

Universidad de las Ciencias Informáticas

Facultad 6



Título: Desarrollo del Componente de presentación de informaciones climáticas para la Plataforma de Televisión Informativa, PRIMICIA

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

Autor: Yanary Hernández Sosa

Tutor(es): Ing. Rayner Pupo Gómez

Junio 2012

Declaración de Autoría

Declaro ser el único autor de la presente tesis 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 ____ días del mes de _____ del año _____.

Yanary Hernández Sosa

Rayner Pupo Gómez

Firma del Autor

Firma del Tutor

Datos de Contacto

Tutor: Rayner Pupo Gómez.

Graduado de Ingeniero en Ciencias Informáticas en el año 2011, trabajador no docente en el centro GEYSED de la Facultad 6 en el proyecto Video Vigilancia.

Correo electrónico: rpgomez@uci.cu

De todo corazón:

A mi mamá por tener tanta confianza en mí. Por ser la persona más especial de mi vida.

A mi papá por estar siempre orgulloso de sus hijas y no permitir que exista algo más importante en el mundo que ellas.

A mi nany por apoyarme siempre, estar a mi lado, ser mi conciencia y guía.

A Yaise por todo el apoyo brindado en la realización de esta tesis. Por ser mi diseñador personal.

A Yami, quien a pesar de encontrarse lejos no podía dejar de mencionar porque ha sido como otra tata para mi hermana y para mí, gracias a ti hoy se me está cumpliendo uno de mis mayores sueños.

A mis primos, que más que eso son como hermanos y por quienes siento un cariño muy especial.

A mis abuelitos, por brindarme tanto amor y cariño en todos estos años

A mi cuñadito, por ayudarme a lograr entender que debía de hacer en la tesis y por cuidar de mi hermana cuando yo no podía.

A Dainovy, Maydelin, Dorgis y Yoandri, grandes amigos con quienes he podido contar desde el 1er año de la carrera y a quienes nunca podré olvidar; gracias por soportarme durante estos 5 años.

A mis otros amigos.....especialmente al piquete de los 11. A David, Alberto, Olivia, Claudia, Maidelis.

A Janet que más que una amiga, ha sido cotutora durante todos estos meses de sufrimiento.

A Geovanys, geo fuiste la persona que nos enseñó a ser verdaderos profesionales; siempre serás nuestro” Best SET”.

A mi tutor por todo el apoyo y esfuerzo brindado durante la realización de esta tesis.

A los miembros del tribunal y a Aliosmi por las recomendaciones realizadas y el apoyo brindado.

A todos los integrantes de PRIMICIA, a los de ahora y los que estuvieron antes, con los cuales he compartido los últimos tres años de la carrera.

A la Revolución Cubana y a Fidel Castro; por brindarnos la posibilidad de formarnos como ingenieros.

A todos los que de una forma u otra me ayudaron en la realización de esta tesis.... Muchas Gracias.

A mi madre:

Porque siempre has estado presente en los momentos difíciles de mi hermana y míos, porque has dedicado todo tu amor y esfuerzo en hacernos la vida mejor, porque nunca has tenido tiempo para pensar si necesitas algo porque siempre nosotras estamos primero. Porque incluso si necesitamos que estés enferma, para nosotras poder estar mejor, lo has estado. Por todos los viajes realizados cada domingo cuando estuve fuera de casa. Por saber que dejarme partir era lo mejor para nosotras a pesar de que no querías que estuviera lejos. Por siempre ir a recibirme cuando llego de la escuela pero nunca a despedirme. Porque con solo veinte tantos años y en medio de tanta tristeza supiste salir adelante por tus hijas. Porque tú, papi y mi hermana son lo más importante que tengo en esta vida.

A mi padre:

Porque siempre has estado presente cuando te he necesitado, se que solo necesito llamarte para que vengas corriendo a darnos tu ayuda. Por ponernos siempre en primer lugar y por delante de todo. Por tus consejos oportunos. Por tu constante lucha y sacrificio para que nosotras y la familia estemos mejor. Por tener la valentía de siempre decir lo que piensas, aunque no nos guste. Por ser mi ejemplo de revolucionario, honradez y tenacidad.

A mi hermana:

Mi nany bella, que siempre ha estado a mi lado. Por ser no solo hermana sino amiga, confidente, cómplice, incluso verdugo cuando has tenido que serlo, mi guía, mi ejemplo a seguir... meta a alcanzar, mi segunda madre cuando mami ha estado lejos. Por todas las responsabilidades que por mí has tenido que adquirir aunque solo me llevas tres años. Por asistir a tantas reuniones "de padres" cuando mami estaba trabajando y no podía ir personalmente.

A Yaise:

Por compartir estos momentos difíciles de mi vida, por saber diferenciar cuando necesito al amigo y cuando al novio. Por estar a mi lado sin dormir, todas las noches de sufrimiento y llanto. Por recordarme que después de lo peor viene lo mejor que te brinda la vida. Por saber, sin tener que habértelo dicho, lo importante que es para mí visitar a mi hermana cada semana. Por darme apoyo y fuerzas cuando más lo necesito. Por ser sencillamente tú.

En Cuba mantenerse actualizados sobre las características del estado del tiempo constituye una actividad primordial dentro de las acciones a realizar a diario, por la compleja situación geográfica en la que se encuentra ubicado el país y por las propias características que definen al clima como un sistema de ciclos complejos de predecir. Con el fin de mantener informadas a varias cantidades de personas en empresas, instituciones o entidades; en la Universidad de las Ciencias Informáticas se desarrolla la Plataforma de Televisión Informativa, PRIMICIA, que muestra a través de un canal de televisión noticias sobre distintos temas como: cultura, deporte, economía, entre otros tipos de informaciones que resulten de interés para el cliente. En el presente trabajo de diploma se desarrolla un componente para la presentación de informaciones meteorológicas especializadas, en este proyecto, con el objetivo de ampliar sus funcionalidades, de permitirle adquirir nuevos clientes y de brindarles a los usuarios la posibilidad de mantenerse actualizados sobre las características de la misma.

Palabras clave: información meteorológica, noticias, televisión.

Índice de Contenidos

Introducción	1
Capítulo 1: Fundamentos teóricos del Componente Climático	6
1.1. Introducción.....	6
1.2. Términos asociados al dominio del problema	6
1.3. Situación problemática	12
1.4. Análisis de otras soluciones existentes	13
1.5. Metodología de desarrollo a emplear	16
1.6. Tecnologías a utilizar.....	19
1.6.1. Lenguaje de programación.....	19
1.6.2. Marco de trabajo	21
1.6.3. Entorno de desarrollo integrado, IDE	23
1.6.4. Herramienta CASE para el modelado.....	25
1.7. Conclusiones.....	27
Capítulo 2: Análisis, diseño e implementación del Componente Climático	28
2.1. Introducción.....	28
2.2. Modelo de dominio	28
2.2.1. Principales eventos del entorno.....	29
2.2.2. Glosario de términos del dominio	29
2.3. Especificación de los requisitos del software	30
2.3.1. Requisitos funcionales	31
2.3.2. Requisitos no funcionales.....	34
2.4. Descripción de la solución propuesta	36
2.4.1. Definición de los actores del sistema	36
2.4.2. Diagrama de casos de uso del sistema	36
2.4.3. Descripción de los casos de uso del sistema	36
2.5. Modelo de análisis.....	42
2.5.1. Diagrama de clases del análisis	43
2.6. Arquitectura de software.....	43
2.6.1. Patrones de arquitectura: Arquitectura en capas	44
2.7. Modelo de diseño	44
2.7.1. Patrones de diseño	45

2.7.2.	Diagrama de clases del diseño	46
2.8	Modelo de despliegue	47
2.9	Diagrama de componentes	48
2.10	Estándar de codificación	49
2.11	Conclusiones	51
Capítulo 3:	Validación y pruebas realizadas al Componente Climático	52
3.1.	Introducción.....	52
3.2.	Pruebas de software	52
3.2.1.	Niveles de prueba	53
3.2.1.1.	Pruebas de aceptación	53
3.2.2.	Tipos de pruebas.....	53
3.2.2.1.	Pruebas de caja negra.....	53
3.2.2.2.	Pruebas de caja blanca	54
3.3.	Validación de la implementación.....	55
3.3.1.	Evaluación de la ejecución de las pruebas de caja blanca.	55
Conclusiones	60
Recomendaciones	61
Bibliografía	62
Anexos	67
Glosario de Términos	70

Introducción

El constante desarrollo del hombre y la necesidad de satisfacer los deseos de comunicación con sus semejantes, de transmitir opiniones, ideas, pensamientos y proyectos ha constituido el motor impulsor para la creación en el mundo de tecnologías más ágiles y potentes en el desarrollo comunicativo. La televisión es uno de los medios de comunicación más utilizado e influyente a nivel mundial por la capacidad que posee de llegar a millones de personas de forma simultánea. En los últimos años han surgido nuevas formas de hacer televisión, gracias al desarrollo tecnológico que ha alcanzado este medio y a la vinculación con campos como la informática y las telecomunicaciones, constituyendo uno de los avances más significativos de la revolución científico-tecnológica.

Cuba no cesa en la constante evolución hacia la informatización educativa de la sociedad teniendo como base las tecnologías de la información y las comunicaciones, TIC. Varios han sido los proyectos que se han realizado en esta área, donde la Universidad de las Ciencias Informáticas, UCI, desde su surgimiento en el año 2002 ha poseído un papel protagónico; permitiendo el desarrollo de sistemas informáticos dirigidos a usuarios nacionales e internacionales. Como parte de este proceso para la creación de productos que brindan un aporte económico y social al país surge la Plataforma de Televisión Informativa, PRIMICIA, la cual ha permitido la visualización de informaciones utilizando los televisores y las computadoras.

PRIMICIA es un producto informático creado en el año 2005 que constituye una solución integral capaz de proveer un canal de televisión para la transmisión automática y constante de informaciones en distintos formatos; siendo el resultado de la generalización de un conjunto de aplicaciones realizadas en el departamento Señales Digitales del centro GEYSED. El actual PRIMICIA se encuentra basado en las experiencias adquiridas a partir de personalizaciones desarrolladas a este producto como es el caso del canal de televisión interno de la UCI, Señal 3, creado desde el año 2005 con el fin de mantener informada a la comunidad universitaria de manera rápida y eficiente sobre temas relevantes e inmediatos. Un año después surge el canal informativo para la Agencia Cubana de Noticias, Señal ACN, el cual constituye un sistema automatizado de teletexto que facilita la transmisión de noticias de los distintos medios de prensa a los cooperantes que cumplen misiones en el extranjero y a los habitantes de las zonas de silencio. Luego nace TV Energía en el año 2007, para mantener informados a los trabajadores y visitantes de la sede central del Ministerio del Poder Popular para la Energía y Petróleos de Venezuela, MENPET.

PRIMICIA se encuentra en un proceso de constante evolución y perfeccionamiento. Es un producto personalizable y ha obtenido varios resultados en disímiles eventos, sin embargo ninguna de las versiones actuales del producto cuenta con una interfaz adecuada para la visualización de noticias climáticas especializadas.

El clima es un sistema de ciclos, complejo de predecir y capaz de ocasionar severos daños a la sociedad y a la economía. Los especialistas cubanos y del resto del mundo realizan a diario considerables esfuerzos para determinar el estado meteorológico a partir de algunas señales sonoras o visuales que desde la atmósfera se emiten, basándose en que estas condiciones atmosféricas y meteorológicas contribuyen en la toma de decisiones efectuadas por las personas. Muchos de los sectores de la sociedad como: alimentación, agua, turismo, transporte, construcción, agricultura y energía utilizan la información y las predicciones climáticas para planificar y orientar sus actividades.

El conocimiento acerca de las variaciones climáticas es de mucha importancia para la reducción de riesgos y la protección de la salud humana, es por esto que la sociedad tiene la necesidad de mantenerse informada de forma constante a través de medios confiables y que muestren de forma clara las características del estado del tiempo. Las secciones emitidas en la televisión nacional para la visualización de noticias de tipo meteorológicas constituyen un ejemplo de sistemas que brindan a la población la posibilidad de mantenerse actualizada sobre las características del pronóstico del tiempo, sin embargo las personas se ven limitadas a observar este tipo de información en los reducidos espacios dedicados a la misma.

Debido a la situación problemática antes expuesta se identifica como **problema a resolver**: La Plataforma de Televisión Informativa, PRIMICIA, no cuenta con una herramienta que le permita la visualización de información climática especializada.

Atendiendo al problema planteado y sobre la base de la necesidad de investigación en el marco de esta situación, se tiene como **objeto de estudio** el proceso de transmisión de noticias en sistemas informativos automatizados y como **campo de acción** la transmisión de noticias climatológicas en la Plataforma de Televisión Informativa, PRIMICIA.

En esta investigación se plantea como **objetivo general** desarrollar un componente informático que permita la visualización de informaciones climáticas especializadas en la Plataforma de Televisión Informativa, PRIMICIA.

Para guiar la investigación se plantea la siguiente **idea a defender**: Con el desarrollo de un componente informático para la visualización de informaciones climáticas especializadas se logrará transmitir noticias relacionadas con el tema del clima de forma automática, permitiendo mantener a los usuarios del canal al tanto de las características del estado del tiempo.

Para cumplir con el objetivo general y dar solución al problema planteado se definen las siguientes **tareas de la investigación**:

1. Caracterizar los sistemas de transmisión de noticias climatológicas.
2. Caracterizar la Plataforma de Televisión Informativa, PRIMICIA.
3. Caracterizar las herramientas y metodologías para el desarrollo.
4. Definir los requisitos funcionales y no funcionales del sistema.
5. Modelar el sistema haciendo uso de una herramienta de modelado.
6. Implementar el componente de presentación de informaciones climáticas de la Plataforma de Televisión Informativa, PRIMICIA.
7. Validar la solución propuesta.

Con la realización de este trabajo de diploma se obtendrán los siguientes resultados:

1. Artefactos vinculados a la metodología utilizada.
2. Código fuente de la aplicación.
3. Informe resultante del proceso de investigación práctico en la construcción de la aplicación.

Durante el proceso investigativo de este trabajo fueron utilizados varios métodos científicos, tanto teóricos como empíricos.

Los métodos teóricos permitieron analizar las características del objeto de estudio de la investigación que no se podían observar a simple vista. El método **analítico-sintético** fue utilizado con el objetivo de establecer, de forma mental, una división del objeto de estudio en varias partes con el fin de realizar un mejor análisis de las características de las mismas. Además se utilizó el método **histórico-lógico** para

estudiar el proceso evolutivo de los sistemas transmisores de información y para saber cómo funcionan las aplicaciones que determinan la predicción meteorológica del tiempo.

Los métodos empíricos permitieron la descripción de las características fundamentales del objeto de estudio a partir de las experiencias obtenidas. En este proceso fue utilizado el método de la **observación**, para obtener información sobre los sistemas de transmisión de noticias que se encuentran actualmente en funcionamiento. También se logró recopilar, gracias a este método, las características presentes en las variables meteorológicas posibles a mostrar en el componente. El método de la **entrevista** fue utilizado con el fin de obtener información sobre la predicción de datos meteorológicos a partir de los conocimientos aportados por varios profesionales capacitados en el tema.

La estructura del documento está definida de la siguiente manera:

Capítulo 1: Fundamentos Teóricos

En este capítulo se exponen las características de los principales conceptos asociados al dominio del problema tales como: televisión, procesos de transmisión en canales de televisión, pronóstico meteorológico, entre otros. Además se realiza la descripción general del objeto de estudio, problema a resolver y estado del arte que precede a la realización de este trabajo. También se caracterizan las herramientas, lenguaje de programación y metodología seleccionada para el desarrollo de la aplicación informática.

Capítulo 2: Presentación de la Solución Propuesta

En el segundo capítulo se hace una descripción de la propuesta de solución del trabajo, con el fin de mostrar los contenidos abordados en la construcción del componente; se identifican los requisitos, se presenta la modelación de la aplicación informática, se diseña la arquitectura y se implementa el Componente de presentación de informaciones climáticas para la Plataforma de Televisión Informativa, PRIMICIA.

Capítulo 3: Validación y pruebas

En este capítulo se realiza el proceso de validar el producto desarrollado para la presentación de informaciones climáticas en PRIMICIA, a partir de las pruebas realizadas a la aplicación y de los

resultados obtenidos en las mismas. Se tiene como principal objetivo en este capítulo verificar si el sistema cumple con las especificaciones planteadas por el cliente.

Además se brindan en este documento conclusiones, recomendaciones, bibliografía, anexos y glosario de términos.

Capítulo 1: Fundamentos teóricos del Componente Climático

1.1. Introducción

Los fundamentos teóricos constituyen la base sobre la cual se realiza todo trabajo investigativo ya que permiten continuar con el desarrollo y la descripción del objeto de estudio, es por ello que en este capítulo se abordan una serie de conceptos que giran alrededor del objetivo general de la investigación y que permiten sintetizar y relacionar los temas vinculados con el problema a resolver. Además se analizan algunas de las soluciones existentes a nivel nacional e internacional y posibles herramientas a utilizar para el desarrollo de la aplicación.

