



UNIVERSIDAD POLITÉCNICA DE PUEBLA

PROGRAMA ACADÉMICO DE INGENIERÍA EN INFORMÁTICA

Sistema Para Transmisión “Streaming”

Alma Delia Salinas Flores

Reporte Técnico PII-05-08-09

COMITÉ EVALUADOR

M.C. Luis Alberto Morales Rosales (*Asesor*)
Dra. María Auxilio Medina Nieto (*Sinodal*)
Dr. Antonio Benítez Ruiz (*Sinodal*)

PROFESORA DE PROYECTO DE INVESTIGACIÓN II:

Dra. María Auxilio Medina Nieto

Juan C. Bonilla, Puebla
Agosto 2009

Índice

Resumen	3
Capítulo 1. Planteamiento del Problema.	
1.1. Introducción.....	5
1.2. Objetivo general.....	6
1.3. Objetivos específicos.....	6
1.4. Justificación.....	6
1.5. Cronograma de actividades	7
1.6. Alcances y limitaciones	7
1.7. Recursos de hardware y software	8
Capítulo 2. Marco teórico	
2.1. Sistemas de tiempo real.	
2.1.1. Definición de STR.	9
2.1.2. Clasificación de los STR	9
2.1.3. Elementos de un STR.	9
2.2. Transmisión “streaming”	
2.2.1. Definición de streaming.....	10
2.2.2. Compañías que usan aplicaciones streaming.....	10
2.2.3. Proceso y requerimientos de transmisión streaming.....	10
Capítulo 3. Diseño.	
3.1. Requerimientos	
3.1.1. Requerimientos funcionales.....	12
3.1.2. Requerimientos no funcionales.....	14
3.2. Casos de uso.....	14
3.3. Especificación de actores.....	20
3.4. Diagrama conceptual.....	20
3.5. Datos de entrada, datos de salida y descripción por módulo.....	21
Capítulo 4. Implementación.	
4.1. Validación de usuario.....	26
4.2. Transferencia de video.....	26
4.3. La interfaz general.....	27
4.4. Clase DataSource.....	28
4.5. Clases Player y Processor	29
4.6. Jerarquía de la clase Player.....	29
4.7. Clase Datasink	29
4.8. Manager y otras clases de control.....	30
4.9. Sockets.....	30
4.10. La clase DataOutputStream.....	31
4.11. La clase DataInputStream.....	32
4.12. Serializable.....	32
4.13. Código y pantallas de captura.....	32
4.14. Almacenamiento.....	33
4.15. Resultados.....	35
Capítulo 5. Conclusiones	37
Referencias.....	38

Índice De Figuras

Figura 1: Elementos y proceso de transmisión streaming.....	11
Figura 2. Diagrama de operaciones del usuario en el servicio.....	14
Figura 3 Diagrama del módulo de validación.....	15
Figura 4. Diagrama del módulo de monitoreo.....	16
Figura 5. Diagrama del módulo de videos.....	17
Figura 6. Diagrama del módulo de fotos.....	18
Figura 6. Diagrama del módulo de ayuda.....	19
Figura 8. Diagrama conceptual del sistema.....	21
Figura 9: Interfaz de la validación de usuario.....	23
Figura 10: Interfaz de la pantalla principal. (Ambiente Monitoreado: Site UPP).....	23
Figura 11: Ejemplo de TextField, JPasswordField y Botones.....	24
Figura 12.1: Jerarquía de DataSource.....	28
Figura 12.2: Atributos de la clase Data Source.....	28
Figura 13.1 Atributos de la clase Player.....	29
Figura 14.1: Jerarquía de la clase Player.....	29
Figura 15: Funcionamiento de los sockets.....	31
Figura 16: Uso de Serializable en el sistema.....	32
Figura 17: Código para la detección de la cámara.....	32
Figura 18: Resultado de la captura de un video en una de las pruebas.....	33
Figura 19: Resultado de la captura de fotografías en una de las pruebas.....	34
Figura 20: Carpeta con los archivos generados.....	34
Figura 21: Panel y JList de Videos.....	35
Figura 22 Panel y JList de Fotos.....	35
Figura 23: Cuadro de diálogo para descargar las fotos y videos.....	35
Figura 24: Resultados en modo consola del servidor.....	36
Figura 25: Resultados en modo consola del cliente.....	36

Índice De Tablas

Tabla 1. Requerimiento funcional SMW00.....	12
Tabla 2. Requerimiento funcional SMW01.....	12
Tabla 3. Requerimiento funcional SMW02.....	13
Tabla 4. Requerimiento funcional SMW03.....	13
Tabla 4. Requerimiento funcional SMW04.....	13
Tabla 4. Requerimiento funcional SMW05.....	14
Tabla 7. Caso de uso CUSMW_01 validación de usuario....	15
Tabla 8. Caso de uso CUSMW_02 monitoreo de ambiente.....	16
Tabla 9. Caso de uso CUSMW_04 módulo de videos.....	17
Tabla 10. Caso de uso CUSMW_05 ayuda módulo de fotos.....	18
Tabla 11. Caso de uso CUSMW_06 ayuda módulo de ayuda.....	19
Tabla 12. Descripción del actor administrador.....	20
Tabla 13. Descripción del actor usuario.....	20
Tabla 14. Descripción del módulo de validación.....	21
Tabla 15. Descripción del módulo de monitoreo.....	22
Tabla 16. Descripción del módulo de reproducción de videos.....	22
Tabla 17. Descripción del módulo de visualización de fotografías.....	22
Tabla 18. Descripción del módulo Ayuda.....	22
Tabla 19. Descripción de componentes de la interfaz.....	24

Resumen

Las nuevas tecnologías junto con el incremento de ancho de banda posibilitan la distribución de contenidos multimedia en red, hoy en día son las aplicaciones en red, compartir datos de manera remota, etc., pero entre los problemas que se presenta el compartir datos es la incompatibilidad de software, las restricciones de uso, los diferentes formatos y aplicaciones usadas en la reproducción y transmisión de los datos, es por ello que es el *streaming* una tecnología que permite la transmisión de datos multimedia en tiempo real o bajo demanda y siendo la segunda la usada en este proyecto debido a las ventajas y beneficios de transmisión y portabilidad propias de la tecnología.

Capítulo 1. Planteamiento del problema

1.1. Introducción

Los grandes avances de la tecnología han experimentado un amplio crecimiento con el que permiten a las personas comunicarse y realizar tareas más rápida y fácilmente, un ejemplo es el internet que permite la comunicación entre la gente de todo el mundo que tenga acceso a él de manera más rápida y a bajo costo. El uso del audio y video se ha incrementado en internet, sin embargo no es el único lugar donde dichos recursos se utilizan, sino también en aplicaciones y sistemas que se usan localmente para monitoreo, videoconferencias, sistemas de educación a distancia, etc. Este tipo de servicios y aplicaciones suele ser de un costo muy alto y más cuando se trata de seguridad, las aplicaciones de monitoreo resultan ser costosas limitadas a las necesidades del usuario y con restricciones. Esta tecnología no sólo se usa como un medio de entretenimiento o comunicación, las empresas también hacen uso de ésta para mejorar y estar a la vanguardia. “El *streaming* es una tecnología que permite enviar una señal de audio o video a través de una red [1] ”. El streaming es rápido, cómodo y sencillo, aunque que precisa internet de alta velocidad.

A continuación se describe de forma general el proceso para este proyecto. Para realizar la visualización de video, el usuario accederá a una aplicación y desde su computadora iniciará la reproducción. Un servidor (quien ofrece el servicio) envía la transmisión de forma continua al tiempo que el video es visualizado. El streaming ofrece una ventaja en la visualización del video porque no requiere realizar una descarga total. Con la descarga de una pequeña parte, el usuario podrá comenzar a ver el contenido, mientras continúa recibiendo la corriente de datos que irá visualizando posteriormente de forma continua.

El proyecto “Sistema para Transmisión Streaming” tiene como objetivo principal el realizar el monitoreo remoto de un medio ambiente usando la tecnología streaming. Para la visualización y captura del video, se desarrolla una interfaz de usuario. La interfaz permite al usuario realizar grabaciones, tomar imágenes y descargarlas.

1.2. Objetivo general

Desarrollar un sistema para monitoreo remoto de un ambiente usando *streaming*.

1.3. Objetivos específicos

- Construir una interfaz gráfica de usuario (GUI) para el sistema.
- Monitorear un lugar en tiempo real a través del sistema.
- Implementar operaciones del manejo del streaming.

