

*Universidad de las Ciencias Informáticas
FACULTAD 6*



Título: *Cliente de grabación y reporte para el sistema de video vigilancia Suria Web.*

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

Autor(a): *Yurisdelvys Almarales Ramos.*

Tutor(es): *Ing. Olga María Rivera Correa.*

Junio, 2012

FRASE

"La posibilidad de realizar un sueño es lo que hace que la vida sea interesante."

Paulo Coelho.

DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos 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 21 días del mes de junio del año 2012.

Yurisdelvys Almarales Ramos.

Ing. Olga María Rivera Correa.

Firma del Autor

Firma del Tutor

DATOS DE CONTACTO

DATOS DE CONTACTO

Tutora: Ing. Olga María Rivera Correa (omrivera@uci.cu)

Profesor instructor graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas (UCI).

AGRADECIMIENTOS

AGRADECIMIENTOS

A mi mamá, por ser una persona especial en mi vida, por su amor infinito, por sus consejos, su apoyo y todas las cosas bellas que ha hecho por mí sin pedir nada a cambio, por ser la mejor madre del mundo, te quiero mamá.

A mi papá por depositar toda su confianza en mí, por su preocupación, por tratar siempre de darme lo mejor, por ser el mejor padre del mundo, te quiero papá.

A mi abuela Delvis, por ser la principal promotora de que escogiera esta carrera, por tenerme siempre en su corazón, por su infinito amor, por ser la mejor de las madres, por estar siempre dispuesta a ayudarme, todo lo que soy hoy te lo debo a ti, te quiero mami.

A mi tía Marlenis, por ser la mejor de todas las tías, por mimarme siempre, por ser otra madre para mí, por su apoyo incondicional, por quererme siempre y por todo su esfuerzo para ayudarme a ser cada día mejor.

A mis hermanos Yaki y Maikelito, por su amor y cariño, por respetarme siempre y por tener siempre una sonrisa para mí, los quiero.

A mi prima Yami, por estar siempre al pendiente de mí, por su apoyo y ayuda, por su profundo cariño, aunque estés lejos siempre vas a estar presente en mi corazón, te quiero.

A todos mis tíos, primos y la familia en general, por su apoyo incondicional por tratarme como la niña linda de la familia, por su inmenso amor y cariño.

A mi tutora Olga, por su ayuda y apoyo incondicional, por sus consejos oportunos, por enseñarme a ser cada día mejor, por ponerse nerviosa junto conmigo, por ser un ejemplo a seguir, por todos los momentos de duro trabajo que dedicó a este trabajo.

A Enrique, por no dejarme sola nunca, por brindarme todo su amor y cariño, por el esfuerzo realizado, por tener paciencia conmigo durante estos cinco años, por ayudarme a ver las cosas desde otro punto de vista.

A mis amigas Yurisel y Linette, por demostrarme que no importa lo lejos que estemos la amistad que nos une es más fuerte, por ser mis hermanas, por estar siempre a mi lado sin importar las circunstancias, por todo eso y mucho más las quiero.

AGRADECIMIENTOS

A mis amigas Magela y Yanet, por los buenos momentos que pasamos juntas, por los consejos y su apoyo incondicional, por ser las mejores amigas del mundo, por demostrarme que la amistad es más fuerte que nada.

A mis compañeros del antiguo 9109, en especial a Grethell, Rubi, Alexaidis, Guillermo y Marcelino, por estar siempre dispuestos a ayudar, por cada momento que pasamos juntos, siempre van a estar en mi corazón.

A las chicas del apto 101 102, por ser como otra familia para mí, por estar siempre dispuestas a escuchar mis exposiciones, por darme apoyo en los momentos que las necesité, por las pijamadas y todos los lindos momentos que pasamos juntas.

A las "Tropas" por permitirme ser parte del grupo, por todos los maravillosos momentos que pasamos juntos, por todas las misiones que cumplimos y por las reflexiones oportunas después de cada misión, los quiero.

A mis amigos del politécnico, en especial a Frank, por no olvidarse de mí y por los buenos momentos que pasamos juntos.

A mis vecinos, en especial a mi "tío" Alexis, por estar siempre al pendiente de mis estudios, por quererme tanto.

A Lourdes, Dailín y Roxana, por sus llamadas, por su preocupación y por su apoyo incondicional.

A los miembros del proyecto Video Vigilancia por su ayuda, por sus críticas y recomendaciones.

A los miembros del tribunal por sus recomendaciones, sugerencias y críticas oportunas que permitieron aumentar mi formación como profesional.

A todos los que de una forma u otra hicieron posible este sueño.

Gracias.

DEDICATORIA

DEDICATORIA

A mi mamá por ser amiga, compañera y madre, por su esfuerzo para que yo cumpliera mi sueño, por enseñarme a ser una persona de bien, por confiar en mí y por cuidarme siempre. A ti mamá por darme la vida. Te adoro mamá.

A mi papá, por confiar siempre en mí, por su preocupación y dedicación. A ti papá por quererme tanto. A mi abuela Delvis, por ser la luz de mis ojos, por darme tanto amor y cariño. A ti mami por ser otra madre para mí, por todo el esfuerzo que ha realizado para ayudarme a cumplir mis sueños, por nunca decir que no a nada, quiero que sepas que ni la mitad de estas palabras expresan todo lo que significas para mí y lo que te quiero. Te amo mami.

A mi tía Marlenis, por quererme tanto, por ser otra madre para mí, por todos tus consejos. A ti tía por dedicar tu tiempo a cuidarme, por ser una fuente de amor infinito, siempre estas presente en mi corazón. Te quiero.

A mis hermanos Yaki y Maikelito, por traerme tanta felicidad en la vida, por tenerme presente siempre.

A ustedes mis hermanos por tenerme siempre como ejemplo, lo que es un orgullo para mí. Los adoro.

A toda mi familia, por su apoyo incondicional, por quererme tanto y por estar siempre dispuestos ayudarme en todo. Los quiero.

Yuri.

RESUMEN

RESUMEN

En la actualidad, los sistemas de vigilancia mediante el uso de cámaras se han convertido en un fuerte apoyo para la seguridad, su incorporación y aplicación en el mercado va dirigida a asegurar un amplio espectro de ambientes y lugares. Se incluye entre sus virtudes ejercer una vigilancia preventiva mediante el registro visual de sucesos, permitiendo la grabación y recuperación de hechos delictivos. Trayendo consigo un gran avance en el mundo de los sistemas de video vigilancia.

El presente trabajo tiene como objetivo implementar una aplicación sobre plataforma web para facilitar la gestión de las grabaciones que se realizan a través de las cámaras y brindar un sistema de reporte que facilite el acceso a la información deseada. Como resultado, se obtiene una interfaz web que permite la grabación de múltiples flujos de video en tiempo real, facilitando en gran medida el control y vigilancia en las principales áreas de las entidades donde se despliegue dicho sistema. Además de brindarle al usuario una forma fácil y sencilla para obtener la información deseada de las acciones que se realizan en el sistema, mediante los reportes que se generan a partir de una fecha especificada. De forma general esta aplicación trae como ventaja un fácil acceso al cliente de grabación y reporte, ya que se podrá acceder a información referente a estos módulos desde cualquier lugar con acceso a la red donde se encuentre desplegada, trayendo consigo una mayor extensibilidad y explotación del producto desarrollado.

PALABRAS CLAVE

Cámaras, Grabación, Reporte, Video Vigilancia

ÍNDICE

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES EXISTENTES.	5
1.1. Introducción.....	5
1.2. Términos asociados a la investigación.....	5
1.2.1. Relación entre Cámara IP, Protocolo IP y Dirección IP.....	5
1.2.2. Tecnología IP y Analógica.....	5
1.2.3. Video vigilancia y video vigilancia IP.....	6
1.2.4. Sistema de Reporte.....	6
1.2.5. Servicio.....	7
1.2.6. Servicio de Grabación.....	7
1.3. Descripción de la situación problemática.....	8
1.4. Características del sistema Suria Vision.....	9
1.5. Análisis sobre soluciones existentes y sus principales limitantes.....	10
1.5.1. Sistemas de video vigilancia basados en la web.....	10
1.5.2. Sistemas de video vigilancia en Cuba.....	11
1.5.3. Conclusiones sobre las Soluciones Existentes.....	11
1.6. Conclusiones.....	12
CAPÍTULO 2. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE....	13
2.1. Introducción.....	13
2.2. Características de los lenguajes de programación a utilizar.....	13
2.3. Características de la tecnología a utilizar.....	13
2.4. Características de la herramienta CASE a utilizar.....	14
2.5. Características de la herramienta para el control de versiones a utilizar.....	15
2.6. Características del IDE desarrollo a utilizar.....	15
2.7. Protocolos a utilizar para la comunicación.....	16
2.8. Bibliotecas a utilizar para el desarrollo de la interfaz de usuario.....	17
2.9. Características de SQLite como sistema gestor de Base de Datos (SGBD).....	18
2.10. Características del lenguaje de modelado para la implementación del sistema.....	18
2.11. Metodología de desarrollo de software a utilizar en la implementación del sistema.....	19
2.12. Conclusiones.....	22
CAPÍTULO 3: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.	23
3.1. Introducción.....	23
3.2. Modelo de Dominio.....	23
3.3. Descripción de la propuesta del sistema.....	23
3.4. Diagrama del Modelo de Dominio.....	24
3.5. Descripción de las clases.....	24
3.6. Requisitos Funcionales del Sistema.....	25
3.7. Requisitos No Funcionales del Sistema.....	26
3.8. Descripción del sistema.....	28
3.8.1. Definición de los actores del sistema.....	28
3.8.2. Listado de los Casos de Uso.....	28
3.8.3. Diagrama de Casos de Uso del sistema.....	29
3.8.4. Descripción textual de los principales casos de uso del sistema.....	29
3.9. Conclusiones.....	43

ÍNDICE

CAPÍTULO 4. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.....	45
4.1. Introducción.....	45
4.2. Arquitectura.....	45
4.3. Arquitectura del Módulo Web.....	45
4.4. Patrones de Diseño de Software.....	46
4.5. Análisis y Diseño.....	47
4.5.1. Diagramas del análisis.....	47
4.5.2. Modelo de Diseño.....	49
4.6. Modelo de Implementación.....	51
4.6.1. Diagrama de Despliegue.....	51
4.6.2. Diagrama de Componentes.....	51
4.7. Métodos de Pruebas.....	52
4.7.1. Pruebas de Caja Negra.....	52
4.7.2. Pruebas de Usabilidad.....	59
4.8. Conclusiones.....	60
CONCLUSIONES GENERALES.....	61
RECOMENDACIONES.....	62
REFERENCIAS BIBLIOGRÁFICAS.....	63
BIBLIOGRAFÍA.....	66
GLOSARIO DE TÉRMINOS.....	69

ÍNDICE DE TABLAS Y FIGURAS

ÍNDICE DE FIGURAS

Figura 1. Diagrama Modelo de Dominio.....	24
Figura 2. Diagrama de Casos de Uso del Sistema.....	29
Figura 3. Representación del patrón arquitectónico dos capas en la aplicación.....	46
Figura 4. Diagrama de Análisis.CU Gestionar Calendario de Grabación.....	48
Figura 5. Diagrama de Análisis.CU Grabar Flujo de Video.....	48
Figura 6. Diagrama de Análisis.CU Visualizar Cámaras.....	48
Figura 7. Diagrama de Clases de Diseño.CU Gestionar Calendario de Grabación.	49
Figura 8. Diagrama de Clases de Diseño.CU Grabar Flujo de Video.	50
Figura 9. Diagrama de Clases de Diseño.CU Visualizar Calendario de Grabación.	50
Figura 10. Diagrama de Despliegue.....	51
Figura 11. Diagrama de Componentes.	52
Figura 12.Representación de las pruebas de caja negra.	59
Figura 13. Representación de las pruebas de usabilidad.....	60

ÍNDICE DE TABLAS

Tabla 1. Actores Definidos del Sistema.....	28
Tabla 2. Descripción del Caso de Uso Gestionar Calendario de Grabación.....	38
Tabla 3. Descripción del Caso de Uso Grabar Flujo de Video.....	40
Tabla 4. Descripción del Caso de Uso Visualizar Cámaras.....	41
Tabla 5. Descripción del Caso de Uso Visualizar Calendario de Grabación.....	43
Tabla 6.Pruebas de caja negra.	58
Tabla 7.Pruebas de usabilidad.....	59

INTRODUCCIÓN

INTRODUCCIÓN

Desde la antigüedad el hombre siempre ha tratado de satisfacer sus necesidades básicas, esto lo ha llevado a una constante evolución, para de esta forma poder mantener su supervivencia. Sin embargo también le ha surgido la necesidad de preservar su seguridad y cuidar sus bienes, evitando que estos sean dañados o utilizados por otras personas sin previa autorización, logrando así un mayor desarrollo social. De esta forma surgió la vigilancia, realizada por personas encargadas de custodiar y mantener el orden del lugar o bien asignado.

Con el transcurrir de los años y el continuo avance tecnológico el ser humano se vio en la necesidad de crear sistemas que le permitieran mantener el control de todo lo que lo rodea, surgiendo así el término de **video vigilancia**, que no es más que toda aquella actividad que supone la colocación de una cámara de grabación, fija o móvil, con la finalidad de garantizar la seguridad de una instalación o de las personas, asegurando el correcto desempeño de las tareas en el entorno laboral, siendo de gran utilidad en diversos ámbitos (1).

Hay informes que indican que desde el año 1965 en los Estados Unidos la policía había estado usando vigilancia por video en lugares públicos y ya en el siguiente año se instalaron cámaras en áreas estratégicas de Nueva York, no pasó mucho tiempo para que se extendiera a otras ciudades. Se realizaba con el uso de Circuitos Cerrado de Televisión (CCTV). Lo que demuestra la amplia aceptación de los usuarios a estos sistemas por su valor y beneficios al brindar un seguimiento en tiempo real del entorno de una instalación, las personas y los bienes, posibilitando eventos de grabación para una posterior investigación y disminuyendo los riesgos de hechos delictivos (2).

En la década de los 70, con el surgimiento de los sistemas de video vigilancia, se logra un avance decisivo que apoya la seguridad de determinadas áreas importantes para las grandes empresas y negocios existentes (3). Estos sistemas han evolucionado considerablemente pasando por varias generaciones donde en la primera generación se emplean señales y ¹transmisión analógica, la segunda generación se basa en métodos de procesamiento y comunicación híbridos analógico-digitales, o completamente digitales. Los sistemas de tercera generación aprovecharán el progreso de las redes de ordenadores de bajo coste y alto rendimiento, y las comunicaciones multimedia fijas y móviles, se trabaja también en obtener muestras de videos en tiempo real (4).

En la actualidad, los sistemas de vigilancia mediante el uso de cámaras, se han convertido en un fuerte apoyo para la seguridad integral, incluyendo entre sus virtudes ejercer una vigilancia preventiva mediante el registro visual de sucesos. Su incorporación y aplicación en el mercado va dirigida a asegurar un amplio espectro de ambientes y lugares tales como: centros comerciales, supermercados,

¹ **Transmisión Analógica:** Consiste en el envío de información en forma de ondas, a través de un medio de transmisión físico.

INTRODUCCIÓN

aeropuertos, viviendas particulares, vías públicas, centros de eventos, transporte público, establecimientos educacionales y empresas de todo tipo (2).

En 1996, la compañía líder a nivel mundial en la producción de sistemas de video vigilancia integrados Axis², muestra al mundo la primera solución de video vigilancia utilizando una cámara de red. Esta solución rompió con las barreras de video vigilancia tradicional e hizo que estos sistemas se hicieran mucho más comunes y populares en la sociedad. La demanda social produjo un enriquecimiento del mercado con potentes soluciones basadas en esta tecnología, las cuales además de ofrecer más funcionalidades y mayor calidad en la visualización de video, eran de costo asequible, por lo que su uso se extendió a sitios como escuelas, oficinas, lugares públicos y hogares (5).

En Cuba estos avances tecnológicos no se han mantenido al margen, pues también se ha visto en la necesidad de comenzar con la investigación en el tema. Algunas empresas hacen inversiones con entidades extranjeras para implantar sistemas de este tipo, actualmente se han instalado en tiendas y centros de trabajo para mantener una seguridad más eficiente, permitiendo detectar todo el personal que se encuentra en la misma y de esta forma realizar un mejor control de todo lo que acontece en el lugar donde se encuentren instalados los mismos.

En la Universidad de Ciencias Informáticas(UCI), en el Centro de Desarrollo GEySED (Geoinformática y Señales Digitales), específicamente en el Departamento de Señales Digitales existe un proyecto productivo en el cual se realizó un sistema de video vigilancia, el mismo lleva como nombre Suria Vision, es una versión de escritorio que cuenta con cuatro componentes fundamentales: el visor o estación de monitoreo, que es el módulo responsable de visualizar los flujos de videos capturados a través de las cámaras, el grabador es el módulo encargado de guardar los flujos de videos obtenidos por las cámaras, el recuperador destinado a recuperar los videos almacenados en los servidores y el gestor que su función es controlar toda la información e interactuar directamente con la base de datos.

Una de las ventajas que tiene Suria Vision es que tiene una arquitectura que permite la incorporación de una alta gama de cámaras. En aras de facilitar el acceso y el intercambio de información del sistema creado, este proyecto se da la tarea de crear una versión web. Desarrollándose inicialmente una primera versión del Visor Web. Luego surge la necesidad de completar el sistema, pues se necesita prestar el servicio de grabación para poder almacenar los flujos de videos que se obtienen a través de las cámaras, brindar la posibilidad de realizar la grabación por calendario o manualmente; además de un sistema de reportes que facilite controlar y obtener la información deseada.

La situación descrita anteriormente lleva a plantearse el siguiente **problema a resolver** ¿Cómo facilitar el acceso a los reportes y servicio de grabación, para obtener un sistema de video vigilancia funcional,

² **Axis:** compañía de cámaras de video vigilancia. (www.axis.com)

INTRODUCCIÓN

desde cualquier lugar con acceso a la red? para llevar un estudio de la problemática y poder desarrollar la investigación se toma como **objeto de estudio**: la informatización de la video vigilancia enmarcándose como **campo de acción**: los procesos de planificación y grabación en el sistema de video vigilancia Suria Vision.

Para limitar el alcance de la investigación y darle solución al problema planteado se concreta como **objetivo general**: Desarrollar una aplicación sobre plataforma web para los servicios de grabación y visualización de reportes del sistema de video vigilancia. Como **idea a defender** se plantea que: “El desarrollo de una aplicación web como módulo integrado del sistema Suria Vision, permitirá aumentar la prestación de servicios, brindando mayor accesibilidad y extensibilidad al producto”.

Las **tareas de investigación** a cumplir son:

1. Estudio del arte de la investigación.
 - 1.1 Definir conceptos asociados al sistema de video vigilancia.
 - 1.2 Definir características de otras aplicaciones que existen de video vigilancia.
2. Definir herramientas y tecnologías a utilizar para la realización de la aplicación.
3. Realizar el Modelo de Negocio del sistema de video vigilancia Suria Vision.
 - 3.1 Realizar Modelo de Dominio del sistema de video vigilancia Suria Vision.
4. Efectuar el levantamiento de requisitos del sistema de video vigilancia Suria Vision.
5. Identificar Casos de Uso del sistema de video vigilancia Suria Vision.
6. Realizar Análisis y Diseño del sistema de video vigilancia Suria Vision.
 - 6.1 Análisis y Diseño del servicio de grabación.
 - 6.2 Análisis y Diseño del servicio de reporte.
7. Implementar funcionalidades del sistema de video vigilancia.
 - 7.1 Implementar servicio de grabación.
 - 7.2 Implementar servicio de reportes.
8. Realizar pruebas de caja negra.

Los **métodos científicos de investigación** que se utilizarán para darle cumplimiento a los objetivos de este trabajo son:

Métodos teóricos:

1. Analítico-Sintético: este método permitirá estudiar las teorías y documentación existente sobre el tema y de esta forma se podrá realizar un resumen sobre los elementos más importantes para la investigación, la tecnología en la implementación de sistemas de video vigilancia web será objeto de estudio en la presente investigación.

INTRODUCCIÓN

2. Análisis histórico-lógico: este método permitirá estudiar los sistemas existentes de video vigilancia web, sus antecedentes y su evolución a lo largo de la historia.
3. Modelación: mediante este método se realizarán los modelos que especifican las características que va a desarrollar la aplicación y que permitirán una mejor comprensión a los desarrolladores para una posterior implementación.

Métodos Empíricos.

Observación: este método permitirá realizar valoraciones para obtener información sobre el funcionamiento de sistemas similares al que se desarrolla, para tener una idea de cómo tiene que ser el sistema.

El siguiente trabajo de diploma estará dividido en 4 capítulos fundamentales:

Capítulo I: Fundamentación teórica. Valoración sobre aplicaciones existentes. En este capítulo, se fundamentan términos técnicos de importancia para la investigación. Se realiza un estudio del arte sobre sistemas que existen en Cuba y el mundo con características similares a las requeridas para la investigación.

Capítulo II: Herramientas y tecnologías para el desarrollo del software. En este capítulo se presentan las tecnologías, herramientas y lenguajes de programación que se utilizarán para el desarrollo de la aplicación.