1.2. Términos asociados al dominio del problema

Televisión

La palabra televisión proviene del vocablo griego “tele” (distancia) y del término latino “visio” (visión). Constituye un sistema para la transmisión y recepción de imágenes en movimiento y sonido a distancia. Esta transmisión puede ser efectuada mediante ondas de radio o por redes especializadas de televisión por cable, donde el receptor de las señales es el televisor (Sanjúan, 2007).

Inicialmente las transmisiones de televisión no incluían sonido y solo transmitían imágenes en blanco y negro. La idea de introducir en una misma emisión vídeo y audio se fue incorporando con posterioridad, planteándose dos posibilidades. En la primera, la televisión estaba destinada a complementar los programas de radio, transmitiendo imágenes de forma auxiliar. La otra entendía la transmisión de imágenes como aspecto principal junto con el sonido y la relacionaba con la transmisión de películas (Sanjúan, 2007).

La televisión posee capacidad de impacto y penetración social, por lo que en la actualidad se ha convertido en un elemento indispensable para el desarrollo de la humanidad. Constituye uno de los medios de comunicación más populares y de mayor audiencia a nivel mundial, es utilizada para satisfacer las necesidades de información y entretenimiento por lo que el volumen de receptores aumenta constantemente.

Existen varios tipos de televisión dentro de las que se encuentran: difusión analógica, difusión digital, televisión terrestre, televisión por cable, televisión por satélite y televisión por IP (Sanjúan, 2007). Este medio le ha permitido a las personas que la información llegue a ellos con más rapidez, pueden conocer las noticias al poco tiempo de que sucedan o incluso en el mismo momento con la transmisión en vivo y directo. En los últimos tiempos se han creado nuevas cadenas de televisión a partir del surgimiento de tecnologías como los satélites de telecomunicaciones y la televisión por cable, permitiendo la transmisión de noticias e informaciones sobre ciencia, deporte, cultura, estado meteorológico, entre otros temas.

Procesos de transmisión en canales de televisión

A finales de los años 80 del siglo XX se empezaron a desarrollar sistemas de digitalización, este proceso al realizarse en la televisión posee varias partes dentro de las que se encuentran las digitalizaciones de la producción y de la transmisión. Esta última se hizo posible gracias a las técnicas de compresión aplicadas en aquel momento y que lograron reducir el tamaño del flujo de datos existente.

Actualmente el proceso de transmisión en canales de televisión puede efectuarse de dos formas: tradicional o automatizada. La forma tradicional consiste en operar un *switch*¹ que recibe señales de video de distintas fuentes como: grabadoras de cintas de video o VTR², señales directas del estudio o de otras televisoras. La persona encargada de realizar estas operaciones se guía por una escaleta permitiendo así especificar el orden en que se mostrará cada espacio y el tiempo de duración. El avance de la informática ha posibilitado crear sistemas para automatizar el proceso de transmisión; permitiendo la gestión de la escaleta, el control total de las señales recibidas, la grabación automática, el control de los VTR, entre otras funcionalidades (Terrero, 2010).

El proceso de transmisión de la televisión digital posee tres áreas que se encuentran en correspondencia con las formas en que este proceso puede desarrollarse, aunque son similares en

¹ Dispositivo de conmutación que permite el control de distintos equipos.

² *Video Tape Recorder* (Grabador de cintas de vídeo).

cuanto a la tecnología que utilizan, dentro de estas se encuentran: vía satélite, por cable y vía radiofrecuencia terrestre, conocida como TDT³.

A nivel nacional se han desarrollado varios canales de televisión para la transmisión de informaciones variadas, formando parte de estos se encuentra la plataforma de televisión informativa desarrollada por la UCI, PRIMICIA. Esta se encarga de mantener actualizados tanto a usuarios nacionales como internacionales sobre noticias de diferentes temas que son visualizadas en pantalla a través de la gestión realizada por el mismo.

Plataforma de Televisión Informativa, PRIMICIA

La Plataforma de Televisión Informativa, PRIMICIA, es un producto informático creado por la Universidad de las Ciencias Informáticas en el año 2005 y que ha sido conceptualizado, desarrollado y soportado en software libre contribuyendo a la soberanía tecnológica del país. PRIMICIA provee un canal de televisión para la transmisión automática y constante de informaciones en distintos formatos como: texto, imagen, sonido y video.

Esta solución puede ser ajustable a las necesidades de cualquier cliente que tenga el interés de mantener informado de manera rápida y constante a un grupo significativo de personas dispersas; es por esto que constituyen posibles clientes las universidades, televisoras, centros de convenciones, sedes empresariales, hoteles, terminales de transportación o cualquier otra entidad que posea una red de televisión y desee transmitir informaciones en los formatos antes mencionados.

PRIMICIA se encuentra estructurada por los subsistemas de Transmisión y Administración, los cuales se relacionan trabajando como un solo sistema para lograr mostrar las noticias a través de la red de televisión de forma satisfactoria. El Subsistema de Administración, desarrollado empleando tecnologías para aplicaciones web, como su nombre lo indica es el responsable de permitir que a través de él se realice la administración del canal y la gestión de noticias y recursos multimedia. El Subsistema de Transmisión, mediante la utilización de tecnologías para aplicaciones de escritorio, es el encargado de visualizar las noticias y materiales publicados. Además, durante la transmisión de las noticias se hace

³ Televisión Digital Terrestre

posible la visualización de informaciones breves, promocionales o de carácter relevante a partir del uso de las infocintas⁴.

Las funcionalidades que presta la Plataforma de Televisión Informativa según los subsistemas son (Leyva, 2011):

Subsistema de Administración:

- Gestión de los usuarios del sistema, permite adicionar y eliminar usuarios, así como establecer y modificar los permisos de acceso en el sistema.
- Gestionar las secciones temáticas del canal, permite establecer el orden de las secciones, horario en que serán mostradas y habilitarlas o deshabilitarlas.
- Funcionalidades para la redacción de noticias, según los formatos definidos para las pantallas; la publicación de las noticias teniendo en cuenta fecha de inicio y fin de la publicación; la gestión de las noticias del canal que permitan modificar, eliminar y archivar las noticias; la administración del archivo de noticias del canal que permitan reutilizar y eliminar las mismas.
- Gestionar Recursos Multimedia, posibilita el almacenamiento, administración y reproducción de recursos multimedia como imágenes, música y video.
- Funcionalidades para la creación y administración de cintillos informativos o infocintas, la administración de los cintillos establece el orden de prioridad de muestra y la habilitación o deshabilitación de los mismos.
- Generación de reportes sobre la actividad del sistema, los reportes se realizarán sobre la actividad de los trabajadores del sistema, permite realizar búsquedas de noticias publicadas atendiendo distintos criterios como fecha de publicación, temática, palabras claves y título. Ofrece facilidades para la impresión de los reportes y la exportación de los reportes a formato digital.
- Administración de la señal del canal, que permite cambiar entre la señal del canal y la televisión en vivo de un canal externo.

⁴ Cintillo informativo que se muestra en la televisión.

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 visualizarán.
- Visualizar noticias, compuestas por pantallas de tipo texto, texto-imagen, imagen y video.
- Reproducir fondo musical, mientras se muestran las noticias, excepto cuando se muestre un video.
- Mostrar comentarios, que oriente al televidente acerca de lo que está observando en las pantallas de tipo imagen.
- Mostrar cintillos informativos o infocintas, para promocionar eventos de última hora o acontecimientos de gran importancia.
- Mostrar información adicional a la noticia, la fecha, hora, tiempo restante de la pantalla y titular de la próxima noticia y sección temática.
- Transmitir televisión en vivo, proveniente de una señal externa.
- Mostrar patrón del canal, cuando este se encuentre fuera de servicio.

Este producto se mantiene en constante avance y perfeccionamiento, cada día el equipo de desarrollo del mismo busca nuevas funcionalidades que le ofrezcan valor añadido a la aplicación y que le permita visualizar otros tipos de informaciones en el canal como es el caso de las meteorológicas. Esta solución informática le brinda al país considerables aportes no solo desde el punto de vista social sino también económico, científico, tecnológico y político.

Meteorología

Desde la antigüedad el hombre se preocupó por observar y dejar constancia de como ocurrían los cambios climáticos, asociando las variaciones de estos con ciclos repetitivos que correspondían con las estaciones del año y con algunos eventos atmosféricos. En la actualidad la meteorología es la ciencia que estudia las propiedades de la atmósfera que rodea la tierra. Los cambios de estas propiedades como son las temperaturas, presión, humedad, vientos, lluvias, y otros, constituyen el tiempo atmosférico. El promedio de todos estos cambios en algún lugar determinado de la tierra

constituye el clima del lugar. La atmósfera constituye el principal elemento del medio ambiente donde el hombre habita proporcionándole los componentes necesarios para que la vida se desarrolle y evolucione, como son el oxígeno y el agua (Acqweather, 2011).

Gracias a los avances tecnológicos ocurridos en el tema, actualmente es posible realizar la descripción, el análisis y la predicción de las variaciones diarias de las condiciones atmosféricas, así como el estudio de los movimientos de esta, cómo se manifiesta su proceso evolutivo y cómo se estructura y compone la misma. Además de realizar investigaciones sobre los datos obtenidos en cada etapa y que permiten mostrar que tan variable puede comportarse el clima, a partir del estudio que se realiza con la climatología.

Climatología

La climatología es la rama de la meteorología que se ocupa del estudio del tiempo pasado en los diferentes lugares de la tierra, utiliza las estadísticas para determinar los valores centrales, particularmente la media o promedio de las diferentes variables meteorológicas con las cuales se pueden clasificar los climas (Acqweather, 2011).

Una correcta planificación del desarrollo de los países debe estar fundamentada en el conocimiento de su clima pues con estas informaciones se puede distinguir, por ejemplo, cuales son los meses más calurosos o los más fríos, cuando llueve más en una región determinada, o cuando un lugar será atacado por huracanes (Acqweather, 2011).

Los conceptos de meteorología y climatología se encuentran relacionados entre sí, mientras que una de estas ciencias se ocupa de los fenómenos que ocurren a corto plazo en las capas bajas de la atmósfera la otra se dedica al estudio del clima y las variaciones que ocurren en este a lo largo del tiempo. Aunque ambas utilizan los mismos parámetros sus objetivos son distintos, es por esto que algunos autores consideran a la climatología como una rama más de la meteorología, basándose en el hecho de que resulta imposible estudiar el clima de cualquier lugar si previamente no se realizan los estudios meteorológicos correspondientes.

Pronóstico Meteorológico

El pronóstico del tiempo existe desde hace miles de años cuando los hombres trataban de avizorar, según algunos patrones, como se comportaría el clima para las próximas horas del día. El pronóstico meteorológico, también denominado pronóstico del tiempo, predicción del tiempo meteorológico, pronóstico climatológico o simplemente boletín del tiempo; es el encargado de prevenir el tiempo meteorológico futuro, por medio de cálculos y a partir del conocimiento de las condiciones atmosféricas (TuTiempo.Net, 2011).

En la actualidad es el resultado de sofisticados modelos numéricos, procesados por potentes computadoras tanto por los servicios meteorológicos nacionales, como por empresas privadas con fines comerciales. El área modelada puede ser la tierra completa a través de los modelos globales o cubrir regiones específicas con los modelos de meso-escala (Acqweather, 2011).

La visualización e interpretación de los pronósticos meteorológicos constituyen procesos importantes que deben de realizarse a diario en cada nación, mediante el mismo se brindan datos e informaciones de peso a tener en cuenta para la toma de decisiones y durante la realización de cualquier actividad de la población en general.

1.3. Situación problemática

En los últimos años la transmisión de noticias se ha visto estrechamente relacionada con la televisión, permitiendo informar sobre temas relevantes a grandes cantidades de personas al mismo tiempo. Los sistemas informativos automatizados son utilizados por parte de las televisoras del mundo y de esta forma dan paso a soluciones que permiten el procesamiento rápido y eficaz de la información. De esta forma constituyen una vía para sustituir los problemas que los antiguos sistemas analógicos producían en los procesos de transmisión del canal.

A nivel internacional existen variados sistemas de transmisión de noticias, aunque la mayoría de estos se encuentran sobre plataformas propietarias que exigen el pago de precios elevados por la prestación de sus servicios. Desde la creación de la UCI un grupo de profesores y estudiantes se propusieron solucionar este problema desarrollando sistemas informativos libres y multiplataforma que logran la

transmisión de noticias televisivas de forma organizada y automática las 24 horas del día. Dentro de los productos desarrollados se encuentran: Señal 3, Señal ACN, TV Energía y PRIMICIA.

En el mundo varios sistemas de transmisión de noticias se especializan en la visualización de informaciones meteorológicas de todo tipo debido a la necesidad que posee la sociedad de mantenerse informada sobre las características de este, teniendo en cuenta que el clima es un sistema complejo de predecir que influye en las actividades diarias de la población en general.

En Cuba mantenerse informados sobre las características del estado del tiempo constituye una necesidad imperiosa para todas las entidades y personas en general, teniendo en cuenta la compleja posición geográfica del archipiélago situado entre las penínsulas de Yucatán y la Florida. Existen varios medios de transmisión de informaciones climáticas a nivel nacional que velan por la protección de la sociedad, la economía y el medio ambiente, sin embargo en la programación televisiva nacional este tipo de datos solo son emitidos en reducidos espacios de tiempo.

PRIMICIA permite transmitir información variada sobre temas económicos, políticos, culturales, deportivos u otros que sean de interés para el cliente. Actualmente debido a la complejidad visual que trae aparejado el proceso de transmisión de información meteorológica, la plataforma no posee ninguna herramienta que le permita la visualización de este tipo de noticias. Por esto se plantea la necesidad de mostrar informaciones como: temperaturas mínimas y máximas, dirección y velocidad de los vientos, estado del tiempo, oleaje o fase de la luna en el canal.

1.4. Análisis de otras soluciones existentes

La evolución de las tecnologías informáticas y de los sistemas de transmisión de información ha permitido crear aplicaciones informáticas que permiten llevar hasta las personas las noticias sobre los distintos temas de interés. El constante desarrollo que se está produciendo en la actualidad ha traído como consecuencia que en el mundo existan varios sistemas para resolver un problema determinado, creando soluciones similares o derivadas de otras ya existentes en el mercado. Este proceso ocurre porque cada empresa productora de software trata de realizar aplicaciones informáticas que respondan a las necesidades de sus clientes, enfocadas en cumplir con un objetivo específico y con las políticas establecidas en la empresa. El mundo de los sistemas de transmisión de informaciones no queda

inmune a este auge existente por lo que a continuación se analizan soluciones desarrolladas en el mundo para resolver el problema planteado en esta investigación.

Weather Indicator

Weather Indicator es un *applet* que muestra en el panel de GNOME una serie de variables climáticas dentro de las que se encuentran: temperatura, velocidad del viento y humedad relativa de una ciudad determinada, además posee previsiones detalladas sobre el estado de los próximos días y sobre varias localidades. Es capaz de brindar alertas informativas sobre cualquier cambio ocurrido en el clima y permite configurar los datos de las temperaturas en varias escalas como: *celsius*, *kelvin* y *fahrenheit*. Aunque se encuentra disponible en los repositorios de Ubuntu las características de este no son actualizadas muy a menudo, por lo que el mismo ha ido quedando obsoleto.

A partir del estudio realizado a esta solución se determinó el tipo de visualización rápida mediante la cual el componente brindará a los usuarios, a partir de iconos y textos, características meteorológicas como: temperatura mínima y máxima, estado del tiempo y fase de la luna.

The Weather Channel

The Weather Channel (TWC), como su nombre lo indica, es un canal para la visualización de informaciones climatológicas. Creado desde 1982, el canal le ha proporcionado a la población mundial información meteorológica oportuna. Esta red de información meteorológica se encuentra disponible las 24 horas del día brindando los datos de las condiciones del tiempo actual, así como el pronóstico del clima para los próximos cinco días. Es visto en más de 96 millones de hogares estadounidenses por lo que constituye uno de los canales de datos meteorológicos, noticias e información más popular del mundo (T.W.C. Interactive, 2011).

Perteneciente a la *The Weather Channel Interactive*, es el principal proveedor de nuevas tecnologías en lo que a información meteorológica concierne; incluye banda ancha, aplicaciones interactivas de televisión, sitios y aplicaciones de Internet, y productos meteorológicos inalámbricos. Dentro de los productos desarrollados por la misma se encuentran: *Desktop Weather*, *The Weather Channel Mobile*, *The Weather Channel Radio Network*, *The Weather Channel Newspaper Services* y el sitio *weather.com*. Proporcionando en forma oportuna las condiciones actuales del tiempo, pronósticos de expertos, información sobre actividades cotidianas y datos de interés sobre salud, viajes y

esparcimiento, estas soluciones meteorológicas ayudan a los usuarios a planificar sus actividades (T.W.C. Interactive, 2011).

