



Facultad 9

Conceptualización del subsistema de transmisión de noticias como aplicación de escritorio para la Plataforma de Televisión Informativa.

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

Autora: Yaniley Méndez Terrero

Tutor: Ing. Reinier Castillo González

Co-Tutor: Msc. Yaneisis Pérez Heredia

Ciudad de la Habana, 2010

“Año del 51 Aniversario del Triunfo de la Revolución”

Declaración de autoría

Declaro que soy la única autora de este trabajo y autorizo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmo la presente a los 2 días del mes de julio del año 2010.

Firma de la autor:

Yaniley Méndez Terrero

Firma del tutor:

Reinier Castillo González

Datos de contacto

Ing. Reinier Castillo González

Profesor adiestrado graduado en el año 2008 de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas con Título de Oro. Profesor de la Facultad # 9. Ha impartido la asignatura de Ingeniería de Software I y II, desempeñándose como Jefe de asignatura. Actualmente imparte la asignatura de Programación IV. Ha jugado el rol de analista principal del proyecto PRIMICIA del Centro de desarrollo de Geoinformática y Sistemas Digitales (GEySED) en dicha facultad y Líder del equipo de desarrollo del proyecto VIRTE VALL en la facultad 15.

Correo electrónico: rcgonzalez@uci.cu



"Aquel que duda y no investiga, se torna no sólo infeliz, sino también injusto".

Blaise Pascal

Agradecimientos

A Dios en primer lugar, por haberme iluminado el camino correcto, por darme fe y confianza.

A la Revolución, por haberme dado la posibilidad de alcanzar uno de mis mayores sueños en la vida, el de formarme como profesional, lo cual no hubiese sido posible si no existiese la misma.

A mi familia en general por su apoyo y preocupación constante.

A todas a mis amistades que siempre me han ayudado y apoyado.

A todas aquellas personas que de una u otra forma me han apoyado a lo largo de estos cinco años, que aunque no mencione sus nombres no se hace necesario porque ellos saben quiénes son.

Dedicatoria

A mis padres Nilda Terrero y Luis Méndez por confiar en mí siempre, por brindarme valiosos consejos que me ayudaron a ser la persona que soy hoy, por alentarme y estar conmigo en las buenas y en las malas, en fin, por ser mis más valiosos tesoros.

A mi hermano Yanoski Méndez por su confianza infinita, por brindarme su fe y cariño, por los recuerdos tan lindos que gracias a él hoy tengo en mi mente, por ser parte de mi vida y por muchísimas cosas lindas que hemos protagonizado juntos.

A mi novio, compañero, amigo y tutor Reinier Castillo por estar a mi lado siempre, por brindarme su amor infinito, por enseñarme a triunfar, por todo eso y más, mil veces gracias mi amor.

A mis amigas de los 5 años de carrera Yenisel, Lisandra, Katy, Mariem y Erielis por compartir conmigo su inmensa amistad, momentos de tristezas y alegrías, por demostrarme que al final del día una sonrisa siempre es válida.

Resumen

En determinados entornos cerrados se hace necesario transmitir informaciones en formato texto en tiempo real o dinámicamente, como son las universidades, centros de convenciones, hoteles, terminales de transportación o sedes ministeriales o empresariales, en las que existen circuitos cerrados de televisión con el objetivo de mantener informados de manera rápida y constante a trabajadores, usuarios o personal en general. En la televisión la transmisión de materiales que contienen informaciones de texto mayormente se emiten previamente grabados y las soluciones existentes son altamente costosas y privativas.

Con el objetivo de satisfacer esta necesidad, en la Universidad de las Ciencias Informáticas un grupo de personas se han dedicado al desarrollo de sistemas informáticos para dar solución a la transmisión de noticias a través de canales de televisión utilizando computadoras. Señal ACN, Señal 3, la plataforma de televisión informativa PRIMICIA y otros sistemas de teletexto implantados en el país, constituyen muestras fehacientes de sistemas informáticos que permiten la transmisión de noticias. La gran mayoría de estos sistemas están desarrollados sobre herramientas con soporte para aplicaciones web, como es el caso de la plataforma PRIMICIA. El navegador o browser es el recurso utilizado por este tipo de aplicaciones para su visualización, el cual tiene como desventaja que suele poner lenta la aplicación, incluyendo las animaciones, efectos y disminuyendo así su propio rendimiento. Sin embargo, las tecnologías de escritorio permiten mejorar estas variables mediante las potencialidades que brindan permitiendo un gran desempeño tanto de hardware como de software.

En el presente trabajo se definen las tecnologías que soportan el desarrollo del subsistema de transmisión de la Plataforma de Televisión Informativa (PRIMICIA) aprovechando las bondades que brindan las tecnologías de escritorio. A partir de las variables animación, efectos gráficos y rendimiento, se caracterizaron, compararon y seleccionaron las tecnologías de escritorio que permitirán el desarrollo del subsistema de transmisión de noticias.

Palabras claves: televisión, transmisión, tecnología.

ÍNDICE

Introducción.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1.1 INTRODUCCIÓN.....	4
1.2 CONCEPTOS ASOCIADOS AL DOMINIO DEL PROBLEMA.	4
1.2.1 <i>Procesos de transmisión en canales de Televisión.</i>	4
1.2.2 <i>Plataforma de Televisión Informativa (PRIMICIA)</i>	5
1.2.3 <i>Aplicaciones multimedia</i>	5
1.2.4 <i>Aplicaciones Web</i>	6
1.2.5 <i>Aplicaciones escritorio.</i>	6
1.2.6 <i>Software Libre.</i>	6
1.3 OBJETO DE ESTUDIO.	7
1.3.1 <i>Descripción General</i>	7
1.3.2 <i>Descripción actual del dominio del problema.</i>	8
1.3.3 <i>Situación problemática.</i>	10
1.4 ANÁLISIS DE OTRAS SOLUCIONES EXISTENTES.....	11
1.4.1 <i>Sistemas de Teletexto en el ámbito internacional.</i>	11
1.4.2 <i>Ámbito nacional.</i>	13
1.5 VALORACIÓN DEL AUTOR SOBRE LOS CRITERIOS EMITIDOS POR LOS ESPECIALISTAS. .	15
1.6 CONCLUSIONES PARCIALES.	15
CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES.....	17
2.1 INTRODUCCIÓN.	17
2.2 LENGUAJES DE PROGRAMACIÓN.....	17
2.2.1 <i>C ++</i>	17
2.2.2 <i>Java</i>	18
2.2.3 <i>C Sharp</i>	20
2.2.4 <i>Fundamentación del Lenguaje de Programación escogido.</i>	21
2.3 LENGUAJE UNIFICADO DE MODELADO (UML).	22
2.4 PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE (RUP).	23
2.5 FRAMEWORK.	24
2.5.1 <i>Platinum C++ Framework.</i>	25
2.5.2 <i>Qt 4.6.0</i>	25
2.5.3 <i>Fundamentación del framework escogido.</i>	28
2.6 ENTORNO DE DESARROLLO INTEGRADO (IDE)	29
2.6.1 <i>Eclipse</i>	29
2.6.2 <i>Qt Creator 1.3.0</i>	30
2.6.3 <i>Netbeans</i>	31
2.6.4 <i>Fundamentación del IDE escogido.</i>	32
2.7 SISTEMA GESTOR DE BASES DE DATOS.	33
2.7.1 <i>MySQL</i>	33
2.7.2 <i>Oracle</i>	34
2.7.3 <i>PostgreSQL 8.3</i>	35
2.7.4 <i>Fundamentación del SGBD escogido.</i>	36
2.8 HERRAMIENTAS DE MODELADO DE SOFTWARE.	37
2.8.1 <i>Rational Rose</i>	37
2.8.2 <i>Visual Paradigm.</i>	38
2.8.3 <i>Fundamentación de la herramienta de modelado escogida.</i>	39
2.9 HERRAMIENTA CONTROL DE VERSIONES.	39

2.9.1 <i>Subversion</i>	40
2.9.2 CVS	41
2.9.3 <i>Fundamentación de la herramienta de control de versiones escogida</i> . ..	42
2.10 DESCRIPCIÓN DE OTRAS TECNOLOGÍAS NECESARIAS EN LA PROPUESTA.....	42
2.10.1 <i>Soporte para el uso de XML</i>	42
2.10.2 <i>Soporte de medias</i>	45
2.11 CONCLUSIONES PARCIALES	51
CAPÍTULO 3. PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS	52
3.1 INTRODUCCIÓN.....	52
3.2 HERRAMIENTAS Y TECNOLOGÍAS PROPUESTAS.....	52
3.3 CARACTERÍSTICAS DEL MÉTODO DELPHI.	53
3.3.1 <i>Fase Preliminar</i>	54
3.3.2 <i>Fase Exploratoria</i>	55
3.3.3 <i>Fase Final</i>	56
Conclusiones Generales	63
Recomendaciones	64
Referencias Bibliográficas	65
Bibliografía	67

ÍNDICE DE TABLAS Y FIGURAS

Tabla 1: Facilidades de sonido en plataformas.....	46
Tabla 2: Formatos de imágenes en Qt.....	49
Figura 1: Sistema Teletexto Fingertext.....	12
Figura 2: RUP en Dos Dimensiones.....	24
Figura 3: Comunicación entre Aplicaciones usando XML.....	43
Figura 4: Esquema de dos analizadores.....	44

Introducción

Los avances tecnológicos han ido evolucionando considerablemente en el mundo con medios de comunicación mucho más eficientes, que permiten la difusión de la información con una alta velocidad a gran parte del mundo. Estos medios desde sus orígenes, han sido y continúan siendo, una referencia clave para el ciudadano en el conjunto de sus actividades cotidianas. Además, permiten conocer el entorno social, las opiniones sobre los temas de la actualidad que modifican la opinión pública y establecen nuevas formas de vínculos y relaciones sociales.

En los países capitalistas los medios de comunicación masiva han asumido una transformación causada por el propio sistema. En estos se mantienen en menor medida la televisión pública y en mayor medida se impone la televisión pagada, que debido a la privatización de sus medios su programación va dirigida al consumismo.

Con el triunfo de la Revolución Cubana, la televisión y la radio poseen un carácter gratuito, de contenido totalmente social y educativo, con el objetivo de contribuir a la formación cultural y educativa de los ciudadanos. A pesar del bloqueo económico impuesto por más de 50 años, nuestro país no se mantiene exento de los adelantos en el mundo de la tecnología de la información. Mediante la computación se han ido desarrollando las crecientes necesidades del hombre, donde los ordenadores juegan un papel importante. Estos están dotados de una gran capacidad de procesamiento para interpretar y analizar de manera muy rápida y eficiente la información. El ordenador es por excelencia, el medio más utilizado debido a la interactividad y prestaciones que brinda. Con el uso de éste, es posible acceder a sitios Web que albergan contenidos tanto de texto como de multimedia, posibilitando que la información universal pueda ser accedida por millares de personas a la vez en tiempo real. Todos estos avances han permitido la informatización de la sociedad.

En medio de la batalla por la informatización de la sociedad surge la Universidad de las Ciencias Informáticas (UCI) como una brillante idea de nuestro Comandante en Jefe. A raíz del surgimiento de la UCI y debido al elevado número de estudiantes que concentra se crean 10 facultades, entre las que se encuentra la Facultad 9. Cada una de estas facultades tiene bien definidas sus líneas investigativas, las cuales propiciaron la creación de polos productivos para desarrollarlas. Un ejemplo de ello es el polo Video y Sonido Digital, actual departamento de Sistemas Digitales, que forma parte de facultad 9 y es el encargado en gran medida de desarrollar productos que posibiliten la transmisión de audio, video e imágenes empleando para ello las potencialidades de los medios de comunicación. En correspondencia con una de las líneas de investigación del polo, se desarrolló la primera versión de la Plataforma de

Televisión Informativa con tecnología web (PRIMICIA), que permite la administración y transmisión de noticias de un canal de televisión informativa que soluciona gran parte de la necesidad de los usuarios, aunque no cubre todas las expectativas.

Como consecuencia de lo antes mencionado se hace necesario incorporarle al producto PRIMICIA las potencialidades que brindan las tecnologías de escritorio en el manejo y transmisión de las noticias. Es por ello que se determinó el siguiente **problema a resolver**: ¿Qué tecnologías utilizar para la creación del subsistema transmisión de noticias en la Plataforma de Televisión Informativa?

Como **objeto de estudio**: las tecnologías que potencian la transmisión, cuyo **campo de acción** son: las tecnologías de escritorio que permitan potenciar la transmisión de noticias del subsistema transmisión de la Plataforma de Televisión Informativa.

Se plantea como **objetivo general**: proponer las tecnologías que permitan potenciar la transmisión de noticias en el subsistema transmisión de la Plataforma de Televisión Informativa.

Hipótesis: El subsistema de transmisión desarrollado con tecnologías de escritorio permite potenciar la animación, los efectos gráficos y el rendimiento, en el proceso de diseño y transmisión de noticias.

Variables:

VD1 efectos gráficos

VD2 animación

VD3 rendimiento

VI4 tecnologías de escritorio

Para dar cumplimiento al objetivo anteriormente planteado se definen las siguientes **tareas de la investigación**:

- Valorar los criterios emitidos por especialistas sobre las tecnologías que posibilitan la creación de una aplicación de escritorio que permita la transmisión de noticias.
- Caracterizar las tecnologías potenciales que formarán parte de la propuesta de solución.
- Comparar las diferentes tecnologías identificadas.
- Seleccionar las tecnologías que formarán parte de la propuesta de solución para el subsistema de transmisión.

- Proponer las tecnologías para el subsistema de transmisión de noticias de la Plataforma de Televisión Informativa.
- Validar la propuesta de las tecnologías seleccionadas para el subsistema de transmisión de noticias de la Plataforma de Televisión Informativa.

Como **métodos de investigación científica** se utilizaron:

Métodos Teóricos:

Análisis Histórico-Lógico: Este método es de suma importancia ya que permitirá realizar un análisis de cómo han evolucionado los sistemas que permiten la transmisión de noticias a lo largo de los años, en los distintos momentos sujetos a las diferentes corrientes de desarrollo científico. Para de esta forma poder obtener una tendencia del uso de las tecnologías que se emplearán en la construcción del subsistema de transmisión de noticias como una aplicación de escritorio.

Analítico-Sintético: Este método permitirá el análisis de todos los procesos de transmisión de noticias que presentan los diversos sistemas informativos, para de esta forma encontrar las características más idóneas que serán incorporadas a la Plataforma de Televisión Informativa.

Métodos Empíricos:

Entrevista a especialistas del proyecto: Con este método se realizaron entrevistas formales para obtener diferentes puntos de vista respecto a la idea a defender.

Al concluir el trabajo de diploma se espera obtener como **posible resultado:**

- Conceptualización de las tecnologías para la creación de una aplicación de escritorio que permita la transmisión de noticias, realizando la justificación teórica de las mismas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

1.1 Introducción.

En el presente capítulo se abordan los conceptos fundamentales que posibilitan un mejor entendimiento de los procesos vinculados al campo de acción. Estos conceptos están basados en las definiciones previas realizadas por diversos especialistas en el tema.

1.2 Conceptos asociados al dominio del problema.

❖ Procesos de transmisión en canales de Televisión.

La prehistoria de la televisión comienza a partir de los descubrimientos técnicos más elementales, que hicieron posible la transmisión y recepción de imágenes en movimiento y sonido a distancia. Durante el siglo pasado surgen los dos primeros modelos inventados por el hombre: la televisión mecánica y la electrónica. En 1937 comenzaron las transmisiones regulares de TV electrónica en Francia y en el Reino Unido. Esto llevó a un rápido desarrollo de la industria televisiva y a un incremento de telespectadores. Es a mediados del siglo XX donde la televisión se convierte en bandera tecnológica de los países europeos y cada uno de ellos va desarrollando sus sistemas de TV nacionales y privados. En 1953 se crea Eurovisión que asocia a varios países de Europa conectando sus sistemas de TV mediante enlaces de microondas. Unos años más tarde, en 1960, se crea Mundovisión que comienza a realizar enlaces con satélites geoestacionarios cubriendo todo el mundo.

Actualmente el proceso de transmisión de las televisoras puede ser de forma tradicional o automatizada. La tradicional consiste en operar un *switch*, al cual llegan señales de video de distintas fuentes como son: grabadoras de cintas de video o VTR (acrónimo del inglés video tape recorder), señales directas del estudio o de otras televisoras. La persona encargada de realizar estas operaciones se guía por una escaleta, que no es más que un esqueleto o esquema del video que ayuda a organizar el plan de grabación y facilita la escritura del pre-guion y del guion definitivo, permitiendo así especificar el orden en que se mostrará cada espacio y el tiempo de duración. En el territorio nacional el canal Cubavisión está implementado basado en la forma tradicional. El avance de la informática, tanto a nivel del *hardware* como del *software* han dado la posibilidad de crear sistemas que automatizan el proceso de transmisión, esto posibilitó gestionar la escaleta, tener control total de las líneas de exterior y entradas de satélites, con grabación automática, titulación y

gráficos, control de VTRs, reproducción de videos desde disco duro en formatos como MPEG-2, AVI3 entre otros.

