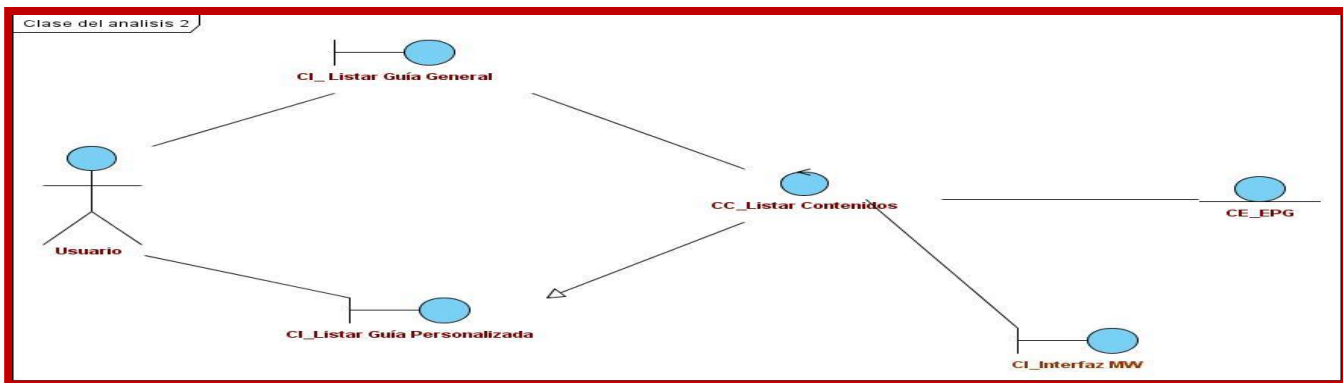


Figura 9: Diagrama de clase del análisis del CU Listar contenidos

3.8.3 Diagrama de clase del análisis del CU Visualizar contenidos

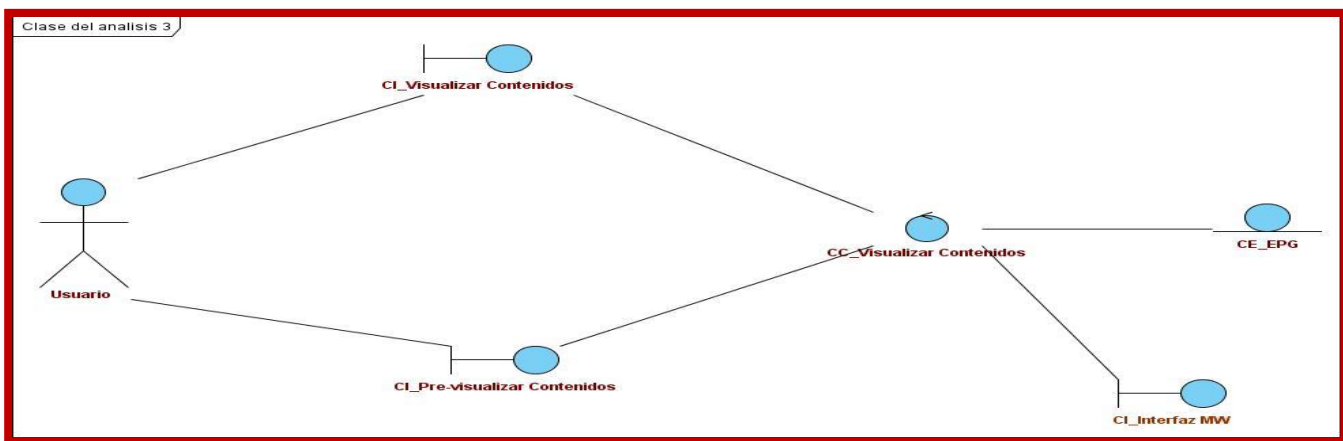


Figura 10: Diagrama de clase del análisis del CU Visualizar contenidos

3.9 Diagramas de colaboración del análisis

Los diagramas de colaboración ilustran las interacciones entre objetos en un formato de grafo o red, en el cual los objetos se pueden colocar en cualquier lugar del diagrama. (13)