Canal El Tiempo

Desde el lanzamiento en octubre de 2010 este canal se ha posicionado como un medio de comunicación líder en el cubrimiento informativo de todas las regiones de Colombia, brindando información actualizada minuto a minuto y explicando las noticias que se transmiten en el mismo, gracias a los más de 300 periodistas de la sala de redacción y de la red de portales de El Tiempo Casa Editorial. Permite ser sintonizado a través de la televisión por medios de operadores y cuenta con un portal donde es posible ver su señal en vivo. Está dirigido especialmente a las personas mayores de 18 años, de todos los niveles socio-económicos, llegando a más de dos millones de hogares de esta nación (Canal El Tiempo, 2011).

Con el análisis realizado a los canales: *The Weather Channel* y Canal El Tiempo; se determinó como visualizar en la aplicación, mediante las vistas en forma de pantalla completa y en forma de infocinta, datos meteorológicos tales como: temperatura mínima y máxima, estado del tiempo, fase de la luna, velocidad y dirección de los vientos. Además en estos canales el ciclo noticioso se realiza durante las 24 horas del día y se especializan en transmitir noticias relacionadas con la meteorología.

Ámbito Nacional

En Cuba las TIC son utilizadas para crear una sociedad donde prime el derecho de todos sus integrantes a los conocimientos y se aboga porque la obtención de este tipo de información no sea un fenómeno excluyente de los países del primer mundo. Basados en una estrategia de desarrollo científico y tecnológico creado por la dirección del país; se están desarrollando aplicaciones que permiten informatizar a la población nacional, a pesar de que no existen todas las condiciones tecnológicas y económicas para una verdadera repercusión social de las actividades realizadas.

El proceso de transmisión de noticias en los canales de la televisión nacional aún se produce de forma tradicional, aunque en este tema la UCI ha realizado su aporte en la última década con la creación de la Plataforma de Televisión Informativa, PRIMICIA, que permite visualizar noticias de forma automática. En Cuba no existe ningún canal dedicado a la visualización de informaciones climáticas, por lo que la población se mantiene informada sobre las características del estado del tiempo mediante

las secciones dedicadas al tema en el Noticiero Nacional de Televisión y en las del sistema informativo de la televisión cubana. También las personas que tienen acceso a Internet pueden actualizarse sobre el estado del tiempo, visitando el sitio del Instituto de Meteorología.

1.5. Metodología de desarrollo a emplear

Las metodologías de desarrollo de software constituyen el conjunto de procedimientos, técnicas, herramientas y soporte documental que ayuda a los desarrolladores a realizar un nuevo software. Representan un marco de trabajo que tiene entre sus principales funciones: guiar, planificar, estructurar, controlar, manipular y dirigir el proceso de desarrollo de sistemas de información (Jacobson, y otros, 2000) (Rodríguez, 2011).

Se pueden clasificar en dos grupos:

Metodologías pesadas o tradicionales: Son las metodologías orientadas al control de los procesos, estableciendo rigurosamente las actividades a desarrollar, herramientas a utilizar y notaciones que se usarán (Carrillo Pérez, y otros, 2008).

Metodologías ligeras o ágiles: Son las metodologías orientadas a interactuar con el cliente y el desarrollo incremental del software, mostrando versiones parcialmente funcionales del software al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando (Carrillo Pérez, y otros, 2008).

En las últimas décadas el desarrollo de software se ha incrementado considerablemente y ha traído como consecuencia la existencia de un amplio cúmulo de productos que tienen como fin brindar a los usuarios un mejor servicio. Sin embargo, los requisitos de un software con respecto a otro varían pues están en dependencia del tipo de proyecto que se esté desarrollando o planea desarrollar, dando lugar a que exista una amplia diversidad de metodologías para la creación del software.

A la hora de determinar el tipo de metodología a utilizar para dirigir el proceso de desarrollo del componente de presentación de informaciones climáticas para PRIMICIA, se estableció la necesidad de utilizar una metodología que fuera capaz de generar amplia documentación debido que este producto se encuentra desarrollado por personal constantemente cambiante. Es importante documentar detalladamente cada uno de los procesos llevados a cabo para que los nuevos miembros

del proyecto puedan tener abundante información sobre las características presentes en cada uno de estos y logren realizar las tareas asignadas con mayor facilidad y rapidez.

La característica de generar abundante documentación se encuentra presente en las metodologías de tipo pesadas por lo que fueron analizadas el Proceso Unificado de Desarrollo (RUP) y *Microsoft Solution Framework*. Estas dos metodologías poseen características similares solo que la *Microsoft Solution Framework* fue desarrollada sobre sistemas propietarios, por lo que no es recomendable su utilización en la creación del componente para PRIMICIA pues este se desarrolla sobre plataformas libres. RUP constituye la metodología mediante la cual PRIMICIA ha realizado todo su proceso de desarrollo por lo que los miembros del proyecto poseen conocimientos sobre la utilización de la misma, lo que reduciría el tiempo empleado para el estudio de otra. Se decide realizar un análisis de las características fundamentales de esta metodología para definir si esta puede ser finalmente utilizada en la elaboración del componente.

Proceso Unificado de Desarrollo (*Rational Unified Process*, RUP)

RUP es un proceso de ingeniería del software y un producto comercial desarrollado por *Rational Software*, una compañía de IBM⁵. Proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades en una organización de desarrollo. Tiene como propósito asegurar que la producción de software de alta calidad se ajuste a las necesidades de los usuarios finales con costos y calendario predecibles.

Es una metodología de desarrollo de software que intenta integrar todos los aspectos a tener en cuenta durante todo el ciclo de vida del software, con el objetivo de hacer abarcables tanto pequeños como grandes proyectos de software (Martínez, y otros).

Plantea principios claves a tener en cuenta durante su utilización: adaptación del proceso, balance de prioridades, colaboración entre equipos, elaboración del nivel de abstracción y deberá permitir enfocarse en la calidad (Martínez, y otros). Esta metodología define varias disciplinas dentro de las que se encuentran: Modelado de Negocios, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Transición, Configuración, Administración del Cambio, Administración de Proyectos y Ambiente. Además divide el desarrollo en 4 fases que definen el ciclo de vida del producto. La fase de inicio tiene

⁵ *International Business Machines* (Corporación Internacional de Máquinas de Negocios).

como objetivo determinar la visión del proyecto y definir lo que se desea realizar. En la fase de elaboración se determina la la arquitectura óptima del proyecto. Por su parte la tercera fase, construcción, tiene como objetivo obtener la capacidad operacional inicial y por último la fase de transmisión tiene la responsabilidad de obtener el producto acabado y definido (Carrillo Pérez, y otros, 2008).

A continuación se presenta una imagen que muestra el esfuerzo a realizar en cada actividad según las fases de RUP.

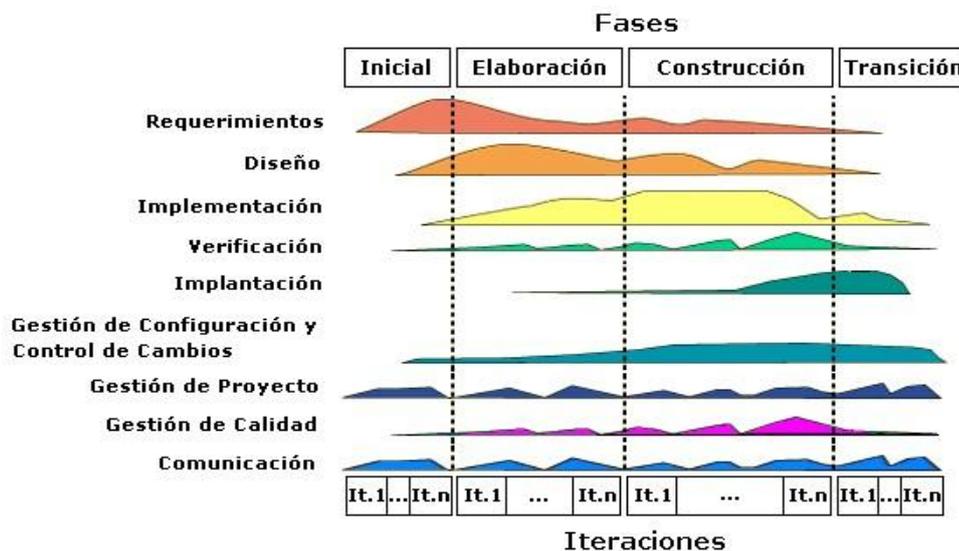


Figura 1: Fases, iteraciones y disciplinas de la metodología RUP.

El desarrollo del software propuesto por RUP tiene tres características esenciales: está dirigido por los casos de uso, está centrado en la arquitectura y es iterativo e incremental (Canós, y otros). Al constituir un proceso que se encuentra dirigido por casos de uso permite guiar las actividades a desarrollar durante todo el proceso de desarrollo. Además al analizar la característica que muestra que la metodología analizada está centrada en la arquitectura se determinó que a partir de la misma se lograría indicar cómo debe ser construido el sistema y brindaría su ayuda a la hora de definir en qué orden realizar este proceso. Se determinó que la característica de RUP que plantea que el mismo es interactivo e incremental es muy importante para la realización del componente; fundamentalmente porque le permite al equipo de desarrollo dividir el producto en partes más pequeñas o mini-proyectos, definiendo una o más iteraciones de las actividades básicas en la creación de este y de donde se

obtiene un incremento que produce un crecimiento en el producto final, permitiéndole al equipo de trabajo elaborar el mismo de forma sencilla y equilibrada.

Escoger una metodología de desarrollo adecuada para guiar la construcción de un software constituye un paso fundamental para que estos puedan ser entregados a tiempo y creados dentro del presupuesto planificado. Según las características y potencialidades brindadas por RUP, antes expuestas, el equipo de trabajo decide seleccionar esta metodología para ser utilizada en el proceso de desarrollo del componente. Fue escogida, en parte, por algunos factores que se encuentran presentes en el proyecto PRIMICIA como: tiempo, economía y recursos humanos. Al ser un proyecto con personal frecuentemente cambiante y que se encuentra en constante evolución se plantea la necesidad de poseer extensa documentación de todas las actividades que se ejecutan dentro del proceso de desarrollo. También se añade a la toma de esta decisión que RUP es una de las metodologías más conocidas y utilizadas, el equipo de desarrollo posee habilidades, conocimientos y familiaridad con esta y existe variada cantidad de material de estudio sobre el uso de forma apropiada de la misma.

1.6. Tecnologías a utilizar

1.6.1. Lenguaje de programación

Los lenguajes de programación son herramientas que permiten crear software. Están constituidos por un conjunto de símbolos y reglas sintácticas y semánticas que definen la estructura y el significado de los elementos y las expresiones de estos. Es utilizado para controlar el comportamiento físico y lógico de una máquina, permitiéndole a uno o más programadores especificar de manera precisa sobre qué datos debe operar una computadora, cómo estos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, representando en forma simbólica y en manera de un texto los códigos que podrán ser leídos y escritos por una persona que no necesariamente posea muchos conocimientos sobre programación (Louden, 2004) (Lenguajes de programación, 2009).

Teniendo como base que el componente para la visualización de las informaciones climatológicas será desarrollado utilizando tecnologías de escritorio y para el subsistema de transmisión de PRIMICIA, la

selección del lenguaje de programación a utilizar en la creación del mismo se rigió por aspectos relacionados con este tipo de tecnologías y las usadas por el subsistema. Además deberá de cumplir con parámetros como: familiarización con el lenguaje, consumo de memoria, eficacia y el tipo de lenguaje de programación que es. Esta investigación no se limita a analizar solo el lenguaje de programación C++, utilizado por el subsistema de transmisión, sino que analiza otros posibles lenguajes que permitan potencializar las características del canal.

Java

Java es un lenguaje de programación y la primera plataforma informática creada por *Sun Microsystems* en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. Se ejecuta en más de 850 millones de ordenadores personales de todo el mundo y en miles de millones de dispositivos, como dispositivos móviles y aparatos de televisión. Es un lenguaje de programación orientado a objetos, distribuido, interpretado, de arquitectura neutral, portable, seguro y fiable; que en su composición tiene elementos de C y C++, lo que mucho más simple sin hacer uso de herramientas de bajo nivel (Oracle, 2012) (Java, 2012).

En la actualidad es muy usual para los usuarios instalar máquinas virtuales de Java en los dispositivos que utilizan una aplicación desarrollada sobre este lenguaje, permitiendo ofrecer portabilidad entre plataformas. Sin embargo este proceso constituye una desventaja para los clientes pues estas aplicaciones generalmente son muy lentas.

C++

C++ es un lenguaje de programación derivado del C que soporta la programación orientada a objetos y la estructurada. Fue creado a mediados de los años 80 por Bjarne Stroustrup con el fin de agregar funcionalidades y características de las que carecía su antecesor, mantiene ventajas en cuanto a riqueza de operadores, expresiones, flexibilidad, concisión y eficiencia. Además, ha eliminado algunas de las dificultades y limitaciones del C original. Actualmente es uno de los lenguajes más potentes a nivel mundial, en gran medida esto se debe a que es un lenguaje compilado. Desde el punto de vista de los lenguajes orientados a objetos, C++ es un lenguaje híbrido y existen varios IDE que lo soportan como: Eclipse y Qt Creator (Lenguajes de programación, 2009) (Louden, 2004) (Programación en castellano, 2011).

Debido a las necesidades existentes de seleccionar un lenguaje de programación rápido y eficiente con el cual el equipo de desarrollo se encuentre familiarizado y que brinde las condiciones óptimas para la construcción exitosa del software, se ha llegado a la conclusión de que el lenguaje a utilizar es C++.

Dentro de las características tenidas en cuenta para su elección se encuentran (Rodríguez, 2011):

1. Es el lenguaje de programación que actualmente utiliza el subsistema de transmisión de PRIMICIA.
2. Es considerado por expertos como el lenguaje más potente para el trabajo con el hardware de la máquina.
3. En tiempo de ejecución, supera a su equivalente en otros lenguajes como C# y Java.
4. Es un lenguaje de propósito general, con el que los desarrolladores alcanzan acceso a recursos primarios como sólo este lenguaje puede ofrecer.
5. Potencia un mejor rendimiento de la caché de las aplicaciones permitiendo que estas sean más rápidas que las desarrolladas en otros lenguajes de programación.

1.6.2. Marco de trabajo

En el proceso de desarrollo de software un marco de trabajo, también definido como infraestructura digital o *framework*, es una estructura conceptual y tecnológica de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. Puede ser considerado como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta. Puede incluir lenguaje interpretado, bibliotecas, soporte a programas y otras herramientas para ayudar a desarrollar y unir los diferentes componentes de un proyecto (Gutiérrez, 2012) (Rodríguez, 2011).

Los objetivos principales que persigue un *framework* son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Existen *framework* de desarrollo en GNU/Linux como es el caso de Qt, que sirven como referencia para enfrentar y resolver nuevos problemas de índole similar.

QT

Qt es un *framework* de desarrollo multiplataforma para aplicaciones con interfaz gráfica de usuario o de consola que viene acompañado de un conjunto de herramientas para facilitar su uso. Incluye una biblioteca de clases intuitiva, integra herramientas de desarrollo y un IDE multiplataforma. Permite realizar aplicaciones avanzadas y desplegarlas en escritorio y sistemas operativos integrados sin tener que reescribir el código fuente, contribuyendo de esta forma con el tiempo y el costo de desarrollo del producto. Es fácil de usar, aprender, mantener y de código reutilizable, además posee un alto rendimiento en tiempo de ejecución y ocupa poco espacio en disco. Desarrollan sus propias herramientas, ideales para los programadores, como es el caso de Qt Creator para programadores C++, Qt Quick para desarrolladores de interfaces y Qt para dispositivos Nokia (Nokia, 2012) (Garrido, 2009).

Algunas de las características tenidas en cuenta para la selección de este marco de trabajo son:

1. Cuenta con abundante documentación y posee una herramienta para visualizar la misma, Qt Assistant (Rodríguez, 2011).
2. Internacionalización de aplicaciones con la ayuda de la herramienta Qt Linguist (Rodríguez, 2011).
3. Posee una arquitectura que soporta el uso de *plugins*, XML y conexión a bases de datos (Rodríguez, 2011).
4. Hasta el momento ha sido liberado bajo dos licencias: LGPL y comercial (Nokia, 2012).
5. Incluye un *framework* de animación que ayuda a construir animaciones fluidas, interfaces gráficas de alto rendimiento sin la molestia de la gestión de estructuras complejas y temporizadores (Rodríguez, 2011).