❖ **Plataforma de Televisión Informativa (PRIMICIA).**

PRIMICIA, es un producto informático creado en el año 2005 por la Universidad de las Ciencias Informáticas (UCI) en el Polo productivo de Video y Sonido Digital, actualmente departamento de Sistemas Digitales de la Facultad 9. Constituye una solución integral que permanece en constante evolución y perfeccionamiento capaz de proveer un canal de televisión para la transmisión automática y constante de informaciones en distintos formatos. El sistema cuenta con funcionalidades genéricas fácilmente escalables, que no dependen de un entorno dado y que no están ligadas a un diseño gráfico específico. De acuerdo a las necesidades de transmisión de informaciones en distintos formatos, el sistema permite la transmisión de contenidos en los siguientes soportes de información: Texto, Texto – Imagen e Imagen y Video. La aplicación está orientada a proveer un canal de televisión básicamente con características informativas, donde se integran contenidos de texto, imagen fija y video. El sistema se encuentra estructurado en dos subsistemas que se relacionan entre sí y actúan como un todo para brindar un resultado final a través de una red de televisión. El Subsistema de Administración es el responsable de que se realice la administración del canal y toda la gestión de las noticias y recursos de multimedia. El Subsistema de Transmisión es el encargado de visualizar las noticias y materiales publicados. Las características de PRIMICIA hacen que pueda ser ajustable a diferentes clientes, por lo que constituye un producto informático del cual pueden beneficiarse televisoras, agencias de noticias, empresas, terminales de transporte, hoteles y cualquier otra entidad que tenga una red de televisión con necesidades de transmitir informaciones en distintos formatos.

❖ **Aplicaciones multimedia.**

El término multimedia se utiliza para referirse a cualquier objeto o sistema que utiliza múltiples medios de expresión (físicos o digitales) para presentar o comunicar información. Según Tay Vaughan, una aplicación multimedia es *cualquier combinación de texto, arte gráfico, sonido, animación y video que llega a usted por cualquier computadora u otros medios*. Las áreas en las que la multimedia cobra más importancia son: la educación, la cultura y el ocio, sin dejar de lado las aplicaciones profesionales (1). De acuerdo a lo anterior se puede definir a un sistema automatizado de televisión aquel que integre en su visualización distintos soportes de comunicación como texto,

imagen, video y sonido como una aplicación multimedia. De esta manera se plantea el desarrollo de un sistema multimedia para la transmisión de las informaciones a través de la televisión.

❖ **Aplicaciones Web.**

Una aplicación web, es una página web especial que tiene una base de datos asociada y que permite una mayor interacción del usuario. Jim Conallen define una aplicación web como: un *sitio donde la navegación y la entrada de datos por parte de un usuario, afectan el estado de la lógica del negocio (2)*. En esencia, una aplicación web usa un sitio como entrada a una aplicación típica. Si no existe lógica del negocio en el servidor, el sistema no puede ser llamado aplicación web. Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. El sistema informativo debe contar con un medio centralizado de almacenamiento, al cual el Departamento Editorial pueda acceder y actualizar las informaciones, así como obtener reportes de las mismas. Mediante el desarrollo de una aplicación web y gracias a sus características, fue posible obtener un sistema de administración para la Plataforma de Televisión que posibilita de manera eficiente las diferentes funcionalidades requeridas.

❖ **Aplicaciones escritorio.**

Se denominan aplicaciones de escritorio a toda aplicación que ha sido desarrollada para ser ejecutada en una plataforma específica, ya sea Windows, Linux, Mac u otras. El desarrollo sobre una plataforma normalmente implica que la aplicación "no" pueda ser ejecutada en otras. Sin embargo, este tipo de aplicación brindaría a la Plataforma de Televisión Informativa ventajas como: mayor capacidad gráfica visual, menor tiempo de respuesta (aplicación más rápida) y mayor personalización.

❖ **Software Libre.**

- Software libre es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y por tanto, una vez obtenido puede ser usado, copiado, estudiado, cambiado y redistribuido libremente.

- El software libre es una cuestión de la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Esto significa precisamente que los usuarios de programas tienen las cuatro libertades esenciales.
- La libertad de ejecutar el programa, para cualquier propósito (libertad 0).
- La libertad de estudiar cómo trabaja el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.
- La libertad de redistribuir copias para que pueda ayudar al prójimo (libertad 2).
- La libertad de mejorar el programa, publicar sus mejoras y versiones modificadas en general, para que se beneficie toda la comunidad (libertad 3). El acceso al código fuente es una condición necesaria.

Cuba ha apostado en los últimos años por la utilización de tecnologías y plataformas libres para el desarrollo de la industria cubana del software. Todo ello con vistas a lograr de esta manera una soberanía tecnológica en este sentido, evitar la compra de costosas licencias que imponen los monopolios informáticos y que en muchas ocasiones le es vetada la compra de las mismas a causa del bloqueo económico al que se ha visto sometida por más de 50 años. Son estas las principales razones por lo que se plantea el desarrollo de la solución integral completamente sobre plataformas libres.

1.3 Objeto de Estudio.

1.3.1 Descripción General.

La Plataforma de Televisión Informativa (PRIMICIA), es un sistema que cuenta con funcionalidades genéricas que pueden ser fácilmente adaptadas a un entorno de negocio determinado. La aplicación orientada a proveer un canal de televisión básicamente con características informativas, puede ser desplegada sobre dos variantes de distribuciones físicas, con la pretensión de adecuarse al cliente valorando aspectos vitales como presupuesto o cantidad de información generada. Una variante es utilizar un servidor para la gestión de los procesos de administración y transmisión.

La otra alternativa posible es la utilización de dos servidores especializados para la gestión independiente de los procesos de administración y transmisión. El sistema se encuentra estructurado en dos subsistemas que se relacionan entre sí: el de administración que es donde se realiza la administración del canal, la gestión de noticias y recursos multimedia, y el subsistema de transmisión, el cual es el encargado

de visualizar las noticias y los materiales publicados. Llegado a este punto, se expone a continuación de forma detallada el proceso de transmisión a través de la funcionalidad de este subsistema:

Prestaciones del Subsistema de Transmisión:

- Generar una cartelera del ciclo de transmisión, mostrando para cada noticia la sección temática y el titular, en el orden que se van a visualizar.
- Visualizar noticias compuestas por pantallas de tipo Texto, Texto-Imagen, Imagen y Video.
- Reproducir un fondo musical mientras se muestran las noticias, excepto cuando se muestra un Video.
- Mostrar en las pantallas de tipo Imagen un comentario que orienta al televidente acerca de lo que está observando.
- Mostrar cintillos informativos o infocintas para promocionar eventos de última hora o acontecimientos de gran importancia.
- Mostrar como información adicional a la noticia: la fecha, hora, tiempo restante de la pantalla, el titular de la próxima noticia y la sección temática.
- Transmitir televisión en vivo proveniente de una señal externa.
- Mostrar el patrón del canal cuando éste se encuentre fuera de servicio.

1.3.2 Descripción actual del dominio del problema.

En la UCI se han desarrollado sistemas informativos automatizados que han sido de gran utilidad para la propia universidad (*Señal 3*) y para otras entidades como el Sistema Automatizado de Teletexto para la Plataforma de Televisión Digital Satelital Cubana, *Señal ACN*, administrado por la Agencia de Cubana de Noticias (*ACN*). Estos sistemas han solucionado problemas a la medida de estas entidades posibilitando la transmisión de noticias televisivas de una forma organizada y automatizada. A partir de estas experiencias se ha logrado conceptualizar y desarrollar el producto informático *PRIMICIA* como una solución integral que permite proveer un canal de televisión informativa completamente automatizado.

El desarrollo del sistema se realizó completamente utilizando software libre, siendo el resultado del empleo e integración de las siguientes herramientas y lenguajes de programación:

- Sistema Operativo: NOVA/LINUX.
- Herramienta de Gestión de Proyecto: dotProject.
- Herramienta de Control de Versiones: Subversion.
- Herramienta de Modelado: Visual Paradigm.
- Entorno de Desarrollo Integrado (IDE): Eclipse PDT.
- Framework de Desarrollo: Symfony.
- Sistema Gestor de Base de Datos: PostgreSQL.
- Lenguajes de Programación: PHP, Javascript, HTML.

Como mismo *PRIMICIA* presenta cualidades ventajosas también se hace evidente la existencia de cualidades negativas. Ejemplo de esto está que el proceso de redacción es sumamente limitado ya que solo se enfoca en cómo va a quedar estructurada la noticia pasando por alto la forma en que se visualizará la misma, evidenciando el hecho de que no le permite al redactor pre - visualizar la noticia durante la redacción. No posibilita tampoco adecuar los elementos visuales de la noticia a su conveniencia, cosa que no debe de ser así porque nadie más que el redactor tiene claramente la intención del mensaje que quiere transmitir y si éste tuviera la oportunidad de adecuar estos elementos la noticia quedaría más completa.

Las propiedades de estos elementos presentes en el proceso de redacción de *PRIMICIA* cambian frecuentemente durante la redacción de la noticia, por lo que no es conveniente que sean almacenadas en la base de datos ya que provocaría sucesivas actualizaciones y produciría una sobrecarga de la misma que atentaría con su mantenimiento con el rendimiento de la aplicación en general. Además solo se requiere que perdure en el tiempo el contenido de la noticia y no la forma en que se presentará la misma. Para almacenar la configuración de todos los elementos que conforman la noticia se estructurará un formato en XML (Extensible Markup Language) haciendo uso de las potenciales que presenta el mismo.

1.3.3 Situación problemática.

En determinados entornos cerrados se hace necesario transmitir informaciones en formato texto en tiempo real o dinámicamente. Un ejemplo de ello son las universidades, centros de convenciones, hoteles, terminales de transportación o sedes ministeriales o empresariales en las que existen circuitos cerrados de televisión con el objetivo de mantener informados de manera rápida y constante a trabajadores, usuarios o personal en general. Con el objetivo de satisfacer esta necesidad, en la Universidad de las Ciencias Informáticas un grupo de personas se han dedicado al desarrollo de sistemas informáticos para dar solución a la transmisión de canales de televisión utilizando computadoras. De esta manera se ha logrado alcanzar conocimientos sobre el “cómo hacer” este tipo de aplicaciones. Durante varios años de trabajo se han obtenido resultados satisfactorios con soluciones a la medida como *Señal 3* y *Señal ACN*, desarrollando además la Plataforma de Televisión Informativa (*PRIMICIA*). Esta última no es más que una solución integral capaz de proveer un canal de televisión para la transmisión automática y constante de informaciones en distintos formatos a través de una red de televisión adecuándose a las diferentes vías de transmisión existentes.

Este sistema fue desarrollado sobre plataformas libres y está compuesto por dos subsistemas: Administración y Transmisión. Mediante la administración del sistema es posible gestionar los usuarios de la plataforma, la transmisión del canal, los recursos multimedia para la confección de las noticias y permite llevar a cabo los procesos de redacción y publicación de la misma. Por su parte el subsistema de transmisión es el encargado de visualizar las noticias publicadas durante el proceso editorial. La plataforma está soportada con herramientas para el desarrollo de aplicaciones web que permiten una mayor interacción del usuario y ciertas flexibilidades dado sus características. Sin embargo, actualmente no cuenta con una versión de escritorio que facilite a la aplicación un menor tiempo de respuesta, mayor personalización y una mayor animación de las imágenes y texto, elementos fundamentales para clientes que soliciten el producto en un momento determinado con las características mencionadas. Así mismo, *Señal 3* y *Señal ACN* no solo cuentan con una versión web, sino también con una de escritorio que se encuentra desarrollada completamente con herramientas privativas. Por tanto los sistemas antes mencionados no satisfacen en su totalidad las necesidades actuales de los clientes, es por ello que se hace necesario definir cuáles son las tecnologías a utilizar para la creación de una aplicación de escritorio que permita la transmisión de noticias en la Plataforma de Televisión Informativa.

1.4 Análisis de otras soluciones existentes.

Es muy común encontrarse disímiles entornos corporativos como grandes empresas, corporaciones y ministerios que en sus edificios centrales tienen instaladas una red interna de televisión por cable. Esta televisión se utiliza para la transmisión de informaciones de corte noticioso en el ámbito internacional, nacional y ministerial. En el mundo existen algunas soluciones como soporte de información rápida que se valen de la televisión y tienen como base sistemas automáticos, los cuales están orientados a brindar informaciones las 24 horas a través de un *Sistema de Teletexto* adaptado. Otras soluciones son los canales informativos en el ámbito nacional, *Señal 3* de la Universidad de las Ciencias Informáticas, *Señal ACN* de la Agencia Cubana de Noticias y la Plataforma de Televisión Informativa (*PRIMICIA*) como una conceptualización de todos estos.

1.4.1 Sistemas de Teletexto en el ámbito internacional.

La mayor parte de los *Sistemas de teletextos* existentes en el mundo se concentran en Europa en países como España, Inglaterra, Suiza, Holanda, Italia, etc. Ejemplo de estos son “Fingertext” y “Cyclone”, dos plataformas de nacionalidad española e inglesa respectivamente. A continuación se realiza el análisis de Fingertext teniendo en cuenta algunas de las tecnologías utilizadas para el desarrollo del mismo.

Fingertext, es un sistema modular de teletexto y subtítulos por teletexto. Fue desarrollado por la joven empresa Anglatècnic S.L. fundada en Enero de 1999 por Enric Torres i Feixas. Anglatècnic está formada por un equipo de Ingenieros de Telecomunicaciones e Ingenieros Informáticos que se dedican a la realización de proyectos en los sectores de *Broadcast Profesional* (Emisoras de Televisión) y Tecnologías de la Información (Nuevas Tecnologías)(3).

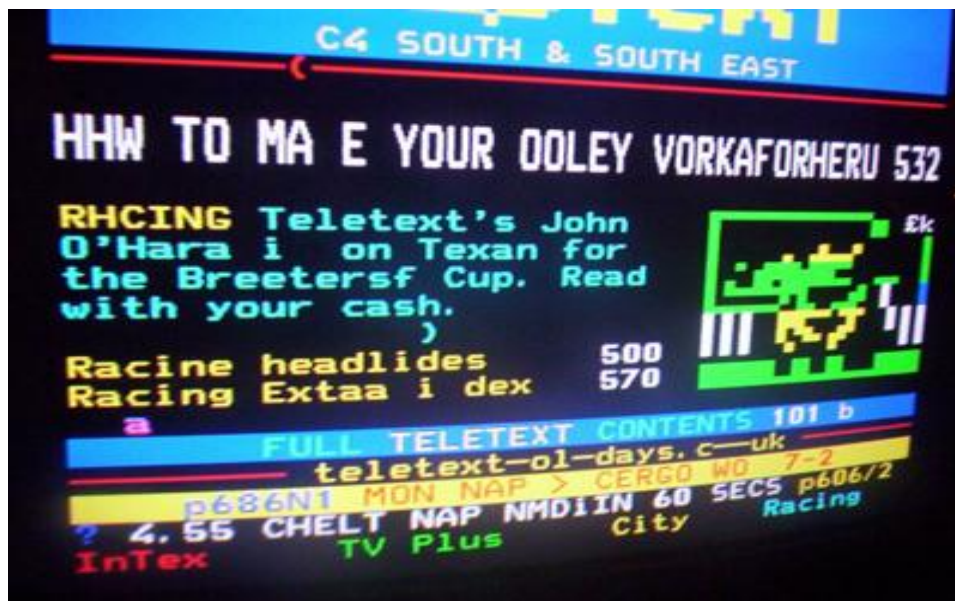


Figura 1: Sistema Teletexto Fingertext

Algunas de las tecnologías empleadas en su construcción:

- **Javascript:** Son scripts incluidos en el código HTML de las páginas que interpreta el navegador. Muy adecuadas para la validación de datos de formularios o para comunicarse con Applets de Java. Anglatècnic lo utiliza en Fingertext para validar los datos de los formularios.
- **Applets Java:** Son aplicaciones que ejecuta el navegador programadas en lenguaje Java mediante estos se puede acceder a ficheros, bases de datos, crear aplicaciones gráficas con animaciones etc.
Anglatècnic aprovecha su potencialidad para crear en Fingertext banners interactivos, gráficos dinámicos, programas dentro de páginas web, etc.
- **Flash:** La tecnología Flash permite crear páginas web interactivas, introducir animaciones en ellas y todo esto permite al Flash crear aplicaciones sofisticadas pero con interfaces agradables que cautiven fácilmente al usuario. En Fingertext se utiliza esta tecnología para presentar ejercicios en cursos interactivos a través de Internet.
- **PHP:** Permiten independizar el diseño de la página de la programación. Está especialmente indicado para trabajar conjuntamente con una base de datos que contiene los datos que se quieren presentar.
Anglatècnic se apoya en este lenguaje para realizar en Fingertext las oficinas virtuales, redacciones virtuales, etc.