1.4. Justificación

Actualmente el *streaming* es una tecnología ampliamente utilizada que se encuentra disponible para la visualización de videos y audio, ya sea en demanda o tiempo real. Sin embargo, es necesario contar con mecanismos que permitan el monitoreo remoto de bajo costo, portabilidad o compatibilidad estándar ya que, hoy por hoy, las aplicaciones existentes son costosas y están sujetas a restricciones de software, incluso al uso de otras aplicaciones como en el caso de los reproductores. Este proyecto se propone como apoyo al funcionamiento de dichas aplicaciones, o bien a la necesidad de contratar con servicios externos para realizar el proceso de transmisión y presentación de datos, es por ello que “el sistema para transmisión streaming” busca reducir el costo así como la portabilidad de una aplicación de streaming.

1.5. Cronograma de actividades

Tiempo	Enero		Febrero				Marzo				Marzo				Abril				Mayo				Junio				Julio			
	Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10	Semana 11	Semana 12	Semana 13	Semana 14	Semana 15	Semana 16	Semana 17	Semana 18	Semana 19	Semana 20	Semana 21	Semana 22	Semana 23	Semana 24	Semana 25	Semana 26	Semana 27	Semana 28	Semana 29	Semana 30
1	█																													
2		█	█																											
3				█	█																									
4						█	█																							
5								█	█	█																				
6									█	█	█	█	█	█																
7														█	█	█	█	█	█	█	█									
8																				█	█	█								
9		█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█

Actividad

- 1 Asignación de proyecto
- 2 Revisar el estado del arte en software y hardware de la cámara
- 3 Pruebas del software de la cámara
- 4 Comenzar el diseño e implementación de la interfaz para la aplicación (GUI)
- 5 Integración del hardware con la GUI
- 6 Diseño e implementación de comandos de manipulación del hardware
- 7 Diseño e implementación de comandos para el almacenamiento de streaming
- 8 Generación de lista de videos almacenados
- 9 Diseñar e implementar la interfaz del módulo de visualización
- 10 Pruebas y evaluación

1.6 Alcances y limitaciones

- ✓ El sistema atenderá a lo más 10 usuarios simultáneamente.

- ✓ El almacenamiento de los videos del usuario estará limitado a 5Mb de capacidad debido al consumo de memoria que genera

1.7 Recursos de hardware y software

1.7.1 Recursos de hardware

- ✓ Procesador AMD Sempron de 1,59 GHz
- ✓ Disco duro de 120 G
- ✓ Memoria RAM 1.5 G
- ✓ Cámara SmileCam 320

1.7.2 Recursos de software

- ✓ Sistema operativo Windows XP Professional
- ✓ SP2
- ✓ Herramienta de desarrollo Java JDK 6
- ✓ API de Java Media Framework (JMF 2.1.1)
- ✓ Editor JCreator Pro 4.5 .0
- ✓ Editor Front Page 2003
- ✓ Demo SmileCam 320 [2]

Capítulo 2. Marco teórico

Este capítulo describe temas relacionados con el sistema propuesto como sistemas en tiempo real, clasificación y componentes, así como conceptos básicos de la tecnología streaming.

2.1.1. Definición de STR

Un sistema de tiempo real (STR) es un sistema informático en el que es significativo el tiempo en el que se producen las acciones. Las acciones deben realizarse dentro de un intervalo de tiempo determinado [7]. Un STR maneja variables de procesos externos (a través de dispositivos especiales de E/S), responde de una manera predecible a estímulos externos impredecibles. Más que ser rápido, debe ser predecible.

2.1.2. Clasificación de los STR

Según [7], los sistemas STR pueden clasificarse en *críticos* (“hard”) y *suaves* (“soft”). El sistema propuesto se considera de tipo suave, ya que un fallo eventual no compromete el funcionamiento correcto [7].

2.1.3. Elementos de un STR

Un STR se compone de [7]:

- Aspectos de integración y de rendimiento
- Manejo de interrupciones
- Bases de datos de tiempo real
- Sistemas operativos de tiempo real
- Lenguajes de tiempo real
- Sincronización y comunicación de tareas

Los STR están caracterizados por la forma en cómo presentan los datos al usuario. A continuación se presenta el streaming, que complementa el tema anterior ya que a través de esta tecnología se puede realizar el monitoreo en tiempo real.

2.2. Transmisión *streaming*

2.2.1. Definición de streaming

Streaming es una tecnología que permite incluir audio, video y otros elementos multimedia en una página web [3]. Los usuarios pueden escuchar o ver archivos multimedia mientras son transmitidos en tiempo real.

2.2.2. Compañías que usan aplicaciones streaming

Son 5 las compañías principales de streaming (con sus respectivos reproductores) [9]:

- Microsoft (con Windows Media - audio y video)
- Real Networks (con Real Media, Real Video y Real Audio)
- Apple (con QuickTime)
- Macromedia (con Flash/Shockwave)
- Nullsoft Streaming Video (.nsv) y Shoutcast, IceCast (versión de software libre de Shoutcast), Live365 streaming de audio

Los tres primeros reproductores de contenido streaming son para plataformas Mac y Windows, mientras que los dos siguientes son sólo para plataforma Windows. El sistema de este proyecto es portable, además tendrá la ventaja de contar con un reproductor propio.

2.2.3. Proceso y requerimientos de transmisión streaming

El uso de datos multimedia es tan fácil como el internet, sin embargo, el proceso de transmisión streaming requiere de un proceso el cual se describe a continuación [6].

1. Usando un navegador web, encontrar un sitio que ofrezca streaming de video o audio
2. El servidor web de la página pide el archivo desde el servidor de streaming
3. El software en el servidor de streaming divide el archivo en trozos y los envía a la computadora utilizando los protocolos en tiempo real

4. El plug-in del navegador (reproductor de Flash o aplicación de una computadora) decodifica y muestra los datos en cuanto lleguen
5. La computadora descarta los datos

Los pasos anteriores requieren de tres componentes básicos que resulten compatibles entre sí: un cliente, un servidor y un flujo de datos. La creación y distribución de un streaming de video o de audio requiere su propio proceso, éste se describe con los siguientes pasos [6].

1. Grabar un video de alta calidad o archivo de audio utilizando película o una grabadora digital
2. Digitalizar los datos de importación en la computadora y, si fuera necesario, convertir con el software de edición
3. Si se va a crear un streaming de video, se hace más pequeño el tamaño de la imagen y se reduce la frecuencia de cuadros
4. Un códec de la computadora comprime el archivo y lo codifica al formato adecuado
5. El cliente carga el archivo en un servidor
6. El servidor envía el archivo a las computadoras de los usuarios.

La Figura 2 muestra los elementos usados en streaming, así como su proceso de creación y distribución.

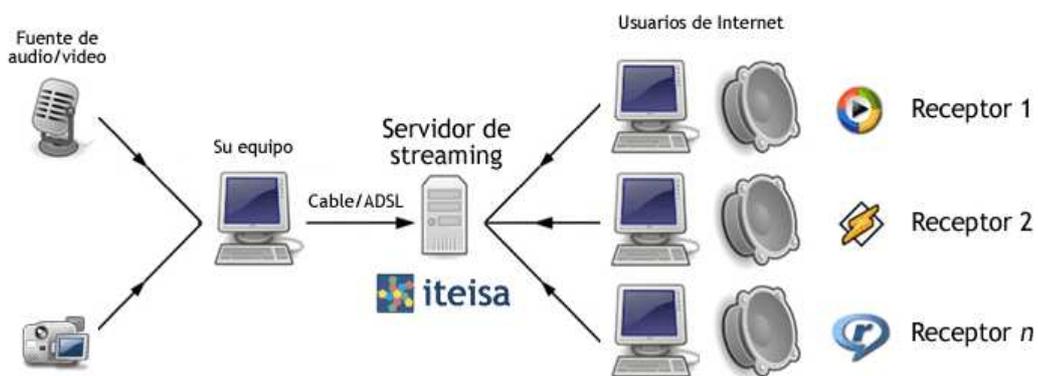


Figura 1: Elementos y proceso de transmisión streaming

Capítulo 3. Diseño

Este capítulo se refiere a la forma en que el sistema cumplirá con los requerimientos funcionales y no funcionales siguientes.

3.1 Requerimientos

3.1.1 Requerimientos funcionales

Las tablas 1 a 6 describen los requerimientos funcionales del sistema.

Tabla 1. Requerimiento funcional SMW00

<i>Identificador:</i>	SMW00
<i>Nombre de requerimiento:</i>	Validación
<i>Descripción corta:</i>	Validará que el usuario esté registrado para hacer uso del servicio.
<i>Descripción detallada:</i>	<ol style="list-style-type: none">1. La ventana muestra las etiquetas de presentación y logotipos de las instituciones.2. La ventana muestra campos para insertar el identificador de usuario y la contraseña a validar.3. La ventana muestra los botones de aceptar y cancelar.