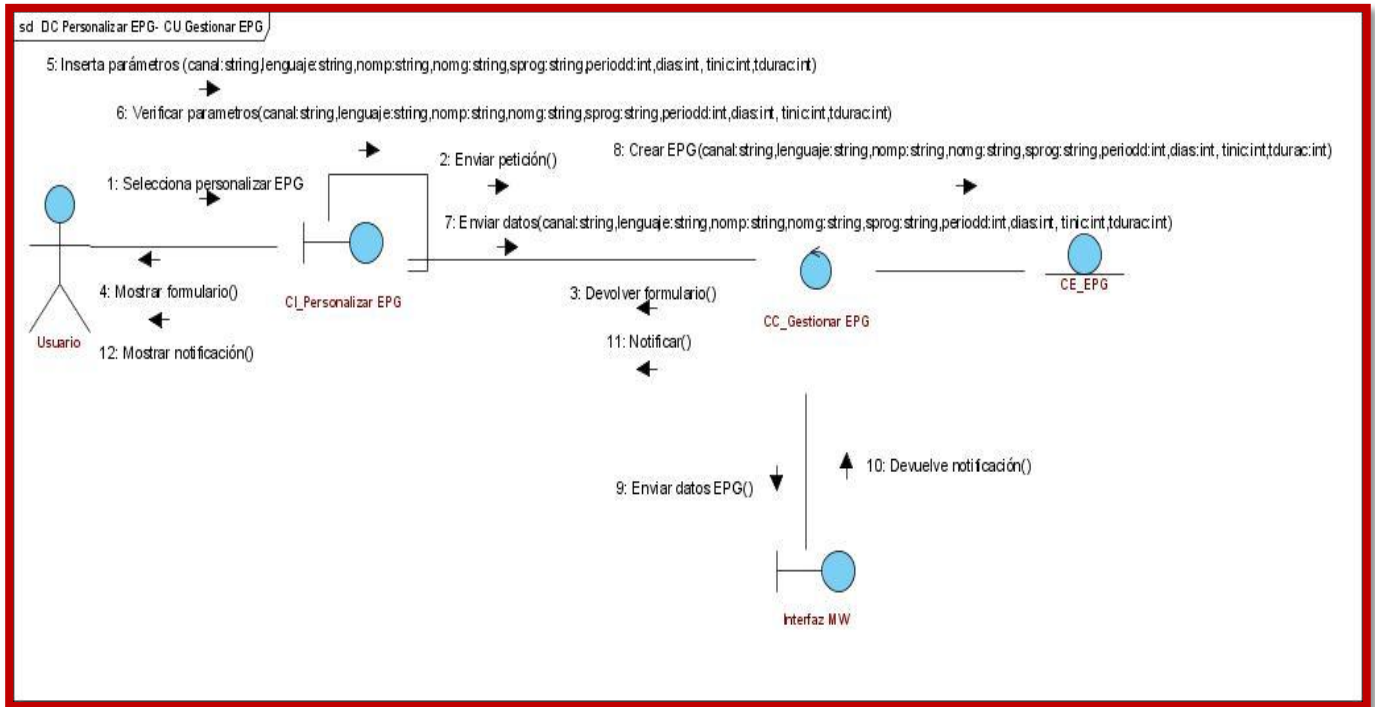


Figura 11: Diagrama de colaboración Personalizar EPG- CU Gestionar EPG

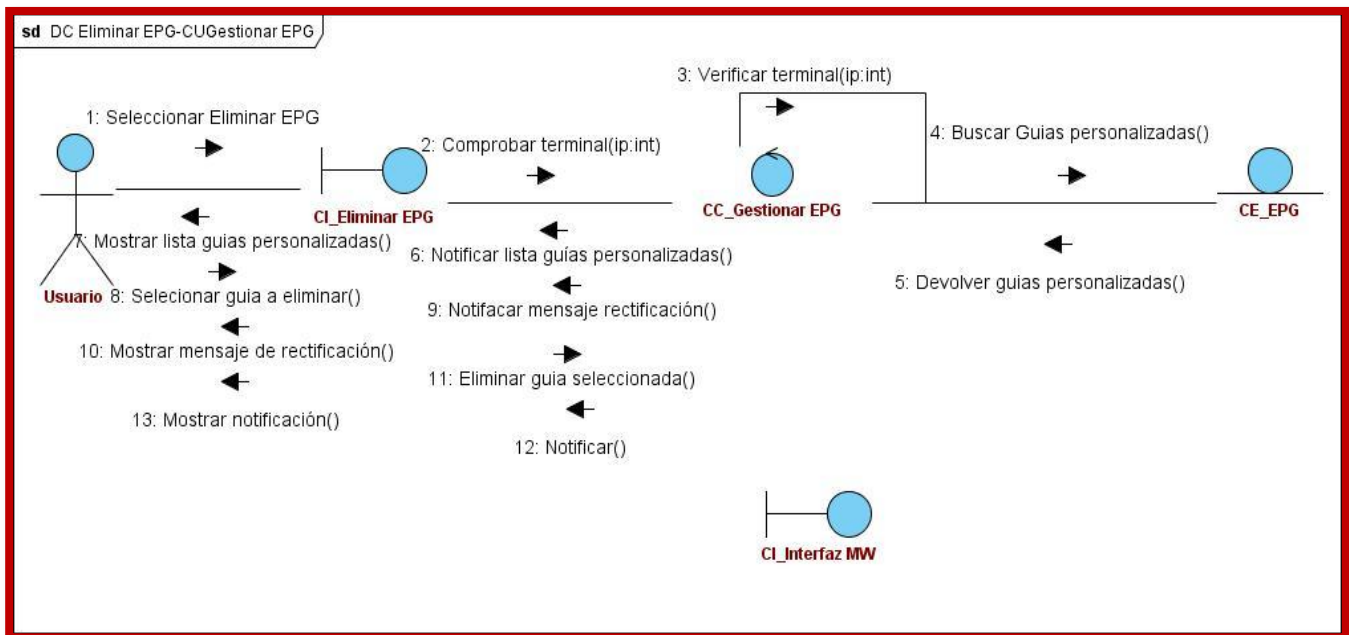


Figura 12: Diagrama de colaboración Eliminar EPG-CU Gestionar EPG

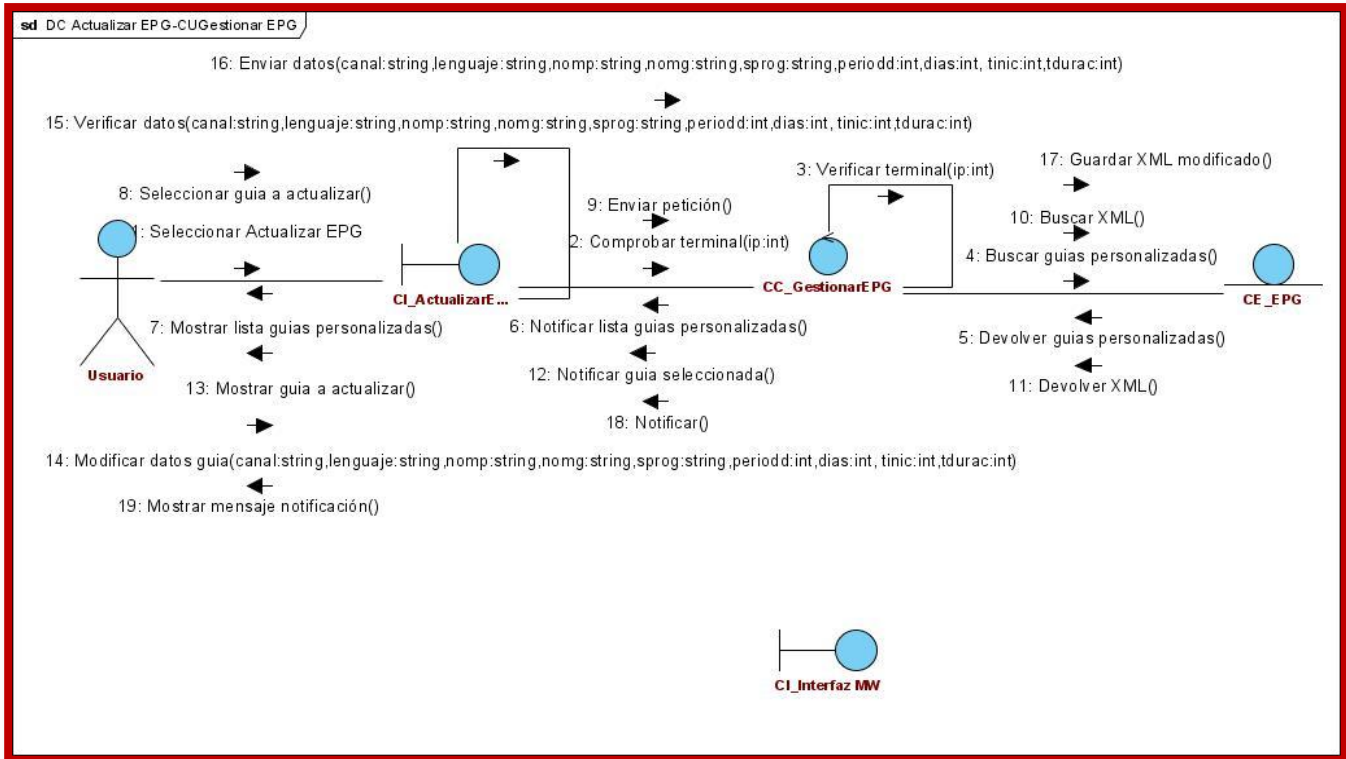


Figura 13: Diagrama de colaboración Actualizar EPG-CU Gestionar EPG

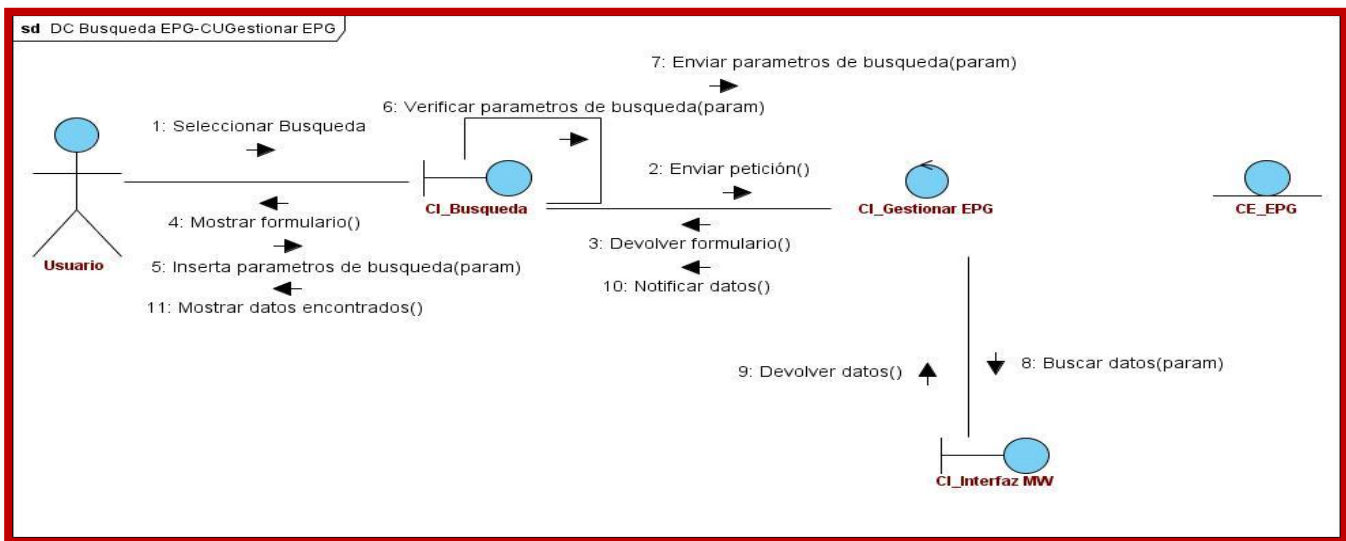


Figura 14: Diagrama de colaboración Búsqueda EPG-CU Gestionar EPG

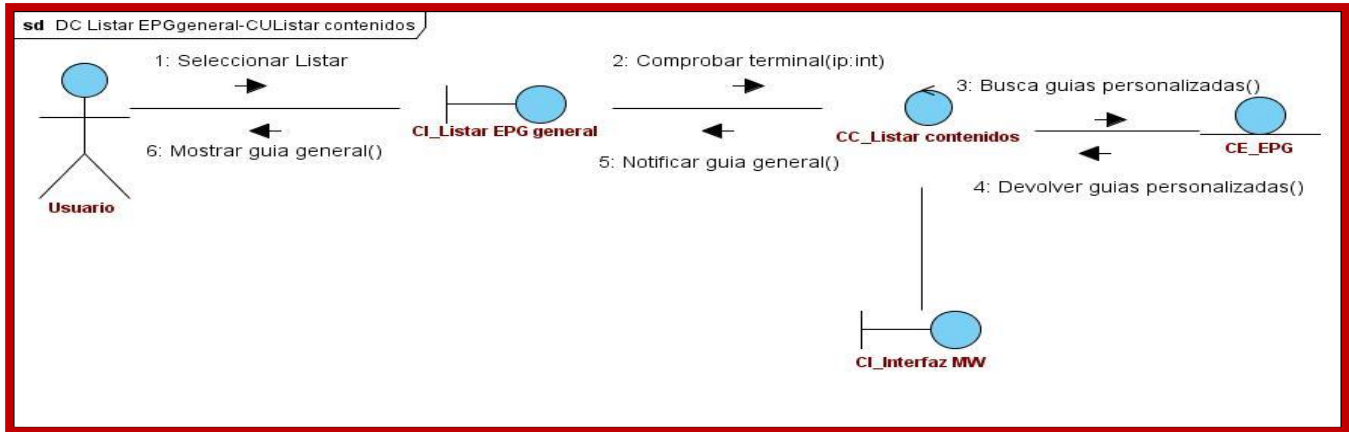


Figura 15: Diagrama de colaboración Listar EPG general-CU Listar contenidos

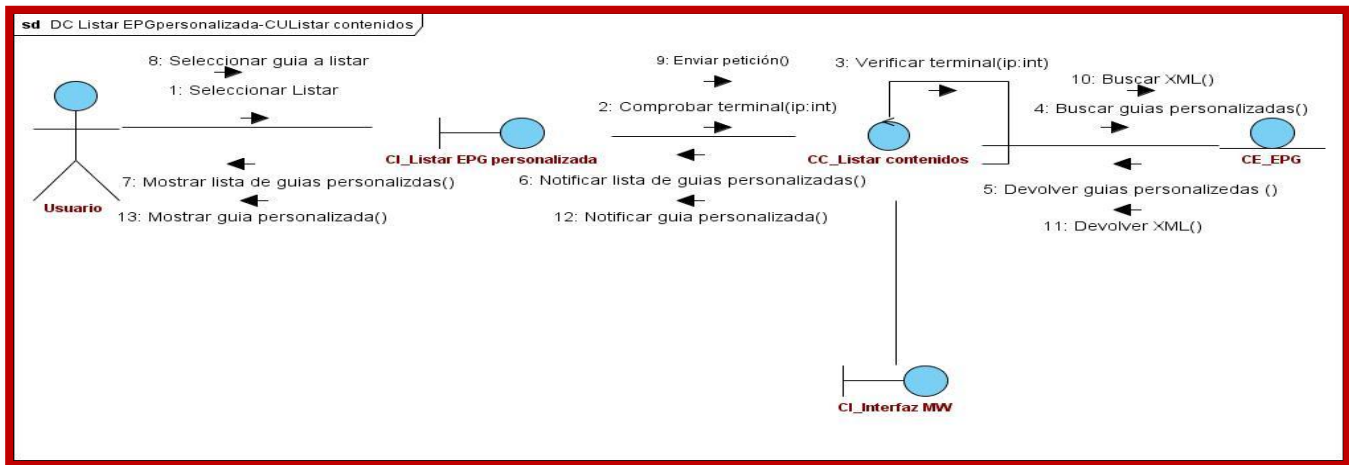


Figura 16: Diagrama de colaboración Listar EPG personalizada-CU Listar contenidos