- **Librerías PHP:** En el lenguaje de programación PHP se pueden utilizar librerías que facilitan tareas de desarrollo. Anglatènic emplea estas colecciones para generar dinámicamente imágenes de páginas del teletexto de las emisoras de televisión, mapeadas adecuadamente a fin que el usuario pueda saltar a las otras páginas indicadas.
- **XML y XSL:** Con XML se puede estructurar fácilmente la información que se quiere presentar. Mediante páginas XSL se puede presentar la información almacenada en formato XML en el formato que más se desee, html, texto, wml (WAP), etc. De esta forma se consigue total independencia entre el diseño de las páginas y los datos que se publican. El grupo de expertos de Anglatènic utiliza esta tecnología para publicar en Fingertext datos meteorológicos.

En un sistema de teletexto, la emisión del propio teletexto se realiza junto con la señal televisiva haciendo uso de las llamadas líneas del intervalo de cancelación de cuadro de la señal de video, es decir, las líneas no utilizadas para la señal de imagen de televisión. Esta característica limita en gran parte la información que se puede emitir, admitiendo sólo textos y mínimamente gráficos. Al poseer este inconveniente no son capaces de satisfacer los requerimientos planteados para el subsistema a desarrollar debido a que se necesita un producto que resulte atractivo para los televidentes y que permita la utilización de imágenes y materiales en formato de video.

Estos *sistemas de teletexto* implementados en Europa ofrecen enormes ventajas desde el punto de vista de la televisión digital, organización por secciones y actualización automática mediante un software que facilita este trabajo. Todas estas características se ajustan a las exigencias de la institución cliente, sin embargo, la posibilidad que le brindan al usuario de decidir mediante su mando a distancia qué noticias desea ver dentro del sinnúmero de informaciones que se ofrecen hacen que no se ajuste para darle solución a la problemática planteada, debido que en ella se requiere que las noticias sean transmitidas sin la necesidad de la interactividad con el usuario.

1.4.2 **Ámbito nacional.**

Cuba cuenta con antecedentes a la solución que se propone, dentro de los que se pueden mencionar *Señal 3* de la Universidad de las Ciencias Informáticas, *Señal ACN* de la Agencia Cubana de Noticias, la Plataforma de Televisión Informativa (*PRIMICIA*) como una conceptualización de ellos y el sistema *El Viajero* de la terminal de ómnibus ASTRO. Estos sistemas tienen como elemento común la posibilidad de brindar información a los usuarios las 24 horas del día, utilizando el empleo de imágenes y videos en la mayoría de los casos.

Señal 3 es un sistema que se encarga de difundir todas las actividades de interés de la Universidad de Ciencias Informáticas. Se utilizó como base para la construcción del sistema *Señal ACN* de la Agencia Cubana de Noticias. El principal objetivo del sistema antes mencionado es posibilitar el acceso a la información nacional a los cooperantes cubanos que cumplen misión en el extranjero y a los cubanos que residen en lugares inaccesibles de la geografía cubana y que reciben la señal de TV satelital cubana.

Señal ACN y *Señal 3* están desarrollados completamente utilizando software propietario (Windows, ASP, SQL, Macromedia Director). Estas tecnologías empleadas no son compatibles con los requisitos planteados por la parte venezolana que necesita un sistema que permita la transmisión de información y que sea desarrollado sobre tecnologías libres.

La Plataforma de Televisión Informativa (*PRIMICIA*) provee un canal de televisión que es soportado y transmitido utilizando computadoras. La solución ha sido pensada para que sea adaptable a distintos entornos. Todo el desarrollo del sistema se enfocó hacia la obtención de funcionalidades genéricas fácilmente escalables, que no dependan de un entorno dado y no atadas a un diseño gráfico específico. Un elemento adicional es que en el estudio de requerimientos de los clientes se puede acordar la personalización o eliminación de alguna funcionalidad. El desarrollo del sistema se realizó completamente utilizando software libre, siendo el resultado del empleo e integración de las tecnologías, herramientas y lenguajes de programación utilizados.

El Viajero, es un sistema que brinda información a todas las personas que se transportan de un lugar a otro en el país. El mismo está desarrollado con las siguientes tecnologías: C# como lenguaje de programación, UML como Lenguaje Unificado de Modelado, RUP como metodología de desarrollo de software y como herramienta de modelado Rational Rose, MySql como Sistema Gestor de Base de Datos y como Entorno Integrado de desarrollo (IDE) Microsoft Visual Studio 2005. Para proveer información al cliente "El Viajero" solamente hace empleo de pantallas que pueden contener texto e imágenes.

Un logro en el plantel universitario es la utilización de la Web y la red de televisión para la transmisión de radio. Lo desarrollado hasta el momento solo incluye la utilización de ciertas bondades tecnológicas, las cuales no posibilitan desarrollar funcionalidades que permitan la automatización de los procesos de transmisión.

Estos sistemas tienen características que se aproximan a lo deseado por el cliente como son: transmisión de texto a través de la televisión, la utilización de recursos multimedia y la organización por secciones

temáticas, el empleo de distintos tipos de pantallas para las noticias y un fondo musical. Sin embargo, ninguno de estos sistemas proporciona la solución al problema planteado en este trabajo.

1.5 Valoración del autor sobre los criterios emitidos por los especialistas.

Haciendo un análisis valorativo de las tecnologías de escritorio basado en el criterio de algunos expertos, es posible arribar a las siguientes conclusiones:

En el mundo la mayoría de los sistemas de teletexto que permiten la transmisión están implementados con tecnologías web, ya que pueden ser accedidos desde cualquier parte del mundo y brindan dado sus características, ciertas bondades tanto al usuario como al desarrollador del sistema. Algunos especialistas como Martín Pérez Mariñán Ingeniero Técnico en Informática de Sistemas y con certificaciones como Sun Certified Java Programmer, Sun Certified Java Developer y Sun Certified Business Component Developer, afirman hoy día que la tendencia a desarrollar este tipo de sistemas con tecnología web va adquiriendo un gran auge, sin embargo, afirman además que las tecnologías de escritorio brindan disímiles potencialidades y comodidades que no deben ser ignoradas a pesar del avance de las tecnologías web.

Si usted realiza una búsqueda detallada en la red de redes, específicamente en foros y blogs personales, encontrará que la opinión de muchos especialistas, conocedores del tema y otras personas en el mundo, coinciden en su mayoría en desarrollar estos sistemas con tecnologías web. El autor del trabajo investigativo considera que el uso de una u otra tecnología no se debe específicamente a cuál de ellas es mejor o no para implementar estos sistemas, sino a las mejoras o beneficios que se desee brinde el sistema, pues ambas tecnologías brindan disímiles bondades en cuanto a rapidez, rendimiento, etc.

Se decide realizar una aplicación de escritorio para la Plataforma de Televisión Informativa ya que se quiere mejorar por parte de los desarrolladores del sistema elementos como animación y efectos pues a través de la web este proceso se realiza más lento; así como también un mayor rendimiento al no tener que depender de un navegador para la ejecución de la aplicación.

1.6 Conclusiones Parciales.

En este capítulo fueron abordados los elementos teóricos que sustentan el problema científico y los objetivos del presente trabajo. Se realizó un estudio crítico y valorativo, sentando las bases teóricas que apoyarán la propuesta de solución. Con la descripción de los procesos de transmisión en canales de

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Televisión se logra dar una referencia de conceptos básicos asociados a la solución. Se analizaron otras soluciones existentes, concluyendo que no resuelven la problemática planteada.

Con el objetivo de estudio en este capítulo, se logra tener un mejor entendimiento de la situación que llevó a la realización del presente trabajo, siendo muy útil para la comprensión de los capítulos siguientes.

CAPÍTULO 2: TENDENCIAS Y TECNOLOGÍAS ACTUALES.

2.1 Introducción.

En este capítulo se realiza un análisis detallado de los principales conceptos y tecnologías que se propondrán para el desarrollo de la plataforma. Se abordan temas relacionados con el uso de tecnologías para desarrollar una aplicación de escritorio, soporte para el uso de XML, así como de medias. Se hace un estudio de los lenguajes de programación más apropiados para la implementación del subsistema de transmisión, el framework de desarrollo, el Sistema Gestor de Bases de Datos (SGBD) que se empleará y otras herramientas. Es importante destacar que para la selección estas tecnologías se realizó además una comparación entre las mismas teniendo en cuenta tres variables específicas definidas previamente: animación, efectos y rendimiento. Basado en estas tres variables, que son los que se necesitan potenciar en el subsistema de transmisión de noticias, se procede finalmente a la selección de las mismas.

2.2 Lenguajes de programación.

Un lenguaje de programación es un idioma artificial diseñado para expresar operaciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden emplearse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión o como modo de comunicación humana (4). El mismo está formado de un conjunto de símbolos, reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

2.2.1 C ++

El nombre C++ fue propuesto por Rick Mascitti en el año 1983, cuando el lenguaje fue utilizado por primera vez fuera de un laboratorio científico. Antes se había usado el nombre "C con clases" y se refiere a que C++ significa "incremento de C" y es una extensión de C.

C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue extender el exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, C++ es un lenguaje híbrido (5). Se trata simplemente del sucesor de un lenguaje de programación hecho por programadores (de alto nivel) para programadores, lo que se traduce en un diseño pragmático al que se le han ido añadiendo todos los elementos que la práctica aconsejaba como necesarios, con independencia de su belleza o purismo conceptual. En el diseño de la *Librería Estándar C++* se ha usado

ampliamente esta dualidad (ser mezcla de un lenguaje tradicional con elementos de POO), lo que ha permitido un modelo muy avanzado de programación extraordinariamente flexible (programación genérica).

Posteriormente se añadieron facilidades de programación genérica, que se sumó a los otros dos paradigmas que ya estaban admitidos (programación estructurada y la programación orientada a objetos). Por esto se suele decir que C++ es un lenguaje de programación multiparadigma (6).

C++, como su ancestro, sigue muy ligado al hardware subyacente, manteniendo una considerable potencia para programación a bajo nivel y en el manejo de la memoria del sistema, pero se le han añadido elementos que le permiten también un estilo de programación con alto nivel de abstracción.

C++ posee un sistema de detección de errores mucho más robusto que el de C, por lo que algunos errores de este serán rápidamente detectados.

En la actualidad, C++ es un lenguaje versátil, potente y general. Su éxito entre los programadores profesionales le ha llevado a ocupar el primer puesto como herramienta de desarrollo de aplicaciones. El C++ mantiene las ventajas del C en cuanto a riqueza de operadores y expresiones, flexibilidad, concisión y eficiencia. Además, ha eliminado algunas de las dificultades y limitaciones del C original.

El C++ es a la vez un lenguaje procedural (orientado a algoritmos) y orientado a objetos. Como lenguaje *orientado a objetos* se basa en una filosofía completamente diferente, que exige del programador un completo cambio de mentalidad.

Este lenguaje de programación ha experimentado un extraordinario éxito desde su creación. Muchos sistemas operativos, compiladores e intérpretes han sido escritos en C++ (el propio Windows y Java). Una de las razones de su éxito es ser un lenguaje de propósito general que se adapta a múltiples situaciones. Ha servido de justificación para el diseño de otros lenguajes que intentan eliminar sus inconvenientes y al mismo tiempo mantener sus virtudes (Ej: C# y Java), y una de sus últimas incorporaciones, las plantillas (ha sido origen de un nuevo paradigma de programación: metaprogramación).

2.2.2 Java

El lenguaje de programación Java, fue diseñado por la compañía Sun Microsystems Inc¹, con el propósito de crear un lenguaje que pudiera funcionar en redes computacionales heterogéneas (redes de

¹ Es una empresa informática de Silicon Valley, fabricante de semiconductores y software.

computadoras formadas por más de un tipo de computadora, ya sean PC, estaciones de trabajo, etc.) y que fuera independiente de la plataforma en la que se vaya a ejecutar. Esto significa que un programa escrito en Java puede ejecutarse en cualquier máquina o plataforma. El lenguaje fue diseñado con las siguientes características:

- Simple: elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos. La filosofía de programación orientada a objetos es diferente a la programación convencional.
- Familiar: como la mayoría de los programadores están acostumbrados a programar en C o en C++, la sintaxis de Java es muy similar al de estos.
- Robusto: el sistema de Java maneja la memoria de la computadora. No te tienes que preocupar por apuntadores, memoria que no se esté utilizando, etc. Java realiza todo esto sin necesidad de que uno se lo indique.
- Seguro: El sistema de Java tiene ciertas políticas que evitan se puedan codificar virus con este lenguaje. Existen muchas restricciones, especialmente para los applets, que limitan lo que se puede y no puede hacer con los recursos críticos de una computadora.
- Portable: Como el código compilado de Java (conocido como byte code) es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el intérprete de Java.
- Independiente a la arquitectura: Al compilar un programa en Java, el código resultante es un tipo de código binario conocido como byte code. Este código es interpretado por diferentes computadoras de igual manera, solamente hay que implementar un intérprete para cada plataforma. De esa manera Java logra ser un lenguaje que no depende de una arquitectura computacional definida.
- Multithreaded. Un lenguaje que soporta múltiples threads es un lenguaje que puede ejecutar diferentes líneas de código al mismo tiempo.
- Interpretado: Java corre en máquina virtual, por lo tanto es interpretado.

- Dinámico: Java no requiere que compiles todas las clases de un programa para que este funcione. Si realizas una modificación a una clase Java se encarga de realizar un Dynamic Bynding o un Dynamic Loading para encontrar las clases.

Java puede funcionar como una aplicación sola o como un "applet", que es un pequeño programa hecho en Java. Los applets de Java se pueden "pegar" a una página de Web (HTML) y con esto se puede tener un programa que cualquier persona que tenga un browser compatible podrá usar.

Entre noviembre de 2006 y mayo de 2007, Sun Microsystems liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de acuerdo con las especificaciones del Java Community Process, de tal forma que prácticamente todo el Java de Sun es ahora software libre (aunque la biblioteca de clases de Sun que se requiere para ejecutar los programas Java aún no lo es).

2.2.3 C Sharp

C# o C Sharp (en inglés C almohadilla) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA² e ISO³.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma.NET, el cual es similar al de Java aunque incluye mejoras derivadas de otros lenguajes. C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Ya existe un compilador implementado que provee el marco de DotGNU - Mono que genera programas para distintas plataformas como Win32, UNIX y Linux. Este lenguaje presenta las siguientes características:

- Sencillez: elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET. Por ejemplo:
 - El código escrito en C# es autocontenido, lo que significa que no necesita de ficheros adicionales (ficheros de cabecera o ficheros IDL) al propio fichero fuente.

² Es una organización internacional basada en membresías de estándares para la comunicación y la información

³ Es el organismo encargado de promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica.

- No se incluyen elementos poco útiles de lenguajes como C++, tales como macros, herencia múltiple y la necesidad de un operador diferente del punto (.) para acceder a miembros de espacios de nombres (::).
- Modernidad: incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular. Ejemplo de ello es la inclusión de una instrucción *foreach* que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico *string* para representar cadenas o la distinción de un tipo *bool* específico para representar valores lógicos.
- Orientado a objetos: como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos. Una diferencia de este enfoque orientado a objetos respecto al de otros lenguajes como C++, es que el de C# es más puro en tanto que no admiten ni funciones, ni variables globales. Todo el código y datos han de definirse dentro de definiciones de tipos de datos, lo que reduce problemas por conflictos de nombres y facilita la legibilidad del código. Soporta todas las características propias del paradigma de programación orientada a objetos: encapsulación, herencia y polimorfismo.

En resumen, C# es un lenguaje de programación que toma las mejores características de lenguajes preexistentes como Visual Basic, Java o C++ y las combina en uno solo. El hecho de ser relativamente reciente no implica que sea inmaduro, pues Microsoft ha escrito la mayor parte de la BCL⁴ usándolo, por lo que su compilador es el más depurado y optimizado de los incluidos en el .NET Framework SDK.

2.2.4 Fundamentación del Lenguaje de Programación escogido.

Se escoge C++ como lenguaje de programación ya que provee disímiles potencialidades para la implementación del subsistema de transmisión de noticias PRIMICIA. Una de las características que supera a Java y C# es la plantilla metaprograming. Al comparar código C++ en tiempo de ejecución, éste supera a su equivalente en C# o Java. C++ tiene un modelo de memoria diferente al de Java / C #, por ello tiene diversas ventajas. No importa la optimización de JIT (Just in Time Compilation), nada va a ir más rápido que el acceso de puntero directo a memoria. El nuevo estándar de C++ 0x impondrá un modelo de memoria simple de los compiladores de C++, lo que simplificará el multiprocesamiento efectivo / paralelo

⁴ Librería de clases base (BCL) es una librería estándar disponible para todos los lenguajes usando el Framework.NET

de código en C++. Ello permite la optimización más fácil y más segura para los compiladores. El modelo de memoria, los tipos atómicos y el almacenamiento local de hilos brindan las garantías básicas para soportar una buena librería de hilos. C++ es considerado por expertos como el lenguaje más potente para el trabajo con el hardware de la máquina.