Tabla 2. Requerimiento funcional SMW01

<i>Identificador:</i>	SMW01
<i>Nombre de requerimiento:</i>	Menú principal
<i>Descripción corta:</i>	Visualizará el menú principal
<i>Descripción detallada:</i>	<ol style="list-style-type: none">1. La ventana muestra las etiquetas de presentación y logotipos de las instituciones.2. La ventana muestra una sección con un panel llamado <i>monitoreo</i> que incluye los botones de manipulación.3. La ventana muestra una segunda sección con un segundo panel llamado <i>videos</i> que incluye los botones de reproducción.

Tabla 3. Requerimiento funcional SMW02

<i>Identificador:</i>	SMW02
<i>Nombre de requerimiento:</i>	Monitoreo
<i>Descripción corta:</i>	Visualizará un espacio en tiempo real
<i>Descripción detallada:</i>	<ol style="list-style-type: none"> 1.El servicio corre en tiempo real 2.El servicio está restringido a usuarios registrados 3.El usuario ingresa a través de un identificador y una contraseña 4.El usuario puede manipular la cámara

Tabla 4. Requerimiento funcional SMW03

<i>Identificador:</i>	SMW03
<i>Nombre de requerimiento:</i>	Grabar video
<i>Descripción corta:</i>	Graba videos en tiempo real
<i>Descripción detallada:</i>	<ol style="list-style-type: none"> 1. El usuario puede grabar un video de su monitoreo 2. El video es guardado por tiempo restringido 3. La grabación se puede pausar 4. La grabación puede continuar 5. La grabación se puede detener

Tabla 5. Requerimiento funcional SMW04

<i>Identificador:</i>	SMW04
<i>Nombre de requerimiento:</i>	Videos
<i>Descripción corta:</i>	Muestra videos grabados
<i>Descripción detallada:</i>	<ol style="list-style-type: none"> 1. El servicio muestra una lista de videos grabados por el usuario 2. Muestra una ventana de reproducción para ver el video 3. Los videos se pueden reproducir 6. Los videos se puede pausar 4. Los videos se puede detener 5. Los videos se pueden descargar.

Tabla 6. Requerimiento funcional SMW05

<i>Identificador:</i>	SMW05
<i>Nombre de requerimiento:</i>	Ayuda
<i>Descripción corta:</i>	Muestra una breve explicación de cómo usar el servicio
<i>Descripción detallada:</i>	Muestra la explicación de cada componente del servicio y cómo usarlo

3.1.2 Requerimientos no funcionales

La disponibilidad del servicio dependerá del buen funcionamiento de la cámara, la accesibilidad del enlace y del servidor. La seguridad se implementa a través de la autenticación de usuarios, no existirán diferentes tipos de usuario, sin embargo, se contará con un administrador del servicio, quien será encargado de la información guardada y de la alta y baja de los usuarios. Las pruebas del servicio se realizarán utilizando una IP privada (192.168.1.66).

3.2 Casos de uso

El servicio cuenta con algunos casos de uso. Por cada caso de uso, existe un módulo. Éstos se describen en las tablas (7-11) y figuras (2-7)

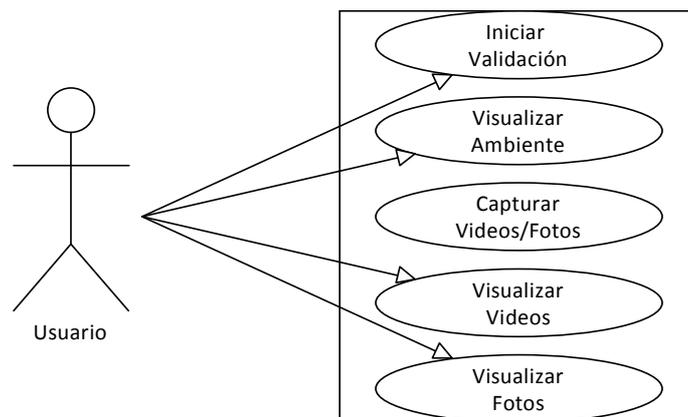


Figura 2. Diagrama de operaciones del usuario en el servicio

Tabla 7. Caso de uso CUSMW_01 validación de usuario

<i>Caso de uso:</i>	CUSMW_01 Validación de usuario
<i>Descripción:</i>	Un usuario ingresa al servicio a través de su identificador y contraseña
<i>Actor principal:</i>	Usuario
<i>Personal involucrado e interés</i>	Administrador: necesita haber dado de alta al usuario
<i>Precondiciones:</i>	Autenticación del usuario
<i>Garantías de éxito: (post condiciones)</i>	La información que mantiene el administrador es consistente y confiable para cualquier usuario autorizado que acceda al servicio
<i>Escenario principal de éxito o flujo básico</i>	<ol style="list-style-type: none"> 1. El usuario se autentifica 2. El usuario monitorea el ambiente
<i>Extensiones o flujos alternativos</i>	<p>Problemas con el navegador</p> <ol style="list-style-type: none"> a) Reintentar o probar otro navegador disponible b) Actualizar navegador <p>La autenticación del usuario no es válida</p> <ol style="list-style-type: none"> a) Acceder nuevamente b) Si el identificador o la contraseña no son válidas, emitir el mensaje correspondiente y mostrar la página de acceso c) Después de 3 intentos fallidos, el usuario deberá enviar un correo al administrador para que éste verifique los datos. <p>El servicio se bloquea</p> <ol style="list-style-type: none"> a) Cerrar sesión y volver a iniciar

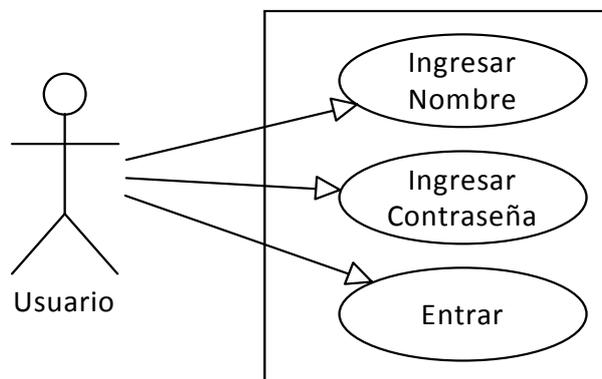


Figura 3 Diagrama del módulo de validación

Tabla 8. Caso de uso CUSMW_02 monitoreo de ambiente

<i>Caso de uso:</i>	CUSMW_02 monitoreo de ambiente
<i>Descripción:</i>	El usuario puede manipular la cámara, grabar un video y tomar fotos
<i>Actor principal:</i>	Usuario
<i>Personal involucrado e interés</i>	Administrador: necesita haber dado de alta al usuario.
<i>Precondiciones:</i>	Autenticación del usuario Entrar al servicio
<i>Garantías de éxito: (post condiciones)</i>	La información que mantiene el administrador es consistente y confiable para cualquier usuario autorizado que acceda al servicio
<i>Escenario principal de éxito o flujo básico</i>	1. El usuario está autenticado 2. El usuario monitorea el ambiente 3. El usuario graba un video
<i>Extensiones o flujos alternativos</i>	Problemas con el navegador a) Reintentar o probar otro navegador disponible b) Actualizar navegador Descarga incompleta de videos a) Volver a iniciar El servicio se bloquea a) Cerrar sesión y volver a iniciar

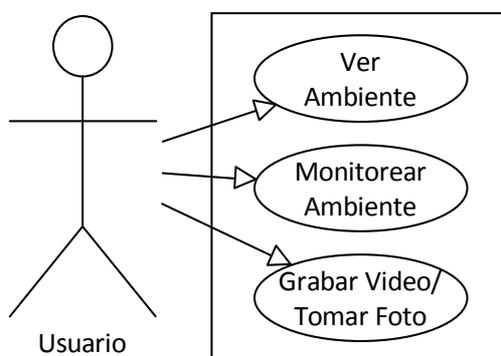


Figura 4. Diagrama del módulo de monitoreo

Tabla 9. Caso de uso CUSMW_04 módulo de videos

<i>Caso de uso:</i>	CUSMW_03 Videos
<i>Descripción:</i>	Un usuario puede ver los videos que ha grabado
<i>Actor principal:</i>	Usuario
<i>Personal involucrado e interés:</i>	Administrador: necesita haber dado de alta al usuario
<i>Precondiciones:</i>	Autenticación del usuario Haber grabado al menos un video
<i>Garantías de éxito: (post condiciones)</i>	La información es consistente y confiable para cualquier usuario autorizado que acceda al servicio
<i>Escenario principal de éxito o flujo básico</i>	<ol style="list-style-type: none"> 1. El usuario ha grabado al menos un video 2. El usuario visualiza los videos grabados 3. El usuario descarga el video
<i>Extensiones o flujos alternativos</i>	<p>Problemas con el reproductor</p> <ol style="list-style-type: none"> a) Reintentar o probar otro reproductor disponible b) Actualizar navegador <p>El servicio se bloquea</p> <ol style="list-style-type: none"> a) Cerrar sesión y volver a iniciar