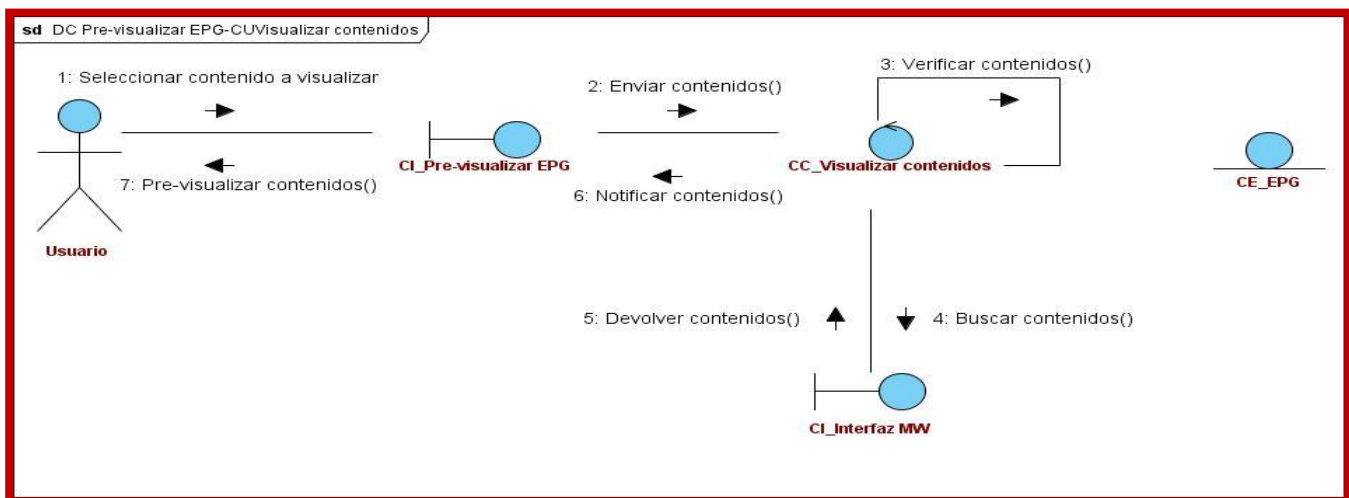


Figura 17: Diagrama de colaboración Pre-visualizar EPG-CU Visualizar contenidos

3.10 Métricas

La medición es fundamental para cualquier disciplina de ingeniería, y la ingeniería del software no es una excepción. La medición nos permite tener una visión más profunda proporcionando un mecanismo para la evaluación objetiva.

Las métricas del software se refieren a un amplio elenco de mediciones para el software de computadora. La medición se puede aplicar al proceso del software con el intento de mejorarlo sobre una base continua. Se puede utilizar en el proyecto del software para ayudar en la estimación, el control de calidad, la evaluación de productividad y el control de proyectos.

Además, el ingeniero de software, puede utilizar la medición para ayudar a evaluar la calidad de los resultados de trabajos técnicos y para ayudar en la toma de decisiones táctica a medida que el proyecto evoluciona.

Por tanto, se aplican las métricas para valorar la calidad de los productos de ingeniería o los sistemas que se construyen. Además que proporcionan una manera sistemática de valorar la calidad basándose en un conjunto de reglas claramente definidas. Se aplican a todo el ciclo de vida permitiendo descubrir y corregir problemas potenciales.