El RAII (adquisición de recursos es de inicialización) en C++, hace el procesamiento mucho más fácil que en C# o incluso en Java. C++ potencia el manejo directo de interrupciones o la manipulación directa de hardware, es considerado la solución ideal en cuanto a la obtención de un mejor rendimiento del hardware. C++ es un lenguaje de plataforma de bajo nivel, con el que los desarrolladores alcanzan acceso a recursos primarios como sólo este lenguaje puede ofrecer. C++ potencia un mejor rendimiento de la caché de las aplicaciones permitiendo que estas sean más rápidas que las desarrolladas en java, principalmente debido a la huella más ligera de memoria. Si se tienen datos contiguos en la memoria, con el empleo correcto de punteros y el manejo directo de memoria se logra un rendimiento excelente que no tiene comparación con los de ningún otro lenguaje.

Permite el tratamiento intensivo de gráficos superior al de Java y C# ya que posee una perfecta integración con OpenGL (Open Graphic Library) al estar ésta programada en C, con ello se potencia la aceleración de hardware en 3D rápida y completa. Algunos de estos efectos son niebla, transparencias, reflexiones, texturas 3D, sombras volumétricas, motion blur (efecto de movimiento) y otros.

2.3 Lenguaje Unificado de Modelado (UML).

El Lenguaje Unificado de Modelado (UML) es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Captura decisiones y conocimiento sobre los sistemas que se deben construir. UML incluye conceptos semánticos, notación, y principios generales. Tiene partes estáticas, dinámicas, de entorno y organizativas.

UML posee elementos gráficos para soportar la captura de requisitos, el análisis, el diseño, la implementación, y las pruebas. Sin embargo UML es una notación y no un proceso o método, es decir, es una herramienta útil para representar los modelos del sistema en desarrollo, pero no ofrece ningún tipo de guía o criterios acerca de cómo obtener esos modelos. Es decir UML indica *qué* es lo que supuestamente hará el sistema, pero no *cómo* lo hará.

Los diagramas de UML se pueden clasificar de la siguiente manera:

Diagramas de estructura estática:

- Diagrama de clases: Conjunto de clases, interfaces, y sus relaciones, conectados como un grafo entre sí y con sus contenidos.
- Diagrama de objetos: Conjunto de objetos y sus relaciones.

Diagramas de comportamiento:

- Diagramas de interacción (secuencia y colaboración): Objetos y sus relaciones, incluyendo los mensajes que pueden ser enviados entre ellos.
- Diagrama de estados: Muestra una máquina de estado que consta de estados, transiciones, eventos y actividades.
- Diagrama de actividad: Son en esencia diagramas de flujo, con algunos elementos adicionales que les permiten expresar conceptos como la concurrencia y la división del trabajo.

Diagrama de casos de uso:

- Sirve para describir las interacciones del sistema con su entorno, identificando los actores, que representan los diferentes roles desempeñados por los usuarios del sistema, y los casos de uso, que corresponden a la funcionalidad que el sistema ofrece a sus usuarios, explicada desde el punto de vista de éstos.

Diagramas de implementación:

- Diagrama de componentes: Se utiliza, para describir la estructura física del código de la aplicación en términos de sus componentes (código fuente, binario o ejecutable) y sus dependencias
- Diagramas de despliegue: configuración de nodos de procesamiento en tiempo de ejecución y los componentes que residen en ellos.

2.4 Proceso Unificado de Desarrollo de Software (RUP).

La metodología que se empleará para guiar el proceso de desarrollo del software es RUP, llamada así por sus siglas en inglés Rational Unified Process.

El Proceso Unificado, es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de

software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto. El Proceso Unificado utiliza el Lenguaje Unificado de Modelado (Unified Modeling Language, UML), para preparar todos los esquemas de un sistema de software.

RUP es una metodología de desarrollo de software que se caracteriza por ser iterativo e incremental, centrado en la arquitectura y guiado por los casos de uso. Divide el proceso de desarrollo en ciclos, teniendo un producto al final de cada ciclo, cada ciclo se divide en fases que finalizan con un hito.

RUP sigue una estructura bidimensional dividiendo el proceso en fases y éstas en flujos de trabajo. Esto puede observarse en la siguiente figura donde el eje horizontal representa el tiempo, muestra aspectos dinámicos del proceso y se expresa como ciclos, fases, iteraciones e hitos; por otro lado el eje vertical contiene los aspectos estáticos del proceso, las actividades, artefactos, trabajadores y flujo de trabajo.

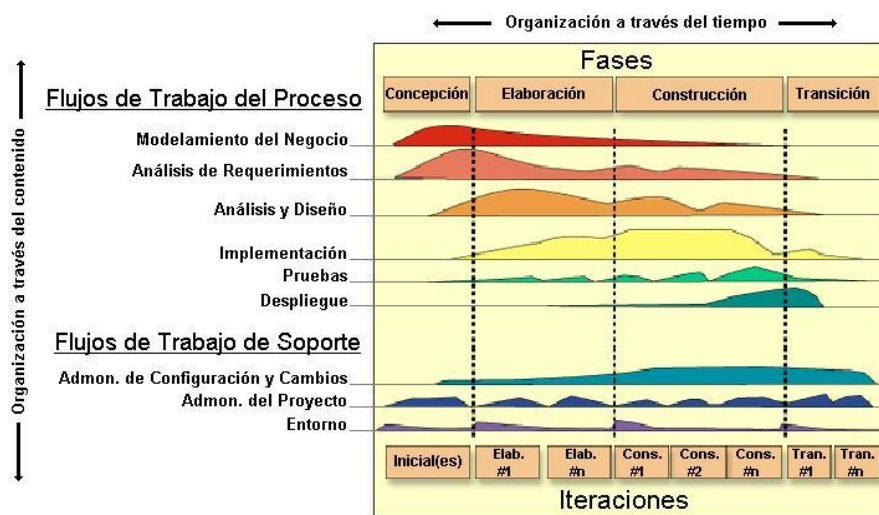


Figura 2: RUP en Dos Dimensiones

2.5 Framework.

Un framework simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además, un framework proporciona estructura al código fuente, forzando al desarrollador a crear código más legible y más fácil de mantener. Por último, un framework facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

2.5.1 Platinum C++ Framework.

Platinum (Pt) C++ Framework permite a los desarrolladores escribir aplicaciones de alto rendimiento para muchas plataformas con un solo código base. Ofrece una gran cantidad de características y es muy fácil de usar.

Es una API consistente para el modelado avanzado de objetos, multithreading, la programación del sistema, plugins, redes, acceso a bases de datos, gráficos, procesamiento XML y las interfaces gráficas de usuario que proporciona en todas las plataformas soportadas.

Las plataformas soportadas son Windows, Windows CE, Linux, QNX, y Unix. Te permite pasar el tiempo en una aplicación sin una tediosa programación de bajo nivel.

Un alto grado de rendimiento se consigue mediante el uso de características modernas de C++. Los programas se beneficiarán de las muchas optimizaciones incluidas dentro del framework.

PT ha sido diseñado desde cero para ser flexible y extensible. Proporciona soluciones sencillas a sus problemas sin ser intruso. Se integra bien con las bases de código ya existentes o de otros conjuntos de herramientas y frameworks. Los programas escritos con Pt tienden a ser significativamente más corto y más fácil de entender.

Publicado bajo la LGPL que permite escribir aplicaciones comerciales de código abierto y ser parte de la gran comunidad que desarrolla Pt y determina su futuro (7).

2.5.2 Qt 4.6.0

Qt es un framework para el desarrollo de aplicaciones multiplataforma. Dentro de sus características se encuentran:

- Compatibilidad multiplataforma con un sólo código fuente.
- Performance de C++.
- Disponibilidad del código fuente.
- Excelente documentación (Qt Assistant).

- Internacionalización de aplicaciones (Qt Linguist).
- Soporte para XML, conexión a bases de datos, SVG, OpenGL, programación para redes.
- Biblioteca de clases intuitiva.
- Herramientas de desarrollo integrado.
- Integración con Visual Studio.
- Utilice los controles de ActiveX en aplicaciones Qt.
- Integra Direct3D para el hardware de gráficos acelerados.
- Soporte para aplicaciones 64-bits en la plataforma Mac con la API Cocoa.
- Librerías para analizar la carga de las aplicaciones haciendo benchmark de las mismas. La librería Qt posee numerosos módulos que permite trabajar con bases de datos, gráficos, imágenes, sonidos, etc.

Las ventajas de usar este framework gráfico son disímiles:

- Es usado en disímiles tecnologías como la telefonía celular (Symbian S606)⁵, aplicaciones de tipo SCADA, dispositivos multipropósito, y otras.
- Es multiplataforma. El mismo código que se escribe en GNU/Linux, puede ser ejecutado fácilmente en Windows XP o en Mac OSX.
- Por las amplias características con que cuenta, se hace muy fácil construir aplicaciones para distintos usos.
- Cuenta con un excelente diseñador gráfico de aplicaciones llamado Qt Designer, el cual hace más fácil todavía el trabajo del programador.
- Tiene un sistema integrado de internacionalización llamado Qt Linguist lo que permite portar las aplicaciones al lenguaje que se desee.

⁵ Symbian S606 es uno de los nuevos productos de Nokia con el sistema operativo del mismo nombre.

Las aplicaciones basadas en Qt tienen una buena respuesta y un buen uso de la memoria. El desarrollo con Qt es muy apropiado para proyectos software de larga escala, tanto comerciales como de libre distribución. Por ejemplo, el escritorio KDE⁶, un conocido entorno de escritorio para Linux y FreeBSD, también el paquete de KOffice, una alternativa para el MS Office y así miles de programas tanto de libre distribución como comerciales.

El conjunto de herramientas Qt tiene un enfoque más eficiente para facilitar la tarea de gestión de memoria para sus programadores: cuando un objeto es suprimido, todos los objetos dependientes se eliminan automáticamente. El enfoque de Qt es no interferir con la libertad del programador para borrar manualmente cuando lo deseen.

La biblioteca Qt está concebida como multiplataforma, permitiendo escribir código que se compilará y ejecutará en distintas plataformas, incluyendo Unix, Linux, FreeBSD o incluso Windows. Se pueden ejecutar aplicaciones de Qt así como aplicaciones de KDE sin tener que estar corriendo el mismo KDE.

Funciona en todas las principales plataformas y tiene un amplio apoyo. El API de la biblioteca cuenta con métodos para acceder a bases de datos mediante SQL, así como uso de XML, gestión de hilos, soporte de red, una API multiplataforma unificada para la manipulación de archivos y una multitud de otros métodos para el manejo de ficheros, además de estructuras de datos tradicionales.

Qt es utilizada en KDE, un entorno de escritorio para sistemas como GNU/Linux o FreeBSD, entre otros. Utiliza el lenguaje de programación C++ de forma nativa, adicionalmente puede ser utilizado en otros lenguajes de programación a través de *bindings*.

Qt incluye un framework de animación que ayuda a construir grandes animaciones, interfaces gráficas de alto rendimiento sin la molestia de la gestión de estructuras complejas, temporizadores, etc. También permite crear animaciones personalizadas y funciones de interpolación. Qt tiene una clase que define las funciones de la funcionalidad compartida por todas las animaciones, la clase `QAbstractAnimation` que es la base de todas las animaciones. `QAbstractAnimation` proporciona las funciones virtuales puras utilizadas por las subclases para seguir el progreso de la animación: la duración (`duration()`) y `updateCurrentTime()`. Las vistas gráficas no están omitidas, se puede animar `QGraphicsWidgets` y `QGraphicsObject` las cuales heredan de `QGraphicsItem`. Cuenta con efectos que pueden ser utilizados para alterar la apariencia de los elementos de la interfaz de usuario, un par de efectos estándar tales como visión borrosa, colorear, proporcionar

⁶ Es un proyecto de software libre para la creación de un entorno de escritorio e infraestructura de desarrollo para diversos sistemas operativos como Linux, Mac OS X, Windows, etc.

sombras y es posible aplicar efectos personalizados (8).Las animaciones son controladas usando las curvas fáciles y pueden ser agrupadas. Esto permite que las animaciones sean de complejidad arbitraria.

El API es fácil de entender con funciones tales como `start ()`, `stop ()`, `pause ()`, y `currentTime ()`.Qt cuenta actualmente con un sistema de triple licencia: GPL v2/v3 para el desarrollo de software de código abierto y software libre, la licencia de pago QPL para el desarrollo de aplicaciones comerciales, y a partir de la versión 4.5 una licencia gratuita pensada para aplicaciones comerciales, LGPL.

2.5.3Fundamentación del framework escogido.

Los frameworks analizados anteriormente están completamente implementados en C++ y por tanto compilado a código nativo. Ello facilitaría una mejor implementación del subsistema de transmisión de noticias, ya que adquiere características semejantes como un fácil trabajo con la BCL, un buen rendimiento de la memoria de la PC.

Ambos frameworks cuentan con una licencia gratuita pensada para aplicaciones comerciales, LGPL. Platinum C++ no cuenta en estos momentos con una documentación tan exquisita como la de Qt, además de no tener una amplia difusión entre los usuarios. Qt incluye un framework de animación que permite realizar excelentes animaciones personalizadas y funciones de interpolación mediante el empleo de una de sus clases `QAbstractAnimation`. Cuenta con efectos que pueden ser utilizados para alterar la apariencia de los elementos de la interfaz de usuario y es posible aplicar efectos personalizados y con un excelente diseñador gráfico de aplicaciones llamado Qt Designer, el cual hace más fácil todavía el trabajo del programador, características que lo hacen ser un candidato perfecto para la implementación del subsistema de transmisión. Mientras que Platinum C++ a pesar de ser una API consistente para el modelado avanzado de gráficos no cuenta con clase alguna para realizar animaciones. Platinum C++ cuenta al igual que Qt con métodos para acceder a bases de datos, así como uso de XML, gestión de hilos y soporte de red; sin embargo, Qt potencia mejor estas funcionalidades ya que cuenta con varias clases que permiten un mejor desempeño en cada una de estas funcionalidades. Qt puede integrarse a Entornos de Desarrollo Integrado (IDE) como Eclipse, Visual Studio, Netbeans y Qt Creator, este último trabaja con las propias librerías de Qt. Por todas las características mencionadas anteriormente, se decide utilizar como framework de desarrollo Qt, específicamente en su versión 4.6.

2.6 Entorno de Desarrollo Integrado (IDE)

Un *entorno de desarrollo integrado* o *IDE*, es un programa compuesto por un conjunto de herramientas que proveen facilidades a los programadores para agilizar el proceso de desarrollo de software.

Un *IDE* es un entorno de programación que ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI), algunas veces contiene también un sistema para llevar a cabo el control de versiones. Muchos de los IDEs modernos llevan integrados además un navegador de clases, un inspector de objetos y diagramas de herencia de clases para ser usados en el desarrollo orientado a objetos. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Estos proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación, tales como, C++, C#, Java Object Pascal, Visual Basic, entre otros. En muchos casos se puede dedicar a un sólo lenguaje de programación, por ejemplo, Borland C++ o puede utilizarse para varios como es el caso de Eclipse.

2.6.1 Eclipse

En la web oficial de *Eclipse*, el mismo se define como “una plataforma (IDE), abierta para todo y para nada en particular”. Eclipse es un *plataforma* porque no es una aplicación acabada de por sí, pero está diseñada para ser extendida indefinidamente con herramientas sofisticadas. Es un *ambiente de desarrollo integrado (IDE)* porque provee herramientas para administrar áreas de trabajo (o workspaces en inglés), para construir, lanzar y depurar aplicaciones, para compartir artefactos con un equipo y versionar el código fuente. Eclipse es *abierto* porque su diseño le permite ser extendido fácilmente por terceras partes. Es apropiado para *todo*, ya que ha sido utilizado exitosamente para construir ambientes para un amplio rango de temas, como el desarrollo en lenguaje de programación Java, Servicios Web, certámenes de programación de juegos. Eclipse no es *para nada en particular*, pues no se enfoca en ningún dominio específico.

Eclipse es un Entorno Integrado de Desarrollo, para todo tipo de aplicaciones libres, inicialmente desarrollado por IBM, y actualmente gestionado por la Fundación Eclipse.

Herramienta para el programador desarrollada principalmente para el desarrollo de aplicaciones Java, facilitando al máximo la gestión de proyectos colaborativos mediante el control de versiones 'CVS', es posible también con subversión, exportar e importar proyectos.

Es posible añadir nuevas funcionalidades al editor, a través de nuevos módulos ('plugins'), para programar en otros lenguajes de programación además de Java como C/C++, PHP, Python, Ruby, Cobol y más (9).

Eclipse no solo es una plataforma que permite desarrollar todo tipo de lenguajes (Java, C++, PHP, Python), sino que viene equipada con toda la maquinaria pesada para poder construir sus propias herramientas y extender sus funcionalidades.