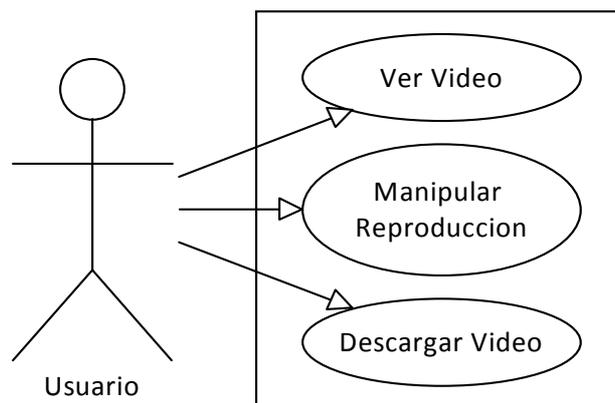


Figura 5. Diagrama del módulo de videos.

Tabla 10. Caso de uso CUSMW_05 ayuda módulo de fotos

<i>Caso de uso:</i>	CUSMW_05 Fotos
<i>Descripción:</i>	Un usuario puede ver las fotos que ha grabado
<i>Actor principal:</i>	Usuario
<i>Personal involucrado e interés</i>	Administrador: necesita haber dado de alta al usuario
<i>Precondiciones:</i>	Autenticación del usuario Haber grabado al menos una foto
<i>Garantías de éxito: (post condiciones)</i>	La información es consistente y confiable para cualquier usuario autorizado que acceda al servicio
<i>Escenario principal de éxito o flujo básico</i>	<ol style="list-style-type: none"> 1. El usuario ha grabado al menos una foto 2. El usuario visualiza las fotos grabadas 3. El usuario descarga la foto
<i>Extensiones o flujos alternativos</i>	<p>Problemas con el reproductor</p> <p>a) Actualizar navegador</p> <p>El servicio se bloquea</p> <p>a) Cerrar sesión y volver a iniciar</p>

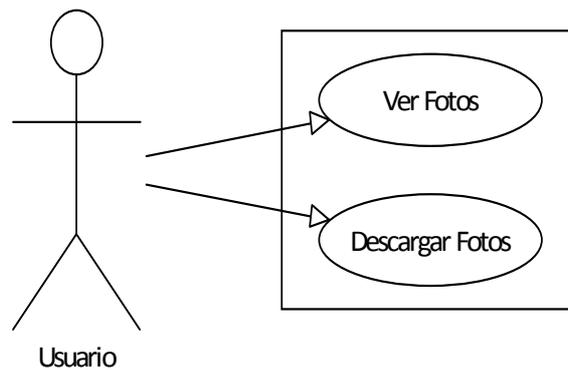


Figura 6. Diagrama del módulo de fotos

Tabla 11. Caso de uso CUSMW_06 ayuda módulo de ayuda

<i>Caso de uso:</i>	CUSMW_06 Ayuda
<i>Descripción:</i>	El usuario puede consultar la ayuda para resolver alguna duda sobre el uso del servicio
<i>Actor principal:</i>	Usuario
<i>Personal involucrado e interés</i>	Administrador: necesita haber definido un tutorial o manual en forma de ayuda.
<i>Precondiciones:</i>	Entrar al servicio, elegir Ayuda → Ayuda
<i>Garantías de éxito: (post condiciones)</i>	La información buscada por el usuario ha sido encontrada y utilizada
<i>Escenario principal de éxito o flujo básico</i>	<ol style="list-style-type: none"> 1. El usuario está autenticado 2. El usuario elige menú ayuda → ayuda y realiza una búsqueda 3. El usuario sigue los pasos de la ayuda 4. El usuario realiza la tarea elegida sin complicaciones
<i>Extensiones o flujos alternativos</i>	<p>Problemas con el navegador</p> <ol style="list-style-type: none"> b) Reintentar o probar otro navegador disponible c) Actualizar navegador <p>El servicio se bloquea:</p> <ol style="list-style-type: none"> a) Cerrar sesión y volver a iniciar

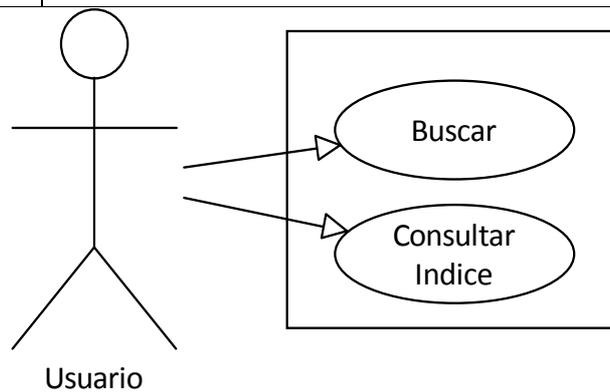


Figura 7. Diagrama del módulo de ayuda.

3.3 Especificación de actores

Las tablas 12 y 13 describen a los actores del sistema.

Tabla 12. Descripción del actor administrador

<i>Descripción:</i>	Persona que administra a los usuarios del servicio
<i>Características:</i>	Persona encargada del servicio, tiene conocimiento total del servicio y de su funcionamiento interno
<i>Relaciones:</i>	Interactúa con el servicio para administrar tanto a usuarios como el hardware; se comunica con los clientes para soporte, quejas y sugerencias.

Tabla 13. Descripción del actor usuario

<i>Descripción:</i>	Persona que hace uso del servicio
<i>Características:</i>	Persona con cuenta de usuario en el servicio, posee conocimientos básicos de computación
<i>Relaciones:</i>	Interactúa con el servicio para monitorear un ambiente

3.4 Diagrama conceptual

La Figura 8 muestra el diagrama conceptual del sistema, en donde el ingreso está restringido por la validación de usuarios. Una vez realizada la validación, el usuario ingresará al sistema encontrándose con un sistema formado por 4 módulos (Panel Monitoreo, Panel Videos, Panel Fotos y Ayuda), éstos a su vez se encuentran subdivididos de acuerdo a los componentes y funciones que realizan cada uno de ellos.