Capítulo III: Presentación de la solución propuesta. En este capítulo se describen los procesos actuales a través del modelado del dominio, se seleccionan los requisitos funcionales y no funcionales, los casos de uso del sistema, así como, los diagramas correspondientes que lo modelan.

Capítulo IV: Construcción de la solución propuesta. En este capítulo se plantea la construcción propuesta en el capítulo anterior, en función de los Diagramas de Clases del Diseño, Modelo de despliegue y Modelo de implementación, además se muestran los resultados de las pruebas realizadas a la aplicación desarrollada.

CAPÍTULO I.FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES EXISTENTES.

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES EXISTENTES.

1.1. Introducción.

En este capítulo se tiene como objetivo realizar una introducción al tema de la investigación, detallando el contenido teórico necesario para la comprensión del funcionamiento del software. Se realizará un estudio acerca de las aplicaciones de video vigilancia web existente en Cuba y el mundo, se hará referencia a un conjunto de conceptos asociados al dominio del problema a resolver.

1.2. Términos asociados a la investigación.

Los términos incluidos en este epígrafe fueron seleccionados por su gran importancia en el estudio del campo de video vigilancia. Es necesario conocer su significado para llegar a entender el tema tratado en la investigación.

1.2.1. Relación entre Cámara IP, Protocolo IP y Dirección IP.

Una **cámara IP** (IP es un acrónimo para Protocolo de Internet por sus siglas en ingles, Internet Protocol) es una cámara que se conecta directamente a la red empleando el protocolo IP y un servidor web. Un **protocolo** es un conjunto de reglas usadas por computadoras para comunicarse unas con otras a través de una red (6).

El **protocolo IP** es parte de la capa de Internet del conjunto de protocolos TCP/IP. Es uno de los protocolos de Internet más importantes ya que permite el desarrollo y transporte de datagramas³ de IP (paquetes de datos), aunque sin garantizar su "entrega" (6) .Se le asigna a la cámara un número único e irrepetible de forma tal que se pueda acceder a sus funciones desde la red, lo cual se llama **dirección IP**, para ser empleadas por aplicaciones capaces de detectarla y emplearla (7).

1.2.2. Tecnología IP y Analógica.

La **tecnología IP** permite resoluciones más altas dependiendo del producto, con lo cual resulta más fácil apreciar los detalles, hay menos ruido en las líneas (de video y control) durante la transmisión de las señales. En los sistemas completamente digitales, la capacidad de almacenamiento de video depende únicamente del tamaño de las unidades de disco duro del servidor (8). Se pueden hacer tareas muy rápidamente, muy exactas, muy precisas y sin detenerse. Las informaciones son almacenadas en secuencias de ceros y unos (secuencias de bits) empleando el protocolo IP para la

³ **Datagramas:** Es la estructura interna de un paquete de datos que se transfiere en una conexión.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

comunicación entre las unidades que componen el sistema a diferencia de la **tecnología analógica** es una señal que varía continuamente, es muy difícil almacenar, manipular, comparar, calcular y recuperar información con exactitud cuando esta ha sido guardada. El almacenamiento y la transmisión de informaciones se dan por medio de variaciones de frecuencia o amplitud de señales eléctricas (9). Esta tecnología requiere una infraestructura que utiliza un sistema de cable coaxial, que es un tipo de cable para transportar señales eléctricas de alta frecuencia, con una cobertura metálica que rodea a un alambre central, esta protege la señal del alambre interior contra interferencias eléctricas (10). Este cable fue diseñado, en el campo de video vigilancia, para transmisiones punto a punto de video desde una cámara hasta una grabadora en el mismo sitio.

1.2.3. *Video vigilancia y video vigilancia IP.*

Actualmente el término de video vigilancia es usado con frecuencia, debido al incremento de actos delictivos en la sociedad, especialmente por entidades jurídicas que manejan este tema con regularidad.

La constitución española define un sistema de **video vigilancia** como *“todo software instalado en un espacio público o privado, para el cuidado y protección de personas y bienes contra intrusión, agresión, robo o hurto. Permitiendo obtener imágenes y sonidos que puedan establecer pautas de comportamiento que supongan intervención policial o afecten a la seguridad ciudadana”* (11).

Empresas de software como Axis expresan: la **video vigilancia** hace referencia a la supervisión local y/o remota de imágenes de video captadas por cámaras (y sonido si las cámaras integran micrófonos). Las imágenes de video vigilancia pueden ser emitidas en tiempo real y/o grabadas (5). La **video vigilancia IP** es una efectiva solución de seguridad que ofrece monitorización y control avanzado en sistemas de seguridad, emplea tecnología analógica para efectuar la vigilancia. Ambos sistemas de video vigilancia tienen básicamente el mismo funcionamiento y objetivo, y pueden incluso coexistir en una misma instalación de video vigilancia (12).

De acuerdo a lo antes planteado puede decirse que la video vigilancia no es más que la práctica de visualizar, grabar, clasificar y almacenar imágenes de personas en espacios públicos y privados. Se refiere a cualquier tecnología cuya finalidad sea detectar, observar, copiar o registrar movimiento, imágenes, sonidos o el estado de una persona.

1.2.4. *Sistema de Reporte.*

Los **reportes** son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado de los datos para mostrarlos por medio de un diseño

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

atractivo y que sea fácil de interpretar por los usuarios. El reporte, de esta forma, confiere una mayor utilidad a los datos. Tienen diversos niveles de complejidad, desde una lista o enumeración hasta gráficos mucho más desarrollados, permiten la creación de etiquetas y la elaboración de facturas, entre otras tareas (10).

Según estudios realizados un reporte es un documento de texto que muestra los datos de una base de datos del modo que se especifique. El reporte se puede diseñar para que muestre los datos más actuales o los datos válidos en el momento de crear el reporte. Los reportes se utilizan primordialmente para presentar, resumir e imprimir los datos de la forma que resulte más apropiada para cada proyecto. Permite realizar impresiones personalizadas así también como etiquetas. Se pueden crear reportes que incorporen cálculos basados en los datos de las tablas para mostrar resultados totales o promedios o bien para generar catálogos (13).

Generalmente, el propósito del reporte es informar. Sin embargo, los reportes pueden incluir elementos, tales como recomendaciones, sugerencias u otras conclusiones que indican posibles acciones futuras que el lector del reporte puede adoptar.

1.2.5. Servicio.

Un **servicio** es el conjunto de actividades identificables e intangibles, no resulta necesariamente la propiedad de algo, son el resultado de esfuerzos humanos o mecánicos que producen un hecho, un desempeño o un esfuerzo que implican generalmente la participación del cliente y que no es posible poseer físicamente, ni transportarlos o almacenarlos, pero que pueden ser ofrecidos (14).

1.2.6. Servicio de Grabación.

El **servicio de grabación** requiere de una cámara para grabar todos los flujos de video que se definan, estos consisten en la captación, grabación, procesamiento, almacenamiento, transmisión y reconstrucción por medios electrónicos digitales o analógicos de una secuencia de imágenes que representan escenas en movimiento. Tiene la capacidad de grabar las imágenes captadas por las cámaras. La grabación puede ser de forma continua, o ser activada por detección de movimiento, programación horaria, etc. La grabación de video y la posterior visualización, se puede realizar de forma local, o remota a través de internet (15).

Teniendo en cuenta lo antes planteado se puede decir que el servicio de grabación consiste en la actividad mediante la cual se realiza una grabación audiovisual de un evento o acto que se está realizando para su posterior visualización. En la grabación quedan señales registradas de las acciones que se ejecutan y las mismas son adquiridas mediante equipos informáticos.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

1.3. Descripción de la situación problemática.

En la actualidad Cuba se encuentra en proceso de informatización de la sociedad, para de esta manera brindarles una mejor forma de vida a los ciudadanos y lograr mayor eficiencia en todo lo que se realiza. Por lo que el país se ha visto en la necesidad de adquirir varios software con el fin de emplearlos en diferentes esferas, uno de estos son los sistemas de video vigilancia los cuales utiliza para controlar todo lo que ocurre en varios lugares al mismo tiempo, como en instituciones en las cuales existen recursos de gran importancia y se debe mantener controlado su acceso o calles que por ser una de las más transitadas es de vital importancia grabar todo lo que acontece en ellas.

Estos sistemas de video vigilancia se pueden utilizar en lugares como tiendas, terminales, aeropuertos, parques y vías públicas, lo cual facilitaría el trabajo de las personas encargadas de cuidar estos sitios, manteniendo el orden y control de todo lo que ocurre, sin necesidad de ocupar varias personas en la tarea de la vigilancia y asegurándonos que todo va a quedar grabado para casos excepcionales.

La utilización de este tipo de software es muy importante y beneficiosa, pero tienen como inconveniente su alto costo, debido a los recursos que se necesitan para su correcto funcionamiento (cámaras, computadoras y otros periféricos). Hoy el país no cuenta con el presupuesto económico necesario para adquirir estos elementos que son indispensables en el buen funcionamiento de los sistemas de video vigilancia.

Por otro lado en el país se hacen grandes esfuerzos para fortalecer la informática y las comunicaciones con el fin de crear soluciones propias. Por tal motivo la Universidad de las Ciencias Informáticas (UCI) es un vivo ejemplo de estos avances. En ella existe un proyecto que lleva por nombre video vigilancia, donde se ha desarrollado un software que lleva por nombre Suria Vision.

Suria Vision es un sistema integral de video vigilancia inteligente, integrado por varios módulos. Una de las principales ventajas que brinda para su uso es que no establece un proveedor de cámara específico en su funcionamiento, por el contrario el sistema soporta cámaras de diferentes modelos; a diferencia de la mayoría de los sistemas de video vigilancia a nivel mundial que no son compatibles con cualquier tipo de fabricante. Esto favorece en especial a Cuba pues debido al bloqueo económico al que se encuentra sometido, no siempre se pueden hacer negociaciones con otros países.

En aras de facilitar el acceso y el intercambio de información del sistema Suria Vision creado, este proyecto se dio a la tarea de crear una versión web que prestara los principales servicios de la versión de escritorio presentada. Inicialmente se desarrolló la estación de visualización la que hoy permite la

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

gestión de roles, usuarios, permisos y cámaras, la visualización en varios modos de trabajo así como la manipulación de las cámaras insertadas acorde a las funcionalidades que estas puedan brindar.

Actualmente esta solución no cuenta con un servicio de grabación el cual se hace indispensable para el almacenamiento y posterior análisis de las medias visualizadas, así como un sistema de reporte que facilite el acceso a la información deseada, lo que constituye el problema a resolver de la presente investigación científica.

1.4. Características del sistema Suria Vision.

El sistema de video vigilancia Suria Vision integra varias aplicaciones, con el objetivo de facilitar la realización de múltiples funciones enfocadas a mejorar la vigilancia, permitiendo almacenar y manipular los flujos de videos que se obtienen a través de las cámaras. Cuenta con cuatro componentes fundamentales:

Visor: Es la estación de monitoreo, módulo que permite visualizar los flujos de videos generados por la cámaras. Tiene la capacidad de reflejar todo el aspecto organizativo con que se manejan las cámaras internamente, es decir que la forma de presentación responda a la disposición física de las cámaras en el lugar del despliegue de la aplicación, además de poder visualizarlas de manera independiente o colectiva(a manera de vistas).

Gestor: Es el módulo fundamental de la aplicación, y todos los demás son agentes que se encargan de realizar tareas específicas. Todo el tráfico de información al pasar por el Gestor es fácil de controlar y supervisar, además de que se gana en flexibilidad en el sistema teniéndolo a él como eje central de la aplicación. Es el que único que interactúa directamente con la Base de Datos.

Grabador: Es el módulo encargado de almacenar los flujos de video obtenidos de las cámaras, bajo petición de un cliente determinado o por configuración, que puede ser por horarios determinados o por la ocurrencia de algún evento. Entre las actividades que realiza se encuentran:

- Gestionar calendario de grabaciones.
- Grabar manualmente.
- Intercambiar información.
- Mostrar cámaras instaladas.
- Registrar log.

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

- Visualizar el calendario de grabación.

Recuperador: Es el módulo encargado de recuperar los videos almacenados en los servidores, hacer búsquedas en estos, servir los resultados al usuario y darle la posibilidad de la reproducción de los mismos a diferentes niveles de velocidad.

1.5. Análisis sobre soluciones existentes y sus principales limitantes.

La seguridad se ha convertido en el principal aspecto de protección tanto física como social. Con los avances de las tecnologías ha surgido un incremento notable de datos y el procesamiento de los mismos. Las empresas e instituciones se han percatado de la necesidad de mantener sus datos, bienes y recursos humanos protegidos. El mercado de la video vigilancia ha prosperado, impulsado por el aumento de hechos delictivos y, en consecuencia, se ha elevado la conciencia de cuidar la seguridad pública y privada. Esto es gracias al vertiginoso cambio tecnológico, en el cual los sistemas analógicos de Circuitos Cerrado de Televisión (CCTV) son remplazados por el video en red, una tecnología que empezó en la empresa Axis, pero ya varias empresas explotan las ventajas de esta técnica y desarrollan soluciones que hacen más fácil la vigilancia de cualquier negocio, empresa y vivienda.

1.5.1. Sistemas de video vigilancia basados en la web.

Las soluciones basadas en la Web también son un acontecimiento en el desarrollo de la video vigilancia por las facilidades que estas pueden proporcionar. Entre las compañías envueltas en el desarrollo de soluciones de este tipo está Cisco, que además cuenta con un amplio catálogo de equipos físicos.

- **Cisco Video Surveillance Stream Manager Software** es uno de los sistemas que brinda esta compañía, basado en una página web sencilla. El sistema sigue el modelo cliente-servidor, donde el cliente es una PC con una cámara conectada al servidor, que es donde se desean controlar los eventos que ocurren durante la vigilancia. El servidor es un alojamiento web habitual y la "secuencia de video" se puede visualizar por medio de cualquier PC conectada a Internet. Este sistema está diseñado para manejar sólo los equipos físicos que brinda esta compañía (16).
- **ZoneMinder** aplicación que se instala solo en servidores Linux que soporten video para Linux. Permite capturar, analizar, grabar y monitorizar desde cualquier tipo de cámara. Se puede conectar las cámaras y visualizar la imagen a través de una aplicación web, programar los horarios de funcionamiento de las cámaras y grabar de forma continua un video o fotografías

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

en secuencia. Además de contar con una interfaz de usuario amigable para el control del sistema o de las cámaras (17).

- **Vidium** ofrece servicios de video en tiempo real, a través de un canal de comunicación seguro y sin que el usuario tenga que instalar ningún programa en su ordenador. Este sistema es capaz de generar alertas por mensajes de texto (SMS) o correo electrónico en función de eventos propios (cortes de corriente o señal) o sucesos de las cámaras (detección de movimiento). Está pensado para instalaciones de hasta 12 cámaras porque con esto satisfacen hasta el 80% de las necesidades del mercado, algo que resulta una limitante para un sistema de video vigilancia de mayor envergadura (18).

1.5.2. Sistemas de video vigilancia en Cuba.

En Cuba también se utilizan software para la video vigilancia entre los que se encuentran: **XYMA SAVE VISION** de Datys empresa perteneciente al MININT, es un software basado en tecnología IP, con un alto grado de modularidad, adaptable a una gran cantidad de entornos, flexible y escalable. Permite mantener la seguridad monitoreando y controlando en tiempo real y de forma histórica cada uno de los movimientos que ocurren en las áreas sensibles que se identifiquen, debido a los componentes con que se desarrolló tiene un alto costo (19).

El software de **Panasonic WV-NS32x**, distribuido por COPEXTEL, solo soporta cámaras Panasonic, permite escoger el monitor con el cual se va a trabajar dependiendo del número de la cámara que se vaya a utilizar, incluye control de zoom, permite seleccionar las funciones que se desean ejecutar en el monitor y la programación de las cámaras para que graben en un tiempo o espacio determinado. Permite cambiar a blanco y negro cuando la grabación se realiza en lugares de poca luz y cuenta con un control auxiliar que permite un manejo más automatizado del software (20).

El **GV-600** comercializado por Seisa, solo es compatible con la tecnología analógica, su instalación es muy fácil y sus opciones de conectividad son ilimitadas, permite la grabación de forma manual, por eventos y registra el inicio y fin de la misma. El video puede grabarse por programas definidos o por detección de movimiento, los videos se pueden almacenar tanto en disco duro como en alguna unidad de almacenamiento extraíble (21) (22).

1.5.3. Conclusiones sobre las Soluciones Existentes.

Luego de realizar un profundo estudio del arte de los sistemas de video vigilancia existentes tanto en el mundo como en Cuba, se pudo apreciar que estos tienen ciertas limitantes para su uso y desempeño. Una aplicación que como Cisco Video Surveillance Stream Manager Software, limita su uso a un

CAPÍTULO I. FUNDAMENTACIÓN TEÓRICA. VALORACIÓN SOBRE APLICACIONES

EXISTENTES.

determinado fabricante, no podrá funcionar correctamente en un entorno donde existan recursos de diferentes marcas, pues inutilizará una parte del sistema de seguridad. Por otro lado, el hecho de que una aplicación web solo pueda ser ejecutada en un determinado sistema operativo, como sucede con ZoneMinder, obliga al usuario a modificar su entorno de trabajo a expensas del uso de la aplicación. También, el hecho de que una aplicación limite la cantidad de recursos a emplear, tal como lo hace Vidium, acota las posibilidades de los sistemas de seguridad y no permite expansiones de los mismos. Un alto costo de sus componentes como presenta XYMA SAVE VISION, es imposible que sea adquirida por cualquier individuo o empresa o si es el caso en que solo soporten cámaras Panasonic como lo hace WV-NS32x, al igual que GV-600 que solo permite cámaras analógicas. Por tales motivos se hace necesaria la implementación de una aplicación de video vigilancia web que resguarde las ventajas de las soluciones estudiadas y a su vez supere las limitantes identificadas en cada una de ellas.

1.6. Conclusiones.

En este capítulo se le da cumplimiento a las tareas de la investigación que permitieron realizar un estudio del avance que ha tenido la video vigilancia y la tecnología IP, logrando establecer de esta forma los principales términos asociados a la investigación. Se logró obtener como resultado un resumen detallado acerca de algunas aplicaciones de video vigilancia sobre plataformas web existentes en Cuba y el mundo, analizándose sus funcionalidades y haciendo énfasis en sus limitantes, lo que permitió llegar a la conclusión de que estas aplicaciones no son del todo efectivas para el país, teniéndose en cuenta las limitantes identificadas como aspectos a superar por la solución que propone este trabajo.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

2.1. Introducción.

Para lograr desarrollar una aplicación que cumpla con las características necesarias para su correcto funcionamiento es necesario emplear tecnologías que permitan realizar un software con calidad. En este caso para el desarrollo de la aplicación se utilizarán las mismas tecnologías empleadas en la versión del Visor Web para lograr una mejor integración de dichos módulos.

A continuación se plantean algunas de las características de cada una de las herramientas y tecnologías empleadas:

2.2. Características de los lenguajes de programación a utilizar.

- **HTML (Lenguaje de Marcado de Hipertexto):** es el lenguaje de programación utilizado para crear páginas web, con él se definen la posición, forma y funcionamiento de las imágenes, textos e hipervínculos incluidos en la página. Este lenguaje indica a los navegadores cómo deben mostrar el contenido de una página web (23).
- **JavaScript:** es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. Gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, utilización de teclas y cargas de páginas. JavaScript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera y Mozilla Firefox (24).
- **C#:** es un lenguaje de alto nivel diseñado para la construcción de una amplia gama de aplicaciones empresariales que se ejecutan en .NET Framework. Una evolución de Microsoft C y C ++, C # un lenguaje orientado a objetos, simple, moderno y seguro. El código C # es potente y fácil de aprender. C#, incorpora las ventajas o mejoras de lenguajes ya existentes tales como C, Java, Visual Basic y C++, lográndose un lenguaje flexible y poderoso. También contiene una biblioteca de clases muy completa y bien diseñada, lo que hace que sea uno de los lenguajes predilectos y más utilizados (25).

2.3. Características de la tecnología a utilizar.

- **ASP.NET (Active Server Page):** es un entorno de programación que forma parte de la plataforma .NET, ideal para crear aplicaciones y servicios web. Su arquitectura ha sido totalmente

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

reconstruida para facilitar al máximo la creación de aplicaciones web dinámicas. Permite a los desarrolladores escribir código más limpio y más fácil de reutilizar y compartir, incrementando el rendimiento y la escalabilidad al poder acceder a lenguajes compilados, no interpretados. Para su desarrollo puede utilizar los lenguajes de programación C#, VB.NET (Visual Basic.NET) o J#. Es completamente orientado a objetos, y entre las principales ventajas de su uso se tienen:

1. Simplifica y facilita el desarrollo, debido al modelo de programación que emplea del lado del servidor.
2. El código fuente se ejecuta en el servidor, lo que hace que las páginas web tengan una alta flexibilidad y potencia.
3. El código HTML producido por la página es enviado al navegador, lo que permite que el código fuente de la aplicación web no pueda ser robado fácilmente.
4. Posee un gran soporte para XML⁴, CSS⁵ u otros estándares web establecidos.
5. Valida la información, a través de controles de validación, ahorrando trabajo a los desarrolladores en la creación de mecanismos de validación de datos.
6. Ofrece un entorno de trabajo organizado, con una división entre la capa de diseño y el código.
7. Las aplicaciones desarrolladas con ASP.NET pueden trabajar con grandes volúmenes de usuarios manteniendo una gran velocidad y alto rendimiento (26).