Qt presenta varias herramientas y componentes que facilitan el trabajo a los desarrolladores, se mencionan algunas a continuación (Garrido, 2009):

1. Librerías Qt: son clases escritas en C++ que facilitan el desarrollo del producto.
2. QtDesigner: utilizado para diseñar y crear interfaces visuales de forma cómoda.

3. QtAssistant: sencillo y potente visor de documentación que proporciona múltiples maneras de acceder a la información.
4. QtLinguist: permite la internalización de programas.
5. QMake: utilizado en la gestión de proyectos, simplifica el proceso de construcción de estos en las diferentes plataformas soportadas.

1.6.3. Entorno de desarrollo integrado, IDE

Un entorno integrado de desarrollo (en inglés *Integrated Development Environment*) es un programa informático compuesto por un conjunto de herramientas de programación, que proveen facilidades a los programadores para escribir códigos y agilizar el proceso de desarrollo de software. Pueden ser aplicaciones por sí solas o ser parte de aplicaciones existentes, además permiten ser utilizadas tanto con un único lenguaje de programación como con varios de estos. Las herramientas que normalmente componen un entorno de desarrollo integrado son las siguientes: editor de texto, compilador, intérprete, herramientas para la automatización, depurador, diseñador para la construcción de interfaces gráficas de usuario y, opcionalmente, un sistema de control de versiones (Rodríguez, 2011) (IDE, 2012) (Alvarez, 2010).

Para la selección del entorno integrado de desarrollo a utilizar en la construcción de la aplicación se definieron los siguientes parámetros a cumplir:

1. Facilidad de uso y abundante documentación.
2. Compatibilidad con GNU/Linux.
3. Integración con el *framework* de desarrollo escogido para la implementación de la aplicación.
4. Soporte para el lenguaje C++.
5. Integración con bibliotecas de Qt.

Eclipse

Es un entorno de desarrollo integrado de código abierto y multiplataforma, utilizado para varios lenguajes dentro de los que se encuentran: C, C++, Python y Java. Es una potente y completa plataforma de programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java. Recoge en una atractiva interfaz una serie de herramientas y funciones primordiales para el trabajo de los desarrolladores de productos informáticos. Posee una arquitectura abierta y basada en *plugins* que le permite conservar el registro de las versiones, generar y mantener la documentación de cada etapa del proyecto (Alvarez, 2010) (IDE, 2012) (EcuRed, 2012).

Qt Creator

Qt Creator es un entorno integrado de desarrollo creado por Trolltech y diseñado para hacer que el desarrollo en C++ de la aplicación Qt sea más rápido y fácil. Está basado en la biblioteca Qt, la biblioteca multiplataforma de interfaces gráficas de usuario. Pensado especialmente para el desarrollo en varias plataformas como: Windows, Linux y Mac OSX. Adaptado a las necesidades de los desarrolladores, permite crear aplicaciones de escritorio y plataformas de dispositivos móviles. Proporciona dos editores visuales integrados: Qt Designer para la creación de interfaces de usuario y Qt Quick Designer para el desarrollo de interfaces de usuario con el lenguaje QML (Rodríguez, 2011) (EcuRed, 2012).

Algunas de las características que se destacan de este IDE son (Programación en castellano, 2011):

1. Avanzado editor de código C++.
2. Tiene integrado diseñadores de interfaz gráfica de usuario.
3. Herramientas para la administración de proyectos.
4. Ayuda sensible al contexto.
5. Depurador visual.
6. Resaltado y autocompletado de código.

Durante la investigación desarrollada, teniendo como base los parámetros planteados para la selección del entorno de desarrollo integrado a utilizar y que el subsistema de transmisión de PRIMICIA utiliza el IDE Qt Creator, el autor ha decidido utilizarlo también para el desarrollo del componente.

1.6.4. Herramienta CASE para el modelado

Las herramientas CASE⁶ son un conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información. Se pueden definir como el conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante los pasos del ciclo de vida de desarrollo de un software (Alfaro, 1999).

Estas herramientas constituyen uno de los factores relevantes para la correcta planificación, estimación, diseño, implementación y detección de errores de los proyectos productivos realizados en la Universidad de las Ciencias Informáticas por lo que los equipos de desarrollo utilizan este tipo de tecnologías con frecuencia para lograr una mejor productividad, eficiencia y eficacia de los productos informáticos, así como la reducción del tiempo de construcción de los mismos. De esta forma se puede evidenciar la importancia que posee realizar una correcta selección de la herramienta CASE para el trabajo deseado, es por esto que a continuación se definen una serie de aspectos con los cuales la misma deberá de cumplir:

1. Facilidad de uso.
2. Modelar el software a través de diagramas UML.
3. Lograr integración con diferentes IDE.
4. Generar código en el lenguaje utilizado para el desarrollo.

Visual Paradigm para UML

Visual Paradigm para UML⁷ es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El

⁶ *Computer Aided Software Engineering* (Ingeniería de Software Asistida por Computador).

⁷ Lenguaje de modelado para especificar o describir métodos o procesos.

software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite representar todos los tipos de diagramas de clases y generar el código asociado a estos en lenguajes como: Java y C++ (Free Download Manager.ORG, 2007). Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque Orientado a Objetos (EcuRed, 2012).

Dentro de las principales características de esta herramienta se encuentran:

1. Disponibilidad en múltiples plataformas como: Windows y Linux (Visual Paradigm International, 2011).
2. Fácil de utilizar, instalar y actualizar: presenta un ambiente gráfico agradable e intuitivo para el usuario (EcuRed, 2012).
3. UML 2.1 habilitado: este es un estándar ampliamente utilizado actualmente en las empresas para el modelado de software que proporciona valiosa ayuda a los profesionales visualizando, comunicando y aplicando sus diseños (Uml.Slideshare, 2011).
4. Diseño centrado en casos de uso y enfocado al negocio que genera un software con mayor calidad (EcuRed, 2012).
5. Interoperabilidad entre diagramas: permite a partir de un diagrama obtener otro que guarde relación con el mismo (Urbino, 2010).
6. Generación de código y compatibilidad para diferentes lenguajes, entre ellos C++ (Uml.Slideshare, 2011).
7. Generador de informes (Visual Paradigm International, 2011).

Según los requisitos planteados para la selección de la herramienta CASE a utilizar en el desarrollo exitoso del producto se decidió emplear Visual Paradigm para UML pues la misma es la que más se ajusta a las necesidades del proyecto, es una herramienta profesional que soporta todo el ciclo de vida del software y permite el desarrollo de aplicaciones con rapidez, facilidad y calidad con un menor costo de producción.

1.7. Conclusiones

En este capítulo fueron abordados una serie de conceptos fundamentales que giran alrededor del objetivo general de la investigación y que constituyen el soporte teórico del desarrollo del componente. Se analizaron algunas soluciones similares existentes a nivel internacional y fue estudiado a fondo el estado actual de los medios de transmisión de información meteorológica, llegando a la conclusión de que a nivel nacional no existe ningún canal dedicado a la visualización de informaciones climáticas por lo que la población se mantiene informada, sobre las características del estado del tiempo, mediante las secciones dedicadas en el Noticiero Nacional de Televisión al tema y algunas pocas personas que tienen acceso a la red lo realizan visitando el sitio del Instituto de Meteorología. Además se describieron las características de cada una de las herramientas y tecnologías informáticas a emplear en el desarrollo del componente de presentación de informaciones climáticas para la Plataforma de Televisión informativa, PRIMICIA, determinando utilizar: metodología de desarrollo RUP, marco de trabajo Qt, IDE Qt Creator, lenguaje de programación C++ y el Visual Paradigm como herramienta CASE, todas estas tecnologías constituyen la base para la realización de forma exitosa del producto resultante de esta investigación.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

Capítulo 2: Análisis, diseño e implementación del Componente Climático

2.1 Introducción

En el presente capítulo se realizará la descripción de una serie de elementos imprescindibles para la comprensión de la propuesta de solución del trabajo, con el fin de mostrar los contenidos abordados en el desarrollo del producto. Con la construcción del modelo de dominio se llegará a un acercamiento mayor del problema a resolver y con la identificación de los requisitos funcionales y no funcionales serán mostradas las cualidades con los cuales deberá de cumplir el sistema. Además se muestran en el mismo un conjunto de diagramas desarrollados durante el análisis y diseño del componente según las características propuestas por la metodología escogida, RUP.

2.2 Modelo de dominio

En el proceso de análisis o investigación orientados a objetos, constituye un paso esencial el poder descomponer el problema en conceptos u objetos individuales: las cosas que se conocen. La designación de modelo conceptual ofrece la ventaja de subrayar fuertemente una concentración en los conceptos del dominio, no en las entidades del software (Larman, 2da Edición). El modelo de dominio es el encargado de capturar los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema (Jacobson, y otros, 2000).

Este modelo conceptual constituye el punto de partida para lograr realizar un diseño acertado del producto a desarrollar, definiendo a partir del mismo un conjunto de clases conceptuales u objetos existentes del mundo real en el dominio del problema planteado. El proyecto en el cual se encuentra enmarcada la realización del componente de presentación de informaciones climáticas constituye un producto personalizable y que se encuentra sujeto a una variedad constante de cambios, por lo que no existe en el mismo un proceso de negocio visible y bien definido. Como resultado a esto se determinó que realizar un modelo de dominio constituiría una forma óptima de manejar los cambios que se producen en el mismo de forma constante.

Capítulo 2: *Análisis, diseño e implementación del Componente Climático*

2.2.1. Principales eventos del entorno

PRIMICIA es una plataforma de televisión que tiene como principal objetivo mantener informada y actualizada a través de un canal de televisión a cualquier empresa, entidad o institución que requiera adquirir de forma inmediata y constante noticias de diferentes temas; según las preferencias de cada cliente. Dividido en dos subsistemas: Administración y Transmisión se encarga de hacerle llegar las informaciones a los usuarios de forma sencilla y clara; utilizando textos, imágenes, videos y sonido. Las noticias son gestionadas desde el subsistema de Administración de la plataforma y a continuación pasan a ser visualizadas mediante el segundo subsistema, Transmisión. Este proceso de transmisión puede ser desarrollado mediante tres formas diferentes: por secciones temáticas, por bloques de noticia o de forma continua.

A continuación se presenta el modelo de dominio y el glosario de términos realizados para lograr un mejor entendimiento de la relación existente entre los conceptos que sustentan el funcionamiento actual de la plataforma.

2.2.2 Glosario de términos del dominio

Imagen: figura, representación, semejanza o apariencia de algo. Constituye la representación visual realizada por el hombre de un objeto, una persona o un animal a través de alguna técnica como la pintura, el diseño o la fotografía.

Noticia: expresan hechos reales de forma abstracta, lejanos en el espacio y en continua renovación en el tiempo, es el componente más importante de los informativos emitidos por los medios de comunicación.

Texto: enunciado o conjunto coherente de enunciados emitidos de forma oral o escrita, que tienen una intención comunicativa a través de los signos y que se encuentran internamente estructurados. La extensión de un texto es variable, este puede estar compuesto por pocas palabras o por millones de estas por lo que es considerado como virtualmente infinito.

Video: sistema de grabación y reproducción de imágenes, que pueden estar acompañadas de sonidos.

XML: acrónimo de *eXtensible Markup Language* (Lenguaje de Etiquetado Extensible). Es un lenguaje de marcado que tiene el objetivo de facilitar la representación, almacenamiento y transmisión de información variada por parte de aplicaciones informáticas, computadoras y medios de comunicación

Capítulo 2: Análisis, diseño e implementación del Componente Climático

digital en general. Permite jerarquizar y estructurar la información, así como describir los contenidos dentro del propio documento y la reutilización de partes del mismo. La información estructurada puede presentar varios contenidos como: texto, imagen y audio.

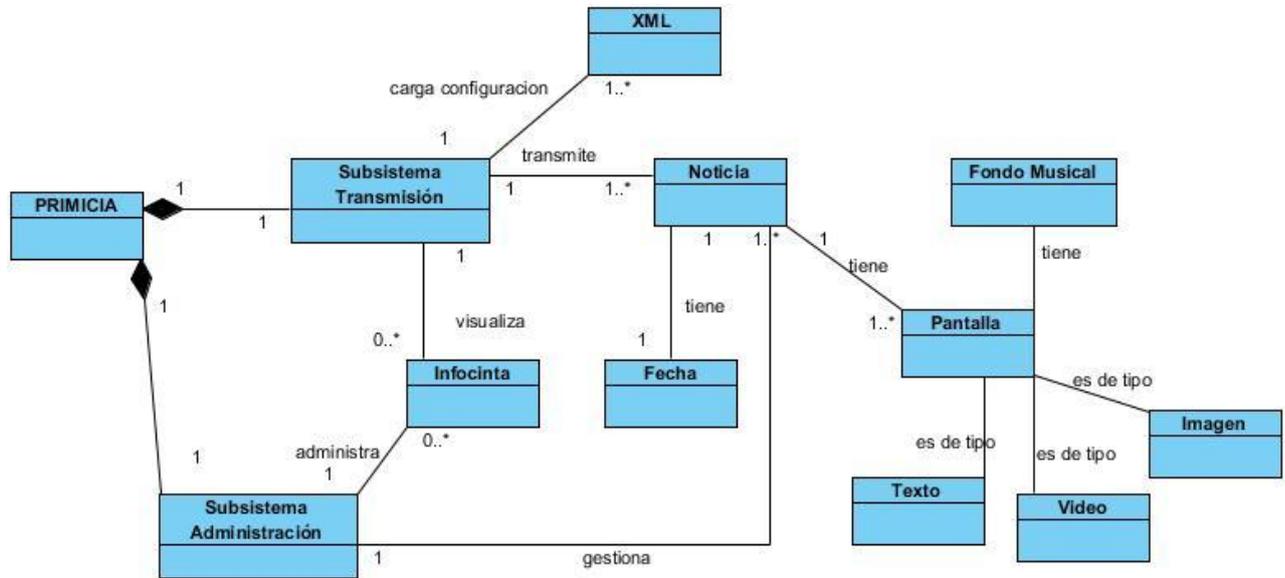


Figura 2: Modelo de dominio para el componente de presentación de informaciones climáticas.

2.3 Especificación de los requisitos del software

El profesor Ian Sommerville define un requisito de software como: “Es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste. En el otro extremo, es una definición detallada y formal de una función del sistema” (EcuRed, 2012).

Por otra parte, según la metodología RUP un requisito de software está definido como:

“Especificación de un comportamiento del sistema observable externamente; por ejemplo, entradas, salidas, funciones y atributos, o atributos del entorno del sistema” (IBM_Corporation, 2011).

Esta definición brindada por RUP plantea, en otras palabras, que los requisitos de software son aquellas propiedades, acciones o restricciones fundamentales que debe satisfacer un producto de software al recibir información, procesarla y brindar un determinado resultado. Para que un sistema sea exitoso deberá de cumplir con cada uno de los requisitos definidos por el cliente, estos constituyen obligaciones o exigencias con las que el sistema debe cumplir y que brindan la base sobre la cual se definirá el alcance y el objetivo del producto final. Los requisitos se dividen en dos tipos, funcionales y no funcionales, según los fines sobre los cuales estén definidos. A continuación se muestran las

Capítulo 2: Análisis, diseño e implementación del Componente Climático

acciones que el sistema debe ser capaz de realizar, mediante los requisitos funcionales y no funcionales planteados para la presentación de las informaciones climáticas.

2.3.1 Requisitos funcionales

Los requisitos funcionales son declaraciones de servicios que el sistema debe proporcionar, define la manera en que éste debe reaccionar a determinadas entradas y cómo se debe comportar en situaciones particulares” (Sommerville, 2004).

Estos requisitos son las características fundamentales del sistema y expresan la capacidad de acción del mismo. Son los que definen lo que el sistema debe realizar para dar solución a lo que el usuario necesita, constituyendo las características propias de cada sistema y las condiciones necesarias que se deben de cumplir para que el producto final tenga la calidad requerida y cumpla con los objetivos propuestos. A continuación se muestran los requisitos funcionales definidos para el componente de presentación de informaciones climáticas:

RF1 Presentar información referente al pronóstico meteorológico en la pantalla: El componente debe brindar la posibilidad de mostrar la información a través de diversas vistas en correspondencia con la relevancia de la información, el interés informativo o la disponibilidad de espacio en la pantalla.

Criterios de entrada: El subsistema de transmisión debe definir el tipo de vista mediante la cual se presentará la información.

Criterios de salida: Presentación de la información según la vista definida por el subsistema de transmisión.

Prioridad: Alta.

Costo de desarrollo: Alto.

RF1.1 Presentar información referente al pronóstico meteorológico a través de vista rápida.

RF1.2 Presentar información referente al pronóstico meteorológico a través de vista en pantalla completa.

RF1.3 Presentar información referente al pronóstico meteorológico a través de vista en forma de infocinta.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

RF2 Visualizar contenidos digitales mediante texto e imagen: El componente debe ser capaz de mostrar información a través de textos, imágenes o la combinación de ambos.