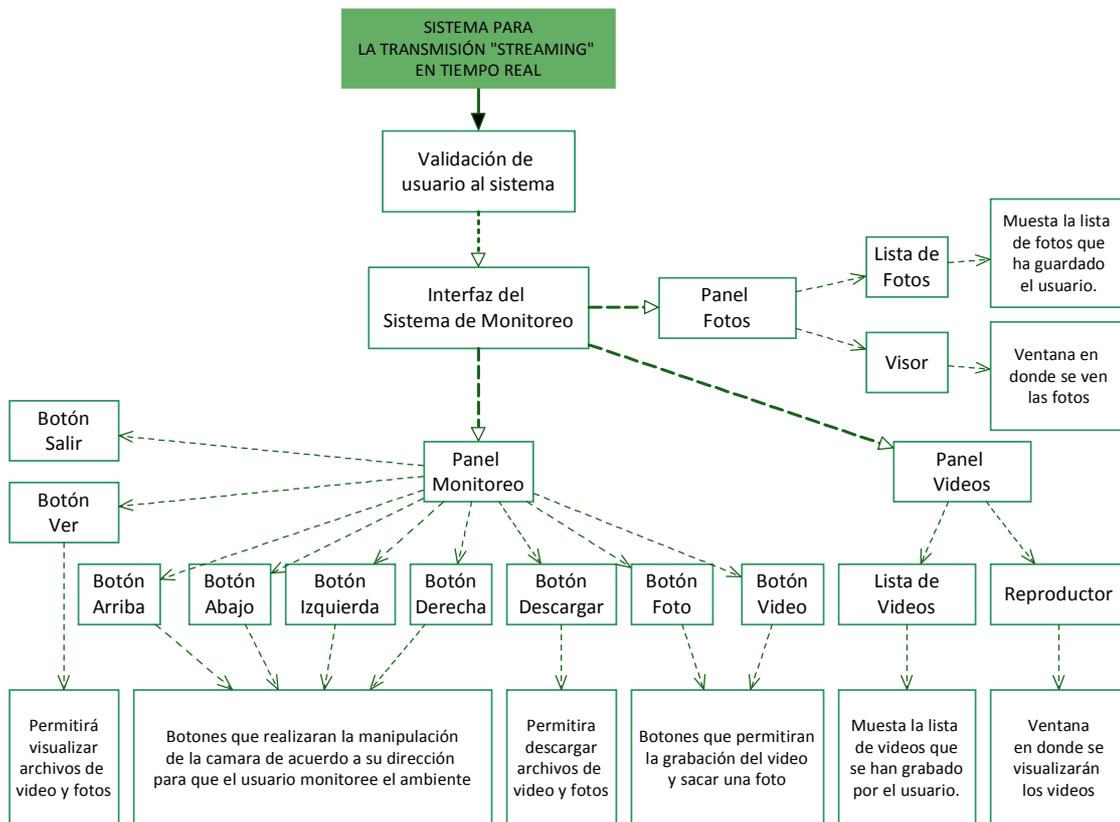


Figura 8. Diagrama conceptual del sistema

3.5 Datos de entrada, datos de salida y descripción por módulo

Las tablas 14 a 18 señalan los datos de entrada y salida de los principales módulos del sistema.

Tabla 14. Descripción del módulo de validación

<i>Descripción</i>	Validará si el usuario está autorizado para usar el servicio
<i>Datos de entrada</i>	Identificador de usuario y de contraseña
<i>Datos de salida</i>	Ingreso a la página web del servicio, si no es usuario permanece la ventana de validación del usuario

Tabla 15. Descripción del módulo de monitoreo

<i>Descripción</i>	Módulo que contendrá botones para poder monitorear al ambiente
<i>Datos de entrada</i>	Datos de la validación aceptados
<i>Datos de salida</i>	Imagen en tiempo real del ambiente. Mensajes de grabación (iniciando grabación, grabación pausada, grabación detenida).

Tabla 16. Descripción del módulo de reproducción de videos.

<i>Descripción</i>	Módulo que contendrá la lista de videos grabados por el usuario así como un visor para la reproducción de videos a reproducir.
<i>Datos de entrada</i>	Selección del archivo a ver.
<i>Datos de salida</i>	Cuadro de diálogo de descarga de un video. Archivos a descargar.

Tabla 17. Descripción del módulo de visualización de fotografías

<i>Descripción</i>	Panel que contendrá la lista de fotos grabadas por el usuario así como un espacio de visualización
<i>Datos de entrada</i>	La foto deseada seleccionada de una lista
<i>Datos de salida</i>	Visualización de la foto Cuadro de diálogo de descarga de una foto

Tabla 18. Descripción del módulo Ayuda.

<i>Descripción</i>	Contendrá la descripción de cada una de las tareas con ejemplos
<i>Datos de entrada</i>	Tarea a consultar
<i>Datos de salida</i>	Descripción y ejemplo de la tarea

La Figura 9 muestra la pantalla para validar al usuario que desee ingresar al servicio, se necesita ingresar el identificador de usuario y una contraseña, después se presionará el botón “Aceptar”.



Figura 9: Interfaz de la validación de usuario

La Figura 10 presenta la interfaz de la pantalla principal del servicio y sus componentes. En la parte superior está el menú principal del servicio, en la parte inferior se encuentra el panel de botones que se explica mas adelante. Se accede a esta pantalla una vez validado el usuario.



Figura 10: Interfaz de la pantalla principal. (Ambiente Monitoreado: Site UPP)

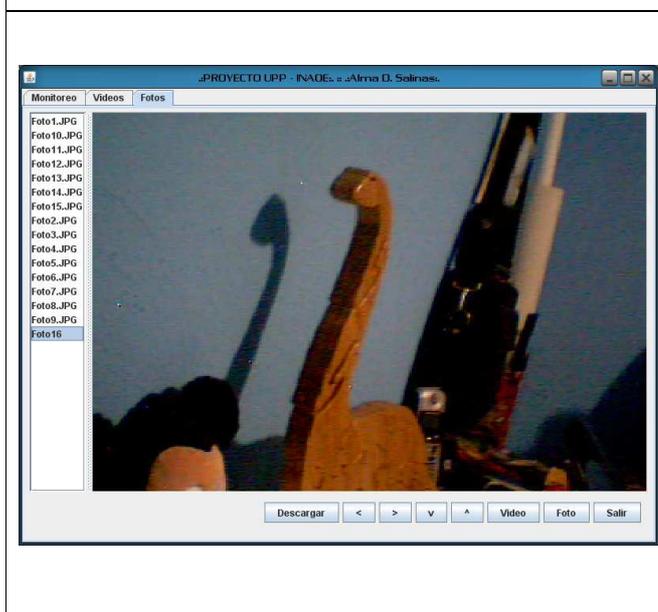
La Tabla 19 describe la función de los componentes de la Figura 10.

Tabla 19. Descripción de componentes de la interfaz

		En estas pestañas se podrán seleccionar las tareas de monitoreo, visualización de videos o fotos
	<p>Nota: El movimiento de la cámara se contempla como trabajo a futuro, estos botones se han creado para la asignación de sus lugares en la interfaz.</p>	Botón que moverá la lente de la cámara hacia arriba
		Botón que moverá la lente de la cámara hacia abajo
		Botón que moverá la lente de la cámara hacia la izquierda
		Botón que moverá la lente de la cámara hacia la derecha
		Botón que permitirá tomar una fotografía
		Botón que permitirá grabar un video del monitoreo que realice el usuario, pausar o detener el video que se esté grabando
		Botón que permitirá descargar una foto o un video seleccionado
		Botón que permitirá visualizar un video o foto guardado en el sistema
		Botón que permitirá salir del sistema



Panel de contenedor “Videos”. Permitirá ver el video seleccionado de la lista junto con los botones de reproducción (pausa, reproducir, volumen, propiedades y barra de proceso).



Panel de contenedor “Fotos”. Permitirá ver las fotos seleccionadas de la lista, así como su descarga

Capítulo 4. Implementación

Este capítulo describe algunas de las funciones desarrolladas para el sistema, así como su utilización. Las funciones para la transmisión de video están implementadas en el lenguaje de programación Java en un Applet, la cual captura y reproduce los datos multimedia.

4.1. Validación de usuario

La validación de usuarios se realiza con el API de Java en donde se crea una ventana que valida al usuario por medio de un identificador de usuario y una contraseña que permiten el ingreso al sistema. Los datos son capturados por medio de un TextField y JPasswordField, se sale o se tiene acceso al sistema por medio de dos botones (Aceptar y Salir).



The image shows a user validation form. It consists of two text input fields. The first field is labeled 'Usuario:' and contains the text 'UsuarioX'. The second field is labeled 'Contraseña' and contains ten dots, indicating a password. Below the fields are two buttons: 'Aceptar' and 'Salir'.

Figura 11: Ejemplo de TextField, JPasswordField y Botones.

4.2. Transferencia de video

La captura de video se realiza mediante el API de Java Media Framework (JMF). JMF permite la captura y reproducción de video y audio en diversos formatos, satisface el cruce entre plataformas, protocolos y contenido neutral, captura y reproduce medios audiovisuales [8]. JMF se utiliza para implementar un reproductor (player) multimedia que recibe y reproduce datos desde fuentes almacenadas localmente o que se están produciendo en tiempo real.

El modelo que utiliza JMF es muy parecido al que utilizan los dispositivos de video. Primero una fuente se encarga de capturar los datos mediante una cámara, (los datos se almacenan como streams). Una vez que los datos son almacenados, se leen y procesan mediante algún dispositivo. Finalmente, éstos se envían a un dispositivo de salida. La herramienta desarrollada con JMF tiene interfaces para capturar y transmitir video. A continuación se describen las clases principales utilizadas.

4.3. La interfaz general

Las siguientes instrucciones fueron usadas para la creación de la interfaz gráfica, se explica:

Validación de usuario: variables de la validación de usuario

JFrame windowVal = new JFrame(".:PROYECTO :: .:Alma D. Salinas.:VALIDA USUARIO"); → Ventana de validación.

JPanel panTitulos = new JPanel (); → Panel contenedor de elementos de validación.

JButton aceptar = new JButton("Aceptar"); → Para los botones de la ventana validación.

JPasswordField JTFcont = new JPasswordField(20); → Recibe la contraseña y oculta los caracteres al digitarlos.