2.4. Características de la herramienta CASE a utilizar.

- **Visual Paradigm:** es una herramienta CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) que utiliza UML (Lenguaje Unificado de Modelado) como lenguaje de modelado. Está diseñada para una amplia gama de usuarios interesados en construir sistemas de software fiables, incluyendo actividades como ingeniería de software, análisis de sistemas y análisis de negocios.

Visual Paradigm es una herramienta que emplea las últimas notaciones de UML, ingeniería inversa, generación del código y exportación/importación XML. Además soporta aplicaciones web, genera código para el lenguaje Java y exporta en formato HTML, está disponible en varios idiomas, es fácil de instalar y fácil de actualizar. Admite compatibilidad con las demás versiones (27).

⁴ **XML (eXtensible Markup Language):** Metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).

⁵ **CSS (Cascading Style Sheets):** Es el lenguaje de hojas de estilo, con él se puede separar el aspecto visual del contenido las páginas web..

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

2.5. Características de la herramienta para el control de versiones a utilizar.

Control de versiones: Para ofrecer seguridad al código fuente y los archivos con que se trabaja en el ciclo de desarrollo del software se hace necesario emplear herramientas para el control de versiones.

- **SubVersion (SVN):** Es un software para el control de versiones, en este no se actualizan los archivos por separados, todo el repositorio tiene un único número de revisión que indica el estado de todos los archivos. Los cambios que se envían por la red solo son actualizados si se logra enviar el paquete completo con éxito, para eliminar el problema de que pueda afectarse la información ya guardada. SubVersion funciona idénticamente con ficheros de texto y ficheros binarios. Ambos tipos de ficheros son almacenados igualmente y comprimidos en el repositorio. Es fácil de mantener y adaptable a cualquier lenguaje (28).

Características de SubVersion:

- Se sigue la historia de los archivos y directorios a través de copias y renombrados.
- Las modificaciones (incluyendo cambios a varios archivos) son atómicas.
- La creación de ramas y etiquetas es una operación más eficiente.
- Se envían sólo las diferencias en ambas direcciones.
- Maneja eficientemente archivos binarios.
- Permite selectivamente el bloqueo de archivos. Se usa en archivos binarios que, al no poder fusionarse fácilmente, conviene que no sean editados por más de una persona a la vez (28).

2.6. Características del IDE desarrollo a utilizar.

IDE de desarrollo: Los IDEs⁶, son aplicaciones claves para una implementación rápida y eficaz del software, los mismos se deben seleccionar en dependencia de los lenguajes en que se piensa desarrollar, puesto que no todos soportan los lenguajes requeridos (29).

- **Microsoft Visual Studio 2010:** es un entorno de desarrollo integrado que simplifica la creación, depuración e implementación de aplicaciones, lanzado por Microsoft, enfocado hacia las necesidades y perspectivas de los desarrolladores de software de la actualidad. Este IDE, soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET. Contiene nuevas formas de trabajo para Windows Presentation Foundation (WPF) y aplicaciones Microsoft Silverlight a modo de lograr aplicaciones enriquecidas de Internet (RIA) de mayor calidad y rapidez. Además, presenta herramientas integradas para el desarrollo de Windows 7, posee mecanismos para una mejor detección de errores y permite un seguimiento rápido del flujo de ejecución de un programa sin necesidad de llamar al depurador (30).

⁶IDE (Integrated Development Environment): Entorno de Desarrollo Integrado.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

Características de Visual Studio 2010

- Interpreta rápidamente el código: El nuevo editor de código facilita el zoom en el código, resalta las referencias de método y trae potentes funciones de superposición.
- Abre nuevas capacidades: Utiliza el soporte integrado de programación paralela para agregar nuevas funcionalidades en la aplicación cuando se ejecuta en una máquina de procesadores de núcleo múltiple.
- Utiliza las habilidades existentes: Trabaja el desarrollo de SharePoint, incluyendo herramientas para componentes web, listas, los flujos de trabajo, eventos y más, por lo que puede traer grandes nuevas herramientas de colaboración personalizadas.
- Dedicar menos tiempo a la depuración: La jerarquía de llamada en línea ayuda rápidamente rastrear el flujo de ejecución de un programa sin invocar al depurador. También puede utilizar la marca de etiquetas de punto de interrupción para realizar una depuración más sencilla (31).

2.7. Protocolos a utilizar para la comunicación.

Comunicación: Para la comunicación, se necesitan tecnologías que permitan la transmisión de datos sobre la red condicionadas por el lenguaje de desarrollo a emplear, así como también protocolos de comunicación para poder enviar y recibir datos. Entre los protocolos de comunicación se encuentran:

- **HTTPS (HyperText Transfer Protocol Security):** se emplea para la transmisión de datos sensibles y para establecer comunicaciones seguras. Sigue los mismos principios de funcionamiento del protocolo HTTP, salvo que cuenta con certificados y métodos de cifrado para establecer la seguridad a través del cifrado de los datos (23).
- **RTSP (Real Time Streaming Protocol):** permite controlar el flujo de datos multimedia sobre la red IP, admite una comunicación bidireccional de alta calidad, aporta un cierto nivel de seguridad y es fácilmente adaptable a distintas plataformas como Windows, Unix o Macintosh (32).
- **TCP/IP (Transmission Control Protocol / Internet Protocol):** es confiable, los datos siempre llegarán de forma ordenada, ya que implementa mecanismos de retransmisión en caso de que el receptor no confirme la recepción de los datos (32).
- **Microsoft Net Remoting:** es una interfaz de programación de aplicaciones (API⁷) para la comunicación. Permite a los programadores construir de una forma rápida aplicaciones distribuidas,

⁷API (Application Programming Interface): Interfaz para programación de aplicaciones.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

debido a las facilidades que ofrece para el desarrollo, estableciendo enlaces entre los componentes de la aplicación sin importar la ubicación de los mismos (33).

2.8. Bibliotecas a utilizar para el desarrollo de la interfaz de usuario.

Interfaz de Usuario: Para que el usuario interactúe a gusto con un sistema necesita, como aspecto primordial, de una interfaz de usuario atractiva que lo motive y le sea entendible e intuitiva en el trabajo que realiza. Para el desarrollo de la misma en tecnología web, se emplean bibliotecas o frameworks que facilitan este trabajo, aunque existen también métodos tradicionales de diseño que emplean imágenes estáticas o animadas y textos decorados. Se empleará la biblioteca de JavaScript ExtJS asociada a la biblioteca Ext.NET para poder emplear ExtJS integrado a la tecnología ASP.NET.

- **ExtJS:** es una biblioteca⁸ JavaScript que permite construir aplicaciones complejas en Internet o RIAs⁹. Esta biblioteca incluye:
 - ✓ Componentes de Interfaz de Usuario de alto nivel y personalizables.
 - ✓ Modelo de componentes extensibles.
 - ✓ Un API fácil de usar.
 - ✓ Licencias Open Source y comerciales.

ExtJS permite crear aplicaciones complejas utilizando componentes predefinidos así como un manejador de layouts (diseños), esto provee una experiencia consistente sobre cualquier navegador, evitando el problema de validar que el código escrito funcione bien en cada uno (Firefox, Internet Explorer, Safari, etc.). También provee un balance entre Cliente – Servidor, donde la carga de procesamiento se distribuye, permitiendo que el servidor pueda manejar más clientes al mismo tiempo. Además emplea comunicación asíncrona y el motor de render¹⁰ puede comunicarse con el servidor sin necesidad de estar sujeta a la aplicación, a un evento o una acción del usuario, dándole la libertad de cargar información sin que el cliente se dé cuenta; por tanto permite la eficiencia de la red, pues el tráfico de red puede disminuir al permitir que la aplicación elija que información desea transmitir al servidor y viceversa (29).

⁸ **Biblioteca:** Colección o conjunto de subprogramas usados para desarrollar software.

⁹ **RIA (Rich Internet Applications):** Aplicaciones de Internet Enriquecidas.

¹⁰ **Motor de render:** Es el encargado de generar o dibujar los componentes de la librería y generar las imágenes a partir de un modelo dado.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

- **Ext.NET:** es una biblioteca de código abierto diseñada para ASP.NET que integra la biblioteca de JavaScript ExtJS para la creación de aplicaciones enriquecidas de Internet. Ext.NET ofrece más de 100 controles de alto rendimiento que pueden ser empleados en ASP.NET para el desarrollo de interfaces de usuario de alta calidad y funcionalidad empleando las bondades de ExtJS, AJAX¹¹ y JSON¹². Además, permite la fusión de estas tecnologías con la plataforma .NET, así como el trabajo con las mismas tanto desde el lado del cliente como del servidor (29).

2.9. Características de SQLite como sistema gestor de Base de Datos (SGBD).

SQLite es un sistema gestor de base de datos distribuido bajo dominio público, implementa un gestor de bases de datos SQL embebido, sin configuración, transaccional y autocontenido de código abierto, totalmente libre y tiene como función hacer un sistema de bases de datos relacional. Se ejecuta en muchas plataformas y sus bases de datos pueden ser fácilmente portadas sin ninguna configuración o administración.

Una de las primeras diferencia entre los motores de bases de datos convencionales es su arquitectura cliente/servidor, pues SQLite es independiente, simplemente se realizan llamadas a subrutinas o funciones de las propias bibliotecas de SQLite, lo cual reduce ampliamente la latencia en cuanto al acceso a las bases de datos. Con lo cual se puede decir que las base de datos compuesta por la definición de las tablas, índices y los propios datos son guardados por un solo fichero estándar y en un solo ordenador (34) (35).

2.10. Características del lenguaje de modelado para la implementación del sistema.

El Lenguaje Unificado de Modelado (UML) prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación.

UML proporciona un vocabulario y una regla para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema.

Los objetivos de UML son muchos pero se pueden sintetizar en sus funciones:

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

¹¹ **AJAX(Asynchronous JavaScript and XML):** JavaScript Asíncrono y XML.

¹² **JSON (JavaScript Object Notation):** Formato ligero de intercambio de datos.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

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

Un modelo UML está compuesto por tres clases de bloques de construcción:

- Elementos: Son abstracciones reales o ficticias (objetos, acciones).
- Relaciones: Relacionan los elementos entre sí.
- Diagramas: Son colecciones de elementos con sus relaciones.

UML resuelve de forma satisfactoria el viejo problema del desarrollo de software como es su modelado gráfico. Además ha llegado a una solución unificada basada en lo mejor que había hasta el momento, lo cual lo hace más excepcional (36) (37).

2.11. Metodología de desarrollo de software a utilizar en la implementación del sistema.

Con el avance significativo que fue teniendo la informática con el pasar del tiempo, profesionales en este campo notaron la necesidad de guiarse por ciertos ejemplos o modelos seleccionados para lograr desarrollar un software con calidad, es decir seguir una metodología.

En este caso se utilizará RUP (Rational Unified Process, por sus siglas en inglés) por ser la más moderna de las metodologías tradicionales, y con mucho auge en el desarrollo de software, siendo la metodología más utilizada en el mundo entero en este ámbito, y es soportada por la herramienta Visual Paradigm que es la herramienta a utilizar para el desarrollo.

Algunas de las ventajas de RUP sobre las otras metodologías tradicionales son:

1. Realiza la evaluación en cada fase, permitiendo cambios de objetivos: entre más temprano se detecten los cambios menos costoso serán para el proyecto.
2. Sigue los pasos intuitivos necesarios a la hora de desarrollar el software y guía detalladamente el desarrollo del software.
3. Permite un seguimiento detallado en cada una de las fases: hace posible que se detecten los errores lo más temprano posible.

RUP:

RUP es un proceso de desarrollo de software que utiliza UML (Unified Modeling Language, por sus siglas en inglés) como lenguaje de modelado de procesos y constituye la metodología más utilizada

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

para el análisis, implementación y documentación de sistemas orientados a objetos. Es dirigido por casos de uso, estos reflejan lo que los usuarios futuros desean y necesitan, se captan cuando se está iniciando el proceso y en esta fase se representan como requerimientos, a partir de ahí guían todo el proceso de desarrollo. También, este proceso es totalmente centrado en la arquitectura porque muestra una visión común del sistema completo, con la que el equipo de desarrollo y los usuarios deben estar de acuerdo. Además, RUP es un proceso iterativo e incremental pues propone que cada fase se desarrolle en iteraciones, y cada iteración tiene que proponerse un incremento en el proceso de desarrollo del software. Los principales elementos de esta metodología son los trabajadores, sus actividades, los artefactos y el flujo de actividades que es el que muestra el resultado observable.

Principales características de RUP:

1. Forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).
2. Pretende implementar las mejores prácticas en Ingeniería de Software.
3. Desarrollo iterativo.
4. Administración de requisitos.
5. Control de cambios.
6. Modelado visual del software.
7. Verificación de la calidad del software.

Fases de RUP:

1. Conceptualización (Concepción o Inicio): se describe el negocio y se identifican los casos de uso del sistema.
2. Elaboración: se define la arquitectura del sistema y se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen.
3. Construcción: el producto se desarrolla a través de iteraciones donde cada iteración involucra tareas de análisis, diseño e implementación. Se documenta tanto el sistema construido como el manejo del mismo. Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario.
4. Transición: se libera el producto y se entrega al usuario para un uso real. Se incluyen tareas de marketing, empaquetado atractivo, instalación, configuración, entrenamiento, soporte, mantenimiento, etc. Se realizan pruebas y reparación de errores.

Al finalizar cada fase se le presentan al cliente los artefactos definidos, es decir, el avance del proyecto para una evaluación de la calidad del mismo desde el punto de vista del cumplimiento o no con las necesidades planteadas por él.

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

Flujos de trabajo de RUP:

Las disciplinas conllevan los flujos de trabajo, los cuales son una secuencia de pasos para la culminación de cada disciplina, estas disciplinas se dividen en dos grupos: las primarias y las de apoyo. Las primarias son las necesarias para la realización de un proyecto de software, aunque para proyectos no muy grandes se pueden omitir algunas, entre ellas se tienen: Modelado del Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas y Despliegue. Las de apoyo son las que como su nombre lo indica sirven de apoyo a las primarias y especifican otras características en la realización de un proyecto de software, entre estas se tienen: Entorno, Gestión del Proyecto, Gestión de Configuración y Cambios. A continuación se describe cada una de estas disciplinas:

- Modelado del negocio: esta disciplina tiene como objetivos comprender la estructura y la dinámica de la organización, identificar posibles mejoras, problemas actuales y los procesos de negocio.
- Requerimientos: esta disciplina tiene como objetivos establecer lo que el sistema debe hacer (Especificar Requisitos), definir los límites del sistema, una interfaz de usuario y realizar una estimación del costo en tiempo de desarrollo.
- Análisis y diseño: esta disciplina define la arquitectura del sistema y tiene como objetivos trasladar requisitos en especificaciones de implementación, al decir análisis se refiere a transformar Casos de Uso (CU) en clases, y al decir diseño se refiere a refinar el análisis para poder implementar los diagramas .
- Implementación: esta disciplina tiene como objetivos implementar las clases de diseño como componentes, asignar los componentes a los nodos, probar los componentes individualmente e integrar los componentes en un sistema ejecutable (enfoque incremental).
- Pruebas: esta disciplina tiene como objetivos verificar la integración de los componentes (prueba de integración), verificar que todos los requisitos han sido implementados (pruebas del sistema) y asegurar que los defectos detectados han sido resueltos antes de la distribución.
- Despliegue: esta disciplina tiene como objetivos asegurar que el producto está preparado para el cliente, proceder a su entrega y recepción por el mismo. En esta disciplina se realizan las actividades de probar el software en su entorno final (Prueba Beta), empaquetarlo, distribuirlo e instalarlo, así como la tarea de enseñar al usuario.
- Gestión y configuración de cambios: es esencial para controlar el número de artefactos producidos por la cantidad de personas que trabajan en un proyecto conjuntamente. Los controles sobre los

CAPÍTULO II. HERRAMIENTAS Y TECNOLOGÍAS PARA EL DESARROLLO DEL SOFTWARE.

cambios son de mucha ayuda ya que evitan confusiones costosas como la compostura de algo que ya se había arreglado.

- Gestión del proyecto: su objetivo es equilibrar los objetivos competitivos, administrar el riesgo y superar restricciones para entregar un producto que satisface las necesidades de ambos clientes con éxito y los usuarios.
- Entorno: esta disciplina se enfoca en las actividades necesarias para configurar el proceso que engloba el desarrollo de un proyecto y describe las requeridas para el progreso de las pautas que apoyan un proyecto. Su propósito es proveer a la organización que desarrollará el software un ambiente en el cual basarse para desarrollar el sistema (38).

2.12. Conclusiones.

Durante la elaboración de este capítulo se logró dar cumplimiento a las tareas de investigación que se definieron para identificar las herramientas y tecnologías que se emplearán en la creación de la aplicación, obteniendo como resultado que se utilizarán las mismas herramientas y tecnologías que se emplearon para la construcción de la primera versión del Visor Web, como lenguajes de programación HTML, JavaScript y C#, como tecnología para el desarrollo ASP.NET, para el modelado de la solución se utilizará UML, como herramienta case Visual Paradigm. Para el control de versiones SubVersion, como IDE de desarrollo Microsoft Visual Studio 2010, como bibliotecas ExtJS y Ext.NET, para la gestión de los datos se utilizará el sistema gestor de base de datos SQLite y como metodología de desarrollo RUP, logrando facilitar con la tecnologías y herramientas antes mencionadas una mejor integración con la estación de visualización ya existente y de esta forma realizar una aplicación que cumpla con la calidad requerida.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

CAPÍTULO III: PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.1. Introducción

En este capítulo se realiza un estudio del modelado del negocio con el objetivo de facilitar la comprensión de los principales conceptos que se utilizan para la descripción de la solución propuesta. Por no existir un negocio real se decide realizar un modelo de dominio donde se describen las entidades que intervienen en los procesos de negocios identificados.

3.2. Modelo de Dominio.

Por no existir un negocio real, y al no poder precisar la estructura de los procesos de negocio, se emplea un Modelo de Dominio o Modelo Conceptual. El Modelo de Dominio permite de manera visual mostrar al usuario los principales conceptos que se manejan en el dominio del problema, logrando facilitar la comprensión del sistema. Además es una representación de las clases conceptuales del mundo real, no de componentes software y emplea un glosario de términos para lograr una mejor concepción de los conceptos asociados (39).

3.3. Descripción de la propuesta del sistema.

Una vez que el usuario interactúe con la interfaz de grabación podrá realizar varias funciones, si desea realizar una grabación ya sea por calendario o manualmente, se solicita el servicio al gestor y este activa el servicio de grabación que tenga mejor disponibilidad en ese momento para responder a la petición realizada. Si el usuario desea conocer las cámaras que se encuentran conectadas, las grabaciones realizadas o los usuarios existentes, deberá acceder a los reportes del sistema, que no es más que un conjunto de datos que le permitirá obtener la información solicitada, dicho sistema de reporte pedirá los datos necesarios al gestor y este accederá a la base de datos para obtener la información solicitada, luego enviará los datos para mostrarlos al usuario.

3.4. Diagrama del Modelo de Dominio.

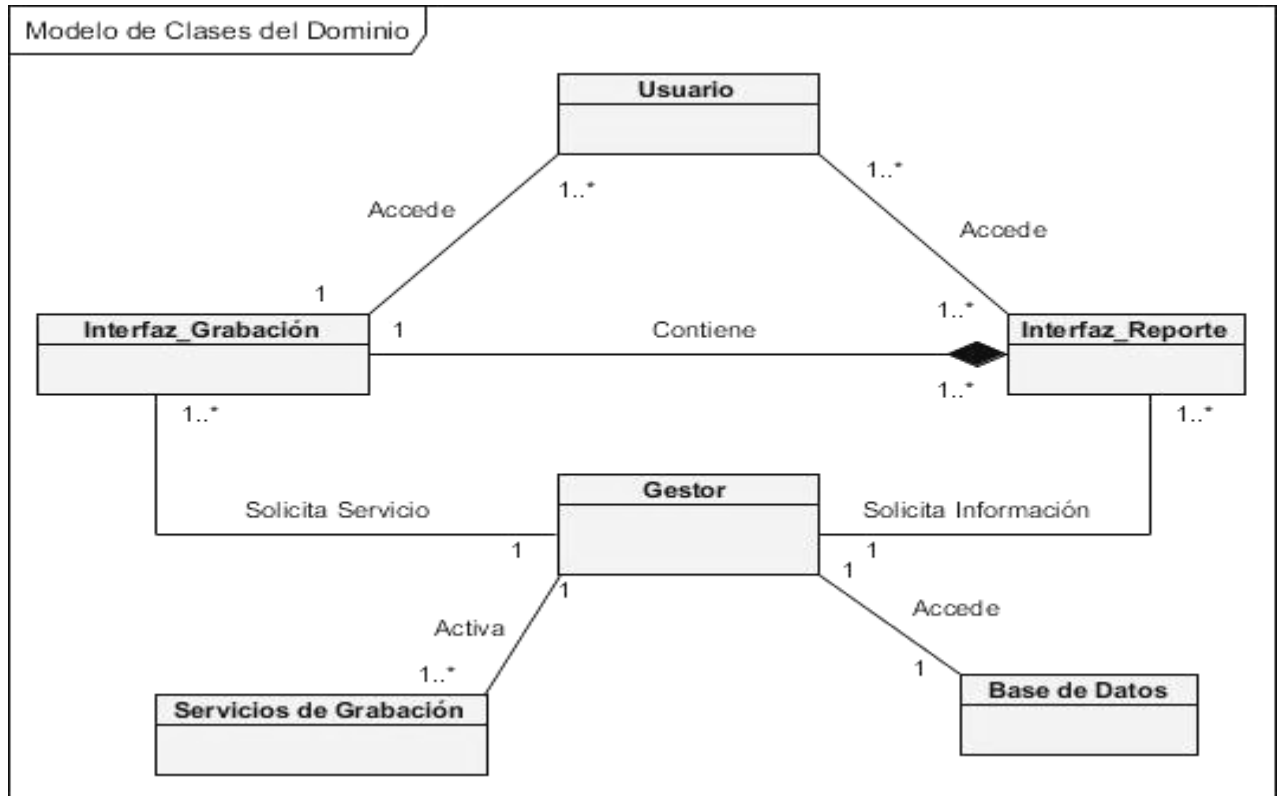


Figura 1. Diagrama Modelo de Dominio.