Criterios de entrada: Determinar según el tipo de visualización que formatos serán utilizados para mostrar la información.

Criterios de Salida: Se muestra la información meteorológica utilizando los formatos de texto e imagen.

Prioridad: Alta.

Costo de desarrollo: Medio.

RF3 Mostrar fecha u horario del pronóstico meteorológico: El componente debe ser capaz de mostrar la fecha o el horario sobre el cual se estará visualizando el pronóstico meteorológico.

Criterios de entrada: Adquirir desde el INSMET⁸ la fecha u horario del cual se mostrará el pronóstico meteorológico.

Criterios de Salida: Se muestra en la pantalla la fecha u horario para el cual se brindará el parte meteorológico.

Prioridad: Alta.

Costo de desarrollo: Bajo.

RF4 Mostrar nombre o identificador de la región correspondiente al pronóstico meteorológico: El componente debe ser capaz de mostrar el nombre o el identificador de la región sobre la cual se estará visualizando el pronóstico meteorológico.

Criterios de entrada: Adquirir el nombre o identificador de la región brindado por el INSMET sobre la cual se mostrará el pronóstico meteorológico.

Criterios de Salida: Se muestra en la pantalla el nombre o identificador de la región para la cual se brindará el parte meteorológico.

Prioridad: Alta.

Costo de desarrollo: Bajo.

⁸ Instituto de Meteorología.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

RF5 Mostrar información relacionada con las temperaturas del pronóstico meteorológico: El componente debe ser capaz de mostrar información referente a las temperaturas (mínimas y máximas) según la región y el horario definido, independientemente del sistema de medidas empleado.

Criterios de entrada: Adquirir la información de las temperaturas brindadas por el INSMET, en correspondencia con la fecha y la región de la cual se visualizará el pronóstico del tiempo.

Criterios de Salida: Se muestran los valores de las temperaturas mínimas y máximas en la pantalla.

Prioridad: Alta.

Costo de desarrollo: Bajo.

RF6 Mostrar información relacionada con los vientos del pronóstico meteorológico: El componente debe ser capaz de mostrar información referente a la velocidad y dirección de los vientos según la región y el horario definido, independientemente del sistema de medidas empleado.

Criterios de entrada: Adquirir desde el INSMET la información de los vientos.

Criterios de Salida: Se muestran en pantalla el valor de la velocidad y dirección de los vientos, correspondientes a la fecha y a la región que se está visualizando.

Prioridad: Alta.

Costo de desarrollo: Bajo.

RF7 Mostrar el estado del tiempo correspondiente al pronóstico meteorológico (nubosidad): El componente debe ser capaz de mostrar el estado del tiempo correspondiente a la región y el horario definido.

Criterios de entrada: Adquirir la información brindada por el INSMET correspondiente al estado del tiempo.

Criterios de Salida: Se muestran las características del estado del tiempo en la pantalla.

Prioridad: Alta.

Costo de desarrollo: Bajo.

RF8 Mostrar la fase de la luna correspondiente al pronóstico meteorológico: El componente debe ser capaz de mostrar la fase de la luna según la fecha del pronóstico meteorológico.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

Criterios de entrada: Determinar según la fecha adquirida del pronóstico meteorológico la fase de la luna correspondiente.

Criterios de Salida: Se muestra la fase de la luna en pantalla.

Prioridad: Alta.

Costo de desarrollo: Bajo.

RF9 Mostrar información relacionada con el oleaje del pronóstico meteorológico: El componente debe ser capaz de mostrar información referente al oleaje según la región y el horario definido.

Criterios de entrada: Adquirir la información sobre el oleaje brindada por el INSMET para la fecha y región del pronóstico.

Criterios de Salida: Se muestran las características del oleaje en la pantalla.

Prioridad: Alta.

Costo de desarrollo: Bajo.

2.3.2 Requisitos no funcionales

Los requisitos no funcionales son las restricciones de los servicios o funciones ofrecidas por el sistema, restringen el espacio de posibles soluciones (Sommerville, 2004).

Estos requisitos son los que se centran fundamentalmente en las características del sistema y no en los rasgos particulares del mismo, es decir, se basan en la carga que se espera tenga que soportar el sistema. Son características adicionales con las cuales debe cumplir el producto y corresponden a variados aspectos tales como: usabilidad, disponibilidad, seguridad, hardware, apariencia y confiabilidad. A continuación se describen los requisitos no funcionales del sistema:

Requisitos de apariencia o interfaz externa: La apariencia del componente debe estar restringida por las mismas condiciones que el producto PRIMICIA, haciendo uso de colores apropiados que denoten un producto profesional. Además deberá de contar con una interfaz intuitiva que le muestre a los usuarios de forma sencilla y clara las características de cada variable meteorológica.

Restricciones en el diseño y la implementación: Para la modelación del producto se utilizará el lenguaje UML y Visual Paradigm como herramienta CASE. La arquitectura debe soportar la migración

Capítulo 2: Análisis, diseño e implementación del Componente Climático

de la interfaz de usuario de forma rápida, para lograr visualizar cualquiera de los cambios que se produzcan.

Requisitos de Disponibilidad: El componente que visualiza las noticias climáticas deberá estar operando los 7 días de la semana y las 24 horas del día, garantizando el buen funcionamiento del servicio prestado por el mismo. Se debe de tener presente que el mismo puede no estar accesible por un error de hardware, de software o de conectividad, ya que el mismo muestra los datos reales que se toman del Instituto de Meteorología a través de la red.

Requisitos de Usabilidad: Se debe mostrar la información de forma lógica y correctamente estructurada. El nuevo componente debe permitir visualizar de diversas formas la información deseada, según las necesidades de la entidad donde se implante.

Requisitos Legales: El producto estará sujeto a las mismas restricciones legales que el resto de la Plataforma de Televisión Informativa, PRIMICIA. Los derechos de autor y otros están determinados por la entidad desarrolladora: UCI.

Requisitos de Hardware: Los requisitos estarán en correspondencia con los utilizados por el subsistema de transmisión de PRIMICIA al que pertenece el componente.

Requisitos de hardware	
Servidor	Transmisión
Procesador	Dual-Core Xeon 1.60 GHz
Memoria RAM	1 Gb
Disco Duro	80 Gb
Tarjeta de Red	Ethernet Gigabit

Tabla 1: Requisitos de Hardware para un servidor.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

2.4 Descripción de la solución propuesta

2.4.1. Definición de los actores del sistema

Los actores del sistema constituyen cada tipo de fenómeno externo con el que debe interactuar el sistema. Para lograr definir de manera acertada los actores es necesario redactar una breve descripción que incluye las responsabilidades que posee el mismo y para qué necesita el actor el sistema. A continuación se muestra una breve descripción del actor determinado:

Sistema: Encargado de visualizar toda la información meteorológica a partir de los datos adquiridos del Instituto de Meteorología y que en su unión conforman el pronóstico meteorológico.

2.4.2. Diagrama de casos de uso del sistema

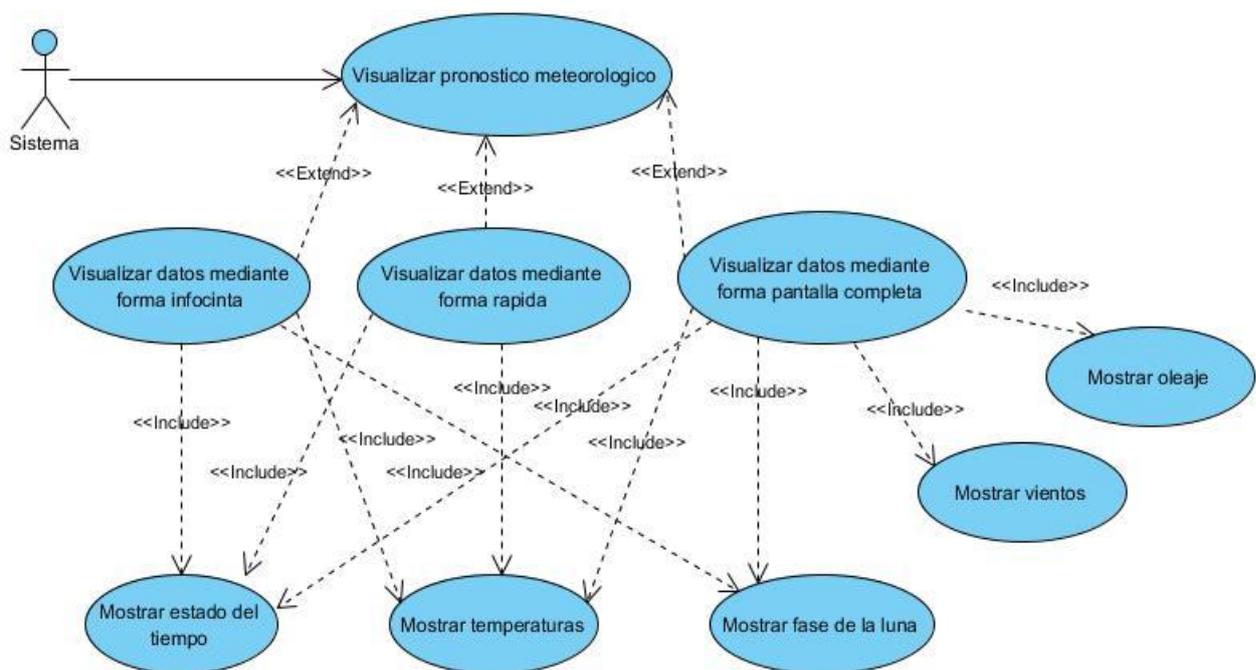


Figura 3: Diagrama de Casos de Uso del componente de presentación de informaciones climáticas.

2.4.3. Descripción de los casos de uso del sistema

Caso de Uso del sistema: Visualizar pronóstico meteorológico.

Caso de Uso:	Visualizar pronóstico meteorológico.
--------------	--------------------------------------

Capítulo 2: Análisis, diseño e implementación del Componente Climático

Actores:	Sistema
Resumen:	El sistema adquiere los datos meteorológicos brindados por el Instituto de meteorología y visualiza el pronóstico por horarios y regiones.
Referencias:	RF1, RF1.1, RF1.2, RF1.3, RF2, RF3, RF4
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<p>1. Luego de haber inicializado el componente de presentación de informaciones climáticas, se leen y salvan los datos meteorológicos brindados por el INSMET en el XML.</p> <p>2. El actor identifica el tipo de visualización mediante la cual se mostrará la información del pronóstico meteorológico.</p>	<p>3. Se muestra en pantalla la información del pronóstico meteorológico en dependencia del tipo de visualización definida:</p>
	<p>a) Vista Rápida. Ir a la descripción del caso de uso Visualizar datos mediante forma vista rápida.</p>
	<p>b) Vista en forma de infocinta o cintillo informativo. Ir a la descripción del caso de uso Visualizar datos mediante forma infocinta.</p>
	<p>c) Vista en pantalla completa. Ir a la descripción del caso de uso Visualizar datos mediante forma pantalla completa.</p>

Tabla 2: Descripción textual del CUS- Visualizar pronóstico meteorológico.

Caso de Uso del sistema: Visualizar datos mediante forma rápida.

Caso de Uso:	Visualizar datos mediante forma rápida.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando el sistema determina que el tipo de

Capítulo 2: Análisis, diseño e implementación del Componente Climático

	visualización a utilizar será vista rápida y adquiere los datos del pronóstico desde el INSMET.
Referencias:	RF1, RF1.1
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Se identifica que el tipo de visualización a utilizar será vista rápida. 2. Se leen los datos adquiridos del pronóstico meteorológico. 	<ol style="list-style-type: none"> 3. Se visualizan en pantalla la fecha, el nombre o identificador de la región y los datos de las temperaturas y del estado del tiempo correspondientes al pronóstico meteorológico.

Tabla 3: Descripción textual del CUS- Visualizar datos mediante forma rápida.

Caso de Uso del sistema: Visualizar datos mediante forma infocinta.

Caso de Uso:	Visualizar datos mediante forma infocinta.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando el sistema determina que el tipo de visualización a utilizar será vista en forma de infocinta y adquiere los datos del pronóstico desde el INSMET.
Referencias:	RF1, RF1.3
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Se identifica que el tipo de visualización a utilizar será vista en forma de infocinta. 	<ol style="list-style-type: none"> 3. Se visualizan en pantalla la fecha, el nombre o identificador de la región y los datos de las temperaturas, estado del tiempo y fase de la luna correspondientes al pronóstico

Capítulo 2: Análisis, diseño e implementación del Componente Climático

2. Se leen los datos adquiridos del pronóstico meteorológico.	meteorológico.
---	----------------

Tabla 4: Descripción textual del CUS- Visualizar datos mediante forma de infocinta.

Caso de Uso del sistema: Visualizar datos mediante forma pantalla completa.

Caso de Uso:	Visualizar datos mediante forma pantalla completa.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando el sistema determina que el tipo de visualización a utilizar será vista en forma de pantalla completa y adquiere los datos del pronóstico desde el INSMET.
Referencias:	RF1, RF1.2
Prioridad:	Crítico

Flujo Normal de Eventos

Acción del Actor	Respuesta del Sistema
1. Se identifica que el tipo de visualización a utilizar será vista en forma de pantalla completa. 2. Se leen los datos adquiridos del pronóstico meteorológico.	3. Se visualizan en pantalla la fecha, el nombre o identificador de la región y los datos de las temperaturas, vientos, estado del tiempo, fase de la luna y oleaje correspondientes al pronóstico meteorológico.

Tabla 5: Descripción textual del CUS- Visualizar datos mediante forma pantalla completa.

Caso de Uso del sistema: Mostrar temperaturas.

Caso de Uso:	Mostrar temperaturas.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando el sistema adquiere la información de

Capítulo 2: Análisis, diseño e implementación del Componente Climático

	las temperaturas brindadas por el Instituto de Meteorología.
Referencias:	RF5
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Se adquieren desde el INSMET los datos sobre las temperaturas mínimas y máximas, para confeccionar el pronóstico meteorológico que se visualizará.	2. Se muestran en la pantalla los valores de las temperaturas mínimas y máximas que se encuentran almacenados en el XML, definidas según el horario y la región de la cual se visualiza el pronóstico del tiempo.

Tabla 6: Descripción textual del CUS- Mostrar temperaturas.

Caso de Uso del sistema: Mostrar vientos.

Caso de Uso:	Mostrar vientos.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando el sistema adquiere la información de los vientos brindada por el Instituto de Meteorología.
Referencias:	RF6
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El usuario adquiere desde el INSMET los datos de la velocidad y la dirección de los vientos que	2. Se muestran en pantalla los valores de la velocidad y la dirección de los vientos que se encuentran almacenados en el XML, según el horario y la región de la cual se visualiza el pronóstico meteorológico.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

confeccionarán el pronóstico meteorológico.	
--	--

Tabla 7: Descripción textual del CUS- Mostrar vientos.

Caso de Uso del sistema: Mostrar estado del tiempo.

Caso de Uso:	Mostrar estado del tiempo.	
Actores:	Sistema	
Resumen:	El caso de uso se inicializa cuando se adquiere la información referente al estado del tiempo brindado por el Instituto de Meteorología.	
Referencias:	RF7	
Prioridad:	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El sistema adquiere los datos del estado del tiempo desde el INSMET para confeccionar el pronóstico meteorológico.	2. Se muestran los valores del estado del tiempo en la pantalla, adquiridos desde el XML, definidos según el horario y la región de la cual se visualiza el pronóstico.	

Tabla 8: Descripción textual del CUS- Mostrar estado del tiempo.

Caso de Uso del sistema: Mostrar fase de la luna.

Caso de Uso:	Mostrar fase de la luna.	
Actores:	Sistema	
Resumen:	El caso de uso se inicializa cuando el sistema determina la fase de la luna para la fecha del pronóstico meteorológico.	
Referencias:	RF8	
Prioridad:	Crítico	

Capítulo 2: Análisis, diseño e implementación del Componente Climático

Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. A partir de la fecha del pronóstico meteorológico el sistema determina la fase de la luna.	2. Se muestra la fase de la luna en la pantalla, en correspondencia con el horario sobre el cual se visualiza el pronóstico del tiempo.

Tabla 9: Descripción textual del CUS- Mostrar fase de la luna.

Caso de Uso del sistema: Mostrar oleaje.

Caso de Uso:	Mostrar oleaje.
Actores:	Sistema
Resumen:	El caso de uso se inicializa cuando se adquiere la información referente al oleaje brindada por el Instituto de Meteorología.
Referencias:	RF9
Prioridad:	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El sistema adquiere los datos referentes al oleaje desde el INSMET.	2. Se muestran los datos del oleaje para el horario de la cual se visualiza el pronóstico meteorológico.

Tabla 10: Descripción textual del CUS- Mostrar oleaje.