JTextField JTFusu = new JTextField(20); → Recibe el nombre del usuario a validar.

Ventana de Monitoreo: Elementos principales de la ventana principal del sistema

JTabbedPane tabPanel = new JTabbedPane(); → Es el contenedor de los paneles de Monitoreo, Videos, Fotos, cada panel se encuentra en cada pestaña del *JTabbedPane*.

JFrame window = new JFrame(".:PROYECTO :: .:Alma D. Salinas.:MONITOREO"); → Ventana: contenedora de tabPanel y botones (panel de botones).

JPanel objetoPanel = new JPanel(); → Panel: diferentes paneles para contener a los módulos.

JButton → Todos los botones del sistema son de tipo *JButton*.

Image img = null; → Variable para trabajar con una imagen

CaptureDeviceInfo dev = CaptureDeviceManager.getDevice("vfw:Microsoft WDM Image Capture (Win32):0"); → Contiene la información de un dispositivo en particular, en este caso de la cámara.

MediaLocator loc = dev.getLocator(); → Indica la ruta del dispositivo

Component compVisor; → Proporciona componentes visuales y de control (para Player)

public static JList listaVideos = new JList(); → Define las listas de videos y de imágenes.

JSplitPane divVideo = new JSplitPane(JSplitPane.HORIZONTAL_SPLIT); → Un divisor para los paneles de videos y fotos divide el panel entre la lista y el panel que muestra.

JScrollPane scrollPaneVideo, scrollPaneFoto; → es la barra de desplazamiento (Scroll) para las listas de videos y fotos.

4.4. Clase DataSource

La clase DataSource se encarga de encapsular la localización, el protocolo de transferencia y el software necesario para la adquisición de datos. Por lo tanto, una vez obtenido un DataSource, no se puede asociar a otra fuente de datos. Se construye a partir de un URL o un MediaLocator a través de la clase Manager. Un MediaLocator es similar a un URL pero se puede construir aunque no se encuentre instalado en el sistema el manejador para el protocolo especificado, ya que el MediaLocator define la localización de la fuente de datos y el protocolo a utilizar de manera automática.

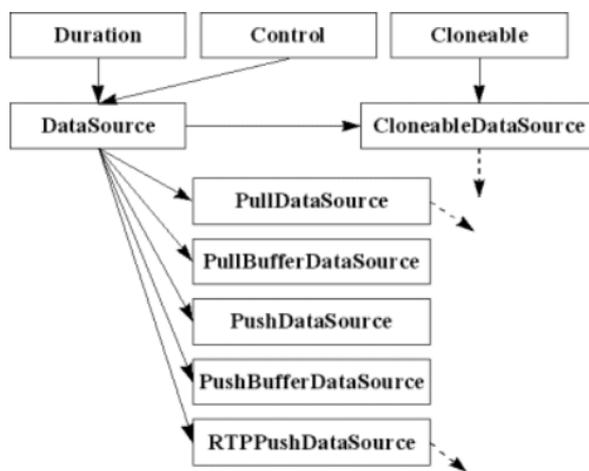


Figura 12.1: Jerarquía de DataSource

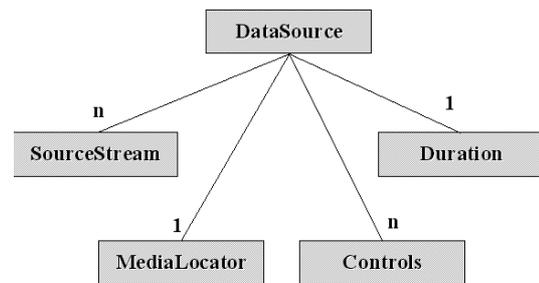


Figura 12.2: Atributos de la clase Data Source

4.5. Clases Player y Processor

Un Player es un objeto encargado de procesar y presentar flujos de datos multimedia. La forma de presentación y los recursos necesarios dependen del tipo de medio que contenga el flujo. Para obtener el flujo de datos se utiliza un objeto DataSource que se enlaza a la entrada del Player. El método createPlayer() recibe un URL o un MediaLocator, construye primero el DataSource y luego el Player.

Los objetos Processor son subclase de Player. El método (createProcessor()) de Processor añade nuevas capacidades, un Processor puede presentar los datos multimedia tal como hace Player, pero la salida puede ser un objeto DataSource que proporciona un “media stream” a otros objetos. Esto permite procesar los datos, cambiar el formato o modificar las características internas del propio proceso.

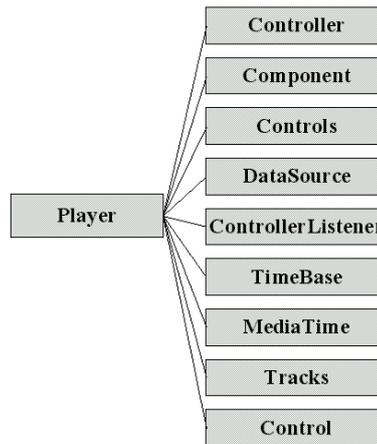


Figura 13.1 Atributos de la clase Player

4.6. Jerarquía de la clase Player

La clase Player es la más importante de las que se describen en este documento, debido a que es la que trata y/o representa los datos multimedia. La Figura 14.1 muestra la jerarquía de esta clase.

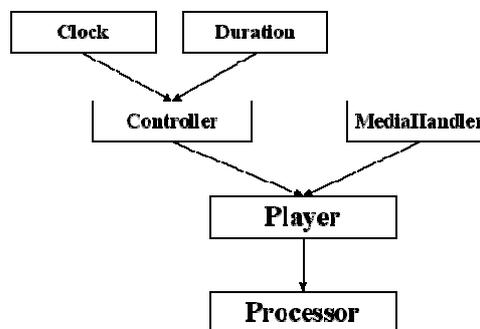


Figura 14.1: Jerarquía de la clase Player

4.7. Clase Datasink

El último paso en el procesamiento de datos multimedia puede ser el almacenamiento o la transmisión. Para ello JMF proporciona la clase DataSink. Un objeto de esta clase se construye a través de la clase Manager usando un DataSource. La función de este objeto es obtener el *media stream* y almacenarlo en un archivo local a través de la red.

4.8. Manager y otras clases de control

Las clases *managers* facilitan el uso de los tipos de objetos. La más básica es *Manager*, ésta es una clase intermedia utilizada en la construcción de los componentes JMF descritos anteriormente (*Player*, *Processor*, *DataSource* y *DataSink*). Con estas clases, JMF permite usar componentes ya existentes y crear nuevas implementaciones. Los siguientes son otros *managers*.

1) *ProcessorModel*.- Es una clase capaz de definir internamente un *Processor*. Cada uno de los *tracks* y tipos de datos internos se usan para la creación de un *Processor* con características definidas o para analizar un *Processor* existente.

2) *Medialocator*.- Define la localización de la fuente de datos y el protocolo a utilizar para obtenerlos.

3) *Component*.- Es una clase AWT *Component*, lo que significa que se pueden incluir en la ventana de una aplicación o en un *Applet*. Al ser componentes definidos en AWT, se pueden cambiar las propiedades y organizarse con objetos *Layout*. Los objetos Control pueden tener este tipo de objetos, los más comunes son *ControlPanelComponent* y *VisualComponent*. Para obtener estos objetos se recurre a los métodos de *Player* (o del *Processor*) *getControlPanelComponent()* y *getVisualComponent()*. *ControlPanelComponent* mostrará al usuario los botones y controles que le sirvan para el manejo de la señal como pueden ser botones de reproducción y pausa o barra de volumen. *VisualComponent* muestra la representación de la propia señal o sus características, así en una señal de video, este objeto es la ventana donde se visualiza el video, en una señal de audio puede verse la forma de onda, un panel de *vúmetros* (instrumento que permite visualizar el nivel de sonido de una señal de audio) o puede no existir.

4.9. Sockets

Los sockets son un sistema de comunicación entre procesos de diferentes máquinas de una red, es decir un punto de comunicación por el cual un proceso puede emitir o recibir información. Utilizan una serie de primitivas para establecer el punto de comunicación, para

conectarse a una máquina remota en un determinado puerto que esté disponible, para escuchar en él, para leer o escribir y publicar información en él, y finalmente para desconectarse.

La Figura 15 muestra el funcionamiento de la conexión de un socket.