El proyecto Eclipse CDT es una herramienta de desarrollo para C/C++ que provee una IDE completamente funcional para la plataforma Eclipse. Dentro de las características que se incluyen están la creación y gestión de proyectos, estándares para la construcción de los mismos, fácil navegabilidad por el código, herramientas de conocimiento como tipos de jerarquía, incluye un navegador, editor de código con resaltado de sintaxis, autocompletado de código, plegado y exploración de hipervínculos, refactorización de código fuente y generación del mismo, herramientas de depuración visual (10).

CDT no viene con compilador, por lo tanto corre por cuenta del usuario descargarlo e instalarlo, y posteriormente configurarlo en Eclipse.

2.6.2 Qt Creator 1.3.0

Qt Creator es un excelente IDE multiplataforma creado por Trolltech para desarrollar aplicaciones en C++ de manera sencilla y rápida. Como su nombre lo indica, está basado en la librería Qt, la biblioteca multiplataforma de interfaces gráficas de usuario. Cuenta con las siguientes características principales:

- Editor avanzado para C++.
- Diseñador de formularios (GUI) integrado.
- Herramientas para la administración y construcción de proyectos.
- Completado automático.
- Depurador visual.

Qt Creator es una aplicación integral y un marco para el desarrollo de aplicaciones de interfaz de usuario de Windows que también se puede implementar en muchos otros escritorios y sistemas integrados de explotación sin reescribir el código fuente.

Este IDE ofrece varias funcionalidades únicas para que la creación de Qt resulte fácil y más placentera, y que soporte plataformas múltiples.

La codificación puede resultar uno de los factores que insumen más tiempo en las aplicaciones de desarrollo, pero el editor de códigos C++ de Qt la torna más fácil y más rápida. Posee revisor de código estático y suministra sugerencias de estilo a medida que se tipea, plegado de código y edición avanzada. Se puede obtener ayuda sensible al contexto, combinación de paréntesis y modos de selección, dos funciones muy útiles que permitirán terminar el trabajo más rápido (11).

Qt Creator ha sido diseñado para ofrecer al programador un entorno de desarrollo suave, integrado y poderoso.

La función revolucionaria es la versión incorporada de Qt Designer. Su inclusión significa que usted puede arrastrar los Qt widgets en el lienzo en blanco de su aplicación y cambiar el código fuente de inmediato (12). Ahora Qt Designer forma parte de Creator y ya no es necesario recurrir a qmake en la línea de comandos ni a un editor de código fuente.

Los sistemas operativos que soporta en forma oficial son:

- GNU/Linux 2.6.x, para versiones de 32 y 64 bits con Qt 4.x instalado. Además hay una versión para Linux con gcc 3.3.
- Mac OS X 10.4 o superior, requiriendo Qt 4.x
- Windows XP y Vista, requiriendo el compilador MinGW y Qt 4.4.3 para MinGW.

2.6.3 Netbeans

IDE NetBeans es un entorno de desarrollo integrado disponible para Windows, Mac, Linux y Solaris. El proyecto NetBeans consiste en un IDE de código abierto y una plataforma de aplicaciones que permiten a los desarrolladores crear rápidamente web, empresa, escritorio y aplicaciones móviles utilizando la plataforma Java, así como JavaFX, PHP, JavaScript y Ajax, Ruby y Ruby on Rails , Groovy y Grails, y C / C++.

- El proyecto NetBeans es compatible con una vibrante comunidad de desarrolladores y ofrece una amplia documentación y recursos de capacitación, así como una amplia selección de plug-ins de otros fabricantes terceros.

- NetBeans IDE 6.8 es el primero en ofrecer soporte completo para toda la especificación Java EE 6 con soporte mejorado para 2.0/Facelets JSF, Java Persistence 2.0, EJB 3.1 incluyendo el uso de EJBs en aplicaciones web, servicios web REST, y GlassFish v3. Netbeans 6.8 también se recomienda para desarrollar con la última JavaFX SDK 1.2.1 y PHP para crear aplicaciones web con el nuevo PHP 5.3 liberación o con el framework de Symfony (13).
- Incluye integración única del Proyecto de Kenai, un entorno de colaboración para acoger proyectos de código abierto, ahora viene con soporte completo de JIRA y mensajería instantánea e integrar mejor seguimiento de incidencias. Netbeans 6.8 tiene opciones para IDE Maven e integración de bases de datos, y ha mejorado el editor y la integración de herramientas de Ruby, Groovy y C / C++.

2.6.4 Fundamentación del IDE escogido.

Se escoge Qt Creator como Entorno Integrado de Desarrollo ya que es uno de los mejores IDEs recomendado para trabajar con el framework Qt, brinda apoyo a la refactorización de código fuente para C++. Qt Creator al estar desarrollado en C++ posee mejor rendimiento que Netbeans, Eclipse. Qt Creator a través de la biblioteca de clases Qt permite construir interfaces de usuario complejas de una forma visual y rápida, lo que deja más tiempo para centrarse en la parte más importante de todo programa. Esto se hace más complejo si se emplea otro Entorno Integrado de Desarrollo (Netbeans o Eclipse). Qt Creator permite utilizar los controles de ActiveX en aplicaciones Qt e integra Direct3D para el hardware de gráficos acelerados. Qt Creator se centra en proporcionar características que ayudan a los nuevos usuarios de Qt a desarrollar más rápido, y también aumentar la productividad de los desarrolladores con experiencia en Qt, limitante en los demás IDEs. Dicho IDE al estar perfectamente integrado con la biblioteca Qt escrita genuinamente en código C++ nativo, potencia el manejo de efectos, transparencias, sombras de forma fácil empleando las clases *QGraphicsItem*, *QGraphicsWidgets* y *QGraphicsObjects*. Además se potencia el manejo de videos y la animación de objetos a través de las clases *QObject*, *QAbstractAnimation*, *QAbstractVideoSurface*, etc. Todo ello se hace más complejo si se emplea Netbeans o Eclipse. Qt Creator integra completamente a Qt Designer. Esta herramienta permite el desarrollo de formularios y presentaciones gráficas para las aplicaciones. Permite la previsualización de la interfaz, soporte para widgets y un editor de propiedades poderoso. Permite acelerar el desarrollo de interfaces de alto rendimiento, a la vez que proporciona una forma fácil de diseñar interfaces gráficas de usuario avanzadas generando el código fuente para las mismas, lo que permite al desarrollador ajustarlo a sus necesidades.

2.7 Sistema Gestor de Bases de Datos.

Los Sistemas Gestores de Bases de Datos son un tipo de software muy específico, dedicado a servir de interfaz entre las bases de datos y las aplicaciones que la utilizan. En la actualidad existe una gran variedad de SGBD, dentro de las principales funciones que satisfacen se encuentran la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad.

2.7.1 MySQL

MySQL es un *Sistema de Gestor de Bases de Datos Relacional* (RDBMS) licenciado bajo la GPL de la GNU⁷. Se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y de distribuirlos para cubrir las necesidades de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos. MySQL es la base de datos de código abierto más popular del mundo y compite con sistemas RDBMS propietarios conocidos, como Oracle, SQL Server y DB2 (14).

MySQL incluye todos los elementos necesarios para instalar el programa, preparar diferentes niveles de acceso de usuario, administrar el sistema y proteger los datos. Puede desarrollar sus propias aplicaciones de base de datos en la mayor parte de los lenguajes de programación utilizados en la actualidad y ejecutarlos en casi todos los sistemas operativos. MySQL utiliza el lenguaje de consulta estructurado (SQL) (15). Este lenguaje permite crear bases de datos, así como agregar, manipular y recuperar datos en función de criterios específicos. Dentro de sus principales características se encuentran:

- Es un servidor de base de datos muy rápido, robusto, multitarea y multiusuario.
- Tiene enfoque relacional y soporta AnsiSQL.
- Es gratis y se puede bajar de www.mysql.com.

⁷ Proyecto iniciado por Richard Stallman con el objetivo de crear un sistema operativo completamente libre.

- Su principal objetivo de diseño fue la velocidad, además de consumir muy pocos recursos, tanto de CPU como de memoria.
- Tiene excelentes utilidades de administración como backup y recuperación de errores.
- Tiene una excelente integración con PHP.
- No hay límites en el tamaño de los registros, además de tener un control de acceso de los usuarios bastante amplio.
- Trabaja en diferentes plataformas además de soportar múltiples idiomas.
- Completo y optimizado uso del SQL, además de contener un MyODBC.
- Usa Listas de Control de Acceso para todas las conexiones, consultas y otras operaciones.

2.7.2 Oracle

Oracle es un sistema de administración de base de datos (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), desarrollado por Oracle Corporation. Es básicamente una herramienta cliente/servidor para la gestión de bases de datos principalmente considerándose como uno de los sistemas de bases de datos más completos (16).

Como principales características destaca su seguridad al garantizar la autenticidad apropiada de los usuarios y la privacidad e integridad de los datos. Posee una alta disponibilidad al soportar efectivamente ambientes mixtos de carga de trabajo, caracterizados por actividades simultáneas de consulta y actualización, escalabilidad para soportar un gran número de usuarios y cargas de trabajo de alto volumen de transacciones.

También posee lectura de multiversión sin bloqueo, siempre proporciona a los usuarios resultados consistentes, al mismo tiempo que no penaliza al rendimiento en la actividad concurrente de actualización, además actualmente es multiplataforma.

Esta herramienta está orientada al acceso remoto y redes (Internet). En la actualidad Oracle se puede implementar en diferentes plataformas: Familia de Microsoft, Unix, Linux, Vms, etc. Es el mayor y más usado Sistema de Gestión de Bases de Datos Relacionales (*RDBMS*, por sus siglas en inglés) en el

mundo, además es soportado en computadoras personales (*PC, por sus siglas en inglés*), microcomputadoras, computadoras con procesamiento paralelo masivo (17). Soporta unos 17 idiomas, funciona automáticamente en más de 80 arquitecturas de hardware y software distintos sin tener la necesidad de cambiar una sola línea de código. Esto es porque más del 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos. Oracle es un sistema comercial que aporta un SGBD que ofrece las particularidades básicas para trabajar en entornos multi-usuario. Las arquitecturas en las que se asienta pueden ser: Intel, Alpha, Sparc, Risc, a nivel de procesadores.

Oracle es perfectamente configurable en entornos "OLTP"(Procesamiento de Transacciones en línea), paralelos, Clúster, e incluso resulta una genial solución a nivel de Datawarehouse y CRM (Administración de Relación con los clientes). La plataforma para la que en realidad está desarrollada Oracle y pensada desde un principio, es Unix , ya que es un sistema que soporta la mayoría de la carga de los sistemas a nivel mundial, así como es un sistema abierto y configurable. Para la gestión, Oracle y Unix se complementan de una manera casi inalcanzable para otros sistemas, ya que para su administración, Oracle se amolda y apoya en la potencia de Unix, a la vez que aprovecha al 100% su versatilidad y configurabilidad.

2.7.3 PostgreSQL 8.3

Es un Sistema Gestor de Base de Datos objeto-relacional. Es ampliamente considerado como una de las alternativas de sistema de bases de datos de código abierto. Soporta gran parte del SQL estándar y muchas funcionalidades modernas como son: consultas complejas, llaves foráneas, disparadores (*triggers*), vistas, integridad transaccional, así como el control de versionado concurrente (MVCC) que es una estrategia de almacenamiento que permite trabajar con grandes volúmenes de datos (18).

Cumple completamente con las características ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) para realizar transacciones seguras. Además, *PostgreSQL* puede ser personalizado por el usuario en muchas formas, según sus necesidades, por ejemplo: adicionando un nuevo tipo de datos, operadores, funciones, y lenguajes procedurales.

Utiliza licencia BSD, por lo que *PostgreSQL* puede ser utilizado, modificado y distribuido por cualquier usuario gratuitamente, para cualquier propósito ya sea con fines privados, comerciales o académicos. Dentro de sus principales características se encuentran:

- Alta concurrencia: mediante un sistema denominado MVCC (Acceso concurrente multiversión) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit.
- Amplia variedad de tipos nativos: provee nativamente soporte para números de precisión arbitraria, texto de largo ilimitado, figuras geométricas (con una variedad de funciones asociadas), direcciones IP (IPv4 e IPv6), bloques de direcciones estilo CIDR y direcciones MAC.
- Soporte SQL92/SQL99: implementa un subconjunto extendido de los estándares SQL92 y SQL99.
- Transacciones: permiten el paso entre dos estados consistentes manteniendo la integridad de los datos.
- Integridad referencial: soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.
- Bloqueos de tabla y filas: ofrece varios modos de bloqueo para controlar el acceso concurrente a los datos en tablas, algunos de estos modos de bloqueo los adquiere PostgreSQL automáticamente antes de la ejecución de una declaración, mientras que otros son proporcionados para ser usados por las aplicaciones.
- Múltiples tipos de datos predefinidos: como todos los manejadores de bases de datos, PostgreSQL implementa los tipos de datos definidos para el estándar SQL3 y aumenta algunos otros.
- Soporte de tipos y funciones de usuario: soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario.
- Conectividad: Posee conectividad TCP/IP, JDBC y ODBC (19).

2.7.4 Fundamentación del SGBD escogido.

Teniendo en cuenta que MySQL puede tener un futuro incierto con su valor de código abierto, al ser comprado por SUN Microsystems, la cual ya ha enseñado su visión futurista con el producto. Dicha corporación plantea que algunas partes de MySQL sólo serían ofrecidas a clientes corporativos que paguen una suscripción por el sistema de base de datos. El SGBD Oracle es muy potente y uno de los gestores con más prestigio a nivel mundial, posee dentro de sus principales características que es

multiplataforma, soporta un gran número de usuarios accediendo en tiempo real y cargas de trabajo de alto volumen de transacciones. Pero al ser propietario no se ajusta a las políticas trazadas por el país y la Universidad de las Ciencias Informáticas de migrar hacia software libre. Se decide utilizar PostgreSQL 8.3 en la aplicación que se quiere obtener, al ser uno de los SGBD más potente del momento y que en sus últimas versiones empieza a tener características semejantes a otros SGBD comerciales, además de ser ideal para sistemas serios en las que la consistencia de la base de datos sea fundamental. Es también, el SGBD sobre el cual está montada la base de datos del subsistema de transmisión de noticias en su versión web, lo cual permitirá un notable aprovechamiento del tiempo de desarrollo al no tener que migrar hacia otro SGBD.

2.8 Herramientas de modelado de software.

2.8.1 Rational Rose.

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 (20). Está basado en 4 vistas fundamentales:

- Vista de casos de uso.
- Vista lógica.
- Vista de componente.
- Vista de despliegue.

Dentro de sus principales características se encuentran:

- Mantiene la consistencia de los modelos del sistema software.
- Chequeo de la sintaxis UML.
- Generación de documentación automáticamente.
- Generación de código a partir de los modelos.

- Ingeniería Inversa (crear modelo a partir código).

Es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo a través del UML y trabajo de muchas metodologías utilizadas por los clientes. Rational Rose no es software libre.

2.8.2 Visual Paradigm.

Visual Paradigm para UML es una herramienta CASE para la modelación UML muy potente y de fácil uso. Te permite dibujar todo tipo de diagramas UML, revertir código fuente a modelos UML, generar código fuente desde los diagramas UML, y es además software libre.

Visual Paradigm for UML incluye los objetos más recientes de UML además de diagramas de casos de uso, diagramas de clase, diagramas de componentes, reversa instantánea para Java, C++, DotNet Exe/dll, XML, XML Schema, y Corba IDL, ofrece soporte para Rational Rose, integración con Microsoft Visio, además permite generar reportes y documentación en HTML/PDF (21).

Dentro de sus principales características se pueden mencionar:

- Visual Paradigm para UML, es una Herramienta CASE que soporta las últimas versiones de UML, así como el modelado de procesos de negocios.
- En adición al soporte de Modelado UML esta herramienta provee de un generador de mapeo de objetos-relacionales para los lenguajes de programación Java .NET y PHP.
- Para desarrolladores independientes existe una versión llamada Community Edition la que se caracteriza por ser de uso No Comercial (22).

Para maximizar la interoperabilidad de los productos de Visual Paradigm con otras aplicaciones, Visual introdujo la importación y exportación de modelos de proyecto desde o hasta un formato XML. Además, la herramienta es colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto. Permite además el control de versiones.

Visual Paradigm es un producto que facilita a las organizaciones diseñar visualmente y con diagramas, integrar y desplegar sus aplicaciones empresariales y sus respectivas bases de datos. La herramienta

ayuda al equipo de software a destacarse en el proceso de modelado-construcción-despliegue, maximizando y acelerando las contribuciones del equipo y los individuos.

2.8.3 Fundamentación de la herramienta de modelado escogida.

Las herramientas CASE anteriormente caracterizadas no solo son capaces de practicar la ingeniería inversa (crear modelo a partir código), sino que generan documentación automática como Web o PDF específicamente en el caso de Visual Paradigm, que incluye además los objetos más recientes de UML como reversa instantánea para Java, C++, DotNet Exe/dll, XML, XML Schema, y Corba IDL.