2.5 Modelo de análisis

El modelo de análisis constituye una representación abstracta del sistema que se está modelando. Es una herramienta mediante la cual se permite tener un mejor control del problema a resolver ya que se encarga de modelar la solución existente para lograr obtener resultados adecuados en la construcción del producto, representando la lógica general del sistema sin entrar en detalles específicos.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

Este tipo de modelo se encuentra constituido por clases del análisis, las cuales se centran en los requisitos funcionales planteados, en la asociación de estas con el contexto del problema y además se encargan de describir la realización de los casos de uso, en RUP estas clases se clasifican en: Interfaz, Entidad y Control. Además se observan en estas: atributos y posibles relaciones a establecerse entre ellas.

2.5.1. Diagrama de clases del análisis

Los diagramas de clases del análisis permiten identificar y mostrar las clases que se utilizan en el sistema, así como las relaciones, funcionalidades y atributos existentes en las mismas. A continuación se presentan los diagramas de clases del análisis realizadas en el desarrollo de este trabajo de diploma para los casos de uso del sistema.

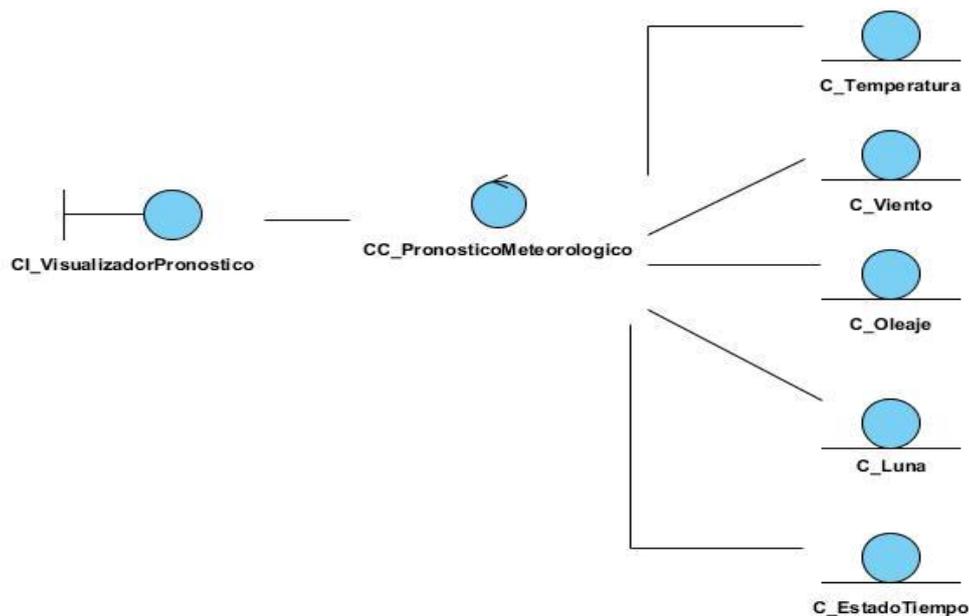


Figura 4: Diagrama de clases del análisis - CU Visualizar pronóstico meteorológico.

2.6 Arquitectura de software

La arquitectura de software es la estructura de las estructuras del sistema, la cual comprende los componentes de software, las propiedades de esos componentes visibles externamente, y las relaciones entre ellos (Pressman, 5ta Edición).

Capítulo 2: Análisis, diseño e implementación del Componente Climático

La definición de la arquitectura de software constituye un paso fundamental a tener en cuenta y a desarrollar de la mejor forma posible durante el proceso de desarrollo del producto, es la encargada de que se produzca una correcta comunicación entre todas las partes involucradas y de proporcionar una visión global del sistema a construir.

2.6.1. Patrones de arquitectura: Arquitectura en capas

La arquitectura en capas soporta un diseño basado en niveles de abstracción crecientes, lo cual a su vez permite la partición de un problema complejo en una secuencia de pasos incrementales. Brinda una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente inferior. Este estilo admite optimizaciones y refinamientos, además de proporcionar una amplia reutilización (Reynoso, y otros, 2004).

Arquitectura en tres capas

Para el desarrollo del componente se definió la arquitectura en tres capas, denominadas como: capa presentación, capa lógica de negocio y capa acceso a datos. La capa de presentación es la encargada de contener la interfaz del sistema y tiene el objetivo de mostrar a los usuarios la información meteorológica de forma agradable y clara, la capa de lógica de negocio es la que se encarga de extraer los datos del pronóstico del tiempo almacenados en la capa inferior y necesarios para realizar el procesamiento que permitirá visualizar la información en la pantalla, por otra parte la capa de acceso a datos es la encargada de adquirir y almacenar la información meteorológica brindada por el INSMET y que posteriormente será utilizada por la capa lógica de negocio.

Se escogió este tipo de arquitectura pues con la misma se reduce las dependencias existentes entre las clases, de forma tal que las capas más bajas no son conscientes de ningún detalle o interfaz de las superiores. Brindando la posibilidad de mejorar el soporte del sistema y de poseer una mejor organización durante el desarrollo del software, además cada nivel tiene funcionalidades diferentes lo que permite el diseño de una arquitectura escalable que puede ser ampliada en caso de ser necesario.

2.7 Modelo de diseño

En esta fase se expande y detalla los modelos de análisis realizados, tomando en cuenta todas las implicaciones y restricciones técnicas. El propósito del diseño es especificar una solución que trabaje y pueda ser fácilmente convertida en código fuente y construir una arquitectura simple y fácilmente extensible (Navarro).

Capítulo 2: *Análisis, diseño e implementación del Componente Climático*

Este tipo de modelo constituye la base lógica o punto de partida para lograr realizar la implementación del sistema y para concebir y documentar el diseño del software. Debe de mantenerse durante todo el ciclo de vida del software ya que se encarga de darle forma al producto, por lo que es necesario tener en cuenta los requisitos, características brindadas por el lenguaje de programación C++ y por el *framework* Qt utilizado.

2.7.1. Patrones de diseño

En la terminología de objetos, el patrón es una descripción de un problema y la solución asociada a este, que recibe un nombre y que puede emplearse en otros contextos. Muchos patrones ofrecen orientación sobre cómo asignar las responsabilidades a los objetos ante determinada categoría de problemas. Los patrones no se proponen descubrir ni expresar nuevos principios de la ingeniería del software sino que intentan codificar el conocimiento, las expresiones y los principios ya existentes (Larman, 2da Edición).

Como resultado del trabajo de diseñadores expertos, existen patrones que engloban todo un cúmulo de expresiones y principios que permiten guiar a futuras generaciones en el proceso de creación de un determinado producto con la calidad requerida. Es por esto que la existencia de los patrones de diseño posee mucha importancia en el desarrollo de software, a partir de estos las clases y objetos que se comunican entre sí son adaptados para resolver un problema de diseño general en un contexto particular, además de permitir la forma de ser aplicados en situaciones diversas u otros contextos.

Los patrones de diseño se dividen en dos grupos: GRASP⁹ y GOF¹⁰. Los patrones GRASP son los encargados de guiar los principios fundamentales de la asignación de responsabilidades en el diseño orientado a objetos y los patrones GOF, pandilla de los cuatro, se clasifican en dependencia del propósito con el cual van a ser utilizados en tres categorías: creacionales, estructurales y de comportamiento. Durante el desarrollo de este trabajo de diploma fueron utilizados los siguientes patrones:

Experto: es el encargado de asignar una responsabilidad al experto en información, constituyendo este la que clase que cuenta con la información necesaria para cubrir la responsabilidad. En este trabajo este tipo de patrón fue utilizado a la hora determinar las responsabilidades con las cuales debía de cumplir cada clase, las funcionalidades que realizarían cada una de ellas. Teniendo como principal

⁹ *General Responsibility Assignment Software Patterns.*

¹⁰ *Gang of Four.*

Capítulo 2: Análisis, diseño e implementación del Componente Climático

objetivo permitir un mejor entendimiento de cómo funciona la aplicación para facilitar su mantenimiento y ampliación.

Creador: es quien guía el proceso de asignación de responsabilidades relacionadas con la creación de objetos. Tiene el objetivo de asignarle a la clase B la responsabilidad de crear una instancia de la clase A, la intención de este patrón es encontrar un creador que necesite conectarse al objeto creado en alguna situación. Fue utilizado durante la construcción del componente cuando en una clase determinada se necesito crear una instancia de otra clase, por ejemplo en la clase CC_PronosticoMeteorologico se utiliza este patrón de forma reiterada cuando se crean objetos de clases como: C_Temperatura, C_EstadoTiempo, C_Luna, C_Viento, entre otras.

Controlador: es el responsable de asignar a clases específicas la responsabilidad de controlar el flujo de eventos en el sistema, facilitando la centralización de las actividades desarrolladas y permitiendo a través de esta interactuar con las demás clases del componente. La utilización de este patrón se evidencia en la clase CC_PronosticoMeteorologico.

Layers: este patrón fue utilizado con el objetivo de organizar en capas el modelo de diseño, los componentes de una capa solo pueden hacer referencia a componentes de capa inmediatamente inferiores. En la aplicación se evidencian las tres capas utilizadas para delimitar la presentación, el negocio y los datos.

2.7.2. Diagrama de clases del diseño

Los diagramas de clases de diseño resumen la definición de las clases e interfaces implementables en software (Larman, 2da Edición).

En este diagrama se muestra de forma más concreta los datos reflejados en los diagramas de clases del análisis de cada uno de los casos de uso planteados, constituyendo así una mayor aproximación del modelo de la solución final a implementar. Se representa la parte estática, las clases y las relaciones existentes entre estas, así como los atributos, métodos y dependencias. A continuación se muestran los diagramas de clases del diseño desarrollados para cada uno de los casos de usos definidos en el transcurso de esta investigación.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

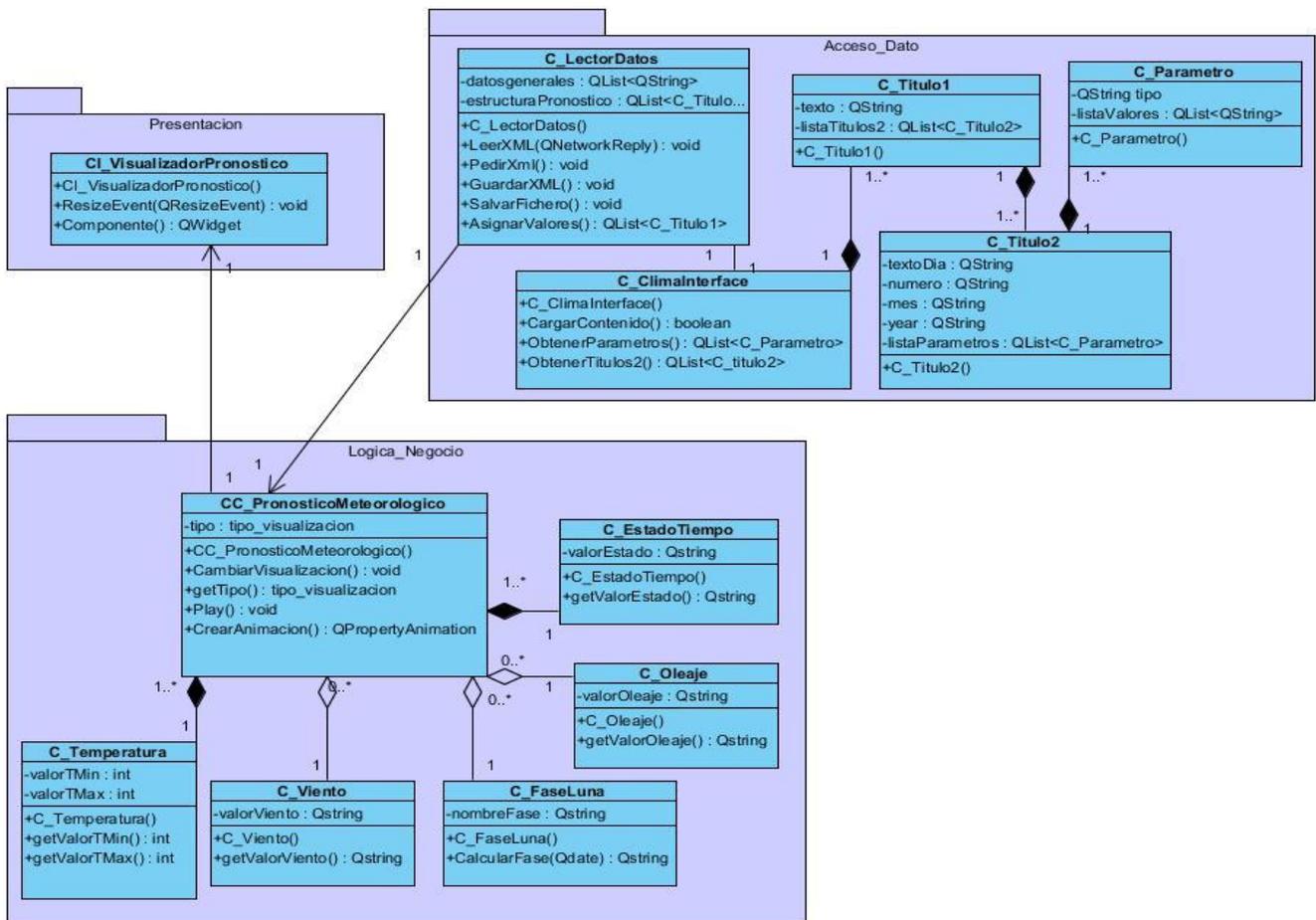


Figura 5: Diagrama de clases del diseño-CU Visualizar pronóstico meteorológico.

2.8 Modelo de despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo y tiene como objetivo capturar la descomposición del sistema en elementos que alojan el proceso (Jacobson, y otros, 2000) (IBM_Corporation, 2011).

Este diagrama permite realizar un esquema visual de la distribución física de los nodos y de los ficheros utilizados por estos en el sistema, muestra la tecnología que se necesita para lograr satisfacer los requerimientos no funcionales de hardware definidos y el correcto funcionamiento de la aplicación. A continuación se presenta el diagrama de despliegue que muestra la aplicación que se desarrolla.

Capítulo 2: *Análisis, diseño e implementación del Componente Climático*



Figura 6: Modelo de despliegue del componente de presentación de informaciones climáticas.

2.9 Diagrama de componentes

Un componente es una parte del sistema sustituible, casi independiente e importante que desempeña una función clara en el contexto de una arquitectura bien definida (Sommerville, 2004).

Los componentes pueden ser ficheros, ejecutables, librerías, tablas de la base de datos o documentos. En este tipo de diagramas intervienen también los paquetes, que son una forma de organización o agrupamiento de diferentes archivos o clases que el desarrollador debe usar (Leyva, 2011). Los componentes pueden ser tratados de forma independiente como funciones, procedimientos o subrutinas.

En los diagramas de componentes que se muestran a continuación y que fueron confeccionados durante el desarrollo de este trabajo, se visualiza la relación física existente entre los componentes considerados necesarios para la programación de los casos de uso definidos con anterioridad. Además son utilizados con el fin de estructurar y controlar los componentes en los sistemas de software, fundamentalmente mostrando las dependencias existentes entre estos.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

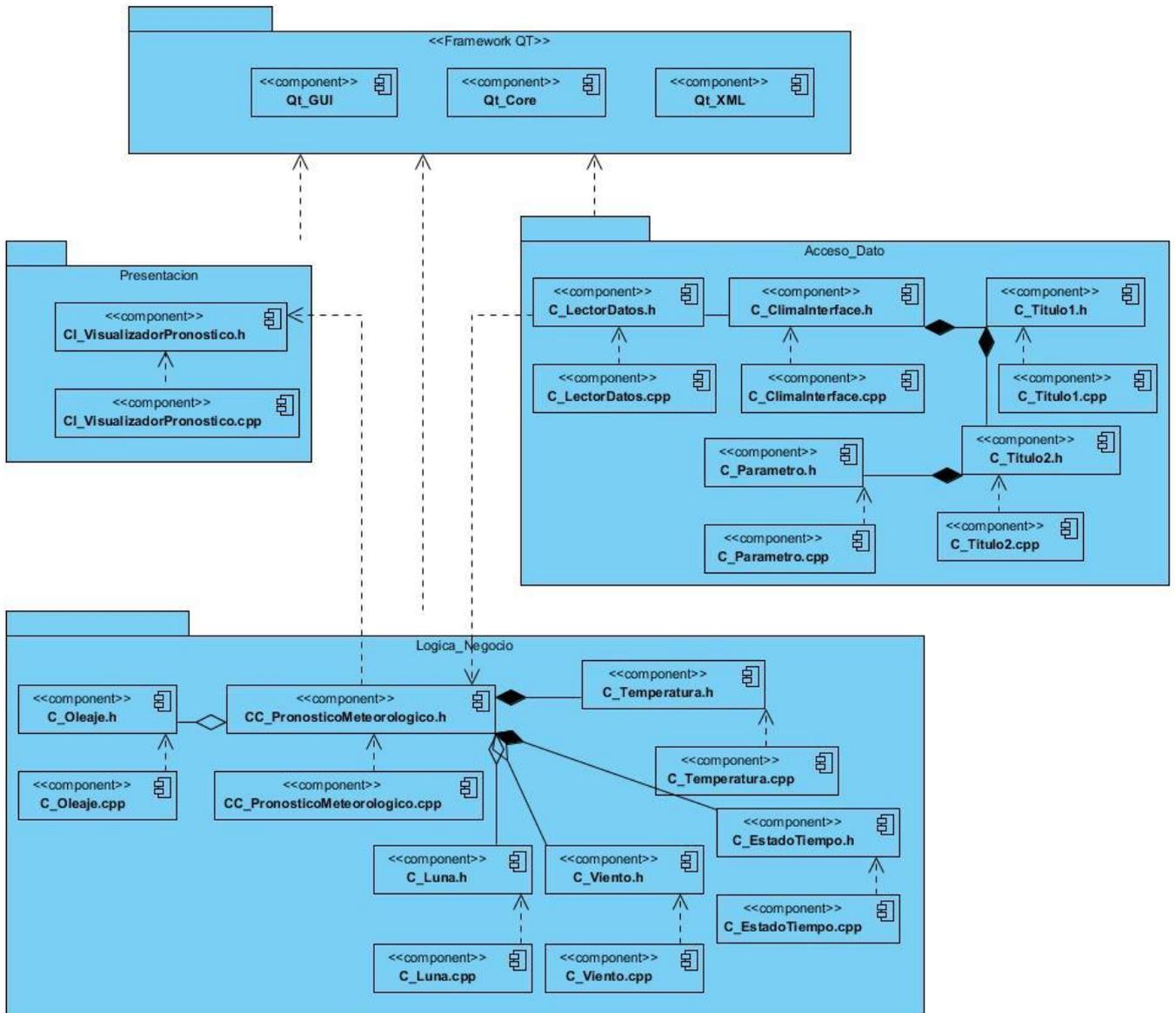


Figura 7: Diagrama de componente- CU Visualizar pronóstico meteorológico.