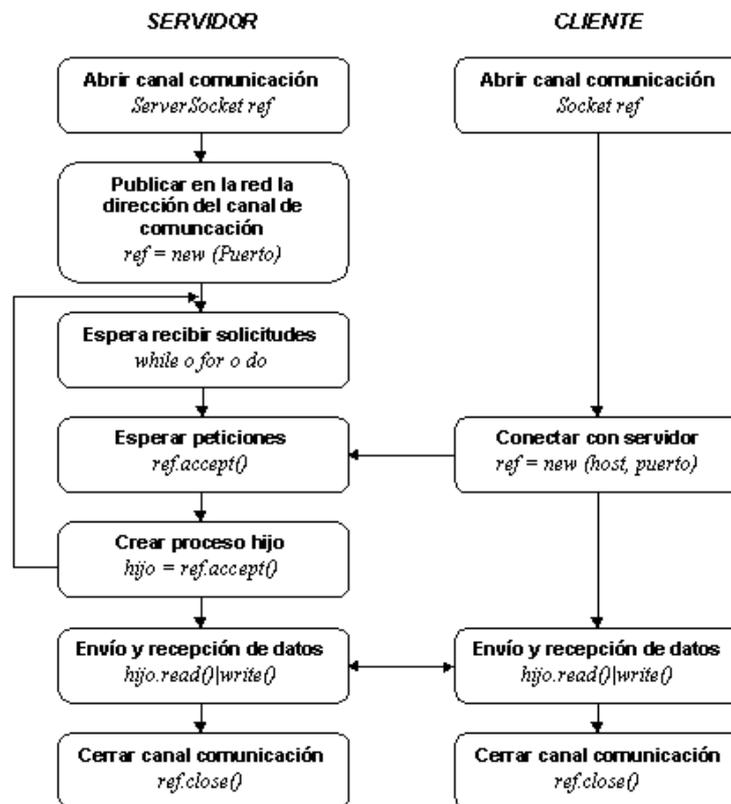


Figura 15: Funcionamiento de los sockets ^[10].

4.10. La clase DataOutputStream

La clase DataOutputStream, heredera de OutputStream, añade a esta la posibilidad de escribir datos "complejos" en un flujo de salida. Los datos "complejos", se refieren al tipo de datos primitivos, pero no restringidos únicamente a bytes y a matrices de bytes.

Mediante la clase DataOutputStream se pueden escribir datos de tipo int, float, double, char, etc. Incluso se pueden escribir algunos objetos, como datos de tipo String en una gran cantidad de formatos.

4.11. La clase `Data InputStream`

La clase `DataInputStream` está preparada para leer datos generados por un objeto `DataOutputStream`. La especificación garantiza que cualquier archivo escrito por un `DataOutputStream`, sobre cualquier plataforma y sistema operativo, será legible correctamente por un `DataInputStream`, sin preocuparse de si las máquinas son "little-endian" o "big-endian".

4.12. `Serializable`

Java facilita el almacenamiento y transmisión del estado de un objeto mediante un mecanismo conocido con el nombre de serialización. La serialización de un objeto consiste en generar una secuencia de bytes para su almacenamiento o transmisión. Después, mediante la deserialización, el estado original del objeto se puede reconstruir. Para que un objeto sea serializable ha de implementar la interfaz `java.io.Serializable` de modo que toda clase que implemente la interfaz podrá transformar sus objetos a cadenas de bytes y viceversa. La Figura 16 muestra cómo se usa esta clase en el sistema.

```
public class ClienteStreaming implements Serializable{
public class ServidorStreaming implements Serializable
{
```

Figura 16: Uso de `Serializable` en el sistema

4.13. Código y pantallas de captura

La identificación del dispositivo con JMF necesita del *manager* `CaptureDeviceManager` para localizarlo. Si se conoce el dispositivo, se puede obtener su información con el método `getDevice()`. De otra forma, se puede llamar al método `getLocator()` del objeto seleccionado para obtener un `MediaLocator` útil para la creación de un `DataSource`, un `Player`, un `Processor` o la localización del dispositivo. La Figura 10 muestra la forma de identificar al dispositivo de video y la localización que éste tiene, como se puede ver, "vfw:Microsoft WDM Image Capture (Win32):0" es el dispositivo a localizar, siendo éste una webcam.

```

|
CaptureDeviceInfo dev = CaptureDeviceManager.getDevice("vfw:Microsoft WDM Image Capture (Win32):0");
MediaLocator loc = dev.getLocator();
```

Figura 17: Código para la detección de la cámara

4.14. Almacenamiento

Para el almacenamiento de los videos es necesario crear un URL o *MediaLocator* de destino, considerando que el *DataSource* es la fuente. Con estos dos objetos se llama al método *createDataSink()* de *Manager* para obtener el objeto destino. Este objeto se abre utilizando el método *open()* y con *start()* inicia la operación de almacenamiento. En el caso de que el *DataSource* sea la salida de un *Processor*, para no perder información es necesario llamar antes del método *start()* de *DataSink* a *start()* de *Processor* (el cual debe estar en el estado *Started*).

Cuando el flujo de datos termina, *DataSink* envía un evento *EndOfStreamEvent* que será necesario atender para destruir el *DataSink* utilizando el método *close()*. En el caso de una captura de datos multimedia, al llamar a los métodos *stop()* y *close()* del *Processor* también se lanza el evento. En el supuesto de saber cuándo se quiere terminar el almacenamiento, no es necesaria la implementación del *Listener*. Las figuras 12 y 13 muestran el almacenamiento que el sistema realiza tanto de imágenes como de video, incluyen los letreros del procesamiento del sistema, así como variables y ubicaciones del almacenamiento de video.

```
General Output
-----Configuration: <Default>-----
INICIA MAIN
INICIA INICIANDO
AGREGA COMP A TABPANEL
CONFIGURA TABPANEL
CONFIGURA BOTONES
AGREGA BOTONES
AGREGA PANELES
TERMINA MAINEntra VIDEO
PLAYER CLOSE: 'TERMINA PLAYER'
PLAYER DEALLOCATE
COMP DISABLE
ENTRA TRY
PARAM: : javax.media.ProcessorModel@1f934ad
PARAMjavax.media.ProcessorModel@1f934ad
PROCESO START
PROCESO: com.sun.media.processor.unknown.Handler@1b16e52
Manager.createRealizedProcessor HECHO
SH: jmapps.util.StateHelper@b2002f
SOURCE HECHO: com.sun.media.multiplexer.BasicMux$BasicMuxDataSource@406199
DESTINO ESTABLECIDO: file://C:/Documents and Settings/Alma/Mis documentos/1 Captura/ServicioWeb/Videol.mov
MEDIALOCATOR DEST: file://C:/Documents and Settings/Alma/Mis documentos/1 Captura/ServicioWeb/Videol.mov
MediaLocator HECHO
numVID: 1
FILEWRITER MANAGER
SOURCE: com.sun.media.multiplexer.BasicMux$BasicMuxDataSource@406199
DEST: file://C:/Documents and Settings/Alma/Mis documentos/1 Captura/ServicioWeb/Videol.mov
Filewriter OPEN
Filewriter OPEN HECHO: video.quicktime
```

Figura 18: Resultado de la captura de un video en una de las pruebas

```

General Output
-----Configuration: <Default>-----
INICIA MAIN
INICIA INICIANDO
AGREGA COMP A TABPANEL
CONFIGURA TABPANEL
CONFIGURA BOTONES
AGREGA BOTONES
AGREGA PANELES
TERMINA MAIN
LA Imagen esta en: C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb\Foto1.JPG
LA Imagen esta en: C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb\Foto2.JPG
LA Imagen esta en: C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb\Foto3.JPG
LA Imagen esta en: C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb\Foto4.JPG
LA Imagen esta en: C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb\Foto5.JPG
Process completed.

```

Figura 19: Resultado de la captura de fotografías en una de las pruebas.

La Figura 13 muestra los archivos almacenados en la carpeta indicada (*C:\Documents and Settings\Alma\Mis documentos\1 Captura\ServicioWeb*)

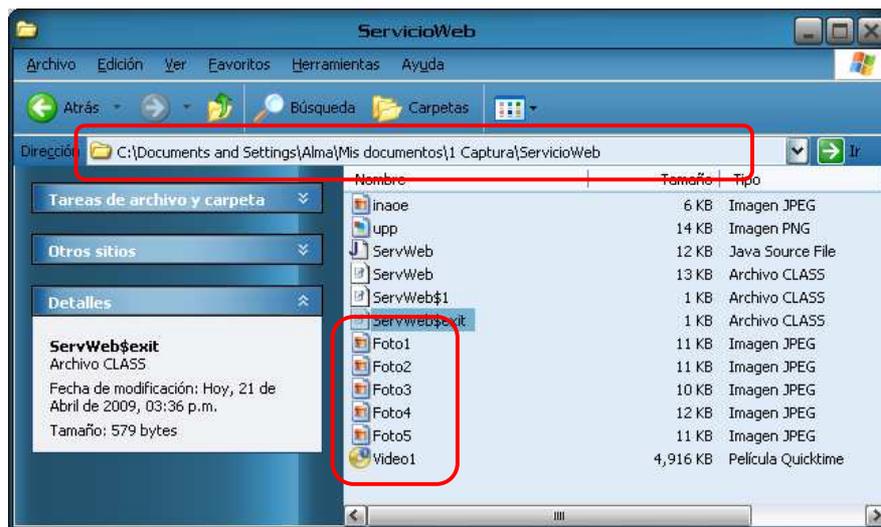


Figura 20: Carpeta con los archivos generados

Para mostrar los archivos, se visualizan en los respectivos paneles de Videos y Fotos de la ventana principal y los nombres de los archivos serán seleccionados de un `JList()`; de `javax.swing.*`; como muestra la Figura 15 y Figura 16.