3.5. Descripción de las clases.

Usuario: Persona encargada de interactuar con la aplicación.

Interfaz_Grabación: Interfaz web que permite visualizar las reglas de grabación planificadas, las tareas en espera y en ejecución y el estado de los procesos de las grabaciones actuales.

Servicios de Grabación: Realiza las acciones solicitadas por la interfaz de grabación, comunicándose con la misma a través de eventos.

Interfaz_Reporte: Interfaz web que permite visualizar los reportes de seguridad, de las cámaras y de las grabaciones realizadas.

Gestor: Es el módulo encargado de proporcionar la información con la cual van a trabajar las clases reporte y servicio de grabación.

Base de Datos: Es una colección de información organizada, de forma tal que facilite el acceso a los datos necesarios.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.6. Requisitos Funcionales del Sistema.

El flujo de trabajo de Requisito tiene como propósito establecer lo que debe hacer el sistema, que no es más que definir los requisitos funcionales. Especificándose de esta manera los límites de la aplicación (40) .

*Nota: Se usa el prefijo **RF** en su nomenclatura-*

RF 1 Autenticar Usuario.

Descripción: El sistema debe permitir a los usuarios que deseen acceder a la aplicación autenticarse, garantizando la seguridad de acceso al sistema.

RF 2 Mostrar listado de Reglas y Tareas, horizontal o vertical.

Descripción: El sistema debe permitir mostrar al usuario el listado de reglas y tareas de forma horizontal o vertical, según este lo desee.

RF 3 Grabar flujo de video desde cámara IP.

RF 3.1 Grabar de forma manual.

RF 3.2 Grabar mediante calendario.

RF 3.3 Responder ante eventos de inicio de grabación.

Descripción: El sistema debe permitir que el usuario inicie un proceso de grabación para una cámara determinada. Este puede realizarse desde un calendario especificando hora de inicio y fin de la grabación o manualmente.

RF 4 Mostrar cámaras existentes en el sistema.

Descripción: El sistema debe permitir mostrar las cámaras que se encuentran instaladas y agruparlas por zonas.

RF 5 Administrar Reglas de Grabación.

RF 5.1 Adicionar Regla de grabación.

RF 5.2 Eliminar Regla de grabación.

RF 5.3 Modificar Regla de grabación.

Descripción: El sistema debe permitir que el usuario agregue, elimine o modifique las reglas de grabación, según lo desee.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

RF 6 Administrar Tareas de Grabación.

RF 6.1 Adicionar Tarea de grabación.

RF 6.2 Eliminar Tarea de grabación.

Descripción: El sistema debe permitir que el usuario agregue o elimine tareas de grabación, según lo desee.

RF 7 Mostrar reporte de las cámaras.

Descripción: El sistema debe permitir mostrar un reporte referente a las cámaras instaladas.

RF 8 Mostrar reporte de las grabaciones.

Descripción: El sistema debe permitir mostrar las grabaciones realizadas en una fecha o entre fechas determinadas.

RF 9 Mostrar reporte de seguridad.

Descripción: El sistema debe permitir mostrar reportes de seguridad en los cuales se evidencie las actividades realizadas por los usuarios.

RF 10 Exportar reportes.

Descripción: El sistema debe permitir exportar los reportes en formato PDF.

3.7. Requisitos No Funcionales del Sistema.

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Son características que hacen atractivo, usable y confiable al sistema. Estos requisitos se agrupan en varias categorías (40).

*Nota: Se usa el prefijo **RNF** en su nomenclatura-*

RNF 1. Usabilidad.

El sistema podrá ser utilizado por cualquier persona con conocimientos medios en informática, por lo que toda la información que se muestre será fácil de comprender. Se prevé que la usabilidad de este producto cuente con un alto nivel de aceptación por los usuarios.

RNF 2. Seguridad.

Se garantizará la integridad, disponibilidad y confidencialidad de la información manipulada por el sistema para ello:

- El sistema permite a los usuarios autenticarse para acceder a la aplicación.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

- El sistema valida todos los datos insertados por el usuario antes de realizar cualquiera de las funcionalidades que brinda.
- El sistema solicita al usuario confirmación, ante acciones irreversibles, permitiendo que este verifique que es realmente la acción que desea realizar.

RNF 3. Apariencia.

La estructura de las interfaces será clara y bien distribuidas para que los usuarios sepan en cada momento qué acción realizar. Los colores serán convenientemente utilizados dada la funcionalidad y objetivos del sistema, siendo claros y sobrios en la mayor parte de la aplicación, logrando una vista agradable a los usuarios y letra visible que contrasten.

RNF 4. Soporte.

El sistema debe ser sencillo y asequible para los usuarios. Se requiere de la instalación del Plugin de VLC (Video LAN), para Mozilla, en su versión 0.8.6 o superior, instalación de Windows XP ó superior en las estaciones de trabajo. Además de la instalación de NET Framework 3.5 ó superior en las estaciones de trabajo.

RNF 5. Hardware.

- 1 GB de Memoria RAM (mínimo), 2GB Memoria RAM (recomendado).
- Microprocesador Intel Pentium IV a 3.0 GHz o superior (recomendado).
- Tarjeta de Red Gigabit Ethernet NIC (recomendado).

Descripción: Las recomendaciones anteriores permiten un funcionamiento eficiente del sistema a desarrollar, pues se necesita de una gran capacidad de memoria y un microprocesador potente para poder manipular todos los flujos de video y realizar las operaciones, sin que se presenten retrasos en las respuestas del sistema.

RNF 6. Software.

- Sistema Operativo Windows XP ó superior.
- NET Framework 3.5 ó superior.
- Navegador Mozilla Firefox 3.6 ó superior.
- Plugin de VLC (Video LAN), para Mozilla, versión 0.8.6 ó superior.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.8. Descripción del sistema.

Haciendo uso de las ventajas que brinda el lenguaje de modelado UML, se representan los requisitos funcionales enunciados anteriormente, por medio del diagrama de casos de uso del sistema. Donde cada uno de estos requisitos abarca una serie de acciones, las cuales se llevan a cabo por medio de los actores. Los artefactos, actores, prototipo de interfaz de usuario, casos de uso y sus descripciones conforman el modelo del sistema.

3.8.1. Definición de los actores del sistema.

Un actor del sistema no es más que un conjunto de roles que los usuarios desempeñan cuando interactúan con los casos de uso. Definiéndose como actores del sistema el Usuario y un Usuario con Privilegios.

Actor	Descripción
Usuario	Rol que representa a las personas que interactúan con el sistema antes de Autenticarse.
Usuario con Privilegios	Rol que representa los usuarios luego de autenticarse en el sistema, cuyos privilegios son asignados de acuerdo al esquema de seguridad establecido. De estos privilegios dependerán las acciones que podrá llevar a cabo dentro de la aplicación.

Tabla 1. Actores Definidos del Sistema.

3.8.2. Listado de los Casos de Uso.

CU1-Gestionar Calendario de Grabación.

CU2- Grabar Flujo de Video.

CU3- Visualizar Cámaras.

CU4- Visualizar Calendario de Grabación.

CU5- Visualizar Reportes.

CU6- Exportar Reportes.

CU7- Autenticar Usuario.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

3.8.3. Diagrama de Casos de Uso del sistema.

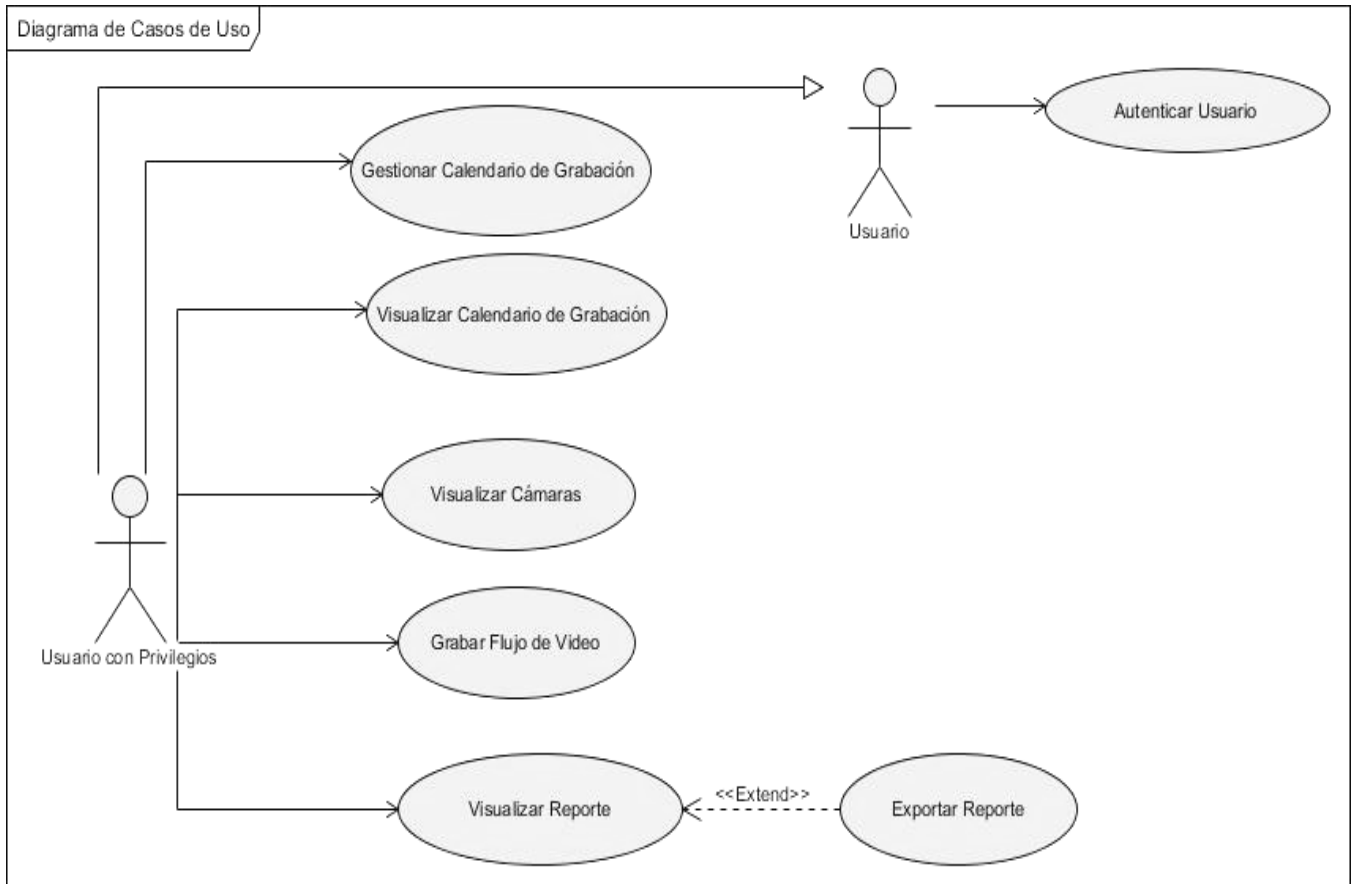


Figura 2. Diagrama de Casos de Uso del Sistema.

3.8.4 Descripción textual de los principales casos de uso del sistema.

En el presente epígrafe se presenta la descripción de los primeros cuatro casos de uso del sistema, por ser considerados arquitectónicamente significativos. La descripción de los casos de uso restantes puede encontrarse en el Anexo 1 del presente trabajo.

Caso de Uso:	Gestionar Calendario de Grabación.
Actores:	Usuario con Privilegios.
Resumen:	El caso de uso se inicia cuando el Usuario con Privilegios selecciona la opción de adicionar, eliminar o editar una regla de grabación.
Precondiciones:	Debe haber conexión con el Gestor. El usuario debe tener permisos para llevar a cabo esta acción. El usuario accedió a la funcionalidad de adicionar, eliminar o editar reglas de grabación.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Referencias	RF 5, RF 5.1, RF 5.2, RF 5.3, RF 6, RF 6.1, RF 6.2
Prioridad	Crítico
Flujo Normal de Eventos “Gestionar Calendario de Grabación”	
Sección “Adicionar Regla de Grabación”.	
Acción del Actor	Respuesta del Sistema
<p>1. El caso de uso inicia cuando el usuario selecciona una de las siguientes opciones:</p> <p style="padding-left: 20px;">a). Adicionar regla de grabación (Interfaz 1) (A).</p> <p style="padding-left: 20px;">b). Editar regla de grabación (Ver sección “Editar Regla de Grabación”)</p> <p style="padding-left: 20px;">c). Eliminar regla de grabación (Ver sección “Eliminar Regla de Grabación”)</p> <p>2. El usuario pulsa la opción “Adicionar regla de grabación”.</p>	<p>3. El Sistema le muestra un wizard¹³ (Interfaz 2) para introducir los siguientes datos:</p> <p style="padding-left: 20px;">B: Nombre de la Regla.</p> <p style="padding-left: 20px;">C: Formato de video.</p> <p style="padding-left: 20px;">D: Descripción de la regla de grabación.</p>
<p>4. El usuario inserta los datos necesarios y selecciona una de las siguientes opciones:</p> <p style="padding-left: 20px;">E: Siguiete.</p> <p style="padding-left: 20px;">F: Cancelar. (Ver sección “Cancelar”)</p> <p>5. El usuario selecciona la opción “Siguiete”.</p>	<p>6. El Sistema verifica que todos los datos estén llenos correctamente.</p> <p>7. Si todos los datos están insertados correctamente el Sistema muestra el segundo paso del wizard (Interfaz 3) para introducir los datos. De lo contrario ver flujo alterno 7a.</p> <p style="padding-left: 20px;">G: Frecuencia (Diaria, Semanal, Mensual o Exacta)</p> <p style="padding-left: 20px;">H: Hora de inicio.</p>
<p>8. El usuario inserta los datos y selecciona una de las siguientes opciones:</p> <p style="padding-left: 20px;">I: Siguiete.</p> <p style="padding-left: 20px;">J: Cancelar. (Ver sección “Cancelar”)</p>	<p>10. El Sistema verifica que todos los datos estén insertados correctamente. De lo contrario ver flujo alterno 10a.</p> <p>11. El Sistema muestra el tercer paso del wizard</p>

¹³ **Wizard:** Es una interfaz de usuario elemental, constituida por una secuencia de ventanas que guían al usuario a través de una serie de pasos bien definida.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

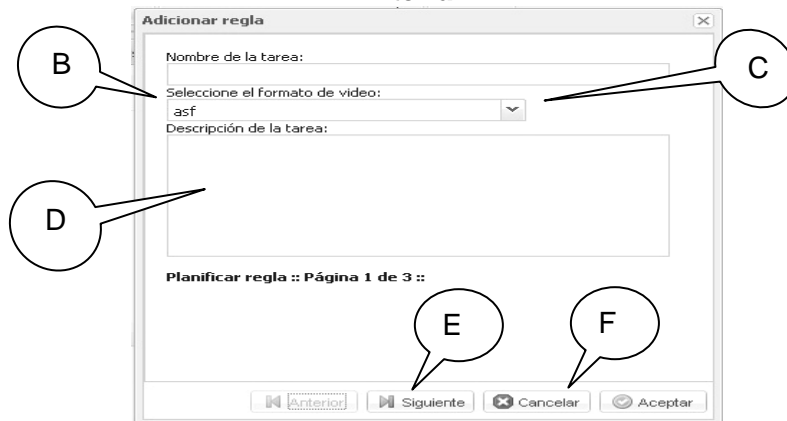
<p>K: Anterior.</p> <p>9. El usuario selecciona la opción “Siguiente”.</p>	<p>(Interfaz 4) para introducir los siguientes datos sobre la duración de la grabación:</p> <p>L: Intervalo de duración (30 minutos, especificados en horas, minutos, segundos)</p>
<p>12. El usuario inserta los datos correspondientes y selecciona una de las siguientes opciones:</p> <p>M: Aceptar.</p> <p>N: Anterior. (Ver sección “Anterior”)</p> <p>O: Cancelar. (Ver sección “Cancelar”)</p> <p>13. El usuario selecciona la opción “Aceptar”.</p>	<p>14. El sistema envía los datos de la regla al Gestor.</p> <p>15. El Gestor informa al cliente de grabación si la regla se puede agregar o no.</p> <p>16. Si la regla es aceptada por el Gestor el sistema genera una nueva regla y con ella las tareas correspondientes a la misma y termina el caso de uso. De lo contrario ver flujo alterno 16a.</p>

Prototipo interfaz “Adicionar Regla de Grabación”.

Interfaz 1



Interfaz 2



Interfaz 3

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Interfaz 4

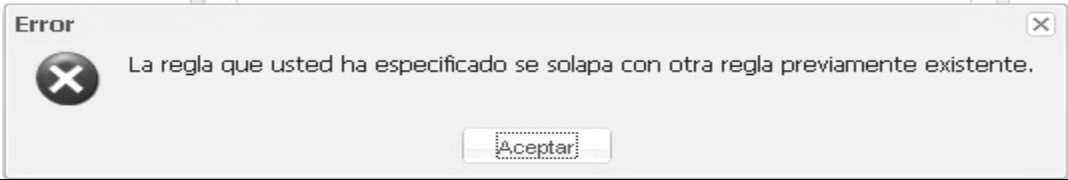
Flujos alternos “Adicionar Regla de Grabación”.

Acción del Actor	Respuesta del Sistema
	7a. Si existe algún campo vacío, el Sistema muestra un mensaje indicándole al usuario que llene todos los campos (“Llene los datos requeridos”). Dando la posibilidad al usuario de insertar los datos nuevamente (se retoma el flujo normal de los eventos a partir de la acción 4.)

Flujos alternos “Adicionar Regla de Grabación”.