Los requisitos del software son la base de las medidas de calidad. Unos estándares específicos definen un conjunto de criterios de desarrollo que guían la manera en que se hace la ingeniería del Software, si no se siguen los criterios, habrá seguramente poca calidad. (14)

3.11 Métricas de calidad aplicadas a los artefactos generados

Métricas para la especificación de requisitos

Davis⁶ 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), compleción, corrección, comprensión, capacidad de verificación, consistencia interna y externa, capacidad de logro, concisión, trazabilidad, capacidad de modificación, exactitud y capacidad de reutilización. Además, 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. (14)

Aunque muchas de las características anteriores parecen ser de naturaleza cualitativa, Davis sugiere que todas puedan representarse usando una o más métricas.

⁶ Carl G Davis: Ingeniero de Software especialista en métricas de calidad orientadas a objetos.

Para determinar la especificidad (ausencia de ambigüedad) de los requisitos. Davis sugiere una métrica basada en la consistencia de la interpretación de los revisores: (14)

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

Donde n_{ui} es el número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas, n_r es el total de requisitos, es decir, la suma de los requisitos funcionales y los no funcionales. Cuanto más cerca de 1 esté el valor Q_1 , menor ambigüedad presentará la especificación.

Para realizar la revisión y obtener un buen análisis sin ambigüedades en los requisitos. Se realizaron 2 revisiones para eliminar todas las que pudieran existir, es por ello que durante la primera revisión se detectó que habíamos requisitos funcionales que realizan una misma acción por lo que no había claridad en el hecho de por qué estaban separados, como es el caso del Listar contenidos de la EPG general y Listar contenidos de la EPG personalizada y el caso del requisito de Rendimiento que no estaba claro la necesidad exigida. Por lo que de los 16 requisitos existentes los revisores tuvieron en 13, interpretaciones semejantes y en 3 requisitos se explicó que no estaban de acuerdo y no tuvieron concordancia por lo que se puede decir que:

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

$$Q_1 = 13 / 16$$

$$Q_1 = 0.81$$

Resultado de la primera revisión: como la especificidad en esta revisión tuvo un resultado superior al 80 %, se puede decir que el grado de ambigüedad de los requisitos no presenta mucha ambigüedad.

Durante la segunda revisión después de un estudio conciso por parte de los revisores todos llegaron a la misma interpretación con relación al requisito Listar contenidos de la EPG, por lo que se puede decir que de los 15 requisitos existentes los revisores acertaron en todos en cuanto a las interpretaciones.

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

$$Q_1 = 15 / 15$$

$$Q_1 = 1$$

Resultado de la segunda revisión: como la especificidad en esta revisión fue uno, se puede decir que no existe grado de ambigüedad de los requisitos.

Métricas para el grado de validación de requisitos

Para el grado de validación de requisitos se utiliza la métrica siguiente, que está dada por:

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

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

En la revisión que se efectuó, todos los requisitos quedaron como correctos, tanto los funcionales como los no funcionales, quedando satisfecho el cliente.

Con lo antes planteado, la métrica queda con los siguientes valores:

$$n_c = 15 \text{ y } n_{nv} = 0 \quad Q_3 = 15 / (15 + 0)$$

$$Q_3 = 1$$

Métricas aplicadas al Diagrama de Casos de Uso del Sistema

Para medir la calidad del DCUS se aplicó un modelo de métricas orientadas a objetos, ésta consta de 4 atributos los cuales son: Consistencia, Correctitud, Completitud y Complejidad. Estos atributos contienen varios factores los cuales tienen una métrica asociada que da un valor cuantitativo del umbral de calidad que tiene dicho modelo.

- La Consistencia se basa en el grado en que los casos de uso del sistema describen las interacciones entre el usuario y el sistema.
- La Correctitud consiste en el grado en que las interacciones entre el actor y el sistema soportan el modelo del negocio.
- La Completitud consiste en el grado en que se han detallado los casos de uso más relevante.
- La Complejidad viene dada por el grado de presentación de los elementos que describen el contexto y la claridad del sistema. (14)

Atributos	Factores
Completitud	<ul style="list-style-type: none"> ➤ ¿Están definidos todos los requisitos que justifican la funcionalidad de los casos de uso? ➤ ¿Se presenta una descripción detallada (descripción extendida esencial) de todos los casos de uso del sistema? ➤ ¿Están todas las acciones del flujo de eventos redactadas en función del responsable? ➤ ¿Se contemplan todas las posibles excepciones a la secuencia normal?

Complejidad	<ul style="list-style-type: none"> ➤ ¿Los elementos dentro del diagrama están adecuadamente ubicados de manera que facilitan su interpretación?
Correctitud	<ul style="list-style-type: none"> ➤ ¿Existe para cada caso de uso por lo menos un usuario responsable? ➤ ¿Representa el caso de uso un requisitos comprensibles por el usuario?
Consistencia	<ul style="list-style-type: none"> ➤ ¿El nombre dado a los casos de uso es una expresión verbal que escribe alguna funcionalidad relevante en el contexto del problema para el usuario? ➤ ¿La descripción del flujo de eventos se inicia con la descripción de una acción externa originada por un actor o por una condición interna del sistema claramente identificable? ➤ ¿Existe una adecuada separación entre el flujo básico de eventos y los flujos alternos?

Tabla 6: Métricas aplicadas al Diagrama de Casos de Uso del Sistema.

3.12 Conclusiones parciales

En el presente capítulo se hizo un análisis detallado del dominio del problema, donde se evidenció el estudio de los conceptos fundamentales por los cuales se realizó el modelo de clases conceptuales del dominio. Esto trajo consigo un buen entendimiento del tema, lográndose así definir los requerimientos funcionales y no funcionales del subsistema de la EPG. Se realizó el diagrama de Casos de Usos del Sistema donde se definieron los CU Gestionar EPG, Listar contenidos y Visualizar contenidos, así como la descripción de cada uno de estos. De manera que también se realizaron los diagramas de clases del análisis y diagramas de colaboración donde se evidencia el funcionamiento del subsistema mediante el envío de mensajes. Se aplicaron las métricas de calidad a los artefactos generados llegando a la conclusión, por los resultados obtenidos en las revisiones, que estos artefactos presentan poco nivel de ambigüedad, lo que trae consigo que tenga una buena calidad el análisis realizado.