2.10 Estándar de codificación

Un estándar de codificación es un conjunto de reglas de notación y nomenclatura, específicas de cada lenguaje de programación, que se usan y se siguen durante la fase de implementación (codificación) de una aplicación y reducen perceptiblemente el riesgo de que los desarrolladores introduzcan errores que no son detectados por los compiladores, reduciendo el tiempo y coste de las actividades de depuración y pruebas necesarias para la detección y corrección de los mismos (Morales, y otros, 2007).

Capítulo 2: *Análisis, diseño e implementación del Componente Climático*

La utilización de un estándar de codificación constituye una buena práctica de programación que reduce considerablemente la posibilidad de cometer errores, se obtiene como producto final una aplicación con código legible y comprensible por otras personas del equipo de desarrollo. Para la elaboración del estándar de codificación utilizado se tuvo en cuenta las características del *framework* de desarrollo Qt, el entorno integrado de desarrollo Qt Creator y el lenguaje de programación C++. Además se tuvieron en cuenta algunas de las reglas planteadas por el centro de desarrollo GEYSED. A continuación se muestran algunas de las características con las cuales debe de cumplir la implementación del componente:

1. Los nombres de cada uno de los elementos del programa deben ser significativos.
2. No manejar en los programas más de una instrucción por línea.
3. Declarar las variables en líneas separadas.
4. Añadir comentarios descriptivos junto a cada declaración de variable, si es necesario.
5. La mayoría de los elementos se deben nombrar usando sustantivos.
6. Los atributos deben comenzar con letra minúsculas.
7. Los métodos deben comenzar con letra mayúsculas.
8. Para nombrar las clases, interfaces y archivos se deberá de utilizar un sustantivo singular, con la primera letra en mayúscula y las demás en minúsculas.
9. El nombre de las clases estará antecedido por las variables C, CC o CI en dependencia del tipo de clase que sea y separado este prefijo por un guión bajo. Para las clases controladoras será utilizado el prefijo CC, para la clase interfaz el prefijo CI y para las demás clases el prefijo C. Ejemplo: CC_PronosticoMeteorologico, CI_WidgetPersonalizado, C_Temperatura.
10. Si el nombre de la clase está compuesto por más de una palabra la primera letra de cada una debe de ser mayúscula. Letras mayúsculas seguidas no son permitidas. Ejemplo: "C_EstadoTIEMPO" no es permitido mientras que "C_EstadoTiempo" si lo es.
11. Para separar palabras se usará la diferenciación entre mayúsculas y minúsculas.

Capítulo 2: Análisis, diseño e implementación del Componente Climático

12. Para los atributos que representan listas, el nombre de este estará constituido en su primera parte por la palabra lista seguida por un nombre sugerente que se le desee asignar al mismo. Ejemplo: listaValores.
13. Las funciones implementadas llevarán comentarios describiendo el objetivo de la función, si se considera necesario, pero no la descripción del funcionamiento.

2.11 Conclusiones

A partir de la utilización de las herramientas y la metodología escogida en el capítulo anterior en el presente se dio paso a la presentación de la solución propuesta. En el mismo se realizó un mayor acercamiento y comprensión del objetivo general propuesto en la investigación, a partir de la identificación de los principales eventos y conceptos del entorno y de la realización del modelo de dominio. Además se definió a partir de los requisitos de software las características con las cuales debía cumplir la aplicación a construir y se tradujeron las mismas a interacciones producidas con el sistema mediante el diagrama de casos de uso. Se muestran en este capítulo además los diagramas de clases del análisis y del diseño y los diagramas de componentes realizados, los cuales constituyeron la base para la implementación del componente a partir de la utilización del estándar de codificación definido.

Capítulo 3: Validación y pruebas realizadas al Componente Climático

3.1. Introducción

El hecho de que el software sea desarrollado por personas condiciona la posibilidad de que el mismo tenga errores. Una buena práctica constituye planificar tareas que estén relacionadas con la garantía de la calidad en todas las fases del flujo de trabajo. Las pruebas de software son el proceso de identificar las fallas o defectos en el sistema con el objetivo de asegurarse de que el mismo cumple con los requisitos del cliente antes de lanzarlo al mercado. En este capítulo se brindan conceptos relacionados con la validación del software y se recogen los resultados obtenidos luego de realizar este proceso a la solución propuesta, evidenciándose en la aplicación de diferentes técnicas de prueba que permiten comprobar la eficiencia del Componente de presentación de informaciones climáticas, para dar respuesta a las necesidades de los clientes.

3.2. Pruebas de software

Las pruebas de software son un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. Las pruebas constituyen el último bastión desde el que se puede evaluar la calidad y, de forma más pragmática, descubrir los errores. Pero las pruebas no deben ser vistas como una red de seguridad (Pressman, 5ta Edición).

El único instrumento adecuado para determinar el status de la calidad de un producto software es el proceso de pruebas. En este proceso se ejecutan pruebas dirigidas a componentes del software o al sistema de software en su totalidad, con el objetivo de medir el grado en que el software cumple con los requerimientos (Pruebas_de_Software, 2005).

Se le realizaron varias pruebas al Componente de presentación de informaciones climáticas con el fin de lograr un producto de alta calidad, aportando con estas la posibilidad de verificar que el sistema está de acuerdo con los requisitos especificados y además se facilita con el uso de las mismas la realización del mantenimiento al software desarrollado.

3.2.1. Niveles de prueba

Existen varios tipos de pruebas posibles a realizar entre las que se encuentran: las unitarias, las de integración, las de sistema y las de aceptación. Las pruebas unitarias son las que tienen como objetivo aislar cada parte del programa y mostrar que las partes individuales son correctas. En las pruebas de integración es donde los módulos individuales de software son combinados y probados como un grupo. Por otra parte las pruebas del sistema tienen como objetivo verificar el sistema software para comprobar si este cumple sus requisitos, dentro de esta fase pueden desarrollarse distintos tipos de pruebas en función de los objetivos de las mismas como son: pruebas funcionales, pruebas de usabilidad, pruebas de rendimiento y pruebas de seguridad. También pueden realizarse las pruebas de aceptación las cuales fueron realizadas para validar el desarrollo del componente de presentación de informaciones climáticas.

3.2.1.1. Pruebas de aceptación

El objetivo de las pruebas de aceptación es validar que un sistema cumple con el funcionamiento esperado y permitir al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento (Ministerio de Administraciones Públicas, 2002).

Las pruebas de aceptación son definidas por el usuario del sistema y preparadas por el equipo de desarrollo, aunque la ejecución y aprobación final corresponden al usuario.

La validación del sistema se consigue mediante la realización de pruebas de caja negra que demuestran la conformidad con los requisitos y que se recogen en el plan de pruebas, el cual define las verificaciones a realizar y los casos de prueba asociados.

3.2.2. Tipos de pruebas

3.2.2.1. Pruebas de caja negra

La prueba de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz de software. Examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del software. Las pruebas de caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software. La prueba de caja negra no es

una alternativa a las técnicas de prueba de caja blanca. Más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que los métodos de caja blanca (Pressman, 5ta Edición).

3.2.2.2. Pruebas de caja blanca

La prueba de caja blanca del software se basa en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de pruebas que ejerciten conjuntos específicos de condiciones y/o bucles. La prueba de caja blanca, denominada a veces prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para obtener los casos de prueba (Pressman, 5ta Edición).

La prueba del camino básico es una técnica de prueba de caja blanca propuesta inicialmente por Tom McCabe. El método del camino básico permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa (Pressman, 5ta Edición).

Los pasos a seguir para la realización de las pruebas de caja blanca son los siguientes:

1. Generar grafo de flujos de datos.
2. Cálculo de complejidad ciclomática, $V(G)$.

$$V(G) = NA \text{ (Número de Aristas)} - NN \text{ (Número de Nodos)} + 2.$$

$$V(G) = P \text{ (Nodos predicados)} + 1.$$

$$V(G) = \text{Número de regiones}.$$

La complejidad ciclomática es una métrica del software que proporciona una medición cuantitativa de la complejidad lógica de un programa. Cuando se usa en el contexto del método de prueba del camino básico, el valor calculado como complejidad ciclomática define el número de caminos independientes del conjunto básico de un programa y da un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecuta cada sentencia al menos una vez (Pressman, 5ta Edición).

3. Determinar los caminos independientes o básicos.
4. Generar un caso de prueba para cada camino de ejecución.

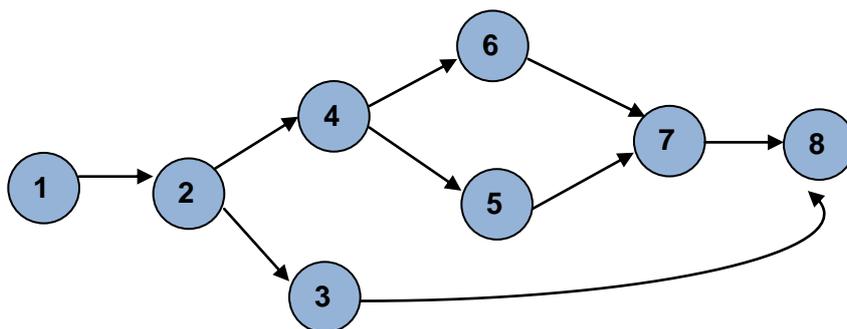
Además existen otros tipos de pruebas que se les pueden realizar al software como es el caso de las pruebas de rendimiento o desempeño que miden tiempos de respuesta, índices de procesamiento de transacciones y otros requisitos sensibles al tiempo. El objetivo de las mismas es verificar y validar los requisitos de desempeño que se han especificado. Además se pueden utilizar las pruebas de resistencia o stress, las cuales se proponen verificar que el sistema funciona apropiadamente y sin errores bajo condiciones de stress como: memoria baja, máximo número de clientes conectados o simulados y múltiples usuarios desempeñando la misma transacción con los mismos datos. Para la validación del componente fueron utilizadas las pruebas de caja blanca, realizadas a las funcionalidades implementadas en el mismo.

3.3. Validación de la implementación

3.3.1. Evaluación de la ejecución de las pruebas de caja blanca.

Método Cambiar Visualización de la clase CC_PronosticoMeteorologico (Ver Anexo 1)

Paso 1: Generar grafo de flujos de datos.



Paso 2: Cálculo de complejidad ciclomática.

$V(G) = \text{Cantidad Aristas} - \text{Cantidad Nodos} + 2$

$V(G) = 9 - 8 + 2 = 3$

Capítulo 3: Validación y pruebas realizadas al Componente Climático

Paso 3: Determinar los caminos básicos.

CB1: 1 2 3 8

CB2: 1 2 4 5 7 8

CB3: 1 2 4 6 7 8

Paso 4: Caso de prueba para el camino básico 1.

Entrada: tipo= vista_rapida

Resultado esperado: Se definen las variables para el pronóstico y se cambia el tipo de visualización para presentar los datos mediante la vista rápida.

Resultado de la prueba: Satisfactorio.

Paso 4: Caso de prueba para el camino básico 2.

Entrada: tipo= vista_pantallaCompleta

Resultado esperado: Se definen las variables para el pronóstico y se cambia el tipo de visualización para presentar los datos mediante la vista en pantalla completa.

Resultado de la prueba: Satisfactorio.

Paso 4: Caso de prueba para el camino básico 3.

Entrada: tipo= vista_formalInfocinta

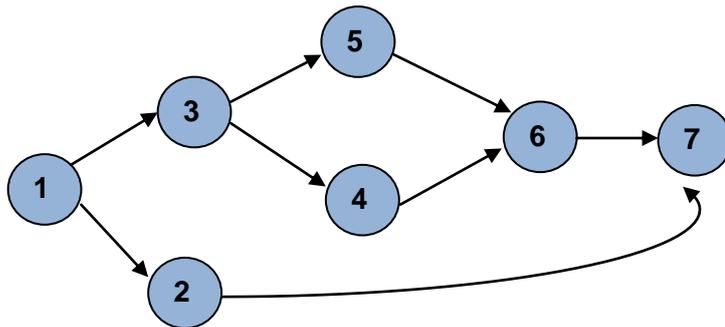
Resultado esperado: Se definen las variables para el pronóstico y se cambia el tipo de visualización para presentar los datos mediante la vista en forma de infocinta.

Resultado de la prueba: Satisfactorio.

Capítulo 3: Validación y pruebas realizadas al Componente Climático

Método Play de la clase CC_PronosticoMeteorologico (Ver Anexo 2)

Paso 1: Generar grafo de flujos de datos.



Paso 2: Cálculo de complejidad ciclomática.

$V(G) = \text{Cantidad Aristas} - \text{Cantidad Nodos} + 2$

$V(G) = 8 - 7 + 2 = 3$

Paso 3: Determinar los caminos básicos.

CB1: 1 2 7

CB2: 1 3 4 6 7

CB3: 1 3 5 6 7

Paso 4: Caso de prueba para el camino básico 1.

Entrada: tipo= vista_rapida

Resultado esperado: Se comienza a desencadenar el proceso de visualización de los títulos 1 de la vista rápida.

Resultado de la prueba: Satisfactorio.

Paso 4: Caso de prueba para el camino básico 2.

Entrada: tipo= vista_pantallaCompleta

Capítulo 3: Validación y pruebas realizadas al Componente Climático

Resultado esperado: Se comienza a desencadenar el proceso de visualización de los títulos 1 de la vista en pantalla completa.

Resultado de la prueba: Satisfactorio.

Paso 4: Caso de prueba para el camino básico 3.

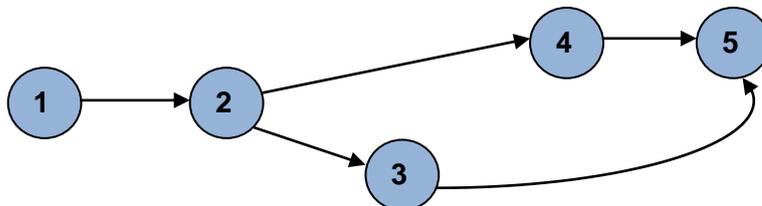
Entrada: tipo= vista_formalInfocinta

Resultado esperado: Se comienza a desencadenar el proceso de visualización de los títulos 1 de la vista en forma de infocinta.

Resultado de la prueba: Satisfactorio.

Método SalvarFichero de la clase C_LectorDatos (Ver Anexo 3)

Paso 1: Generar grafo de flujos de datos.



Paso 2: Cálculo de complejidad ciclomática.

$V(G) = \text{Cantidad Aristas} - \text{Cantidad Nodos} + 2$

$V(G) = 5 - 5 + 2 = 2$

Paso 3: Determinar los caminos básicos.

CB1: 1 2 4 5

CB2: 1 2 3 5

Capítulo 3: Validación y pruebas realizadas al Componente Climático

Paso 4: Caso de prueba para el camino básico 1.

Entrada: Qstring dirección

Resultado esperado: Se convierte la dirección al tipo de dato QFile, luego en el caso de que no se puede abrir la dirección se procede a salir del método, se rompe la sentencia de código.