Acción del Actor	Respuesta del Sistema
	10a. Si existe algún campo vacío, el Sistema muestra un mensaje indicándole al usuario que llene todos los campos (“Llene los datos requeridos”). Dando la posibilidad al usuario de

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

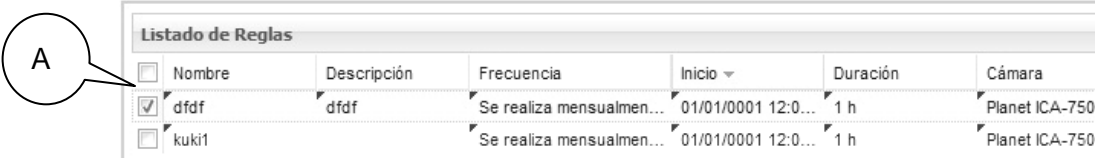
	insertar los datos nuevamente (se retoma el flujo normal de los eventos a partir de la acción 8.)
Flujos alternos “Adicionar Regla de Grabación”.	
Acción del Actor	Respuesta del Sistema
	16a. Si la regla definida se solapa con alguna existente, el Sistema muestra un mensaje indicando que no puede ser creada, porque se solapa con una regla existente y de esta forma termina el Caso de Uso (Interfaz 1).
Prototipo interfaz Flujo alternativo “Adicionar Regla de Grabación”.	
Interfaz 1	
	
Poscondiciones:	Se adiciona una nueva regla de grabación, que genera a su vez tareas de grabación.
Sección “Editar Regla de Grabación”.	
1. El usuario escoge una regla de grabación y selecciona la opción “Modificar Regla de Grabación” (Interfaz 1) (A), (Interfaz 2) (B). De lo contrario ver flujo alternativo 1a.	2. El Sistema muestra un wizard (Interfaz 3) para modificar los datos: C: Nombre de la regla. D: Formato de grabación. E: Descripción de la regla de grabación.
3. El usuario inserta los datos y selecciona una de las siguientes opciones: F: Siguiente. G: Cancelar. (Ver sección “Cancelar”)	4. El Sistema verifica que todos los datos estén llenos correctamente. 5. Si los datos son correctos el Sistema muestra el segundo paso del wizard (Interfaz 4) para modificar los datos. De lo contrario ver flujo alternativo 5a. H: Frecuencia (Diaria, Semanal, Mensual o Exacta) I: Hora de inicio.
7. El usuario inserta los datos y selecciona una de las siguientes opciones: J: Siguiente.	8. El Sistema verifica que todos los datos estén llenos correctamente. 9. Si todos los datos insertados son correctos el

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

<p>K: Cancelar. (Ver sección “Cancelar”) L: Anterior (Ver sección “Anterior”) 7. El usuario selecciona la opción “Siguiente”.</p>	<p>Sistema muestra el tercer paso del wizard (Interfaz 5) para modificar los datos. De lo contrario ver flujo alterno 9a. M: Intervalo de duración (30 minutos, especificados en horas, minutos, segundos)</p>
<p>10. El usuario inserta los datos y selecciona una de las siguientes opciones: N: Aceptar. O: Cancelar. (Ver sección “Cancelar”) P: Anterior (Ver sección “Anterior”) 11. El usuario selecciona la opción “Aceptar”.</p>	<p>12. El Sistema envía al gestor la información de la regla modificada. 13. El gestor envía la confirmación si se puede o no agregar la regla. 14. Si el Gestor confirma que se puede crear la regla, se crea la regla de grabación con las tareas correspondientes y finaliza el caso de uso. De lo contrario ver flujo alterno 14a.</p>


Prototipo de Interfaz “Editar Regla de Grabación”

Interfaz 1



Nombre	Descripción	Frecuencia	Inicio	Duración	Cámara
<input checked="" type="checkbox"/> ddfd	dfdf	Se realiza mensualmen...	01/01/0001 12:0...	1 h	Planet ICA-750
<input type="checkbox"/> kuki1		Se realiza mensualmen...	01/01/0001 12:0...	1 h	Planet ICA-750

Interfaz 2



Interfaz 3

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Adicionar regla

Nombre de la tarea: ddfd

Seleccione el formato de video: asf

Descripción de la tarea: ddfd

Planificar regla :: Página 1 de 3

Anterior | Siguiente | Cancelar | Aceptar

Interfaz 4

Adicionar regla

Cuando se inicia la grabación:

- Todos los días.
- Cada día de la semana.
- Todos los meses.
- En una fecha específica.

Fecha específica a grabar:

Fecha: []

Hora de inicio: 12 : 0 : 0 (h:m:s)

Cuándo se inicia la grabación :: Página 2 de 3 ::

Anterior | Siguiente | Cancelar | Aceptar

Interfaz 5

Adicionar regla

Fin del intervalo de grabación:

- Teinta minutos después del inicio de la grabación.
- Especificar el tiempo de duración.
- Terminar a una hora específica.

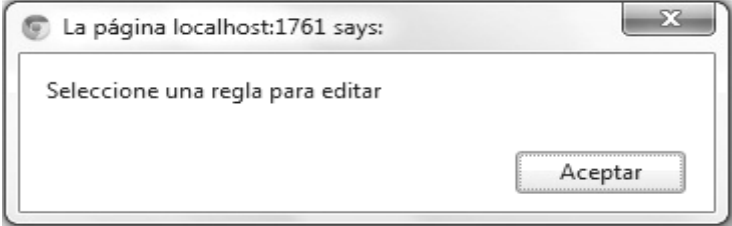
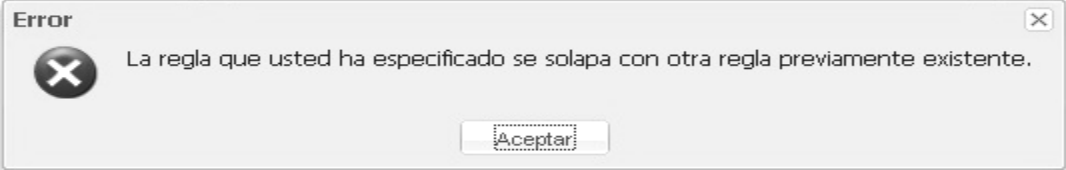
Duración de la grabación :: Página 3 de 3 ::

Anterior | Siguiente | Cancelar | Aceptar

Flujos Alternos “Editar Regla de Grabación”.

Acción del Actor	Respuesta del Sistema
1a.El usuario no seleccionó una regla.(Interfaz 1)	
	5a. Si existe algún campo vacío, el Sistema muestra un mensaje indicándole al usuario que

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

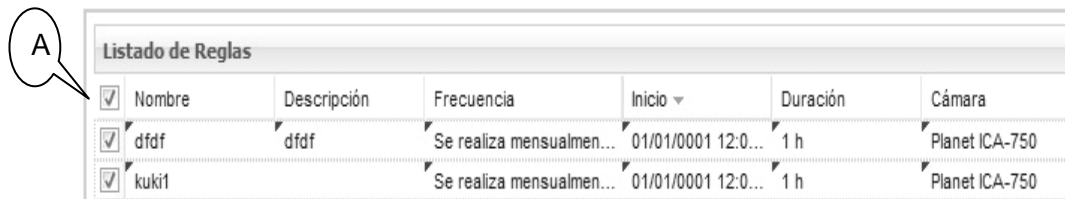
	llene todos los campos (“Llene los datos requeridos”). Dando la posibilidad al usuario de insertar los datos nuevamente (se retoma el flujo normal de los eventos a partir de la acción 3.)
Flujos Alternos “Editar Regla de Grabación”.	
Acción del Actor	Respuesta del Sistema
	9a. Si existe algún campo vacío, el Sistema muestra un mensaje indicándole al usuario que llene todos los campos (“Llene los datos requeridos”). Dando la posibilidad al usuario de insertar los datos nuevamente (se retoma el flujo normal de los eventos a partir de la acción 6.)
Flujos Alternos “Editar Regla de Grabación”.	
Acción del Actor	Respuesta del Sistema
	14a. Si la regla definida se solapa con alguna existente, el Sistema muestra un mensaje indicando que no puede ser creada, porque se solapa con una regla existente y de esta forma termina el Caso de Uso (Interfaz 2).
Prototipo de Interfaz flujos alternos “Editar Regla de Grabación”.	
Interfaz 1	
	
Interfaz 2	
	
Poscondiciones	Se edita la regla de grabación generándose una nueva tarea de grabación, permaneciendo las tareas generadas hasta ese momento.
:	

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Sección “Eliminar Regla de Grabación”.	
<p>1. El usuario selecciona una o varias reglas de grabación (Interfaz 1) (A).</p> <p>2. Selecciona la opción de eliminar reglas seleccionadas (Interfaz 2) (B).</p>	<p>3. El Sistema pide confirmación al usuario (Interfaz 3).</p>
<p>4. El usuario selecciona una de las siguientes opciones: C: Si D: No (Ver flujo alterno 4a.)</p> <p>5. El usuario selecciona la opción C. (Ver flujo alterno 5a.)</p>	<p>6. El Sistema solicita al Gestor que elimine las reglas de grabación de la base de datos.</p> <p>7. Se elimina la regla de grabación de la interfaz principal y termina el caso de uso.</p>

Prototipo de Interfaz “Eliminar Regla de Grabación.”

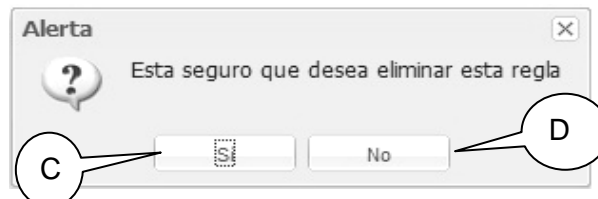
Interfaz 1



Interfaz 2



Interfaz 3



Flujos Alternos “Eliminar Regla de Grabación.”

Acción del Actor	Respuesta del Sistema
4a. El usuario selecciona la opción D.	5a. El Sistema cierra el mensaje de confirmación y termina el Caso de Uso.
Poscondiciones:	Se elimina la regla de grabación, y se mantienen las tareas de grabación

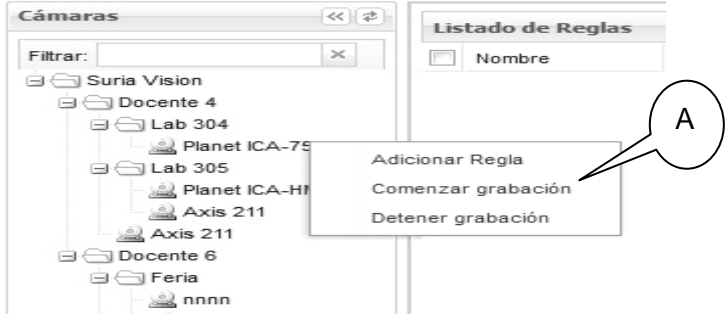
CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

generadas hasta ese momento.	
Sección “Cancelar”.	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la opción “Cancelar”.	2. El Sistema cancela operación, cierra la interfaz y termina el caso de uso.
Sección “Anterior”.	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona la opción “Anterior”.	2. El Sistema regresa a la interfaz anterior del wizard.

Tabla 2. Descripción del Caso de Uso Gestionar Calendario de Grabación.

Caso de Uso:	Grabar Flujo de Video.
Actores:	Usuario con Privilegios.
Resumen:	El caso de uso se inicia cuando el Usuario con Privilegios selecciona una cámara y la opción “Comenzar a Grabar” o cuando se ejecuta una tarea programada, el Sistema hace una petición de inicio de grabación al Gestor. El caso de uso termina cuando el Usuario con Privilegios selecciona la opción “Terminar Grabación” o la tarea programada llega a su fin.
Precondiciones:	El Gestor debe estar “Disponible”. El usuario tiene que tener permisos para realizar esta acción.
Referencias	RF 3, RF 3.1, RF 3.2, RF 3.3
Prioridad	Crítico
Flujo Normal de Eventos “Grabar Flujo de Video”.	
Sección “Grabar de Forma Manual”.	
Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando el usuario selecciona una cámara y la opción “Comenzar Grabación” (Interfaz 1)(A).	2. El Sistema envía al Gestor una petición de inicio de grabación. 3. El Gestor notifica al Sistema si puede comenzar a grabar o no. 4. Si el Gestor notifica que puede comenzar a grabar y el Sistema comienza la grabación, terminando así el caso de uso. De lo contrario ver flujo alterno 4a.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Prototipo de Interfaz “Grabar de Forma Manual”.	
Interfaz 1	
	
Flujos Alternos “Grabar de Forma Manual”	
Acción del Actor	Respuesta del Sistema
	4a. El Sistema notifica al usuario que no se puede realizar la grabación y termina el caso de uso.
Poscondiciones:	Comienza una grabación manual.
Sección “Terminar Grabación Manual”.	
1. El usuario selecciona una cámara y escoge la opción “Detener Grabación” (Interfaz 1)(A).	2. El Sistema muestra un menú con las grabaciones que se están ejecutando para dicha cámara (Interfaz 2) (B).
3. El usuario selecciona una de las grabaciones.	4. El Sistema le pide confirmación al usuario (Interfaz 3).
5. El usuario selecciona una de las siguientes opciones: C: Si D: No	7. El Sistema solicita al Gestor que termine el proceso de grabación seleccionado. (Ver flujo alternativo 7a.)
6. El usuario selecciona la opción C. (Ver flujo alternativo 6a.)	8. El caso de uso termina cuando el Sistema elimina la grabación seleccionada de la lista de grabaciones en ejecución.
Prototipo de Interfaz “Terminar Grabación Manual”.	
Interfaz 1	

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.



Flujos Alternos “Terminar Grabación Manual”.

Acción del Actor	Respuesta del Sistema
6a. El usuario selecciona la opción C.	7a. El Sistema cierra el mensaje de confirmación y cancela la operación, terminando así el caso de uso.
Poscondiciones:	Se agrega una nueva tarea de grabación en la interfaz principal.

Sección “Grabar por Calendario”.

Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando se ejecuta una tarea programada.	2. El Sistema envía al Gestor una petición de inicio de grabación. 3. El Gestor notifica al sistema si puede comenzar a grabar o no. 4. Si el Gestor notifica que puede comenzar a grabar y el sistema comienza la grabación, terminando así el caso de uso. De lo contrario ver flujo alternativo 4a.

Prototipo de Interfaz (No tiene Interfaz)

Flujos Alternos “Grabar por Calendario”.

Acción del Actor	Respuesta del Sistema
	4a. El Gestor notifica al Sistema que no se puede realizar la grabación.
Poscondiciones:	Comienza una grabación por calendario.

Tabla 3. Descripción del Caso de Uso Grabar Flujo de Video.

Caso de Uso:	Visualizar cámaras.
Actores:	Usuario con Privilegios.
Resumen:	El caso de uso se inicia cuando el Usuario con Privilegios se ha autenticado en el

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

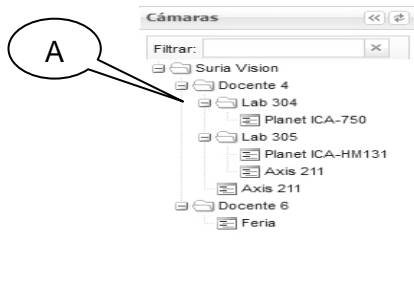
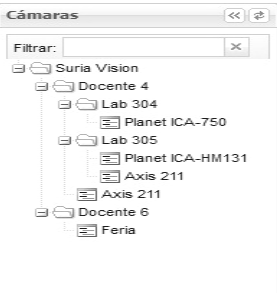


	Sistema y termina cuando el Sistema muestra una vista de árbol que incluye todas las cámaras manejadas por la aplicación.
Precondiciones:	El Gestor debe estar “Disponible”. El usuario tiene que tener permisos para realizar esta acción.
Referencias	RF 4
Prioridad	Crítico
Flujo Normal de Eventos “Visualizar Cámaras”.	
Sección “Visualizar Cámaras”.	
Acción del Actor	Respuesta del Sistema
1. El caso de uso inicia cuando el usuario accede a la interfaz de grabación.	1. El Sistema carga la lista de cámaras desde el Gestor. 2. El caso de uso termina cuando el Sistema construye una vista en forma de árbol con la jerarquía de cámaras, agrupadas por según su ubicación física (Interfaz 1) (A).
Prototipo de Interfaz “Visualizar Cámaras”.	
Interfaz 1	
 <p>The screenshot shows a window titled 'Cámaras' with a search bar labeled 'Filtrar:'. Below the search bar is a tree view of camera locations. The tree structure is as follows:</p> <ul style="list-style-type: none"> Suria Vision <ul style="list-style-type: none"> Docente 4 <ul style="list-style-type: none"> Lab 304 <ul style="list-style-type: none"> Planet ICA-750 Lab 305 <ul style="list-style-type: none"> Planet ICA-HM131 Axis 211 Axis 211 Docente 6 <ul style="list-style-type: none"> Feria 	
Poscondiciones:	Se muestra un árbol de cámaras agrupadas según la zona física en la que se encuentran.

Tabla 4. Descripción del Caso de Uso Visualizar Cámaras.

Caso de Uso:	Visualizar Calendario de Grabación.
Actores:	Usuario con Privilegios.
Resumen:	El caso de uso se inicia cuando el Usuario con Privilegios selecciona la opción “Mostrar Calendario en forma de Listas Horizontales” o cuando selecciona la opción “Mostrar Calendario en forma de Listas Verticales”.
Precondiciones:	El Gestor debe estar “Disponible”. El usuario tiene que tener permisos para realizar esta acción.

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

Referencias	RF 2
Prioridad	Crítico
Flujo Normal de Eventos “Visualizar Calendario de Grabación”.	
Sección “Visualizar Calendario en forma de Listas Horizontales”.	
Acción del Actor	Respuesta del Sistema
<p>1. El caso de uso inicia cuando el usuario accede a la aplicación y selecciona una de las cámaras o grupos listados en el árbol de cámaras (Interfaz 1).</p> <p>2. El usuario selecciona la opción “Visualizar Calendario en forma de Listas Horizontales” (Interfaz 2) (A).</p>	<p>3. El caso de uso termina cuando el Sistema muestra las reglas y las tareas de grabación en forma de listas horizontal para la cámara o el grupo seleccionado (Interfaz 3) (B).</p>
Prototipo de Interfaz “Visualizar Calendario en forma de Listas Horizontales”.	
<p>Interfaz 1</p> 	
<p>Interfaz 2</p> 	
<p>Interfaz 3</p> 	
Sección “Visualizar Calendario en forma de Listas Verticales”.	
Acción del Actor	Respuesta del Sistema

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

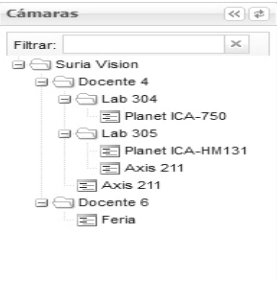
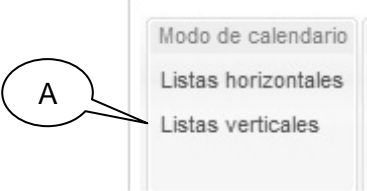
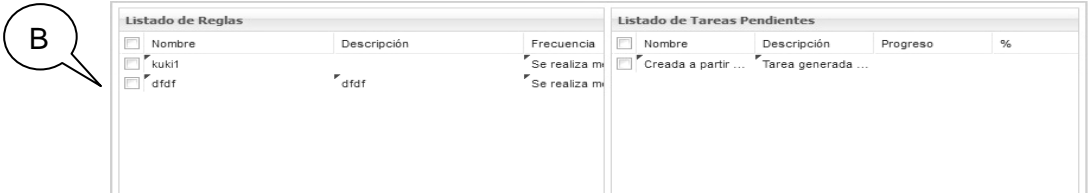
<p>1. El caso de uso inicia cuando el usuario accede a la aplicación y selecciona una de las cámaras o grupos listados en el árbol de cámaras (Interfaz 1).</p> <p>2. El usuario selecciona la opción “Visualizar Calendario en forma de Listas Verticales” (Interfaz 2) (A).</p>	<p>2. El caso de uso termina cuando el Sistema muestra las reglas y las tareas de grabación en forma de listas vertical para la cámara o grupo seleccionado (Interfaz 3) (B).</p>
Prototipo de Interfaz “Visualizar Calendario en forma de Listas Verticales”.	
<p>Interfaz 1</p>  <p>Interfaz 2</p>  <p>Interfaz 3</p> 	
Poscondiciones:	Se muestra las reglas y tareas de grabación según el nodo seleccionado en el árbol de cámaras.

Tabla 5. Descripción del Caso de Uso Visualizar Calendario de Grabación.

3.9. Conclusiones.

Con la realización de este capítulo se le dio cumplimiento a las tareas de la investigación propuestas para el mismo. Presentándose dentro de los resultados obtenidos, los requisitos funcionales y no funciones del sistema, el modelo de dominio y los casos de uso (CU) con sus respectivas descripciones. Todos estos artefactos en su conjunto facilitarán la comprensión de las funcionalidades

CAPÍTULO III. PRESENTACIÓN DE LA SOLUCIÓN PROPUESTA.

identificadas, brindando al desarrollador una mejor visión del sistema y un orden de prioridad en el desarrollo de los CU propuestos según su criticidad.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

CAPÍTULO IV. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

4.1. Introducción.

En este capítulo se realizará el diseño, implementación y pruebas del sistema, siguiendo el flujo de trabajo de la metodología RUP. Describiéndose la construcción de la aplicación, las principales características del patrón arquitectónico a utilizar, así como también los patrones de diseños empleados, los diagramas de clases del diseño y las pruebas de caja negra que se realizarán luego de terminada la aplicación.

4.2. Arquitectura.

La arquitectura del software es la forma en que se organizan e interactúan los componentes del sistema. Por tanto antes de definir la arquitectura para el sistema modular propuesto en la presente investigación es necesario conocer las particularidades del sistema Suria Vision con el cual tiene una estrecha relación. El Sistema posee una arquitectura en pizarra, en su variante tablero de control. Esta desacopla el sistema en componentes denominados agentes autónomos, estos suelen ser programas especializados en una tarea concreta o elemental. Todos ellos cooperan para alcanzar una meta común, son independientes en la realización atómica de su funcionalidad, pero dependen de una entrada de información externa, que es provista por otros agentes, y a su vez producen un resultado que puede ser entrada de otros agentes.

Cada agente tiene sus propios objetivos, desconoce los objetivos de los demás, y tampoco conoce el objetivo global (la solución del problema). Sin embargo, se produce una cooperación inconsciente entre ellos que lleva a una meta más importante. También se rigen por interfaces estándares que permiten cambios en el ambiente externo del mismo sin que este sufra modificaciones en su funcionamiento interno. Todo el trabajo de los agentes autónomos, está coordinado por un elemento central, denominado Repositorio Activo, el cual entrega y recibe información de los agentes y coordina su funcionamiento. En el caso del sistema Suria Vision, el gestor que es el módulo fundamental de la aplicación cumple con la función de Repositorio Activo y todos los demás son agentes que se encargan de realizar tareas específicas. Todo el tráfico de información al pasar por el Gestor es fácil de controlar y supervisar, además de que se gana en flexibilidad en el sistema teniéndolo a él como eje central de la aplicación (41).