Ambos permiten dibujar todo tipo de diagramas UML. Visual Paradigm posibilita la importación y exportación de modelos de proyecto desde o hasta un formato XML, soporta múltiples usuarios trabajando sobre el mismo proyecto. Se seleccionó Visual Paradigm como herramienta CASE por las potencialidades mencionadas.

2.9 Herramienta control de versiones.

Se denomina *Control de Versiones*, a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo. Los sistemas de control de versiones facilitan la administración de las distintas versiones de cada producto desarrollado, así como las posibles especializaciones realizadas. Un sistema de control de versiones debe proporcionar:

Mecanismos de almacenaje para los elementos que debe gestionar (ej. archivos de texto, imágenes, documentación).

Posibilidad de realizar cambios sobre los elementos almacenados (ej. modificaciones parciales, añadir, borrar, renombrar o mover elementos).

Registro histórico de las acciones realizadas con cada elemento o conjunto de elementos (normalmente pudiendo volver o extraer un estado anterior del producto).

Todos los sistemas de control de versiones se basan en disponer de un repositorio, que es el conjunto de información gestionada por el sistema. Este repositorio contiene el historial de versiones de todos los elementos gestionados. Cada uno de los usuarios puede crearse una copia local duplicando el contenido del repositorio para permitir su uso. Es posible duplicar la última versión o cualquier versión almacenada en

el historial. Este proceso suele conocerse como check out o desproteger. Para modificar la copia local existen dos semánticas básicas:

Exclusivos: para poder realizar un cambio es necesario marcar en el repositorio el elemento que se desea modificar y el sistema se encargará de impedir que otro usuario pueda modificar dicho elemento.

Colaborativos: en el que cada usuario se descarga la copia y la modifica, el sistema automáticamente mezcla las diversas modificaciones. El principal problema es la posible aparición de conflictos que deban ser solucionados manualmente o las posibles inconsistencias que surjan al modificar el mismo fichero por varias personas no coordinadas. Además, esta semántica no es apropiada para ficheros binarios.

Aunque un sistema de control de versiones puede realizarse de forma manual, es muy aconsejable disponer de herramientas que faciliten esta gestión y un ejemplo de ello podría ser el Subversion.

2.9.1 Subversion.

Subversion (SVN) es un sistema de control de versiones libre y de código abierto. Uno de los reemplazos más populares de Concurrent Versions System (CVS), ofreciendo mejoras en muchas de las dificultades que presenta este último. Subversion maneja ficheros y directorios a través del tiempo. Hay un árbol de ficheros en un repositorio central, esto permite que se recuperen versiones antiguas de los datos o examinar el historial de cambios de los mismos. En este aspecto, mucha gente piensa en los sistemas de versiones como en una especie de “máquina del tiempo”. Subversion es un sistema general que puede ser usado para administrar cualquier conjunto de ficheros ya sea código fuente o de otro tipo. Dentro de sus principales características se pueden mencionar:

- Versionado de directorios: implementa un sistema de ficheros versionado “virtual” que sigue los cambios sobre árboles de directorios completos a través del tiempo.
- Verdadero historial de versiones: permite añadir, borrar, copiar y renombrar ficheros y directorios. Los ficheros nuevos que son añadidos comienzan con un historial nuevo, limpio y completamente suyo.
- Envíos atómicos: permite a los desarrolladores construir y enviar los cambios como fragmentos lógicos e impide que ocurran problemas cuando sólo una parte de los cambios enviados lo hace con éxito.

- Versionado de metadatos: cada fichero y directorio tiene un conjunto de propiedades —claves y sus valores —asociados a él. Se puede crear y almacenar cualquier par arbitrario de clave/valor que desee. Las propiedades son versionadas a través del tiempo, al igual que el contenido de los ficheros.
- Manipulación consistente de datos: Subversion expresa las diferencias del fichero usando un algoritmo de diferenciación binario, que funciona idénticamente con ficheros de texto (legibles para humanos) y ficheros binarios (ilegibles para humanos).
- Ramificación y etiquetado eficientes: se pueden crear ramas y etiquetas simplemente copiando el proyecto, usando un mecanismo similar al enlace duro. De este modo estas operaciones toman solamente una cantidad de tiempo pequeña y constante.
- Hackability: Subversion no tiene un equipaje histórico, está implementado como una colección de bibliotecas compartidas en C con APIs bien definidas. Esto hace a Subversion extremadamente fácil de mantener y reutilizable por otras aplicaciones y lenguajes.

2.9.2 CVS

El sistema de versiones concurrentes (CVS), también conocido como Concurrent Versioning System, es una aplicación informática que implementa un sistema de control de versiones: mantiene el registro de todo el trabajo y los cambios en los ficheros que forman un proyecto y permite que distintos desarrolladores (potencialmente situados a gran distancia) colaboren. CVS se ha hecho popular en el mundo del software libre. Sus desarrolladores difunden el sistema bajo la licencia GPL.

CVS utiliza una arquitectura cliente-servidor: un servidor guarda la(s) versión(es) actual(es) del proyecto y su historial. Con el sistema CVS el cliente y servidor pueden estar en la misma máquina. Este sistema tiene la tarea de mantener el registro de la historia de las versiones del programa de un proyecto solamente con desarrolladores locales. Los clientes pueden también comparar diferentes versiones de archivos, solicitar una historia completa de los cambios, o sacar una "foto" histórica del proyecto tal como se encontraba en una fecha determinada o en un número de revisión determinado (23).

Los archivos en el *repositorio* sobre la plataforma CVS no pueden ser renombrados, estos deben ser agregados con otro nombre y luego eliminados. El protocolo CVS no provee una manera de que los directorios puedan ser eliminados o renombrados, cada archivo en cada subdirectorío debe ser eliminado y

re-agregado con el nuevo nombre. Soporte limitado para archivos *Unicode* con nombres de archivo no *ASCII*.

2.9.3 Fundamentación de la herramienta de control de versiones escogida.

La creación de ramas y etiquetas en la herramienta Subversion es una operación más eficiente, tiene costo de complejidad constante ($O(1)$) y no lineal ($O(n)$) como en CVS. Con la herramienta Subversion se envían sólo las diferencias en ambas direcciones, en CVS siempre se envían al servidor archivos completos. Además, Subversion maneja eficientemente archivos binarios a diferencia de CVS que los trata internamente como si fueran de texto.

Subversion sigue la historia de los archivos y directorios a través de copias y renombrados, las modificaciones (incluyendo cambios a varios archivos) son atómicas, mientras que CVS no provee una manera de que los directorios puedan ser eliminados o renombrados, cada archivo en cada subdirectororio debe ser primeramente eliminado y luego re-agregado con el nuevo nombre. Ambas herramientas son de código abierto.

2.10 Descripción de otras tecnologías necesarias en la propuesta.

Para la implementación del subsistema de transmisión de noticias de la Plataforma de Televisión Informativa (PRIMICIA) se hace necesario describir otras tecnologías que apoyarán la propuesta de solución. En el desarrollo de este epígrafe, se analizará que soportes tiene el framework de desarrollo Qt para cada una de las tecnologías descritas y de esta forma lograr mayor comodidad y rapidez a la hora de su utilización.

2.10.1 Soporte para el uso de XML.

Al analizar todos los procesos que ocurren en el subsistema de transmisión de noticias se hace necesario optimizar el proceso de almacenamiento de la información. Las propiedades de los elementos configurables en el proceso de redacción de la noticia cambian frecuentemente durante la redacción de la misma. Es por ello que no es conveniente que sean almacenadas en la base de datos, ya que provocaría sucesivas actualizaciones y produciría una sobrecarga de la misma que atentaría contra su mantenimiento y con el rendimiento de la aplicación en general. Para almacenar la configuración de todos los elementos que conforman la noticia se estructurará en un formato XML (*Extensible Markup Language*), haciendo uso de la potencialidad que presenta el mismo mostrada a continuación.

- Independencia entre la Presentación y los Datos: es la separación entre el contenido (los *datos*) y la presentación (el *formato de los datos*). Por ejemplo, los datos de una reserva de avión se han de presentar en HTML para un cliente web o en PDF si se va a imprimir. Esta característica presente en los documentos XML, posibilita que una vez generado el documento XML que contiene todos los detalles de la forma en que se mostrará la noticia pueda ser presentada lo mismo en un ambiente web como en uno escritorio, todo depende del analizador de XML que se emplee.
- Comunicación entre aplicaciones: permite que exista una comunicación entre diferentes aplicaciones sin que el formato esté ligado a ninguna presentación (**Fig. 3**). En muchas ocasiones, comunidades de organizaciones o empresas de un cierto sector, definen un vocabulario XML común para la interacción entre ellos (documentos de pedidos, facturas, etc). En el caso de PRIMICIA servirá como medio de comunicación entre el subsistema de administración, donde se lleva a cabo la redacción de la noticia y la creación del documento XML y el subsistema de transmisión que es el encargado de leer el documento XML generado y mostrar la noticia.
- Configuración: para guardar los datos de configuración de las aplicaciones, como lo realiza Apache, Tomcat Enterprise Java Beans (EJB) o para guardar los datos de cualquier aplicación.

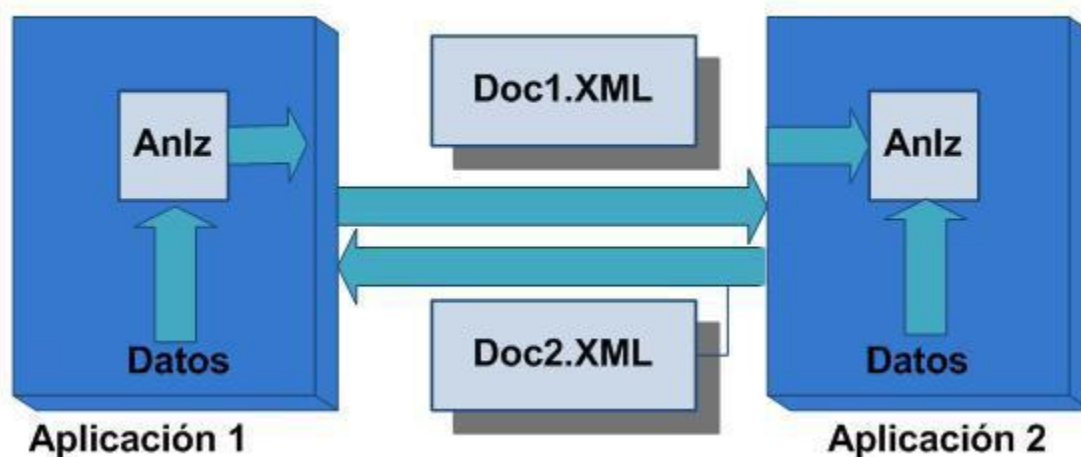


Figura 3: Comunicación entre Aplicaciones usando XML

Un aspecto importante cuando se trabaja con información representada en XML, es que abundan las herramientas para analizar, acceder, tratar texto XML y transformar los datos que contiene. Para ello se utilizan componentes que ofrecen una o las dos API estandarizadas: SAX y DOM llamados Analizadores o Parseres (**Fig. 4**). SAX es muy utilizado en el procesamiento de XML basado en eventos y tiene como

característica principal que lee el documento de principio a fin sin tener que cargarlo completamente en memoria. Por su parte DOM tiene como característica fundamental que carga todo el documento en memoria en una estructura de árbol, lo que facilita el acceso a cada uno de los elementos del documento (nodos del árbol), lo que requiere más memoria, pero hace que sea más fácil de manipular la estructura del documento.

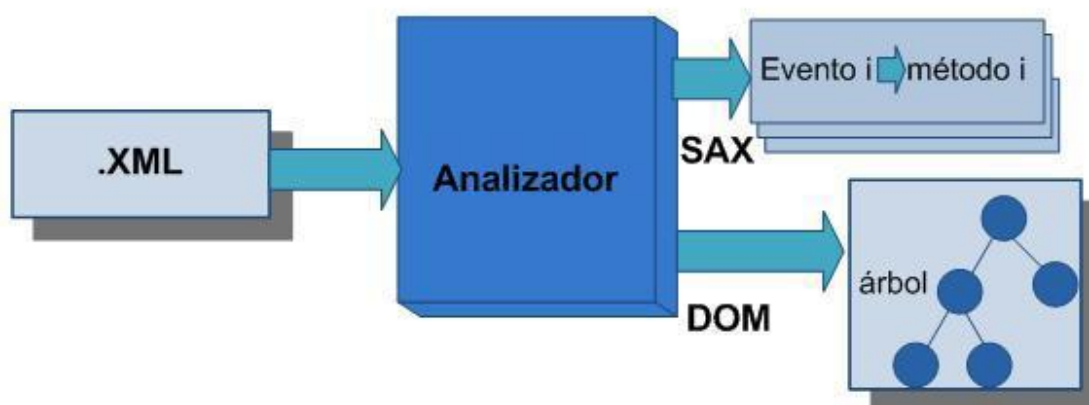


Figura 4: Esquema de dos analizadores

SAX es una interfaz estándar basada en eventos para los analizadores XML. La interfaz Qt sigue el diseño de la implementación de Java SAX2 (24).

La interfaz SAX2 es un mecanismo orientado a eventos para proporcionar al usuario información del documento. Un "evento" en este contexto, significa algo reportado por el analizador, por ejemplo, se ha detectado una etiqueta de inicio o una etiqueta de cierre.

Si bien este es un rápido y sencillo enfoque para leer documentos XML, la manipulación es difícil porque los datos no se almacenan, simplemente son manipulados y descartados. El módulo Qt XML proporciona una clase abstracta, *QXmlReader*, que define una interfaz para los potentes lectores SAX2. Qt incluye una lectura de implementación *QXmlSimpleReader*, que es fácil de adaptar a través de subclases.

La clase *QXmlInputSource* proporciona los datos de entrada para las subclases *QXmlReader*. Todas las subclases de *QXmlReader* leen la entrada del documento XML desde esta clase.

Otra de las API es DOM (Document Object Model), que proporciona una interfaz para acceder y modificar el contenido y la estructura de un archivo XML. Tiene una vista jerárquica del documento (una vista de árbol). Es un modelo de objeto del documento que está residente en memoria, es por ello que se hace mucho más fácil su manipulación. Todos los nodos en el árbol de documentos DOM son subclases de *QDomNode*. El documento en sí mismo es representado como un objeto *QDomDocument*.

La clase *QDomNode* es la clase base para todos los nodos en un árbol DOM. Muchas funciones en el DOM retornan un *QDomNode*. La clase *QDomDocument* representa un documento XML completo. Conceptualmente es la raíz del árbol de documento y provee el acceso primario de los datos del documento.

Otro aspecto importante a tener en cuenta cuando se va a desarrollar aplicaciones usando XML es la validación del mismo mediante XML-Schemas. Estos elementos le proporcionan al parser una gramática por la cual se va a regir a la hora de crear o validar los documentos XML. El *XML-Schema* posee algunas ventajas como por ejemplo que son escritos en formato XML y soportan un mayor número de tipos de datos.

Hay dos formas de QtXmlPatterns que se pueden utilizar para validar esquemas: puede utilizar la API C++ en su aplicación Qt usando las clases *QXmlSchema* y *QXmlSchemaValidator* o puede emplearse la utilidad de línea de comando denominado *xmlpatternsvalidator* (ubicado en el "bin" del directorio de su Qt build) (25).

La clase *QXmlSchema* carga, compila y valida archivos XML Schema W3C. Se utiliza para representar los esquemas que se ajustan al esquema XML 1.0 (26). Esta clase solo está introducida hasta el momento en la versión 4.6 de Qt, que es precisamente la que se empleará en el desarrollo del subsistema de transmisión.

La clase *QXmlSchemaValidator* valida documentos con instancias de XML frente a un esquema XML de W3C. Las cargas de la clase *QXmlSchemaValidator*, analiza un documento de instancia XML y lo valida contra un esquema XML de W3C que ha sido compilado con *QXmlSchema*.

2.10.2 Soporte de medias.

Los objetos *media* son una instancia de la clase *mediaobject*, le permite iniciar, pausar y detener la reproducción de un flujo de medios de comunicación, es decir, que proporcionan un control básico de la reproducción. Usted puede pensar en el objeto como un sencillo reproductor multimedia.

Los datos de la media es proporcionada por recursos media, la cual se mantiene para los objetos media. La fuente de la media es un objeto independiente - una instancia de *MediaSource* - en *Phonon*. El contenido de la fuente será interpretado por los objetos media.

Un objeto media es siempre instanciado con el constructor por defecto y se suministra con una fuente media. Como complemento a los objetos media, *Phonon* también proporciona la clase *MediaController*, que proporciona control sobre funciones que son opcionales para una media determinada.

2.10.2.1 Audio

La clase *QSound* proporciona acceso a la plataforma de audio .Qt provee la operación de audio más comúnmente requerida en aplicaciones GUI: reproducir asincrónicamente un archivo de sonido. Para ello, es muy fácil utilizar la función estática *play ()*.