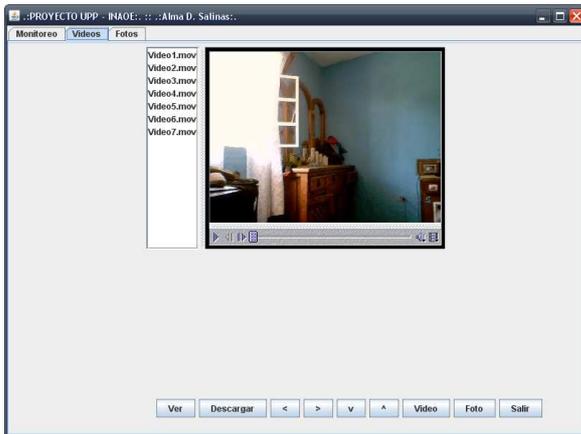


Figura 21: Panel y JList de Videos

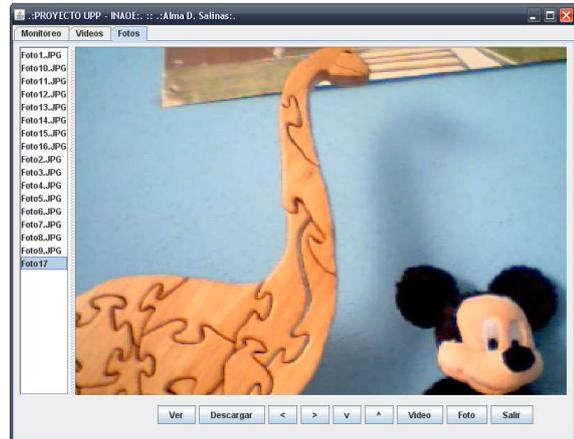


Figura 22 Panel y JList de Fotos

Finalmente, la Figura 14 muestra la ventana generada por el botón de “Descargar”, que sirve para poder descargar el video o foto correspondiente, indicando la ruta destino donde el archivo será guardado.

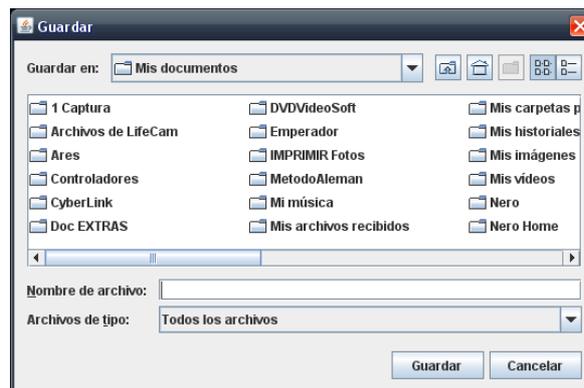


Figura 23: Cuadro de diálogo para descargar las fotos y videos.

4.15. Resultados

Como resultado se tienen las siguientes figuras que muestran tanto al cliente como de servidor, se crearon etiquetas para indicar el proceso del sistema.

```
General Output
1 -----Configuration: <Default>-----
2 Servidor INICIADO
3
4 Cliente conectado
5 La IP de cliente es:192.168.1.64
6 Puerto del cliente:1078
7 Enviando Datos...
8 Cliente desconectado
9
10
11 CLASE RUN
12
13 LLAMANDO A SERV-WEB...
14
15 SALE DE VALIDANDO
16 INICIA FUNCION VALIDADO
17 PLAYER VISOR CREADO
18 PLAYER VISOR INICIADO
19 AGREGA COMP-VISOR A MONITOREO
20 PLAYER REPRODUCTOR CREADO
21 AGREGA COMP-REPRODUCTOR A VIDEOS
22 CONFIGURA BOTONES
23 AGREGÓ BOTONES
24 INICIA TABPANEL
25 CONFIGURA TABPANEL
26 LISTA Videos Almacenadas
27 LISTA Fotos Almacenadas
28 PANEL Videos Almacenados
29 PANEL Fotos Almacenados
30 AGREGA PANELES
31
32 Process completed.
33
```

Figura 24: Resultados en modo consola del servidor

```
General Output
Configuration: <Default>
Cliente conectado
La IP de cliente es:192.168.1.64
El puerto donde escucha es:1078
La Fecha De Conexion Al Servidor: Wed Aug 19 09:01:40 CDT 2009
-i
Process completed.
```

Figura 25: Resultados en modo consola del cliente

Al utilizar el API JMF, fue posible implementar la captura y reproducción de datos multimedia, el resultado obtenido es un sistema de monitoreo cuyas funciones son:

- ✓ Monitoreo remoto
- ✓ Tomar video
- ✓ Sacar fotografías
- ✓ Guardar archivos de videos o fotografías en una carpeta determinada
- ✓ Almacenar los archivos en algún otra carpeta seleccionada por el usuario
- ✓ Visualizar los archivos

Capítulo 5. Conclusiones

La tecnología streaming permitió la implementación del monitoreo remoto del “Sistema Para Transmisión “Streaming” el cual puede ser aplicado como un sistema usado en el área de seguridad.

Se puede concluir que dicho sistema cumple con los objetivos por que el sistema cuenta con una interfaz gráfica para el usuario con el cual es capaz de monitorear un ambiente de manera remota usando la tecnología streaming, realizar un video y tomar imágenes (fotografías) de ese ambiente.

Como trabajo a futuro se implementarán los movimientos de la cámara, donde el usuario pueda ser capaz de manipular libremente el dispositivo ya que en esta etapa solo se menciona el funcionamiento de los botones que ya han sido agregados al sistema, así como la libre manipulación de la grabación ya que la grabación realizada es de solo 2 minutos.

Referencias

[1] Austerberry, David, © 2004. The technology of video and audio streaming. 2º Edición Focal Press.

[2] <http://smilecam.com/home/english/index.asp>
Biometrics & Security Solution SINTEC Corp.
Copyright © 2000
Ingreso: 30 de Mayo de 2008

[3] <http://www.streamspain.es/soporte.asp>
StreamES
Copyright © 2008 Stream UK
Ingreso: 018 de Junio de 2008

[4] <http://www.w3c.es/Divulgacion/GuiasBreves/ServiciosWeb>
The World Wide Web Consortium (W3C)
Copyright © 1994-2005 W3C® (MIT, ERCIM, Keio).
Ingreso: 13 Junio de 2008

[5] <http://www.pergaminovirtual.com.ar/definicion/Servidor.html>
Pergaminovirtual.com
Copyright © 1998-2007
Ingreso: 13 Junio de 2008

[6] <http://electronics.howstuffworks.com/streaming-video-and-audio4.htm>
HowStuffWorks
Copyright © 1998-2008 HowStuffWorks, Inc.
Ingreso: 01 Julio de 2008

[7] http://www.portalplanetasedna.com.ar/web_internet.htm
Control-Systems.NET
Copyright © Control-Systems.NET. All right reserved.
Ingreso: 05 Julio de 2008

[8] Garcia Avenue, Mountain View, 1998-1999 Sun Microsystems, Inc. *JAVA^a MEDIA FRAMEWORK API GUIDE*, (<http://java.sun.com/javase/technologies/desktop/media/jmf/>).

[9] <http://streamingreal.com/preguntasfrecuentes.php>
Colombia Redes.
Copyright © Todos los Derechos Reservados@Control Systems.NET, 2008.
Ingreso: 12 Enero de 2009

[10] http://pisuerga.inf.ubu.es/lsi/Invest/Java/Tuto/V_2.htm
Guía de Iniciación al Lenguaje JAVA
1997-1998 (Primera promoción).
Ingreso: 11 Agosto de 2009.