4.3. Arquitectura del Módulo Web.

De forma general el Sistema Suria Vision se encuentra diseñado con la arquitectura en pizarra, pero la interfaz del cliente de grabación y reporte se encuentra diseñada bajo el patrón arquitectónico en Capas, definiéndose la utilización de dos capas. Este patrón descompone una aplicación en un conjunto de capas independientes y ordenadas jerárquicamente. Cada nivel o capa usa los servicios de la capa inmediatamente inferior y ofrece servicios a la inmediatamente superior. Permite estructurar

CAPÍTULO IV. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

aplicaciones que se pueden descomponer en grupos de subtareas, donde cada grupo está en un determinado nivel de abstracción, ver figura 4.

Entre sus principales ventajas se encuentra que admite una amplia reutilización del código generado, permitiendo el fraccionamiento de problemas complejos y facilitando la localización de errores. A continuación se muestran las capas:

- **Capa de presentación:** esta capa reúne todos los aspectos del software que tiene que ver con las interfaces y la interacción con los usuarios.
- **Capa lógica del negocio:** en esta capa se concentra el código implementado, donde se responde a las solicitudes del usuario, permitiendo automatizar los procesos de negocio que lleva a cabo el usuario (42).

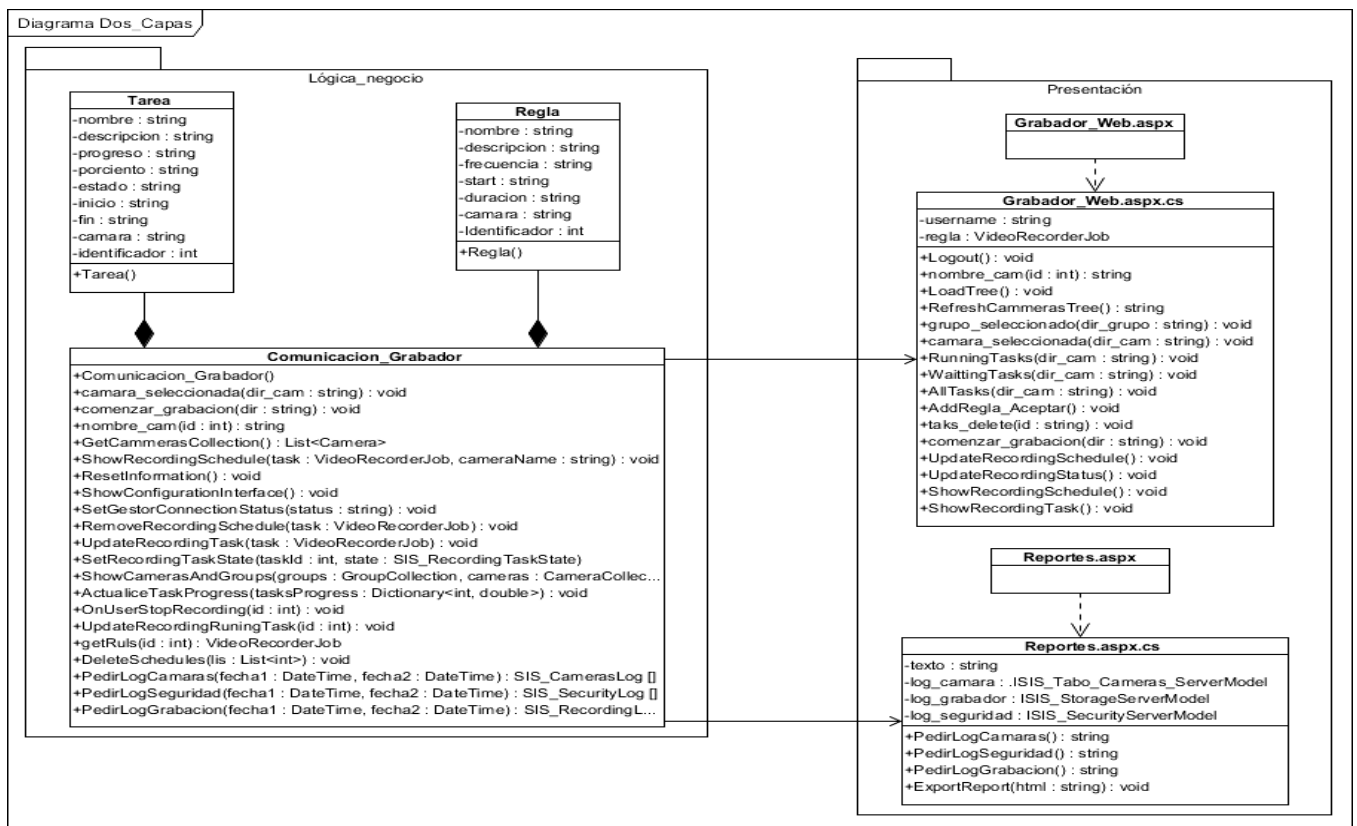


Figura 3. Representación del patrón arquitectónico dos capas en la aplicación.

4.4. Patrones de Diseño de Software.

Para construir una solución eficaz se necesita más que una arquitectura, ya que existen otros parámetros a tener en cuenta para realizar el diseño del software, entre estos se encuentran los patrones de diseño, los cuales se refieren a cuestiones y aspectos fundamentales del diseño.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

Se emplearán cinco patrones GRASP¹⁴ y dos patrones GoF¹⁵, entre los que se encuentran:

Patrones GRASP:

- **Experto:** garantizará que cada clase cumpla con sus responsabilidades, según la información que contenga y las funcionalidades que se deseen implementar. Este patrón se emplea en las clases, Regla y Tarea ya que cada una de ellas se especializa en crear reglas y tareas únicamente.
- **Creador:** cada clase instancia y crea las clases que le son necesarias para cumplir sus funcionalidades, el mismo se emplea en la clase Comunicacion_Grabador la cual contiene objetos de las clases con las cuales se comunica.
- **Bajo Acoplamiento:** permite la reutilización aumentando la productividad, el diseño de las clases más independientes y reduce el impacto de los cambios. En la figura anterior se ve evidenciado dicho patrón, ya que las clases están relacionadas de manera que se crean sólo las dependencias necesarias para desempeñar sus responsabilidades.
- **Alta Cohesión:** permite simplificar el mantenimiento y brindar un alto grado de funcionalidad, es utilizado en la clase Comunicacion_Grabador.
- **Controlador:** permite asignar la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas, se utiliza en la clase Comunicacion_Grabador que se encarga de controlar todas las acciones que se realizan en la aplicación (42).

Patrones Gof:

- **Fachada:** provee una interfaz unificada simple para acceder a una interfaz o grupo de interfaces de un subsistema, se emplea en la interfaz ISIS_ClientPuppet la cual sirve de portada para la comunicación con la clase SIS_ClientMaster (42).

4.5. Análisis y Diseño.

En este epígrafe se realizará el análisis y diseño de la aplicación, lo cual permitirá obtener una mejor información sobre cómo estará estructurado el sistema y las pautas a seguir para su futura implementación.

4.5.1. Diagramas del análisis.

El análisis se encarga de describir qué hace el sistema, apoyándose solamente en los requisitos funcionales y en el análisis de la descripción de cada caso de uso, el cual facilita la construcción del diseño (43).

¹⁴ GRASP (General Responsibility Assignment Software Patterns): Patrones generales para asignar responsabilidades.

¹⁵ GoF(Gang of Four):Pandilla de los cuatro.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

A continuación, se muestran los diagramas de análisis de los tres primeros casos de uso, por ser los identificados como los CU más críticos para el sistema, teniendo en cuenta su funcionamiento en la aplicación. Los restantes diagramas pueden encontrarse en el Anexo 2 del presente trabajo.

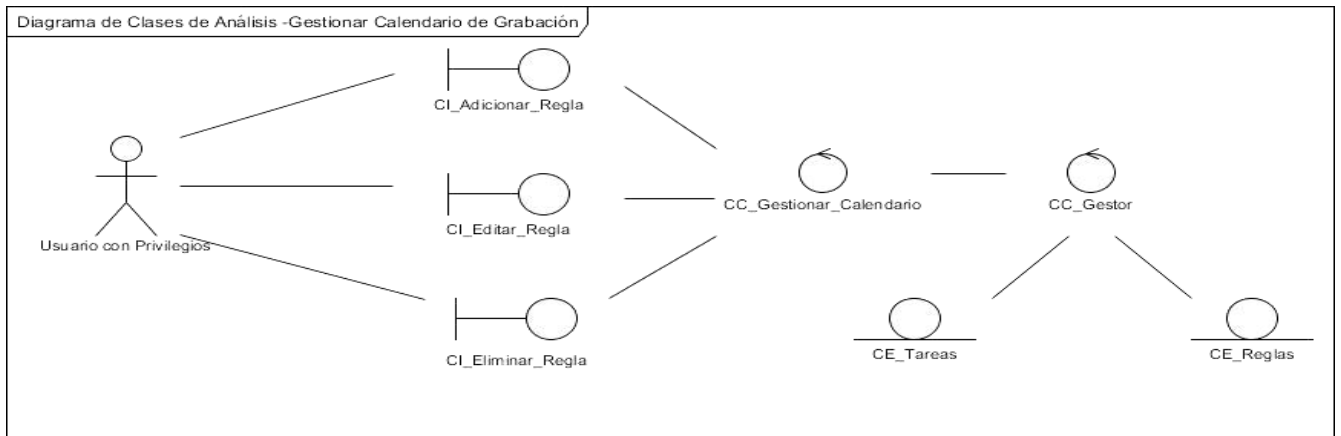


Figura 4. Diagrama de Análisis.CU Gestionar Calendario de Grabación.

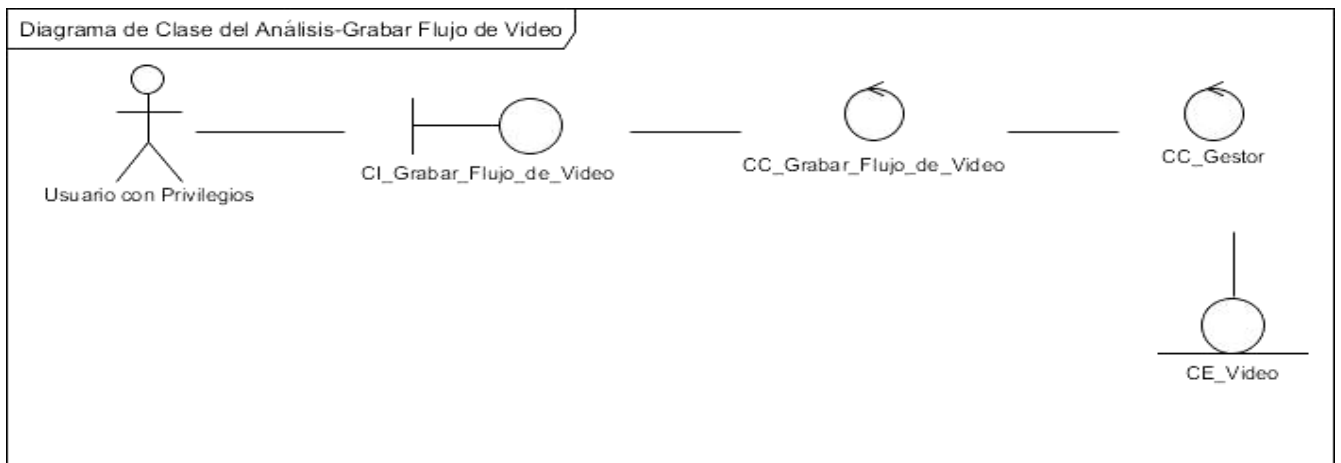


Figura 5. Diagrama de Análisis.CU Grabar Flujo de Video.

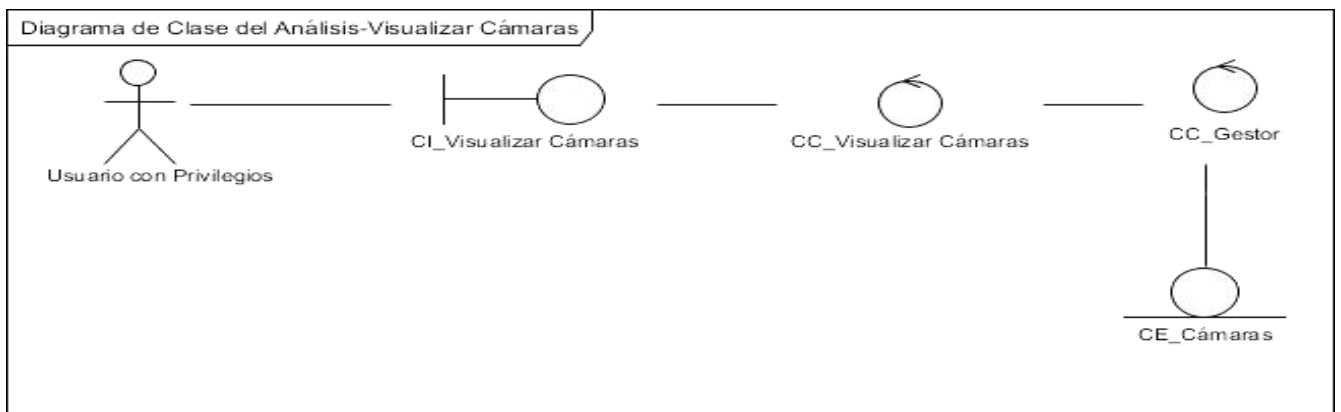


Figura 6. Diagrama de Análisis.CU Visualizar Cámaras.

CAPÍTULO IV. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

4.5.2. Modelo de Diseño.

El Modelo de diseño describe la realización física de los casos de uso centrándose en el cumplimiento de los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación. Además sirve de abstracción a la implementación del sistema, lo cual facilita el trabajo de los desarrolladores ya que es tomado como principal entrada de la fase de implementación pues describe las interfaces y clases que se van a utilizar (43).

4.5.2.1. Diagramas de Clases del Diseño.

Un diagrama de clases es un diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño para lograr una mejor interpretación de la implementación del sistema. Estos diagramas describen de las clases sus atributos, métodos y las relaciones entre ellas (43).

A continuación, se muestran los diagramas de clases del diseño de los tres primeros casos de uso del sistema. Los diagramas correspondientes a los casos de uso restantes fueron ubicados en el Anexo 3 del presente trabajo.

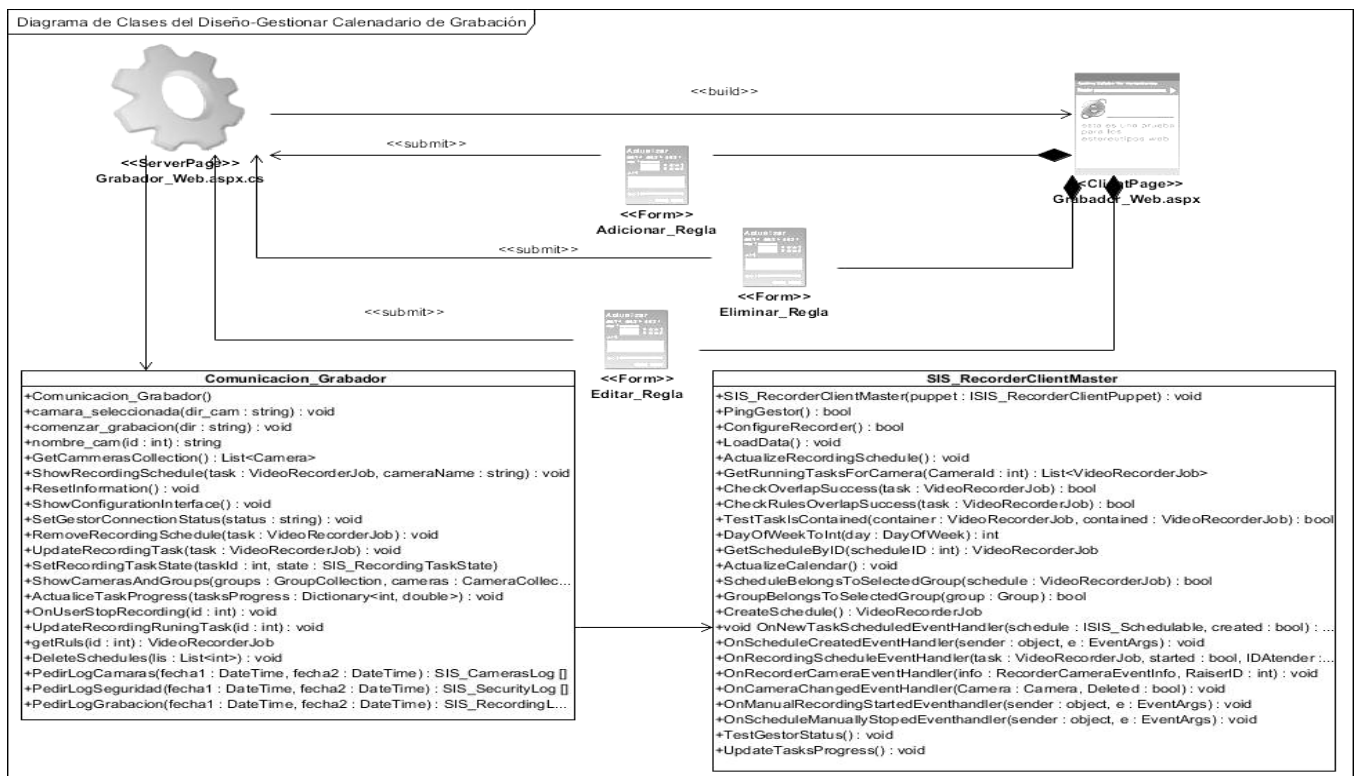


Figura 7. Diagrama de Clases de Diseño.CU Gestionar Calendario de Grabación.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

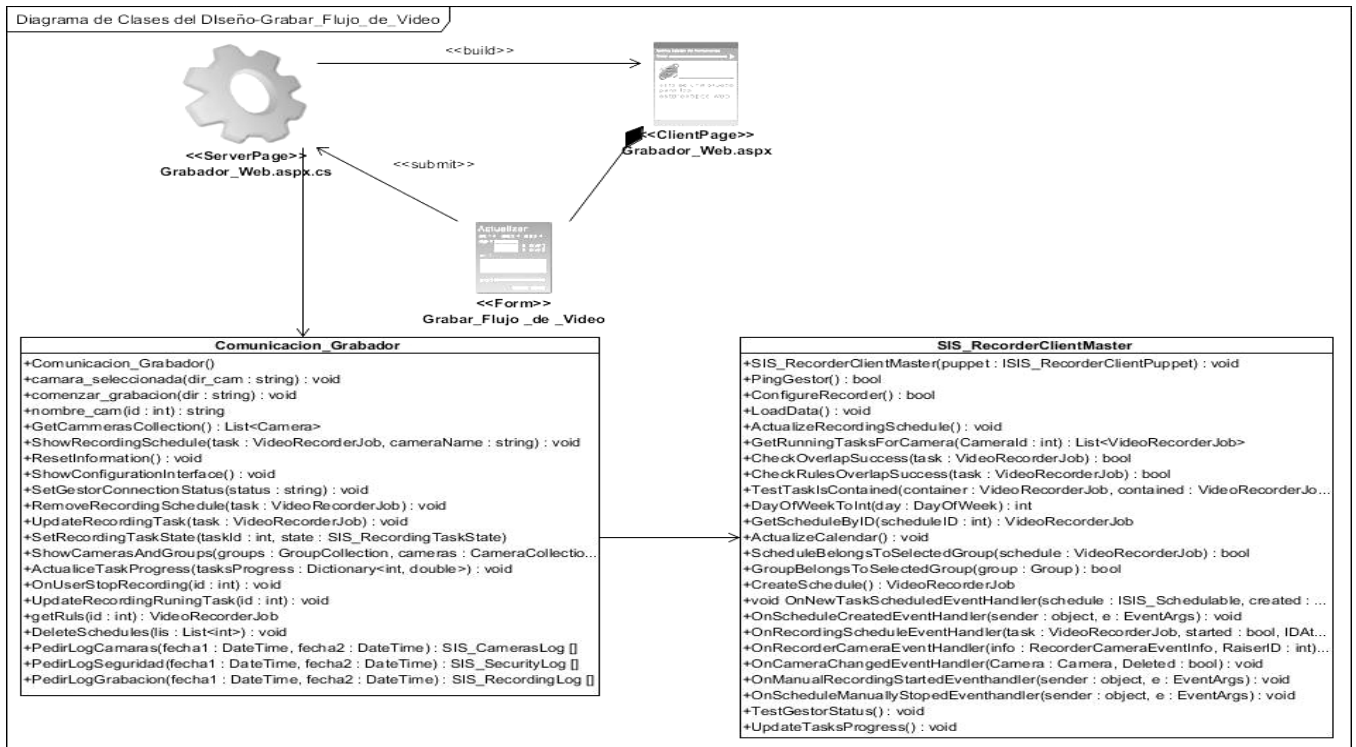


Figura 8. Diagrama de Clases de Diseño.CU Grabar Flujo de Video.