Una vez creado el objeto *QSound* puede ser consultado por su *fileName ()* y un número total de *loops ()* que indica el tiempo que se estará reproduciendo. El número de repeticiones puede ser alterado usando la función *setLoops ()*. Mientras el sonido se esté reproduciendo la función *loopsRemaining ()* retornará el número restante de repeticiones. Para determinar si el sonido ha terminado de reproducirse se debe emplear la función *isFinished ()*.

La reproducción del sonido usando el objeto *QSound* puede usar más memoria que la función estática, pero éste también puede reproducirse de forma inmediata. Para determinar si existen facilidades de sonido en la plataforma usted puede usar la función estática *isAvailable ()*.

Plataforma	Formato de Audio
Microsoft Windows	Lo utiliza el sistema multimedia subyacente, sólo son soportados archivos de sonido con formato WAVE.
X11	La Red del sistema de audio se utiliza si está disponible (NAS) soporta archivos WAVE.
Mac OS X	Se utiliza NSSound. Todos los formatos que soporta NSSound, incluyendo los formatos QuickTime, son compatibles con Qt para Mac OS X.
Qt for Embedded Linux	Es usado un servidor de mezcla de sonido. Solo el formato WAVE es soportado.

Tabla 1: Facilidades de sonido en plataformas

2.10.2.2 Video

Para la reproducción de un video, se proporciona *VideoWidget*. Los widget automáticamente elegirán un dispositivo disponible para la reproducción del video, que suele ser una tecnología entre la aplicación Qt y la tarjeta gráfica, como DirectShow en Windows.

El *VideoWidget* no tiene por qué ajustarse a una categoría, es clasificado automáticamente por *VideoCategory*, sólo hay que asegurar que el audio también se incluye en la misma categoría. Para realizar el flujo de video, Phonon lee la dirección URL del archivo de video dondequiera que ésta esté.

Existen dos formas de realizar las transmisiones:

VOD (video bajo demanda): el usuario puede interactuar con el recurso que se pone a disposición haciéndole peticiones al servidor, entiéndase como, reproducir, poner pausa o parar.

Brocadas: los usuarios no pueden interactuar con el recurso, solo se puede manejar desde el servidor (27).

Muchos son los protocolos que permiten la transmisión por streaming como por ejemplo RTSP, MMS, RTP, UDP y otros. Hasta el momento solo se han probado dos de ellos en la plataforma PRIMICIA: RTSP Y MMS.

2.10.2.3 Texto

Qt contiene la clase *QGraphicsTextItem* que proporciona un elemento de texto que se puede agregar a un *QGraphicsScene* para mostrar texto con formato.

Para cambiar el texto del elemento, se debe pasar un objeto *QString* al constructor de la clase *QGraphicsTextItem*, o llamar la función `setHtml ()` / `setPlainText ()`. *QGraphicsTextItem* utiliza el tamaño de formato del texto y la fuente asociada a proporcionar una aplicación razonable de `boundingRect ()`, `shape ()`, y `contains ()`. La fuente puede ser cambiada llamando `setFont ()`.

Es posible crear un documento editable mediante la configuración de bandera de `Qt::TextEditorInteraction` utilizando la función `setTextInteractionFlags ()`.

El ancho de texto del elemento puede ser cambiado usando `setTextWidth ()` y obtenido mediante `textwidth ()`.

2.10.2.4 Imagen

Qt proporciona cuatro clases para el manejo de datos de la imagen: `QImage`, `QPixmap`, `QBitmap` y `QPicture`. `QImage` está diseñado y optimizado para E / S, y para la manipulación y acceso directo a píxeles, mientras que `QPixmap` está diseñado y optimizado para mostrar imágenes en pantalla. `QBitmap` es sólo una clase de conveniencia que hereda de `QPixmap` y que facilita la implementación. Por último, la clase `QPicture` es un dispositivo que graba y reproduce comandos `QPainter` (28).

La clase `QPixmap` es una representación de la imagen fuera de pantalla que se puede utilizar como un dispositivo de pintura. Es la clase que se utiliza actualmente para la manipulación de la imagen ya que no se desea modificarla sino mostrarla.

Un objeto `QPixmap` se puede visualizar fácilmente en la pantalla usando la clase `QLabel` que incluye la propiedad `pixmap` o una de las subclases `QAbstractButton` (como `QPushButton` y `QToolButton`).

Además de los constructores ordinarios, un objeto `QPixmap` se puede construir utilizando el `grabWidget` estática () y `grabWindow ()`, funciones que crea un `QPixmap` y pinta el widget dado.

Hay funciones para convertir entre `QImage` y `QPixmap`. Por lo general, la clase `QImage` se utiliza para cargar un archivo de imagen, opcionalmente manipular los datos de la imagen antes que el objeto `QImage` sea convertido en un `QPixmap` para ser mostrado en pantalla. Alternativamente, si no se desea la manipulación, el archivo de imagen se puede cargar directamente en un `QPixmap`. En Windows, la clase `QPixmap` soporta la conversión entre `HBITMAP` y `QPixmap`. En Symbian, la clase `QPixmap` también soporta la conversión entre `CFbsBitmap` y `QPixmap`.

Formato	Descripción	Soporte de Qt
BMP	Windows Bit Map	Leer/escribir
GIF	Graphic Interchange Format (opcional)	Leer
JPG	Joint Photographic Experts Group	Leer/escribir

JPEG	Joint Photographic Experts Group	Leer/escribir
PNG	Portable Networks Graphics	Leer/escribir
PBM	Portable Bitmap	Leer
PGM	Portable Graymap	Leer
PPM	Portable Pixmap	Leer/escribir
XBM	X11 Bitmap	Leer/escribir
XPM	X11 Pixmap	Leer/escribir

Tabla 2: Formatos de imágenes en Qt

2.10.3 Uso del timer

La clase *QTimer* proporciona temporizadores repetitivos y con muy corta duración.

La clase *QTimer* proporciona una interfaz de programación de alto nivel para los temporizadores. Para usarlo, cree un objeto *QTimer*, conecte su señal de *timeout ()* a los slots correspondientes, y llame la función *start ()*. A partir de entonces se emitirá el *timeout ()* de la señal a intervalos constantes.

En las aplicaciones multiproceso, puede utilizar *QTimer* en cualquier subproceso que tiene un ciclo de eventos. Para iniciar un ciclo de eventos de un hilo sin interfaz gráfica, use *QThread::exec ()*. Qt usa la afinidad del temporizador de hilo para determinar qué hilo emitirá el *timeout ()* de la señal. Debido a esto, debe iniciar y detener el temporizador en su hilo, no es posible iniciar un temporizador desde otro hilo (29).

Una alternativa al uso *QTimer* es llamar *QObject::startTimer ()* por su objeto y reimplementar el *QObject::TimerEvent ()* del controlador de eventos en su clase (que debe heredar *QObject*). La desventaja es que *TimerEvent ()* no admiten las funciones de tan alto nivel como contadores de tiempo de disparo único o señales.

Otra alternativa al uso de *QTimer* es utilizar *QBasicTimer*. Típicamente es menos incómodo que el uso de *QObject::startTimer ()* directamente.

2.10.4 Programación por hilos.

La clase *QThread* proporciona hilos independientes de la plataforma. Un objeto *QThread* representa un hilo de control separado dentro del programa que comparte datos con todos los otros hilos en el proceso, pero se ejecuta de forma independiente. En lugar de comenzar en *main ()*, comenzará a ejecutar en *QThread::run ()*. De forma predeterminada, *run ()* inicia el ciclo de eventos llamando a *exec ()*.

Esto creará un *QTcpSocket* en el hilo y luego ejecutará el bucle del hilo de eventos. Utilice el método *start ()* para comenzar la ejecución. La ejecución termina cuando retornas de *run ()*, al igual que una aplicación hace cuando sale de *main ()*. *QThread* le notifica a través de una señal cuando el hilo se inicia *started ()*, se acaba *finished ()*, o puede utilizar *isFinished ()* y *isRunning ()* para consultar el estado del hilo. Utilice *wait ()* para bloquear hasta que el hilo ha terminado su ejecución.

Cada *QThread* puede tener su propio ciclo de eventos. Puede iniciar el ciclo de eventos llamando a *exec ()* y puede detenerlo llamando a *exit ()*. Tras un ciclo de eventos en un hilo, hace posible la conexión de señales de otros hilos utilizando un mecanismo llamando en cola las conexiones. También hace posible el uso de las clases que requieren el bucle de eventos, tales como *QTimer* y *QTcpSocket*, en el hilo.

Es fácil desarrollar aplicaciones Qt portables multihilo y tomar ventaja de los multiprocesadores de las máquinas. La programación multihilo es también un paradigma útil para realizar operaciones que consumen mucho tiempo.

El *QtConcurrent* namespace proporciona un API de alto nivel que lo hacen posible escribir programas multi-hilo sin necesidad de utilizar primitivas de hilos de bajo nivel tales como exclusiones mutuas, bloqueos de lectura y escritura, condiciones de espera o semáforos. Los programas escritos con *QtConcurrent* automáticamente ajustan el número de hilos utilizados en función del número disponible de núcleos de procesamiento.

2.11 Conclusiones Parciales

En este capítulo se realizó un análisis de las tecnologías sobre las cuales se trabajará para el desarrollo del subsistema de transmisión de noticias PRIMICIA, realizando para ello su correspondiente caracterización y comparación para escoger los lenguajes de programación, sistema gestor de base de datos, framework y otras. Al terminar este capítulo se logra dominar la fundamentación teórica necesaria para comprender la construcción del subsistema de transmisión de noticias.

CAPÍTULO 3. PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS.

3.1 Introducción.

Luego de haber conceptualizado las tecnologías para desarrollar el subsistema de transmisión de noticias como una aplicación de escritorio, se hace necesario proponer cuáles de ellas formarán parte de la propuesta de solución, así como validar dicha propuesta. Para ello se tomarán los criterios de los expertos mediante el empleo del *Método Delphi*. Para lograr dicha validación se aplicará una encuesta a los desarrolladores del subsistema de transmisión de noticias, así como a su líder de proyecto. Luego se procesarán las encuestas aplicadas para obtener los argumentos necesarios para acreditar o rechazar la propuesta.

3.2 Herramientas y tecnologías propuestas.

Las tecnologías que formarán parte de la propuesta de solución son:

- Lenguaje de programación: C++, ya que este permite potenciar el rendimiento del hardware de la máquina.
- Lenguaje de modelado: UML, para especificar, visualizar, construir y documentar artefactos de un sistema de software.
- Metodología de desarrollo de software: RUP, para guiar el proceso de desarrollo del software.
- Framework: Qt 4.6, debido a que el mismo incluye un framework de animación que permite realizar excelentes animaciones personalizadas y funciones de interpolación. Además, cuenta con efectos que pueden ser utilizados para alterar la apariencia de los elementos de la interfaz de usuario. Es posible también aplicar efectos personalizados.
- Entorno Integrado de Desarrollo: Qt Creator 1.3, ya que se centra en proporcionar características que ayudan a los nuevos usuarios de Qt a desarrollar más rápido, así como aumentar la productividad de los desarrolladores con experiencia en Qt. El mismo está perfectamente integrado con las bibliotecas de clases Qt escrita genuinamente en código C++ nativo, ello potencia el manejo

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

de efectos, transparencias, sombras, videos y animación de objetos de forma fácil. Qt Creator integra completamente a Qt Designer, el cual posibilita el desarrollo de formularios y presentaciones gráficas para las aplicaciones, la previsualización de la interfaz, soporte para widgets y un editor de propiedades poderoso. Permite además acelerar el desarrollo de interfaces de alto rendimiento y a la vez proporciona una forma fácil de diseñar interfaces gráficas de usuario avanzadas generando el código fuente para las mismas, lo que permite al desarrollador ajustarlo a sus necesidades.

- Sistema Gestor de Base de Datos: PostgreSQL, ya que soporta gran parte del SQL estándar y muchas funcionalidades (consultas complejas, llaves foráneas, disparadores, vistas, integridad transaccional, así como el control de versionado concurrente (MVCC) que es una estrategia de almacenamiento que permite trabajar con grandes volúmenes de datos). Cumple completamente con las características ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) para realizar transacciones seguras. Utiliza licencia BSD, por lo que puede ser utilizado, modificado y distribuido por cualquier usuario gratuitamente para cualquier propósito ya sea con fines privados, comerciales o académicos.
- Herramienta de modelado de software: Visual Paradigm 6.4, ya que posibilita la importación y exportación de modelos de proyecto desde o hasta un formato XML, soporta múltiples usuarios trabajando sobre el mismo proyecto. Además permite dibujar todo tipo de diagramas UML, revertir código fuente a modelos UML, generar código fuente desde los diagramas. Es un producto que facilita a las organizaciones diseñar visualmente y con diagramas, integrar y desplegar sus aplicaciones empresariales y sus respectivas bases de datos. La herramienta ayuda al equipo de software a destacarse en el proceso de modelado-construcción-despliegue, maximizando y acelerando las contribuciones del equipo y los individuos.

3.3 Características del método Delphi.

El método Delphi es una técnica grupal de análisis de opinión, parte de una suposición fundamental y de que el criterio de un individuo particular es menos fiable que el de un grupo de personas en igualdad de condiciones, en general utiliza e investiga la opinión de expertos. “La técnica Delphi es como un método de investigación sociológica, que independientemente de que pertenece al tipo de entrevista de profundidad en grupo, se aparta de ellas agregando características particulares”. Se basa específicamente en la interrogación a expertos con la ayuda de cuestionarios sucesivos, a fin de poner manifiesto convergencias

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

de opiniones y deducir eventuales consensos. Es una consulta a un grupo de expertos en forma individual por medio de la interacción sucesiva de un cuestionario, apoyado por los resultados promedio de la ronda anterior a fin de generar convergencia de opiniones.

El principio de este método es la inteligencia colectiva tratando de lograr un consenso de opiniones expresadas individualmente por un grupo de personas seleccionadas cuidadosamente como expertos calificados en torno al tema. Todo ello se realiza por medio de la iteración sucesiva de un cuestionario retroalimentado de los resultados promedio de la ronda anterior aplicando cálculos estadísticos. Las principales características del método están dadas por el anonimato de los participantes (excepto el o los investigadores), iteración (manejar tantas rondas como sean necesarias), retroalimentación controlada, respuesta de grupo en forma estadística y justificación de respuestas.

Existen tres etapas o fases esenciales en la aplicación del método:

- **Fase preliminar:** Se delimita el contexto, los objetivos, el diseño, los elementos básicos del trabajo y la selección de los expertos.
- **Fase exploratoria:** Elaboración y aplicación de los cuestionarios.
- **Fase final:** Análisis estadísticos y presentación de la información.

3.3.1 Fase Preliminar

Durante esta primera fase primeramente se determina el objetivo a alcanzar con la aplicación del método seleccionado, después se realiza la selección del tipo de Delphi a utilizar y por último se produce la selección de los expertos que protagonizarán la evaluación de los resultados.

3.3.1.1 Objetivo a alcanzar

El principal objetivo a alcanzar con la aplicación del método Delphi es que teniendo como base la opinión de un grupo de expertos, los cuales algunos de ellos utilizan diariamente el subsistema de transmisión de noticias, se logre la validación de la propuesta de solución planteada de una manera rápida e íntegra, demostrando su efectividad y aplicación desde el punto de vista de los mencionados expertos en el tema.

3.3.1.2 Selección del tipo de Delphi a utilizar.

El método Delphi se puede clasificar de diferentes maneras teniendo en cuenta varios aspectos: por objetivos, por conducción, entre otros tipos. Para realizar la selección del tipo idóneo para esta investigación se tuvieron en cuenta las siguientes características: disminución de tiempo y flexibilidad de las respuestas. Después de un estudio realizado se llega a la conclusión que el método Delphi conocido como Delphi cara-cara, es el seleccionado para realizar la validación de la propuesta de solución. En esta variante del método Delphi, el cuestionario se lleva personalmente a cada integrante del panel, a quien se le hace la entrevista en forma individual, lo cual permite aumentar la flexibilidad de las respuestas, pues el entrevistador puede resolver cualquier duda o ambigüedad que se le presente al panelista en relación a las preguntas del cuestionario. Por otra parte se logra considerables ventajas de tiempo y se logra disminuir el porcentaje de deserción de los expertos.

3.3.1.3 Selección de expertos.

El criterio seleccionado para la selección de los expertos fue de *Efectividad en su actividad profesional*, este es el criterio más utilizado por ser el que logra mayor objetividad en la evaluación del resultado y a la vez es el más cómodo para proceder a realizar la selección de los expertos. Para aumentar la calidad de la evaluación se trata de seleccionar expertos de reconocida experiencia profesional avalada por su alta calificación científico-técnica, reconocido prestigio profesional, conocimiento profundo del tema objeto de investigación y resultados satisfactorios en el trabajo.

El panel de expertos estará compuesto por:

- Desarrolladores del subsistema transmisión de noticias (arquitecto, implementadores).
- Líder de proyecto.