Conclusiones generales

Con la culminación del análisis de la propuesta de la Guía Electrónica de Programación se pudo arribar a las siguientes conclusiones:

- La investigación ha permitido hacer un estudio de sistemas IPTV que utilizan Guías Electrónicas de Programación (EPG), lo cual ha posibilitado poder caracterizar las EPG y así entender el proceso de funcionamiento para realizar el análisis de este subsistema.
- A lo largo del desarrollo del análisis de la Guía Electrónica de programación se logró modelar el proceso de visualización de los contenidos en un sistema IPTV a través de la interrelación de los conceptos principales asociados a dicho proceso.
- Con la culminación del desarrollo de la investigación se logró definir los requerimientos funcionales y no funcionales, y se pudieron generar los artefactos: Diagrama de Casos de Uso del Sistema, Diagrama de Clases del Análisis y los Diagramas de Colaboración, dejando así todo listo para continuar con la fase de diseño del subsistema.

Recomendaciones

Como resultado del desarrollo y las conclusiones de esta investigación, se expresan las siguientes recomendaciones:

- Que se siga con el resto de las fases de desarrollo de la EPG para que pueda ser puesto en funcionamiento.
- Realizar un análisis más detallado de las recomendaciones de la UIT para este tipo de subsistemas.
- Incorporarle más funcionalidades y aplicaciones a la EPG como el grabador personal de video (PVR por sus siglas en inglés).

Bibliografía citada

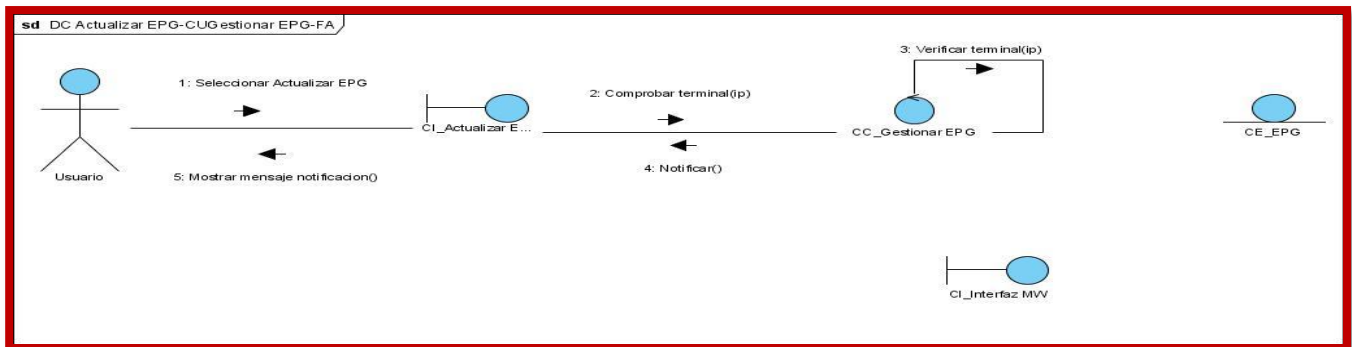
1. MINISTERIO DE INDUSTRIA, TURISMO Y COMERCIO. TDT(Televisión Digital Terrestre). [En línea] 1 de octubre de 2009. [Citado el: 1 de diciembre de 2009.] <http://www.televisiodigital.es>.
2. Huawei Technologies Co., Ltd. Huawei. [En línea] Huawei Technologies Co., Ltd, 2009. <http://www.huawei.com/es/catalog.do?id=-4>.
3. Martínez, Yarisel Rodríguez. Diseño de un Sistema Gestor de Guía Electrónica de Programación. Habana : Universidad de las Ciencias Informáticas, 2009.
4. Proyecto GNU. La Definición de Software Libre. [En línea] 2008. <http://www.gnu.org/philosophy/free-sw.es.html>.
5. EPG Technologies. [En línea] EPG Technologies, 2009. <http://www.epgtech.net/Services/SharepointDevelopment/tabid/139/epgfunc-2.aspx>.
6. MerchantCircle.com. [En línea] MerchantCircle Inc, 2006-2010. [Citado el: 25 de febrero de 2010.] <http://www.merchantcircle.com/business/Epg.Technologies.Inc.815-334-8387/article-124>.
7. WRN. [En línea] WRN, 2009. [Citado el: 2 de marzo de 2010.] <http://www.wrn.org/corporate/wrn-partners-with-all-in-media-to-licence-guide-software-2/>.
8. Vázquez, Carlos Luis Serrano Rosales , Solangel Rodríguez. Desarrollo de Biblioteca de Métodos Numéricos (BMN), referente a Sistemas de Ecuaciones Lineales, Sistemas de Gran Dimensión y Poco Densos e Integración Numérica. Ciudad de la Habana : Universidad de las Ciencias Informáticas, 2008.
9. José H. Canós, Patricio Letelier y M^a Carmen Penadés. Metodologías Ágiles en el Desarrollo de Software. Valencia : DSIC -Universidad Politécnica de Valencia.
10. Macias, Yolanda Benitez. Sistema para la gestión y control de la cartelera del canal cultural de la Universidad de las Ciencias Informáticas. Ciudad de la Habana : Universidad de la Ciencias Informáticas, 2008.
11. Cataldi, Zulma. Metodología de diseño, desarrollo y evaluación de software educativo. 2000.
12. BOOCH, GRADY, RUMBAUCH, JAMES y JACOBSON, IVAR. The UML specification documents. CA, Santa Clara : s.n., 1997.
13. Larman, Craig. UML y Patrones.
14. Pressman, Roger S. Ingeniería de Software. Un enfoque práctico. [ed.] Concepción Fernández. 5ta Edición. La Habana : Editorial Félix Varela, 2005. págs. 331-332.