Resultado de la prueba: Satisfactorio.

Paso 4: Caso de prueba para el camino básico 2.

Entrada: Qstring dirección, QDomDocument salvar.

Resultado esperado: Se convierte la dirección al tipo de dato QFile, luego en el caso de que se puede abrir la dirección se procede a guardar en ella los datos almacenados en el QDomDocument a salvar.

Resultado de la prueba: Satisfactorio.

Conclusiones

A partir de la investigación realizada para dar cumplimiento al objetivo general planteado en el presente trabajo de diploma se generó una serie de artefactos que le permiten a PRIMICIA presentar, mediante el proceso de transmisión del canal, informaciones de tipo meteorológicas. De forma general se concluye que:

- A partir de la investigación realizada a las funcionalidades del producto PRIMICIA actual, se determinó que para el mismo resulta de mucha importancia poder ampliar sus prestaciones, tipo de información a mostrar y posibles clientes.
- Con la utilización de las herramientas y metodología seleccionada se logró realizar y documentar el proceso de análisis, diseño e implementación del Componente de presentación de informaciones climáticas; concluyendo que la utilización de las mismas fue la adecuada para lograr obtener un producto final que cumple con todas las especificaciones planteadas por el cliente.
- La aplicación desarrollada otorga valor añadido al producto PRIMICIA, brindándole la posibilidad de adquirir nuevos clientes al permitir mostrar noticias relacionadas con temperaturas, vientos, estado del tiempo, fase de la luna y oleaje; a partir de una interfaz sencilla e intuitiva para los usuarios de la plataforma.

Recomendaciones

El objetivo general trazado en el presente trabajo de diploma fue alcanzado satisfactoriamente; sin embargo se recomienda:

- Desplegar el componente de presentación de informaciones climáticas en la Plataforma de Televisión Informativa, PRIMICIA.
- Continuar desarrollando nuevos tipos de visualización.
- Desarrollar una nueva versión para un proveedor internacional de datos meteorológicos.
- Proponer la implantación de sistemas de este tipo en hospitales, universidades, hoteles, otras sedes empresariales y ministeriales.

Bibliografía

Visual Paradigm International. 2011. Visual Paradigm. Visual Paradigm for UML. [En línea] 2011. [Citado el: 10 de diciembre de 2011.] <http://www.visual-paradigm.com>.

Acqweather. 2011. Meteorología, clima y desastre. Acqweather. [En línea] 2011. [Citado el: 24 de noviembre de 2011.] <http://www.acqweather.com/CLIMATOLOGIA/CLIMATOLOGIA.html>.

—. **2011.** Meteorología, clima y desastre. Acqweather. [En línea] 2011. [Citado el: 24 de noviembre de 2011.] <http://www.acqweather.com/METEOROLOGIA/METEOROLOGIA.html>.

Alfaro, Félix Murillo. 1999. Instituto Nacional de Estadísticas e Informática. Herramientas CASE. [En línea] 1999. [Citado el: 2011 de noviembre de 25.] <http://www.inei.gob.pe>.

Alvarez, Elaine Morales. 2010. Módulo de Gestión de Errores de la Plataforma de Televisión Informativa PRIMICIA. Ciudad de la Habana : s.n., 2010.

Cabrera, Lisandra Delgado. 2010. Diseño de las nuevas funcionalidades del Módulo de Redacción de la Plataforma de Televisión Informativa PRIMICIA. Ciudad de la Habana : s.n., 2010.

Canal El Tiempo. 2011. Canal ET. [En línea] 2011. [Citado el: 15 de noviembre de 2011.] <http://www.canaleltiempo.tv>.

Carrillo Pérez, Isaías, Pérez González, Rodrigo y Rodríguez Martín, Aureliano David. 2008. Metodología de desarrollo del software. 2008.

Delgado, Jordan. 2011. GlobeRed. Teoría de programación. [En línea] junio de 2011. [Citado el: 18 de enero de 2012.] <http://teoria-de-programacion.globered.com/categoria.asp?idcat=34>.

EcuRed. 2012. Eclipse, entorno de desarrollo integrado. Eclipse, entorno de desarrollo integrado. [En línea] 2012. [Citado el: 22 de enero de 2012.] http://www.ecured.cu/index.php/Eclipse,_entorno_de_desarrollo_integrado.

—. **2012.** Qt creator. [En línea] 2012. [Citado el: 9 de enero de 2012.] http://www.ecured.cu/index.php/Qt_Creator.

—. **2012.** Requisitos de Software. [En línea] 2012. [Citado el: 27 de enero de 2012.] http://www.ecured.cu/index.php/Requisitos_de_Software.

- . 2012. Visual Paradigm. EcuRed . [En línea] 2012. [Citado el: 25 de enero de 2012.] http://www.ecured.cu/index.php/Visual_Paradigm Sitio EcuRed.
- Free Download Manager.ORG. 2007.** Free Download Manager. Sitio de descargas de software. [En línea] 2007. [Citado el: 12 de diciembre de 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p.
- Garrido, Salvador Alemany. 2009.** Introduccion a QT. Programacion grafica en C++ con Qt4. 2009.
- Gutiérrez, Javier J. 2012.** ¿Qué es un framework web? [En línea] 2012. [Citado el: 22 de enero de 2012.] <http://www.cssblog.es/guias/Framework.pdf>.
- Gutiérrez, Javier J., y otros. 2006.** Modelos de pruebas para pruebas del sistema. [En línea] 2006. [Citado el: 7 de mayo de 2012.] <http://ceur-ws.org/Vol-227/paper07.pdf>.
- IBM_Corporation. 2011.** Rational Unified Process. Ayuda oficial en español de RUP. [En línea] 2011. [Citado el: 25 de noviembre de 2011.]
- IDE. 2012.** Entornos de desarrollo integrado. Concepto de IDE. [En línea] 2012. [Citado el: 25 de enero de 2012.] <http://petra.euitio.uniovi.es/~i1667065/HD/documentos/Entornos%20de%20Desarrollo%20Integrado.pdf>.
- INSMET. 2012.** Instituto de Meteorología de la República de Cuba. Instituto de Meteorología. [En línea] 2012. <http://www.insmet.cu/>.
- Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** El Proceso Unificado de Desarrollo de Software. Madrid : Addison Wesley : s.n., 2000. 84-7829-036-2.
- Java. 2012.** Que es Java ? Características de Java como Lenguaje de programación. [En línea] 2012. [Citado el: 20 de enero de 2012.] <http://www.infor.uva.es/~jmrr/tgp/java/JAVA.html>.
- Larman, Craig. 2da Edición.** UML y Patrones. Introducción al Análisis y Diseño Orientado a Objetos. s.l. : Prentice Hall, 2da Edición. 9701702611.
- Lenguajes de programación. 2009.** Lenguajes de programación. [En línea] 2009. [Citado el: 10 de enero de 2012.] <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>.

Leyva, Osmany Torres. 2011. Implementación del Subsistema de Configuración de la Plataforma de Televisión Informativa, PRIMICIA. La Habana, Cuba : s.n., 2011.

Louden, Kenneth C. 2004. Lenguajes de programación: principios y prácticas. Mexico : s.n., 2004.

Martínez, Alejandro y Martínez, Raúl. Guía a Rational Unified Process. Universidad de Castilla la Mancha : s.n.

Matos, Heikel Villar. 2008. Implementación de los subsistemas de Administración y Transmisión de la Plataforma de Televisión Informativa PRIMICIA. Ciudad de la Habana : s.n., 2008.

Ministerio de Administraciones Públicas. 2002. Metodología MÉTRICA. Técnicas y Prácticas. 2002.

Morales, Yaquelin Y. Rodríguez y Román, Adisneisy C. Remedio. 2007. Propuesta de una Guía para estandarizar la codificación en la Universidad. Ciudad de la Habana : s.n., 2007.

Navarro, Mendoza. Diseño del Sistema de Tarjeta de Crédito Con UML.

Nokia. 2012. QT. Sitio oficial de QT. [En línea] 2012. [Citado el: 24 de enero de 2012.] <http://qt.nokia.com/products>.

Oracle. 2012. JAVA. Sitio oficial de JAVA. [En línea] 2012. [Citado el: 20 de enero de 2012.] http://www.java.com/es/download/faq/whatis_java.xml.

OrganiZator. 2012. OrganiZator. Curso C++. [En línea] 2012. [Citado el: 16 de enero de 2012.] http://www.zator.com/Cpp/E1_2.htm .

Pressman, Roger S. 5ta Edición. Ingeniería de Software. Un enfoque práctico. s.l. : Mc Graw Hill, 5ta Edición.

Programación en castellano. 2011. Programación en castellano. Qt Creator, un completo entorno de desarrollo . [En línea] 2011. [Citado el: 10 de diciembre de 2011.] http://www.programacion.com/noticia/qt_creator-_un_completo_entorno_de_desarrollo_1723.

Pruebas_de_software. 2005. Pruebas de Software. Gestión de Calidad y Pruebas de Software. [En línea] 2005. [Citado el: 5 de mayo de 2012.] <http://pruebasdesoftware.com/laspruebasdesoftware.htm>.

Pruebas_de_Software. 2005. Pruebas de software. Gestión de Calidad y Pruebas de Software. [En línea] 2005. [Citado el: 5 de mayo de 2012.] <http://pruebasdesoftware.com/laspruebasdesoftware.htm>.

- Reynoso, Carlos y Kiccillof, Nicolás. 2004.** Estilos y Patrones en la Estrategia de Arquitectura de Microsoft. Buenos Aires : s.n., 2004.
- Rodríguez, Raúl Lugo. 2011.** Análisis, diseño e implementación del subsistema de Transmisión de la plataforma de televisión informativa PRIMICIA versión 2.0. La Habana, Cuba : s.n., 2011.
- Sanjúan, Olga Pérez. 2007.** Recordando la Historia: El concepto de "televisión" en sus orígenes. s.l. : ISSN 0210-3923, 2007.
- Santos, William De la Cruz De los y Altamirano, Leopoldo. 2009.** Programación paralela y concurrente en C++. Universidad Juárez. México : s.n., 2009.
- Schneider, Richard. 1968.** Contabilidad industrial. Madrid : Aguilar, 1968.
- Sommerville, Ian. 2004.** Ingeniería del Software. Madrid : PEARSON EDUCATION S.A, 2004. 84-7829-074-5.
- T.W.C. Interactive. 2011.** The Weather Channel . [En línea] 2011. [Citado el: 30 de noviembre de 2011.] <http://weather.com/>.
- Terrero, Yaniley Mendez. 2010.** Conceptualización del subsistema de transmisión de noticias como aplicación de escritorio para la Plataforma de Televisión Informativa. Ciudad de la Habana, Cuba : s.n., 2010.
- TuTiempo.Net. 2011.** Diccionario de meteorología. Pronóstico Meteorológico. [En línea] 2011. [Citado el: 20 de noviembre de 2011.] http://www.tutiempo.net/diccionario/pronostico_meteorologico.html.
- Uml.Slideshare, Visual Paradigm For. 2011.** Visual Paradigm For Uml. Slideshare. [En línea] 2011. [Citado el: 14 de Diciembre de 2011.] <http://www.slideshare.net/vanquishdarkenigma/visual-paradigm-for-uml>.
- Urbino, Rafael Jacobo Hidalgo. 2010.** Análisis, diseño e implementación de una herramienta que permita la centralización de la información gestionada por el módulo Resultados de la Colección Multisaber . Ciudad de la Habana : s.n., 2010.
- Valdés, Geovanys Bermúdez. 2008.** Propuesta de Sistema Operativo con kernel Linux para la Plataforma de Televisión Informativa, PRIMICIA. Ciudad de la Habana : s.n., 2008.

Visual Paradigm for UML. 2011. Visual Paradigm for UML. Herramienta CASE para modelamiento UML. [En línea] 2011. [Citado el: 11 de diciembre de 2011.] <http://www.todoprogramas.com/programalinux/visualparadigmforuml>.

Anexos

Anexo 1: Código del método Cambiar Visualización de la clase CC_PronosticoMeteorologico

```

void CC_PronosticoMeteorologico::CambiarVisualizacion()
{
    /*Variables para mostrar el Estado del Tiempo*/           1
    titulo_2_estado= new QGraphicsTextItem;
    QString estado=estadoTiempo.GetValorEstado();
    titulo_2_estado->setHtml(estado);
    /*Variables para mostrar las Temperaturas*/
    titulo_2_2tmin= new QGraphicsTextItem;
    int temperaturaMin=temperatura.GetValorTMin();
    titulo_2_2tmin->setHtml(QString::number(temperaturaMin));
    titulo_2_2tmax= new QGraphicsTextItem;
    int temperaturaMax=temperatura.GetValorTMax();
    titulo_2_2tmax->setHtml(QString::number(temperaturaMax));
    /*Variables para mostrar la fase de la Luna*/
    titulo_2_2luna= new QGraphicsTextItem;
    Fase fase=luna.CalcularFase();
    titulo_2_2luna->setHtml(QString(fase));
    /*Variables para mostrar el Viento*/
    dirViento = new QGraphicsTextItem;
    QString direccion=viento.GetDireccion();
    dirViento->setHtml(direccion);
    titulo_2_2wiento = new QGraphicsTextItem;
    int velocidad=viento.GetVelocidad();
    titulo_2_2wiento->setHtml(QString::number(velocidad));

    if(tipo==vista_rapida)                                     2
    {
        MostrarVista_Rapida();                                3
    }
    else if(tipo==vista_pantallaCompleta)                     4
    {
        MostrarVista_PantallaCompleta();                      5
    }
    else                                                       6
    {
        MostrarVista_FormaInfocinta();                          7
    }                                                         8
}

```

Anexo 2: Código del método Play de la clase CC_PronosticoMeteorologico

```
void CC_PronosticoMeteorologico::Play()
{
    if (tipo==vista_rapida)           1
    {
        MostrarTitulos1_VistaRapida(); 2
    }
    else if (tipo==vista_pantallaCompleta) 3
    {
        MostrarTitulos1_VistaPantallaCompleta(); 4
    }
    else                               5
    {
        MostrarTitulos1_VistaFormaInfocinta();
    }
}
```

Anexo 3: Código del método Salvar Fichero de la clase LectorDatos

```
void LectorDatos::SalvarFichero(QString direccion, QDomDocument salvar)
{
    QFile ficheroFile(direccion);
    if (!ficheroFile.open(QFile::WriteOnly | QFile::Truncate))
    {
        return;
    }
    QTextStream salida(&ficheroFile);
    salida << salvar.toString();
    ficheroFile.close();
}
```

Glosario de Términos

Applet: componente de software que corre en el contexto de otro programa. No puede correr de manera independiente, ofrece información gráfica y puede interactuar con el usuario; normalmente lleva a cabo una función en específico que carece de uso independiente.

Celsius: Unidad adoptada convencionalmente para medir la temperatura. Actualmente están en uso el grado Celsius o centígrado y el grado Fahrenheit.

Centígrado: unidad de temperatura que equivale a la centésima parte de la diferencia entre los puntos de fusión del hielo y de ebullición del agua, a la presión normal. (Simbología: °C).

Escaleta: esqueleto o esquema del video que ayuda a organizar el plan de grabación y facilita la escritura del pre-guion y del guion definitivo.

Fahrenheit: unidad de temperatura de la escala Fahrenheit, que asigna el valor 32 al punto de fusión del hielo y el valor 212 al de ebullición del agua. (Simbología: °F).

GNOME: es una comunidad internacional dedicada a hacer software eficiente y que cualquier persona pueda utilizar, independientemente del idioma que hable o de capacidades técnicas o físicas presentes en los usuarios.

Infocinta: Cintillo informativo que se muestra en la televisión, utilizados para promocionar eventos de última hora, noticias relevantes o acontecimientos de gran importancia.

Iteraciones: es un ciclo de desarrollo completo que genera como resultado una entrega de producto ejecutable (interna o externa).

Kelvin: unidad de temperatura del Sistema Internacional. Es igual al grado centígrado, pero en la escala de temperatura absoluta el 0 está fijado en $-273,16^{\circ}\text{C}$. (Simbología: K).

Meso-escala: estudio que se realiza en la meteorología a los sistemas del tiempo atmosférico más pequeños que la escala sinóptica meteorológica. Un fenómeno de meso-escala posee una duración de 1 a 12 horas o una extensión horizontal entre 1 y 100 km o una altura de 1 a 10 km. Ejemplos de estos fenómenos constituyen las brisas del mar y los tornados.

Meteorología: ciencia que se encarga de estudiar la atmósfera y los meteoros.

Plugins: módulo de software o hardware que añade una característica o un servicio específico a un sistema más grande.

Pronóstico: calendario en el cual se incluye el anuncio de los fenómenos astronómicos y meteorológicos.

Teletexto: es un servicio de información que muestra por la televisión las noticias en forma de texto combinado con imágenes, sonidos y videos.