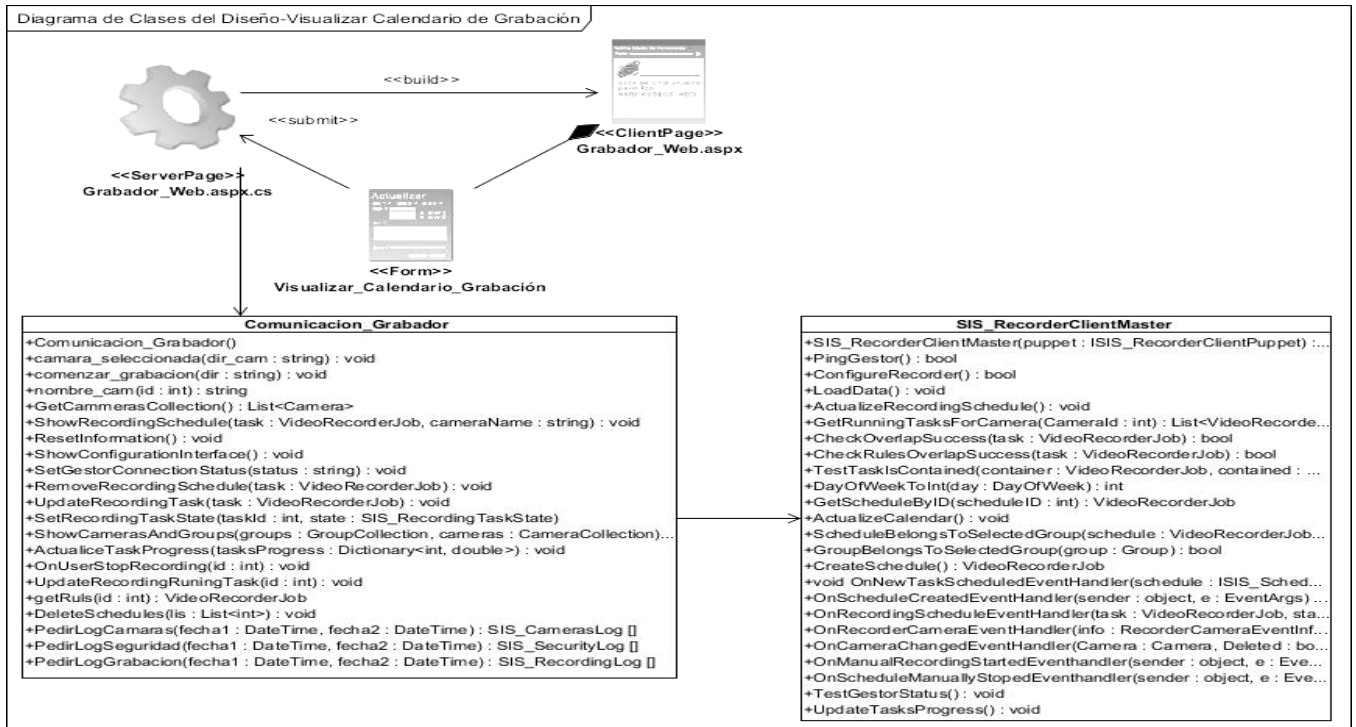


Figura 9. Diagrama de Clases de Diseño.CU Visualizar Calendario de Grabación.

4.6. Modelo de Implementación.

El modelo de implementación está compuesto por un conjunto de subsistemas y componentes que establecen la composición física de la implementación del sistema, tiene como objetivos implementar las clases de diseño como componentes, asignar los componentes a los nodos¹⁶, probar los componentes individualmente, integrar los componentes en un sistema ejecutable. Utiliza los Diagramas de Componentes y Diagrama de Despliegue para comprender cómo se organizan y dependen unos de otros (43).

4.6.1. Diagrama de Despliegue.

El Diagrama de Despliegue es utilizado para mostrar las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Describen la topología del sistema, la estructura de los elementos de hardware y el software que se ejecuta. Muestra los dispositivos que se encuentran en un sistema y su distribución en el mismo. En la siguiente figura se muestra el diagrama de despliegue correspondiente a la aplicación (43).

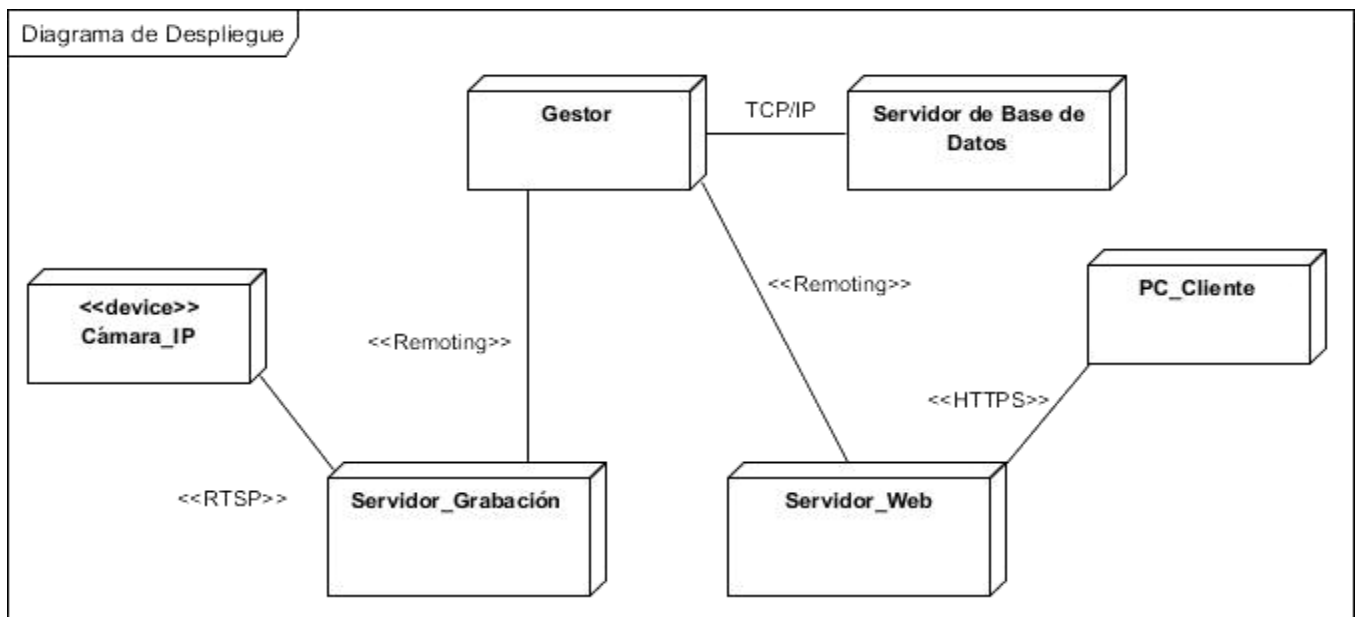


Figura 10. Diagrama de Despliegue.

4.6.2. Diagrama de Componentes.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones

¹⁶ **Nodos:** Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria.

CAPÍTULO IV. CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

informáticas. Pueden ser simples archivos, paquetes y bibliotecas cargadas dinámicamente. A continuación se muestra en la siguiente figura el diagrama de componente de la aplicación (43).

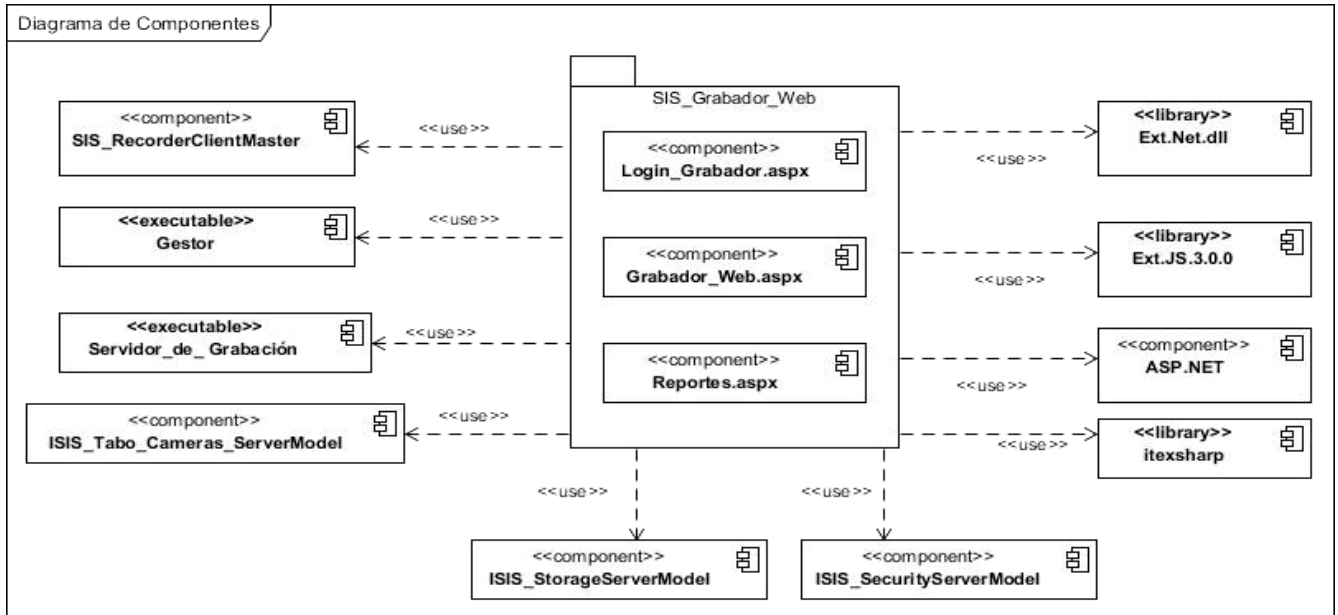


Figura 11. Diagrama de Componentes.

4.7. Métodos de Pruebas.

Algo que es de vital importancia durante el desarrollo de software es la realización de pruebas. Las pruebas no son más que técnicas de validación de software para verificar el resultado de la implementación probando cada construcción. El objetivo de estas pruebas es descubrir errores en la aplicación que antes no se habían descubierto. Existen dos métodos para la realización de las mismas, las pruebas de caja blanca o estructural y las pruebas de caja negra o funcional siendo estas últimas las que se le realizarán a la aplicación, ya que el principal objetivo de estas pruebas es verificar las entradas y salidas de los datos. Además se realizarán pruebas de usabilidad para comprobar el cumplimiento del requisito no funcional de usabilidad, ambas se realizan en el nivel de pruebas del sistema (43).

4.7.1. Pruebas de Caja Negra.

Estas pruebas tienen por objetivo comprobar que los sistemas desarrollados cumplan con las funciones específicas para los cuales han sido creados, el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y los posibles resultados ante las salidas reales de la aplicación, esto generalmente se define en los casos de prueba preparados antes del inicio de las pruebas (43).

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

Técnica de Partición equivalente

El método de caja negra contiene varias técnicas entre las que se encuentran, análisis de valores límite (AVL), prueba de tabla ortogonal y partición equivalente siendo esta ultima la empleada para la realización de las pruebas de la aplicación desarrollada. Esta técnica divide el dominio de entrada de un programa en clases de datos válidos e inválidos, a partir de las cuales pueden derivarse casos de prueba. La partición equivalente se esfuerza por definir un caso de prueba que descubra ciertas clases de errores, reduciendo así el número total de casos de prueba que deben desarrollarse (44).

Se realizó un diseño de Caso de Prueba (CP) para cada CU, teniendo un total de 7CP. Como resultado de la primera iteración de las pruebas se obtuvieron 9 no conformidades de ellas 4 altas, 3 medias y 2 bajas en la segunda iteración los resultados fueron satisfactorios, lo que significa que la aplicación desarrollada cumple con la calidad requerida, ver tabla 6 y figura 15.

Caso de Prueba “Caso de Uso Gestionar Calendario de Grabación”.					
Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo Central	Resultados Esperados	Resultados Obtenidos
SC 1 Adicionar Regla de Grabación.	EC 1.1: Adicionar Regla de Grabación.	El usuario con privilegios adiciona reglas de grabación para ello el sistema muestra una interfaz para introducir los datos generales de la regla de grabación, el usuario introduce los datos y si estos están correctamente y la regla de grabación definida no se solape con una de las ya existentes, el sistema envía los datos al Gestor, si la regla es aceptada por este el	Módulo Grabador/ /Seleccionar una cámara/en Reglas de Grabación/ botón Adicionar Regla de Grabación. Módulo Grabador/ /clic derecho sobre una cámara/opción Adicionar Regla de Grabación.	El sistema debe adicionar correctamente la regla de grabación.	El sistema adiciona correctamente la regla de grabación.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

		sistema genera una nueva regla y con ella las tareas correspondientes a la misma.			
EC 1.2: Cancelar Adicionar Regla de Grabación.	El usuario selecciona la opción "Cancelar". El sistema cierra la ventana (wizard) y termina el Caso de Uso.	Módulo Grabador/ /Seleccionar una cámara/en Reglas de Grabación/ Opción Adicionar Regla de Grabación/ botón "Cancelar". Módulo Grabador/ /clic derecho sobre una cámara / Opción Adicionar Regla de Grabación/ botón "Cancelar". Otro camino puede ser: Módulo Visor/ Botón Cliente de Grabación /clic derecho sobre una cámara/ Adicionar regla/botón Cancelar.	El sistema no debe adicionar la regla de grabación.	El sistema no adiciona la regla de Grabación.	
EC 1.3: Siguiete Adicionar Regla de Grabación.	El usuario selecciona la opción "Siguiete". El sistema muestra la siguiente ventana (wizard) y termina el Caso de Uso.	Módulo Grabador / /Seleccionar una cámara/en Reglas de Grabación/ Opción Adicionar Regla de Grabación/ botón "Siguiete".	El sistema debe mostrar la siguiente ventana.	El sistema muestra la siguiente ventana.	

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

			Módulo Grabador / /clic derecho sobre una cámara/ Opción Adicionar Regla de Grabación/ botón “Siguiete”.		
EC: 1.4: Fallas en Adicionar Regla de Grabación.	El usuario con privilegios adiciona reglas de grabación, dejando algún campo vacío, el sistema muestra un mensaje indicándole al usuario que llene todos los campos. Mensaje: “Llene los datos requeridos”.	Módulo Grabador / Seleccionar una cámara/en Reglas de Grabación/ Opción Adicionar Regla de Grabación/ botón “Siguiete” (se repite este paso dos veces)/botón “Aceptar”. Módulo Grabador / clic derecho sobre una cámara / Opción Adicionar regla / botón “Siguiete” (se repite este paso dos veces)/ botón “Aceptar”. Módulo Visor/ Botón Cliente de grabación /Seleccionar una cámara/en Reglas de Grabación/ Opción Adicionar Regla de grabación/ botón “Siguiete” ”/ botón “Siguiete” / botón “Aceptar”.	El sistema debe mostrar un mensaje indicando que se deben llenar todos los campos requeridos.	El sistema muestra un mensaje “Llene los campos requeridos ”.	
EC 1.5:	Si la regla adicionada se solapa	Módulo Grabador / Seleccionar	El sistema	El sistema	

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

	Adicionar Regla de grabación cuando la regla adicionada se solapa.	con alguna existente, el sistema muestra un mensaje indicando que no puede ser creada, porque se solapa con una regla existente y de esta forma termina el Caso de Uso. Mensaje: “La regla que usted ha especificado se solapa con otra regla previamente existente”.	una cámara/en Reglas de Grabación/ Opción Adicionar Regla de grabación/ (Se llenan todos los campos) botón “Siguiente” (se repite este paso tres veces)/botón “Aceptar”. Módulo Grabador / clic derecho sobre una cámara/ Adicionar regla/(Se llenan todos los campos) botón “Siguiente” (se repite este paso tres veces)/botón “Aceptar”.	debe mostrar un mensaje indicando que la regla adicionada se solapa.	muestra un mensaje “La regla que usted ha especificado se solapa con otra regla previamente existente ”.
SC 2	EC 2.1:	El usuario con privilegios selecciona una regla de grabación y pulsa la botón “Editar Regla de Grabación”, modifica los datos, el sistema verifica que todos los datos están llenos y que la regla de grabación no se solape con una de las ya existente, el sistema solicita al Gestor que modifique la regla de grabación en la base de datos.	Módulo Grabador / Seleccionar una regla en el listado de regla / Opción Editar Regla de Grabación/se modifican todos los campos y pulsa botón “Siguiente” (se repite este paso, dos veces)/ se modifican los campos, clic botón “Aceptar”.	El sistema debe modificar correctamente la regla de grabación.	El sistema modifica correctamente la regla de grabación.
	EC 2.2:	El usuario selecciona la opción	Módulo Grabador / Seleccionar	El sistema no	El sistema no

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

	<p>Cancelar Editar Regla de Grabación.</p>	<p>“Cancelar”. El sistema cierra la ventana y termina el Caso de Uso.</p>	<p>una regla en el listado de regla / Opción Editar Regla de Grabación/Se modifican o no los campos y pulsa botón “Cancelar”. Módulo Grabador / Seleccionar una regla en el listado de regla / Opción Editar Regla de Grabación/Se modifican todos los campos y pulsa botón “Siguiente”/ se modifican o no todos los campos y pulsa botón “Cancelar”.</p>	<p>debe modificar la regla de grabación seleccionada.</p>	<p>modifica la regla de grabación seleccionada.</p>
	<p>EC: 2.3: Editar Regla de Grabación con campos vacíos.</p>	<p>El usuario con privilegios desea modificar alguna regla, dejando algún campo vacío, el sistema muestra un mensaje indicándole al usuario que debe llenar todos los campos. Mensaje: “Llene los datos requeridos”</p>	<p>Módulo Grabador / Seleccionar una regla en el listado de regla / Opción Editar Regla de Grabación/Se modifican dejando campos en blanco. Módulo Grabador / Seleccionar una regla en el listado de regla / Opción Editar Regla de Grabación/(Se modifican los campos) botón “Siguiente” (se repite este paso dos veces)/botón</p>	<p>El sistema debe mostrar un mensaje indicando de se deben llenar todos los campos requeridos.</p>	<p>El sistema muestra un mensaje “Llene los datos requeridos ”.</p>

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

			“Aceptar”		
SC 3 Eliminar Regla de Grabación.	EC 3.1: Eliminar Regla de Grabación.	El usuario con privilegio selecciona una o varias reglas de grabación, selecciona la opción de eliminar reglas seleccionadas, el sistema pide confirmación al usuario, confirma que desea eliminar, el Gestor elimina las reglas de grabación de la base de datos.	Módulo Grabador / Seleccionar una regla o varias reglas en el listado de reglas / botón Eliminar las reglas seleccionadas/ botón “Sí”.	El sistema debe pedir confirmación al usuario para eliminar la(s) regla(s) de grabación, si se confirma, el sistema debe eliminar la(s) regla(s).	El sistema pide confirmación al usuario para eliminar la(s) regla(s) de grabación, si se confirma el sistema elimina la(s) regla(s).
	EC 3.2: Eliminar Regla de Grabación cuando no se confirma.	El usuario con privilegios no confirma que desea eliminar la(s) regla(s) de grabación, entonces el sistema cierra el diálogo y no se elimina la(s) regla(s) seleccionada(s).	Módulo Grabador / Seleccionar una regla o varias reglas en el listado de reglas / botón Eliminar las reglas seleccionadas/ botón “No”.	El sistema no debe eliminar la(s) regla(s) de grabación seleccionada.	El sistema no elimina la(s) regla(s) de grabación seleccionada(s)

Tabla 6.Pruebas de caja negra.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

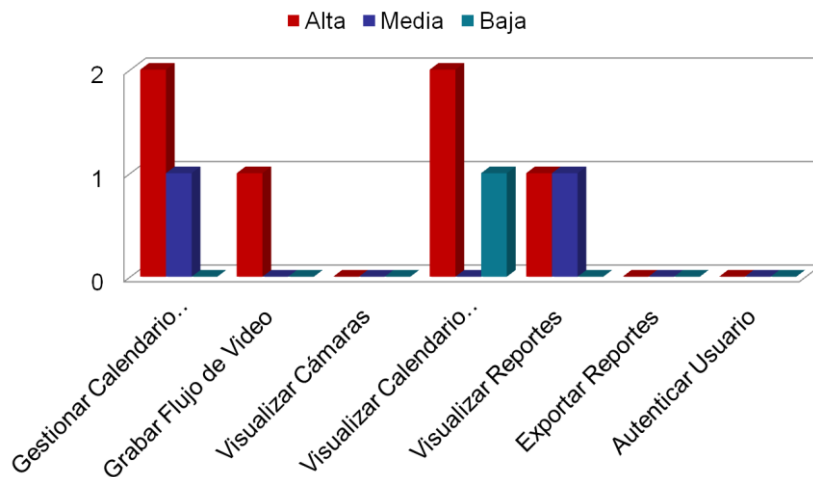


Figura 12.Representación de las pruebas de caja negra.

4.7.2. Pruebas de Usabilidad

Para validar el requisito de usabilidad propuesto en la investigación se realizaron pruebas a la aplicación, para ello se tomó una población de 10 personas, asumiendo que los usuarios con conocimientos bajos no han interactuado con un software, los de conocimientos medios han trabajado con aplicaciones sencillas y los de conocimientos altos son profesionales que están capacitados en estos temas, ver tabla 7:

Encuestados	Profesión	Conocimientos informáticos
3	Profesores del departamento de humanidades.	Bajos
5	Estudiantes de 2do y 3er año de informática.	Medios
2	Ingenieros en ciencias informáticas sin conocimientos del negocio.	Altos

Tabla 7.Pruebas de usabilidad.