Bibliografía consultada

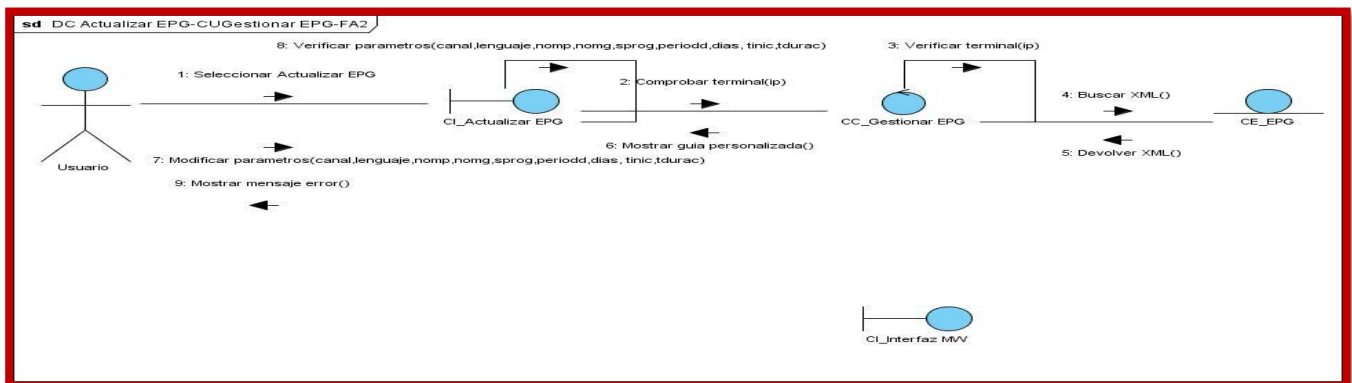
1. MINISTERIO DE INDUSTRIA, TURISMO Y COMERCIO. TDT(Televisión Digital Terrestre). [En línea] 1 de octubre de 2009. [Citado el: 1 de diciembre de 2009.] <http://www.televisiandigital.es>.
2. Huawei Technologies Co., Ltd. Huawei. [En línea] Huawei Technologies Co., Ltd, 2009. <http://www.huawei.com/es/catalog.do?id=-4>.
3. Martínez, Yarisel Rodríguez. Diseño de un Sistema Gestor de Guía Electrónica de Programación. Habana : Universidad de las Ciencias Informáticas, 2009.
4. Proyecto GNU. La Definición de Software Libre. [En línea] 2008. <http://www.gnu.org/philosophy/free-sw.es.html>.
5. EPG Technologies. [En línea] EPG Technologies, 2009. <http://www.epgtech.net/Services/SharepointDevelopment/tabid/139/epgfunc-2.aspx>.
6. MerchantCircle.com. [En línea] MerchantCircle Inc, 2006-2010. [Citado el: 25 de febrero de 2010.] <http://www.merchantcircle.com/business/Epg.Technologies.Inc.815-334-8387/article-124>.
7. WRN. [En línea] WRN, 2009. [Citado el: 2 de marzo de 2010.] <http://www.wrn.org/corporate/wrn-partners-with-all-in-media-to-licence-guide-software-2/>.
8. Vázquez, Carlos Luis Serrano Rosales , Solangel Rodríguez. Desarrollo de Biblioteca de Métodos Numéricos (BMN), referente a Sistemas de Ecuaciones Lineales, Sistemas de Gran Dimensión y Poco Densos e Integración Numérica. Ciudad de la Habana : Universidad de las Ciencias Informáticas, 2008.
9. José H. Canós, Patricio Letelier y M^a Carmen Penadés. Metodologías Ágiles en el Desarrollo de Software. Valencia : DSIC -Universidad Politécnica de Valencia.
10. Macias, Yolanda Benitez. Sistema para la gestión y control de la cartelera del canal cultural de la Universidad de las Ciencias Informáticas. Ciudad de la Habana : Universidad de la Ciencias Informáticas, 2008.
11. Cataldi, Zulma. Metodología de diseño, desarrollo y evaluación de software educativo. 2000.
12. BOOCH, GRADY, RUMBAUCH, JAMES y JACOBSON, IVAR. The UML specification documents. CA, Santa Clara : s.n., 1997.
13. Larman, Craig. UML y Patrones.
14. Pressman, Roger S. Ingeniería de Software. Un enfoque práctico. [ed.] Concepción Fernández. 5ta Edición. La Habana : Editorial Félix Varela, 2005. págs. 331-332.
15. Huidobro, José Manuel. IPTV, la televisión a través de internet. 2008.
16. Taltechnologies.net. [En línea] Taltechnologies.net. [Citado el: 2 de Diciembre de 2009.] www.taltechnologies.net.

17. Mundo 2.0-Contact. [En línea] Mundo-Contact. [Citado el: 10 de diciembre de 2009.]
http://www.mundo-contact.com/enlinea_detalle.php?recordID=1612.
18. Proyectosagiles.org. [En línea] proyectosagiles.org. [Citado el: 13 de diciembre de 2009.]
<http://www.proyectosagiles.org/>.
19. Aranova. [En línea] Aranova, 2006-2007. [Citado el: 5 de diciembre de 2009.]
<http://www.aranova.es>.
20. Fundación telefónica. [En línea] Fundación telefónica. [Citado el: 20 de Noviembre de 2009.]
http://sociedadinformacion.fundacion.telefonica.com/DYC/SHI/seccion=1188&idioma=es_ES&id=2009100116310030&activo=4.do?elem=4642.
21. Mediosdigitales.info. [En línea] mediosdigitales.info, 2006. [Citado el: 23 de enero de 2010.]
<http://mediosdigitales.info/2006/03/04/pocos-saben-que-es-iptv/>.
22. Pressman, Roger S. Ingeniería de Software. Un enfoque práctico. [ed.] Félix Varela. La Habana : s.n., 1998.