3.3.2 Fase Exploratoria

Durante esta fase las principales actividades son la confección y aplicación de las encuestas, para la realización de estas actividades se tuvieron muy presentes la longitud y el tipo de las preguntas encuestadas. Un planteamiento demasiado conciso provoca una excesiva variedad de interpretaciones y uno muy largo requiere administrar demasiados elementos de una sola vez, por lo que la forma en que se

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

realizó el cuestionario fue con planteamientos de mediana longitud. Un cuestionario muy largo puede inducir al desinterés de los expertos y uno muy corto puede obstaculizar los objetivos a alcanzar, por lo que se elaboró un cuestionario ameno y atractivo con preguntas precisas y necesarias. Existen distintos tipos de preguntas que pueden ser formuladas a los panelistas con el objetivo de extraer información útil a los objetivos planteados. El cuestionario realizado a los expertos antes mencionados está compuesto por varios tipos de preguntas:

Preguntas abiertas: Para señalar justificaciones de una determinada estimación y mencionar factores que pueden influir en el comportamiento de una determinada cuestión.

Preguntas cerradas (votación): Se presentan dos o más alternativas a consideración de los panelistas, los cuales deben votar por una de ellas. Esta votación se realiza de acuerdo a algún criterio, el cual es señalado en el cuestionario y que puede ser factibilidad, probabilidad de ocurrencia, etc.

En el Anexo 2 se muestran el conjunto de preguntas encuestadas al panel de expertos, las cuales permitirán evaluar la factibilidad de las herramientas y tecnologías escogidas para el desarrollo del subsistema de transmisión de noticias.

3.3.3 Fase Final

La actividad fundamental de esta fase es el procesamiento y estudio de la información. La evaluación del cuestionario se hizo de forma manual pues la cantidad de información y datos a procesar no son excesivamente voluminosos ni complicados en su procesamiento. La principal meta de este método es llegar a un consenso entre los expertos, pero como no existe una única forma de medir consenso, se utilizan las siguientes reglas las cuales son las adoptadas para los estudios Delphi a nivel mundial:

- En las preguntas con dos alternativas, cuando una de ellas acumula al menos el 70% existe un alto grado de consenso entre los expertos.
- En las preguntas con más de dos alternativas, cuando alguna de ellas acumula al menos el 50% existe un alto grado de consenso entre los expertos.

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

3.3.3.1 Análisis de los resultados de la 1ra ronda.

A continuación se presenta una evaluación de los resultados de la aplicación del método Delphi. Cumplido el procesamiento de los cuestionarios, gracias al aporte de los expertos, los resultados de la solución propuesta se presentan a continuación:

Lenguaje	Consenso %
C++	77.78
C#	22.22
Java	0.00

Framework	Consenso %
Qt 4.6	88.89
Platinum C ++	0.00
Framework.Net	11.11
Hibernate	0.00

SGBD	Consenso %
PostgreSQL8.3	66.67
Oracle	0.00
MySQL	33.33

IDE	Consenso %
Qt Creator 1.3	88.89
Visual Studio	11.11
Netbeans	0.00
Eclipse	0.00

Herramienta modelado	Consenso %
Visual Paradigm	88.89
Rational Rose	11.11

Lenguaje modelado	Consenso %
UML	100
BPMN	0.00

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

Es importante destacar un aspecto relevante en los resultados obtenidos: el *consenso de los expertos*.

- El *consenso de los expertos* para los temas previstos en las preguntas con más de dos alternativas fue de 80.56 %, lo cual implica un alto nivel de consenso en las opiniones suministradas.
- El *consenso de los expertos* para los temas previstos en las preguntas con dos alternativas fue del 94.45 %, lo cual implica un alto nivel de consenso en las opiniones suministradas.

Pregunta sobre el Lenguaje de programación:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 77.78 % de los panelistas concordaron en que el lenguaje C++ es el idóneo para el desarrollo del subsistema de transmisión de noticias.

Pregunta sobre el framework de desarrollo:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 88.89 % de los panelistas concordaron en que el framework Qt 4.6 es el idóneo para el desarrollo del subsistema de transmisión de noticias.

Pregunta sobre el IDE:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 88.89 % de los panelistas concordaron en que el IDE Qt Creator 1.3 es el idóneo para el desarrollo del subsistema de transmisión de noticias.

Pregunta sobre el SGBD:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 66.67 % de los panelistas concordaron en que el SGBD PostgreSQL 8.3 es el idóneo para el desarrollo del subsistema de transmisión de noticias.

Pregunta sobre el lenguaje de modelado:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 100 % de los panelistas concordaron en que el lenguaje de modelado UML es el idóneo para el desarrollo del subsistema de transmisión de noticias.

Pregunta sobre la herramienta de modelado:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. El 88.89 % de los panelistas concordaron en que la herramienta de modelado *Visual Paradigm 6.4* es la más idónea para el desarrollo del subsistema de transmisión de noticias

3.3.3.2 Análisis de los resultados de la 2da ronda.

3.3.3.2.1 Soporte matemático empleado.

En la aplicación del método Delphi en esta investigación, se realizan dos rondas de consultas a los expertos, es decir, los individuos pertenecientes al panel, donde a partir de la retroalimentación realizada de los resultados de la primera ronda se confecciona un nuevo cuestionario donde se concreta el mayor consenso de las opiniones de los expertos.

Para medir el consenso y el coeficiente de variación entre los valores de las observaciones, es posible medirlos mediante la siguiente fórmula.

$$CV = \frac{S}{\bar{x}} * 100$$

Donde:

- S -> es la desviación estándar.
- \bar{x} -> es la media.

Las condiciones para determinar el nivel de consenso son las siguientes:

- Si el coeficiente de variación es del 30 % y hasta el 20 %, muestra un consenso aceptable.
- Si el coeficiente de variación es menor que el 20 % y hasta el 10 %, señala un consenso mayor.
- Si el coeficiente de variación es menor que el 10 %, indica un alto consenso.

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

Para calcular la desviación estándar se emplea la siguiente fórmula:

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}}$$

Donde:

- x_i -> son los valores de cada una de las observaciones.
- n -> es la cantidad de observaciones.

Para calcular la media se emplea la siguiente fórmula:

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

Al concluir el procesamiento del segundo cuestionario, gracias al aporte de los expertos y aplicando las diferentes fórmulas antes mencionadas, se arribó a los resultados que se presentan a continuación:

Pregunta sobre el Lenguaje de programación:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. Se determinó que el coeficiente de variación sobre los criterios emitidos por los expertos sobre el *lenguaje de programación propuesto "C++"* es de 19.34 %. Ello indica que el consenso entre los expertos tiene la categoría de "Mayor".

Pregunta sobre el framework de desarrollo:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. Se determinó que el coeficiente de variación sobre los criterios emitidos por los expertos sobre el *framework de desarrollo propuesto "Qt 4.6"* es de 6.82 %. Ello indica que el consenso entre los expertos tiene la categoría de "Alto".

Pregunta sobre el IDE:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. Se determinó un coeficiente de variación sobre los criterios emitidos por los expertos sobre el *entorno*

CAPÍTULO 3: PROPUESTA Y VALIDACIÓN DE LAS TECNOLOGÍAS SELECCIONADAS

integrado de desarrollo propuesto “Qt Creator 1.3” es de 6.82 %. Ello indica que el consenso entre los expertos tiene la categoría de “Alto”.

Pregunta sobre el SGBD:

Esta pregunta la respondieron 9 entrevistados, lo cual representa un 100% del total del panel. Se determinó un coeficiente de variación sobre los criterios emitidos por los expertos sobre *el Sistema Gestor de Bases de Datos propuesto “PostgreSQL”* es de 23.03 %. Ello indica que el consenso entre los expertos tiene la categoría de “Aceptable”.

3.4 Conclusiones Parciales

En este capítulo se ha validado la propuesta de las tecnologías seleccionadas para el desarrollo del subsistema de transmisión de noticias de la Plataforma de Televisión Informativa PRIMICIA, para ello se ha empleado el método Delphi mediante el panel de expertos. Los panelistas tuvieron un buen grado de concordancia en todas las preguntas, se aplicaron dos rondas de entrevistas, donde se concluye según el criterio de los expertos que:

- El lenguaje de programación C++ es el más idóneo para lograr un elevado rendimiento del subsistema de transmisión.
- El framework Qt 4.6 garantizará no solo el rendimiento del subsistema de transmisión, sino también logra potenciar la animación y los efectos gráficos.
- El IDE Qt Creator 1.3 es el más recomendado, ya que posee una completa integración con el framework Qt 4.6, proporciona un diseñador de formularios (GUI) integrado y un editor de código avanzado para C++.
- El SGBD PostgreSQL 8.3 proveerá un buen manejo y almacenamiento de los datos persistentes.
- El Subversion es la herramienta de control de versiones más potente, es libre y de código abierto.
- La herramienta de modelado Visual Paradigm 6.4 según sus características es la más idónea para el desarrollo del subsistema de transmisión, ya que permite generar código fuente a partir de los diagramas UML y es software libre.

Teniendo en cuenta los resultados obtenidos con la aplicación del método Delphi para certificar la propuesta realizada, se puede concluir que la misma ha sido validada satisfactoriamente.

Conclusiones Generales

Durante todo el desarrollo de la presente investigación se han adquirido conocimientos que han permitido dar respuesta a la problemática actualmente surgida en la Universidad de las Ciencias Informáticas, específicamente en el Centro Productivo *GEySED*. Primeramente se llevó a cabo una caracterización bien detallada de cada una de las tecnologías candidatas para el desarrollo del subsistema de transmisión de noticias de la Plataforma de Televisión Informativa (PRIMICIA). Dicha caracterización facilitó el desarrollo de una comparación entre estas tecnologías teniendo en cuenta las tres variables que se necesitan potenciar en el subsistema de transmisión: animación, efectos y rendimiento. Como cumplimiento de los objetivos y tareas trazadas, se arriba a las siguientes conclusiones:

- Se realizó un estudio concreto de las tendencias y tecnologías actuales, lo cual permitió seleccionar las tecnologías adecuadas para el desarrollo de la solución propuesta con un alto grado de actualidad científica.
- Se logró desarrollar una propuesta de las tecnologías más potentes para mejorar del subsistema de transmisión de noticias de la Plataforma de Televisión Informativa (PRIMICIA), las cuales harán posible obtener un producto informático con un alto rendimiento y eficiencia.
- Se logró validar la propuesta de tecnologías a partir del criterio de expertos, lo cual sustenta que el producto final podrá contar con una alta calidad y aceptación en el mercado nacional e internacional.

Recomendaciones

A lo largo de este trabajo se apreció como se cumplieron cada uno de los objetivos trazados en el mismo. Además se observa claramente que el desarrollo obtenido potenciará el subsistema de transmisión de noticias de la Plataforma de Televisión Informativa (PRIMICIA), no obstante, se realizan varias recomendaciones que darán continuación a dicho trabajo:

- Aplicar de forma práctica la propuesta de tecnologías seleccionadas para el desarrollo del subsistema de transmisión con el fin de obtener un producto competente en el mercado.

- Proponer la implantación de sistemas de este tipo en hospitales, universidades, hoteles, otras sedes empresariales y ministeriales.

Referencias Bibliográficas.

1. **Vaughan, Tay.** *Todo el poder de la Multimedia.* México D.F: McGraw-Hill, 1994. pág. 561.
2. **Conallen, Jim.** *Modeling Web application architectures with UML, Communications of the ACM.* 10, 1999, Vol. 42, pág. 63 - 70.
3. **Sistema teletexto Europa,** [En línea] [Citado el: 28 de 02 de 2010.] <http://www.anglatecnic.com/es-6-Fingertext-%28Sistema-Teletexto%29.html>
4. **2009.** www.lenguajes-de-programacion.com. [En línea] 2009. [Citado el: 25 de 01 de 2010.] <http://www.lenguajes-de-programacion.com/programacion-java.shtml>.
5. **Planetacodigo.** [En línea] [Citado el: 28 de 01 de 2010.] <http://www.planetacodigo.com>.
6. **C++.** [En línea] [Citado el: 30 de 01 de 2010.] <http://es.wikipedia.org/wiki/C%2B%2B>
7. **Platinum framework,** [En línea] [Citado el: 1 de 03 de 2010.] www.pt-framework.sourceforge.net
8. **Qt 4**[En línea] [Citado el: 12 de 01 de 2010.] <http://doc.qt.nokia.com/4.6/qt4-6-intro.html>
9. **Eclipse IDE** [En línea] [Citado el: 11 de 03 de 2010.] http://campusvirtual.unex.es/cala/epistemowikia/index.php?title=Eclipse_SDK
10. **Eclipse** [En línea] [Citado el: 11 de 03 de 2010.] <http://www.eclipse.org/cdt/>
11. **Qt creator IDE** [En línea] [Citado el: 5 de 03 de 2010.] <http://www.osalt.com/es/qt-creator>
12. **Qt creator** [En línea] [Citado el: 5 de 03 de 2010.] <http://www.tuxradar.com/content/qt-creator>
13. **NetBeans IDE** [En línea] [Citado el: 15 de 05 de 2010.] <http://netbeans.org/features/cpp>
14. **TRAMUÑAS, J.** *Introducción a la Documática. Los sistemas de bases de datos y los SGBD. DL HU-159-2002.* ed. Zaragoza: KRONOS, Disponible en: <http://tramullas.com/documatica/2-4.html>

15. *Sun cerrará algunos componentes nuevos de MySQL.* <http://diarioti.com/gate/n.php?id=17422>.
16. **OLGA PONS CAPOTE, N. M.** *INTRODUCCIÓN A LOS SISTEMAS DE BASES DE DATOS.*
17. **SÁNCHEZ, G. C.** (2001). *SISTEMAS GESTORES DE BASES DE DATOS*
18. **PostgreSQL.org.** 2009. *PostgreSQL.* [En línea] 2009.
http://postgresql.org.pe/articles/pgsql_alta_disponibilidad.pdf.
19. **Ruiz, Santiago Gómez.** 2007. *Microsoft SQL Server, MySQL, PostgreSQL*
20. **Rational Case** [En línea] [Citado el: 20 de 03 de 2010.]
<http://www.taringa.net/posts/downloads/883942/Rational-Rose-Enterprise-Edition-2002.html>
21. **Visual Paradigm como herramienta Case** [En línea] [Citado el: 20 de 03 de 2010.]
<http://www.todoprogramas.com/programalinux/visualparadigmforum1>
22. **Visual Paradigm** [En línea] [Citado el: 15 de 03 de 2010.]
<http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>
23. **CVS** [En línea] [Citado el: 17 de 03 de 2010.] www.osmosislatina.com/soporte/cvs.html
24. **Java SAX2** [En línea] [Citado el: 15 de 05 de 2010.] <http://doc.qt.nokia.com/4.6/xml-sax.html>
25. **QtXmlPatterns** [En línea] [Citado el: 15 de 05 de 2010.]
<http://doc.qt.nokia.com/4.6/xmlprocessing.html>
26. **QXmlSchema** [En línea] [Citado el: 16 de 05 de 2010.]
<http://doc.qt.nokia.com/4.6/qxmlschema.html>
27. **Lattre, Alexis de, y otros.** *VideoLAN Streaming Howto.* s.l. : VideoLAN, 2002-2005
28. **QImage** [En línea] [Citado el: 18 de 05 de 2010.] <http://doc.qt.nokia.com/4.6/qimage.html>
29. **QTimer** [En línea] [Citado el: 18 de 05 de 2010.] <http://doc.qt.nokia.com/4.6/qtimer.html>

Bibliografía.

1. Breve historia surgimiento aplicaciones web: Disponible en <http://www.agendistas.com/computadoras/aplicaciones-multimedia.html>
2. Sobre aplicaciones. Disponible en: <http://www.solucionaweb.com/aplicaciones.php>
3. Web o escritorio. Disponible en: <http://www.gnuconsultores.com/es/ingenieria/desarrollo/escritorio>
4. Software Libre. Disponible en: <http://www.gnu.org/philosophy/free-sw.es.html>
5. Introducción a los lenguajes de programación. Disponible en: <http://es.kioskea.net/contents/langages/langages.php3>
6. Introducción a la programación. Disponible en: <http://www.lenguajes-de-programacion.com/programacion-java.shtml>
7. Lenguajes de programación. Disponible en: <http://www.monografias.com/trabajos/lengprog/lengprog.shtml>
8. Framework Qt. Disponible en: <http://qt.nokia.com>
9. Framework multiplataforma. Disponible en: http://gluc.unicauca.edu.co/wiki/index.php/Un_peque%C3%B1o_an%C3%A1lisis_de_wxWidgets._Un_framework_para_desarrollar_aplicaciones_multiplataforma
10. Sobre software libre. Disponible en: <http://softlibre.barrapunto.com/article.pl?sid=00/11/15/013208>
11. Qt Creator. Disponible en: <http://deuteria.blogspot.com/2009/03/hoy-ha-salido-qt-creator-10-y-yo-sigo.html>