A todos los encuestados se les explicó de forma breve el objetivo de la aplicación propuesta y sus principales funcionalidades. Luego se les permitió interactuar con la aplicación un tiempo determinado (de 10 a 15 minutos). Obteniéndose los siguientes resultados:

- De los usuarios con conocimientos bajos en informática uno de ellos interactuó con la aplicación satisfactoriamente, el resto mostró dificultad para realizar algunas de las funcionalidades.
- De los usuarios con conocimientos medios en informática, uno tuvo dificultad para realizar algunas operaciones; con el resto se obtuvieron los resultados esperados.

CAPÍTULO IV.CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA.

- Respecto a los usuarios con conocimientos altos en informática los resultados fueron satisfactorio.

Resultados: de un total de 10 encuestados, 7 de ellos interactuaron correctamente con la aplicación y el resto mostró dificultad para realizar algunas operaciones especificadas. Lo que garantiza el cumplimiento del requisito de usabilidad propuesto, ver figura 17.

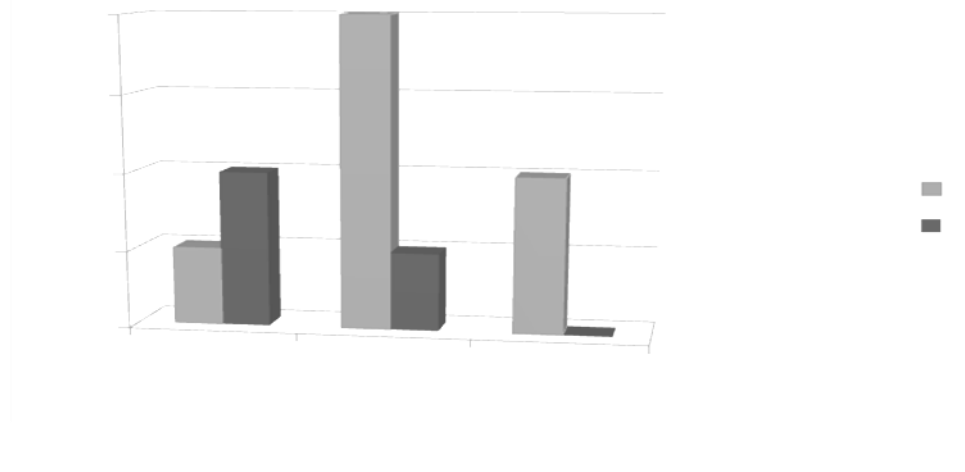


Figura 13. Representación de las pruebas de usabilidad.

4.8. Conclusiones.

En este capítulo se le dio cumplimiento a las tareas de la investigación propuestas para la realización del mismo, logrando obtener como resultado la arquitectura del sistema y los patrones de diseño que se emplearon, lo cual indicó al desarrollador como iba a estar organizada la aplicación. Se obtuvieron los diagramas de Clases del Diseño, Despliegue y Componentes para brindar una descripción más detallada de la estructura del sistema. Llegando finalmente a la realización de las pruebas funcionales y de usabilidad, obteniendo un software que cumple con todas las funcionalidades y calidad necesaria para su correcto funcionamiento.

CONCLUSIONES GENERALES

CONCLUSIONES GENERALES

Con el presente trabajo de diploma se logró dar cumplimiento a todas las tareas de la investigación propuestas inicialmente, obteniendo como resultado una interfaz web para el módulo de grabación y reporte del sistema de video vigilancia Suria Vision, llegando a las siguientes conclusiones:

1. Los sistemas analizados de video vigilancia, presentan como limitante que el hardware depende completamente del fabricante, además de que tienen un alto precio lo cual trae como consecuencia que no sean fácil de adquirir.
2. Se desarrolló una interfaz web que permite realizar la grabación de los flujos de video que se obtienen a través de las cámaras IP, ya sea de forma manual o por calendario, lo cual facilitará la realización de la planificación de las grabaciones que se deseen programar.
3. Se desarrolló un módulo que permite visualizar y exportar los reportes generados, para facilitar el acceso a la información deseada y de esta forma tener un mejor control de los sucesos que ocurren en el sistema.
4. Con el desarrollo de la interfaz de grabación y reporte se logró brindar mayor extensibilidad y acceso al sistema de video vigilancia Suria Vision, pues no será necesario instalar el sistema para su utilización.

RECOMENDACIONES

RECOMENDACIONES

Durante la realización del presente trabajo se obtuvieron diferentes ideas que pueden permitir la realización de una aplicación web más completa, por tanto se recomienda:

1. Realizar estudios para la implementación de un módulo de recuperación accesible desde la web, que permita recuperar los videos almacenados en los servidores, hacer búsquedas en estos, servir los resultados al usuario y brindándole la posibilidad de reproducirlos.
2. Realizar estudios para la implementación de un módulo que permita la detección de movimientos en aplicaciones web, el cual active una tarea de grabación cuando se detecte algún movimiento, lo cual traerá como ventaja una mejor utilización y aprovechamiento del módulo grabador.

REFERENCIAS BIBLIOGRÁFICAS

REFERENCIAS BIBLIOGRÁFICAS

1. **Inteco.** Inteco. [Online] [Cited: Septiembre 15, 2011.] http://inteco.es/pressRoom/Prensa/Actualidad_INTECO/guia_videovigilancia.
2. **Articles Free Blogs.** Articles Web Development. [Online] [Cited: Septiembre 15, 2011.] <http://articulosobredesarrolloweb.blogspot.com/2009/10/la-historia-de-video-vigilancia.htm>.
3. **Catarina.** Catarina. [Online] [Cited: Septiembre 15, 2011.] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/lezama_l_a/capitulo1.pdf.
4. **RialTrade d.o.o.** Video Vigilancia. [Online] [Cited: Septiembre 15, 2011.] <http://www.videovigilancia.eu.com/empresa/nuestra-historia.html>.
5. **Axis.** Axis Communications. [Online] [Cited: Septiembre 16, 2011.] http://www.axis.com/es/prensa/pr2010/100205_axis_sicur_es.pdf.
6. **Kioskea .Net.** Kioskea .Net. [Online] [Cited: Septiembre 16, 2011.] <http://es.kioskea.net/contents/internet/protip.php3>.
7. **uServers Comunicaciones, S.C.** Users. [Online] [Cited: Septiembre 18, 2011.] http://web.userservers.net/ayuda/soluciones/dominios/que-es-una-direccion-ip_NTk.htm.
8. **Heitel.** Heitel. [Online] [Cited: Septiembre 18, 2011.] http://www.heitel.com/upload/downloads/es/04-data-sheets/analog_vs_digital_es.pdf.
9. **Unicromm.** Unicromm. [Online] [Cited: Septiembre 20, 2011.] http://www.unicrom.com/Tut_analogico_digital.asp.
10. **Definición.de.** Definición.de. [Online] [Cited: Septiembre 22, 2011.] <http://definicion.de/cable-coaxial/>.
11. **Carrasco, Ignacio.** Borrmarkt, SA. [Online] 2005. [Cited: Junio 5, 2012.] http://www.borrmarkt.es/articulo_seguritecna.php?id=2275.
12. **Casadomo.** Casadomo. [Online] [Cited: Septiembre 22, 2011.] <http://www.casadomo.com/noticiasDetalle.aspx>.
13. **ORACLE.** ORACLE. [Online] [Cited: junio 5, 2012.] <http://docs.sun.com/app/docs/doc/820-6183/6nhhega68?l=es&a=view>.
14. **Promonegocios.** Promonegocios. [Online] [Cited: Septiembre 23, 2011.] <http://www.promonegocios.net/mercadotecnia-servicios/definicion-servicios.html>.
15. **Mastermagazine.** Mastermagazine. [Online] [Cited: Septiembre 25, 2011.] <http://www.mastermagazine.info/termino/5610.php>.
16. **Cisco.** Cisco. [Online] [Cited: Septiembre 25, 2011.] <http://www.cisco.com/en/US/products/ps6940/index.html>.

REFERENCIAS BIBLIOGRÁFICAS

17. **Bitelia.** Bitelia. [Online] [Cited: Septiembre 25, 2011.] <http://bitelia.com/2010/10/videovigilancia-en-software-libre>.
18. **Vidium.** Vidium. [Online] [Cited: Septiembre 28, 2011.] <http://www.vidium.es/>.
19. **DATYS.** *Xyma Save Vision.Sistema de Video Vigilancia IP. White Paper.*
20. **Panasonic.** *Addendum for WV-CU950/650 and WJ-SX150 Series.*
21. **Video Vigilancia.** Video Vigilancia. [Online] [Cited: Septiembre 28, 2011.] <http://www.videovigilancia.eu.com/226-tarjeta-de-adquisicion-de-video-geovision-gv-600-4>.
22. **Remote-Security.** Remote-Security. [Online] [Cited: Septiembre 28, 2011.] <http://www.remote-security.com/geovision.htm>.
23. **Stanek, William Robert.** *Web Creación de Docuemntos al Descubierto.* 84-89660-98-0.
24. **HTMLPOINT.com.** HTMLPOINT.com. [Online] 2006. [Cited: Octubre 6, 2011.] http://www.htmlpoint.com/javascript/corso/js_02.htm.
25. **Microsoft Corporation.** Microsoft Corporation. [Online] 2000. [Cited: Octubre 10, 2011.] <http://msdn.microsoft.com/en-us/library/aa287558.aspx>.
26. **Serrano Pérez, Jorge.** *Programación con ASP.NET.* España : ANAYA MULTIMEDIA, 2002. 9788441513426 978-84-415-1342-6 8441513422 84-415-1342-2.
27. **Carrera, Dra Sara.** *Adquisicion semi-automtica del conocimiento: una arquitectura preliminar.* 2007.
28. **Hernández Sirvent, Javier.** *CREACIÓN DE UN SERVIDOR WEB CON SVN.*
29. **Santiago Marrero, Danieyis and Acosta Pereira, Pedro Orlando.** *Desarrollo del Módulo Web de Monitorización y Administración del Sistema de Video Vigilancia.* La Habana : s.n., 2011.
30. **Microsoft.** *Visual Studio 2010.*
31. **Wordpress.** Wordpress. [Online] [Cited: Octubre 15, 2011.] <http://wordpress.com/2010/11/07/top-10-beneficios-de-visual-studio-2010-professional/>.
32. **Torres Laguna, Angel.** *Propuesta de Servidor Streaming de software libre para la captura y transmisión de video y sonido digital.* La Habana : Universidad de las Ciencias Informáticas : s.n., 2009. TD_2726_09.
33. **García Sánchez, Antonio Javier, et al.** *Plataformas Middleware comerciales para la integración de flujos de vídeo bruto.* Cartagena : s.n., 2004. 1698-2924.
34. **Martin, Daniel Maldonado.** *SQLite, el motor de base de datos ágil y robusto.* 2008.
35. **Ponsoda Montiel, Daniel.** *Introducción a SQLite.*
36. **Hernández Orallo, Enrique.** *El Lenguaje Unificado de Modelado (UML).*
37. **Cueva Lovelle, Juan Manuel.** *Introducción a UML.Lenguaje para modelar Objetos.* 1999.
38. **Rueda Chacón, Julio César.** *Aplicación de la metodología RUP para el estándar J2EE.*

REFERENCIAS BIBLIOGRÁFICAS

39. Loja. [Online] Enero 2008. [Cited: Enero 15, 2012.] <http://www.utpl.edu.ec/eva/descargas/material/175/G18401.8.pdf>.
40. **Pressman, Roger S.** *Ingeniería del Software*. 2005. 970-10-5473-3.
41. **Semanat Aldana, Edmis Deivis and Verdecia Four, Leonor.** *Sistema de Video Vigilancia*. La Habana : s.n., 2009.
42. **Larman, Craig.** *UML y Patrones, Introducción al análisis y diseño orientado a objeto*. Mexico : Prentice Hall, 1999.
43. **Jacobson, Ivar, Booch, Grady and Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. Madrid : Perason Educacion, 2000. 84-7829-063-2.
44. **Pressman, Roger S.** *Ingeniería del Software, Un enfoque Práctico*. Sexta Edición. s.l. : McGraw-Hill.

BIBLIOGRAFÍA

BIBLIOGRAFÍA

1. **Inteco.** Inteco. [Online] [Cited: Septiembre 15, 2011.] http://inteco.es/pressRoom/Prensa/Actualidad_INTECO/guia_videovigilancia.
2. **Articles Free Blogs.** Articles Web Development. [Online] [Cited: Septiembre 15, 2011.] <http://articulosobredesarrolloweb.blogspot.com/2009/10/la-historia-de-video-vigilancia.htm>.
3. **Catarina.** Catarina. [Online] [Cited: Septiembre 15, 2011.] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/lezama_l_a/capitulo1.pdf.
4. **RialTrade d.o.o.** Video Vigilancia. [Online] [Cited: Septiembre 15, 2011.] <http://www.videovigilancia.eu.com/empresa/nuestra-historia.html>.
5. **Axis.** Axis Communications. [Online] [Cited: Septiembre 16, 2011.] http://www.axis.com/es/prensa/pr2010/100205_axis_sicur_es.pdf.
6. **Kioskea .Net.** Kioskea .Net. [Online] [Cited: Septiembre 16, 2011.] <http://es.kioskea.net/contents/internet/protip.php3>.
7. **uServers Comunicaciones, S.C.** Users. [Online] [Cited: Septiembre 18, 2011.] http://web.userservers.net/ayuda/soluciones/dominios/que-es-una-direccion-ip_NTk.htm.
8. **Heitel.** Heitel. [Online] [Cited: Septiembre 18, 2011.] http://www.heitel.com/upload/downloads/es/04-data-sheets/analog_vs_digital_es.pdf.
9. **Unicromm.** Unicromm. [Online] [Cited: Septiembre 20, 2011.] http://www.unicrom.com/Tut_analogico_digital.asp.
10. **Definición.de.** Definición.de. [Online] [Cited: Septiembre 22, 2011.] <http://definicion.de/cable-coaxial/>.
11. **Carrasco, Ignacio.** Borrmart, SA. [Online] 2005. [Cited: Junio 5, 2012.] http://www.borrmart.es/articulo_seguritecna.php?id=2275.
12. **Casadomo.** Casadomo. [Online] [Cited: Septiembre 22, 2011.] <http://www.casadomo.com/noticiasDetalle.aspx>.
13. **ORACLE.** ORACLE. [Online] [Cited: junio 5, 2012.] <http://docs.sun.com/app/docs/doc/820-6183/6nhhega68?l=es&a=view>.
14. **Promonegocios.** Promonegocios. [Online] [Cited: Septiembre 23, 2011.] <http://www.promonegocios.net/mercadotecnia-servicios/definicion-servicios.html>.
15. **Mastermagazine.** Mastermagazine. [Online] [Cited: Septiembre 25, 2011.] <http://www.mastermagazine.info/termino/5610.php>.
16. **Cisco.** Cisco. [Online] [Cited: Septiembre 25, 2011.] <http://www.cisco.com/en/US/products/ps6940/index.html>.

BIBLIOGRAFÍA

17. **Bitelia.** Bitelia. [Online] [Cited: Septiembre 25, 2011.] <http://bitelia.com/2010/10/videovigilancia-en-software-libre>.
18. **Vidium.** Vidium. [Online] [Cited: Septiembre 28, 2011.] <http://www.vidium.es/>.
19. **DATYS.** *Xyma Save Vision.Sistema de Video Vigilancia IP. White Paper.*
20. **Panasonic.** *Addendum for WV-CU950/650 and WJ-SX150 Series.*
21. **Video Vigilancia.** Video Vigilancia. [Online] [Cited: Septiembre 28, 2011.] <http://www.videovigilancia.eu.com/226-tarjeta-de-adquisicion-de-video-geovision-gv-600-4>.
22. **Remote-Security.** Remote-Security. [Online] [Cited: Septiembre 28, 2011.] <http://www.remote-security.com/geovision.htm>.
23. **Stanek, William Robert.** *Web Creación de Docuemntos al Descubierto.* 84-89660-98-0.
24. **HTMLPOINT.com.** HTMLPOINT.com. [Online] 2006. [Cited: Octubre 6, 2011.] http://www.htmlpoint.com/javascript/corso/js_02.htm.
25. **Microsoft Corporation.** Microsoft Corporation. [Online] 2000. [Cited: Octubre 10, 2011.] <http://msdn.microsoft.com/en-us/library/aa287558.aspx>.
26. **Serrano Pérez, Jorge.** *Programación con ASP.NET.* España : ANAYA MULTIMEDIA, 2002. 9788441513426 978-84-415-1342-6 8441513422 84-415-1342-2.
27. **Carrera, Dra Sara.** *Adquisicion semi-automtica del conocimiento: una arquitectura preliminar.* 2007.
28. **Hernández Sirvent, Javier.** *CREACIÓN DE UN SERVIDOR WEB CON SVN.*
29. **Santiago Marrero, Danieyis and Acosta Pereira, Pedro Orlando.** *Desarrollo del Módulo Web de Monitorización y Administración del Sistema de Video Vigilancia.* La Habana : s.n., 2011.
30. **Microsoft.** *Visual Studio 2010.*
31. **Wordpress.** Wordpress. [Online] [Cited: Octubre 15, 2011.] <http://wordpress.com/2010/11/07/top-10-beneficios-de-visual-studio-2010-professional/>.
32. **Torres Laguna, Angel.** *Propuesta de Servidor Streaming de software libre para la captura y transmisión de video y sonido digital.* La Habana : Universidad de las Ciencias Informáticas : s.n., 2009. TD_2726_09.
33. **García Sánchez, Antonio Javier, et al.** *Plataformas Middleware comerciales para la integración de flujos de vídeo bruto.* Cartagena : s.n., 2004. 1698-2924.
34. **Martin, Daniel Maldonado.** *SQLite, el motor de base de datos ágil y robusto.* 2008.
35. **Ponsoda Montiel, Daniel.** *Introducción a SQLite.*
36. **Hernández Orallo, Enrique.** *El Lenguaje Unificado de Modelado (UML).*
37. **Cueva Lovelle, Juan Manuel.** *Introducción a UML.Lenguaje para modelar Objetos.* 1999.
38. **Rueda Chacón, Julio César.** *Aplicación de la metodología RUP para el estándar J2EE.*

BIBLIOGRAFÍA

39. Loja. [Online] Enero 2008. [Cited: Enero 15, 2012.] <http://www.utpl.edu.ec/eva/descargas/material/175/G18401.8.pdf>.
40. **Pressman, Roger S.** *Ingeniería del Software*. 2005. 970-10-5473-3.
41. **Semanat Aldana, Edmis Deivis and Verdecia Four, Leonor.** *Sistema de Video Vigilancia*. La Habana : s.n., 2009.
42. **Larman, Craig.** *UML y Patrones, Introducción al análisis y diseño orientado a objeto*. Mexico : Prentice Hall, 1999.
43. **Jacobson, Ivar, Booch, Grady and Rumbaugh, James.** *El Proceso Unificado de Desarrollo de Software*. Madrid : Perason Educacion, 2000. 84-7829-063-2.
44. **Pressman, Roger S.** *Ingeniería del Software, Un enfoque Práctico*. Sexta Edición. s.l. : McGraw-Hill, 2007. p. 438.
45. **Fernández Lanvin, Daniel.** *Definición de una arquitectura software para el diseño de aplicaciones web basadas en tecnología Java-J2EE*. 20217190Y.
46. **Teniente López, Ernest, et al.** *Diseño de sistemas software en UML*. Barcelona : ETSAB. 84-8301-724-5 .
47. **Llorente, César de la Torre.** *Guía de Arquitectura N-Capas orientada al Dominio con .NET 4.0*. 2010.

GLOSARIO DE TÉRMINOS

GLOSARIO DE TÉRMINOS

API (Application Programming Interface): Interfaz para programación de aplicaciones.

AJAX (Asynchronous JavaScript and XML): JavaScript Asíncrono y XML.

Bits: Unidad de medida de información equivalente a la elección entre dos posibilidades igualmente probables.

Biblioteca (del inglés library): Colección o conjunto de subprogramas usados para desarrollar software.

CSS (Cascading Style Sheets): Es el lenguaje de hojas de estilo, con él se puede separar el aspecto visual del contenido las páginas web.

Datagramas: Es la estructura interna de un paquete de datos que se transfiere en una conexión.

Framework: Marco de trabajo o conjunto de bibliotecas orientadas a la reutilización a gran escala de componentes de software para el desarrollo rápido de aplicaciones.

HTML(HyperText Markup Language): Es el lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto.

JSON (JavaScript Object Notation): Formato ligero de intercambio de datos.

Híbridos: Todo aquello que es producto de elementos de distinta naturaleza.

IDE (Integrated Development Environment): Entorno de Desarrollo Integrado.

RIAs: Aplicaciones de Internet Enriquecidas.

Motor de Render: Es el encargado de generar o dibujar los componentes de la librería y generar las imágenes a partir de un modelo dado.

Sitio web: Es una colección de páginas web relacionadas y comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.

Software: Es todo el conjunto intangible de datos y programas almacenados en un computador.

Transmisión analógica: Consiste en el envío de información en forma de ondas, a través de un medio de transmisión físico.

XML (eXtensible Markup Language): Metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C).