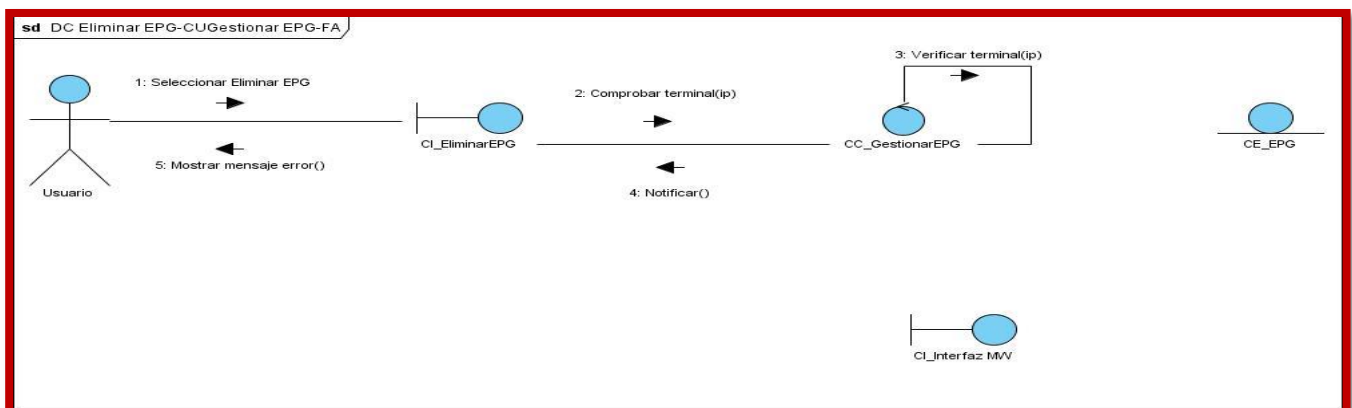
Anexo 1: Diagrama de colaboración Actualizar EPG-CU Gestionar EPG-Flujo Alternativo



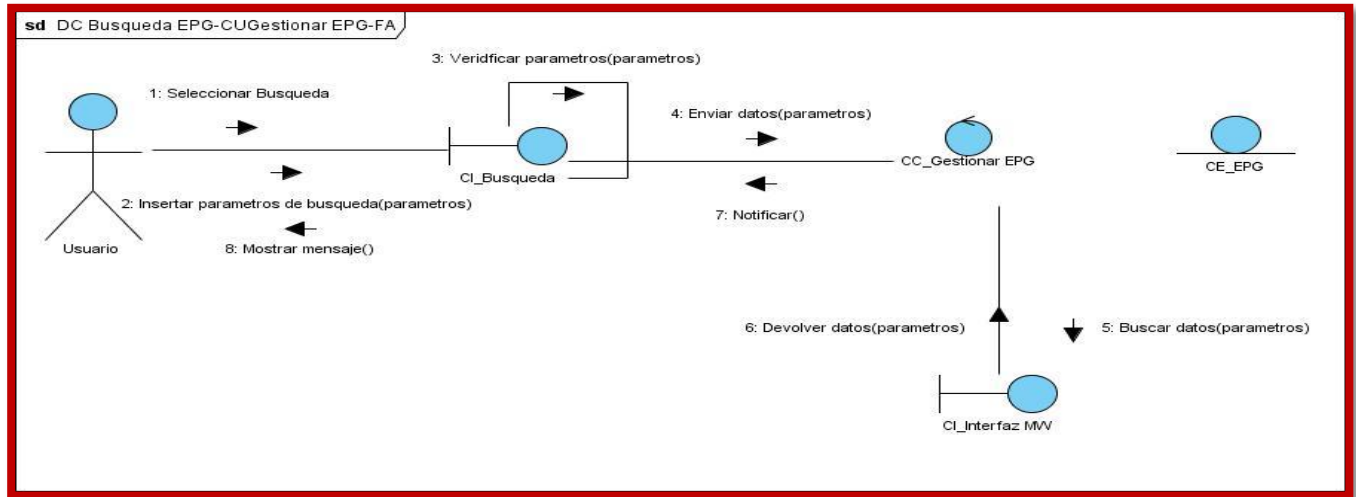
Anexo 2: Diagrama de Colaboración Actualizar EPG-CU Gestionar EPG-Flujo Alternativo 2



Anexo 3: Diagrama de Colaboración Eliminar EPG-CU Gestionar EPG-Flujo Alternativo



Anexo 4: Diagrama de Colaboración Búsqueda EPG-CU Gestionar EPG-Flujo Alterno



Glosario de términos

TV Digital: La televisión digital es el conjunto de tecnologías de transmisión y recepción de imagen y sonido que se transmite en formato digital, en contraste con la televisión estándar que codifica la señal de manera analógica. La televisión digital codifica la señal de forma binaria en ceros y unos. Proporciona una mayor calidad en la señal.

TV analógica: Tipo de televisión que usa la señal analógica. Hasta los primeros años del siglo XXI la televisión era analógica totalmente.

IPTV: La televisión digital por ADSL (Bucle de Abonado Digital Asimétrico, siglas en español) es el resultado de la aplicación de la tecnología digital a la señal de televisión. Para luego transmitirla por medio de protocolos asimétricos (XDSL por sus siglas en inglés).

EPG: La EPG es el acrónimo inglés de Electronic Program Guide, en español guía electrónica de programación, es una aplicación interactiva que se asemeja a un teletexto pero más avanzado. Con una alta calidad visual de sus gráficos y textos. Presenta como objetivo centrarse en la programación de canales.

Streaming: Transmisión de audio y video online de manera continua, se caracteriza por la visualización de los contenidos en el cliente sin la necesidad de esperar la descarga completa del fichero, lo que constituye una gran ventaja para el trabajo con medias debido al gran tamaño que pueden llegar a tener estas.

Metadatos: Datos de la información que esta almacenada en el sistema, describe las características del contenido, es decir información sobre la información guardada.

UML (Unified Modeling Language): Lenguaje Unificado de Modelado, usado para modelar sistemas de software.

Visual Paradigm: Herramienta de Ingeniería de Software Asistida por Computadora (CASE, por sus siglas en inglés, Computer-Aided Software Engineering) que permite realizar ingeniería tanto directa como inversa. Utiliza UML como lenguaje de modelado y RUP como base de desarrollo.

Middleware: El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.